



Logic

Luca Campa

Philipp Dablander

Aaron Groß

Aart Middeldorp

Alexander Montag

Johannes Niederhauser

Vera Schmitt

Outline

- 1. Summary of Previous Lecture**
- 2. Adequacy**
- 3. Fairness**
- 4. Intermezzo**
- 5. LTL Model Checking Algorithm**
- 6. Further Reading**
- 7. Exam**

Definitions

model $\mathcal{M} = (S, \rightarrow, L)$ and $X \subseteq S$

- ▶ $[[\varphi]] = \{s \in S \mid \mathcal{M}, s \models \varphi\}$
- ▶ $\text{pre}_{\forall}(X) = \{s \in S \mid t \in X \text{ for all } t \text{ with } s \rightarrow t\}$
- ▶ $\text{pre}_{\exists}(X) = \{s \in S \mid s \rightarrow t \text{ for some } t \in X\}$

$$\llbracket \top \rrbracket = S$$

$$\llbracket \perp \rrbracket = \emptyset$$

$$\llbracket \neg \varphi \rrbracket = S - \llbracket \varphi \rrbracket$$

$$\llbracket \varphi \wedge \psi \rrbracket = \llbracket \varphi \rrbracket \cap \llbracket \psi \rrbracket$$

$$\llbracket \varphi \vee \psi \rrbracket = \llbracket \varphi \rrbracket \cup \llbracket \psi \rrbracket$$

$$\llbracket \varphi \rightarrow \psi \rrbracket = (S - \llbracket \varphi \rrbracket) \cup \llbracket \psi \rrbracket$$

$$\text{pre}_\forall(X) = S - \text{pre}_\exists(S - X)$$

$$\llbracket p \rrbracket = \{s \in S \mid p \in L(s)\}$$

$$\llbracket \text{AX } \varphi \rrbracket = \text{pre}_\forall(\llbracket \varphi \rrbracket)$$

$$\llbracket \text{EX } \varphi \rrbracket = \text{pre}_\exists(\llbracket \varphi \rrbracket)$$

$$\llbracket \text{AF } \varphi \rrbracket = \llbracket \varphi \rrbracket \cup \text{pre}_\forall(\llbracket \text{AF } \varphi \rrbracket)$$

$$\llbracket \text{EF } \varphi \rrbracket = \llbracket \varphi \rrbracket \cup \text{pre}_\exists(\llbracket \text{EF } \varphi \rrbracket)$$

$$\llbracket \text{AG } \varphi \rrbracket = \llbracket \varphi \rrbracket \cap \text{pre}_\forall(\llbracket \text{AG } \varphi \rrbracket)$$

$$\llbracket \text{EG } \varphi \rrbracket = \llbracket \varphi \rrbracket \cap \text{pre}_\exists(\llbracket \text{EG } \varphi \rrbracket)$$

$$\llbracket \text{A}[\varphi \text{ U } \psi] \rrbracket = \llbracket \psi \rrbracket \cup (\llbracket \varphi \rrbracket \cap \text{pre}_\forall(\llbracket \text{A}[\varphi \text{ U } \psi] \rrbracket))$$

$$\llbracket \text{E}[\varphi \text{ U } \psi] \rrbracket = \llbracket \psi \rrbracket \cup (\llbracket \varphi \rrbracket \cap \text{pre}_\exists(\llbracket \text{E}[\varphi \text{ U } \psi] \rrbracket))$$

Lemma

- ▶ $\llbracket \text{AF } \varphi \rrbracket$ is least fixed point of monotone function $F_{\text{AF}}(X) = \llbracket \varphi \rrbracket \cup \text{pre}_{\forall}(X)$
- ▶ $\llbracket \text{EG } \varphi \rrbracket$ is greatest fixed point of monotone function $F_{\text{EG}}(X) = \llbracket \varphi \rrbracket \cap \text{pre}_{\exists}(X)$
- ▶ $\llbracket \text{E}[\psi \text{ U } \varphi] \rrbracket$ is least fixed point of monotone function $F_{\text{EU}}(X) = \llbracket \psi \rrbracket \cup (\llbracket \varphi \rrbracket \cap \text{pre}_{\exists}(X))$

Theorem (Knaster–Tarski)

every **monotone** function $F: \mathcal{P}(S) \rightarrow \mathcal{P}(S)$ with $|S| = n$ admits

- ▶ **least fixed point** $\mu F = F^n(\emptyset)$
- ▶ **greatest fixed point** $\nu F = F^n(S)$

symbolic model checking = (CTL) model checking with **BDDs**

Definitions

- ▶ **LTL (linear-time temporal logic)** formulas are built from
 - ▶ atoms p, q, r, p_1, p_2, \dots
 - ▶ logical connectives $\perp, \top, \neg, \wedge, \vee, \rightarrow$
 - ▶ **temporal connectives** X, F, G, U, W, R

according to following BNF grammar:

$$\varphi ::= \perp \mid \top \mid p \mid (\neg\varphi) \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid (\varphi \rightarrow \varphi) \mid \\ (X\varphi) \mid (F\varphi) \mid (G\varphi) \mid (\varphi U \varphi) \mid (\varphi W \varphi) \mid (\varphi R \varphi)$$

- ▶ **path** in model $\mathcal{M} = (S, \rightarrow, L)$ is infinite sequence $s_1 \rightarrow s_2 \rightarrow \dots$
- ▶ **satisfaction** $\pi \models \varphi$ of LTL formula φ with respect to path $\pi = s_1 \rightarrow s_2 \rightarrow \dots$ in model \mathcal{M} is defined by induction on φ
- ▶ **satisfaction** $\mathcal{M}, s \models \varphi$ of LTL formula φ with respect to state $s \in S$ in model \mathcal{M} is defined as "for all paths $\pi = s \rightarrow \dots$ $\pi \models \varphi$ "

Definition

LTL formulas φ and ψ are **semantically equivalent** ($\varphi \equiv \psi$) if

$$\pi \models \varphi \iff \pi \models \psi$$

for all models $\mathcal{M} = (S, \rightarrow, L)$ and paths π in \mathcal{M}

Definition

LTL formulas φ and ψ are **semantically equivalent** ($\varphi \equiv \psi$) if

$$\pi \models \varphi \iff \pi \models \psi$$

for all models $\mathcal{M} = (S, \rightarrow, L)$ and paths π in \mathcal{M}

Remark

$$\pi \not\models \varphi \iff \pi \models \neg\varphi$$

Definition

LTL formulas φ and ψ are **semantically equivalent** ($\varphi \equiv \psi$) if

$$\pi \models \varphi \iff \pi \models \psi$$

for all models $\mathcal{M} = (S, \rightarrow, L)$ and paths π in \mathcal{M}

Remark

$$\pi \not\models \varphi \iff \pi \models \neg\varphi \quad \mathcal{M}, s \models \varphi \implies \mathcal{M}, s \not\models \neg\varphi$$

Definition

LTL formulas φ and ψ are **semantically equivalent** ($\varphi \equiv \psi$) if

$$\pi \models \varphi \iff \pi \models \psi$$

for all models $\mathcal{M} = (S, \rightarrow, L)$ and paths π in \mathcal{M}

Remark

$$\pi \not\models \varphi \iff \pi \models \neg\varphi \quad \mathcal{M}, s \models \varphi \implies \mathcal{M}, s \not\models \neg\varphi \quad \mathcal{M}, s \not\models \varphi \not\implies \mathcal{M}, s \models \neg\varphi$$

Theorem

$$\neg X \varphi \equiv X \neg \varphi$$

$$\neg F \varphi \equiv G \neg \varphi$$

$$\neg G \varphi \equiv F \neg \varphi$$

$$\neg(\varphi U \psi) \equiv \neg \varphi R \neg \psi$$

$$\neg(\varphi R \psi) \equiv \neg \varphi U \neg \psi$$

$$\varphi U \psi \equiv \varphi W \psi \wedge F \psi$$

$$\varphi W \psi \equiv \varphi U \psi \vee G \varphi$$

$$\varphi U \psi \equiv \neg(\neg \psi U (\neg \varphi \wedge \neg \psi)) \wedge F \psi$$

$$F(\varphi \vee \psi) \equiv F \varphi \vee F \psi$$

$$G(\varphi \wedge \psi) \equiv G \varphi \wedge G \psi$$

$$F \varphi \equiv T U \varphi$$

$$G \varphi \equiv \perp R \varphi$$

$$\varphi W \psi \equiv \psi R(\varphi \vee \psi)$$

$$\varphi R \psi \equiv \psi W(\varphi \wedge \psi)$$

Part I: Propositional Logic

algebraic normal forms, binary decision diagrams, conjunctive normal forms, DPLL, Horn formulas, natural deduction, Post's adequacy theorem, resolution, SAT, semantics, sorting networks, soundness and completeness, syntax, Tseitin's transformation

Part II: Predicate Logic

natural deduction, quantifier equivalences, resolution, semantics, Skolemization, syntax, undecidability, unification

Part III: Model Checking

adequacy, branching-time temporal logic, CTL*, fairness, linear-time temporal logic, model checking algorithms, symbolic model checking

Part I: Propositional Logic

algebraic normal forms, binary decision diagrams, conjunctive normal forms, DPLL, Horn formulas, natural deduction, Post's adequacy theorem, resolution, SAT, semantics, sorting networks, soundness and completeness, syntax, Tseitin's transformation

Part II: Predicate Logic

natural deduction, quantifier equivalences, resolution, semantics, Skolemization, syntax, undecidability, unification

Part III: Model Checking

adequacy, branching-time temporal logic, CTL*, fairness, linear-time temporal logic, model checking algorithms, symbolic model checking

Outline

1. Summary of Previous Lecture

2. Adequacy

LTL CTL

3. Fairness

4. Intermezzo

5. LTL Model Checking Algorithm

6. Further Reading

7. Exam

Theorem

$\{X, U\}$, $\{X, W\}$ and $\{X, R\}$ are **adequate** sets of temporal connectives for LTL

Theorem

$\{X, U\}$, $\{X, W\}$ and $\{X, R\}$ are **adequate** sets of temporal connectives for LTL

Proof

$$F\varphi \equiv \top U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

$$\varphi R \psi \equiv \neg(\neg \varphi U \neg \psi)$$

$$\varphi W \psi \equiv \varphi U \psi \vee G\varphi$$

Theorem

$\{X, U\}$, $\{X, W\}$ and $\{X, R\}$ are **adequate** sets of temporal connectives for LTL

Proof

$$F\varphi \equiv \top U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

$$\varphi R \psi \equiv \neg(\neg \varphi U \neg \psi)$$

$$\varphi W \psi \equiv \varphi U \psi \vee G\varphi$$

$$\varphi R \psi \equiv \psi W(\varphi \wedge \psi)$$

$$\varphi U \psi \equiv \neg(\neg \varphi R \neg \psi)$$

$$F\varphi \equiv \top U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

Theorem

$\{X, U\}$, $\{X, W\}$ and $\{X, R\}$ are **adequate** sets of temporal connectives for LTL

Proof

$$F\varphi \equiv T U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

$$\varphi R \psi \equiv \neg(\neg \varphi U \neg \psi)$$

$$\varphi W \psi \equiv \varphi U \psi \vee G \varphi$$

$$\varphi R \psi \equiv \psi W (\varphi \wedge \psi)$$

$$\varphi U \psi \equiv \neg(\neg \varphi R \neg \psi)$$

$$F\varphi \equiv T U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

$$\varphi U \psi \equiv \neg(\neg \varphi R \neg \psi)$$

$$F\varphi \equiv T U \varphi$$

$$G\varphi \equiv \neg F \neg \varphi$$

$$\varphi W \psi \equiv \varphi U \psi \vee G \varphi$$

Theorem

$\{X, U\}$, $\{X, W\}$ and $\{X, R\}$ are adequate sets of temporal connectives for LTL

Proof

$$\begin{array}{lll} F\varphi \equiv \top U \varphi & \varphi R \psi \equiv \psi W (\varphi \wedge \psi) & \varphi U \psi \equiv \neg(\neg\varphi R \neg\psi) \\ G\varphi \equiv \neg F \neg\varphi & \varphi U \psi \equiv \neg(\neg\varphi R \neg\psi) & F\varphi \equiv \top U \varphi \\ \varphi R \psi \equiv \neg(\neg\varphi U \neg\psi) & F\varphi \equiv \top U \varphi & G\varphi \equiv \neg F \neg\varphi \\ \varphi W \psi \equiv \varphi U \psi \vee G\varphi & G\varphi \equiv \neg F \neg\varphi & \varphi W \psi \equiv \varphi U \psi \vee G\varphi \end{array}$$

Theorem

$\{U, R\}$, $\{U, W\}$, $\{U, G\}$, $\{F, W\}$ and $\{F, R\}$ are **adequate** sets of temporal connectives for LTL fragment consisting of **negation-normal forms** without X

Outline

1. Summary of Previous Lecture

2. Adequacy

LTL CTL

3. Fairness

4. Intermezzo

5. LTL Model Checking Algorithm

6. Further Reading

7. Exam

Theorem

set of temporal connectives is **adequate** for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\Leftarrow)

▶ $AX\varphi \equiv \neg EX\neg\varphi$ and $EX\varphi \equiv \neg AX\neg\varphi$

Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ \text{EU} \end{array} \right.$

Proof (\Leftarrow)

- ▶ $AX \varphi \equiv \neg EX \neg \varphi$ and $EX \varphi \equiv \neg AX \neg \varphi$
- ▶ $EF \varphi \equiv E[T \cup \varphi]$
- ▶ $AG \varphi \equiv \neg EF \neg \varphi$

Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\Leftarrow)

- ▶ $AX \varphi \equiv \neg EX \neg \varphi$ and $EX \varphi \equiv \neg AX \neg \varphi$
- ▶ $EF \varphi \equiv E[T \cup \varphi]$
- ▶ $AG \varphi \equiv \neg EF \neg \varphi$
- ▶ $A[\varphi \cup \psi] \equiv \neg(E[\neg \psi \cup (\neg \varphi \wedge \neg \psi)] \vee EG \neg \psi)$
- ▶ $AF \varphi \equiv A[T \cup \varphi]$

Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\Leftarrow)

- ▶ $AX \varphi \equiv \neg EX \neg \varphi$ and $EX \varphi \equiv \neg AX \neg \varphi$
- ▶ $EF \varphi \equiv E[T \cup \varphi]$
- ▶ $AG \varphi \equiv \neg EF \neg \varphi$
- ▶ $A[\varphi \cup \psi] \equiv \neg(E[\neg \psi \cup (\neg \varphi \wedge \neg \psi)] \vee EG \neg \psi)$
- ▶ $EG \varphi \equiv \neg AF \neg \varphi$

Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, \mathbf{AU}\} \\ EU \end{array} \right.$

Proof (\Leftarrow)

- ▶ $AX \varphi \equiv \neg EX \neg \varphi$ and $EX \varphi \equiv \neg AX \neg \varphi$
- ▶ $EF \varphi \equiv E[T \cup \varphi]$
- ▶ $AG \varphi \equiv \neg EF \neg \varphi$

- ▶ $AF \varphi \equiv A[T \cup \varphi]$
- ▶ $EG \varphi \equiv \neg AF \neg \varphi$

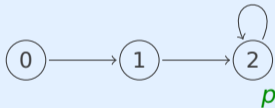
Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\implies)

► consider model \mathcal{M}



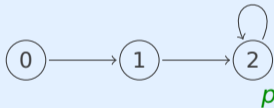
Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\implies)

► consider model \mathcal{M}



► $\mathcal{M}, 0 \not\models EXp$ and $\mathcal{M}, 1 \models EXp$

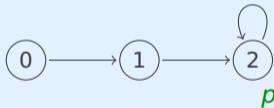
Theorem

set of temporal connectives is adequate for CTL \iff

it contains $\left\{ \begin{array}{l} \text{at least one of } \{AX, EX\} \\ \text{at least one of } \{EG, AF, AU\} \\ EU \end{array} \right.$

Proof (\implies)

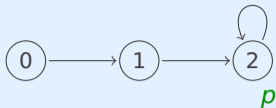
▶ consider model \mathcal{M}



▶ $\mathcal{M}, 0 \not\models EXp$ and $\mathcal{M}, 1 \models EXp$

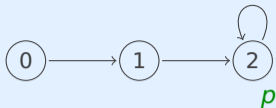
▶ for every CTL formula φ not containing EX and AX:

$$\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 1 \models \varphi$$



Proof (\implies , cont'd)

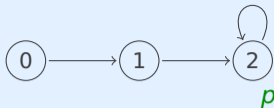
induction on φ



Proof (\Rightarrow , cont'd)

induction on φ

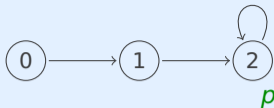
- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$



Proof (\Rightarrow , cont'd)

induction on φ

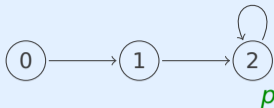
- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$



Proof (\Rightarrow , cont'd)

induction on φ

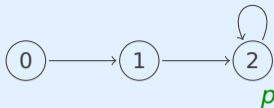
- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi$



Proof (\implies , cont'd)

induction on φ

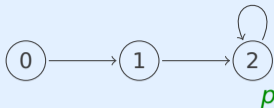
- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi \iff \mathcal{M}, 1 \not\models \psi$



Proof (\Rightarrow , cont'd)

induction on φ

- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi \iff \mathcal{M}, 1 \not\models \psi \iff \mathcal{M}, 1 \models \varphi$

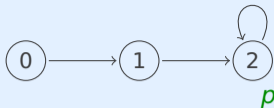


Proof (\Rightarrow , cont'd)

induction on φ

- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi \iff \mathcal{M}, 1 \not\models \psi \iff \mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \psi_1 \wedge \psi_2$ then

$$\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 0 \models \psi_2$$

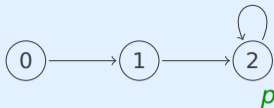


Proof (\Rightarrow , cont'd)

induction on φ

- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi \iff \mathcal{M}, 1 \not\models \psi \iff \mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \psi_1 \wedge \psi_2$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 0 \models \psi_2 \\
 &\iff \mathcal{M}, 1 \models \psi_1 \text{ and } \mathcal{M}, 1 \models \psi_2
 \end{aligned}$$

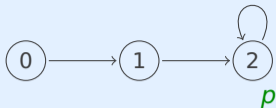


Proof (\Rightarrow , cont'd)

induction on φ

- ▶ if φ is atom or $\varphi = \perp$ then $\mathcal{M}, 0 \not\models \varphi$ and $\mathcal{M}, 1 \not\models \varphi$
- ▶ if $\varphi = \top$ then $\mathcal{M}, 0 \models \varphi$ and $\mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \neg\psi$ then $\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \not\models \psi \iff \mathcal{M}, 1 \not\models \psi \iff \mathcal{M}, 1 \models \varphi$
- ▶ if $\varphi = \psi_1 \wedge \psi_2$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 0 \models \psi_2 \\
 &\iff \mathcal{M}, 1 \models \psi_1 \text{ and } \mathcal{M}, 1 \models \psi_2 \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$

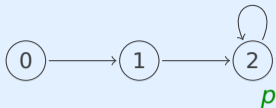


Proof (\implies , cont'd)

induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\}$$

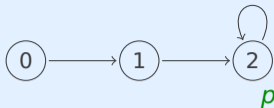


Proof (\Rightarrow , cont'd)

induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\begin{aligned} \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\} \\ &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{1, 2\} \end{aligned}$$

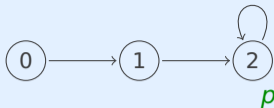


Proof (\implies , cont'd)

induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{1, 2\} \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$



Proof (\Rightarrow , cont'd)

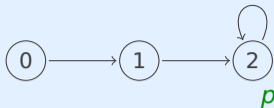
induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{1, 2\} \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$

► if $\varphi = \text{AG } \psi$ or $\varphi = \text{EG } \psi$ then

$$\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, i \models \psi \text{ for all } i \in \{0, 1, 2\}$$



Proof (\Rightarrow , cont'd)

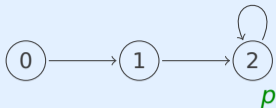
induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{1, 2\} \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$

► if $\varphi = \text{AG } \psi$ or $\varphi = \text{EG } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for all } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for all } i \in \{1, 2\}
 \end{aligned}$$



Proof (\Rightarrow , cont'd)

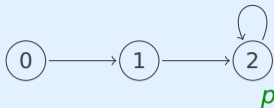
induction on φ

► if $\varphi = \text{AF } \psi$ or $\varphi = \text{EF } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for some } i \in \{1, 2\} \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$

► if $\varphi = \text{AG } \psi$ or $\varphi = \text{EG } \psi$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, i \models \psi \text{ for all } i \in \{0, 1, 2\} \\
 &\iff \mathcal{M}, i \models \psi \text{ for all } i \in \{1, 2\} \iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$



Proof (\Rightarrow , cont'd)

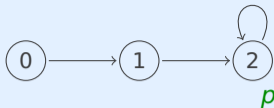
induction on φ

► if $\varphi = A[\psi_1 U \psi_2]$ or $\varphi = E[\psi_1 U \psi_2]$ then

$$\mathcal{M}, 0 \models \varphi \iff \mathcal{M}, 0 \models \psi_2 \text{ or}$$

$$\mathcal{M}, 1 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ or}$$

$$\mathcal{M}, 2 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 1 \models \psi_1$$

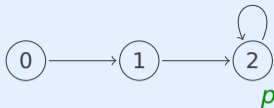


Proof (\Rightarrow , cont'd)

induction on φ

► if $\varphi = A[\psi_1 U \psi_2]$ or $\varphi = E[\psi_1 U \psi_2]$ then

$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, 0 \models \psi_2 \text{ or} \\
 &\quad \mathcal{M}, 1 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ or} \\
 &\quad \mathcal{M}, 2 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 1 \models \psi_1 \\
 &\iff \mathcal{M}, 1 \models \psi_2 \text{ or} \\
 &\quad \mathcal{M}, 2 \models \psi_2 \text{ and } \mathcal{M}, 1 \models \psi_1
 \end{aligned}$$



Proof (\Rightarrow , cont'd)

induction on φ

► if $\varphi = A[\psi_1 U \psi_2]$ or $\varphi = E[\psi_1 U \psi_2]$ then

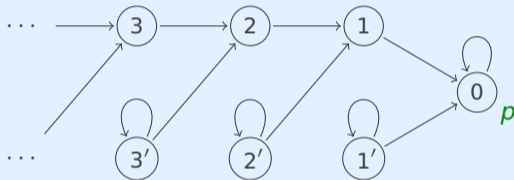
$$\begin{aligned}
 \mathcal{M}, 0 \models \varphi &\iff \mathcal{M}, 0 \models \psi_2 \text{ or} \\
 &\quad \mathcal{M}, 1 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ or} \\
 &\quad \mathcal{M}, 2 \models \psi_2 \text{ and } \mathcal{M}, 0 \models \psi_1 \text{ and } \mathcal{M}, 1 \models \psi_1 \\
 &\iff \mathcal{M}, 1 \models \psi_2 \text{ or} \\
 &\quad \mathcal{M}, 2 \models \psi_2 \text{ and } \mathcal{M}, 1 \models \psi_1 \\
 &\iff \mathcal{M}, 1 \models \varphi
 \end{aligned}$$

Theorem

... at least one of {EG, AF, AU}

Proof (\implies)

► consider model \mathcal{M}

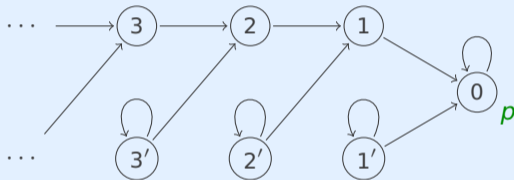


Theorem

... at least one of {EG, AF, AU}

Proof (\implies)

► consider model \mathcal{M}



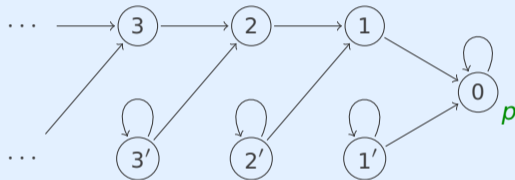
► $\mathcal{M}, i \models \text{AF } p$ for all $i \geq 0$

Theorem

... at least one of {EG, AF, AU}

Proof (\implies)

► consider model \mathcal{M}



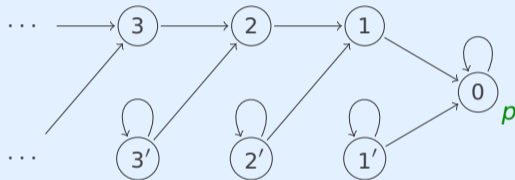
► $\mathcal{M}, i \models \text{AF } p$ for all $i \geq 0$ and $\mathcal{M}, i' \not\models \text{AF } p$ for all $i' > 0$

Theorem

... at least one of {EG, AF, AU}

Proof (\implies)

► consider model \mathcal{M}



► $\mathcal{M}, i \models \text{AF } p$ for all $i \geq 0$ and $\mathcal{M}, i' \not\models \text{AF } p$ for all $i' > 0$

► for every CTL formula φ not containing EG, AF and AU there exists $n_\varphi > 0$ such that

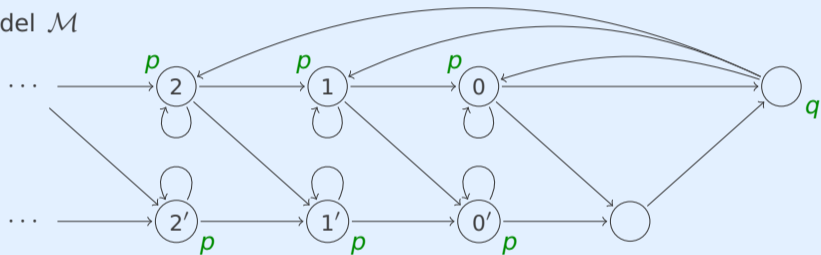
$$\mathcal{M}, n_\varphi \models \varphi \iff \mathcal{M}, n'_\varphi \models \varphi$$

Theorem

... EU

Proof (\Rightarrow)

► consider model \mathcal{M}

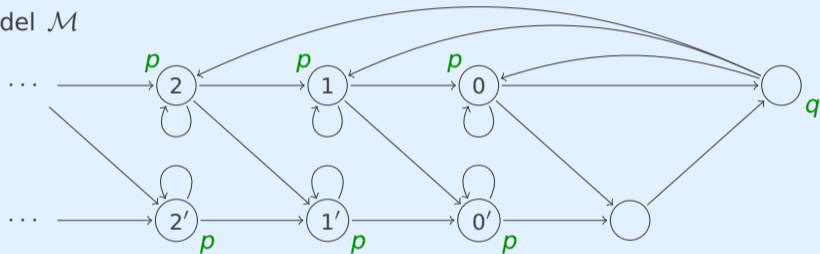


Theorem

... EU

Proof (\Rightarrow)

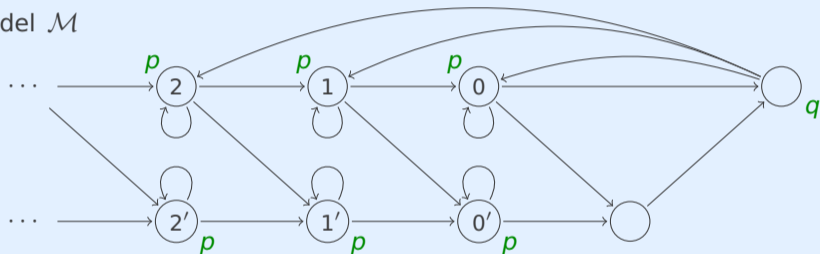
► consider model \mathcal{M}



► $\mathcal{M}, i \models E[pUq]$ and $\mathcal{M}, i' \not\models E[pUq]$ for all $i \geq 0$

Proof (\Rightarrow)

- ▶ consider model \mathcal{M}



- ▶ $\mathcal{M}, i \models E[pUq]$ and $\mathcal{M}, i' \not\models E[pUq]$ for all $i \geq 0$
- ▶ for every CTL formula φ not containing EU there exists $n_\varphi \geq 0$ such that

$$\mathcal{M}, n_\varphi \models \varphi \iff \mathcal{M}, n'_\varphi \models \varphi$$

Outline

1. Summary of Previous Lecture
2. Adequacy
- 3. Fairness**
4. Intermezzo
5. LTL Model Checking Algorithm
6. Further Reading
7. Exam

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL
- ▶ eliminate such behaviour by imposing **fairness constraints**

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL
- ▶ eliminate such behaviour by imposing fairness constraints

Definitions

- ▶ path $s_1 \rightarrow s_2 \rightarrow \dots$ is **fair** with respect to set C of CTL formulas if for all $\psi \in C$
 $s_i \models \psi$ for infinitely many i

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL
- ▶ eliminate such behaviour by imposing fairness constraints

Definitions

- ▶ path $s_1 \rightarrow s_2 \rightarrow \dots$ is fair with respect to set C of CTL formulas if for all $\psi \in C$
 $s_i \models \psi$ for infinitely many i ($GF \psi$ in LTL)

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL
- ▶ eliminate such behaviour by imposing fairness constraints

Definitions

- ▶ path $s_1 \rightarrow s_2 \rightarrow \dots$ is fair with respect to set C of CTL formulas if for all $\psi \in C$
$$s_i \models \psi \text{ for infinitely many } i \quad (\text{GF } \psi \text{ in LTL})$$
- ▶ formulas in C are called **fairness constraints**

Motivation

- ▶ model may contain behaviour which is unrealistic or guaranteed not to happen
- ▶ such behaviour is (typically) not expressible in CTL
- ▶ eliminate such behaviour by imposing fairness constraints

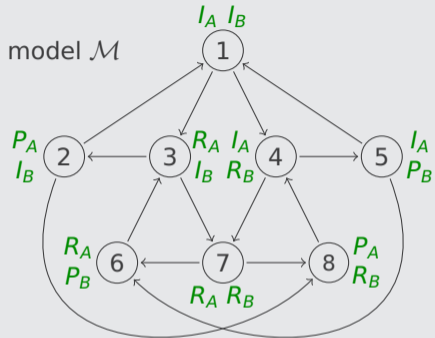
Definitions

- ▶ path $s_1 \rightarrow s_2 \rightarrow \dots$ is fair with respect to set C of CTL formulas if for all $\psi \in C$

$$s_i \models \psi \text{ for infinitely many } i \quad (\text{GF } \psi \text{ in LTL})$$

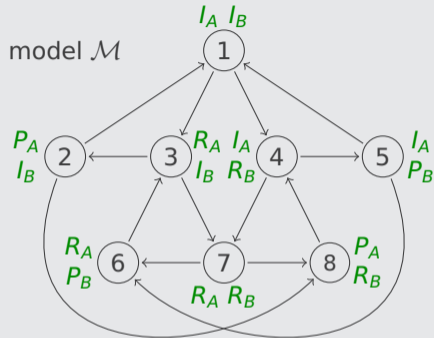
- ▶ formulas in C are called fairness constraints
- ▶ A_C (E_C) denotes A (E) restricted to paths that are fair with respect to C

Example



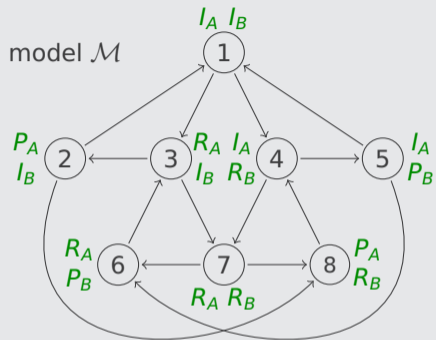
- ▶ path $1(376)^\omega$ is fair with respect to $\{I_B, P_B\}$

Example



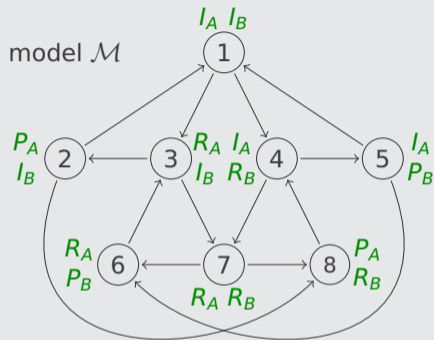
- ▶ path $1(376)^\omega$ is fair with respect to $\{I_B, P_B\}$ but not with respect to $\{I_A\}$

Example



- ▶ path $1(376)^\omega$ is fair with respect to $\{I_B, P_B\}$ but not with respect to $\{I_A\}$
- ▶ $\mathcal{M}, 1 \not\models A_{\{R_B\}} F P_B$

Example



- ▶ path $1(376)^\omega$ is fair with respect to $\{I_B, P_B\}$ but not with respect to $\{I_A\}$
- ▶ $\mathcal{M}, 1 \not\models A_{\{R_B\}} F P_B$ because path $1(478)^\omega$ is fair with respect to R_B but $\mathcal{M}, i \not\models P_B$ for $i \in \{1, 4, 7, 8\}$

Lemma

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G \top)]$$

Lemma

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

Lemma

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

New Algorithm (CTL Model Checking with Fairness Constraints)

required only for $E_c G \varphi$:

- ① restrict graph to states satisfying φ

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

New Algorithm (CTL Model Checking with Fairness Constraints)

required only for $E_c G \varphi$:

- ① restrict graph to states satisfying φ
- ② compute non-trivial strongly connected components (SCCs)

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

New Algorithm (CTL Model Checking with Fairness Constraints)

required only for $E_c G \varphi$:

- ① restrict graph to states satisfying φ
- ② compute non-trivial strongly connected components (SCCs)
- ③ **remove SCC S if there exists constraint $\psi \in C$ such that $s \not\models \psi$ for all states $s \in S$**

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

New Algorithm (CTL Model Checking with Fairness Constraints)

required only for $E_c G \varphi$:

- ① restrict graph to states satisfying φ
- ② compute non-trivial strongly connected components (SCCs)
- ③ remove SCC S if there exists constraint $\psi \in C$ such that $s \not\models \psi$ for all states $s \in S$
- ④ label all states in resulting SCCs

$$E_c[\varphi U \psi] \equiv E[\varphi U (\psi \wedge E_c G T)]$$

$$E_c X \varphi \equiv EX(\varphi \wedge E_c G T)$$

New Algorithm (CTL Model Checking with Fairness Constraints)

required only for $E_c G \varphi$:

- ① restrict graph to states satisfying φ
- ② compute non-trivial strongly connected components (SCCs)
- ③ remove SCC S if there exists constraint $\psi \in C$ such that $s \not\models \psi$ for all states $s \in S$
- ④ label all states in resulting SCCs
- ⑤ compute and label all states that can reach labelled state in restricted graph computed in step ①

Outline

1. Summary of Previous Lecture
2. Adequacy
3. Fairness
- 4. Intermezzo**
5. LTL Model Checking Algorithm
6. Further Reading
7. Exam

Question

Which of the following statements hold for all models $\mathcal{M} = (S, \rightarrow, L)$ and states $s \in S$?

- A** $\mathcal{M}, s \models A_{\{p\}} F(p \vee AG p)$
- B** $\mathcal{M}, s \models E_{\{p\}} X(p \wedge EX p)$
- C** $\mathcal{M}, s \models E_{\{p \vee q\}} [p U q]$
- D** $\mathcal{M}, s \models A_{\{p\}} G(AF p)$



Question

Which of the following statements hold for all models $\mathcal{M} = (S, \rightarrow, L)$ and states $s \in S$?

- A** $\mathcal{M}, s \models A_{\{p\}} F(p \vee AG p)$
- B** $\mathcal{M}, s \models E_{\{p\}} X(p \wedge EX p)$
- C** $\mathcal{M}, s \models E_{\{p \vee q\}} [p U q]$
- D** $\mathcal{M}, s \models A_{\{p\}} G(AF p)$



Outline

1. Summary of Previous Lecture
2. Adequacy
3. Fairness
4. Intermezzo
- 5. LTL Model Checking Algorithm**
6. Further Reading
7. Exam

Theorem

satisfaction of LTL formulas in finite models is **decidable**

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with **fairness constraints**

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use automata techniques

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use **automata techniques**

Basic Strategy

$\mathcal{M}, s \models \varphi ?$

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use **automata techniques**

Basic Strategy

$\mathcal{M}, s \models \varphi$?

- ▶ construct **labelled Büchi automaton** $A_{\neg\varphi}$ for $\neg\varphi$

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use **automata techniques**

Basic Strategy

$\mathcal{M}, s \models \varphi$?

- ▶ construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- ▶ combine $A_{\neg\varphi}$ and \mathcal{M} into single automaton $A_{\neg\varphi} \times \mathcal{M}$

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use **automata techniques**

Basic Strategy

$\mathcal{M}, s \models \varphi$?

- ▶ construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- ▶ combine $A_{\neg\varphi}$ and \mathcal{M} into single automaton $A_{\neg\varphi} \times \mathcal{M}$
- ▶ determine whether there exists accepting path in $A_{\neg\varphi} \times \mathcal{M}$

Theorem

satisfaction of LTL formulas in finite models is decidable

Two Approaches

- ① translate into CTL model checking with fairness constraints
- ② use **automata techniques**

Basic Strategy

$\mathcal{M}, s \models \varphi$?

- ▶ construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- ▶ combine $A_{\neg\varphi}$ and \mathcal{M} into single automaton $A_{\neg\varphi} \times \mathcal{M}$
- ▶ determine whether there exists accepting path in $A_{\neg\varphi} \times \mathcal{M}$

formula φ in LTL fragment with U and X as only temporal operators

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

- ① **consistent with respect to propositional logic**

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① **consistent with respect to propositional logic:** for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① **consistent with respect to propositional logic:** for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① **consistent with respect to propositional logic:** for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① consistent with respect to propositional logic: for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

② **locally consistent with respect to U**

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① consistent with respect to propositional logic: for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

② **locally consistent with respect to U**: for all $\varphi_1 \text{ U } \varphi_2 \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_2 \in B \implies \varphi_1 \text{ U } \varphi_2 \in B$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① consistent with respect to propositional logic: for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

② **locally consistent with respect to U**: for all $\varphi_1 \text{ U } \varphi_2 \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_2 \in B \implies \varphi_1 \text{ U } \varphi_2 \in B$$

$$\blacktriangleright \varphi_1 \text{ U } \varphi_2 \in B \text{ and } \varphi_2 \notin B \implies \varphi_1 \in B$$

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① consistent with respect to propositional logic: for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

② locally consistent with respect to U: for all $\varphi_1 \text{ U } \varphi_2 \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_2 \in B \implies \varphi_1 \text{ U } \varphi_2 \in B$$

$$\blacktriangleright \varphi_1 \text{ U } \varphi_2 \in B \text{ and } \varphi_2 \notin B \implies \varphi_1 \in B$$

③ **maximal**

Definition

set $B \subseteq \mathcal{C}(\varphi)$ is **elementary** if it is

① consistent with respect to propositional logic: for all $\varphi_1 \wedge \varphi_2 \in \mathcal{C}(\varphi)$ and $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_1 \wedge \varphi_2 \in B \iff \varphi_1 \in B \text{ and } \varphi_2 \in B$$

$$\blacktriangleright \psi \in B \implies \neg\psi \notin B$$

$$\blacktriangleright \top \in \mathcal{C}(\varphi) \implies \top \in B$$

② locally consistent with respect to \cup : for all $\varphi_1 \cup \varphi_2 \in \mathcal{C}(\varphi)$

$$\blacktriangleright \varphi_2 \in B \implies \varphi_1 \cup \varphi_2 \in B$$

$$\blacktriangleright \varphi_1 \cup \varphi_2 \in B \text{ and } \varphi_2 \notin B \implies \varphi_1 \in B$$

③ **maximal**: for all $\psi \in \mathcal{C}(\varphi)$

$$\blacktriangleright \psi \notin B \implies \neg\psi \in B$$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$

► $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$

► $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ not elementary

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$ elementary

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$ elementary
- ▶ $\{a, \neg b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$

formula φ in LTL fragment with U and X as only temporal operators

Definition

closure $\mathcal{C}(\varphi)$ of φ consists of all subformulas of φ and their negations, identifying $\neg\neg\psi$ and ψ

Example

$$\mathcal{C}(aU(\neg a \wedge b)) = \{a, \neg a, b, \neg b, \neg a \wedge b, \neg(\neg a \wedge b), aU(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$$

- ▶ $\{a, b, \neg a \wedge b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ elementary
- ▶ $\{\neg a, \neg b, \neg(\neg a \wedge b), aU(\neg a \wedge b)\}$ not elementary
- ▶ $\{a, b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$ elementary
- ▶ $\{a, \neg b, \neg(\neg a \wedge b), \neg(aU(\neg a \wedge b))\}$ elementary

Definitions

- ▶ **states** of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$

Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ **initial** states are those states containing φ

Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ **transition relation** Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X \psi \in \mathcal{C}(\varphi)$ $X\psi \in A \iff \psi \in B$

Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ **transition relation** Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X\psi \in \mathcal{C}(\varphi)$ $X\psi \in A \iff \psi \in B$
 - ② for all $\varphi_1 \mathbf{U} \varphi_2 \in \mathcal{C}(\varphi)$ $\varphi_1 \mathbf{U} \varphi_2 \in A \iff \varphi_2 \in A$ or both $\varphi_1 \in A$ and $\varphi_1 \mathbf{U} \varphi_2 \in B$

Example 1

$$\varphi = X a$$

Example 1

$$\varphi = X a$$

$$\triangleright C(\varphi) = \{a, \neg a, X a, \neg X a\}$$

Example 1

$$\varphi = X a$$

► $\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states ① $\{a, X a\}$ ② $\{a, \neg X a\}$ ③ $\{\neg a, X a\}$ ④ $\{\neg a, \neg X a\}$

Example 1

$$\varphi = X a$$

$$\triangleright \mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\triangleright \text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\triangleright \text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

Example 1

$$\varphi = X a$$

▶ $\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$

▶ states 1 $\{a, X a\}$ 2 $\{a, \neg X a\}$ 3 $\{\neg a, X a\}$ 4 $\{\neg a, \neg X a\}$

▶ initial states 1 3

▶ transitions

	1	2	3	4
1				
2				
3				
4				

Example 1

$$\varphi = X a$$

► $\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states 1 $\{a, X a\}$ 2 $\{a, \neg X a\}$ 3 $\{\neg a, X a\}$ 4 $\{\neg a, \neg X a\}$

► initial states 1 3

► transitions

	1	2	3	4
1	✓	✓		
2				
3				
4				

Example 1

$$\varphi = X a$$

► $C(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states 1 $\{a, X a\}$ 2 $\{a, \neg X a\}$ 3 $\{\neg a, X a\}$ 4 $\{\neg a, \neg X a\}$

► initial states 1 3

► transitions

	1	2	3	4
1	✓	✓		
2			✓	✓
3				
4				

Example 1

$$\varphi = X a$$

► $\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states 1 $\{a, X a\}$ 2 $\{a, \neg X a\}$ 3 $\{\neg a, X a\}$ 4 $\{\neg a, \neg X a\}$

► initial states 1 3

► transitions

	1	2	3	4
1	✓	✓		
2			✓	✓
3	✓	✓		
4				

Example 1

$$\varphi = X a$$

► $\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states 1 $\{a, X a\}$ 2 $\{a, \neg X a\}$ 3 $\{\neg a, X a\}$ 4 $\{\neg a, \neg X a\}$

► initial states 1 3

► transitions

	1	2	3	4
1	✓	✓		
2			✓	✓
3	✓	✓		
4			✓	✓

Example 1

$$\varphi = X a$$

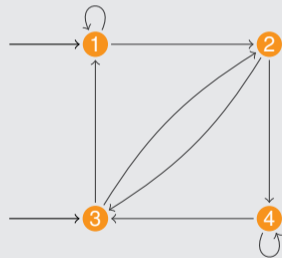
► $C(\varphi) = \{a, \neg a, X a, \neg X a\}$

► states 1 {a, X a} 2 {a, $\neg X a$ } 3 { $\neg a$, X a} 4 { $\neg a$, $\neg X a$ }

► initial states 1 3

► transitions

	1	2	3	4
1	✓	✓		
2			✓	✓
3	✓	✓		
4			✓	✓



Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ transition relation Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X\psi \in \mathcal{C}(\varphi)$ $X\psi \in A \iff \psi \in B$
 - ② for all $\varphi_1 \mathbf{U} \varphi_2 \in \mathcal{C}(\varphi)$ $\varphi_1 \mathbf{U} \varphi_2 \in A \iff \varphi_2 \in A$ or both $\varphi_1 \in A$ and $\varphi_1 \mathbf{U} \varphi_2 \in B$
- ▶ **trace** is infinite sequence of valuations of propositional atoms

Example 1

$$\varphi = X a$$

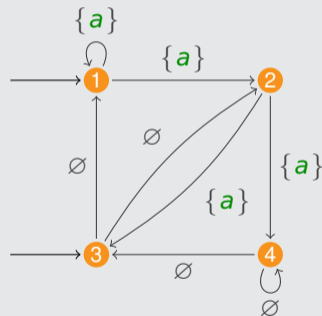
$$\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

transitions

	$\textcircled{1}$	$\textcircled{2}$	$\textcircled{3}$	$\textcircled{4}$	
$\textcircled{1}$	✓	✓			$\{a\}$
$\textcircled{2}$			✓	✓	$\{a\}$
$\textcircled{3}$	✓	✓			\emptyset
$\textcircled{4}$			✓	✓	\emptyset



Example 1

$$\varphi = X a$$

$$\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

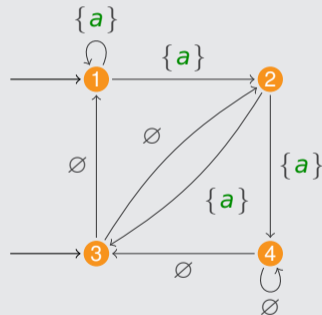
transitions

	1	2	3	4	
1	✓	✓			{a}
2			✓	✓	{a}
3	✓	✓			∅
4			✓	✓	∅

$$\text{trace } t_1 = \{a\} \{a\} \{a\} \emptyset^\omega$$

$$\text{trace } t_2 = \emptyset \{a\} \emptyset \{a\}^\omega$$

$$\text{trace } t_3 = \{a\} \emptyset \emptyset \{a\}^\omega$$



Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ transition relation Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X\psi \in \mathcal{C}(\varphi)$ $X\psi \in A \iff \psi \in B$
 - ② for all $\varphi_1 \mathbf{U} \varphi_2 \in \mathcal{C}(\varphi)$ $\varphi_1 \mathbf{U} \varphi_2 \in A \iff \varphi_2 \in A$ or both $\varphi_1 \in A$ and $\varphi_1 \mathbf{U} \varphi_2 \in B$
- ▶ trace is infinite sequence of valuations of propositional atoms
- ▶ trace t is **accepted** if there exists path π in A_φ such that
 - ① π starts in initial state of A_φ

Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ transition relation Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X\psi \in \mathcal{C}(\varphi)$ $X\psi \in A \iff \psi \in B$
 - ② for all $\varphi_1 \mathbf{U} \varphi_2 \in \mathcal{C}(\varphi)$ $\varphi_1 \mathbf{U} \varphi_2 \in A \iff \varphi_2 \in A$ or both $\varphi_1 \in A$ and $\varphi_1 \mathbf{U} \varphi_2 \in B$
- ▶ trace is infinite sequence of valuations of propositional atoms
- ▶ trace t is **accepted** if there exists path π in A_φ such that
 - ① π starts in initial state of A_φ
 - ② π corresponds to trace t : $t_i = \{p \in \pi_i \mid p \text{ is atom}\}$ for all i

Definitions

- ▶ states of **automaton** A_φ are elementary subsets of $\mathcal{C}(\varphi)$
- ▶ initial states are those states containing φ
- ▶ transition relation Δ of A_φ : $(A, B) \in \Delta$ if and only if
 - ① for all $X \psi \in \mathcal{C}(\varphi)$ $X \psi \in A \iff \psi \in B$
 - ② for all $\varphi_1 \mathbf{U} \varphi_2 \in \mathcal{C}(\varphi)$ $\varphi_1 \mathbf{U} \varphi_2 \in A \iff \varphi_2 \in A$ or both $\varphi_1 \in A$ and $\varphi_1 \mathbf{U} \varphi_2 \in B$
- ▶ trace is infinite sequence of valuations of propositional atoms
- ▶ trace t is **accepted** if there exists path π in A_φ such that
 - ① π starts in initial state of A_φ
 - ② π corresponds to trace t : $t_i = \{p \in \pi_i \mid p \text{ is atom}\}$ for all i
 - ③ π visits infinitely many states satisfying $\neg(\psi_1 \mathbf{U} \psi_2) \vee \psi_2$, for every $\psi_1 \mathbf{U} \psi_2 \in \mathcal{C}(\varphi)$

Example 1

$$\varphi = X a$$

$$\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

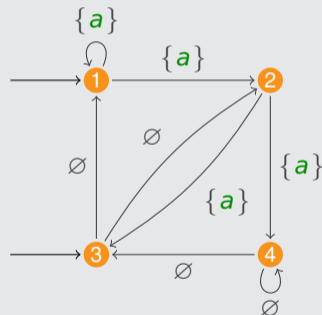
transitions

	1	2	3	4	
1	✓	✓			{a}
2			✓	✓	{a}
3	✓	✓			∅
4			✓	✓	∅

trace $t_1 = \{a\} \{a\} \{a\} \emptyset^\omega$ is accepted: path $\textcircled{1} \textcircled{1} \textcircled{2} \textcircled{4}^\omega$

trace $t_2 = \emptyset \{a\} \emptyset \{a\}^\omega$

trace $t_3 = \{a\} \emptyset \emptyset \{a\}^\omega$



Example 1

$$\varphi = X a$$

$$\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

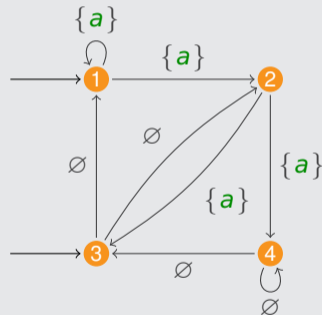
transitions

	$\textcircled{1}$	$\textcircled{2}$	$\textcircled{3}$	$\textcircled{4}$	
$\textcircled{1}$	✓	✓			$\{a\}$
$\textcircled{2}$			✓	✓	$\{a\}$
$\textcircled{3}$	✓	✓			\emptyset
$\textcircled{4}$			✓	✓	\emptyset

trace $t_1 = \{a\} \{a\} \{a\} \emptyset^\omega$ is accepted: path $\textcircled{1} \textcircled{1} \textcircled{2} \textcircled{4}^\omega$

trace $t_2 = \emptyset \{a\} \emptyset \{a\}^\omega$ is accepted: path $\textcircled{3} \textcircled{2} \textcircled{3} \textcircled{1}^\omega$

trace $t_3 = \{a\} \emptyset \emptyset \{a\}^\omega$



Example 1

$$\varphi = X a$$

$$\mathcal{C}(\varphi) = \{a, \neg a, X a, \neg X a\}$$

$$\text{states} \quad \textcircled{1} \{a, X a\} \quad \textcircled{2} \{a, \neg X a\} \quad \textcircled{3} \{\neg a, X a\} \quad \textcircled{4} \{\neg a, \neg X a\}$$

$$\text{initial states} \quad \textcircled{1} \quad \textcircled{3}$$

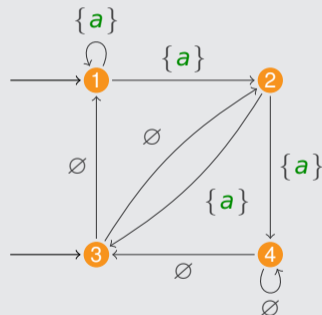
transitions

	1	2	3	4	
1	✓	✓			{a}
2			✓	✓	{a}
3	✓	✓			∅
4			✓	✓	∅

trace $t_1 = \{a\} \{a\} \{a\} \emptyset^\omega$ is accepted: path $\textcircled{1} \textcircled{1} \textcircled{2} \textcircled{4}^\omega$

trace $t_2 = \emptyset \{a\} \emptyset \{a\}^\omega$ is accepted: path $\textcircled{3} \textcircled{2} \textcircled{3} \textcircled{1}^\omega$

trace $t_3 = \{a\} \emptyset \emptyset \{a\}^\omega$ is not accepted



Example ②

$$\varphi = a U b$$

Example 2

$$\varphi = a \cup b$$

$$\triangleright \mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$$

Example 2

$$\varphi = a \cup b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

Example 2

$$\varphi = a \text{ U } b$$

$$\text{▶ } \mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \text{ U } b, \neg(a \text{ U } b)\}$$

$$\text{▶ states} \quad \textcircled{1} \{a, b, \varphi\} \quad \textcircled{2} \{\neg a, b, \varphi\} \quad \textcircled{3} \{a, \neg b, \varphi\} \quad \textcircled{4} \{a, \neg b, \neg\varphi\} \quad \textcircled{5} \{\neg a, \neg b, \neg\varphi\}$$

$$\text{▶ initial states} \quad \textcircled{1} \quad \textcircled{2} \quad \textcircled{3}$$

Example 2

$$\varphi = a \cup b$$

► $C(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤
①	✓	✓	✓	✓	✓

$\{a, b\}$

Example 2

$$\varphi = a \cup b$$

► $C(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$

Example 2

$$\varphi = a \cup b$$

► $C(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$
③	✓	✓	✓			$\{a\}$

Example 2

$$\varphi = a \cup b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \cup b, \neg(a \cup b)\}$

► states 1 $\{a, b, \varphi\}$ 2 $\{\neg a, b, \varphi\}$ 3 $\{a, \neg b, \varphi\}$ 4 $\{a, \neg b, \neg\varphi\}$ 5 $\{\neg a, \neg b, \neg\varphi\}$

► initial states 1 2 3

► transitions

	1	2	3	4	5	
1	✓	✓	✓	✓	✓	$\{a, b\}$
2	✓	✓	✓	✓	✓	$\{b\}$
3	✓	✓	✓			$\{a\}$
4				✓	✓	$\{a\}$

Example 2

$$\varphi = a \text{ U } b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \text{ U } b, \neg(a \text{ U } b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$
③	✓	✓	✓			$\{a\}$
④				✓	✓	$\{a\}$
⑤	✓	✓	✓	✓	✓	\emptyset

Example 2

$$\varphi = a \text{ U } b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \text{ U } b, \neg(a \text{ U } b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$
③	✓	✓	✓			$\{a\}$
④				✓	✓	$\{a\}$
⑤	✓	✓	✓	✓	✓	\emptyset

► acceptance condition: paths cycling in state ③ are not accepting

Example 2

$$\varphi = a \text{ U } b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \text{ U } b, \neg(a \text{ U } b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$
③	✓	✓	✓			$\{a\}$
④				✓	✓	$\{a\}$
⑤	✓	✓	✓	✓	✓	\emptyset

► acceptance condition: paths cycling in state ③ are not accepting

► $\{a\}^\omega$ is rejected

Example 2

$$\varphi = a \text{ U } b$$

► $\mathcal{C}(\varphi) = \{a, \neg a, b, \neg b, a \text{ U } b, \neg(a \text{ U } b)\}$

► states ① $\{a, b, \varphi\}$ ② $\{\neg a, b, \varphi\}$ ③ $\{a, \neg b, \varphi\}$ ④ $\{a, \neg b, \neg\varphi\}$ ⑤ $\{\neg a, \neg b, \neg\varphi\}$

► initial states ① ② ③

► transitions

	①	②	③	④	⑤	
①	✓	✓	✓	✓	✓	$\{a, b\}$
②	✓	✓	✓	✓	✓	$\{b\}$
③	✓	✓	✓			$\{a\}$
④				✓	✓	$\{a\}$
⑤	✓	✓	✓	✓	✓	\emptyset

► acceptance condition: paths cycling in state ③ are not accepting

► $\{a\}^\omega$ is rejected and $\{b\} \cap \{a\}^\omega$ is accepted

Basic Strategy

$\mathcal{M}, s \models \varphi ?$

- ▶ construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- ▶ combine $A_{\neg\varphi}$ and \mathcal{M} into single automaton $A_{\neg\varphi} \times \mathcal{M}$
- ▶ determine whether there exists accepting path in $A_{\neg\varphi} \times \mathcal{M}$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = a U b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

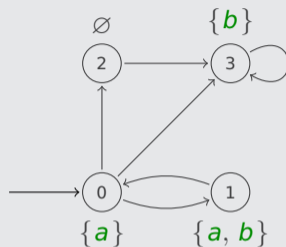
acceptance condition: paths cycling in state 3 are not accepting

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



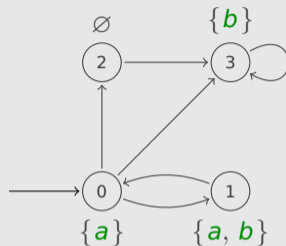
acceptance condition: paths cycling in state 3 are not accepting

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

\rightarrow 4 0

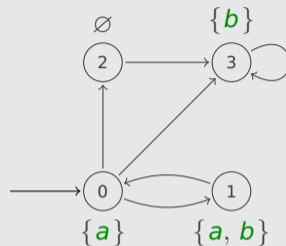
\rightarrow 5 0

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

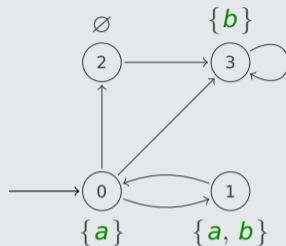
$$\begin{array}{l} \rightarrow 40 \\ \rightarrow 50 \end{array} \mid \{41, 42, 43, 51, 52, 53\}$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

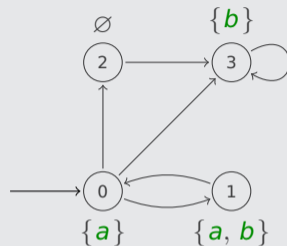
$$\begin{array}{l} \rightarrow 40 \mid \{41, 42, 43, 51, 52, 53\} \\ \rightarrow 50 \mid \emptyset \end{array}$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

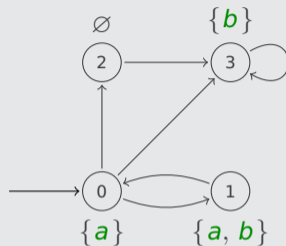
$$\begin{array}{l} \rightarrow 40 \\ \rightarrow 50 \end{array} \left| \begin{array}{l} \{ 42, 43, 51, 52, 53 \} \\ \emptyset \end{array} \right.$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

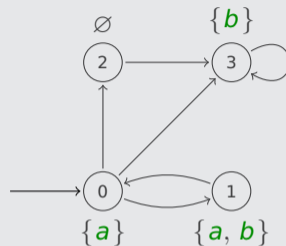
$$\begin{array}{l} \rightarrow 4\ 0 \\ \rightarrow 5\ 0 \end{array} \left| \begin{array}{l} \{ \quad \quad \quad 4\ 3, 5\ 1, 5\ 2, 5\ 3 \} \\ \emptyset \end{array} \right.$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

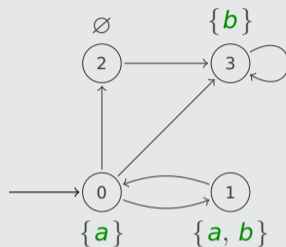
$$\begin{array}{l} \rightarrow 4\ 0 \mid \{ \quad \quad \quad 5\ 1, 5\ 2, 5\ 3 \} \\ \rightarrow 5\ 0 \mid \emptyset \end{array}$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

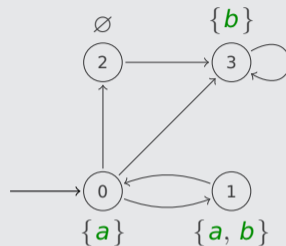
$$\begin{array}{l}
 \rightarrow 4\ 0 \mid \{ \hspace{10em} 5\ 2, 5\ 3 \} \\
 \rightarrow 5\ 0 \mid \emptyset
 \end{array}$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

\rightarrow 4 0 | { 5 2 }

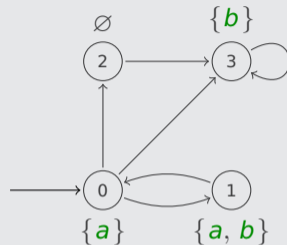
\rightarrow 5 0 | \emptyset

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

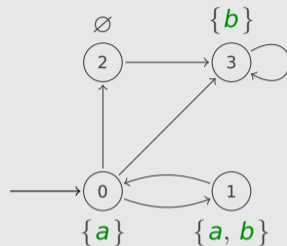
$$\begin{array}{l} \rightarrow 40 \mid \{ \quad \quad \quad 52 \quad \} \quad 52 \mid \{ 13, 23, 33, 43, 53 \} \\ \rightarrow 50 \mid \emptyset \end{array}$$

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

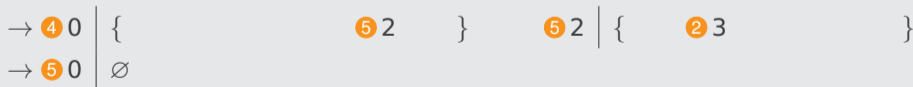
	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

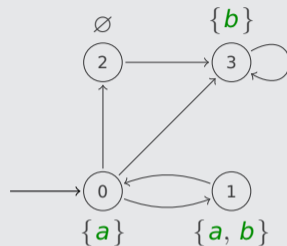


Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$

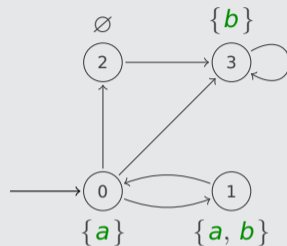


Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$



Basic Strategy

$\mathcal{M}, s \models \varphi$?

- ▶ construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- ▶ combine $A_{\neg\varphi}$ and \mathcal{M} into single automaton $A_{\neg\varphi} \times \mathcal{M}$
- ▶ determine whether there exists accepting path in $A_{\neg\varphi} \times \mathcal{M}$

Theorem

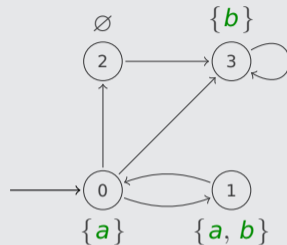
$\mathcal{M}, s \models \varphi \iff A_{\neg\varphi} \times \mathcal{M}$ has no accepting paths

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aU b$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$



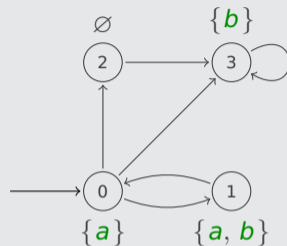
► accepting path 4 0 $\xrightarrow{\{a\}}$ 5 2 $\xrightarrow{\emptyset}$ 2 3 $\xrightarrow{\{b\}}$ 2 3 $\xrightarrow{\{b\}}$...

Example

labelled Büchi automaton $A_{\neg\varphi}$ for $\varphi = aUb$

	1	2	3	4	5
$\{a, b, \varphi\}$ 1	✓	✓	✓	✓	✓
$\{\neg a, b, \varphi\}$ 2	✓	✓	✓	✓	✓
$\{a, \neg b, \varphi\}$ 3	✓	✓	✓		
$\rightarrow \{a, \neg b, \neg\varphi\}$ 4				✓	✓
$\rightarrow \{\neg a, \neg b, \neg\varphi\}$ 5	✓	✓	✓	✓	✓

model \mathcal{M}



acceptance condition: paths cycling in state 3 are not accepting

► product automaton $A_{\neg\varphi} \times \mathcal{M}$



► accepting path $4\ 0 \xrightarrow{\{a\}} 5\ 2 \xrightarrow{\emptyset} 2\ 3 \xrightarrow{\{b\}} 2\ 3 \xrightarrow{\{b\}} \dots \implies \mathcal{M}, 0 \not\models \varphi$

Outline

1. Summary of Previous Lecture
2. Adequacy
3. Fairness
4. Intermezzo
5. LTL Model Checking Algorithm
- 6. Further Reading**
7. Exam

Huth and Ryan

- ▶ Section 3.2.5
- ▶ Section 3.4.5
- ▶ Section 3.6.2
- ▶ Section 3.6.3

Huth and Ryan

- ▶ Section 3.2.5
- ▶ Section 3.4.5
- ▶ Section 3.6.2
- ▶ Section 3.6.3

Baier and Katoen

- ▶ Section 5.2 of **Principles of Model Checking** (MIT Press 2008)

Important Concepts

- ▶ A_C
- ▶ A_φ
- ▶ adequacy
- ▶ closure
- ▶ E_C
- ▶ elementary set
- ▶ fair path
- ▶ fairness constraints
- ▶ labelled Büchi automaton
- ▶ trace

Important Concepts

- ▶ A_C
- ▶ A_φ
- ▶ adequacy
- ▶ closure
- ▶ E_C
- ▶ elementary set
- ▶ fair path
- ▶ fairness constraints
- ▶ labelled Büchi automaton
- ▶ trace

homework for June 11

Important Concepts

- ▶ A_C
- ▶ A_φ
- ▶ adequacy
- ▶ closure
- ▶ E_C
- ▶ elementary set
- ▶ fair path
- ▶ fairness constraints
- ▶ labelled Büchi automaton
- ▶ trace

homework for June 11

next week (June 8): online evaluation in presence

Important Concepts

- ▶ A_C
- ▶ A_φ
- ▶ adequacy
- ▶ closure
- ▶ E_C
- ▶ elementary set
- ▶ fair path
- ▶ fairness constraints
- ▶ labelled Büchi automaton
- ▶ trace

homework for June 11

next week (June 8): online evaluation in presence \implies bring device

Outline

1. Summary of Previous Lecture
2. Adequacy
3. Fairness
4. Intermezzo
5. LTL Model Checking Algorithm
6. Further Reading
- 7. Exam**

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

Preparation

- ▶ study previous exams

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

Preparation

- ▶ study previous exams
- ▶ review homework exercises and solutions

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

Preparation

- ▶ study previous exams
- ▶ review homework exercises and solutions
- ▶ study slides

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

Preparation

- ▶ study previous exams
- ▶ review homework exercises and solutions
- ▶ study slides
- ▶ visit Tutorium (Wednesday, 10:15–11:00, RR 26)

First Exam on June 29

- ▶ registration in LFU:online is required before 23:59 on June 15
- ▶ strict deadline: late email requests will be ignored
- ▶ de-registration is possible until 10:00 on June 26
- ▶ closed book
- ▶ second exam on September 15, third exam on February 4, 2027

Preparation

- ▶ study previous exams
- ▶ review homework exercises and solutions
- ▶ study slides
- ▶ visit Tutorium (Wednesday, 10:15–11:00, RR 26)
special **exam preparation session** on June 24, 10:15–11:45, SR 1