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

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

**question:** does $\mathcal{T} \models \varphi$ hold?
LTL model checking problem

given: finite transition system $T$ over $AP$
(without terminal states)
LTL-formula $\varphi$ over $AP$

question: does $T \models \varphi$ hold?

basic idea: try to refute $T \models \varphi$
LTL model checking problem

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

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

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

$$\pi \not\models \varphi$$
LTL model checking problem

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

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

**basic idea:** try to refute $\mathcal{T} \models \varphi$ by searching
for a path $\pi$ 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 $T$ over $AP$
LTL-formula $\varphi$ over $AP$

question: does $T \models \varphi$ hold?

1. construct an NBA $A$ for $Words(\neg \varphi)$
The LTL model checking problem

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

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

1. construct an $NBA$ $\mathcal{A}$ for $Words(\neg \varphi)$
2. search a path $\pi$ in $\mathcal{T}$ with
   \[ trace(\pi) \in Words(\neg \varphi) \]
The LTL model checking problem

given: finite transition system $\mathcal{I}$ over $AP$
LTL-formula $\varphi$ over $AP$

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

1. construct an NBA $\mathcal{A}$ for $\text{Words}(\neg \varphi)$
2. search a path $\pi$ in $\mathcal{I}$ with
   \[
   \text{trace}(\pi) \in \text{Words}(\neg \varphi) = \mathcal{L}_\omega(\mathcal{A})
   \]
The LTL model checking problem

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

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

1. construct an NBA $\mathcal{A}$ for $\text{Words}(\neg \varphi)$
2. search a path $\pi$ in $\mathcal{T}$ with
   \[ \text{trace}(\pi) \in \text{Words}(\neg \varphi) = \mathcal{L}_\omega(\mathcal{A}) \]

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

finite transition system $\mathcal{T}$

LTL formula $\varphi$

LTL model checking

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

yes

no
Automata-based LTL model checking

finite transition system $T$

LTL formula $\varphi$

NBA $A$ for $\neg \varphi$

"bad behaviors"

LTL model checking

does $T \models \varphi$ hold?

yes

no
Automata-based LTL model checking

finite transition system $\mathcal{T}$

LTL formula $\varphi$

NBA $\mathcal{A}$ for $\neg \varphi$

“bad behaviors”

$LTL$ model checking via persistence checking

$\mathcal{T} \otimes \mathcal{A} \models \Diamond \Box \text{ no final state}$

yes

no
Automata-based LTL model checking

finite transition system $\mathcal{T}$

$LTL$ formula $\varphi$

NBA $\mathcal{A}$ for $\neg \varphi$

"bad behaviors"

$LTL$ model checking via persistence checking

$\mathcal{T} \otimes \mathcal{A} \models \Diamond \Box \text{ no final state}\, ?$

yes

no $\pm$ error indication
<table>
<thead>
<tr>
<th>safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
</table>

Safety and LTL model checking

ltlmc3.2-20
<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td>$\mathcal{L}(A) \subseteq (2^\text{AP})^+$</td>
</tr>
</tbody>
</table>
## Safety and LTL model checking

<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the “bad behaviors”</td>
</tr>
<tr>
<td>$\mathcal{L}(A) \subseteq (2^{AP})^+$</td>
<td>$\mathcal{L}_\omega(A) = \text{Words}(\neg \varphi)$</td>
</tr>
</tbody>
</table>
Safety and LTL model checking

<table>
<thead>
<tr>
<th>safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the “bad behaviors”</td>
</tr>
<tr>
<td>$\mathcal{L}(A) \subseteq (2^{AP})^+$</td>
<td>$\mathcal{L}_\omega(A) = \text{Words}(\neg \varphi)$</td>
</tr>
<tr>
<td>$\text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset$</td>
<td></td>
</tr>
</tbody>
</table>

$LTLMC3.2-20$
<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the “bad behaviors”</td>
</tr>
<tr>
<td>$\mathcal{L}(A) \subseteq (2^{\text{AP}})^+$</td>
<td>$\mathcal{L}_\omega(A) = \text{Words}(\neg \varphi)$</td>
</tr>
<tr>
<td>$\text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset$</td>
<td>$\text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset$</td>
</tr>
</tbody>
</table>
### Safety and LTL model checking

<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the “bad behaviors”</td>
</tr>
<tr>
<td>$\mathcal{L}(A) \subseteq (2^\text{AP})^+$</td>
<td>$\mathcal{L}_\omega(A) = \text{Words}(\neg \varphi)$</td>
</tr>
<tr>
<td>$\text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset$</td>
<td>$\text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset$</td>
</tr>
<tr>
<td>invariant checking in the product</td>
<td></td>
</tr>
<tr>
<td>$T \otimes A \models \Box \neg F$</td>
<td></td>
</tr>
</tbody>
</table>

---

$ltlmc3.2-20$
## Safety and LTL model checking

<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the &quot;bad behaviors&quot;</td>
</tr>
<tr>
<td>$\mathcal{L}(A) \subseteq (2^{AP})^+$</td>
<td>$\mathcal{L}_\omega(A) = \text{Words}(\neg \varphi)$</td>
</tr>
<tr>
<td>$\text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset$</td>
<td>$\text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset$</td>
</tr>
<tr>
<td>invariant checking in the product $T \otimes A \models \Box \neg F$ ?</td>
<td>persistence checking in the product $T \otimes A \models \Diamond \Box \neg F$ ?</td>
</tr>
<tr>
<td>Safety property ( E )</td>
<td>LTL-formula ( \varphi )</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>\textbf{NFA} for the bad prefixes for ( E ) ( \mathcal{L}(A) \subseteq (2^{AP})^+ )</td>
<td>\textbf{NBA} for the “bad behaviors” ( \mathcal{L}_\omega(A) = \text{Words}(\neg \varphi) )</td>
</tr>
<tr>
<td>( \text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset )</td>
<td>( \text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset )</td>
</tr>
<tr>
<td>\text{Invariant checking} in the product ( T \otimes A \models \Box \neg F ) ?</td>
<td>\text{Persistence checking} in the product ( T \otimes A \models \Diamond \Box \neg F ) ?</td>
</tr>
<tr>
<td>\text{error indication:} ( \hat{\pi} \in \text{Paths}_{\text{fin}}(T) ) s.t. ( \text{trace}(\hat{\pi}) \in \mathcal{L}(A) )</td>
<td></td>
</tr>
</tbody>
</table>
## Safety and LTL model checking

<table>
<thead>
<tr>
<th>Safety property $E$</th>
<th>LTL-formula $\varphi$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFA</strong> for the bad prefixes for $E$</td>
<td><strong>NBA</strong> for the “bad behaviors”</td>
</tr>
<tr>
<td>$\mathcal{L}(\mathcal{A}) \subseteq (2^{AP})^+$</td>
<td>$\mathcal{L}_\omega(\mathcal{A}) = \text{Words}(\neg \varphi)$</td>
</tr>
<tr>
<td>$\text{Traces}_{\text{fin}}(\mathcal{T}) \cap \mathcal{L}(\mathcal{A}) = \emptyset$</td>
<td>$\text{Traces}(\mathcal{T}) \cap \mathcal{L}_\omega(\mathcal{A}) = \emptyset$</td>
</tr>
<tr>
<td>invariant checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Box \neg F$</td>
<td>persistence checking in the product $\mathcal{T} \otimes \mathcal{A} \models \Diamond \Box \neg F$</td>
</tr>
<tr>
<td>error indication: $\hat{\pi} \in \text{Paths}_{\text{fin}}(\mathcal{T})$ s.t. $\text{trace}(\hat{\pi}) \in \mathcal{L}(\mathcal{A})$</td>
<td>error indication: prefix of a path $\pi$ s.t. $\text{trace}(\pi) \in \mathcal{L}_\omega(\mathcal{A})$</td>
</tr>
</tbody>
</table>
Safety vs LTL model checking
Safety vs LTL model checking

\[ T \models \text{safety property } E \]

iff \[ Traces_{\text{fin}}(T) \cap L(A) = \emptyset \]

where \( A \) is an NFA for the bad prefixes

\[ T \models \text{LTL-formula } \varphi \]

iff \[ Traces(T) \cap L_\omega(A) = \emptyset \]

where \( A \) is an NBA for \( \neg \varphi \)
Safety vs LTL model checking

\[ T \models \text{safety property } E \]

iff \[ \text{Traces}_{\text{fin}}(T) \cap L(A) = \emptyset \]

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 \( T \otimes A \) s.t. \( q_n \in F \)

\[ T \models \text{LTL-formula } \varphi \]

iff \[ \text{Traces}(T) \cap L_\omega(A) = \emptyset \]

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 \( T \otimes A \) s.t. \( q_i \in F \) for infinitely many \( i \in \mathbb{N} \)
Safety vs LTL model checking

\( T \models \text{safety property } E \)

iff \( \text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset \)

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 \( T \otimes A \) s.t. \( q_n \in F \)

iff \( T \otimes A \models \Box \neg F \)

\[ \]

\( T \models \text{LTL-formula } \varphi \)

iff \( \text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset \)

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 \( T \otimes A \) s.t. \( q_i \in F \) for infinitely many \( i \in \mathbb{N} \)

iff \( T \otimes A \models \Diamond \Box \neg F \)
Safety vs LTL model checking

\( T \models \text{safety property } E \)

iff \( \text{Traces}_{\text{fin}}(T) \cap \mathcal{L}(A) = \emptyset \)

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 \( T \otimes A \) s.t. \( q_n \in F \)

iff \( T \otimes A \models \square \neg F \leftarrow \text{invariant checking} \)

\( T \models \text{LTL-formula } \varphi \)

iff \( \text{Traces}(T) \cap \mathcal{L}_\omega(A) = \emptyset \)

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 \( T \otimes A \) s.t. \( q_i \in F \) for infinitely many \( i \in \mathbb{N} \)

iff \( T \otimes A \models \diamond \square \neg F \leftarrow \text{persistence checking} \)
Recall: nondeterministic Büchi automata

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

- $Q$ finite set of states
- $\Sigma$ alphabet
- $\delta : Q \times \Sigma \rightarrow 2^Q$ 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
- \( \Sigma \) alphabet
- \( \delta : Q \times \Sigma \rightarrow 2^Q \) 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 \( \pi \) is accepting if \( \exists i \in \mathbb{N}. q_i \in F \)
Recall: nondeterministic Büchi automata

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

- $Q$ finite set of states
- $\Sigma$ alphabet
- $\delta : Q \times \Sigma \rightarrow 2^Q$ 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:

$$\mathcal{L}_\omega(\mathcal{A}) \overset{\text{def}}{=} \text{set of infinite words over } \Sigma \text{ that have an accepting run in } \mathcal{A}$$
Recall: nondeterministic Büchi automata

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

- $Q$ finite set of states
- $\Sigma$ alphabet \[\text{here: } \Sigma = 2^{AP}\]
- $\delta : Q \times \Sigma \rightarrow 2^Q$ 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:

\[
\mathcal{L}_\omega(\mathcal{A}) \overset{\text{def}}{=} \text{set of infinite words over } \Sigma \text{ that have an accepting run in } \mathcal{A}
\]
For each **LTL** formula $\varphi$ over $AP$ there is an 
**NBA** $A$ over the alphabet $2^AP$ such that 

$$Words(\varphi) = L_\omega(A)$$
From LTL to NBA

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

- $Words(\varphi) = L_\omega(A)$
- $size(A) = O(exp(|\varphi|))$
For each LTL formula \( \varphi \) over \( AP \) there is an NBA \( A \) over the alphabet \( 2^{AP} \) such that

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

**proof:** ... later ...


NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = ? \]
NBA for LTL formulas

$q_0 \xrightarrow{\text{true}} q_1 \xrightarrow{\neg a} q_F \xrightarrow{\text{true}} q_F$

$L_\omega(A) = \text{Words}(\bigcirc \neg a)$
NBA for LTL formulas

\[ L_\omega(A) = \text{Words}(\bigcirc \neg a) \]

\[ L_\omega(A) = ? \]
NBA for LTL formulas

\[ L_\omega(A) = \text{Words}(\bigcirc \neg a) \]

\[ L_\omega(A) = \text{Words}(a \lor b) \]
NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = \text{Words}(\circ \neg a) \]

\[ \mathcal{L}_\omega(A) = \text{Words}(a \lor b) \]

\[ \mathcal{L}_\omega(A) = ? \]
NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = \text{Words}(\bigcirc \neg a) \]

\[ \mathcal{L}_\omega(A) = \text{Words}(a \lor b) \]

\[ \mathcal{L}_\omega(A) = \text{Words}(\square a) \]
NBA for LTL formulas

\[ L_\omega(A) = ? \]
\( \mathcal{L}_\omega(A) = \text{Words}(\Box \Diamond a) \)
NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = \text{Words}(\square \Diamond a) \]

\[ \mathcal{L}_\omega(A) = ? \]
NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = \text{Words}(\Box \Diamond a) \]

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

e.g., \( \emptyset \emptyset \emptyset \emptyset \ldots = \emptyset^\omega \)
\[ (\{a\} \{b\})^\omega \]

\{ are accepted by \( A \) \}
NBA for LTL formulas

\[ \mathcal{L}_\omega(A) = \text{Words}(\Box \Diamond a) \]

\[ \mathcal{L}_\omega(A) = \text{Words}(\Box (a \rightarrow \Diamond b)) \]

e.g., \[ \emptyset \emptyset \emptyset \emptyset \ldots = \emptyset^\omega \]
\[ (\{a\} \{b\})^\omega \] are accepted by \( A \)
NBA for LTL formula

\[ \mathcal{L}_w(A) = ? \]
NBA for LTL formula

$L_\omega(\mathcal{A}) = \text{Words}(\Diamond \Box a)$
NBA for LTL formula

\[ \mathcal{L}_\omega(A) = \text{Words}(\Diamond \Box a) \]

possible runs for \( \{a\}^\omega \)

\begin{align*}
q_0 & \quad q_0 & \quad q_0 & \quad q_0 & \quad q_0 & \quad q_0 & \ldots & \text{not accepting} \\
q_0 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \ldots & \text{accepting} \\
q_0 & \quad q_0 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \ldots & \text{accepting} \\
q_0 & \quad q_0 & \quad q_0 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \quad q_1 & \ldots & \text{accepting} \\
\vdots & & & & & & & & \vdots & \end{align*}
NFA and NBA for safety properties
Let $\mathcal{A}$ be an NFA for the language of all bad prefixes for a safety property $E$. 
Let $A$ be an NFA for the language of all bad prefixes for a safety property $E$. Then:

$$\mathcal{L}_ω(A) = \overline{E} = (2^{AP})^ω \setminus E$$
Let \( A \) be an **NFA** for the language of all bad prefixes for a safety property \( E \). Then:

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

**Example:** \( E \equiv \text{“never a twice in a row”} \)
Let $\mathcal{A}$ be an NFA for the language of all bad prefixes for a safety property $E$. Then:

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

Example: $E \equiv \text{“never a twice in a row”}$

$$\varphi = \square(a \rightarrow \Diamond \neg a)$$
Let $\mathcal{A}$ be an NFA for the language of all bad prefixes for a safety property $E$. Then:

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

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

Example: $E \equiv \text{“never a twice in a row”}$

$$\varphi = \Box(a \rightarrow \Diamond \neg a)$$
Let $A$ be an **NFA** for the language of all bad prefixes for a safety property $E$. Then:

$$L_\omega(A) = \overline{E} = (2^{AP})^\omega \setminus E = \text{Words}(\neg \varphi)$$

**Wrong**, if $L(A)$ = language of minimal bad prefixes even if $A$ is a non-blocking DFA

Example: $E \equiv \text{“never a twice in a row”}$

Diagram:

- $q_0 \xrightarrow{\neg a} q_0$
- $q_0 \xrightarrow{a} q_1$
- $q_1 \xrightarrow{a} q_2$
- $q_2 \xrightarrow{\text{true}} q_3$
- $q_3 \xrightarrow{\text{true}} q_3$
- $q_1 \xrightarrow{\neg a} q_1$
- $q_2 \xrightarrow{\text{true}} q_2$

$L_\omega(A) = \emptyset$
Note: In the previous slides, you have to assume that the NFA has a self loop labeled true for each accept state (or else that the NFA is deterministic). If the NFA doesn't meet one of these conditions, then when considered as an NBA, it might not recognize E.
LTL model checking

finite transition system $\mathcal{T}$

LTL formula $\varphi$

NBA $\mathcal{A}$ for $\neg \varphi$

$LTL$ model checking

persistence checking

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

yes

no + counterexample
LTL model checking

finite transition system $\mathcal{T}$

LTL formula $\varphi$

later

NBA $\mathcal{A}$ for $\neg \varphi$

$LTL$ model checking

persistence checking

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

yes

no + counterexample
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, \( Q_0 \cap F = \emptyset \)
Recall: product transition system

\[ \mathcal{T} = (S, \text{Act}, \rightarrow, S_0, \text{AP}, L) \]

TS without terminal states

\[ \mathcal{A} = (Q, 2^{\text{AP}}, \delta, Q_0, F) \]

NBA or NFA

non-blocking, \( Q_0 \cap F = \emptyset \)

product-TS \( \mathcal{T} \otimes \mathcal{A} \overset{\text{def}}{=} (S \times Q, \text{Act}, \rightarrow', S'_0, \text{AP}', L') \)
Recall: product transition system

\[ T = (S, \text{Act}, \rightarrow, S_0, AP, L) \]  \text{TS without terminal states}

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

non-blocking, \( Q_0 \cap F = \emptyset \)

product-TS \( T \otimes A \overset{\text{def}}{=} (S \times Q, \text{Act}, \rightarrow', S'_0, AP', L') \)

initial states: \( S'_0 = \{ \langle s_0, q \rangle : s_0 \in S_0, q \in \delta(\text{Q}_0, \text{L}(s_0)) \} \)

labeling: \( AP' = Q, L'(\langle s, q \rangle) = \{ q \} \)
Recall: product transition system

\( \mathcal{T} = (S, Act, \rightarrow, S_0, AP, L) \)  \hspace{2cm} TS without terminal states

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

non-blocking,  \( Q_0 \cap F = \emptyset \)

product-TS  \( \mathcal{T} \otimes \mathcal{A} \)  \hspace{1cm} def  \hspace{1cm} (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:

\[
\begin{align*}
\langle s, q \rangle & \xrightarrow{\alpha} \langle s', q' \rangle \\
& \quad \land q' \in \delta(q, L(s'))
\end{align*}
\]
Example: LTL model checking

LTL formula $\varphi = \square \Diamond \text{green}$
Example: LTL model checking

\begin{align*}
\text{LTL formula } \varphi &= \Box \Diamond \text{green} \\
\text{NBA } \mathcal{A} \text{ for the complement } \\
\neg \varphi &\equiv \Diamond \Box \neg \text{green}
\end{align*}
Example: LTL model checking

**TS $T$**

- Red $q_0$ to Green $q_0$
- Green $q_0$ to Red $q_0$
- Red $q_0$ to Green $q_0$
- Green $q_0$ to Red $q_0$

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

**NBA $\mathcal{A}$** for the complement $\neg \varphi \equiv \Diamond \Box \neg \text{green}$

**TS $T$**

- Red $q_0$ to Green $q_0$
- Green $q_0$ to Red $q_0$
- Red $q_0$ to Green $q_0$
- Green $q_0$ to Red $q_0$

**NBA $\mathcal{A}$**

- Red $q_F$ to Green $q_F$
- Green $q_F$ to Red $q_F$
- Red $q_F$ to Green $q_F$
- Green $q_F$ to Red $q_F$

reachable fragment of the product TS $T \otimes \mathcal{A}$
Example: LTL model checking

LTL formula $\varphi = \square \lozenge \text{green}$

NBA $\mathcal{A}$ for the complement

$\neg \varphi \equiv \lozenge \square \neg \text{green}$

initial states:

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

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

$= \delta(q_0, \emptyset)$

$= \{q_0, q_F\}$
Example: LTL model checking

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

NBA $A$ for the complement

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

transition

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

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

$= \delta(q_0, \emptyset)$

$= \{q_0, q_F\}$
Example: LTL model checking

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

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

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 $T$

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

NBA $A$ for the complement $\neg \varphi \equiv \Diamond \Box \neg \text{green}$

$T \otimes A \models \Diamond \Box \neg F$

hence: $T \models \varphi$
Example: LTL model checking

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

“each (repeatedly) sent message will eventually be delivered”
Example: LTL model checking

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

“each (repeatedly) sent message will eventually be delivered”

$\mathcal{T} \not\models \varphi$
Example: LTL model checking

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

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

$\mathcal{T} \not\models \varphi$
Example: LTL model checking

TS $T$

- `start`
- `try_to_send`
- `lost`
- `delivered`

NBA $\mathcal{A}$ for $\neg \varphi \equiv \Diamond (\text{try} \land \square \neg \text{del})$

- `q_0` transitions:
  - `try` to `q_F`
  - `true` to `q_F`
  - `\neg \text{del}` to `true`

- `q_F` transitions:
  - `del` to `q_1`
  - `true` to `q_1`

- `q_1` transitions:
  - `try` to `q_1`
  - `lost` to `q_1`

reachable fragment of the product-TS
Example: LTL model checking

\( \text{NBA } \mathcal{A} \text{ for } \neg \varphi \equiv \Diamond (\text{try} \land \Box \neg \text{del}) \)

set of atomic propositions \( \mathcal{AP}' = \{q_0, q_1, q_F\} \)
Example: LTL model checking

TS $\mathcal{T}$

- start
- try_to_send
- lost
- delivered

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

- $q_0$
- $q_F$
- $q_1$
- true
- $\neg \text{del}$

$\mathcal{T} \otimes \mathcal{A} \not\models \Diamond \Box \neg F$
Example: LTL model checking

**TS \( \mathcal{T} \)**

- **start**
- **try_to_send**
- **lost**
- **delivered**

**NBA \( \mathcal{A} \) for**

\[ \neg \varphi \equiv \Diamond (\text{try} \land \Box \neg \text{del}) \]

\[ T \otimes A \not\models \Diamond \Box \neg F \]

**hence:** \( \mathcal{T} \not\models \varphi \)
LTL model checking

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

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

given: finite TS $T$, LTL-formula $\varphi$

question: does $T \models \varphi$ hold?

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

check whether $T \otimes A \models \Diamond \Box \neg F$
LTL model checking

given: finite TS $T$, LTL-formula $\varphi$

question: does $T \models \varphi$ hold?

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

check whether $T \otimes A \models \Diamond \Box \neg F$ ←
persistence checking
nested DFS
LTL model checking

given: finite TS $\mathcal{T}$, LTL-formula $\varphi$
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 $\mathcal{T} \otimes \mathcal{A} \models \lozenge \Box \neg F$

IF $\mathcal{T} \otimes \mathcal{A} \models \lozenge \Box \neg F$
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 $\lozenge \Box \neg F$
return “no” and $s_0 \ldots s_n \ldots s_n$
Complexity of LTL model checking

given: finite TS $T$, LTL-formula $\varphi$

question: does $T \models \varphi$ hold?

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

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

IF $T \otimes A \models \Diamond \Box \neg F$

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 $T \otimes A$ and $\Diamond \Box \neg F$

return “no” and $s_0 \ldots s_n \ldots s_n$

time complexity: $O(size(T) \cdot size(A))$
LTL model checking

finite transition system $\mathcal{T}$

LTL formula $\varphi$

NBA $\mathcal{A}$ for $\neg \varphi$

$LTL$ model checking

persistence checking

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

yes

no + counterexample
LTL model checking

finite transition system $\mathcal{T}$

LTL formula $\varphi$

NBA $\mathcal{A}$ for $\neg \varphi$

LTL model checking

persistence checking

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

yes

no + counterexample
For each LTL formula $\varphi$ there is an NBA $A$ s.t.

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

$$L_\omega(\mathcal{A}) = \text{Words}(\varphi)$$

**LTL** formula $\varphi$

**NBA** $\mathcal{A}$ s.t.

$$L_\omega(\mathcal{A}) = \text{Words}(\varphi)$$

nondeterministic Büchi automaton
From LTL to NBA

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

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

**LTL** formula $\varphi$

**GNBA** $\mathcal{G}$ s.t.

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

**NBA** $\mathcal{A}$ s.t.

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

generalized NBA

several acceptance sets

nondeterministic

Büchi automaton

1 acceptance set
For each LTL formula $\varphi$ there is an NBA $A$ s.t. $L_\omega(A) = \text{Words}(\varphi)$

1. LTL formula $\varphi$
2. GNBA $G$ s.t. $L_\omega(G) = \text{Words}(\varphi)$
3. NBA $A$ s.t. $L_\omega(A) = L_\omega(G)$

- generalized NBA $k$ acceptance sets
- $k$ copies of $G$
- nondeterministic Büchi automaton 1 acceptance set
idea: encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $\mathcal{G}$
idea: encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td></td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td></td>
</tr>
<tr>
<td>until $U$</td>
<td></td>
</tr>
</tbody>
</table>
**Encoding of LTL semantics in a GNBA**

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true, \neg, \wedge$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td></td>
</tr>
<tr>
<td>until $U$</td>
<td></td>
</tr>
</tbody>
</table>
**Encoding of LTL semantics in a GNBA**

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td></td>
</tr>
</tbody>
</table>
**Encodings of LTL semantics in a GNBA**

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>via expansion law</td>
</tr>
</tbody>
</table>
**Encoding of LTL semantics in a GNBA**

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic <em>true</em>, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>via expansion law</td>
</tr>
</tbody>
</table>

$$\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$$
**Encoding of LTL semantics in a GNBA**

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

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>via expansion law</td>
</tr>
</tbody>
</table>

$$\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$$

encoded in the states
idea: encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>via expansion law</td>
</tr>
</tbody>
</table>

$\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$

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 $\varphi$ by appropriate components of the GNBA $\mathcal{G}$

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>propositional logic</strong> true, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>expansion law, <strong>least fixed point</strong></td>
</tr>
</tbody>
</table>

$$\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$$

- encoded in the states
- encoded in the transition relation
- acceptance condition
LTL formula $\varphi \leadsto GNBA \ G$ for $\text{Words}(\varphi)$
LTL $\leadsto$ GNBA

LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

states of $G \equiv$ (certain) sets of subformulas of $\varphi$
LTL $\leadsto$ GNBA

LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

states of $G \equiv$ (certain) sets of subformulas of $\varphi$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$
LTL $\rightsquigarrow$ GNBA

LTL formula $\varphi$ $\rightsquigarrow$ GNBA $G$ for $Words(\varphi)$

states of $G \equiv$ (certain) sets of subformulas of $\varphi$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in Words(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

$$A_0 \ A_1 \ A_2 \ A_3 \ \ldots \ \in \ Words(\varphi)$$
LTL formula $\varphi \rightsquigarrow$ GNBA $\mathcal{G}$ for $\text{Words}(\varphi)$

states of $\mathcal{G} \overset{\triangleq}{=} \text{(certain) sets of subformulas of } \varphi$ s.t. each word $\sigma = \mathcal{A}_0 \mathcal{A}_1 \mathcal{A}_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $\mathcal{B}_0 \mathcal{B}_1 \mathcal{B}_2 \ldots$ in $\mathcal{G}$

where $\mathcal{B}_i = \{ \psi \in \text{cl}(\varphi) : \mathcal{A}_i \mathcal{A}_{i+1} \mathcal{A}_{i+2} \ldots \models \psi \}$
LTL formula $\varphi \leadsto$ GNBA $\mathcal{G}$ for $\text{Words}(\varphi)$

states of $\mathcal{G} \cong$ (certain) sets of subformulas of $\varphi$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

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

set of subformulas of $\varphi$ and their negations
LTL formula \( \varphi \) \( \Leftrightarrow \) GNBA \( G \) for Words(\( \varphi \))

states of \( G \) \( \cong \) (certain) sets of subformulas of \( \varphi \)

s.t. each word \( \sigma = A_0 \ A_1 \ A_2 \ldots \in \text{Words}(\varphi) \) can be extended to an accepting run \( B_0 \ B_1 \ B_2 \ldots \) in \( G \)

Example: \( \varphi = a \ U(\neg a \ \& \ b) \)
LTL formula $\varphi \leadsto$ GNBA $\mathcal{G}$ for $\text{Words}(\varphi)$

states of $\mathcal{G} \triangleq$ (certain) sets of subformulas of $\varphi$

s.t. each word $\sigma = A_0 \ A_1 \ A_2 \ldots \in \text{Words}(\varphi)$ can be
extended to an accepting run $B_0 \ B_1 \ B_2 \ldots \ $ in $\mathcal{G}$

Example: $\varphi = a \bigcup (\neg a \land b)$

$$\begin{align*}
\{a\} & \quad \{a\} & \quad \{a, b\} & \quad \{b\} & \quad \emptyset & \quad \emptyset & \ldots \models \varphi
\end{align*}$$
LTL $\leadsto$ GNBA

LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

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

Example: $\varphi = a \mathcal{U}(\neg a \land b)$

\[
\begin{array}{ccccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset & \ldots \\
B_0 & B_1 & B_2 & B_3 & B_4 & B_5
\end{array}
\]

$\models \varphi$
LTL $\rightsquigarrow$ GNBA

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

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

Example: $\varphi = a U (\neg a \land b)$ $\psi = \neg a \land b$

where the $B_i$’s are subsets of
$\{a, \neg a, b, \neg b, \psi, \neg \psi, \varphi, \neg \varphi\}$
LTL $\leadsto$ GNBA

LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

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

Example: $\varphi = a \mathbf{U}(\neg a \land b) \quad \psi = \neg a \land b$

$$\begin{align*}
\{a\} & \quad \{a\} & \quad \{a, b\} & \quad \{b\} & \quad \emptyset & \quad \emptyset & \quad \ldots & \models \varphi
\end{align*}$$

just for better readability:
tuple rather than set notation
LTL $\rightsquigarrow$ GNBA

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

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

Example: $\varphi = a U (\neg a \land b)$  $\psi = \neg a \land b$

$$\begin{array}{cccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset & \ldots
\end{array}$$

$\models \varphi$
LTL formula $\varphi \rightsquigarrow$ GNBA $G$ for $\text{Words}(\varphi)$

states of $G \cong$ (certain) sets of subformulas of $\varphi$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b)$, $\psi = \neg a \land b$

\[
\begin{align*}
\{a\} & \downarrow \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset & \ldots & \models \varphi
\end{align*}
\]

\[
\begin{array}{ccccccc}
a & \neg b & \neg \psi & \varphi \\
\downarrow & \downarrow & \downarrow & \\
\neg b & \neg \psi & \varphi & \varphi
\end{array}
\]
LTL $\rightsquigarrow$ GNBA

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

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

Example: $\varphi = a \cup (\neg a \land b)$, $\psi = \neg a \land b$

\[
\begin{array}{cccccc}
\{a\} & \downarrow & \{a\} & \downarrow & \{a, b\} & \downarrow \\
\downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 a & \neg b & \neg \psi & \neg \varphi & a & \neg a \\
\neg \psi & \varphi & \varphi & \varphi & \psi & \varphi
\end{array}
\]

$\varphi \models \varphi$
LTL $\rightarrow$ GNBA

LTL formula $\varphi$ $\rightarrow$ GNBA $\mathcal{G}$ for $\text{Words}(\varphi)$

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

Example: $\varphi = a U (\neg a \land b)$ $\quad \psi = \neg a \land b$

$$\begin{align*}
\{a\} & \downarrow \{a\} & \downarrow \{a, b\} & \downarrow \{b\} & \downarrow \emptyset & \downarrow \emptyset & \ldots & \models \varphi
\end{align*}$$

$$\begin{array}{ccccccc}
a & \neg b & \neg \psi & \varphi \\
\neg a & \neg b & \neg \psi & \varphi \\
\neg a & \neg b & \neg \psi & \varphi \\
\neg a & \neg b & \neg \psi & \varphi \\
\end{array}$$
LTL formula $\varphi \leadsto \text{GNBA } G$ for $\text{Words}(\varphi)$

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

Example: $\varphi = a U (\neg a \land b) \quad \psi = \neg a \land b$

\[
\begin{align*}
\{a\} & \quad \{a\} & \quad \{a, b\} & \quad \{b\} & \quad \emptyset & \quad \emptyset & \ldots & \models \varphi \\
a & \quad a & \quad a & \quad \neg a & \quad \neg a & \quad \neg a & \ldots \\
\neg b & \quad \neg b & \quad b & \quad b & \quad \neg b & \quad \neg b & \ldots \\
\neg \psi & \quad \neg \psi & \quad \neg \psi & \quad \psi & \quad \psi & \quad \psi & \ldots \\
\varphi & \quad \varphi & \quad \varphi & \quad \varphi & \quad \varphi & \quad \varphi & \ldots
\end{align*}
\]
Let $\varphi$ be an LTL formula. Then:

$$\text{subf}(\varphi) \overset{\text{def}}{=} \text{set of all subformulas of } \varphi$$
Let $\varphi$ be an LTL formula. Then:

\[
\text{subf}(\varphi) \overset{\text{def}}{=} \text{set of all subformulas of } \varphi
\]

\[
\text{cl}(\varphi) \overset{\text{def}}{=} \text{subf}(\varphi) \cup \{ \neg \psi : \psi \in \text{subf}(\varphi) \}
\]

where $\psi$ and $\neg \neg \psi$ are identified.
Let $\varphi$ be an LTL formula. Then:

\[
\text{subf}(\varphi) \overset{\text{def}}{=} \text{set of all subformulas of } \varphi
\]

\[
\text{cl}(\varphi) \overset{\text{def}}{=} \text{subf}(\varphi) \cup \{\neg \psi : \psi \in \text{subf}(\varphi)\}
\]

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

Example: If $\varphi = a \mathcal{U}(\neg a \land b)$ then

\[
\text{cl}(\varphi) = \{a, b, \neg a \land b, \varphi\} \cup \{\neg a, \neg b, \neg (\neg a \land b), \neg \varphi\}
\]
Let $\varphi$ be an LTL formula. Then:

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

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

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

**Example:** if $\varphi = a \cup (\neg a \land b)$ then

$\text{cl}(\varphi) = \{a, b, \neg a \land b, \varphi\} \cup \{\neg a, \neg b, \neg(\neg a \land b), \neg \varphi\}$

**Example:** if $\varphi' = \square a$
Closure of LTL formulas

Let $\varphi$ be an LTL formula. Then:

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

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

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

**Example:** if $\varphi = a \cup (\neg a \land b)$ then

$$\text{cl}(\varphi) = \{a, b, \neg a \land b, \varphi\} \cup \{\neg a, \neg b, \neg (\neg a \land b), \neg \varphi\}$$

**Example:** if $\varphi' = \Box a = \neg \Diamond \neg a = \neg (true \cup \neg a)$
Let $\varphi$ be an LTL formula. Then:

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

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

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

**Example:** if $\varphi = a \cup (\neg a \land b)$ then

$$\text{cl}(\varphi) = \{a, b, \neg a \land b, \varphi\} \cup \{\neg a, \neg b, \neg (\neg a \land b), \neg \varphi\}$$

**Example:** if $\varphi' = \Box a = \neg \Diamond \neg a = \neg (\text{true} \cup \neg a)$ then

$$\text{cl}(\varphi') = \{a, \neg a, \text{true}, \neg \text{true}, \Box a, \neg \Box a\}$$
Elementary formula-sets

LTLMC3.2-50
Let $B \subseteq cl(\varphi)$. $B$ is called elementary if:
Let $B \subseteq \text{cl}(\varphi)$. $B$ 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 $U$: $\forall \varphi: B \subseteq \text{cl}(\varphi)$
Let $B \subseteq \text{cl}(\varphi)$. $B$ 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 $U$:
Let $B \subseteq \text{cl}(\varphi)$. $B$ is called elementary if:

(1) $B$ is consistent w.r.t. propositional logic
   - if $\psi \in B$ then $\neg \psi \notin B$
   - if $\psi_1 \land \psi_2 \in B$ 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 $\text{U}$:
Elementary formula-sets

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

1. $B$ is consistent w.r.t. propositional logic
   - if $\psi \in B$ then $\neg \psi \notin B$
   - if $\psi_1 \land \psi_2 \in B$ then $\neg \psi_1 \notin B$ and $\neg \psi_2 \notin B$
   - if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg (\psi_1 \land \psi_2) \notin B$

2. $B$ is maximal consistent

3. $B$ is locally consistent with respect to until $\mathbf{U}$:
Let $B \subseteq \text{cl}(\varphi)$. $B$ is called elementary if:

1. $B$ is consistent w.r.t. propositional logic
   - if $\psi \in B$ then $\neg \psi \notin B$
   - if $\psi_1 \land \psi_2 \in B$ then $\neg \psi_1 \notin B$ and $\neg \psi_2 \notin B$
   - if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg (\psi_1 \land \psi_2) \notin B$
   - if $\text{false} \in \text{cl}(\varphi)$ then $\text{false} \notin B$

2. $B$ is maximal consistent

3. $B$ is locally consistent with respect to until $U$: 
Let $B \subseteq \text{cl}(\varphi)$. $B$ is called elementary if:

1. $B$ is consistent w.r.t. propositional logic
   - if $\psi \in B$ then $\neg \psi \notin B$
   - if $\psi_1 \land \psi_2 \in B$ then $\neg \psi_1 \notin B$ and $\neg \psi_2 \notin B$
   - if $\psi_1 \in B$ and $\psi_2 \in B$ then $\neg (\psi_1 \land \psi_2) \notin B$
   - if $false \in \text{cl}(\varphi)$ then $false \notin B$

2. $B$ is maximal consistent
   - if $\psi \in \text{cl}(\varphi) \setminus B$ then $\neg \psi \in B$

3. $B$ is locally consistent with respect to until $U$: 

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

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

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

(2) $B$ is maximal consistent

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

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

- if $\psi_1 U \psi_2 \in B$ and $\neg \psi_2 \in B$ then $\neg \psi_1 \notin B$
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 \( \psi_1 \land \psi_2 \in B \) then \( \neg\psi_1 \notin B \) and \( \neg\psi_2 \notin B \)
   - if \( \psi_1 \in B \) and \( \psi_2 \in B \) then \( \neg(\psi_1 \land \psi_2) \notin B \)
   - if \text{false} \in cl(\varphi) \) then \text{false} \notin B \)

2. \( B \) is maximal consistent
   - if \( \psi \in cl(\varphi) \setminus B \) then \( \neg\psi \notin B \)

3. \( B \) is locally consistent with respect to until \( \text{U} \):
   - if \( \psi_1 \text{U} \psi_2 \in B \) and \( \neg\psi_2 \in B \) then \( \neg\psi_1 \notin B \)
   - if \( \psi_2 \in B \) and \( \psi_1 \text{U} \psi_2 \in cl(\varphi) \) then \( \neg(\psi_1 \text{U} \psi_2) \notin B \)
Elementary formula-sets

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

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

<table>
<thead>
<tr>
<th>$\psi \notin B$</th>
<th>iff</th>
<th>$\neg \psi \in B$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\psi_1 \land \psi_2 \in B$</td>
<td>iff</td>
<td>$\psi_1 \in B$ and $\psi_2 \in B$</td>
</tr>
<tr>
<td>$true \in cl(\varphi)$</td>
<td>implies</td>
<td>$true \in B$</td>
</tr>
</tbody>
</table>

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

if $\psi_1 U \psi_2 \in B$ and $\psi_2 \notin B$ then $\psi_1 \in B$

if $\psi_2 \in B$ then $\psi_1 U \psi_2 \in B$
Let $\phi = a \mathcal{U}(\neg a \land b)$.

$B_1 = \{a, b, \neg a \land b, \phi\}$
Let $\varphi = a \mathcal{U}(\neg a \land b)$.

$B_1 = \{a, b, \neg a \land b, \varphi\}$ not elementary propositional inconsistent
Let \( \varphi = a \cup (\neg a \land b) \).

\[ B_1 = \{ a, b, \neg a \land b, \varphi \} \quad \text{not elementary} \]

\[ B_2 = \{ \neg a, b, \varphi \} \quad \text{propositional inconsistent} \]
Elementary or not?

Let $\varphi = a U (\neg a \land b)$.

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

not elementary
propositional inconsistent

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

not elementary, not maximal
as
$\neg (\neg a \land b) \notin B_2$
Elementary or not?

Let $\varphi = a U (\neg a \land b)$.

$B_1 = \{ a, b, \neg a \land b, \varphi \}$ not elementary

propositional inconsistent

$B_2 = \{ \neg a, b, \varphi \}$ not elementary, not maximal

as $\neg a \land b \notin B_2$

$\neg (\neg a \land b) \notin B_2$

$B_3 = \{ \neg a, b, \neg a \land b, \neg \varphi \}$
Let $\varphi = a \mathbf{U} (\neg a \land b)$.

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

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

$B_3 = \{\neg a, b, \neg a \land b, \neg \varphi\}$ not elementary
not locally consistent for $\mathbf{U}$
Elementary or not?

Let \( \varphi = a U (\neg a \land b) \).

\[ B_1 = \{ a, b, \neg a \land b, \varphi \} \quad \text{not elementary} \]

\[ B_2 = \{ \neg a, b, \varphi \} \quad \text{propositional inconsistent} \]

\[ B_3 = \{ \neg a, b, \neg a \land b, \neg \varphi \} \quad \text{not elementary} \]

\[ B_4 = \{ \neg a, \neg b, \neg (\neg a \land b), \neg \varphi \} \quad \text{not locally consistent for } U \]

\[ \neg (\neg a \land b) \notin B_2 \]

\[ \neg \varphi \notin B_2 \]
Elementary or not?

Let \( \varphi = a \lor (\neg a \land b) \).

\[ B_1 = \{a, b, \neg a \land b, \varphi\} \quad \text{not elementary} \]

propositional inconsistent

\[ B_2 = \{\neg a, b, \varphi\} \quad \text{not elementary, not maximal} \]

as \( \neg a \land b \notin B_2 \)

\( \neg (\neg a \land b) \notin B_2 \)

\[ B_3 = \{\neg a, b, \neg a \land b, \neg \varphi\} \quad \text{not elementary} \]

not locally consistent for \( \lor \)

\[ B_4 = \{\neg a, \neg b, \neg (\neg a \land b), \neg \varphi\} \quad \text{elementary} \]
Example: elementary formula-sets

closure $\text{cl}(\varphi)$:

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

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

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

For $\varphi = a \cup (\neg a \land b)$, the elementary sets are:

1. $\{a, b, \neg(\neg a \land b), \varphi\}$
2. $\{a, \neg b, \neg(\neg a \land b), \varphi\}$
3. $\{\neg a, b, \neg a \land b, \varphi\}$
4. $\{a, b, \neg(\neg a \land b), \neg \varphi\}$
5. $\{a, \neg b, \neg(\neg a \land b), \neg \varphi\}$
6. $\{\neg a, b, \neg a \land b, \neg \varphi\}$
7. $\{\neg a, b, \neg(\neg a \land b), \neg \varphi\}$
### Encoding of LTL semantics in a GNBA

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$:

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>expansion law, least fixed point</td>
</tr>
</tbody>
</table>

\[
\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))
\]

- $\psi_1 U \psi_2$ encoded in the states
- $\psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$ encoded in the transition relation
- $\psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))$ acceptance condition
**Encoding of LTL semantics in a GNBA**

**idea:** encode the semantics of the operators appearing in $\varphi$ by appropriate components of the GNBA $G$:

<table>
<thead>
<tr>
<th>semantics of ...</th>
<th>encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>propositional logic $true$, $\neg$, $\land$</td>
<td>in the states</td>
</tr>
<tr>
<td>next $\bigcirc$</td>
<td>in the transition relation</td>
</tr>
<tr>
<td>until $U$</td>
<td>expansion law, least fixed point</td>
</tr>
</tbody>
</table>

$$
\psi_1 U \psi_2 \equiv \psi_2 \lor (\psi_1 \land \bigcirc(\psi_1 U \psi_2))
$$
GNBA for LTL-formula $\varphi$
GNBA for LTL-formula $\varphi$

$G = (Q, 2^AP, \delta, Q_0, F)$
GNBA for LTL-formula $\varphi$

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

state space: $Q = \{ B \subseteq \text{cl}(\varphi) : B \text{ is elementary} \}$
GNBA for LTL-formula $\varphi$

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

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 $\varphi$

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

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

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 $\delta(B, A) = \emptyset$
GNBA for LTL-formula $\varphi$

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

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 $\delta(B, A) = \emptyset$

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

\begin{align*}
\bigcirc \psi &\in B \text{ iff } \psi \in B' \\
\psi_1 \cup \psi_2 &\in B \text{ iff } (\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')
\end{align*}
GNBA for LTL-formula $\varphi$

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

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

$$\bigcirc \psi \in B \text{ iff } \psi \in B'$$
$$\psi_1 U \psi_2 \in B \text{ iff } (\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 U \psi_2 \in B')$$

acceptance set $F = \{F_{\psi_1 U \psi_2} : \psi_1 U \psi_2 \in cl(\varphi)\}$
GNBA for LTL-formula $\varphi$

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

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

\[
\begin{align*}
\bigcirc \psi & \in B \quad \text{iff} \quad \psi \in B' \\
\psi_1 \cup \psi_2 & \in B \quad \text{iff} \quad (\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')
\end{align*}
\]

acceptance set $F = \{ F_{\psi_1 \cup \psi_2} : \psi_1 \cup \psi_2 \in cl(\varphi) \}$

where $F_{\psi_1 \cup \psi_2} = \{ B \in Q : \psi_1 \cup \psi_2 \notin B \lor \psi_2 \in B \}$
Example: GNBA for $\varphi = \Box a$
Example: GNBA for $\varphi = \bigcirc a$
Example: GNBA for $\varphi = \Diamond a$

initial states: formula-sets $B$ with $\Diamond a \in B$
Example: GNBA for $\varphi = \Diamond a$

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

transition relation:
if $\Diamond a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$
Example: GNBA for $\varphi = \Diamond a$

Initial states: formula-sets $B$ with $\Diamond a \in B$

Transition relation:
if $\Diamond a \in B$ 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 $\varphi = \Box a$

Initial states: formula-sets $B$ with $\Box a \in B$

Transition relation:
- If $\Box a \in B$ then $\delta(B, B \cap \{a\}) = \{B' : a \in B'\}$
- If $\Box a \notin B$ then $\delta(B, B \cap \{a\}) = \{B' : a \notin B'\}$
Example: GNBA for $\varphi = \Diamond a$

Initial states: formula-sets $B$ with $\Diamond a \in B$

Transition relation:
- If $\Diamond a \in B$ then $\delta(B, B \cap \{ a \}) = \{ B' : a \in B' \}$
- If $\Diamond a \notin B$ then $\delta(B, B \cap \{ a \}) = \{ B' : a \notin B' \}$
Example: GNBA for $\varphi = \square a$

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

transition relation:

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

if $\square a \notin B$ then $\delta(B, B \cap \{a\}) = \{B' : a \notin B'\}$
Example: GNBA for $\varphi = \Diamond a$

set of acceptance sets:
Example: GNBA for $\varphi = \Diamond a$

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

*hence*: all words having an *infinite run* are accepted
Example: GNBA for $\varphi = \Diamond a$

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

$\emptyset \quad \{a\} \quad \{a\} \quad \emptyset \quad \emptyset \quad \ldots \quad \models \Diamond a$
Example: GNBA for $\varphi = \bigcirc a$

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

$\emptyset \quad \{a\} \quad \{a\} \quad \emptyset \quad \emptyset \quad \ldots \quad \models \bigcirc a$
Example: GNBA for $\varphi = \Diamond a$

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

$$\emptyset \quad \{a\} \quad \{a\} \quad \emptyset \quad \emptyset \quad \ldots \quad \models \Diamond a$$
Example: GNBA for $\varphi = \Diamond a$

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

$\emptyset \rightarrow \{a\} \rightarrow \{a\} \rightarrow \emptyset \rightarrow \emptyset \ldots \models \Diamond a$
Example: GNBA for $\varphi = \Diamond a$

set of acceptance sets: $F = \emptyset$

$\emptyset \downarrow \{a\} \downarrow \{a\} \emptyset \emptyset \ldots \models \Diamond a$
Example: GNBA for $\varphi = \Diamond a$

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

$\emptyset \downarrow \{a\} \downarrow \{a\} \downarrow \emptyset \downarrow \emptyset \ldots \models \Diamond a$

accepting run
Soundness of the GNBA for $\varphi = \square a$

for all words $\sigma = A_0 A_1 A_2 A_3 \ldots \in L_\omega(G)$: $A_1 = \{a\}$
Soundness of the GNBA for $\varphi = \Diamond a$

for all words $\sigma = A_0 A_1 A_2 A_3 \ldots \in \mathcal{L}_w(G)$: $A_1 = \{a\}$

proof:
Soundness of the GNBA for \( \varphi = \bigcirc a \) for all words \( \sigma = A_0 A_1 A_2 A_3 \ldots \in L_\omega(G) \): \( A_1 = \{a\} \)

proof: Let \( B_0 B_1 B_2 \ldots \) be an accepting run for \( \sigma \).
Soundness of the GNBA for $\varphi = \Diamond a$

for all words $\sigma = A_0 A_1 A_2 A_3 \ldots \in L_\omega(G)$:  $A_1 = \{a\}$

proof: Let $B_0 B_1 B_2 \ldots$ be an accepting run for $\sigma$.

$\implies \Diamond a \in B_0$
Soundness of the GNBA for \( \varphi = \Box a \)

For all words \( \sigma = A_0 A_1 A_2 A_3 \ldots \in \mathcal{L}_\omega(G) \): \( A_1 = \{ a \} \)

**proof:** Let \( B_0 B_1 B_2 \ldots \) be an accepting run for \( \sigma \).

\[ \implies \Box a \in B_0 \text{ and therefore } a \in B_1 \]
Soundness of the GNBA for $\varphi = \Diamond a$

for all words $\sigma = A_0 A_1 A_2 A_3 \ldots \in \mathcal{L}_\omega(G)$: $A_1 = \{a\}$

proof: Let $B_0 B_1 B_2 \ldots$ be an accepting run for $\sigma$.

$\Rightarrow$ $\Diamond a \in B_0$ and therefore $a \in B_1$

$\Rightarrow$ the outgoing edges of $B_1$ have label $\{a\}$
Soundness of the GNBA for $\varphi = \Diamond a$

For all words $\sigma = A_0 A_1 A_2 A_3 \ldots \in \mathcal{L}_\omega(G)$: $A_1 = \{a\}$

Proof: Let $B_0 B_1 B_2 \ldots$ be an accepting run for $\sigma$.

$\implies \Diamond a \in B_0$ and therefore $a \in B_1$

$\implies$ the outgoing edges of $B_1$ have label $\{a\}$

$\implies \{a\} = B_1 \cap AP = A_1$
Example: GNBA for $\varphi = a U b$
Example: GNBA for $\varphi = a \mathbin{U} b$

\[
\begin{align*}
a, b, a \mathbin{U} b & \quad \neg a, \neg b, \neg (a \mathbin{U} b) \\
\neg a, \neg b, a \mathbin{U} b & \quad a, \neg b, \neg (a \mathbin{U} b) \\
\neg a, b, a \mathbin{U} b & \\
\end{align*}
\]

locally inconsistent:
\[
\begin{align*}
\{ a, b, \neg (a \mathbin{U} b) \} \\
\{ \neg a, b, \neg (a \mathbin{U} b) \} \\
\{ \neg a, \neg b, a \mathbin{U} b \}
\end{align*}
\]
Example: GNBA for $\varphi = a \mathcal{U} b$

- $a, b, a \mathcal{U} b$
- $a, \neg b, a \mathcal{U} b$
- $\neg a, b, a \mathcal{U} b$

Initial states: $B$ with $\varphi = a \mathcal{U} b \in B$
Example: GNBA for $\varphi = a \text{ U } b$

\[ \rightarrow \quad a, b, a \text{ U } b \quad \neg a, \neg b, \neg (a \text{ U } b) \]

\[ \rightarrow \quad a, \neg b, a \text{ U } b \quad a, \neg b, \neg (a \text{ U } b) \]

\[ \rightarrow \quad \neg a, b, a \text{ U } b \]

initial states: $B$ with $\varphi = a \text{ U } b \in B$
Example: GNBA for $\varphi = a U b$

$\rightarrow a, b, a U b \quad \neg a, \neg b, \neg(a U b)$

$\rightarrow a, \neg b, a U b \quad a, \neg b, \neg(a U b)$

$\rightarrow \neg a, b, a U b$

initial states: $B$ with $\varphi = a U b \in B$

acceptance condition: just one set of accept states

$F = \text{set of all } B \text{ with } \varphi \not\in B \text{ or } b \in B$
Example: GNBA for $\varphi = a \cup b$ ← NBA

$\longrightarrow \quad a, b, a \cup b$  

$\longrightarrow \quad a, \neg b, a \cup b$  

$\longrightarrow \quad \neg a, b, a \cup b$

initial states: $B$ with $\varphi = a \cup b \in B$

acceptance condition: just one set of accept states

$F = \text{set of all } B \text{ with } \varphi \notin B \text{ or } b \in B$
Example: (G)NBA for $\varphi = a \mathbf{U} b$

initial states: $B$ with $\varphi = a \mathbf{U} b \in B$

acceptance condition: just one set of accept states

$F = \text{set of all } B \text{ with } \varphi \not\in B \text{ or } b \in B$
Example: (G)NBA for $\varphi = a \cup b$

$\rightarrow a, b, a \cup b$

$\rightarrow a, \neg b, a \cup b$

$\rightarrow \neg a, b, a \cup b$

$\rightarrow \neg a, \neg b, \neg (a \cup b)$

$\rightarrow a, \neg b, \neg (a \cup b)$

$\rightarrow \neg a, b, a \cup b$

$\neg a, \neg b, \neg (a \cup b)$

transition relation: $B' \in \delta(B, B \cap AP)$ iff

$a \cup b \in B \iff (b \in B \lor (a \in B \land a \cup b \in B'))$
Example: (G)NBA for $\varphi = a \cup b$

Transition relation: $B' \in \delta(B, B \cap AP)$ iff

$$a \cup b \in B \iff (b \in B \lor (a \in B \land a \cup b \in B'))$$
Example: (G)NBA for \( \varphi = a \mathbin{U} b \)

transition relation: \( B' \in \delta(B, B \cap AP) \) iff

\[
a \mathbin{U} b \in B \iff (b \in B \lor (a \in B \land a \mathbin{U} b \in B'))
\]
Example: (G)NBA for $\varphi = a \cup b$

transition relation: $B' \in \delta(B, B \cap AP)$ iff

$$a \cup b \in B \iff (b \in B \lor (a \in B \land a \cup b \in B'))$$
Example: (G)NBA for $\varphi = a \mathsf{U} b$

transition relation: $B' \in \delta(B, B \cap \mathsf{AP})$ iff

$$a \mathsf{U} b \in B \iff (b \in B \lor (a \in B \land a \mathsf{U} b \in B'))$$
Example: (G)NBA for $\varphi = a \mathbin{U} b$

transition relation: $B' \in \delta(B, B \cap \text{AP})$ iff

$$a \mathbin{U} b \in B \iff (b \in B \lor (a \in B \land a \mathbin{U} b \in B'))$$
Example: (G)NBA for $\varphi = a \text{ U } b$
Example: (G)NBA for $\varphi = a \cup b$
Example: (G)NBA for $\varphi = a \cup b$

$$\{a\} \quad \{a\} \quad \{a, b\} \quad \emptyset \quad \emptyset \quad \emptyset \quad \vdash a \cup b$$
Example: (G)NBA for $\varphi = a \cup b$
Example: (G)NBA for $\varphi = a \mathbin{U} b$

\[
\begin{align*}
\{a\} & \quad \{a\} & \quad \{a, b\} & \quad \emptyset & \quad \emptyset & \quad \emptyset & \ldots & \models a \mathbin{U} b
\end{align*}
\]
Example: $(G)\text{NBA}$ for $\varphi = a \cup b$

$$\varphi = a \cup b$$

$$\neg a, \neg b, \neg (a \cup b)$$

$$a, \neg b, \neg (a \cup b)$$

$$\neg a, b, a \cup b$$

$$a, b, a \cup b$$

$$\neg a, b, a \cup b$$

$$\neg a \land \neg b$$

$$a \land b$$

$$\{a\} \downarrow \{a\} \downarrow \{a, b\} \downarrow \emptyset \downarrow \emptyset \downarrow \emptyset \downarrow \emptyset \downarrow \emptyset \ldots \models a \cup b$$
Example: (G)NBA for $\varphi = a \cup b$

The diagram illustrates the (G)NBA construction for the formula $\varphi = a \cup b$. The states and transitions are labeled with formulas and sets of variables, highlighting the path from initial states to accepting states through the calculation of $\neg a \land \neg b$.

The bottom part of the diagram shows the implication $\varnothing \rightarrow a \cup b$ by tracing through the state transitions and formula evaluations.
Example: (G)NBA for \( \varphi = a \mathbin{U} b \)

\[
\begin{align*}
\{ a \} & \quad \{ a \} & \quad \{ a, b \} & \quad \emptyset & \quad \emptyset & \quad \emptyset & \quad \ldots & \models a \mathbin{U} b
\end{align*}
\]
Example: (G)NBA for \( \varphi = a \mathbin{U} b \)
Example: (G)NBA for $\varphi = a \mathop{U} b$

$\{a\} \{a\} \{a\} \{a\} \ldots \not\models \varphi$
Example: (G)NBA for $\varphi = a \mathbin{U} b$

$\varphi = a \land b$

$\neg a, \neg b, \neg (a \mathbin{U} b)$

$\neg a \mathbin{U} b$

$q_0$

$\neg a, \neg b, a \mathbin{U} b$

$\{a\} \{a\} \{a\} \{a\} \ldots \not\models \varphi$

only 1 infinite run: $q_0 q_0 q_0 \ldots$
Example: (G)NBA for $\varphi = a \cup b$

Only 1 infinite run: $q_0 \ q_0 \ q_0 \ \ldots$
Example: (G)NBA for $\varphi = a \cup b$

only 1 infinite run: $q_0 \ q_0 \ q_0 \ldots$ not accepting
GNBA for LTL-formula $\varphi$

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

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

$\bigcirc \psi \in B$ iff $\psi \in B'$

$\psi_1 \cup \psi_2 \in B$ iff $(\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')$

acceptance set $F = \{F_{\psi_1 \cup \psi_2} : \psi_1 \cup \psi_2 \in cl(\varphi)\}$

where $F_{\psi_1 \cup \psi_2} = \{B \in Q : \psi_1 \cup \psi_2 \notin B \lor \psi_2 \in B\}$
.... of the construction LTL formula $\varphi \leadsto \text{GNBA } G$
Let \( \varphi \) be an LTL-formula and \( G = (Q, 2^{AP}, \delta, Q_0, F) \) be the constructed GNBA.

Claim: \( \text{Words}(\varphi) = \mathcal{L}_\omega(G) \)
Soundness

Let $\varphi$ be an LTL-formula and $G = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$ be the constructed GNBA.

**Claim:** $\text{Words}(\varphi) = L_\omega(G)$

“$\subseteq$” show: each infinite word $A_0 A_1 A_2 \ldots \in (2^{AP})^\omega$

with $A_0 A_1 A_2 \ldots \models \varphi$

has an accepting run in $G$
Let $\varphi$ be an LTL-formula and $G = (Q, 2^{AP}, \delta, Q_0, F)$ be the constructed GNBA.

Claim: $Words(\varphi) = L_\omega(G)$

"$\subseteq$" show: each infinite word $A_0 A_1 A_2 \ldots \in (2^{AP})^\omega$

with $A_0 A_1 A_2 \ldots \models \varphi$

has an accepting run in $G$

"$\supseteq$" show: for all infinite words $A_0 A_1 A_2 \ldots \in L_\omega(G)$:

$A_0 A_1 A_2 \ldots \models \varphi$
Soundness

Let $\varphi$ be an LTL-formula and $G = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$ be the constructed GNBA.

Claim: $\text{Words}(\varphi) = \mathcal{L}_\omega(G)$

“$\subseteq$” show: each infinite word $A_0 A_1 A_2 \ldots \in (2^{AP})^\omega$
with $A_0 A_1 A_2 \ldots \models \varphi$
has an accepting run in $G$

“$\supseteq$” show: for all infinite words $A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:
$A_0 A_1 A_2 \ldots \models \varphi$
Accepting runs for the elements of $\text{Words}(\varphi)$

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

states of $G \cong$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$
Accepting runs for the elements of $\text{Words}(\varphi)$

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

states of $G \cong$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$
Accepting runs for the elements of $\text{Words}(\varphi)$

LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

states of $G \models$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be
extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b)$
Accepting runs for the elements of $\text{Words}(\varphi)$

LTL formula $\varphi \leadsto \text{GNBA } G$ for $\text{Words}(\varphi)$

states of $G \equiv$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b)$

$$\{a\} \quad \{a\} \quad \{a, b\} \quad \{b\} \quad \emptyset \quad \emptyset \quad \ldots \models \varphi$$
Accepting runs for the elements of \( \text{Words}(\varphi) \)

LTL formula \( \varphi \leadsto \text{GNBA } G \) for \( \text{Words}(\varphi) \)

states of \( G \) \( \equiv \) elementary formula-sets \( B \subseteq \text{cl}(\varphi) \)

s.t. each word \( \sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi) \) can be extended to an accepting run \( B_0 B_1 B_2 \ldots \) in \( G \)

Example: \( \varphi = a U (\neg a \land b) \)

\[
\begin{align*}
\{a\} & \downarrow B_0 \\
\{a\} & \downarrow B_1 \\
\{a, b\} & \downarrow B_2 \\
\{b\} & \downarrow B_3 \\
\emptyset & \downarrow B_4 \\
\emptyset & \downarrow B_5 \\
\ldots & \vdash \varphi
\end{align*}
\]
Accepting runs for the elements of $Words(\varphi)$

LTL formula $\varphi \leadsto$ GNBA $G$ for $Words(\varphi)$

states of $G \triangleq$ elementary formula-sets $B \subseteq cl(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in Words(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b) \quad \psi = \neg a \land b$

\[
\begin{array}{ccccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset & \ldots \\
B_0 & B_1 & B_2 & B_3 & B_4 & B_5 \\
\end{array}
\]

where the $B_i$'s are states in $G$, i.e., elementary subsets of $\{a, \neg a, b, \neg b, \psi, \neg \psi, \varphi, \neg \varphi\}$
Accepting runs for the elements of $\text{Words}(\varphi)$

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

states of $G \models$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b) \quad \psi = \neg a \land b$

\[
\begin{array}{ccccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset & \ldots \\
\downarrow & & & & & & \\
& a & & & & & \\
& \neg b & & & & & \\
& \neg \psi & & & & & \\
& \varphi & & & & &
\end{array}
\]

$\models \varphi$
Accepting runs for the elements of $\text{Words}(\varphi)$

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

states of $\mathcal{G} \models$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$
s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $\mathcal{G}$

Example: $\varphi = a U(\neg a \land b)$ \hspace{0.5cm} $\psi = \neg a \land b$

\[
\begin{array}{cccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset \\
\downarrow & \downarrow & & & & \\
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset \\
\end{array}
\]

$\varphi \models \varphi$
LTL formula $\varphi \leadsto$ GNBA $G$ for $\text{Words}(\varphi)$

states of $G \supseteq$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a \cup (\neg a \land b) \quad \psi = \neg a \land b$

$$
\begin{align*}
\{a\} & \rightarrow \{a\} \rightarrow \{a, b\} \rightarrow \{b\} \rightarrow \emptyset \rightarrow \emptyset \rightarrow \ldots \models \varphi
\end{align*}
$$
Accepting runs for the elements of \( \text{Words}(\varphi) \)

LTL formula \( \varphi \leadsto \) GNBA \( G \) for \( \text{Words}(\varphi) \)

states of \( G \) \( \equiv \) elementary formula-sets \( B \subseteq \text{cl}(\varphi) \)

s.t. each word \( \sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi) \) can be extended to an accepting run \( B_0 B_1 B_2 \ldots \) in \( G \)

Example: \( \varphi = a U (\neg a \land b) \) \quad \psi = \neg a \land b

\[
\begin{array}{cccccccc}
\{a\} & \downarrow & \{a\} & \downarrow & \{a, b\} & \downarrow & \{b\} & \downarrow & \emptyset & \emptyset & \ldots & \models \varphi \\
 a & \neg b & \neg \psi & \varphi & a & \neg b & \neg \psi & \varphi & a & b & \neg b & \psi & \varphi
\end{array}
\]
Accepting runs for the elements of $\text{Words}(\varphi)$

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

states of $G \models$ elementary formula-sets $B \subseteq \text{cl}(\varphi)$

s.t. each word $\sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi)$ can be extended to an accepting run $B_0 B_1 B_2 \ldots$ in $G$

Example: $\varphi = a U (\neg a \land b)$ \hspace{1cm} $\psi = \neg a \land b$
Accepting runs for the elements of \( \text{Words}(\varphi) \)

LTL formula \( \varphi \leadsto \text{GNBA } G \) for \( \text{Words}(\varphi) \)

states of \( G \) \( \cong \) elementary formula-sets \( B \subseteq \text{cl}(\varphi) \)

s.t. each word \( \sigma = A_0 A_1 A_2 \ldots \in \text{Words}(\varphi) \) can be extended to an accepting run \( B_0 B_1 B_2 \ldots \) in \( G \)

Example: \( \varphi = a U(\neg a \land b) \quad \psi = \neg a \land b \)

\[
\begin{array}{cccccc}
\{a\} & \{a\} & \{a, b\} & \{b\} & \emptyset & \emptyset \\
\downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
a & a & a & \neg a & \neg a & \neg a \\
\neg b & \neg b & b & b & \neg b & \neg b \\
\neg \psi & \neg \psi & \psi & \psi & \neg \psi & \neg \psi \\
\varphi & \varphi & \varphi & \varphi & \varphi & \varphi \\
\end{array}
\]

\( \ldots \quad \models \varphi \)
GNBA for LTL-formula $\varphi$

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

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

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 $\delta(B, A) = \emptyset$

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

- $\Diamond \psi \in B$ iff $\psi \in B'$
- $\psi_1 \cup \psi_2 \in B$ iff $(\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')$

acceptance set $F = \{ F_{\psi_1 \cup \psi_2} : \psi_1 \cup \psi_2 \in \text{cl}(\varphi) \}$

where $F_{\psi_1 \cup \psi_2} = \{ B \in Q : \psi_1 \cup \psi_2 \notin B \lor \psi_2 \in B \}$
Elementary formula-sets

\[ B \subseteq \text{cl}(\varphi) \] is elementary iff:

(i) \( B \) is maximal consistent w.r.t. prop. logic, i.e., if \( \psi, \psi_1 \land \psi_2 \in \text{cl}(\varphi) \) then:

\[
\begin{align*}
\psi & \not\in B \quad \text{iff} \quad \neg \psi \in B \\
\psi_1 \land \psi_2 & \in B \quad \text{iff} \quad \psi_1 \in B \quad \text{and} \quad \psi_2 \in B \\
\text{true} & \in \text{cl}(\varphi) \quad \text{implies} \quad \text{true} \in B
\end{align*}
\]

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

\[
\begin{align*}
\text{if } \psi_1 \mathbf{U} \psi_2 & \in B \quad \text{and} \quad \psi_2 \not\in B \quad \text{then} \quad \psi_1 \in B \\
\text{if } \psi_2 & \in B \quad \text{then} \quad \psi_1 \mathbf{U} \psi_2 \in B
\end{align*}
\]
Let $\varphi$ be an LTL-formula and $G = (Q, 2^{AP}, \delta, Q_0, F)$ be the constructed GNBA.

Claim: $\text{Words}(\varphi) = L_\omega(G)$

“$\subseteq$” show: each infinite word $A_0 A_1 A_2 \ldots \in (2^{AP})^\omega$

with $A_0 A_1 A_2 \ldots \models \varphi$

has an accepting run in $G$

“$\supseteq$” show: for all infinite words $A_0 A_1 A_2 \ldots \in L_\omega(G)$:

$A_0 A_1 A_2 \ldots \models \varphi$
Soundness

Let $\varphi$ be an LTL-formula and $G = (Q, 2^{AP}, \delta, Q_0, F)$ be the constructed GNBA.

Claim: \( \text{Words}(\varphi) = L_\omega(G) \)

“⊆” show: each infinite word $A_0 A_1 A_2 \ldots \in (2^{AP})^\omega$

with $A_0 A_1 A_2 \ldots \models \varphi$

has an accepting run in $G$

“⊇” show: for all infinite words $A_0 A_1 A_2 \ldots \in L_\omega(G)$:

$A_0 A_1 A_2 \ldots \models \varphi$
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \\exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 \ A_1 \ A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 \ A_1 \ A_2 \ldots \in \mathcal{L}_\omega(G)$:

$$\Rightarrow \text{ there is an accepting run } B_0 \ B_1 \ B_2 \ldots \text{ for } \sigma$$
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:

$$\implies \text{there is an accepting run } B_0 B_1 B_2 \ldots \text{ for } \sigma$$

$$\implies B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \text{ is a path in } G$$
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:

$\Rightarrow$ there is an accepting run $B_0 B_1 B_2 \ldots$ for $\sigma$

$\Rightarrow B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. $\varphi \in B_0$
Proof of $\mathcal{L}_\omega(G) \subseteq Words(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in L_\omega(G)$:

$$\Rightarrow \text{ there is an accepting run } B_0 B_1 B_2 \ldots \text{ for } \sigma$$

$$\Rightarrow B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \text{ is a path in } G \text{ s.t. } \varphi \in B_0$$

as $B_0 \in Q_0$
Proof of $L_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$  \hspace{1cm} (*)

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in L_\omega(G)$:

$\Rightarrow$ there is an accepting run $B_0 B_1 B_2 \ldots$ for $\sigma$

$\Rightarrow$ $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. $\varphi \in B_0$

and (*) holds as $B_0 \in Q_0$
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:

$\implies$ there is an accepting run $B_0 B_1 B_2 \ldots$ for $\sigma$

$\implies$ $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. $\varphi \in B_0$

and (*) holds

$\implies$ $\sigma = A_0 A_1 A_2 \ldots \models \varphi$
Proof of $\mathcal{L}_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \, \exists \, j \geq 0. \, B_j \in F \quad (*)$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

The claim yields that for each $\sigma = A_0 A_1 A_2 \ldots \in \mathcal{L}_\omega(G)$:

$$\implies \quad \text{there is an accepting run } B_0 B_1 B_2 \ldots \text{ for } \sigma$$

$$\implies \quad B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \text{ is a path in } G \text{ s.t. } \varphi \in B_0$$

and $(*)$ holds

$$\implies \quad \sigma = A_0 A_1 A_2 \ldots \models \varphi$$
Proof of $L_\omega(G) \subseteq \text{Words}(\varphi)$

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \ \exists \ j \geq 0. B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \ \text{iff} \ \ A_0 A_1 A_2 \ldots \models \psi$$

*Proof by structural induction on $\psi$*

}\]
Proof of $L_\omega(G) \subseteq \text{Words}(\varphi)$

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

**Proof** by structural induction on $\psi$

**base of induction:**

$$\psi = \text{true}$$

$$\psi = a \in AP$$
**Proof of** $L_\omega(G) \subseteq \text{Words}(\varphi)$

**Claim:** If $B_0 \stackrel{A_0}{\rightarrow} B_1 \stackrel{A_1}{\rightarrow} B_2 \stackrel{A_2}{\rightarrow} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \ \exists j \geq 0. \ B_j \in F \quad (*)$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

**Proof** by structural induction on $\psi$

**Base of induction:**
- $\psi = \text{true}$
- $\psi = a \in \text{AP}$

**Induction step:**
- $\psi = \neg \psi'$
- $\psi = \psi_1 \land \psi_2$
- $\psi = \bigcirc \psi'$
- $\psi = \psi_1 \cup \psi_2$
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \; \exists j \geq 0. \; B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

Base of induction:
Base of induction

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[
\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F
\]

then for all formulas \( \psi \in cl(\varphi) \):

\[
\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi
\]

Base of induction:

Suppose \( \psi = \text{true} \in cl(\varphi) \).
Claim: If \( B_0 \rightarrow A_0 \rightarrow B_1 \rightarrow A_1 \rightarrow B_2 \rightarrow A_2 \rightarrow \ldots \) is a path in \( G \) s.t.
\[
\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F
\]
then for all formulas \( \psi \in \text{cl}(\varphi) \):
\[
\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi
\]

Base of induction:
Suppose \( \psi = \text{true} \in \text{cl}(\varphi) \). Then \( \text{true} \in B_0 \)

\textit{note:} \( \text{true} \) is contained in all elementary formula-sets
Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t. 
\[
\forall F \in \mathcal{F} \; \exists j \geq 0. \; B_j \in F
\]
then for all formulas \( \psi \in \text{cl}(\varphi) \):
\[
\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi
\]

Base of induction:

Suppose \( \psi = \text{true} \in \text{cl}(\varphi) \). Then \( \text{true} \in B_0 \) and
\[
A_0 A_1 A_2 \ldots \models \text{true}
\]

note: \( \text{true} \) is contained in all elementary formula-sets
\( \text{true} \) holds for all paths/traces
Base of induction

Claim: If $B_0 \rightarrow A_0 \rightarrow B_1 \rightarrow A_1 \rightarrow B_2 \rightarrow A_2 \rightarrow \ldots$ is a path in $\mathcal{G}$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

Base of induction:

Suppose $\psi = \text{true} \in \text{cl}(\varphi)$. Then $\text{true} \in B_0$ and

$$A_0 A_1 A_2 \ldots \models \text{true}$$

Let $\psi = a \in AP$. 
Base of induction

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \; \exists j \geq 0. \; B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi$$

Base of induction:

Suppose $\psi = \text{true} \in \text{cl}(\varphi)$. Then $\text{true} \in B_0$ and

$$A_0 A_1 A_2 \ldots \models \text{true}$$

Let $\psi = a \in \text{AP}$. Then:

$$a \in B_0$$
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. 
\[ \forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \]
then for all formulas $\psi \in \text{cl}(\varphi)$:
\[ \psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi \]

Base of induction:

Suppose $\psi = \text{true} \in \text{cl}(\varphi)$. Then $\text{true} \in B_0$ and 
\[ A_0 A_1 A_2 \ldots \models \text{true} \]

Let $\psi = a \in \text{AP}$. Then:
\[ a \in B_0 \iff a \in A_0 \]
Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[
\forall F \in \mathcal{F} \, \exists j \geq 0. \, B_j \in F
\]

then for all formulas \( \psi \in \text{cl}(\varphi) \):

\[
\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi
\]

Base of induction:

Suppose \( \psi = \text{true} \in \text{cl}(\varphi) \). Then \( \text{true} \in B_0 \) and

\[
A_0 A_1 A_2 \ldots \models \text{true}
\]

Let \( \psi = a \in \text{AP} \). Then:

\[
a \in B_0 \iff a \in A_0
\]
Claim: If \( \mathcal{G} \) is a path in \( \mathcal{G} \) s.t.
\[
\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F
\]
then for all formulas \( \psi \in \text{cl}(\varphi) \):
\[
\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi
\]

Base of induction:
Suppose \( \psi = \text{true} \in \text{cl}(\varphi) \). Then \( \text{true} \in B_0 \) and
\[
A_0 A_1 A_2 \ldots \models \text{true}
\]
Let \( \psi = a \in \text{AP} \). Then:
\[
a \in B_0 \iff a \in A_0
\]
Claim: If \[ \begin{array}{c} B_0 \rightarrow A_0 \\ A_0 \rightarrow B_1 \\ A_1 \rightarrow B_2 \\ A_2 \rightarrow \cdots \end{array} \] is a path in \( G \) s.t.

\[ \forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F \]

then for all formulas \( \psi \in \text{cl}(\varphi) \):

\[ \psi \in B_0 \iff A_0 A_1 A_2 \cdots \models \psi \]

Base of induction:

Suppose \( \psi = \text{true} \in \text{cl}(\varphi) \). Then \( \text{true} \in B_0 \) and

\[ A_0 A_1 A_2 \cdots \models \text{true} \]

Let \( \psi = a \in AP \). Then:

\[ a \in B_0 \iff a \in A_0 \iff A_0 A_1 A_2 \cdots \models a \]
Induction step: negation

LTLMC3.2-61
Induction step: negation

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \vdash \psi$$

Induction step: for $\psi = \neg \psi'$:
Induction step: negation

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \ \exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \ \text{iff} \ A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \neg \psi'$:

$$\psi \in B_0$$
Induction step: negation

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.
\[ \forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F \]
then for all formulas \( \psi \in \text{cl}(\varphi) \):
\[ \psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi \]

Induction step: for \( \psi = \neg \psi' \):
\[ \psi \in B_0 \quad \text{iff} \quad \psi' \not\in B_0 \quad \text{(maximal consistency)} \]
Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.
\[
\forall F \in \mathcal{F} \quad \exists j \geq 0. \; B_j \in F
\]
then for all formulas \( \psi \in cl(\varphi) \):
\[
\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi
\]

Induction step: for \( \psi = \neg \psi' \):
\[
\psi \in B_0
\]
iff \( \psi' \notin B_0 \) (maximal consistency)
iff \( A_0 A_1 A_2 \ldots \not\models \psi' \) (induction hypothesis)
Induction step: negation

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} ...$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \neg \psi'$:

$$\psi \in B_0$$

iff $\psi' \not\in B_0$ (maximal consistency)

iff $A_0 A_1 A_2 \ldots \not\models \psi'$ (induction hypothesis)

iff $A_0 A_1 A_2 \ldots \models \psi$ (semantics of $\neg$)
**Elementary formula-sets**

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

(i) $B$ is maximal consistent w.r.t. prop. logic, i.e., if $\psi, \psi_1 \land \psi_2 \in \text{cl}(\varphi)$ then:

\[
\begin{align*}
\psi \not\in B & \iff \neg \psi \in B \\
\psi_1 \land \psi_2 \in B & \iff \psi_1 \in B \text{ and } \psi_2 \in B \\
\text{true} \in \text{cl}(\varphi) & \text{ implies } \text{true} \in B
\end{align*}
\]

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

\[
\begin{align*}
\text{if } \psi_1 \mathbf{U} \psi_2 \in B \text{ and } \psi_2 \not\in B & \text{ then } \psi_1 \in B \\
\text{if } \psi_2 \in B & \text{ then } \psi_1 \mathbf{U} \psi_2 \in B
\end{align*}
\]
Elementary formula-sets

\( B \subseteq \text{cl}(\varphi) \) is elementary iff:

(i) \( B \) is maximal consistent w.r.t. prop. logic, i.e., if \( \psi, \psi_1 \land \psi_2 \in \text{cl}(\varphi) \) then:

\[
\psi \not\in B \iff \neg \psi \in B
\]

\[
\psi_1 \land \psi_2 \in B \iff \psi_1 \in B \text{ and } \psi_2 \in B
\]

\[
\text{true} \in \text{cl}(\varphi) \text{ implies } \text{true} \in B
\]

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

\[
\text{if } \psi_1 \mathbf{U} \psi_2 \in B \text{ and } \psi_2 \not\in B \text{ then } \psi_1 \in B
\]

\[
\text{if } \psi_2 \in B \text{ then } \psi_1 \mathbf{U} \psi_2 \in B
\]
Induction step: conjunction

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[ \forall F \in \mathcal{F} \quad \exists j \geq 0. \; B_j \in F \]

then for all formulas \( \psi \in \text{cl}(\varphi) \):

\[ \psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi \]

Induction step: for \( \psi = \psi_1 \land \psi_2 \)
Induction step: conjunction

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all formulas $\psi \in cl(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \psi_1 \land \psi_2$

$$\psi \in B_0$$
Induction step: conjunction

Claim: If $B_0 \xRightarrow{A_0} B_1 \xRightarrow{A_1} B_2 \xRightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \ \exists \ j \geq 0. \ B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \psi_1 \land \psi_2$

$$\psi \in B_0$$

iff $\psi_1, \psi_2 \in B_0$ (maximal consistency)
Induction step: conjunction

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \psi_1 \land \psi_2$

$$\psi \in B_0$$

iff $\psi_1, \psi_2 \in B_0$ (maximal consistency)

iff $A_0 A_1 A_2 \ldots \models \psi_1$ and $A_0 A_1 A_2 \ldots \models \psi_2$ (IH)
Induction step: conjunction

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$\forall F \in \mathcal{F} \supseteq j \geq 0. B_j \in F$

then for all formulas $\psi \in cl(\varphi)$:

$\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step: for $\psi = \psi_1 \land \psi_2$

$\psi \in B_0$

iff $\psi_1, \psi_2 \in B_0$ (maximal consistency)

iff $A_0 A_1 A_2 \ldots \models \psi_1$ and $A_0 A_1 A_2 \ldots \models \psi_2$ (IH)

iff $A_0 A_1 A_2 \ldots \models \psi$ (semantics of $\land$)
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[ \forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \]

then for all formulas $\psi \in cl(\varphi)$:

$\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step: for $\psi = \bigcirc\psi'$:
GNBA for LTL-formula $\varphi$

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

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

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 $\delta(B, A) = \emptyset$

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

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

$$\psi_1 U \psi_2 \in B \quad \text{iff} \quad (\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 U \psi_2 \in B')$$

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

where $F_{\psi_1 U \psi_2} = \{ B \in Q : \psi_1 U \psi_2 \notin B \lor \psi_2 \in B \}$
Induction step: next step

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} ...$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \bigcirc \psi'$:
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi$$

**Induction step:** for $\psi = \Box \psi'$:

$$\psi \in B_0$$
**Induction step: next step**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} ...$ is a path in $G$ s.t.

\[ \forall F \in F \exists j \geq 0. B_j \in F \]

then for all formulas $\psi \in \text{cl}(\varphi)$:

\[ \psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi \]

**Induction step:** for $\psi = \Box \psi'$:

\[ \psi \in B_0 \]

iff $\psi' \in B_1$  

(definition of $\delta$)
Induction step: next step

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in F \exists j \geq 0. B_j \in F$$

then $B_1 \in \delta(B_0, A_0)$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \text{ iff } A_0 A_1 A_2 \ldots \models \psi$$

Induction step: for $\psi = \bigcirc \psi'$:

$$\psi \in B_0$$

iff $\psi' \in B_1$ (definition of $\delta$)

iff $A_1 A_2 A_3 \ldots \models \psi'$ (induction hypothesis)
Induction step: next step

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \quad B_1 \in \delta(B_0, A_0)$$

then for all formulas $\psi \in \text{cl}(\varphi)$:

$$\psi \in B_0 \quad \text{iff} \quad A_0 A_1 A_2 \ldots \models \psi$$

**Induction step:** for $\psi = \Box \psi'$:

$$\psi \in B_0$$

iff $\psi' \in B_1$ (definition of $\delta$)

iff $A_1 A_2 A_3 \ldots \models \psi'$ (induction hypothesis)

iff $A_0 A_1 A_2 A_3 \ldots \models \psi$ (semantics of $\Box$)
Induction step: until LTLMC3.2-63
Recall: elementary formula-sets

\( B \subseteq \text{cl}(\varphi) \) is elementary iff:

(i) \( B \) is maximal consistent w.r.t. prop. logic, i.e., if \( \psi, \psi_1 \land \psi_2 \in \text{cl}(\varphi) \) then:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Implication</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \psi \not\in B )</td>
<td>( \neg \psi \in B )</td>
</tr>
<tr>
<td>( \psi_1 \land \psi_2 \in B )</td>
<td>( \psi_1 \in B ) and ( \psi_2 \in B )</td>
</tr>
<tr>
<td>( \text{true} \in \text{cl}(\varphi) )</td>
<td>( \text{true} \in B )</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Condition 1</th>
<th>Implication 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \psi_1 U \psi_2 \in B ) and ( \psi_2 \not\in B )</td>
<td>( \psi_1 \in B )</td>
</tr>
<tr>
<td>( \psi_2 \in B )</td>
<td>( \psi_1 U \psi_2 \in B )</td>
</tr>
</tbody>
</table>
Recall: GNBA for LTL-formula $\varphi$

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

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

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

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

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

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

$\bigcirc \psi \in B$ iff $\psi \in B'$

$\psi_1 \cup \psi_2 \in B$ iff $(\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')$

acceptance set $F = \left\{ F_{\psi_1 \cup \psi_2} : \psi_1 \cup \psi_2 \in \text{cl}(\varphi) \right\}$

where $F_{\psi_1 \cup \psi_2} = \left\{ B \in Q : \psi_1 \cup \psi_2 \notin B \lor \psi_2 \in B \right\}$
Recall: GNBA for LTL-formula $\varphi$

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

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 $\delta(B, A) = \emptyset$

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

$\bigcirc \varphi \in B \iff \varphi \in B'$

$\psi_1 \cup \psi_2 \in B \iff (\psi_2 \in B) \lor (\psi_1 \in B \land \psi_1 \cup \psi_2 \in B')$

acceptance set $F = \{ F_{\psi_1 \cup \psi_2} : \psi_1 \cup \psi_2 \in cl(\varphi) \}$

where $F_{\psi_1 \cup \psi_2} = \{ B \in Q : \psi_1 \cup \psi_2 \notin B \lor \psi_2 \in B \}$
Induction step: until LTLMC3.2-63
Induction step: until

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $\mathcal{G}$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$: 
**Induction step: until (part “⇐”)**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. 

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 U \psi_2$:**

“⇐”: Suppose $A_0 A_1 A_2 \ldots \models \psi$. 
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \, \exists j \geq 0. \, B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

"$\Leftarrow$": Suppose $A_0 A_1 A_2 \ldots \models \psi$. Let $j \geq 0$ s.t.

$A_j A_{j+1} A_{j+2} \ldots \models \psi_2$

$A_{j-1} A_j A_{j-1} \ldots \models \psi_1$

$A_{j-2} A_{j-1} A_j \ldots \models \psi_1$

\vdots

$A_0 A_1 A_2 A_3 \ldots \models \psi_1$
Induction step: until (part “$\Leftarrow$”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in F \quad \exists j \geq 0. \quad B_j \in F$$

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

“$\Leftarrow$”: Suppose $A_0 A_1 A_2 \ldots \models \psi$. Let $j \geq 0$ s.t.

$A_j A_{j+1} A_{j+2} \ldots \models \psi_2 \quad \xRightarrow{IH} \quad \psi_2 \in B_j$

$A_{j-1} A_j A_{j-1} \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_{j-1}$

$A_{j-2} A_{j-1} A_j \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_{j-2}$

\[ \vdots \]

$A_0 A_1 A_2 A_3 \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_0$
Induction step: until (part “$\Leftarrow$”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \quad \text{[B_j is elementary]}$$

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

$\Leftarrow$: Suppose $A_0 A_1 A_2 \ldots \models \psi$. Let $j \geq 0$ s.t.

$$A_j A_{j+1} A_{j+2} \ldots \models \psi_2 \quad \implies \quad \psi_2 \in B_j \quad \Rightarrow \quad \psi \in B_j$$

$$A_{j-1} A_j A_{j-1} \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_{j-1}$$

$$A_{j-2} A_{j-1} A_j \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_{j-2}$$

$$\vdots$$

$$A_0 A_1 A_2 A_3 \ldots \models \psi_1 \quad \Rightarrow \quad \psi_1 \in B_0$$
Induction step: until (part “⇐⇒”)  

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} ...$ is a path in $G$ s.t.  

\[ \forall F \in \mathcal{F} \exists j \geq 0. \, B_j \in F \quad \text{and} \quad B_j \in \delta(B_{j-1}, A_{j-1}) \]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

\[ \text{Induction step for } \psi = \psi_1 \cup \psi_2: \]

“⇐⇒”: Suppose $A_0 A_1 A_2 \ldots \models \psi$. Let $j \geq 0$ s.t.  

\[ A_j A_{j+1} A_{j+2} \ldots \models \psi_2 \implies \psi_2 \in B_j \implies \psi \in B_j \]

\[ A_{j-1} A_j A_{j-1} \ldots \models \psi_1 \implies \psi_1 \in B_{j-1} \land \psi \in B_{j-1} \]

\[ A_{j-2} A_{j-1} A_j \ldots \models \psi_1 \implies \psi_1 \in B_{j-2} \]

\[ \vdots \]

\[ A_0 A_1 A_2 A_3 \ldots \models \psi_1 \implies \psi_1 \in B_0 \]
Induction step: until (part “\(\Leftarrow\)”) 

Claim: If \(B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots\) is a path in \(G\) s.t.

\[
\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \quad \text{such that} \quad B_{j-1} \in \delta(B_{j-2}, A_{j-2})
\]

then for all \(\psi \in cl(\varphi)\): \(\psi \in B_0\) iff \(A_0 A_1 A_2 \ldots \models \psi\)

Induction step for \(\psi = \psi_1 \cup \psi_2\): 

“\(\Leftarrow\)” : Suppose \(A_0 A_1 A_2 \ldots \models \psi\). Let \(j \geq 0\) s.t.

\[
\begin{align*}
A_j A_{j+1} A_{j+2} \ldots \models \psi_2 & \quad \Rightarrow \quad \psi_2 \in B_j \quad \Rightarrow \quad \psi \in B_j \\
A_{j-1} A_j A_{j-1} \ldots \models \psi_1 & \quad \Rightarrow \quad \psi_1 \in B_{j-1} \quad \land \quad \psi \in B_{j-1} \\
A_{j-2} A_{j-1} A_j \ldots \models \psi_1 & \quad \Rightarrow \quad \psi_1 \in B_{j-2} \quad \land \quad \psi \in B_{j-2} \\
\vdots & \quad \vdots \\
A_0 A_1 A_2 A_3 \ldots \models \psi_1 & \quad \Rightarrow \quad \psi_1 \in B_0
\end{align*}
\]
**Induction step: until (part “⇐”)**

**Claim:** If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[
\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F
\]

then for all \( \psi \in \text{cl}(\varphi) \) : \( \psi \in B_0 \iff A_0 A_1 A_2 \ldots \models \psi \)

**Induction step for \( \psi = \psi_1 \cup \psi_2 \):**

“⇐” : Suppose \( A_0 A_1 A_2 \ldots \models \psi \). Let \( j \geq 0 \) s.t.

\[
\begin{align*}
A_j A_{j+1} A_{j+2} \ldots & \models \psi_2 \quad \implies \quad \psi_2 \in B_j \\
A_{j-1} A_j A_{j-1} \ldots & \models \psi_1 \quad \implies \quad \psi_1 \in B_{j-1} \\
A_{j-2} A_{j-1} A_j \ldots & \models \psi_1 \quad \implies \quad \psi_1 \in B_{j-2} \\
& \vdots \\
A_0 A_1 A_2 A_3 \ldots & \models \psi_1 \quad \implies \quad \psi_1 \in B_0
\end{align*}
\]
Induction step: until (part “⇒”)

LTLMC3.2-64
Induction step: until (part "\(\implies\)")

Claim: If \(B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots\) is a path in \(G\) s.t.

\[
\forall F \in \mathcal{F} \quad \exists \ j \geq 0. \ B_j \in F
\]

then for all \(\psi \in \text{cl}(\varphi)\): \(\psi \in B_0\) iff \(A_0 A_1 A_2 \ldots \models \psi\)

Induction step for \(\psi = \psi_1 \cup \psi_2\):
Induction step: until (part “⇒”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t. 
$$\forall F \in \mathcal{F} \, \exists j \geq 0. \, B_j \in F$$
then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:
“⇒” Suppose $\psi \in B_0$. 

Induction step: until (part “⇒”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in F \exists j \geq 0. B_j \in F$$

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 U \psi_2$:**

“$\models$” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, \ldots
Induction step: until (part "\(\Rightarrow\)")

Claim: If \(B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots\) is a path in \(G\) s.t.
\[
\forall F \in \mathcal{F} \quad \exists \ j \geq 0. \ B_j \in F
\]
then for all \(\psi \in \text{cl}(\varphi)\):
\(\psi \in B_0\) iff \(A_0 A_1 A_2 \ldots\) \(\models \psi\)

Induction step for \(\psi = \psi_1 \cup \psi_2\):

"\(\Rightarrow\)" Suppose \(\psi \in B_0\). There exists \(j \geq 0\) with \(\psi_2 \in B_j\), since otherwise \(\forall j \geq 0. \ \psi_2 \notin B_j\)
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.
\[ \forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F \]
then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

“$\implies$” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, since otherwise $\forall j \geq 0. \quad \psi_2 \not\in B_j$ and therefore:
\[ \psi \in B_0 \land \psi_2 \not\in B_0 \]
Induction step: until (part “⇒⇒⇒”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

“⇒⇒⇒” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, since otherwise $\forall j \geq 0. \psi_2 \notin B_j$ and therefore:

$$\psi \in B_0 \land \psi_2 \notin B_0$$

$\Rightarrow \psi \in B_1$
Induction step: until (part “⇒⇒⇒”)
Induction step: until (part “⇒”)

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 U \psi_2$:

“⇒” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, since otherwise $\forall j \geq 0. \; \psi_2 \notin B_j$ and therefore:

$$\psi \in B_0 \land \psi_2 \notin B_0$$

$$\Rightarrow \quad \psi \in B_1 \land \psi_2 \notin B_1$$

$$\Rightarrow \quad \psi \in B_2$$
Induction step: until (part “⇒⇒⇒”)  

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[
\forall F \in \mathcal{F} \\exists \ j \geq 0. \ B_j \in F
\]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

“⇒⇒⇒” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, since otherwise $\forall j \geq 0. \ \psi_2 \not\in B_j$ and therefore:

\[
\psi \in B_0 \land \psi_2 \not\in B_0 \\
\Rightarrow \psi \in B_1 \land \psi_2 \not\in B_1 \\
\Rightarrow \psi \in B_2 \land \psi_2 \not\in B_2 \\
\vdots
\]
**Induction step: until (part “⇒”)**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[
\forall F \in F \exists j \geq 0. \ B_j \in F
\]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

---

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

“⇒” Suppose $\psi \in B_0$. There exists $j \geq 0$ with $\psi_2 \in B_j$, since otherwise $\forall j \geq 0. \ \psi_2 \notin B_j$ and therefore:

\[
\begin{align*}
\psi & \in B_0 \land \psi_2 \notin B_0 \\
\Rightarrow \psi & \in B_1 \land \psi_2 \notin B_1 \\
\Rightarrow \psi & \in B_2 \land \psi_2 \notin B_2 \\
& \vdots
\end{align*}
\]

$\Rightarrow \forall j \geq 0. \ B_j \notin F_\psi$ where

\[
F_\psi = \{ B : \psi \notin B \text{ or } \psi_2 \in B \}
\]
Induction step: until (part “⇒”)

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[
\forall F \in \mathcal{F} \; \exists j \geq 0. \; B_j \in F
\]

then for all \( \psi \in \text{cl}(\varphi) \): \( \psi \in B_0 \) iff \( A_0 A_1 A_2 \ldots \models \psi \)

Induction step for \( \psi = \psi_1 \cup \psi_2 \):

“⇒⇒⇒” Suppose \( \psi \in B_0 \). There exists \( j \geq 0 \) with \( \psi_2 \in B_j \), since otherwise \( \forall j \geq 0. \; \psi_2 \notin B_j \) and therefore:

\[
\psi \in B_0 \land \psi_2 \notin B_0 \quad \Rightarrow \quad \psi \in B_1 \land \psi_2 \notin B_1 \\
\Rightarrow \quad \psi \in B_2 \land \psi_2 \notin B_2 \\
\vdots
\]

\( \Rightarrow \forall j \geq 0. \; B_j \notin F_\psi \) where

\[
F_\psi = \{ B : \psi \notin B \text{ or } \psi_2 \in B \}
\]

Contradiction!
Induction step: until (part “⇒”)
**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} ...$ is a path in $G$ s.t.
\[\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F\]
then for all $\psi \in \cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**
Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$
**Induction step: until (part “⇒”)**

Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[ \forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 U \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

\[ \xRightarrow{IH} \quad A_j A_{j+1} \ldots \models \psi_2 \]
Induction step: until (part “⇒”)

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.

\[
\forall F \in \mathcal{F} \exists j \geq 0. \ B_j \in F
\]

then for all \( \psi \in \text{cl}(\varphi) \): \( \psi \in B_0 \) iff \( A_0 A_1 A_2 \ldots \models \psi \)

Induction step for \( \psi \ = \ \psi_1 \cup \psi_2 \):

Let \( \psi \in B_0 \) and \( j \geq 0 \) minimal s.t. \( \psi_2 \in B_j \)

\[
\text{IH} \quad A_j A_{j+1} \ldots \models \psi_2
\]

\( \neg \psi_2 \in B_{j-1} \)

\( \neg \psi_2 \in B_{j-2} \)

\[\vdots\]

\( \neg \psi_2 \in B_1 \)

\( \neg \psi_2 \in B_0 \)
**Induction step: until (part “⇒”)**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in F \exists j \geq 0. B_j \in F$$

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 U \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

$$\models \text{IH} \quad A_j A_{j+1} \ldots \models \psi_2$$

$$\neg \psi_2 \in B_{j-1}$$

$$\neg \psi_2 \in B_{j-2}$$

$$\vdots$$

$$\neg \psi_2 \in B_1$$

$$\neg \psi_2, \psi \in B_0 \quad \longleftrightarrow \text{by assumption}$$
Induction step: until (part “⇒”)

Claim: If \( B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots \) is a path in \( G \) s.t.
\[
\forall F \in F \; \exists \; j \geq 0. \; B_j \in F
\]
then for all \( \psi \in \text{cl}(\varphi) \): \( \psi \in B_0 \) iff \( A_0 A_1 A_2 \ldots \models \psi \)

Induction step for \( \psi = \psi_1 \cup \psi_2 \):

Let \( \psi \in B_0 \) and \( j \geq 0 \) minimal s.t. \( \psi_2 \in B_j \)
\[
\text{IH} \quad A_j A_{j+1} \ldots \models \psi_2
\]

\( \neg \psi_2 \in B_{j-1} \)
\( \neg \psi_2 \in B_{j-2} \)
\( \vdots \)
\( \neg \psi_2 \in B_1 \)
\( \neg \psi_2, \psi_1, \psi \in B_0 \) ← local consistency w.r.t. \( \cup \)
**Induction step: until (part “⇒”)**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[ \forall F \in \mathcal{F} \exists j \geq 0. B_j \in F \]

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

\[ \text{IH} \quad A_j A_{j+1} \ldots \models \psi_2 \]

\[ \neg \psi_2 \in B_{j-1} \]
\[ \neg \psi_2 \in B_{j-2} \]
\[ \vdots \]
\[ \neg \psi_2, \psi_1, \psi \in B_1 \]
\[ \neg \psi_2, \psi_1, \psi \in B_0 \quad \leftarrow \text{local consistency w.r.t. } \cup \]
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \exists j \geq 0. B_j \in F$$

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

$$\text{IH} \quad A_j A_{j+1} \ldots \models \psi_2$$

$\neg \psi_2, \psi_1, \psi \in B_{j-1}$

$\neg \psi_2, \psi_1, \psi \in B_{j-2}$

$\vdots$

$\neg \psi_2, \psi_1, \psi \in B_1$

$\neg \psi_2, \psi_1, \psi \in B_0$  \hspace{1cm} ← local consistency w.r.t. $\cup$
Induction step: until (part “$\Rightarrow$”)

Claim: If $B_0 \stackrel{A_0}{\rightarrow} B_1 \stackrel{A_1}{\rightarrow} B_2 \stackrel{A_2}{\rightarrow} \ldots$ is a path in $G$ s.t.

\[
\forall F \in \mathcal{F} \; \exists \; j \geq 0. \; B_j \in F
\]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

$\models_{\text{IH}} A_j A_{j+1} \ldots \models \psi_2$

$\neg \psi_2, \psi_1, \psi \in B_{j-1} \quad \models \quad A_{j-1} A_j \ldots \models \psi_1$

$\vdots$

$\neg \psi_2, \psi_1, \psi \in B_1$

$\neg \psi_2, \psi_1, \psi \in B_0 \quad \leftarrow \text{local consistency w.r.t. } \cup$
**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

\[
\forall F \in \mathcal{F} \quad \exists j \geq 0. \quad B_j \in F
\]

then for all $\psi \in cl(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

\[
\text{IH} \quad A_j A_{j+1} \ldots \models \psi_2
\]

\[
\neg \psi_2, \psi_1, \psi \in B_{j-1} \quad \Rightarrow \quad A_{j-1} A_j \ldots \models \psi_1
\]

\[
\neg \psi_2, \psi_1, \psi \in B_{j-2} \quad \Rightarrow \quad A_{j-2} A_{j-1} \ldots \models \psi_1
\]

\[
\vdots
\]

\[
\neg \psi_2, \psi_1, \psi \in B_1
\]

\[
\neg \psi_2, \psi_1, \psi \in B_0 \quad \leftarrow \text{local consistency w.r.t. } U
\]
**Induction step: until (part “implies”)**

**Claim:** If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.

$$\forall F \in \mathcal{F} \quad \exists j \geq 0. \; B_j \in F$$

then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

**Induction step for $\psi = \psi_1 \cup \psi_2$:**

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

\[
\begin{align*}
\text{IH} & \quad \Rightarrow \quad A_j A_{j+1} \ldots \models \psi_2 \\
\neg \psi_2, \psi_1, \psi & \in B_{j-1} \quad \Rightarrow \quad A_{j-1} A_j \ldots \models \psi_1 \\
\neg \psi_2, \psi_1, \psi & \in B_{j-2} \quad \Rightarrow \quad A_{j-2} A_{j-1} \ldots \models \psi_1 \\
\vdots & \quad \vdots \quad \vdots \quad \vdots \\
\neg \psi_2, \psi_1, \psi & \in B_1 \quad \Rightarrow \quad A_1 A_2 A_3 \ldots \models \psi_1 \\
\neg \psi_2, \psi_1, \psi & \in B_0
\end{align*}
\]
Induction step: until (part “⇒”)
Claim: If $B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots$ is a path in $G$ s.t.
\[
\forall F \in \mathcal{F} \ \exists j \geq 0. \ B_j \in F
\]
then for all $\psi \in \text{cl}(\varphi)$: $\psi \in B_0$ iff $A_0 A_1 A_2 \ldots \models \psi$

Induction step for $\psi = \psi_1 \cup \psi_2$:

Let $\psi \in B_0$ and $j \geq 0$ minimal s.t. $\psi_2 \in B_j$

\[
\begin{array}{c}
\text{IH} \\
\text{IH}
\end{array}
\quad \Rightarrow \quad
\begin{array}{c}
A_j A_{j+1} \ldots \models \psi_2 \\
A_{j-1} A_j \ldots \models \psi_1
\end{array}
\]

\[
\begin{array}{c}
\neg \psi_2, \psi_1, \psi \in B_{j-1} \\
\vdots \\
\neg \psi_2, \psi_1, \psi \in B_0
\end{array}
\quad \Rightarrow \quad
\begin{array}{c}
A_{j-1} A_j \ldots \models \psi_1 \\
\vdots \\
A_0 A_1 A_2 \ldots \models \psi_1
\end{array}
\]

\[
\downarrow
\]
Induction step: until (part “\(\implies\)”)}

Claim: If \(B_0 \xrightarrow{A_0} B_1 \xrightarrow{A_1} B_2 \xrightarrow{A_2} \ldots\) is a path in \(G\) s.t.
\[\forall F \in \mathcal{F} \quad \exists j \geq 0. B_j \in F\]
then for all \(\psi \in \text{cl}(\varphi)\):  \(\psi \in B_0\) iff \(A_0 A_1 A_2 \ldots \models \psi\)

Induction step for \(\psi = \psi_1 \cup \psi_2\):
Let \(\psi \in B_0\) and \(j \geq 0\) minimal s.t. \(\psi_2 \in B_j\)

\[\models_{\text{IH}} \quad A_j A_{j+1} \ldots \models \psi_2\]
\[
\neg \psi_2, \psi_1, \psi \in B_{j-1} \quad \implies \quad A_{j-1} A_j \ldots \models \psi_1
\]
\[
\vdots
\]
\[
\neg \psi_2, \psi_1, \psi \in B_0 \quad \implies \quad A_0 A_1 A_2 \ldots \models \psi_1
\]
\[
\Downarrow
\]
\[
A_0 A_1 A_2 \ldots \models \psi = \psi_1 \cup \psi_2
\]
Complexity: LTL $\rightsquigarrow$ NBA
For each \textbf{LTL} formula $\varphi$, there is an \textbf{NBA} $\mathcal{A}$ s.t. 
\[
\mathcal{L}_\omega(\mathcal{A}) = \text{Words}(\varphi)
\]
For each LTL formula $\varphi$, there is an NBA $\mathcal{A}$ s.t.

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

$$L_\omega(\mathcal{A}) = \text{Words}(\varphi)$$

**Diagram:**
- **LTL formula $\varphi$**
- **GNBA $\mathcal{G}$**
- **NBA $\mathcal{A}$**

**Size:** $\text{size}(\mathcal{G}) \cdot |\mathcal{F}|$
For each LTL formula $\varphi$, there is an NBA $\mathcal{A}$ s.t.

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

LTL formula $\varphi$

GNBA $\mathcal{G}$

NBA $\mathcal{A}$

size: $\text{size}(\mathcal{G}) \cdot |\mathcal{F}|$

$|\mathcal{F}| = \text{number of acceptance sets in } \mathcal{G}$
For each LTL formula $\varphi$, there is an NBA $\mathcal{A}$ s.t.

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

LTL formula $\varphi$

GNBA $\mathcal{G}$

NBA $\mathcal{A}$

size: $\text{size}(\mathcal{G}) \cdot |\mathcal{F}|$

$|\mathcal{F}| = \text{number of acceptance sets in } \mathcal{G}$

$\leq |\varphi|$
For each LTL formula $\varphi$, there is an NBA $\mathcal{A}$ s.t.

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

\[
\mathcal{L}_\omega(\mathcal{A}) = \text{Words}(\varphi) \quad \text{and} \\
\text{size}(\mathcal{A}) \leq 2^{\lvert\text{cl}(\varphi)\rvert} \cdot \lvert\varphi\rvert.
\]
For each LTL formula $\varphi$, there is an NBA $A$ s.t.

$$
\mathcal{L}_\omega(A) = \text{Words}(\varphi) \quad \text{and} \\
\text{size}(A) \leq 2^{\text{cl}(\varphi)} \cdot |\varphi| = 2^O(|\varphi|)
$$

**LTL formula $\varphi$**

**GNBA $\mathcal{G}$**

size: $2^{\text{cl}(\varphi)}$

**NBA $A$**

size: $\text{size}(\mathcal{G}) \cdot |\mathcal{F}|$

$|\mathcal{F}| = \text{number of acceptance sets in } \mathcal{G}$

$\leq |\varphi|$
Size of NBA for LTL formulas

LTLMC3.2-68
For the proposed transformation \( \text{LTL} \rightsquigarrow \text{NBA} \):

The constructed NBA for LTL formulas are often unnecessarily complicated.
For the proposed transformation \textbf{LTL} $\mapsto \text{NBA}$:

The constructed NBA for LTL formulas are often unnecessarily complicated.

NBA for $\bigcirc a$

constructed GNBA has 4 states and 8 edges
For the proposed transformation $\text{LTL} \leadsto \text{NBA}$:

The constructed NBA for LTL formulas are often unnecessarily complicated.

NBA for $a U b$

constructed (G)NBA has 5 states and 20 edges.
For the proposed transformation $\text{LTL} \rightsquigarrow \text{NBA}$:

- The constructed NBA for LTL formulas are often unnecessarily complicated.

... but there exists LTL formulas $\varphi_n$ such that

- $|\varphi_n| = \mathcal{O}(\text{poly}(n))$
- each NBA for $\varphi_n$ has at least $2^n$ states
LT-propsities that have no “small” NBA
consider the following family of LT-properties \((E_n)_{n \geq 1}\):

\[
E_n = \begin{cases} 
\text{set of all infinite words over } 2^{AP} \text{ of the form } \\
A_1 A_2 A_3 \ldots A_n A_1 A_2 A_3 \ldots A_n B_1 B_2 B_3 B_4 \ldots 
\end{cases}
\]
consider the following family of LT-properties \((E_n)_{n \geq 1}\):

\[ E_n = \begin{cases} 
\text{set of all infinite words over } 2^{AP} \text{ of the form } \\
A_1 A_2 A_3 \ldots A_n A_1 A_2 A_3 \ldots A_n B_1 B_2 B_3 B_4 \ldots \\
= xx \\
\text{for some } x \in (2^{AP})^* \\
\text{of length } n \\
\in (2^{AP})^\omega \\
\text{arbitrary} 
\end{cases} \]
LT-properties that have no “small” NBA

consider the following family of LT-properties \((E_n)_{n \geq 1}\):

\[ E_n = \begin{cases} 
\text{set of all infinite words over } 2^{AP} \text{ of the form} \\
A_1 A_2 A_3 \ldots A_n A_1 A_2 A_3 \ldots A_n B_1 B_2 B_3 B_4 \ldots \\
= xx \\
\text{for some } x \in (2^{AP})^* \\
\text{of length } n 
\end{cases} \in (2^{AP})^\omega \\
\text{arbitrary}
\]

LTL formula \(\varphi_n\) with \(\text{Words}(\varphi_n) = E_n\)
LT-properties that have no “small” NBA

consider the following family of LT-properties \((E_n)_{n \geq 1}\):

\[
E_n = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form } \begin{array}{c}
A_1 A_2 A_3 \ldots A_n A_1 A_2 A_3 \ldots A_n \\
B_1 B_2 B_3 B_4 \ldots
\end{array}
\right. \\
= xx \\
\text{for some } x \in (2^{AP})^* \\
\text{of length } n \\
\in (2^{AP})^\omega \\
\text{arbitrary}
\]

LTL formula \(\varphi_n\) with \(\text{Words}(\varphi_n) = E_n\)

\[
\varphi_n = \bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \leftrightarrow \bigcirc^{i+n} a)
\]
consider the following family of LT-properties \((E_n)_{n \geq 1}\):

\[
E_n = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form} \right. \\
\begin{align*}
A_1 A_2 A_3 \ldots A_n A_1 A_2 A_3 \ldots A_n B_1 B_2 B_3 B_4 \ldots \\
&= xx \\
&\text{for some } x \in (2^{AP})^* \\
&\text{of length } n
\end{align*}
\right\} 
\in (2^{AP})^\omega
\]

for arbitrary

LTL formula \(\varphi_n\) with \(\text{Words}(\varphi_n) = E_n\)

\[
\varphi_n = \bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \leftrightarrow \bigcirc^{i+n} a)
\]

length \(O(poly(n))\)
LT-property $E_n$ for $n=1$

$$E_1 = \left\{ \text{set of all infinite words over } 2^\text{AP} \text{ of the form } \right.$$  
\[A A B_1 B_2 B_3 B_4 \ldots \text{ where } A, B_j \subseteq \text{AP} \text{ for } j \geq 0\]
LT-property $E_n$ for $n=1$

$E_1 = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form} \right.$

$$A A B_1 B_2 B_3 B_4 \ldots \text{ where } A, B_j \subseteq AP \text{ for } j \geq 0$$

NBA for $E_1$ if $AP = \{a\}$:

![NBA Diagram]

- From $q_0$ to $q_1$: $a$
- From $q_1$ to $q_2$: $a$
- From $q_2$ to $q_2$: $\neg a$
- From $q_2$ to $q_0$: $\neg a$
- From $q_2$: Loop $\text{true}$
LT-property $E_n$ for $n=1$

\[ E_1 = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form} \right. \]

\[ A A B_1 B_2 B_3 B_4 \ldots \text{ where } A, B_j \subseteq AP \text{ for } j \geq 0 \]

NBA for $E_1$ if $AP = \{a\}$:

```
LTL-formula:
a ↔ □ a
```
**LT-property** $E_n$ for $n=1$

$E_1 = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form} \right. \\
A A B_1 B_2 B_3 B_4 \ldots \text{ where } A, B_j \subseteq AP \text{ for } j \geq 0 \left. \right\}$

NBA for $E_1$ if $AP = \{a, b\}$:
LT-property $E_n$ for $n=1$

\[ E_1 = \left\{ \text{set of all infinite words over } 2^{AP} \text{ of the form} \right. \]
\[ A A B_1 B_2 B_3 B_4 \ldots \text{ where } A, B_j \subseteq AP \text{ for } j \geq 0 \]

NBA for $E_1$ if $AP = \{a, b\}$:

LTL-formula:
\[ (a \leftrightarrow \bigcirc a) \land (b \leftrightarrow \bigcirc b) \]
LT property $E_n$ for $n=2$ and $AP = \{a\}$

$$E_2 = \{ A_1 A_2 A_1 A_2 \sigma : A_1, A_2 \subseteq AP, \sigma \in (2^{AP})^\omega \}$$
LT property $E_n$ for $n=2$ and $AP = \{a\}$

$$E_2 = \{A_1 A_2 A_1 A_2 \sigma : A_1, A_2 \subseteq AP, \sigma \in (2^{AP})^\omega \}$$

LTL-formula: $(a \leftrightarrow ОООa) \land (Оa \leftrightarrow ООООa)$
LT property $E_n$ for $n=2$ and $AP = \{a\}$

**general case:** each NBA for $E_n$ has $\geq 2^n$ states
LT property $E_n$ for $n=2$ and $AP = \{a\}$

**general case:** each NBA for $E_n$ has $\geq 2^n$ states

$$E_n = \text{Words}(\varphi_n) \text{ where } \varphi_n = \bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \leftrightarrow \bigcirc^{n+i} a)$$
LT property $E_n$ for $n=2$ and $AP = \{a\}$

**general case:** each NBA for $E_n$ has $\geq 2^n$ states

$E_n = \text{Words}(\varphi_n)$ where $\varphi_n = \bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \leftrightarrow \bigcirc^{n+i} a)$