Overview

Introduction

Modelling parallel systems

Linear Time Properties

Regular Properties

Linear Temporal Logic (LTL)

- syntax and semantics of LTL
- automata-based LTL model checking ←
- complexity of LTL model checking

Computation-Tree Logic

Equivalences and Abstraction

LTL model checking problem

LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
(without terminal states)
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold **?**

LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
(without terminal states)
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold ?

basic idea: try to refute $\mathcal{T} \models \varphi$

LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
(without terminal states)
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold **?**

basic idea: try to refute $\mathcal{T} \models \varphi$ by searching for a path π in $\mathcal{T}$ s.t.

$$\pi \not\models \varphi$$

LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
(without terminal states)
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold **?**

basic idea: try to refute $\mathcal{T} \models \varphi$ by searching for a path π in $\mathcal{T}$ s.t.

$$\pi \not\models \varphi, \text{ i.e., } \pi \models \neg\varphi$$

The LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold ?

1. construct an **NBA** $\mathcal{A}$ for $Words(\neg\varphi)$

The LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold ?

1. construct an **NBA** $\mathcal{A}$ for $Words(\neg\varphi)$
2. search a path π in $\mathcal{T}$ with
$$trace(\pi) \in Words(\neg\varphi)$$

The LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
LTL-formula φ over AP

question: does $\mathcal{T} \models \varphi$ hold ?

1. construct an **NBA** $\mathcal{A}$ for $Words(\neg\varphi)$
2. search a path π in $\mathcal{T}$ with

$$trace(\pi) \in Words(\neg\varphi) = \mathcal{L}_\omega(\mathcal{A})$$

The LTL model checking problem

given: finite transition system $\mathcal{T}$ over AP
LTL-formula φ over AP

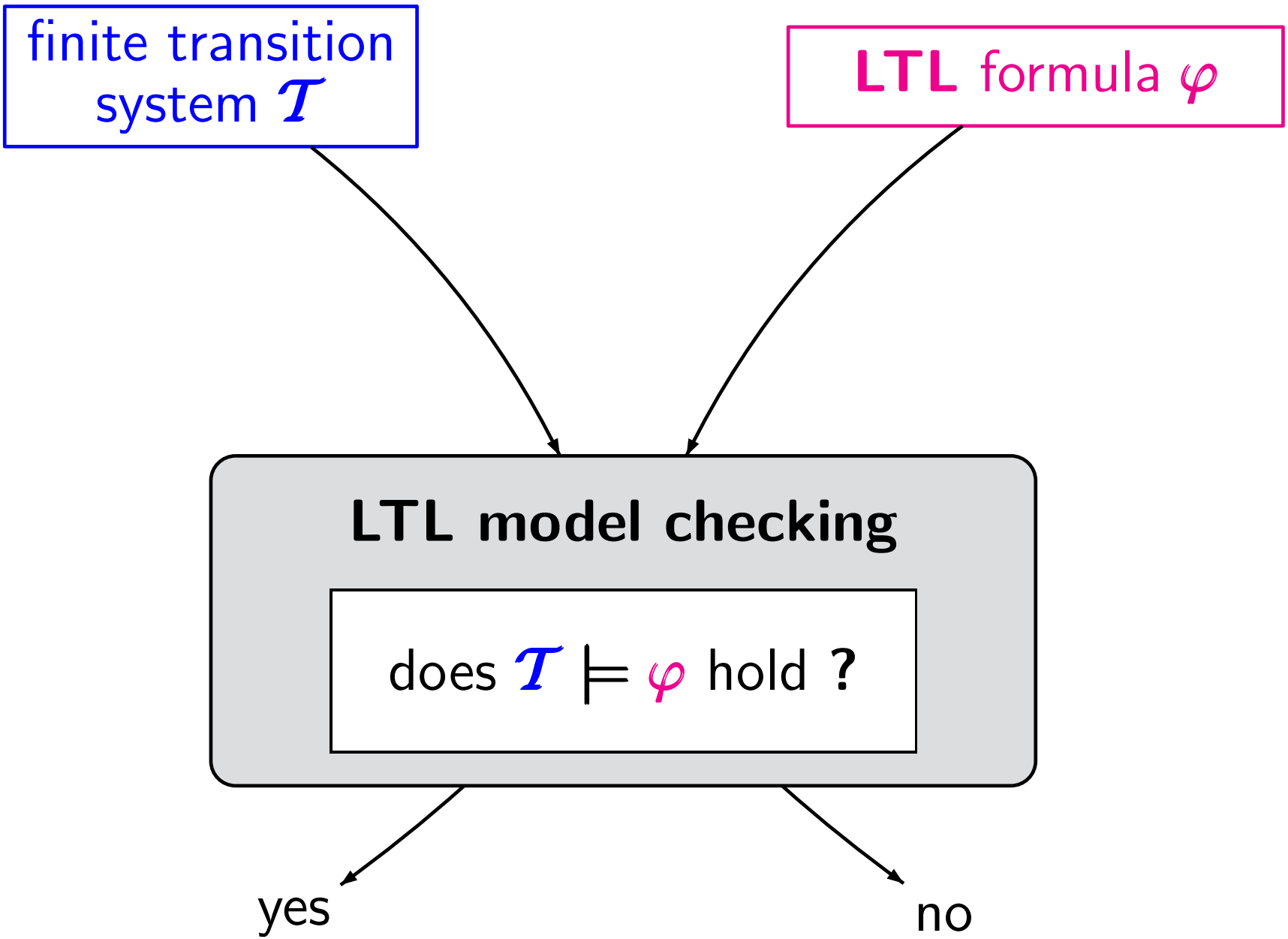
question: does $\mathcal{T} \models \varphi$ hold ?

1. construct an **NBA** $\mathcal{A}$ for $Words(\neg\varphi)$
2. search a path π in $\mathcal{T}$ with

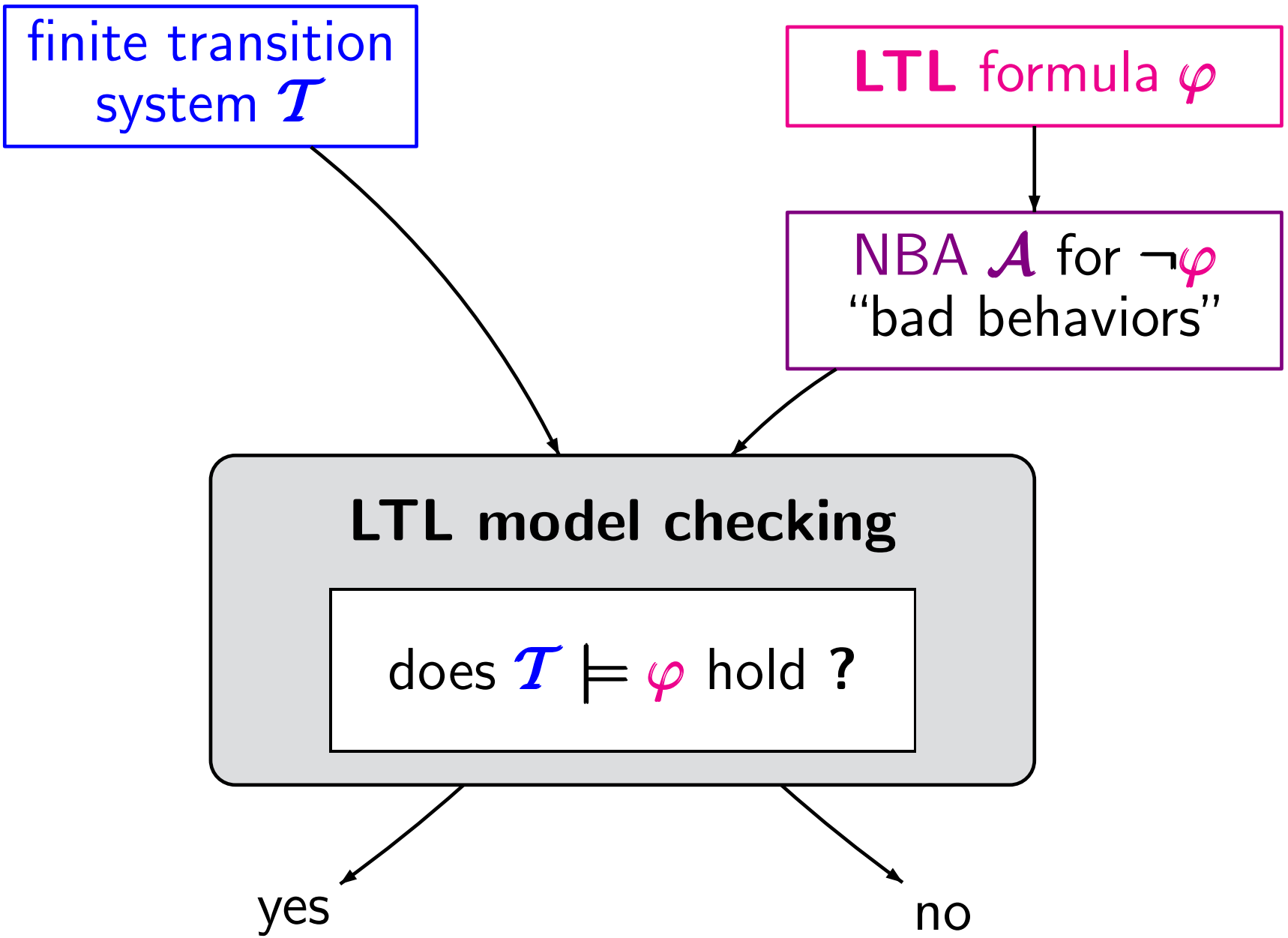
$$trace(\pi) \in Words(\neg\varphi) = \mathcal{L}_\omega(\mathcal{A})$$

construct the product-TS $\mathcal{T} \otimes \mathcal{A}$
search a path in the product that meets the acceptance condition of $\mathcal{A}$

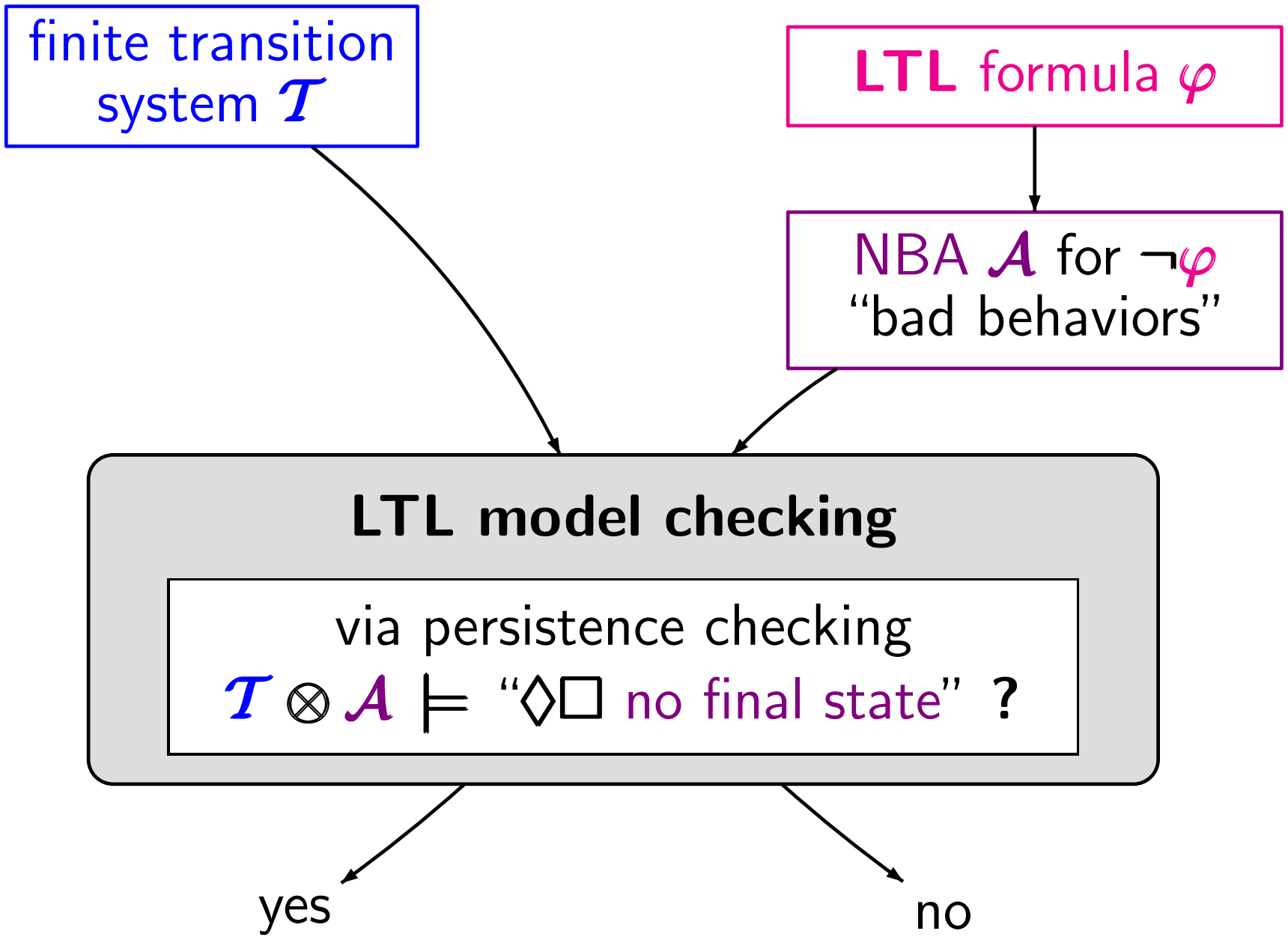
Automata-based LTL model checking



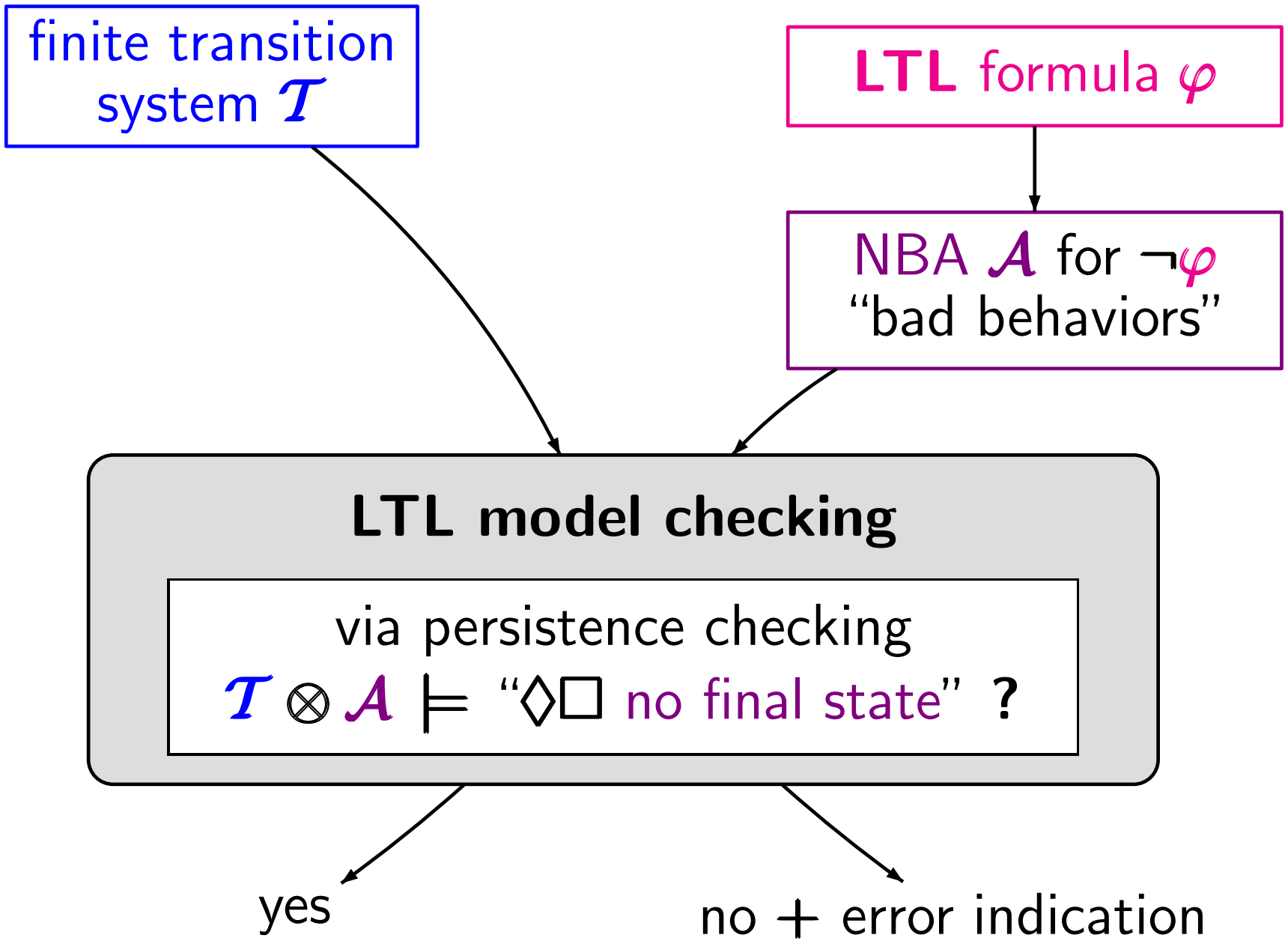
Automata-based LTL model checking



Automata-based LTL model checking



Automata-based LTL model checking



Safety and LTL model checking

Safety and LTL model checking

safety property E	LTL-formula φ

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	$\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	$\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$
invariant checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$?	

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	$\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$
invariant checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$?	persistence checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$?

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	$\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$
invariant checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$?	persistence checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$?
error indication: $\widehat{\pi} \in \mathit{Paths}_{\mathit{fin}}(\mathcal{T})$ s.t. $\mathit{trace}(\widehat{\pi}) \in \mathcal{L}(\mathcal{A})$	

Safety and LTL model checking

safety property E	LTL-formula φ
NFA for the bad prefixes for E $\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$	**NBA** for the "bad behaviors" $\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\neg\varphi)$
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$	$\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$
invariant checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$?	persistence checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$?
error indication: $\widehat{\pi} \in \mathit{Paths}_{\mathit{fin}}(\mathcal{T})$ s.t. $\mathit{trace}(\widehat{\pi}) \in \mathcal{L}(\mathcal{A})$	error indication: prefix of a path π s.t. $\mathit{trace}(\pi) \in \mathcal{L}_\omega(\mathcal{A})$

Safety vs LTL model checking

Safety vs LTL model checking

$\mathcal{T} \models$ safety property E

iff $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$

where $\mathcal{A}$ is an NFA for the bad prefixes

$\mathcal{T} \models$ LTL-formula φ

iff $\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$

where $\mathcal{A}$ is an NBA for $\neg\varphi$

Safety vs LTL model checking

$\mathcal{T} \models$ safety property E

iff $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$

iff there is no path fragment $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \ldots \langle s_n, q_n\rangle$ in $\mathcal{T} \otimes \mathcal{A}$ s. t. $q_n \in F$

$\mathcal{T} \models$ LTL-formula φ

iff $\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$

iff there is no path $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \langle s_2, q_2\rangle \ldots$ in $\mathcal{T} \otimes \mathcal{A}$ s.t. $q_i \in F$ for infinitely many $i \in \mathbb{N}$

Safety vs LTL model checking

$\mathcal{T} \models$ safety property E

iff $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$

iff there is no path fragment $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \ldots \langle s_n, q_n\rangle$ in $\mathcal{T} \otimes \mathcal{A}$ s. t. $q_n \in F$

iff $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$

$\mathcal{T} \models$ LTL-formula φ

iff $\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$

iff there is no path $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \langle s_2, q_2\rangle \ldots$ in $\mathcal{T} \otimes \mathcal{A}$ s.t. $q_i \in F$ for infinitely many $i \in \mathbb{N}$

iff $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

Safety vs LTL model checking

$\mathcal{T} \models$ safety property E

iff $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \varnothing$

iff there is no path fragment $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \ldots \langle s_n, q_n\rangle$ in $\mathcal{T} \otimes \mathcal{A}$ s. t. $q_n \in F$

iff $\mathcal{T} \otimes \mathcal{A} \models \Box\neg F$ $\longleftarrow$ invariant checking

---

$\mathcal{T} \models$ LTL-formula φ

iff $\mathit{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \varnothing$

iff there is no path $\langle s_0, q_0\rangle \langle s_1, q_1\rangle \langle s_2, q_2\rangle \ldots$ in $\mathcal{T} \otimes \mathcal{A}$ s.t. $q_i \in F$ for infinitely many $i \in \mathbb{N}$

iff $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$ $\longleftarrow$ persistence checking

Recall: nondeterministic Büchi automata

NBA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$

- Q finite set of states
- Σ alphabet
- $\delta : Q \times \Sigma \to 2^Q$ transition relation
- $Q_0 \subseteq Q$ set of initial states
- $F \subseteq Q$ set of final states, also called accept states

Recall: nondeterministic Büchi automata

NBA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$

- Q finite set of states
- Σ alphabet
- $\delta : Q \times \Sigma \to 2^Q$ transition relation
- $Q_0 \subseteq Q$ set of initial states
- $F \subseteq Q$ set of final states, also called accept states

run for a word $A_0 A_1 A_2 \ldots \in \Sigma^\omega$:

state sequence $\pi = q_0 q_1 q_2 \ldots$ where $q_0 \in Q_0$ and $q_{i+1} \in \delta(q_i, A_i)$ for $i \geq 0$

run π is accepting if $\overset{\infty}{\exists}\, i \in \mathbb{N}.\ q_i \in F$

Recall: nondeterministic Büchi automata

NBA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$

- Q finite set of states
- Σ alphabet
- $\delta : Q \times \Sigma \to 2^Q$ transition relation
- $Q_0 \subseteq Q$ set of initial states
- $F \subseteq Q$ set of final states, also called accept states

accepted language $\mathcal{L}_\omega(\mathcal{A}) \subseteq \Sigma^\omega$ is given by:

$$\mathcal{L}_\omega(\mathcal{A}) \stackrel{\text{def}}{=} \text{set of infinite words over } \Sigma \text{ that have an accepting run in } \mathcal{A}$$

Recall: nondeterministic Büchi automata

NBA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$

- Q finite set of states
- Σ alphabet $\longleftarrow$ here: $\Sigma = 2^{AP}$
- $\delta : Q \times \Sigma \to 2^Q$ transition relation
- $Q_0 \subseteq Q$ set of initial states
- $F \subseteq Q$ set of final states, also called accept states

accepted language $\mathcal{L}_\omega(\mathcal{A}) \subseteq \Sigma^\omega$ is given by:

$$\mathcal{L}_\omega(\mathcal{A}) \stackrel{\text{def}}{=} \text{set of infinite words over } \Sigma \text{ that have an accepting run in } \mathcal{A}$$

From LTL to NBA

From LTL to NBA

For each **LTL** formula φ over AP there is an **NBA** $\mathcal{A}$ over the alphabet 2^{AP} such that

$$Words(\varphi) = \mathcal{L}_\omega(\mathcal{A})$$

From LTL to NBA

For each **LTL** formula φ over AP there is an **NBA** $\mathcal{A}$ over the alphabet 2^{AP} such that

- $Words(\varphi) = \mathcal{L}_\omega(\mathcal{A})$
- $size(\mathcal{A}) = \mathcal{O}(\exp(|\varphi|))$

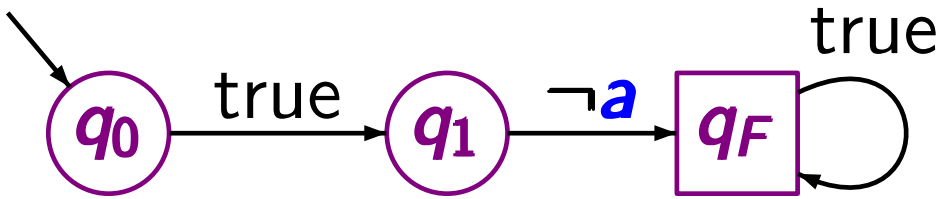
From LTL to NBA

For each **LTL** formula φ over AP there is an **NBA** $\mathcal{A}$ over the alphabet 2^{AP} such that

- $Words(\varphi) = \mathcal{L}_\omega(\mathcal{A})$
- $size(\mathcal{A}) = \mathcal{O}(\exp(|\varphi|))$

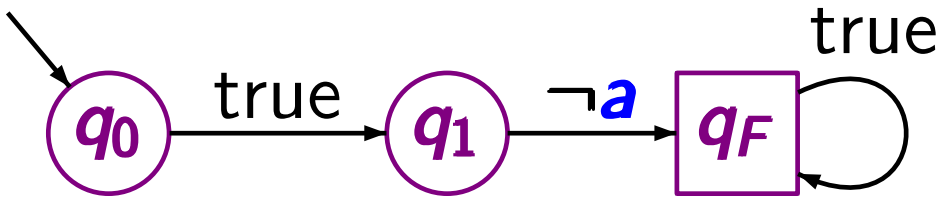
proof: ... later ...

NBA for LTL formulas



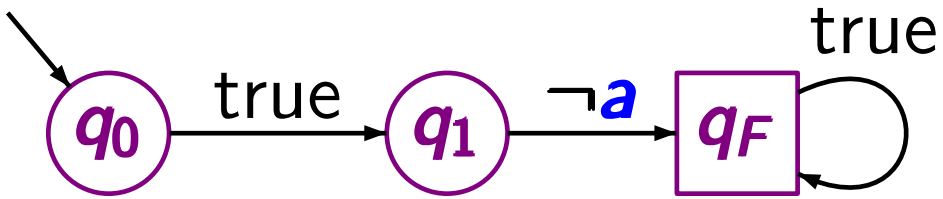
$$\mathcal{L}_\omega(\mathcal{A}) = \,?$$

NBA for LTL formulas

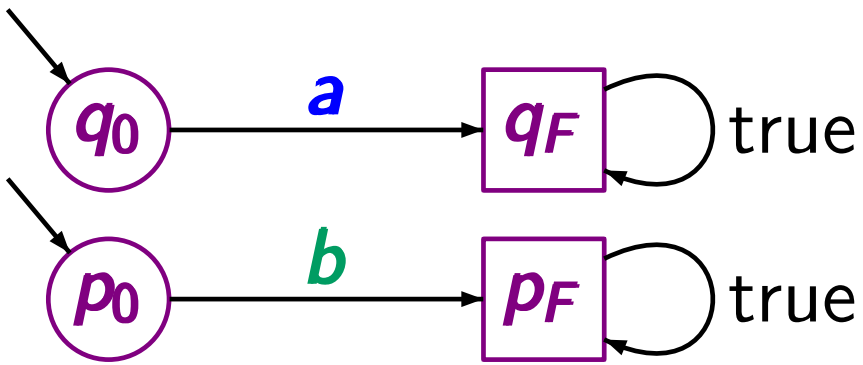


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\bigcirc \neg a)$$

NBA for LTL formulas

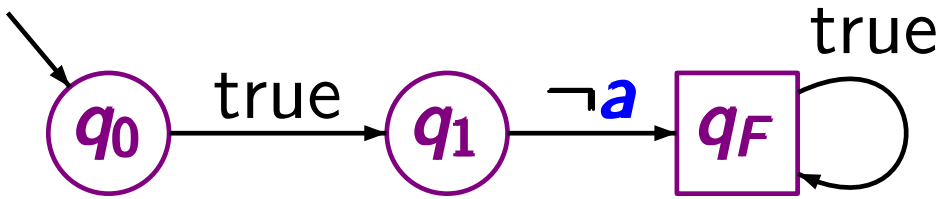


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\bigcirc \neg a)$$

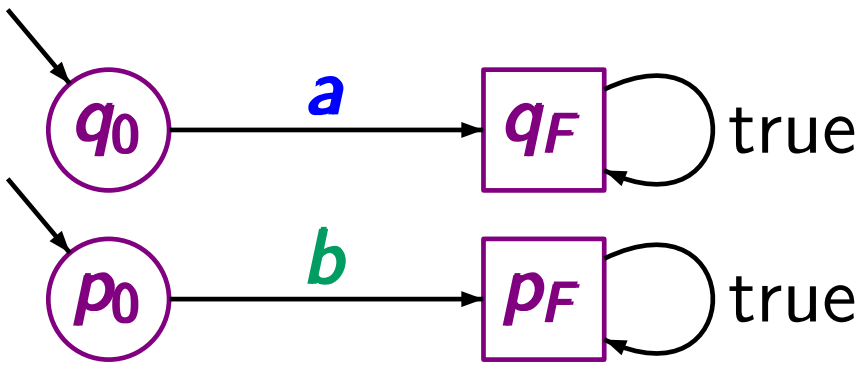


$$\mathcal{L}_\omega(\mathcal{A}) = \,?$$

NBA for LTL formulas

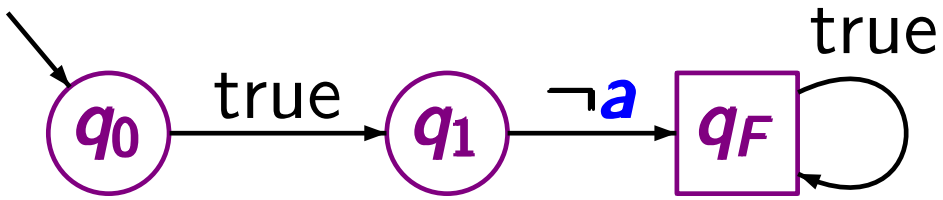


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\bigcirc \neg a)$$

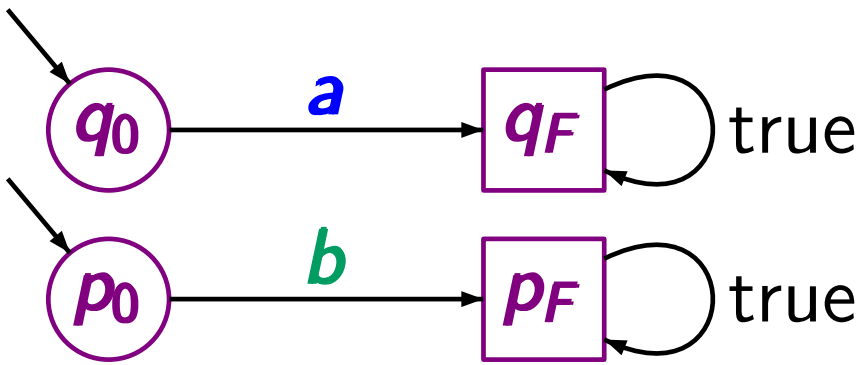


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(a \vee b)$$

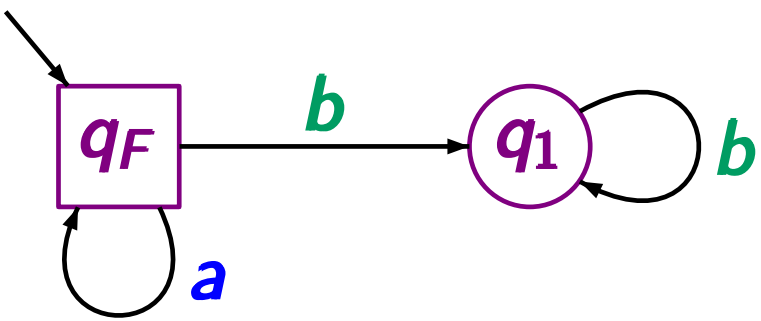
NBA for LTL formulas



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\bigcirc \neg a)$$

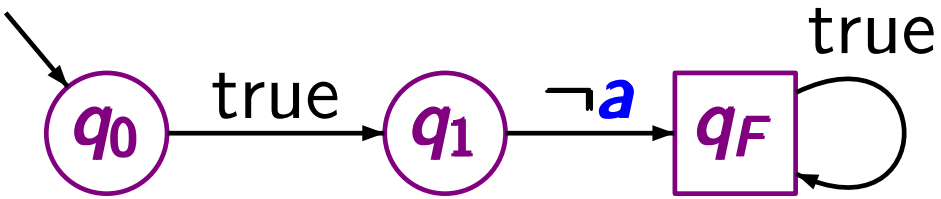


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(a \vee b)$$

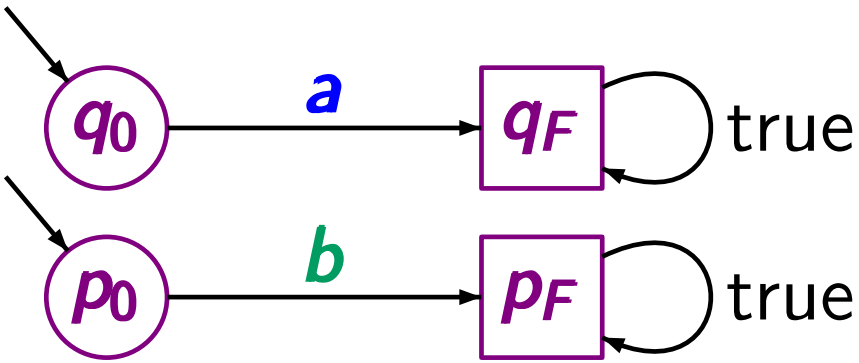


$$\mathcal{L}_\omega(\mathcal{A}) = ?$$

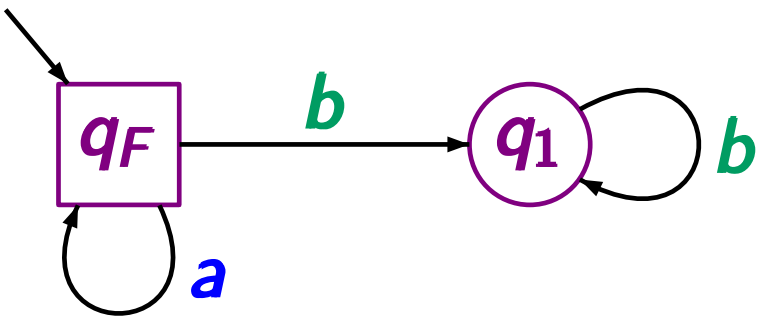
NBA for LTL formulas



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\bigcirc \neg a)$$

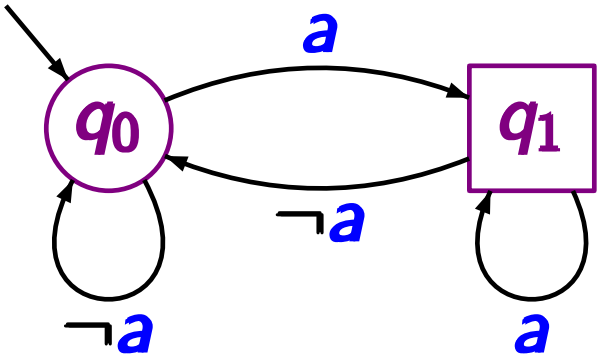


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(a \vee b)$$



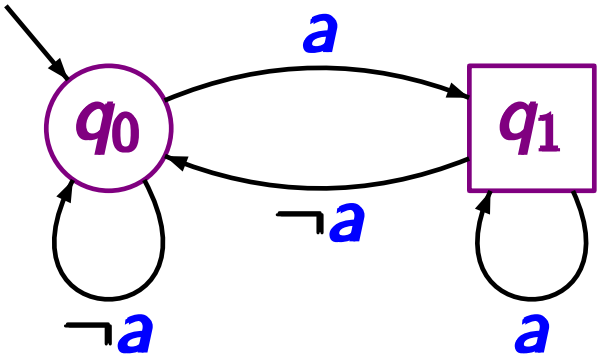
$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box a)$$

NBA for LTL formulas



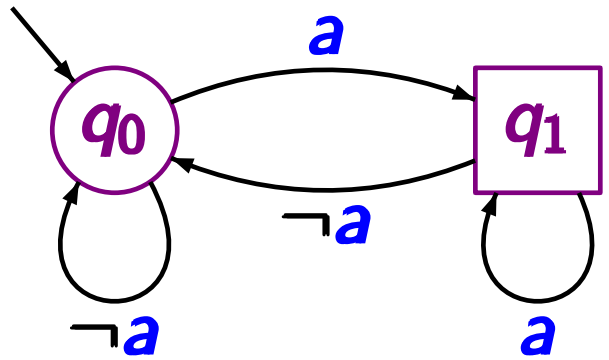
$$\mathcal{L}_\omega(\mathcal{A}) = ?$$

NBA for LTL formulas

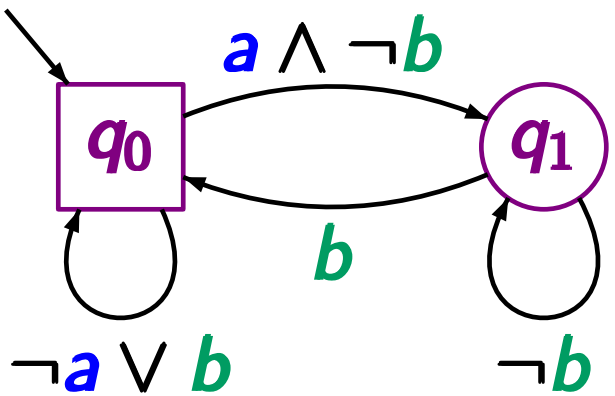


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box\Diamond a)$$

NBA for LTL formulas

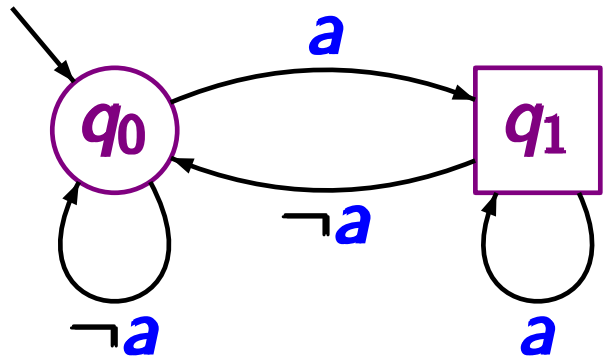


$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box\Diamond a)$$

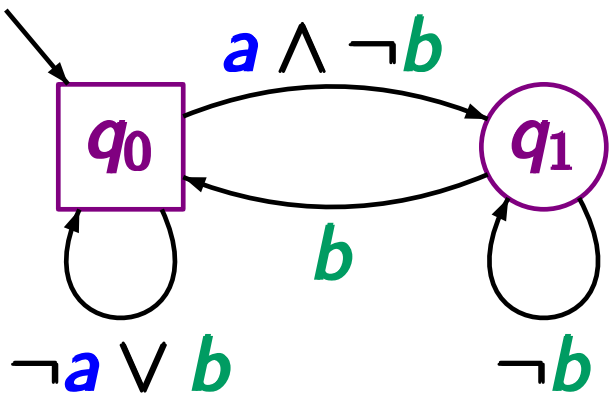


$$\mathcal{L}_\omega(\mathcal{A}) = ?$$

NBA for LTL formulas



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box\Diamond a)$$



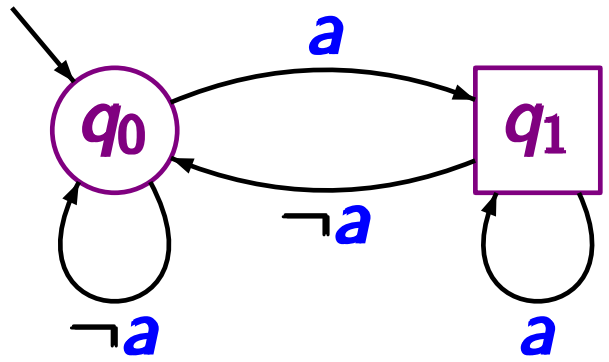
$$\mathcal{L}_\omega(\mathcal{A}) = \,?$$

e.g., $\varnothing\,\varnothing\,\varnothing\,\varnothing\ldots = \varnothing^\omega$

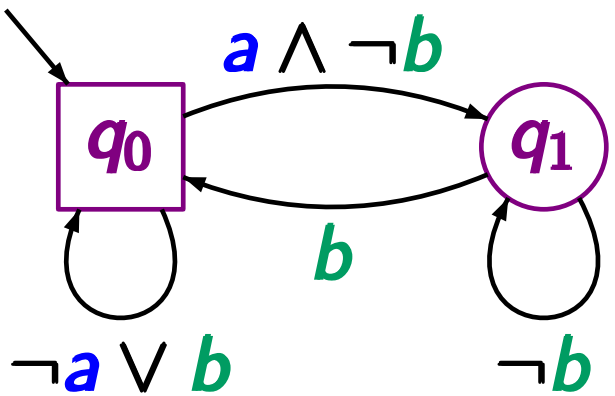
$(\{a\}\,\{b\})^\omega$

} are accepted by $\mathcal{A}$

NBA for LTL formulas



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box\Diamond a)$$



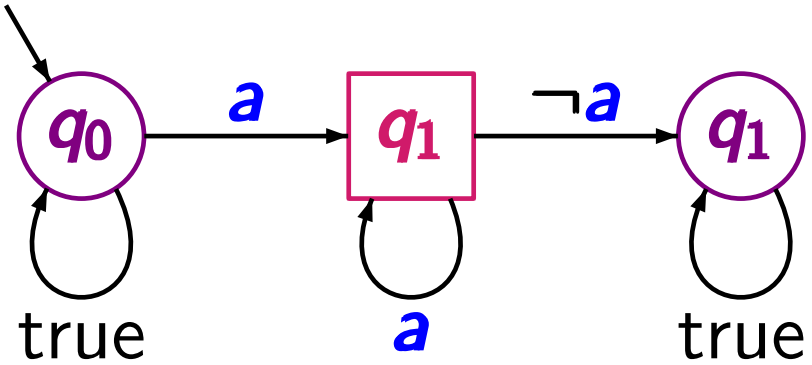
$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Box(a \rightarrow \Diamond b))$$

e.g., $\varnothing\,\varnothing\,\varnothing\,\varnothing\ldots = \varnothing^\omega$

$(\{a\}\,\{b\})^\omega$

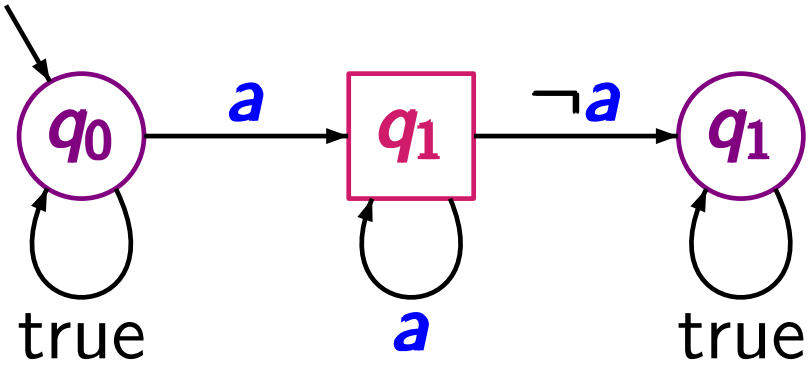
} are accepted by $\mathcal{A}$

NBA for LTL formula



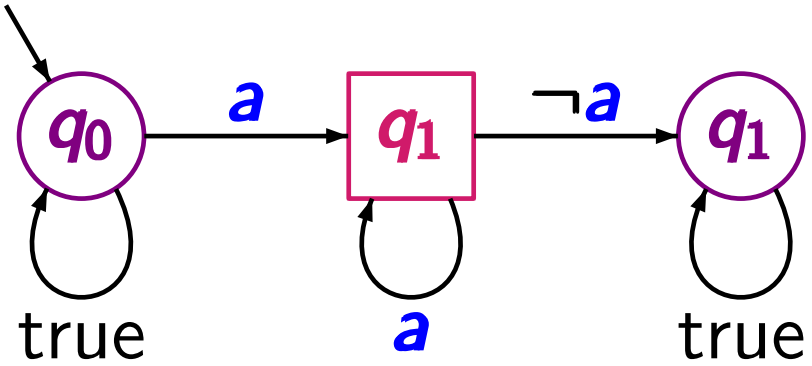
$$\mathcal{L}_\omega(\mathcal{A}) = \,?$$

NBA for LTL formula



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Diamond\Box a)$$

NBA for LTL formula



$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\Diamond\Box a)$$

possible runs for $\{a\}^\omega$

$q_0\ q_0\ q_0\ q_0\ q_0\ q_0\ \ldots$	not accepting
$q_0\ q_1\ q_1\ q_1\ q_1\ q_1\ \ldots$	accepting
$q_0\ q_0\ q_1\ q_1\ q_1\ q_1\ \ldots$	accepting
$q_0\ q_0\ q_0\ q_1\ q_1\ q_1\ \ldots$	accepting
$\vdots$	

NFA and NBA for safety properties

NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E.

NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

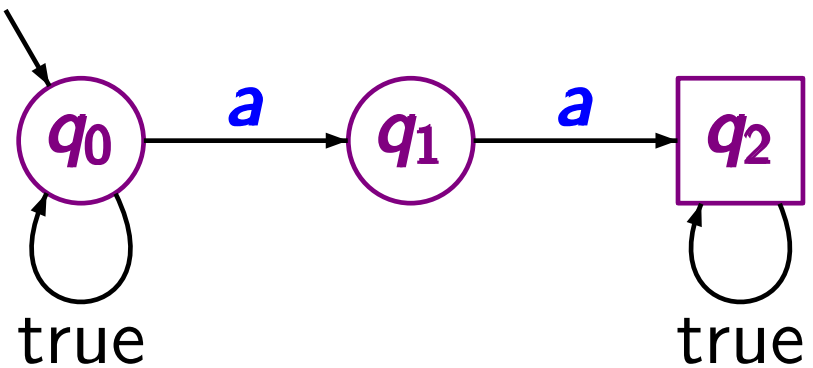
$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E$$

NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E$$

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"

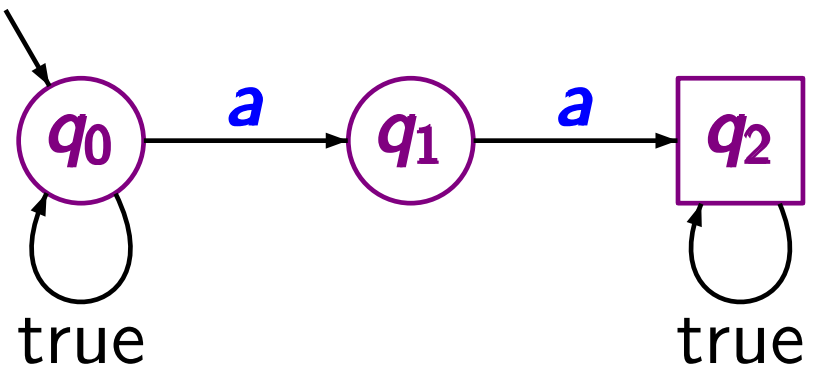


NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E = \mathit{Words}(\neg\varphi)$$

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"



$$\varphi = \Box(a \rightarrow \bigcirc\neg a)$$

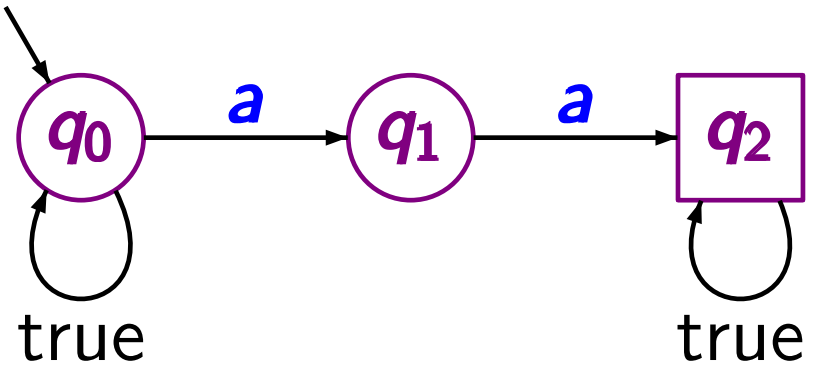
NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E = \mathit{Words}(\neg\varphi)$$

wrong, if $\mathcal{L}(\mathcal{A})$ = language of minimal bad prefixes

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"



$$\varphi = \Box(a \rightarrow \bigcirc\neg a)$$

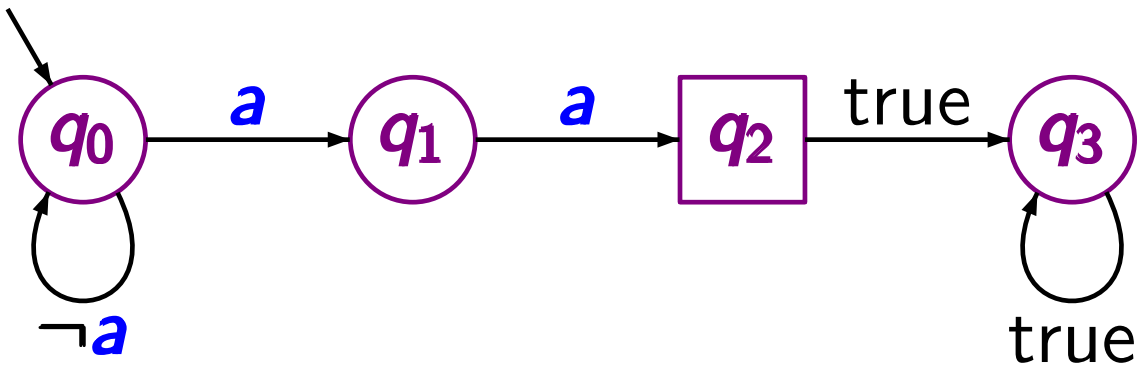
NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E = \mathit{Words}(\neg\varphi)$$

wrong, if $\mathcal{L}(\mathcal{A})$ = language of minimal bad prefixes

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"



$$\mathcal{L}_\omega(\mathcal{A}) = \varnothing$$

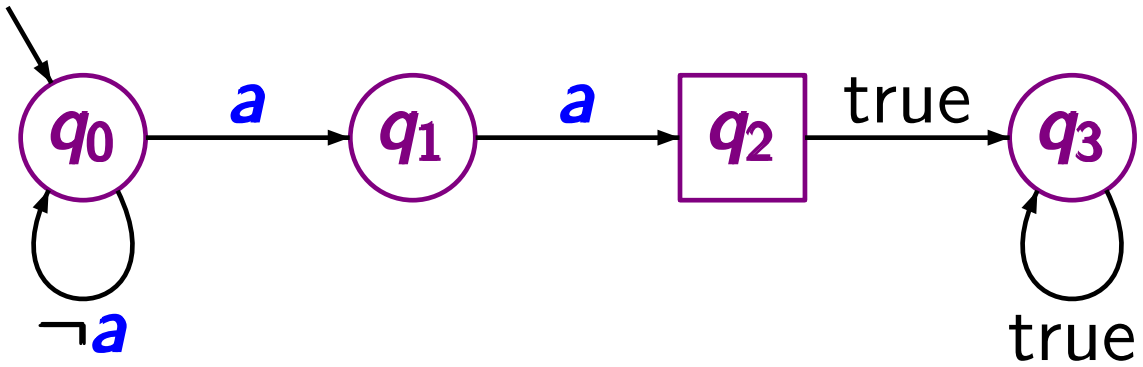
NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E = \mathit{Words}(\neg\varphi)$$

wrong, if $\mathcal{L}(\mathcal{A})$ = language of minimal bad prefixes
even if $\mathcal{A}$ is a non-blocking DFA

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"



$$\mathcal{L}_\omega(\mathcal{A}) = \varnothing$$

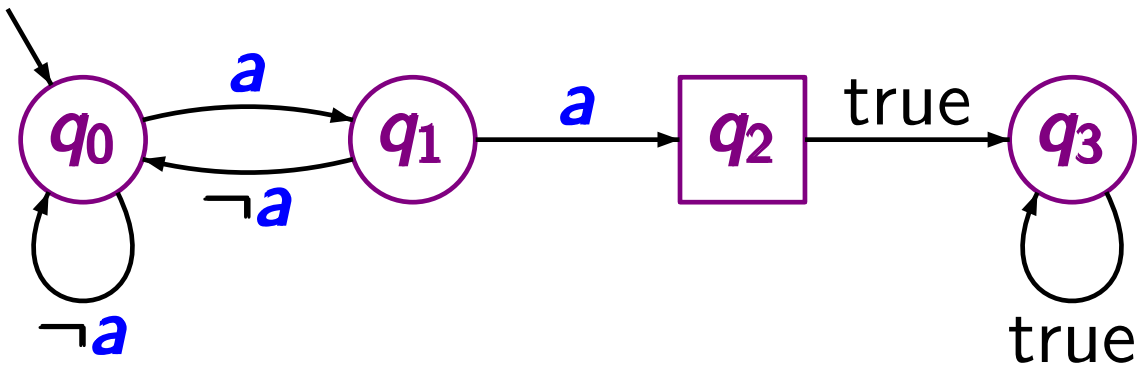
NFA and NBA for safety properties

Let $\mathcal{A}$ be an **NFA** for the language of all bad prefixes for a safety property E. Then:

$$\mathcal{L}_\omega(\mathcal{A}) = \overline{E} = (2^{AP})^\omega \setminus E = \mathit{Words}(\neg\varphi)$$

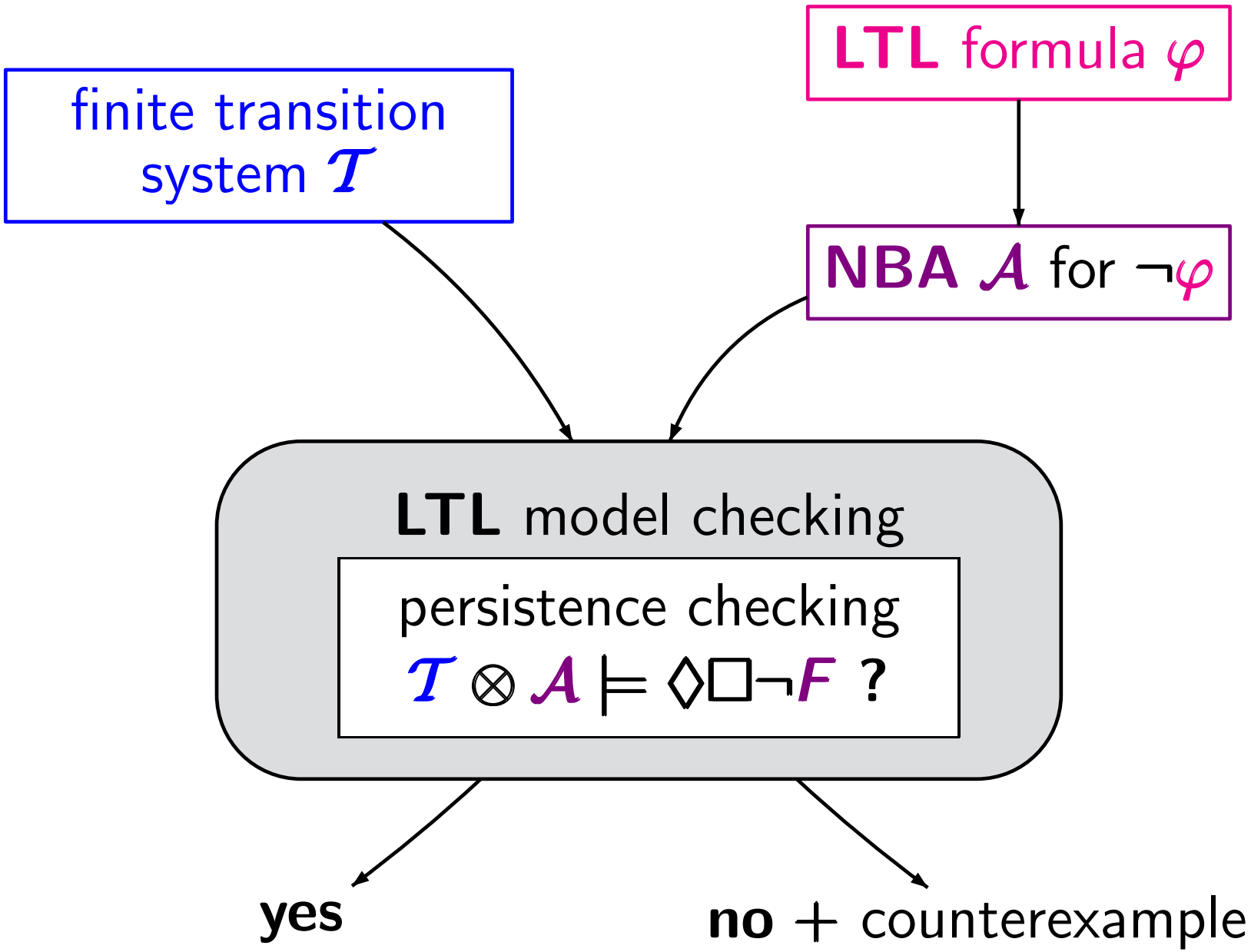
wrong, if $\mathcal{L}(\mathcal{A})$ = language of minimal bad prefixes
even if $\mathcal{A}$ is a non-blocking DFA

Example: $E \mathrel{\widehat{=}}$ "never a twice in a row"

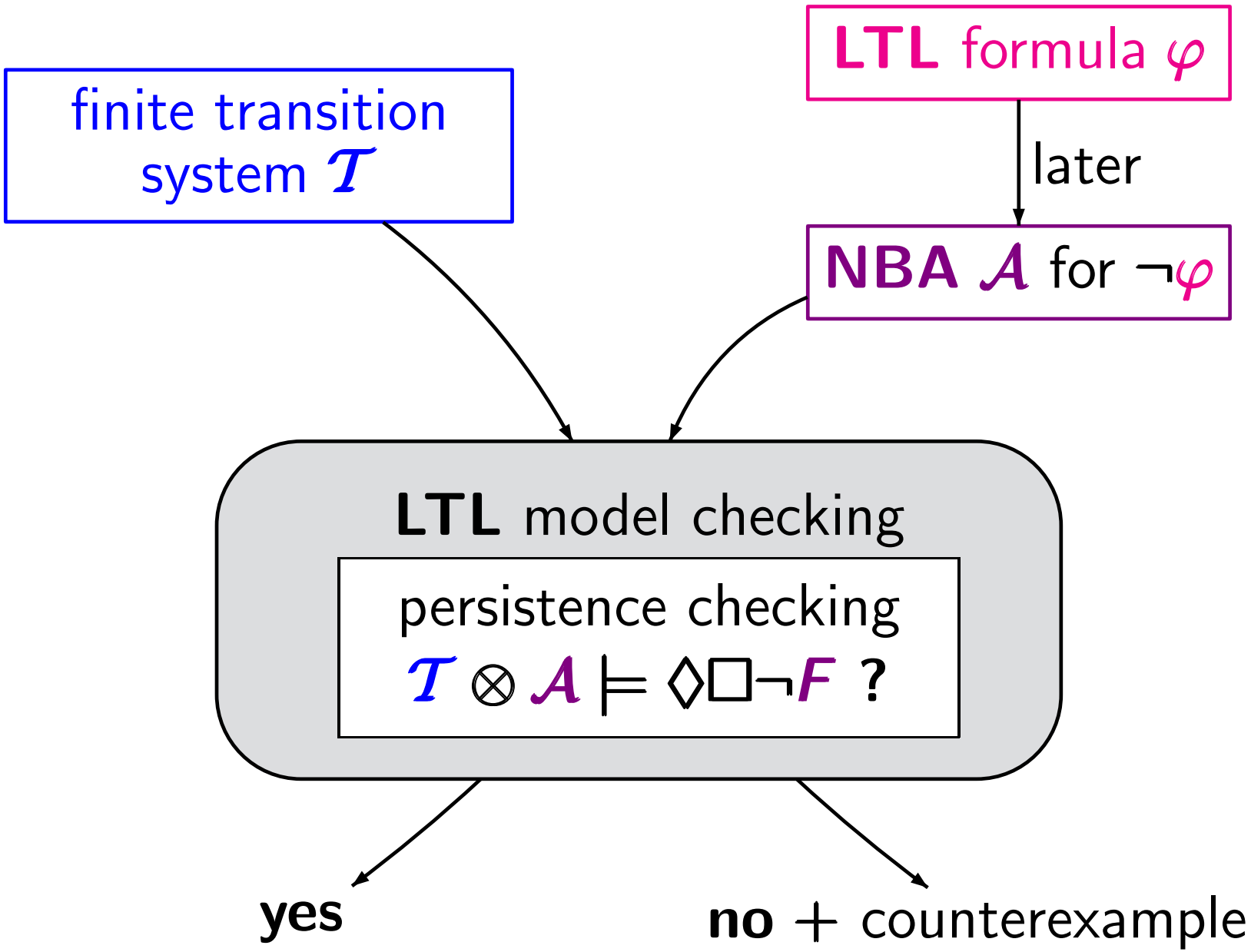


$$\mathcal{L}_\omega(\mathcal{A}) = \varnothing$$

LTL model checking



LTL model checking



Recall: product transition system

$\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ TS without terminal states

$\mathcal{A} = (Q, 2^{AP}, \delta, Q_0, F)$ NBA or NFA

non-blocking, $Q_0 \cap F = \varnothing$

Recall: product transition system

$\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ TS without terminal states

$\mathcal{A} = (Q, 2^{AP}, \delta, Q_0, F)$ NBA or NFA

non-blocking, $Q_0 \cap F = \varnothing$

product-TS $\mathcal{T} \otimes \mathcal{A} \stackrel{\text{def}}{=} (S \times Q, Act, \rightarrow', S_0', AP', L')$

Recall: product transition system

$\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ TS without terminal states

$\mathcal{A} = (Q, 2^{AP}, \delta, Q_0, F)$ NBA or NFA

non-blocking, $Q_0 \cap F = \varnothing$

product-TS $\mathcal{T} \otimes \mathcal{A} \stackrel{\text{def}}{=} (S \times Q, Act, \rightarrow', S_0', AP', L')$

initial states: $S_0' = \{\langle s_0, q \rangle : s_0 \in S_0, q \in \delta(Q_0, L(s_0))\}$

labeling: $AP' = Q$, $L'(\langle s, q \rangle) = \{q\}$

Recall: product transition system

$\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ TS without terminal states

$\mathcal{A} = (Q, 2^{AP}, \delta, Q_0, F)$ NBA or NFA

non-blocking, $Q_0 \cap F = \varnothing$

product-TS $\mathcal{T} \otimes \mathcal{A} \stackrel{\text{def}}{=} (S \times Q, Act, \rightarrow', S_0', AP', L')$

initial states: $S_0' = \{\langle s_0, q \rangle : s_0 \in S_0, q \in \delta(Q_0, L(s_0))\}$

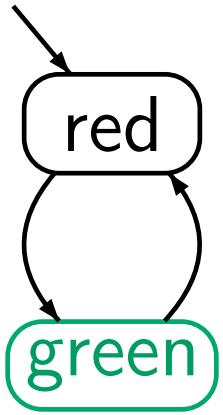
labeling: $AP' = Q$, $L'(\langle s, q \rangle) = \{q\}$

transition relation:

$$\frac{s \xrightarrow{\alpha} s' \ \wedge \ q' \in \delta(q, L(s'))}{\langle s, q \rangle \xrightarrow{\alpha}' \langle s', q' \rangle}$$

Example: LTL model checking

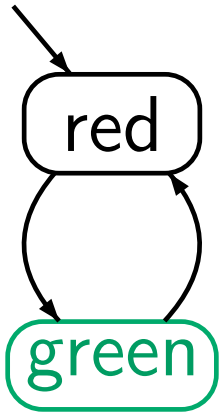
TS $\mathcal{T}$



LTL formula $\varphi = \Box\Diamond$green

Example: LTL model checking

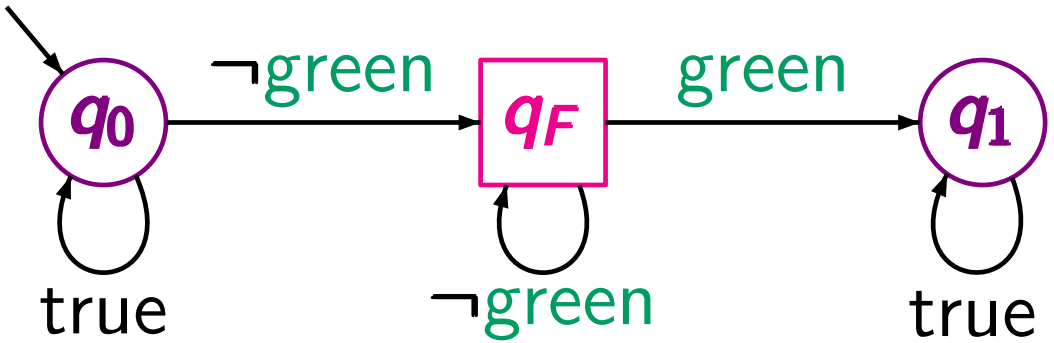
TS $\mathcal{T}$



LTL formula $\varphi = \Box\Diamond$green

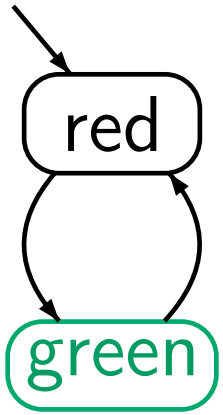
NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg$green



Example: LTL model checking

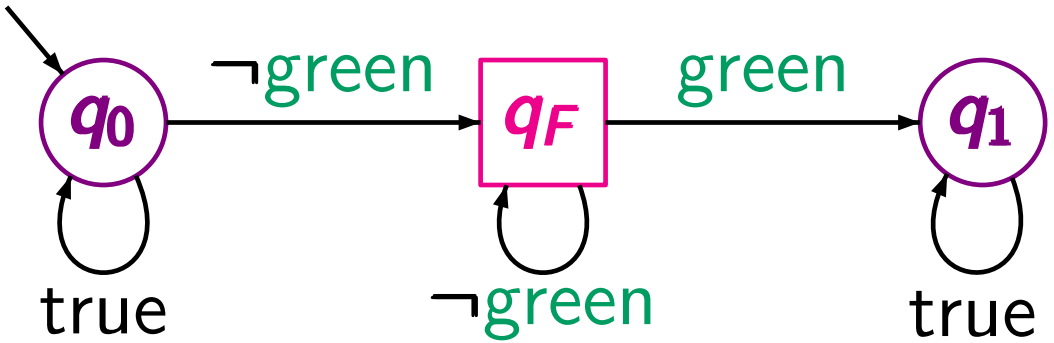
TS $\mathcal{T}$

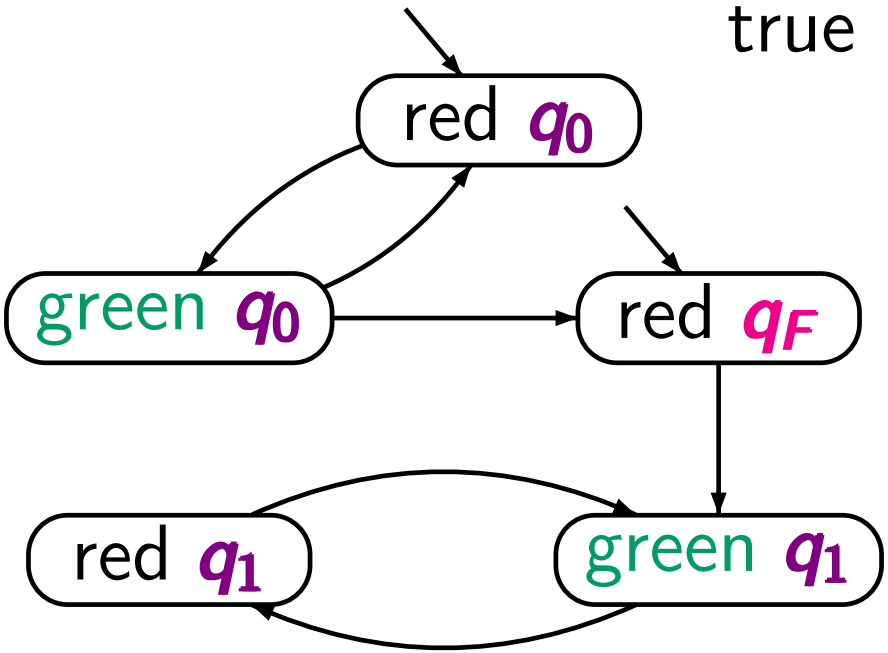


LTL formula $\varphi = \Box\Diamond$green

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg$green

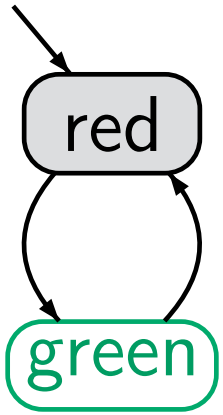




reachable fragment of the product TS $\mathcal{T} \otimes \mathcal{A}$

Example: LTL model checking

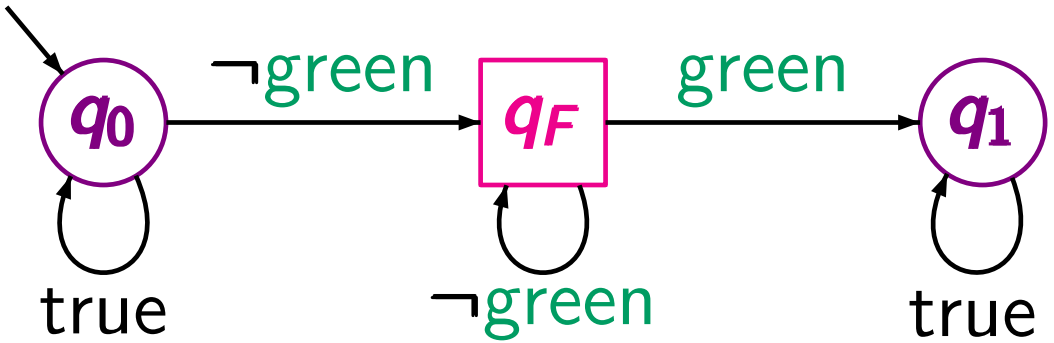
TS $\mathcal{T}$

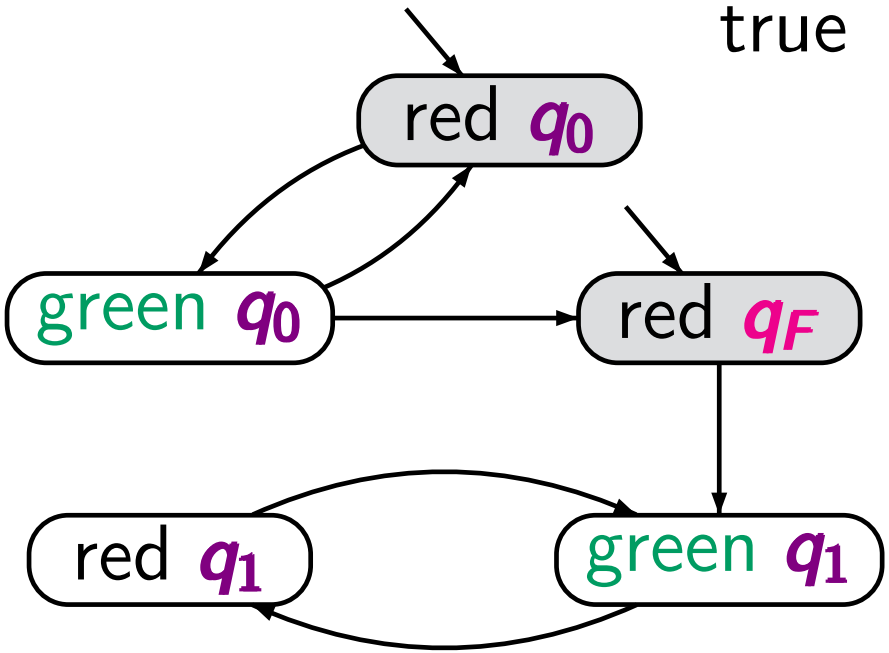


LTL formula $\varphi = \Box\Diamond$green

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg$green





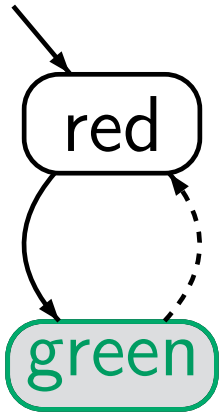
initial states:

$\langle \text{red}, q \rangle$ where

$$q \in \delta(q_0, L(\text{red})) = \delta(q_0, \varnothing) = \{q_0, q_F\}$$

Example: LTL model checking

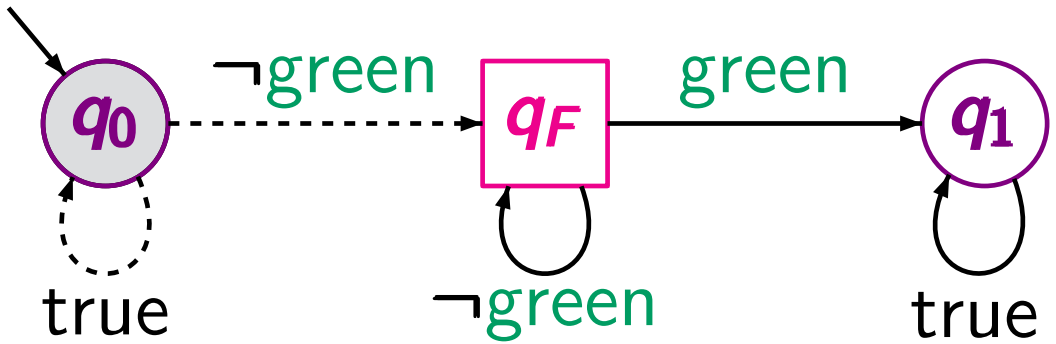
TS $\mathcal{T}$

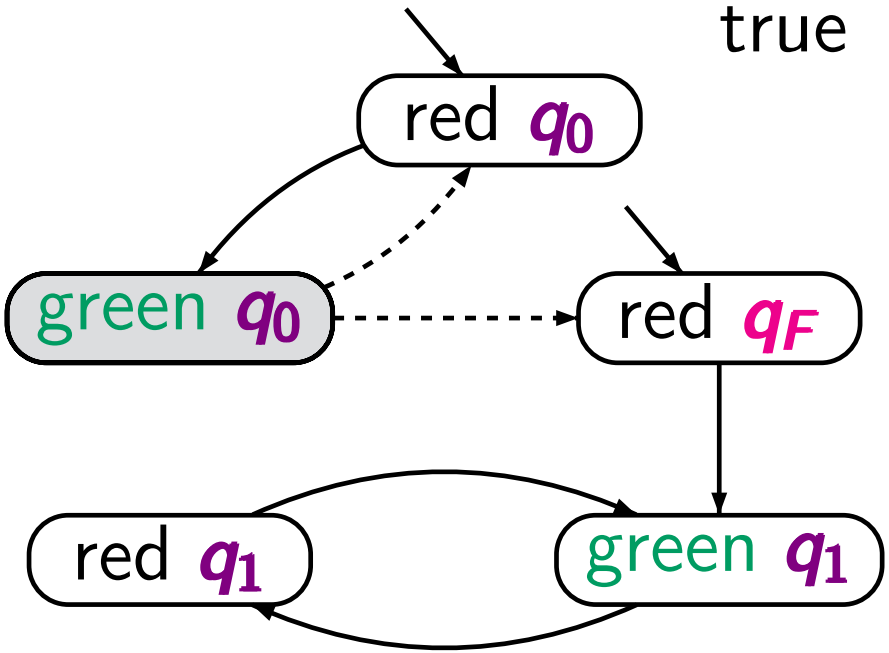


LTL formula $\varphi = \Box\Diamond\text{green}$

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg\text{green}$





transition

$\langle \text{green}, q_0 \rangle \rightarrow \langle \text{red}, q \rangle$

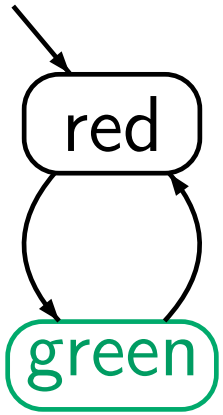
$q \in \delta(q_0, L(\text{red}))$

$= \delta(q_0, \varnothing)$

$= \{q_0, q_F\}$

Example: LTL model checking

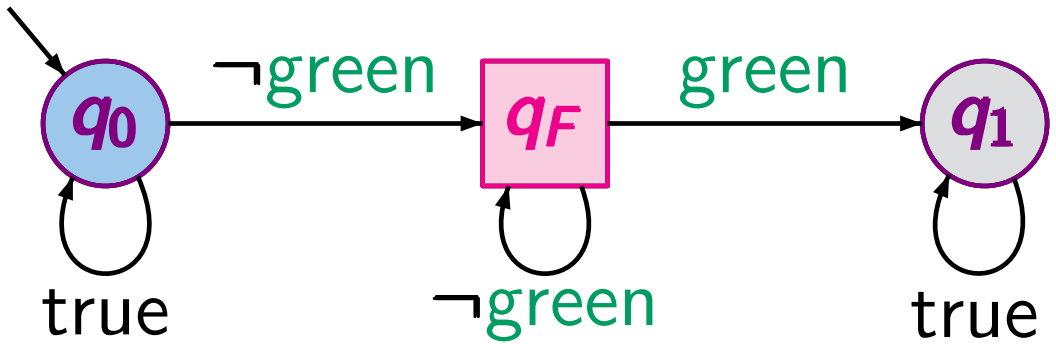
TS $\mathcal{T}$

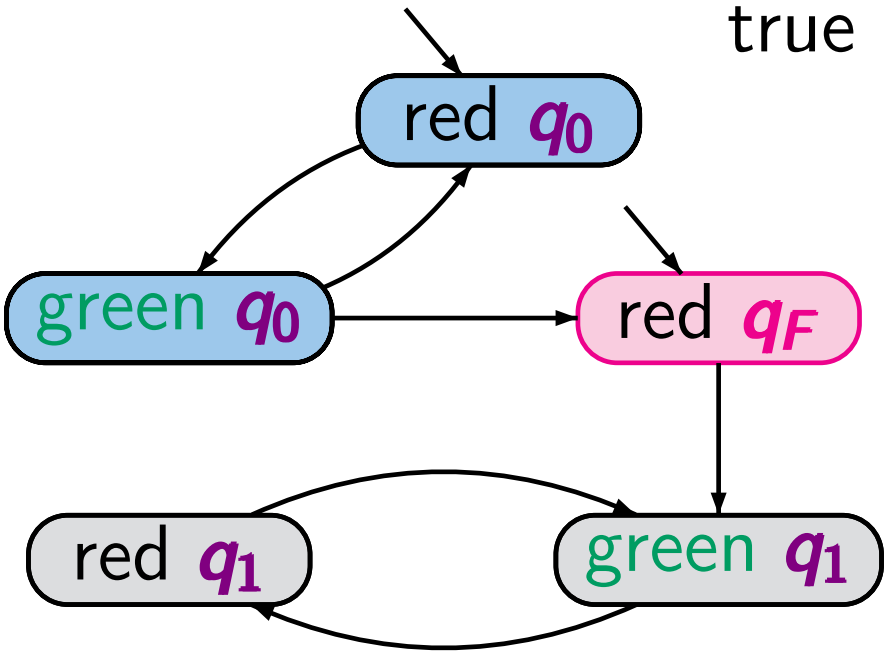


LTL formula $\varphi = \Box\Diamond\text{green}$

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg\text{green}$





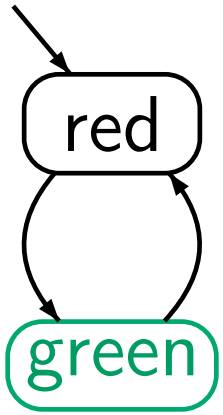
atomic propositions

$AP' = \{q_0, q_F, q_1\}$

obvious labeling function

Example: LTL model checking

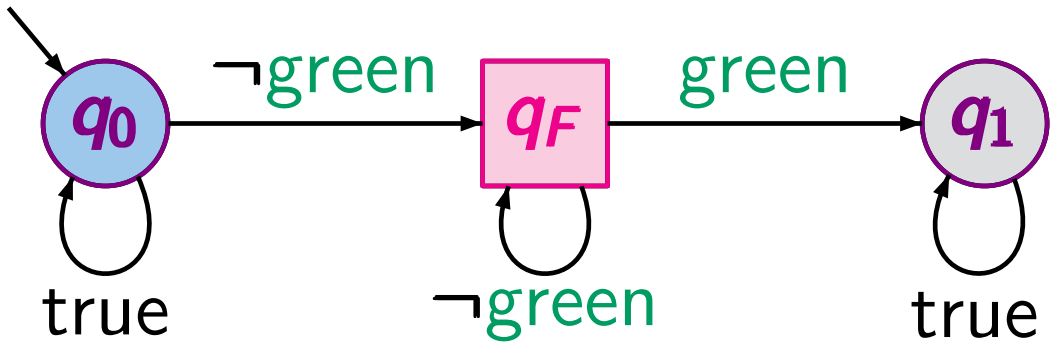
TS $\mathcal{T}$

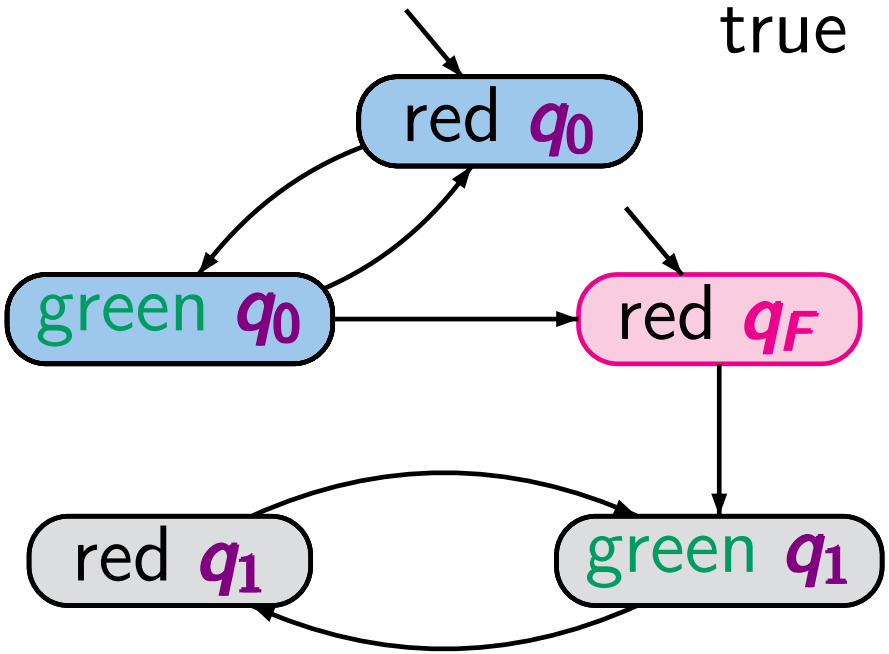


LTL formula $\varphi = \Box\Diamond\text{green}$

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg\text{green}$

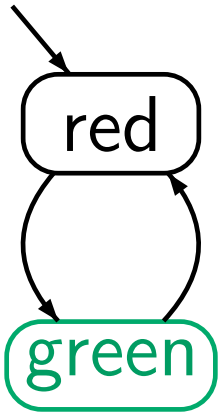




$\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

Example: LTL model checking

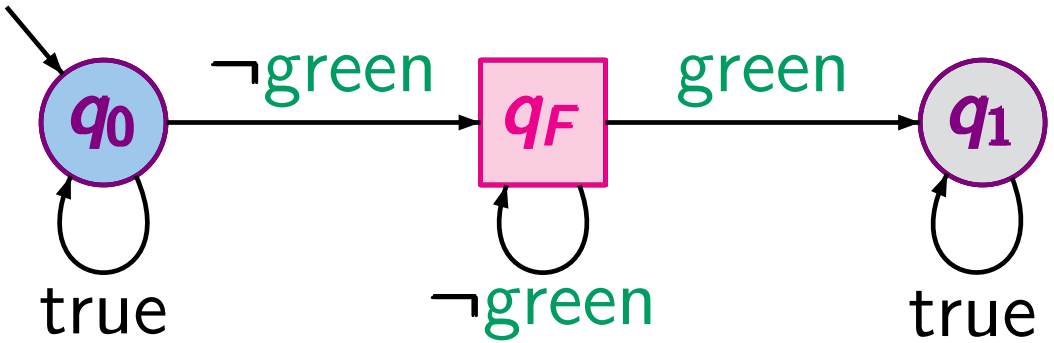
TS $\mathcal{T}$

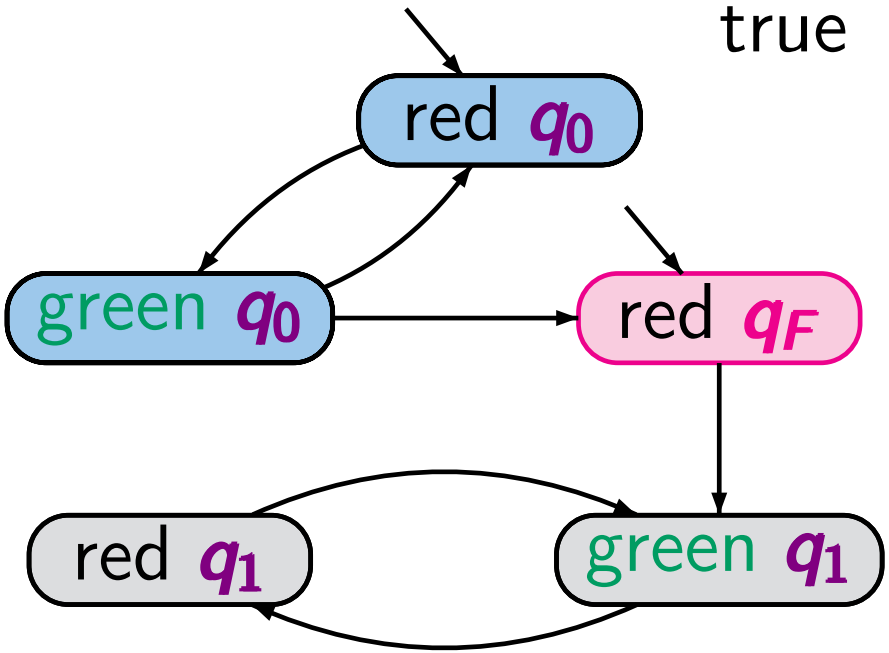


LTL formula $\varphi = \Box\Diamond$green

NBA $\mathcal{A}$ for the complement

$\neg\varphi \equiv \Diamond\Box\neg$green

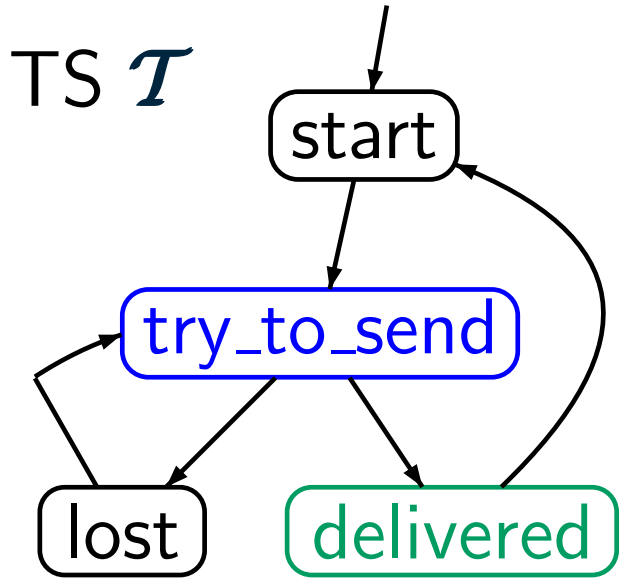




$\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

hence: $\mathcal{T} \models \varphi$

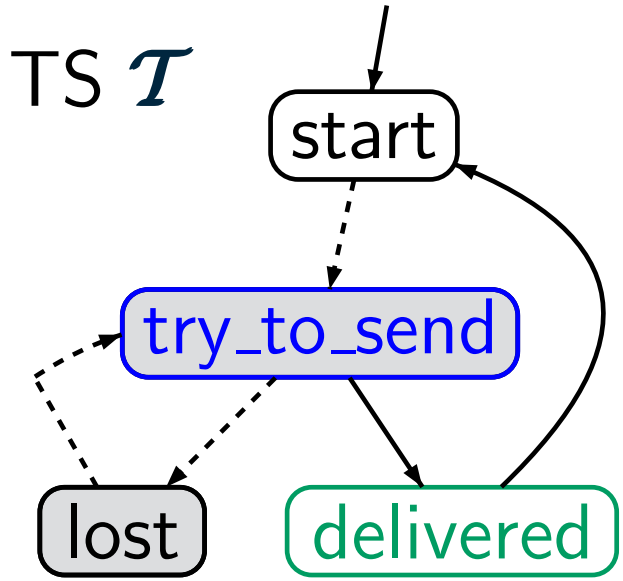
Example: LTL model checking



LTL formula $\varphi = \Box(\textit{try} \rightarrow \Diamond \textit{del})$

"each (repeatedly) sent message will eventually be delivered"

Example: LTL model checking

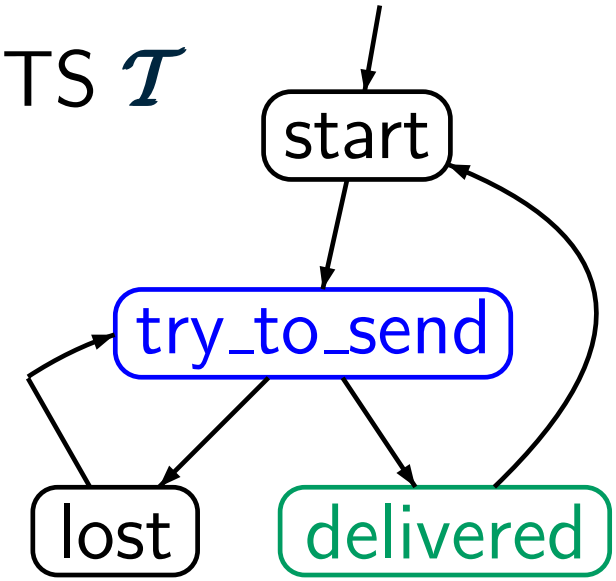


LTL formula $\varphi = \Box(\mathit{try} \rightarrow \Diamond \mathit{del})$

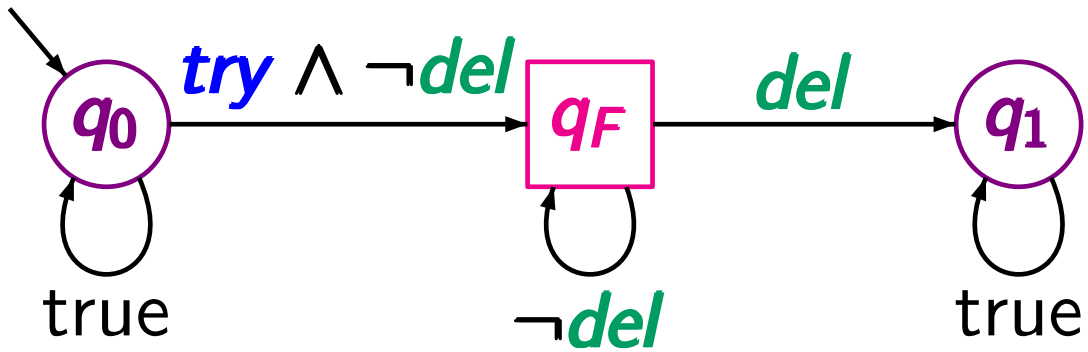
"each (repeatedly) sent message will eventually be delivered"

$\mathcal{T} \not\models \varphi$

Example: LTL model checking



NBA $\mathcal{A}$ for $\neg\varphi \equiv \Diamond(\mathit{try} \wedge \Box\neg\mathit{del})$

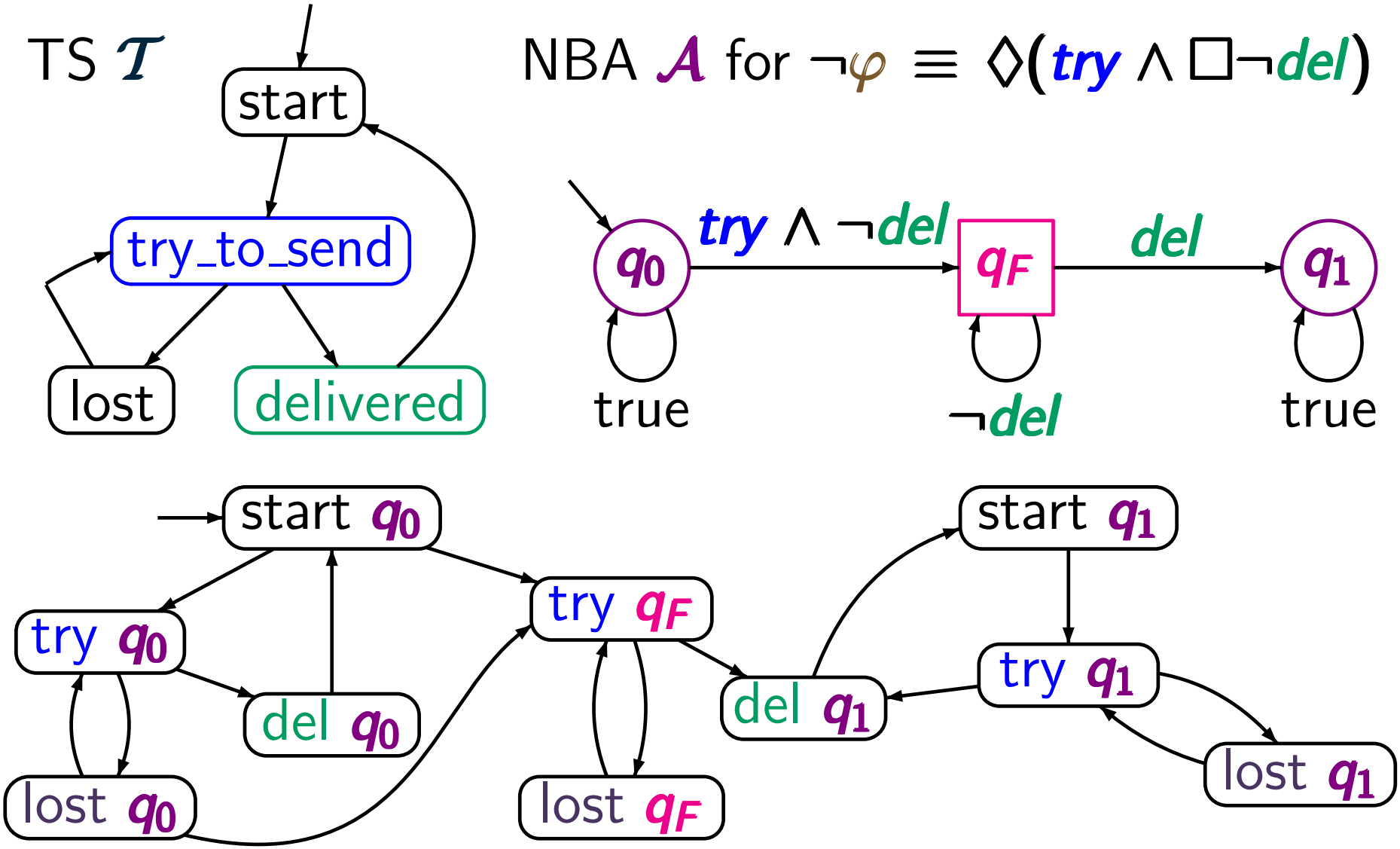


LTL formula $\varphi = \Box(\mathit{try} \rightarrow \Diamond\mathit{del})$

"each (repeatedly) sent message will eventually be delivered"

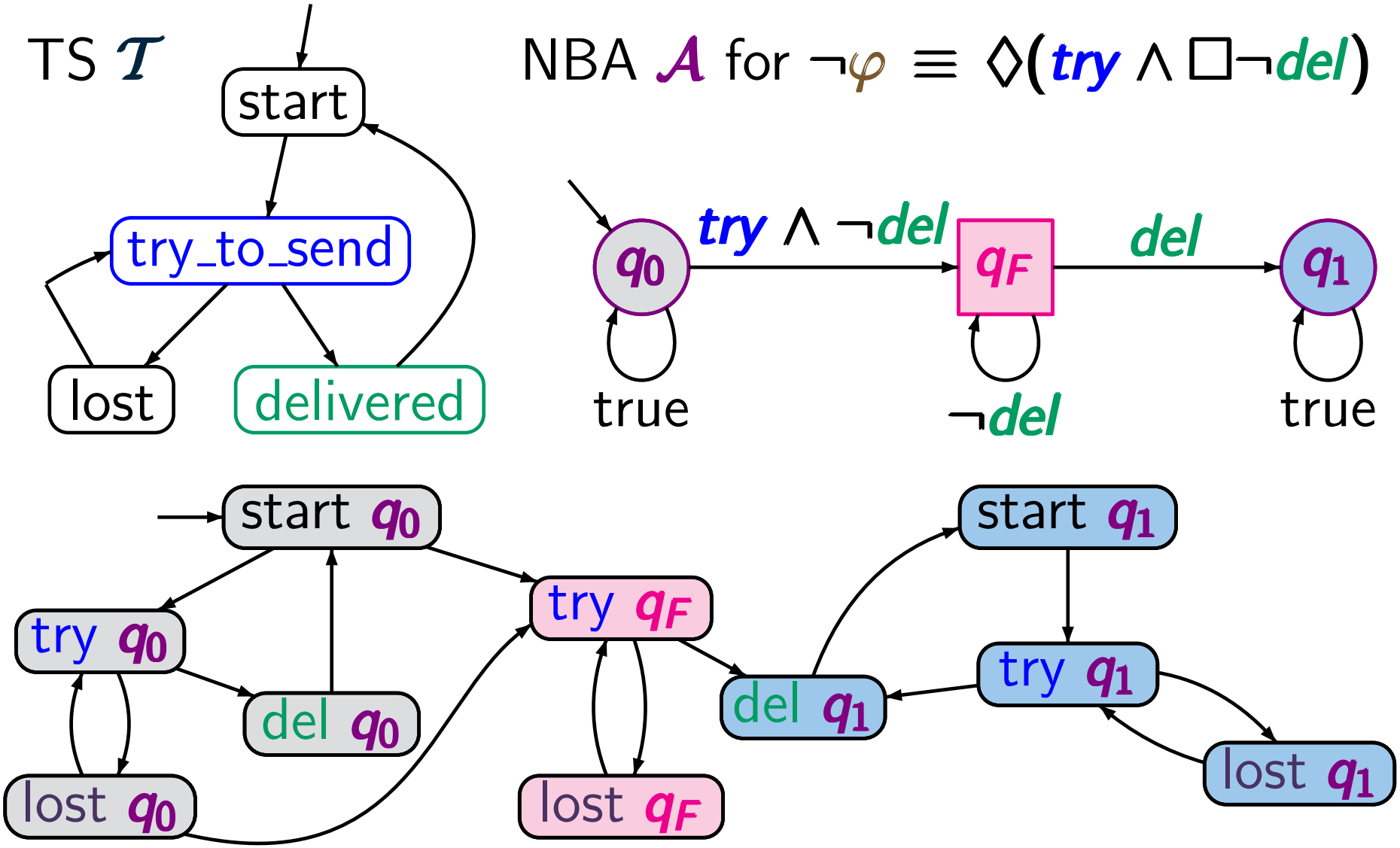
$\mathcal{T} \not\models \varphi$

Example: LTL model checking



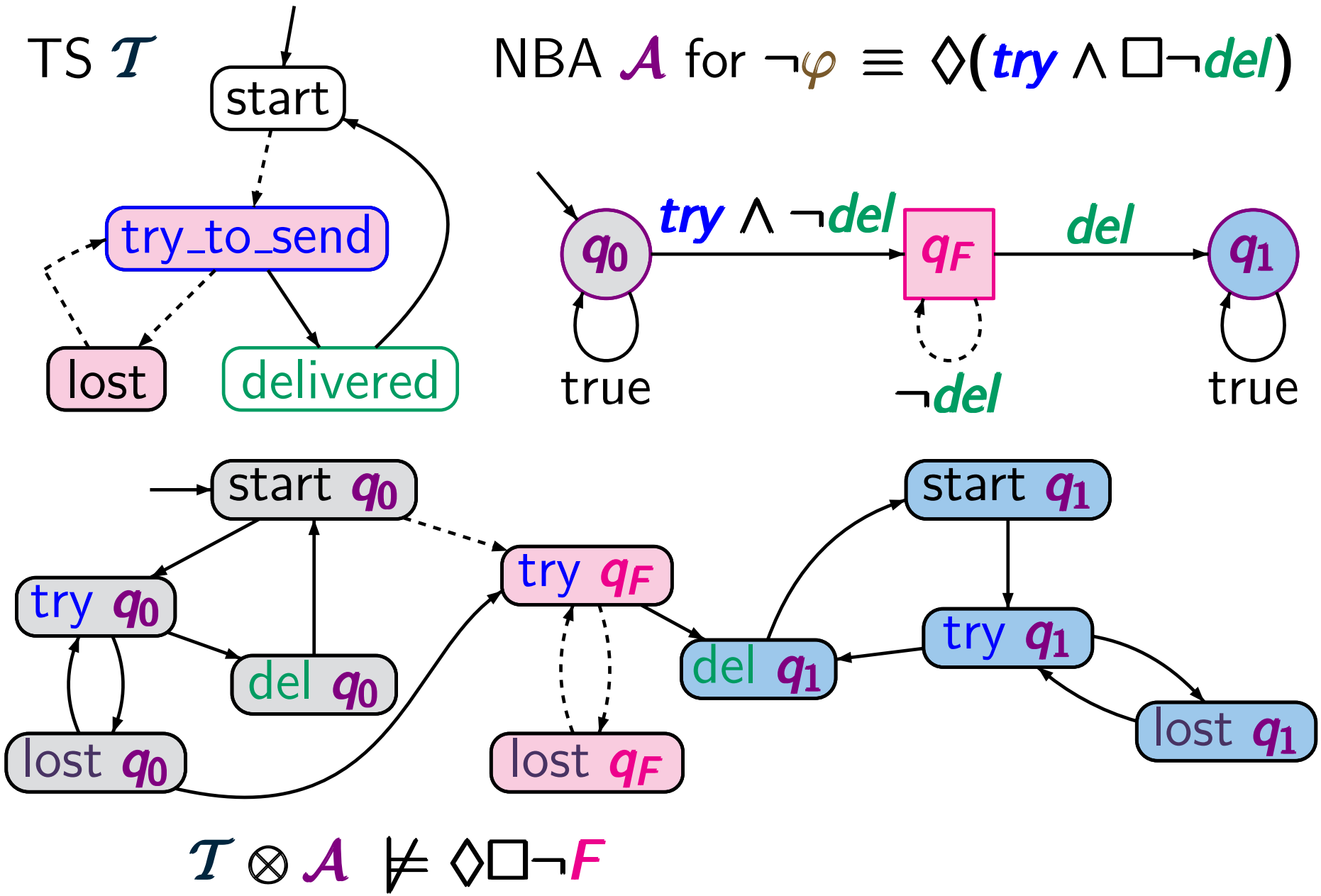
reachable fragment of the product-TS

Example: LTL model checking

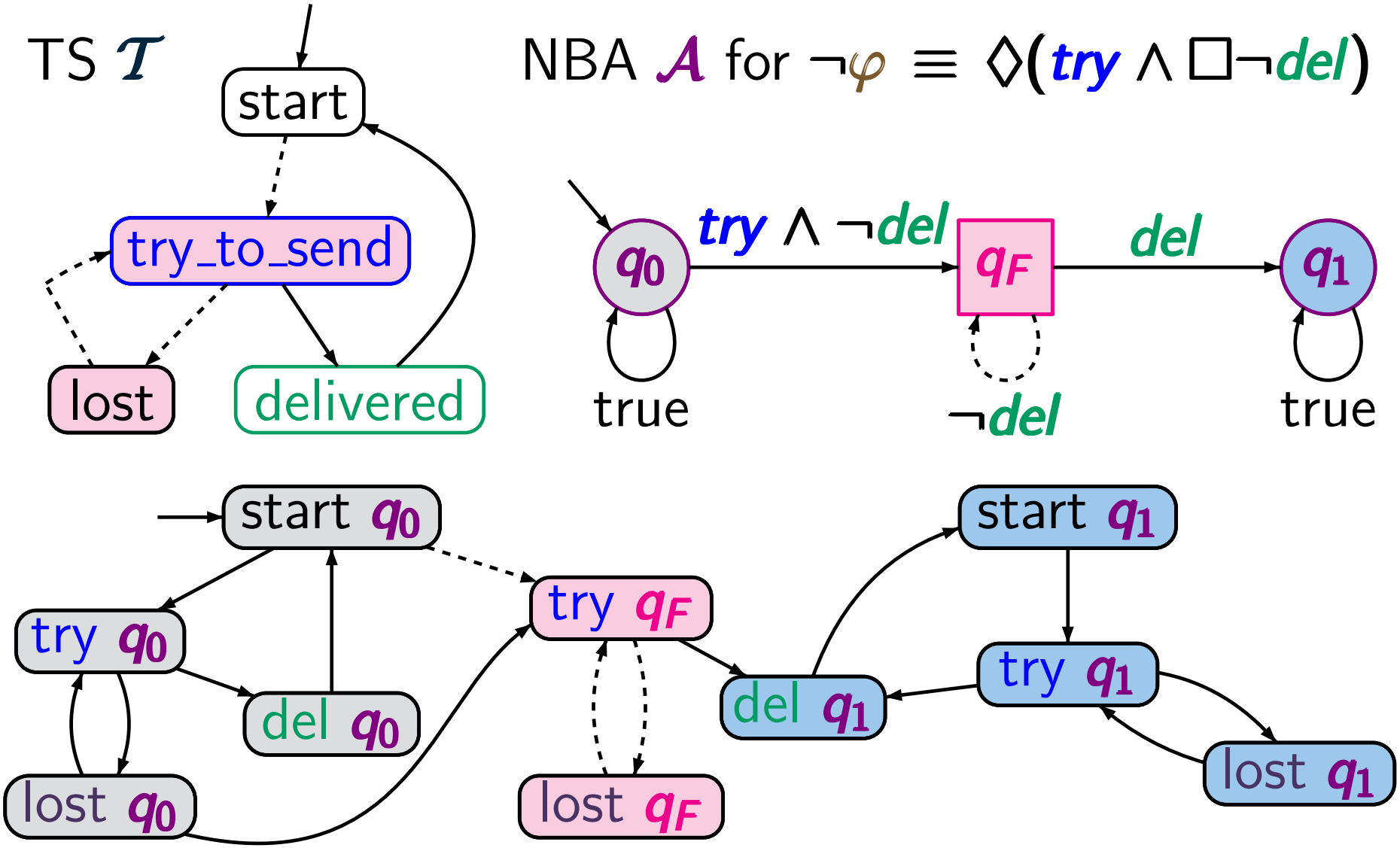


set of atomic propositions $AP' = \{q_0, q_1, q_F\}$

Example: LTL model checking



Example: LTL model checking



$\mathcal{T} \otimes \mathcal{A} \not\models \Diamond\Box\neg F$ hence: $\mathcal{T} \not\models \varphi$

LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula φ

question: does $\mathcal{T} \models \varphi$ hold **?**

LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula φ

question: does $\mathcal{T} \models \varphi$ hold **?**

construct an NBA $\mathcal{A}$ for $\neg\varphi$ and the product $\mathcal{T} \otimes \mathcal{A}$

check whether $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula φ

question: does $\mathcal{T} \models \varphi$ hold **?**

construct an NBA $\mathcal{A}$ for $\neg\varphi$ and the product $\mathcal{T} \otimes \mathcal{A}$

check whether $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$ $\longleftarrow$ persistence checking nested **DFS**

LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula φ

question: does $\mathcal{T} \models \varphi$ hold **?**

construct an NBA $\mathcal{A}$ for $\neg\varphi$ and the product $\mathcal{T} \otimes \mathcal{A}$

check whether $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$ ⟵ persistence checking nested **DFS**

IF $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

THEN return "**yes**"

ELSE compute a counterexample

$\langle s_0, p_0\rangle \ldots \langle s_n, p_n\rangle \ldots \langle s_n, p_n\rangle$

for $\mathcal{T} \otimes \mathcal{A}$ and $\Diamond\Box\neg F$

return "**no**" and $s_0 \ldots s_n \ldots s_n$

Complexity of LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula φ

question: does $\mathcal{T} \models \varphi$ hold ?

~~construct an NBA $\mathcal{A}$ for $\neg\varphi$ and the product $\mathcal{T} \otimes \mathcal{A}$~~

check whether $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$ ⟵ persistence checking nested **DFS**

IF $\mathcal{T} \otimes \mathcal{A} \models \Diamond\Box\neg F$

THEN return "**yes**"

ELSE compute a counterexample

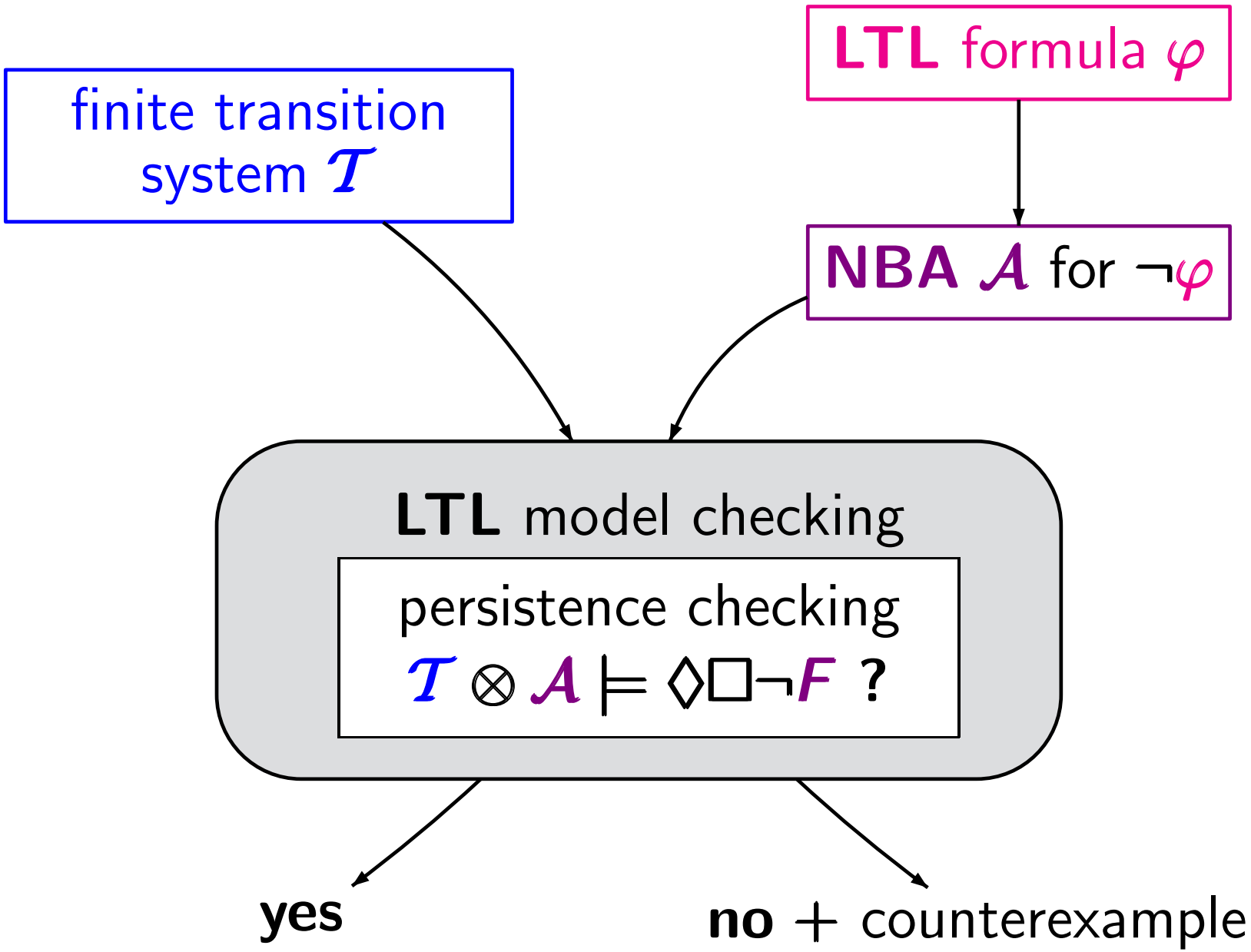
$\langle s_0, p_0\rangle \ldots \langle s_n, p_n\rangle \ldots \langle s_n, p_n\rangle$

for $\mathcal{T} \otimes \mathcal{A}$ and $\Diamond\Box\neg F$

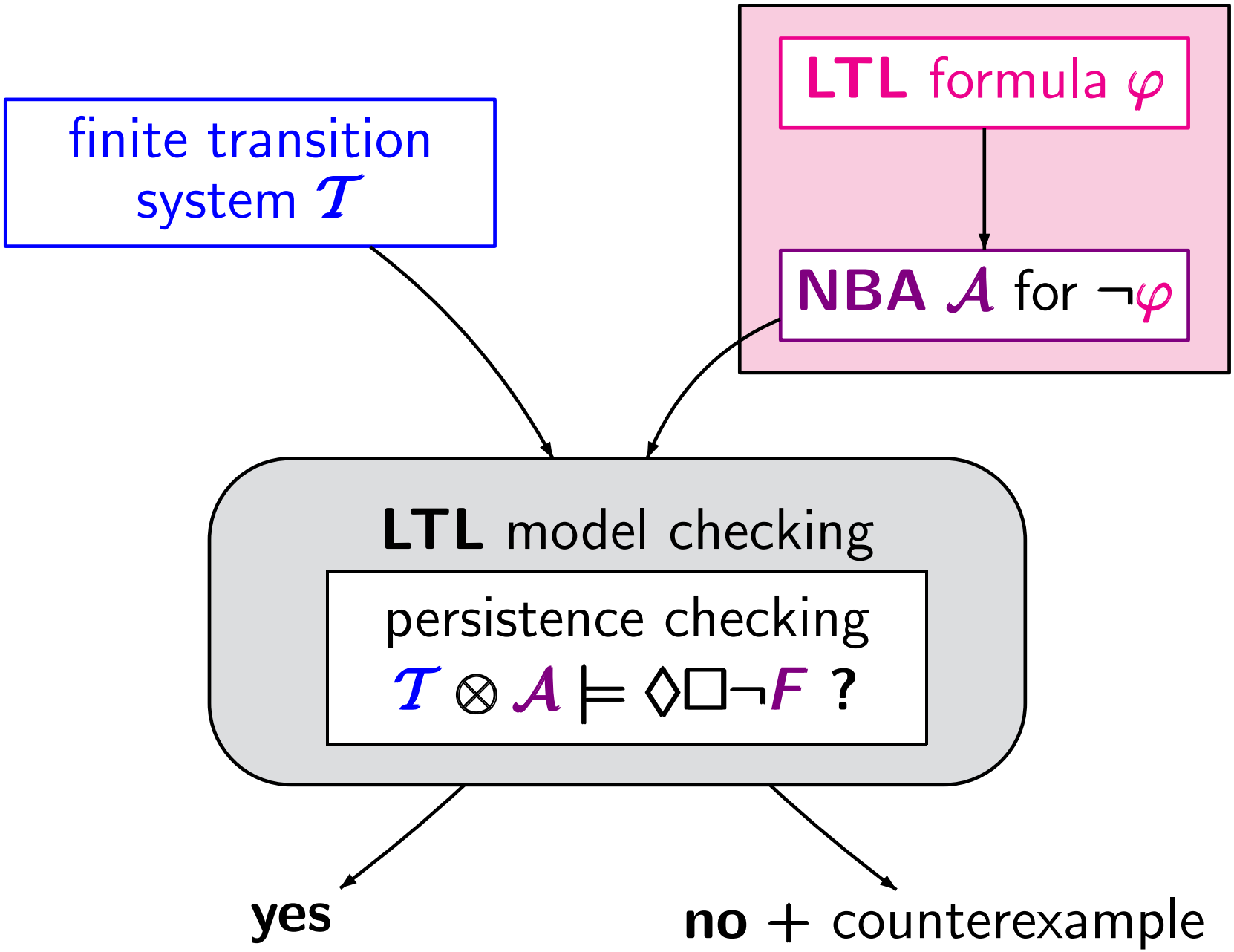
return "**no**" and $s_0 \ldots s_n \ldots s_n$

time complexity: $\mathcal{O}(size(\mathcal{T}) \cdot size(\mathcal{A}))$

LTL model checking



LTL model checking



From LTL to NBA

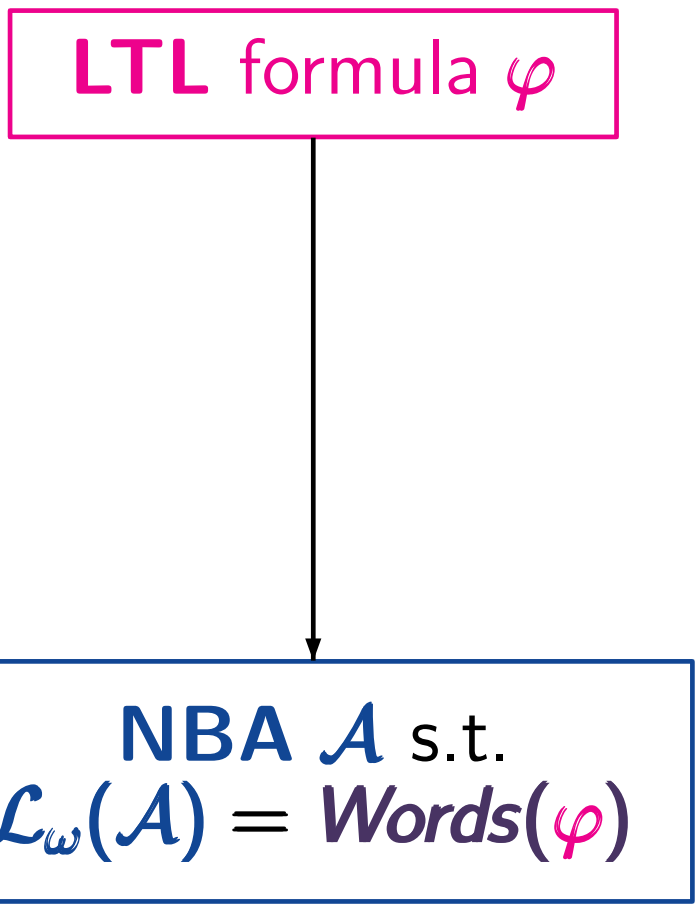
From LTL to NBA

For each **LTL** formula φ there is an **NBA** $\mathcal{A}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\varphi)$$

From LTL to NBA

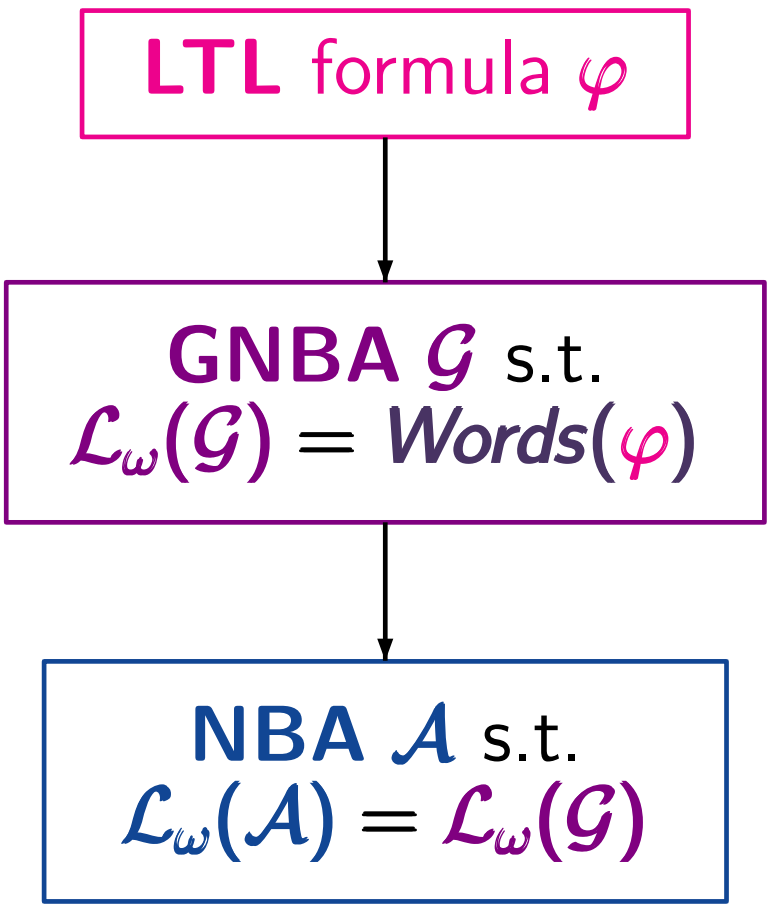
For each **LTL** formula φ there is an **NBA** $\mathcal{A}$ s.t.
$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\varphi)$



nondeterministic
Büchi automaton

From LTL to NBA

For each **LTL** formula φ there is an **NBA** $\mathcal{A}$ s.t.
$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\varphi)$

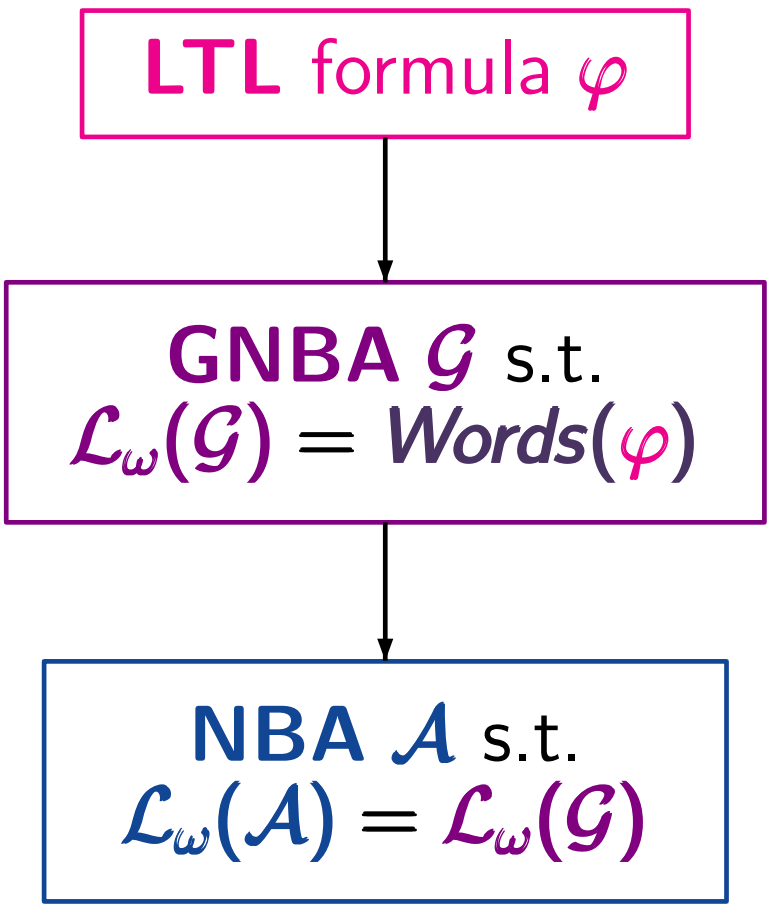


generalized NBA
several acceptance sets

nondeterministic
Büchi automaton
1 acceptance set

From LTL to NBA

For each **LTL** formula φ there is an **NBA** $\mathcal{A}$ s.t.
$\mathcal{L}_\omega(\mathcal{A}) = \mathit{Words}(\varphi)$



generalized NBA
k acceptance sets

k copies of $\mathcal{G}$

nondeterministic
Büchi automaton
1 acceptance set

Encoding of LTL semantics in a GNBA

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	
next $\bigcirc$	
until $\mathbf{U}$	

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	
until $\mathbf{U}$	

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until $\mathbf{U}$	

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until $\mathbf{U}$	via expansion law

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until U	via expansion law

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until U	via expansion law

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$

encoded in the states

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic *true*, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until U	via expansion law

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$

encoded in the states

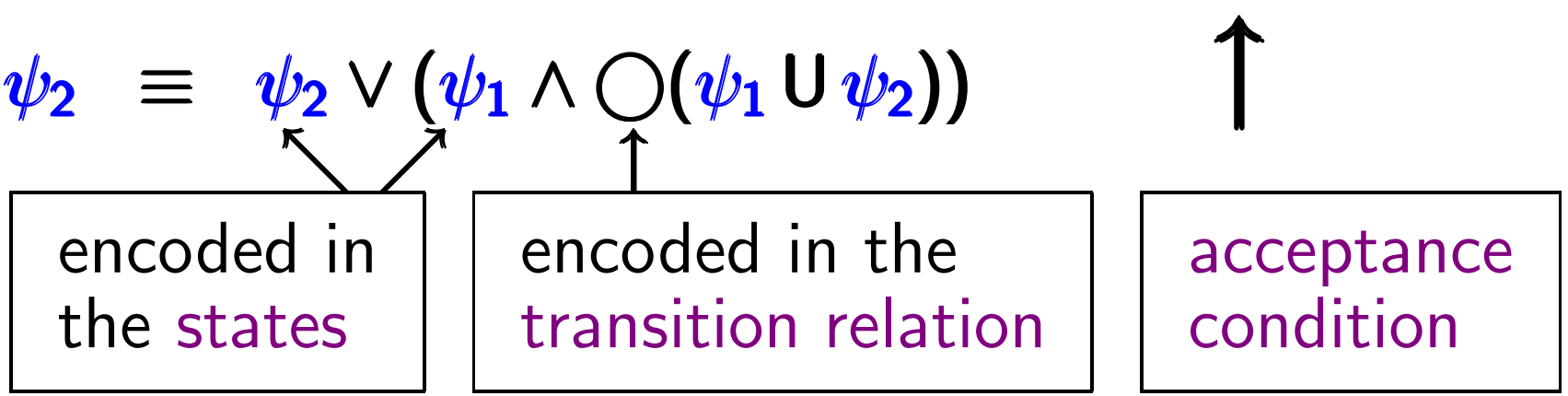
encoded in the transition relation

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$

semantics of ...	encoding
propositional logic ***true***, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until U	expansion law, **least fixed point**

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$



LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

$$A_0 \quad A_1 \quad A_2 \quad A_3 \quad \ldots \quad \in \mathit{Words}(\varphi)$$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

$$\begin{array}{cccccc} A_0 & A_1 & A_2 & A_3 & \ldots & \in \mathit{Words}(\varphi) \\ \downarrow & \downarrow & \downarrow & \downarrow & & \\ B_0 & B_1 & B_2 & B_3 & \ldots & \text{accepting run} \end{array}$$

where $B_i = \{\psi \in \mathit{cl}(\varphi) : A_i A_{i+1} A_{i+2} \ldots \models \psi\}$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

$$\begin{array}{cccccc} A_0 & A_1 & A_2 & A_3 & \ldots & \in \mathit{Words}(\varphi) \\ \downarrow & \downarrow & \downarrow & \downarrow & & \\ B_0 & B_1 & B_2 & B_3 & \ldots & \text{accepting run} \end{array}$$

where $B_i = \{\psi \in \mathit{cl}(\varphi) : A_i A_{i+1} A_{i+2} \ldots \models \psi\}$

set of subformulas of φ and their negations

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$

$\{a\}$ $\{a\}$ $\{a, b\}$ $\{b\}$ $\varnothing$ $\varnothing$ $\ldots \models \varphi$

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$

$$\underset{\downarrow}{\{a\}}\ \underset{\downarrow}{\{a\}}\ \underset{\downarrow}{\{a,b\}}\ \underset{\downarrow}{\{b\}}\ \underset{\downarrow}{\varnothing}\ \underset{\downarrow}{\varnothing}\ \ldots \models \varphi$$
$$B_0\ B_1\ B_2\ B_3\ B_4\ B_5$$

LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$

$$\underset{\downarrow}{\{a\}}\ \underset{\downarrow}{\{a\}}\ \underset{\downarrow}{\{a,b\}}\ \underset{\downarrow}{\{b\}}\ \underset{\downarrow}{\varnothing}\ \underset{\downarrow}{\varnothing}\ \ldots \models \varphi$$
$$B_0\ B_1\ B_2\ B_3\ B_4\ B_5$$

where the B_i's are subsets of
$\{a, \neg a, b, \neg b, \psi, \neg\psi, \varphi, \neg\varphi\}$

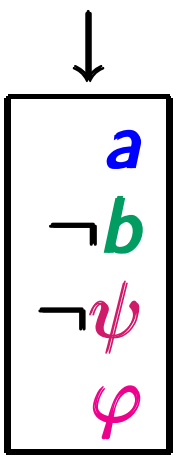
LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$

$\{a\}$ $\{a\}$ $\{a, b\}$ $\{b\}$ $\varnothing$ $\varnothing$ $\ldots$ $\models \varphi$



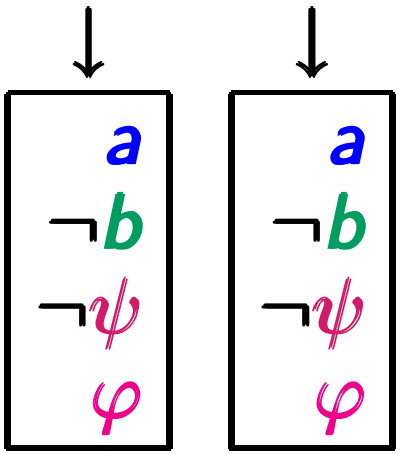
just for better readability:
tuple rather than set notation

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\quad\quad$ $\psi = \neg a \wedge b$

$\{a\}$ $\{a\}$ $\{a, b\}$ $\{b\}$ $\varnothing$ $\varnothing$ $\ldots$ $\models \varphi$



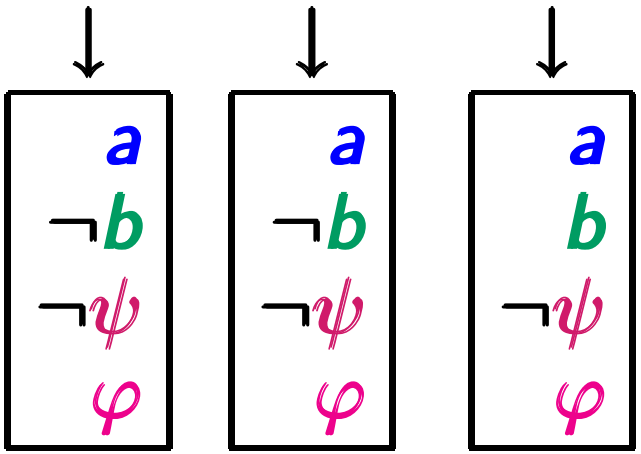
LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$

$\{a\}$ $\{a\}$ $\{a, b\}$ $\{b\}$ $\varnothing$ $\varnothing$ $\ldots$ $\models \varphi$

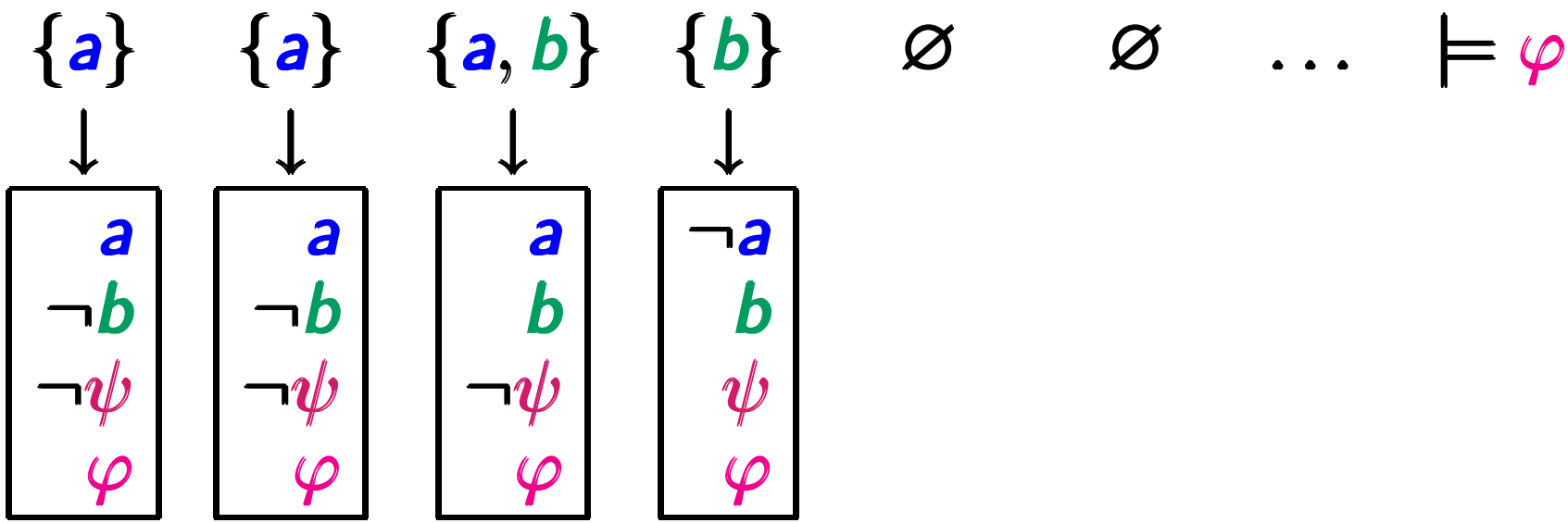


LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$

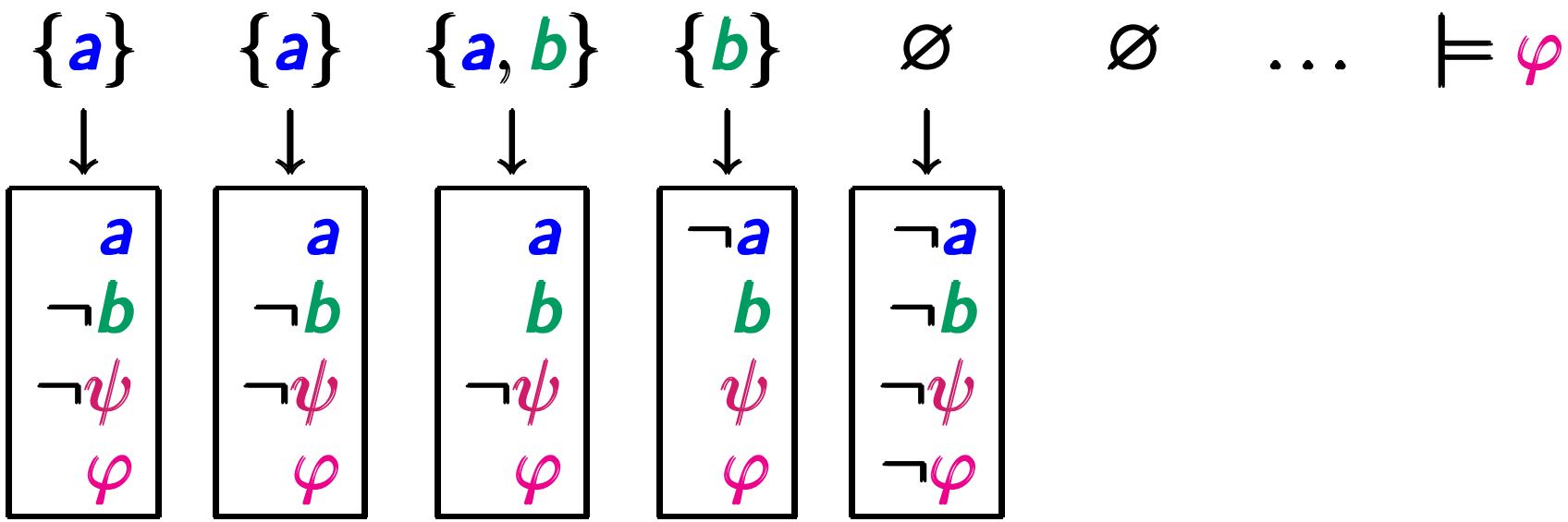


LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$

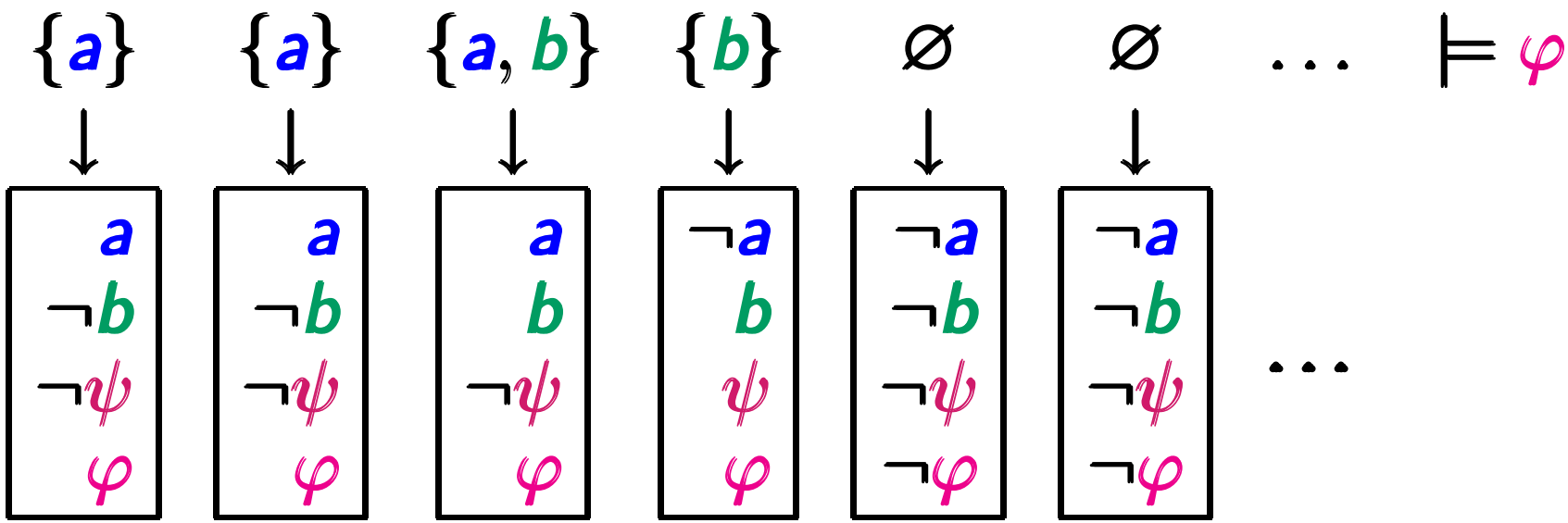


LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\mathit{Words}(\varphi)$

states of $\mathcal{G}$ $\widehat{=}$ (certain) sets of subformulas of φ
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \mathit{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a \,\mathsf{U}(\neg a \wedge b)$ $\qquad \psi = \neg a \wedge b$



Closure of LTL formulas

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

$$\mathit{cl}(\varphi) \stackrel{\text{def}}{=} \mathit{subf}(\varphi) \cup \{\neg\psi : \psi \in \mathit{subf}(\varphi)\}$$

where ψ and $\neg\neg\psi$ are identified

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

$$\mathit{cl}(\varphi) \stackrel{\text{def}}{=} \mathit{subf}(\varphi) \cup \{\neg\psi : \psi \in \mathit{subf}(\varphi)\}$$

where ψ and $\neg\neg\psi$ are identified

Example: if $\varphi = a\,\mathsf{U}(\neg a \wedge b)$ then

$$\mathit{cl}(\varphi) = \{a, b, \neg a \wedge b, \varphi\} \cup \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

$$\mathit{cl}(\varphi) \stackrel{\text{def}}{=} \mathit{subf}(\varphi) \cup \{\neg\psi : \psi \in \mathit{subf}(\varphi)\}$$

where ψ and $\neg\neg\psi$ are identified

Example: if $\varphi = a\,\mathsf{U}(\neg a \wedge b)$ then

$$\mathit{cl}(\varphi) = \{a, b, \neg a \wedge b, \varphi\} \cup \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$

Example: if $\varphi' = \Box a$

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

$$\mathit{cl}(\varphi) \stackrel{\text{def}}{=} \mathit{subf}(\varphi) \cup \{\neg\psi : \psi \in \mathit{subf}(\varphi)\}$$

where ψ and $\neg\neg\psi$ are identified

Example: if $\varphi = a\,\mathsf{U}(\neg a \wedge b)$ then

$$\mathit{cl}(\varphi) = \{a, b, \neg a \wedge b, \varphi\} \cup \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$

Example: if $\varphi' = \Box a = \neg\Diamond\neg a = \neg(\mathit{true}\,\mathsf{U}\,\neg a)$

Closure of LTL formulas

Let φ be an LTL formula. Then:

$$\mathit{subf}(\varphi) \stackrel{\text{def}}{=} \text{set of all subformulas of } \varphi$$

$$\mathit{cl}(\varphi) \stackrel{\text{def}}{=} \mathit{subf}(\varphi) \cup \{\neg\psi : \psi \in \mathit{subf}(\varphi)\}$$

where ψ and $\neg\neg\psi$ are identified

Example: if $\varphi = a\,\mathsf{U}(\neg a \wedge b)$ then

$$\mathit{cl}(\varphi) = \{a, b, \neg a \wedge b, \varphi\} \cup \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$

Example: if $\varphi' = \Box a = \neg\Diamond\neg a = \neg(\mathit{true}\,\mathsf{U}\,\neg a)$ then

$$\mathit{cl}(\varphi') = \{a, \neg a, \mathit{true}, \neg\mathit{true}, \Box a, \neg\Box a\}$$

Elementary formula-sets

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

(2) B is maximal consistent

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

if $\psi \in B$ then $\neg\psi \notin B$

(2) B is maximal consistent

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

if $\psi \in B$ then $\neg\psi \notin B$

if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$

(2) B is maximal consistent

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

if $\psi \in B$ then $\neg\psi \notin B$

if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$

if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg(\psi_1 \wedge \psi_2) \notin B$

(2) B is maximal consistent

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

- if $\psi \in B$ then $\neg\psi \notin B$
- if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$
- if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg(\psi_1 \wedge \psi_2) \notin B$
- if $\mathit{false} \in cl(\varphi)$ then $\mathit{false} \notin B$

(2) B is maximal consistent

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

- if $\psi \in B$ then $\neg\psi \notin B$
- if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$
- if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg(\psi_1 \wedge \psi_2) \notin B$
- if $\mathit{false} \in cl(\varphi)$ then $\mathit{false} \notin B$

(2) B is maximal consistent

- if $\psi \in cl(\varphi) \setminus B$ then $\neg\psi \in B$

(3) B is locally consistent with respect to until $\mathbf{U}$:

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

- if $\psi \in B$ then $\neg\psi \notin B$
- if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$
- if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg(\psi_1 \wedge \psi_2) \notin B$
- if $\mathit{false} \in cl(\varphi)$ then $\mathit{false} \notin B$

(2) B is maximal consistent

- if $\psi \in cl(\varphi) \setminus B$ then $\neg\psi \in B$

(3) B is locally consistent with respect to until U:

- if $\psi_1 \,\mathsf{U}\, \psi_2 \in B$ and $\neg\psi_2 \in B$ then $\neg\psi_1 \notin B$

Elementary formula-sets

Let $B \subseteq cl(\varphi)$. B is called elementary if:

(1) B is consistent w.r.t. propositional logic

- if $\psi \in B$ then $\neg\psi \notin B$
- if $\psi_1 \wedge \psi_2 \in B$ then $\neg\psi_1 \notin B$ and $\neg\psi_2 \notin B$
- if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg(\psi_1 \wedge \psi_2) \notin B$
- if $\mathit{false} \in cl(\varphi)$ then $\mathit{false} \notin B$

(2) B is maximal consistent

- if $\psi \in cl(\varphi) \setminus B$ then $\neg\psi \in B$

(3) B is locally consistent with respect to until U:

- if $\psi_1 \,\mathsf{U}\, \psi_2 \in B$ and $\neg\psi_2 \in B$ then $\neg\psi_1 \notin B$
- if $\psi_2 \in B$ and $\psi_1 \,\mathsf{U}\, \psi_2 \in cl(\varphi)$ then $\neg(\psi_1 \,\mathsf{U}\, \psi_2) \notin B$

Elementary formula-sets

$B \subseteq cl(\varphi)$ is elementary iff:

(i) B is maximal consistent w.r.t. prop. logic,
i.e., if ψ, $\psi_1 \wedge \psi_2 \in cl(\varphi)$ then:

$\psi \notin B$	iff	$\neg\psi \in B$
$\psi_1 \wedge \psi_2 \in B$	iff	$\psi_1 \in B$ and $\psi_2 \in B$
$\mathit{true} \in cl(\varphi)$	implies	$\mathit{true} \in B$

(ii) B is locally consistent with respect to until U,
i.e., if $\psi_1 \,\mathsf{U}\, \psi_2 \in cl(\varphi)$ then:

if $\psi_1 \,\mathsf{U}\, \psi_2 \in B$ and $\psi_2 \notin B$ then $\psi_1 \in B$

if $\psi_2 \in B$ then $\psi_1 \,\mathsf{U}\, \psi_2 \in B$

Elementary or not?

Let $\varphi = a \mathsf{U} (\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$

not elementary

propositional inconsistent

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$ not elementary, not maximal
as $\neg a \wedge b \notin B_2$
$\neg(\neg a \wedge b) \notin B_2$

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$ not elementary, not maximal
as $\neg a \wedge b \notin B_2$
$\neg(\neg a \wedge b) \notin B_2$

$B_3 = \{\neg a, b, \neg a \wedge b, \neg\varphi\}$

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$ not elementary, not maximal
as $\neg a \wedge b \notin B_2$
$\neg(\neg a \wedge b) \notin B_2$

$B_3 = \{\neg a, b, \neg a \wedge b, \neg\varphi\}$ not elementary
not locally consistent for U

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$ not elementary, not maximal
as $\neg a \wedge b \notin B_2$
$\neg(\neg a \wedge b) \notin B_2$

$B_3 = \{\neg a, b, \neg a \wedge b, \neg\varphi\}$ not elementary
not locally consistent for U

$B_4 = \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$

Elementary or not?

Let $\varphi = a \,\mathsf{U}(\neg a \wedge b)$.

$B_1 = \{a, b, \neg a \wedge b, \varphi\}$ not elementary
propositional inconsistent

$B_2 = \{\neg a, b, \varphi\}$ not elementary, not maximal
as $\neg a \wedge b \notin B_2$
$\neg(\neg a \wedge b) \notin B_2$

$B_3 = \{\neg a, b, \neg a \wedge b, \neg\varphi\}$ not elementary
not locally consistent for U

$B_4 = \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$ elementary

Example: elementary formula-sets

closure $cl(\varphi)$:

- set of all subformulas of φ and their negations
- ψ and $\neg\neg\psi$ are identified

elementary formula-sets: subsets B of $cl(\varphi)$

- maximal consistent w.r.t. propositional logic
- locally consistent w.r.t. U

For $\varphi = a\,\mathsf{U}(\neg a \wedge b)$, the elementary sets are:

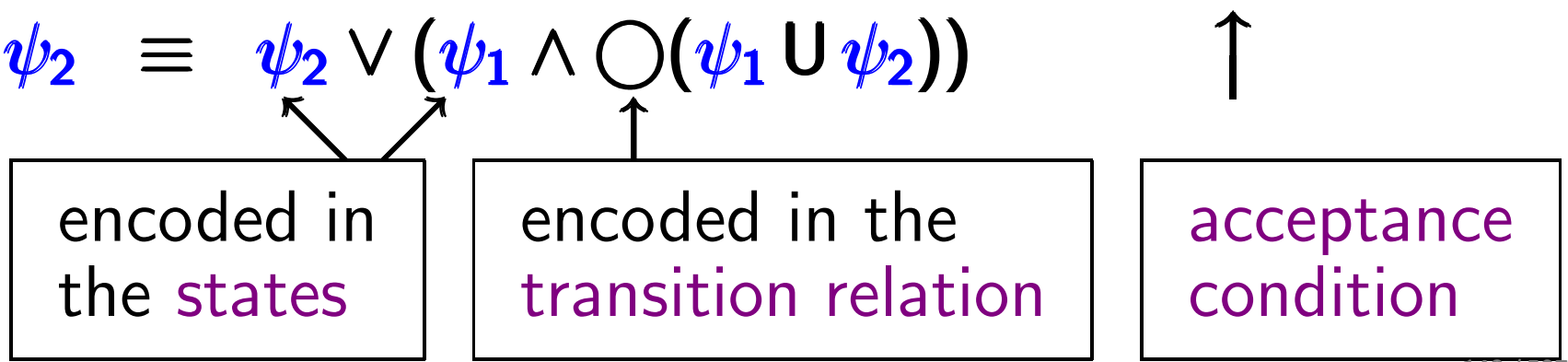
$$\{\ a,\ \ b, \neg(\neg a \wedge b), \varphi\} \qquad \{\ a,\ \ b, \neg(\neg a \wedge b), \neg\varphi\}$$
$$\{\ a, \neg b, \neg(\neg a \wedge b), \varphi\} \qquad \{\ a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$
$$\{\neg a,\ \ b,\ \ \neg a \wedge b\ , \varphi\} \qquad \{\neg a, \neg b, \neg(\neg a \wedge b), \neg\varphi\}$$

Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$:

semantics of ...	encoding
propositional logic *true*, $\neg$, $\wedge$	in the states
next $\bigcirc$	in the transition relation
until U	expansion law, least fixed point

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$



Encoding of LTL semantics in a GNBA

idea: encode the semantics of the operators appearing in φ by appropriate components of the GNBA $\mathcal{G}$:

semantics of ...	encoding
propositional logic *true*, $\neg$, $\wedge$	in the states ⟵ elementary formula sets
next $\bigcirc$	in the transition relation
until U	expansion law, least fixed point

$$\psi_1 \,\mathsf{U}\, \psi_2 \quad \equiv \quad \psi_2 \vee (\psi_1 \wedge \bigcirc(\psi_1 \,\mathsf{U}\, \psi_2))$$

↑

elementary formula sets

encoded in the transition relation

acceptance condition

GNBA for LTL-formula φ

GNBA for LTL-formula φ

$$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$$

GNBA for LTL-formula φ

$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

GNBA for LTL-formula φ

$$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

initial states: $Q_0 = \{B \in Q : \varphi \in B\}$

GNBA for LTL-formula φ

$$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

initial states: $Q_0 = \{B \in Q : \varphi \in B\}$

transition relation: for $B \in Q$ and $A \in 2^{AP}$:

if $A \neq B \cap AP$ then $\delta(B, A) = \varnothing$

GNBA for LTL-formula φ

$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

initial states: $Q_0 = \{B \in Q : \varphi \in B\}$

transition relation: for $B \in Q$ and $A \in 2^{AP}$:

if $A \neq B \cap AP$ then $\delta(B, A) = \varnothing$

if $A = B \cap AP$ then $\delta(B, A) =$ set of all $B' \in Q$ s.t.

$$\bigcirc\psi \in B \quad \text{iff} \quad \psi \in B'$$

$$\psi_1 \,\mathsf{U}\, \psi_2 \in B \quad \text{iff} \quad (\psi_2 \in B) \vee (\psi_1 \in B \wedge \psi_1 \,\mathsf{U}\, \psi_2 \in B')$$

GNBA for LTL-formula φ

$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

initial states: $Q_0 = \{B \in Q : \varphi \in B\}$

transition relation: for $B \in Q$ and $A \in 2^{AP}$:

if $A \neq B \cap AP$ then $\delta(B, A) = \varnothing$

if $A = B \cap AP$ then $\delta(B, A) =$ set of all $B' \in Q$ s.t.

$$\bigcirc\psi \in B \quad \text{iff} \quad \psi \in B'$$

$$\psi_1 \,\mathsf{U}\, \psi_2 \in B \quad \text{iff} \quad (\psi_2 \in B) \vee (\psi_1 \in B \wedge \psi_1 \,\mathsf{U}\, \psi_2 \in B')$$

acceptance set $\mathcal{F} = \{F_{\psi_1 \,\mathsf{U}\, \psi_2} : \psi_1 \,\mathsf{U}\, \psi_2 \in cl(\varphi)\}$

GNBA for LTL-formula φ

$\mathcal{G} = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$

state space: $Q = \{B \subseteq cl(\varphi) : B \text{ is elementary}\}$

initial states: $Q_0 = \{B \in Q : \varphi \in B\}$

transition relation: for $B \in Q$ and $A \in 2^{AP}$:

if $A \neq B \cap AP$ then $\delta(B, A) = \varnothing$

if $A = B \cap AP$ then $\delta(B, A) =$ set of all $B' \in Q$ s.t.

$$\bigcirc\psi \in B \quad \text{iff} \quad \psi \in B'$$

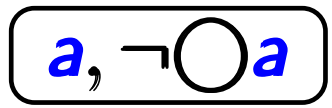
$$\psi_1 \,\mathsf{U}\, \psi_2 \in B \quad \text{iff} \quad (\psi_2 \in B) \vee (\psi_1 \in B \wedge \psi_1 \,\mathsf{U}\, \psi_2 \in B')$$

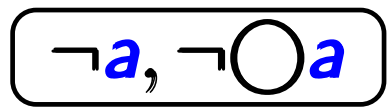
acceptance set $\mathcal{F} = \{F_{\psi_1 \,\mathsf{U}\, \psi_2} : \psi_1 \,\mathsf{U}\, \psi_2 \in cl(\varphi)\}$

where $F_{\psi_1 \,\mathsf{U}\, \psi_2} = \{B \in Q : \psi_1 \,\mathsf{U}\, \psi_2 \notin B \vee \psi_2 \in B\}$

Example: GNBA for $\varphi = \bigcirc a$

Example: GNBA for $\varphi = \bigcirc a$





Example: GNBA for $\varphi = \bigcirc a$

initial states: formula-sets B with $\bigcirc a \in B$

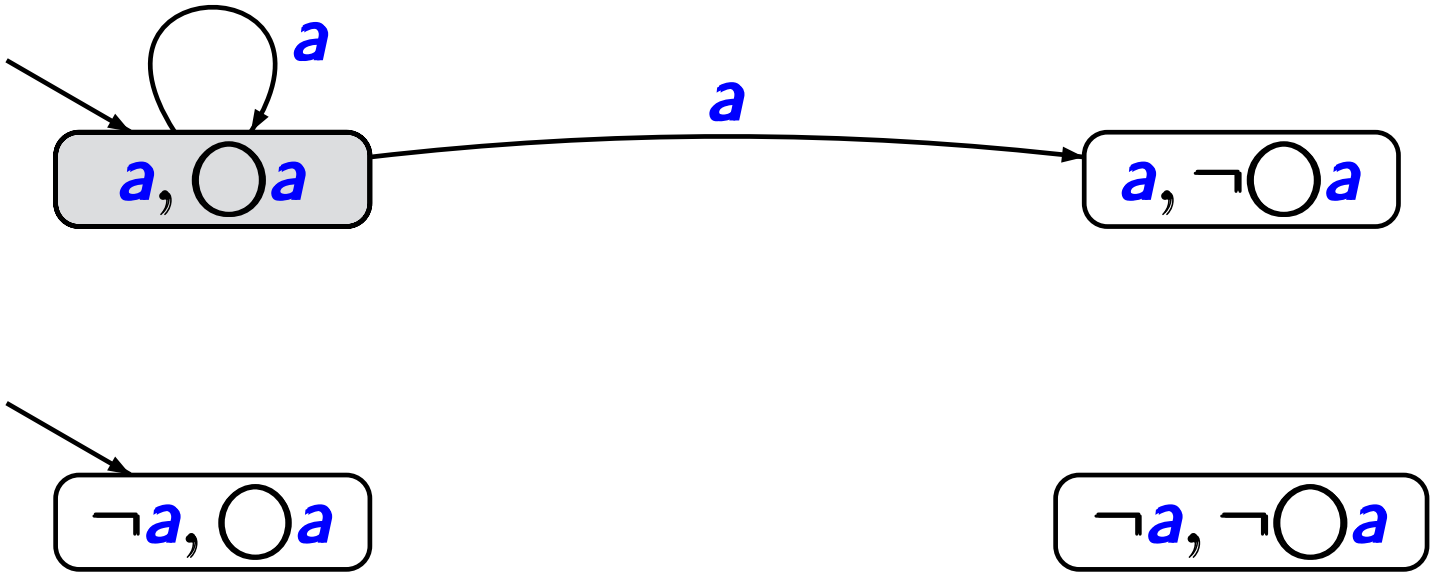
Example: GNBA for $\varphi = \bigcirc a$

initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

Example: GNBA for $\varphi = \bigcirc a$

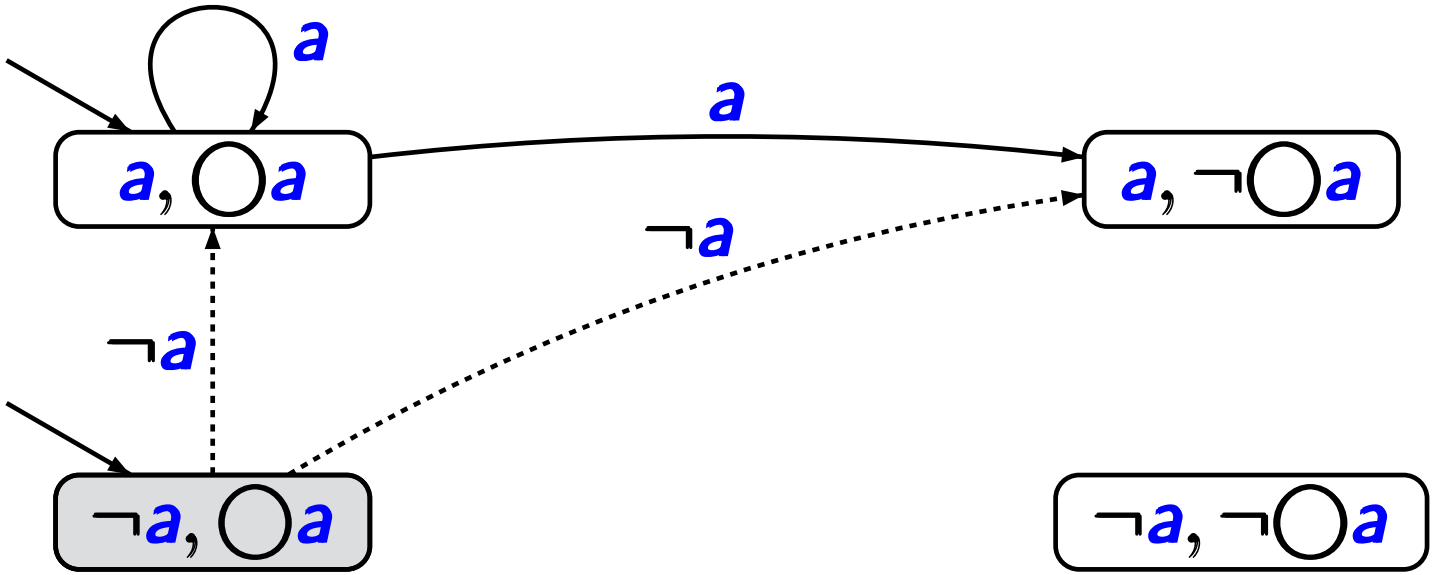


initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

Example: GNBA for $\varphi = \bigcirc a$

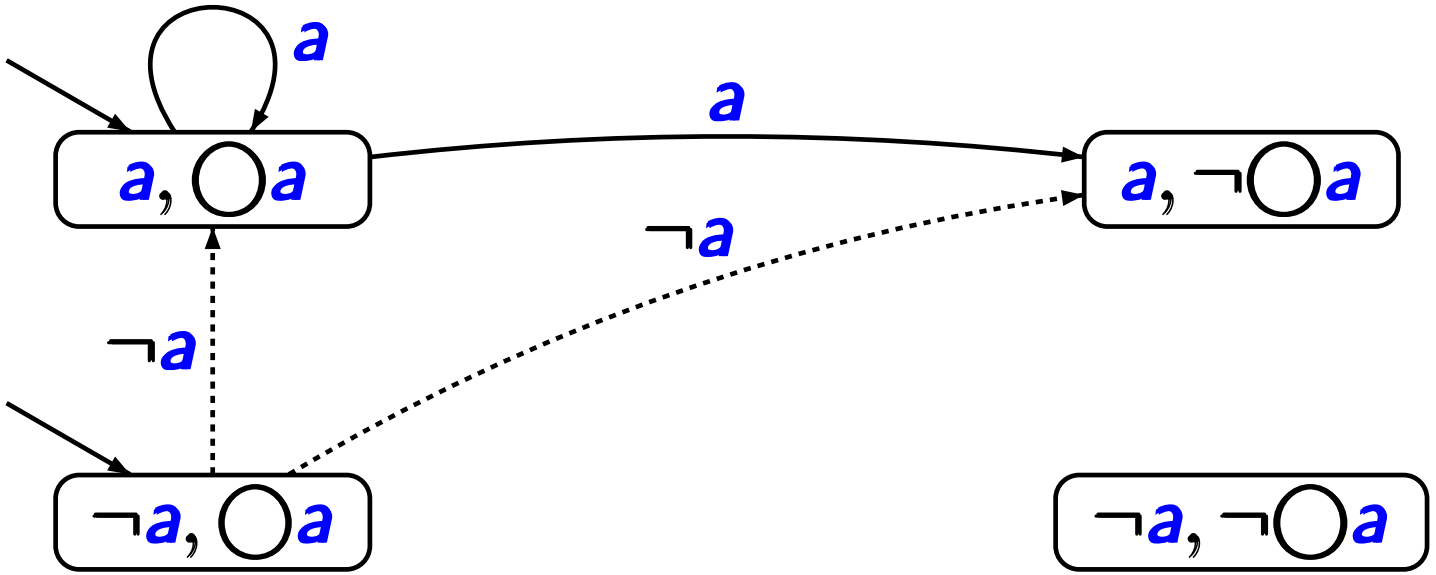


initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

Example: GNBA for $\varphi = \bigcirc a$



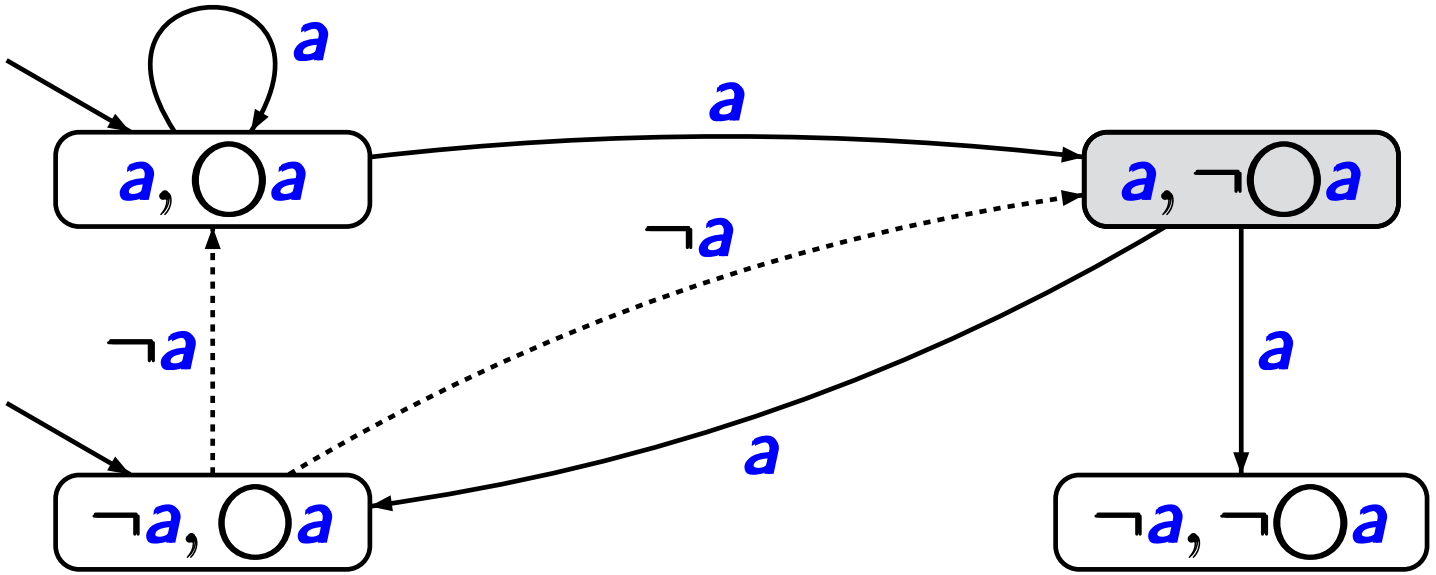
initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

if $\bigcirc a \notin B$ then $\delta(B, B \cap \{a\}) = \{B' : a \notin B'\}$

Example: GNBA for $\varphi = \bigcirc a$



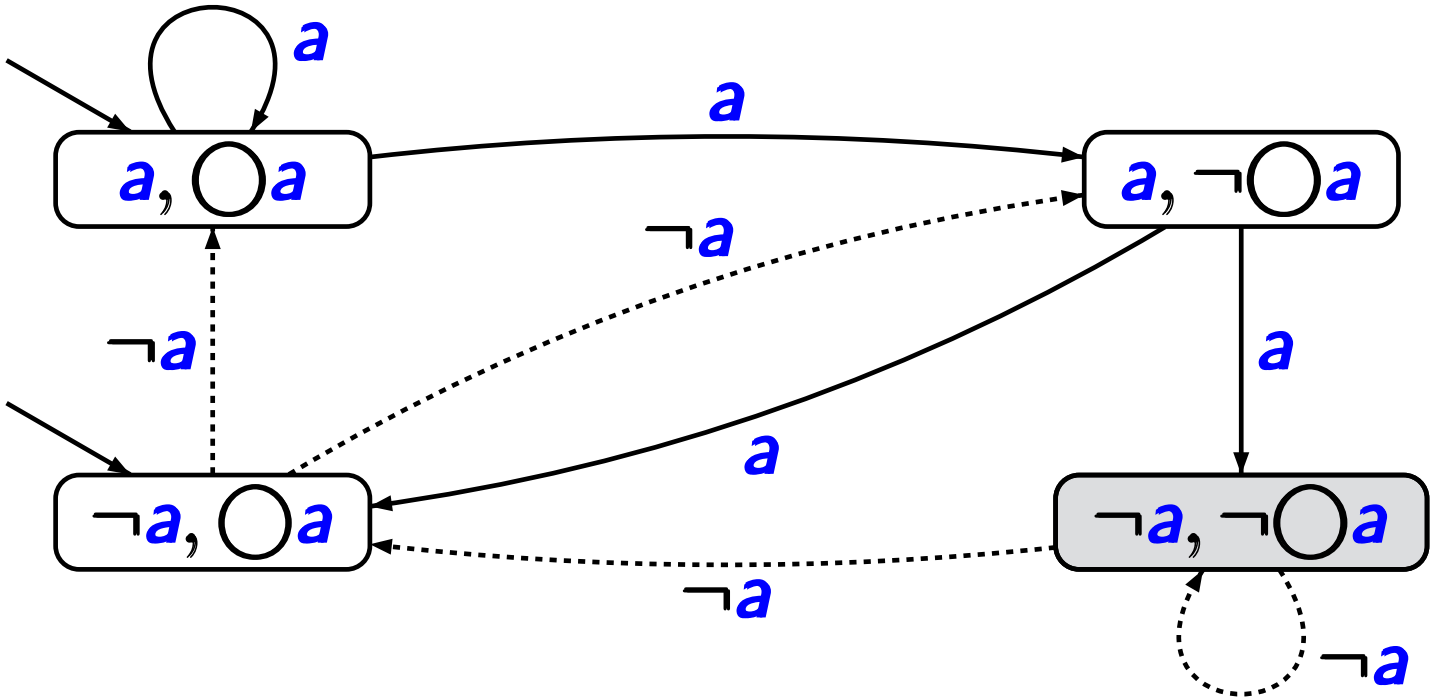
initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

if $\bigcirc a \notin B$ then $\delta(B, B \cap \{a\}) = \{B' : a \notin B'\}$

Example: GNBA for $\varphi = \bigcirc a$



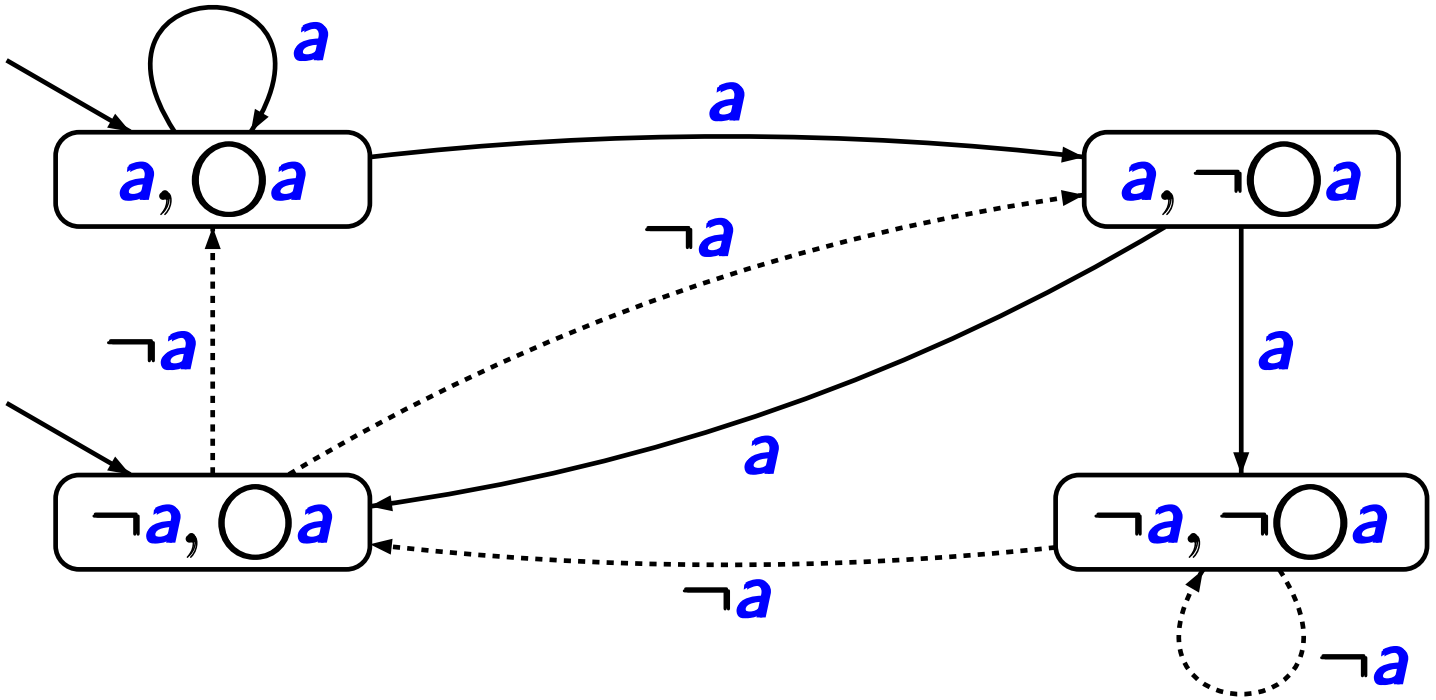
initial states: formula-sets B with $\bigcirc a \in B$

transition relation:

if $\bigcirc a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$

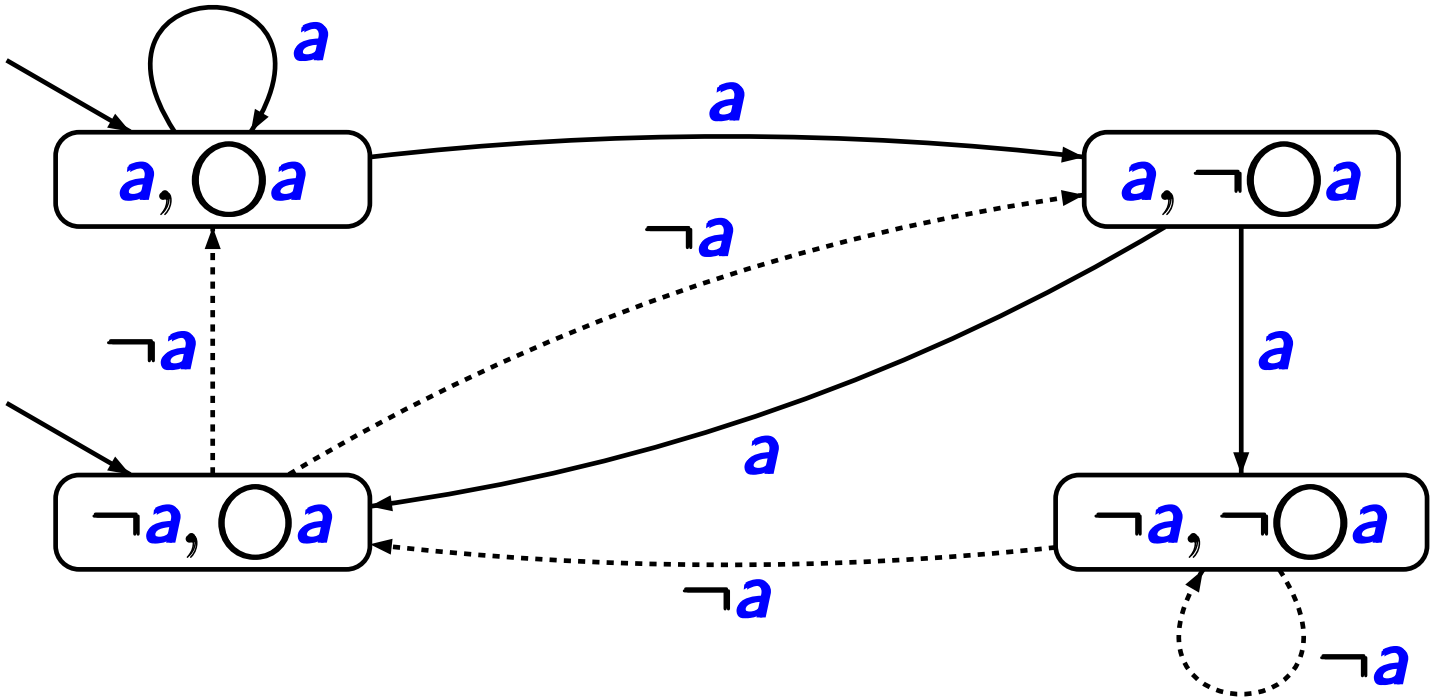
if $\bigcirc a \notin B$ then $\delta(B, B \cap \{a\}) = \{B' : a \notin B'\}$

Example: GNBA for $\varphi = \bigcirc a$



set of acceptance sets:

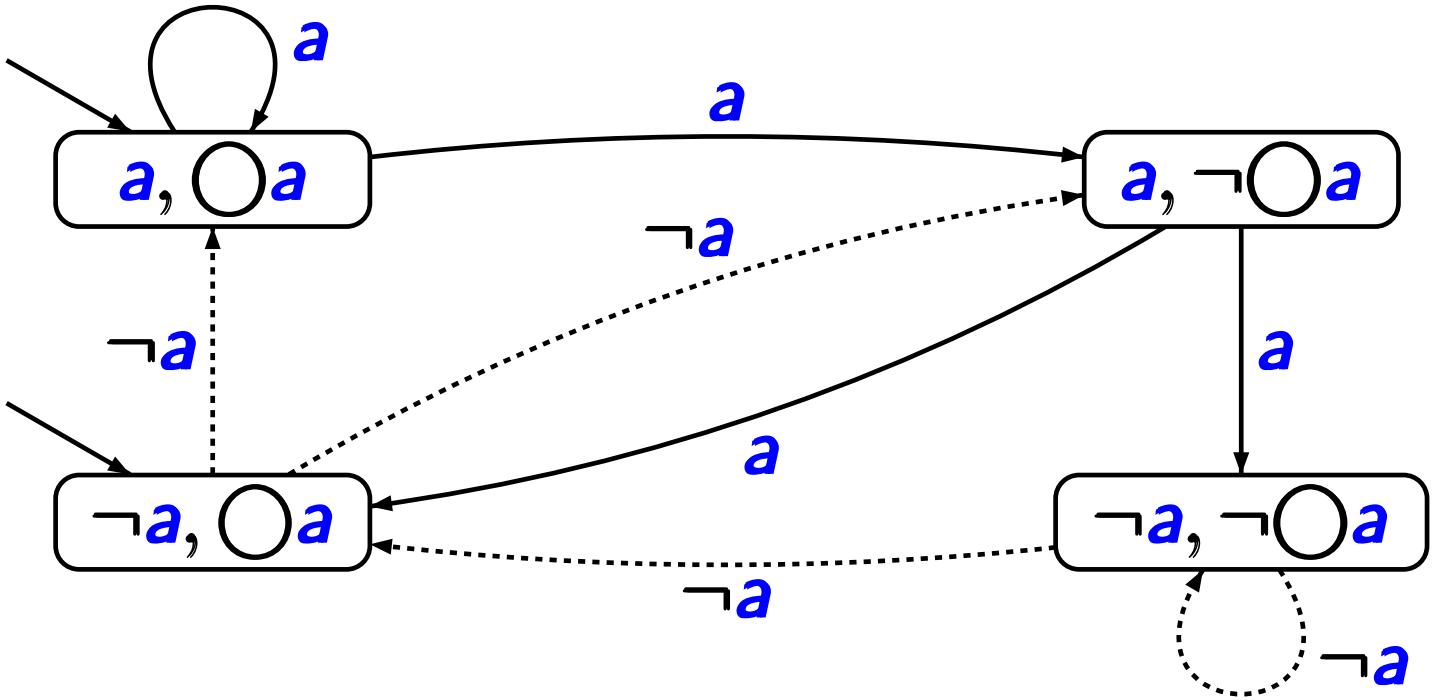
Example: GNBA for $\varphi = \bigcirc a$



set of acceptance sets: $\mathcal{F} = \varnothing$

hence: all words having an infinite run are accepted

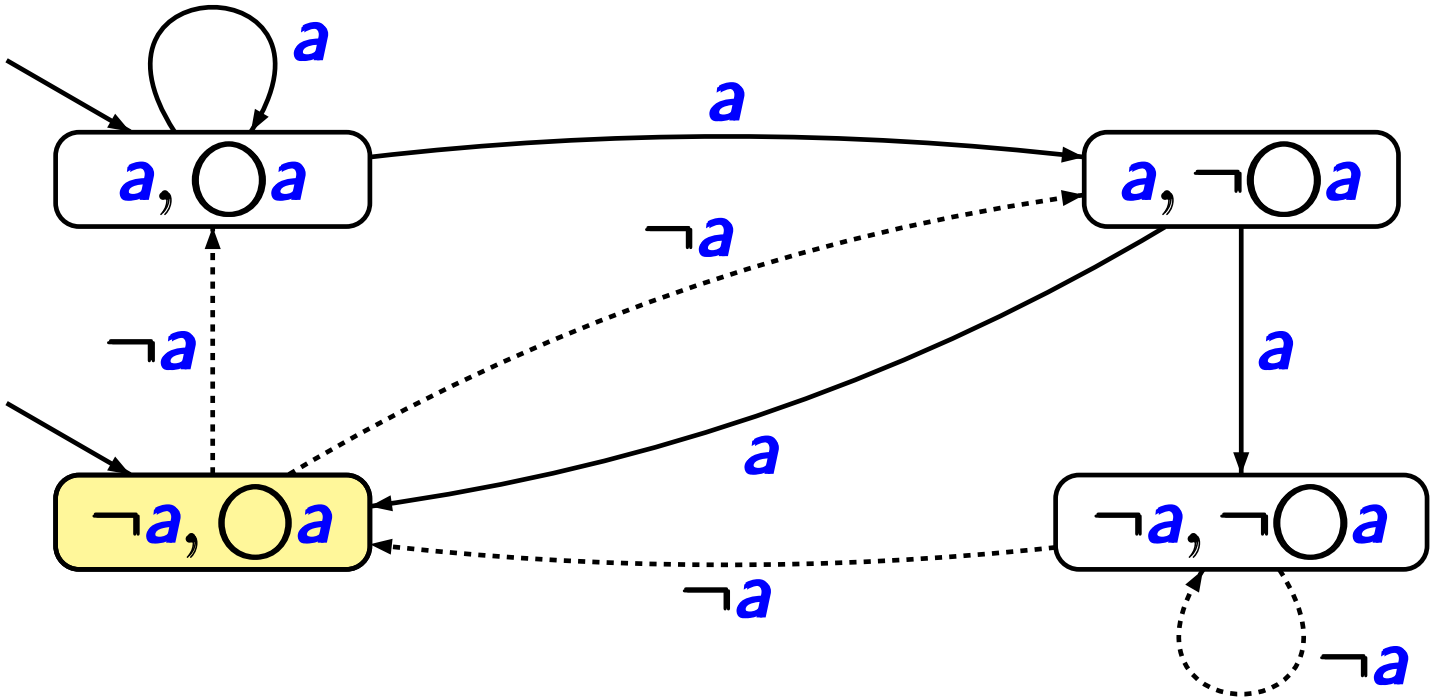
Example: GNBA for $\varphi = \bigcirc a$



set of acceptance sets: $\mathcal{F} = \varnothing$

$$\varnothing \quad \{a\} \quad \{a\} \quad \varnothing \quad \varnothing \quad \ldots \quad \models \bigcirc a$$

Example: GNBA for $\varphi = \bigcirc a$



set of acceptance sets: $\mathcal{F} = \varnothing$

$$\varnothing \quad \{a\} \quad \{a\} \quad \varnothing \quad \varnothing \quad \ldots \quad \models \bigcirc a$$