Overview

Introduction

Modelling parallel systems

Linear Time Properties

Regular Properties

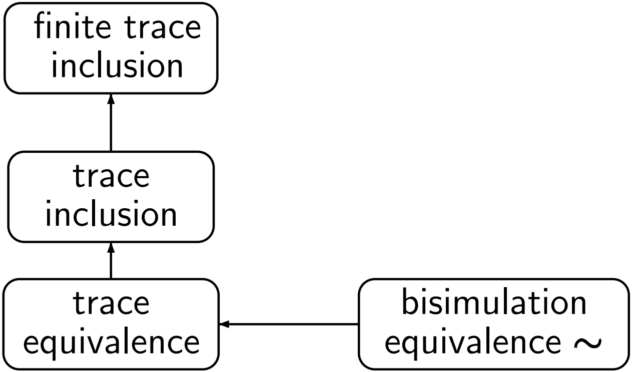
Linear Temporal Logic (LTL)

Computation-Tree Logic

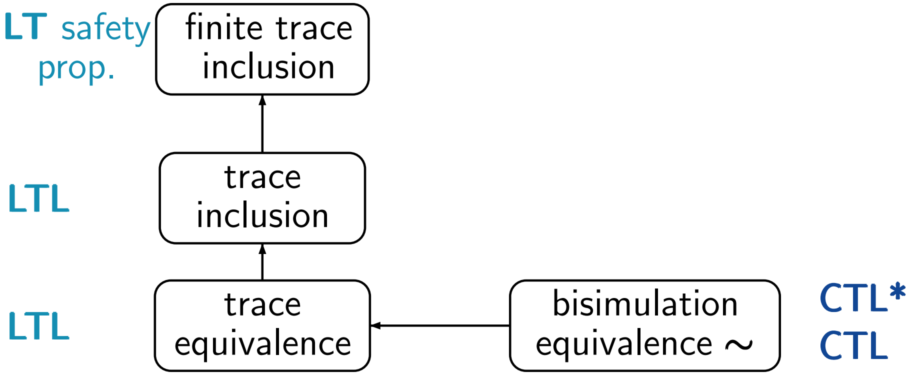
Equivalences and Abstraction

- bisimulation
- CTL, CTL*-equivalence
- computing the bisimulation quotient
- abstraction stutter steps
- simulation relations ⟵

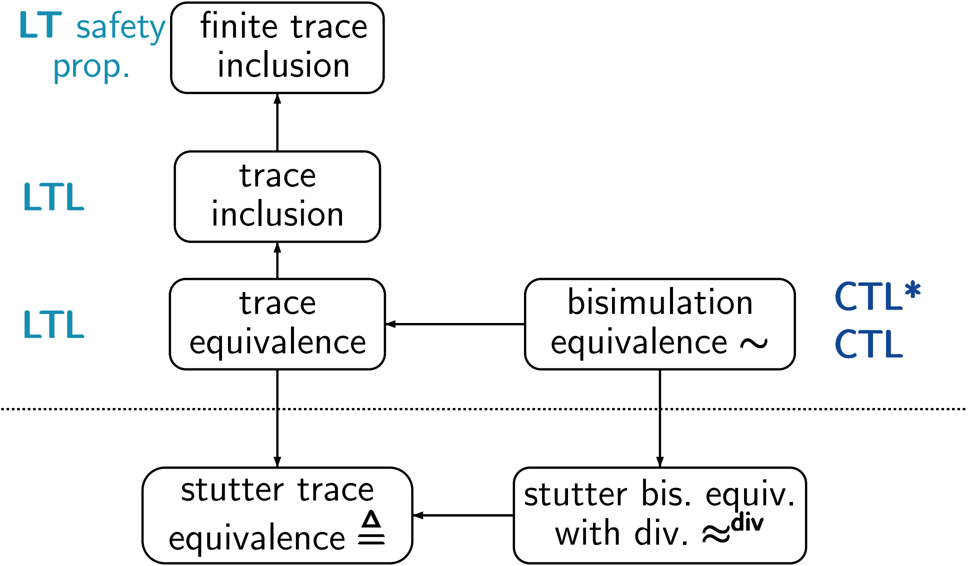
Equivalences and preorders



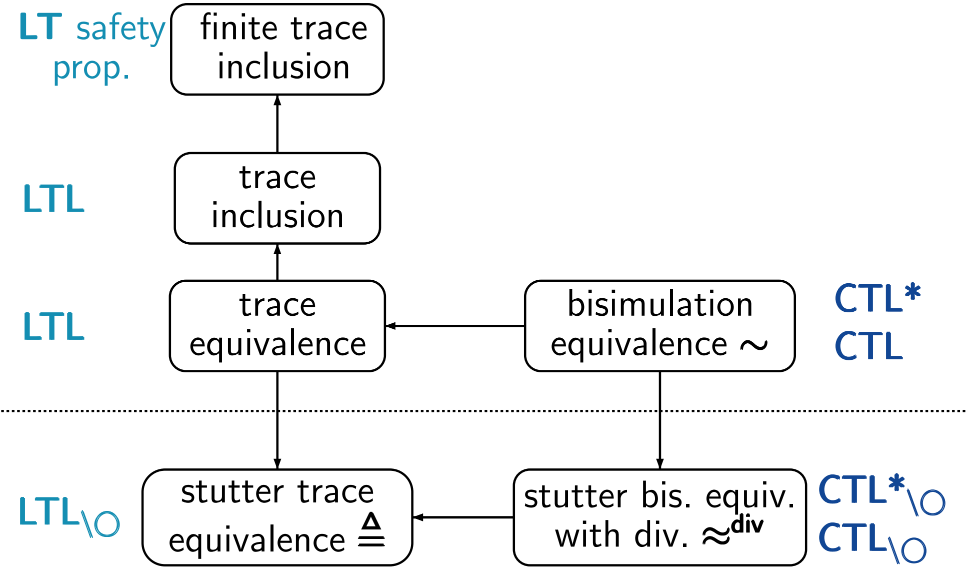
Logical characterizations



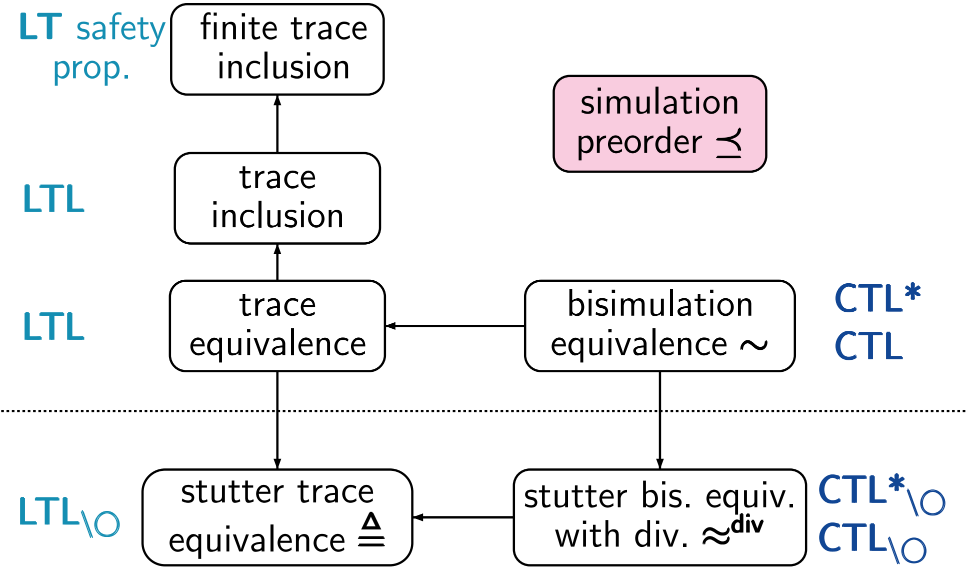
Logical characterizations



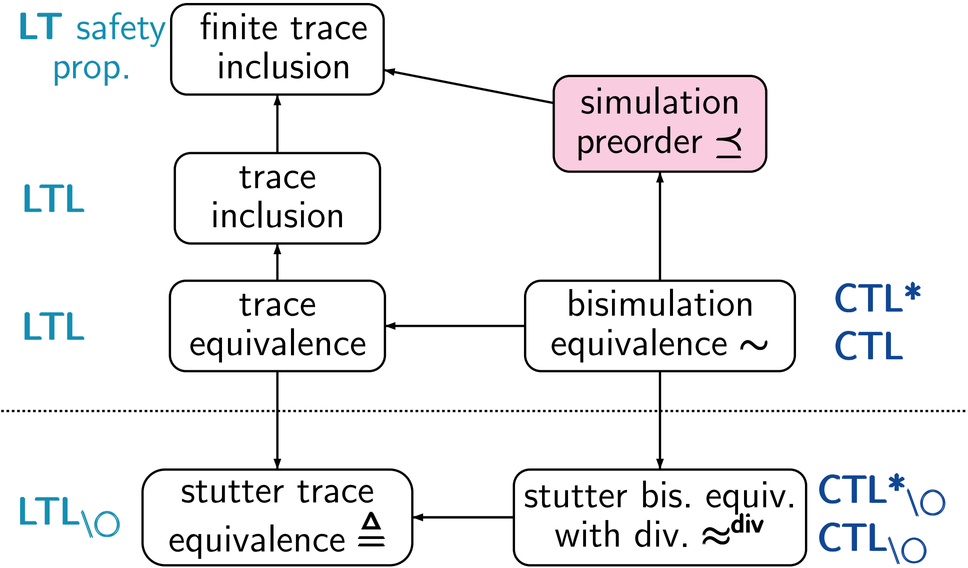
Logical characterizations



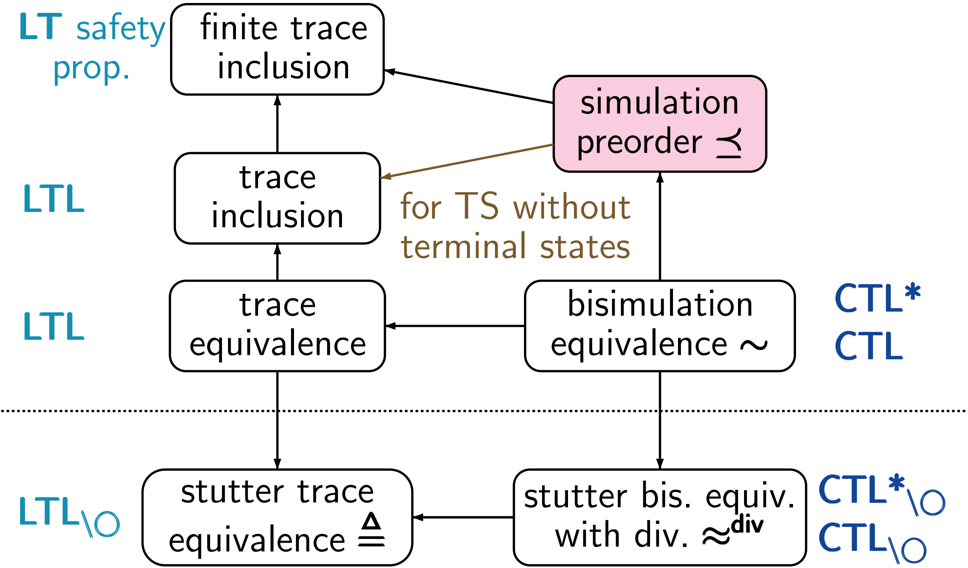
Logical characterizations



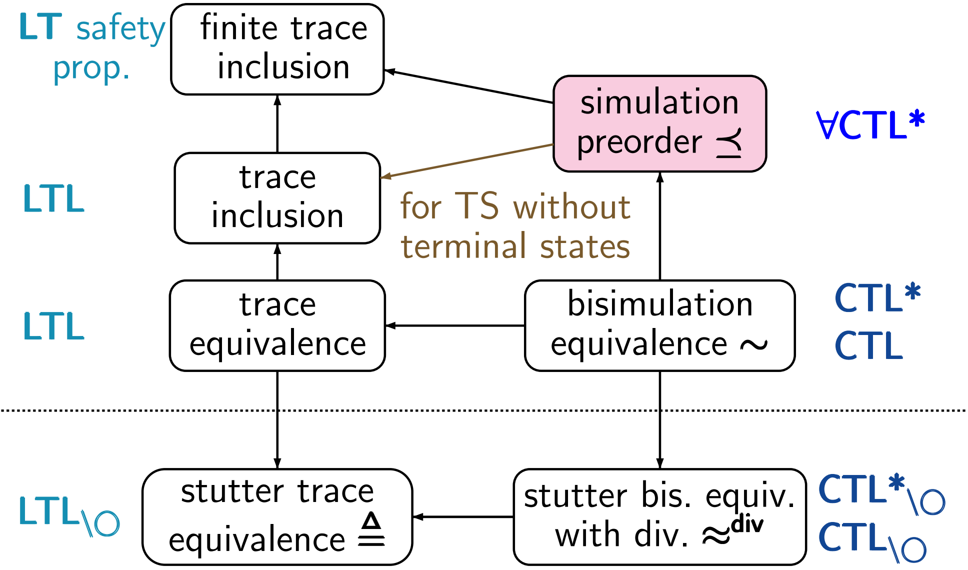
Logical characterizations



Logical characterizations



Logical characterizations



Logical characterization

for bisimulation equivalence $\sim_{\mathcal{T}}$:

$s_1 \sim_{\mathcal{T}} s_2$	iff	s_1, s_2 satisfy the same **CTL*** formulas
	iff	s_1, s_2 satisfy the same **CTL** formulas

Logical characterization

for bisimulation equivalence $\sim_{\mathcal{T}}$:

$s_1 \sim_{\mathcal{T}} s_2$	iff	s_1, s_2 satisfy the same **CTL*** formulas
	iff	s_1, s_2 satisfy the same **CTL** formulas

for the simulation preorder $\preceq_{\mathcal{T}}$:

Logical characterization

for bisimulation equivalence $\sim_{\mathcal{T}}$:

$s_1 \sim_{\mathcal{T}} s_2$	iff	s_1, s_2 satisfy the same **CTL*** formulas
	iff	s_1, s_2 satisfy the same **CTL** formulas

for the simulation preorder $\preceq_{\mathcal{T}}$:

by a sublogic $\mathbb{L}$ of **CTL*** that subsumes **LTL**

Logical characterization

for bisimulation equivalence $\sim_{\mathcal{T}}$:

$s_1 \sim_{\mathcal{T}} s_2$	iff	s_1, s_2 satisfy the same **CTL*** formulas
	iff	s_1, s_2 satisfy the same **CTL** formulas

for the simulation preorder $\preceq_{\mathcal{T}}$:

by a sublogic $\mathbb{L}$ of **CTL*** that subsumes **LTL**

$s_1 \preceq_{\mathcal{T}} s_2$ iff for all formulas $\Phi \in \mathbb{L}$:

$s_2 \models \Phi$ implies $s_1 \models \Phi$

for bisimulation equivalence $\sim_{\mathcal{T}}$:

$s_1 \ \sim_{\mathcal{T}} \ s_2$ iff s_1, s_2 satisfy the same **CTL*** formulas
iff s_1, s_2 satisfy the same **CTL** formulas

for the simulation preorder $\preceq_{\mathcal{T}}$:

by a sublogic $\mathbb{L}$ of **CTL*** that subsumes **LTL**

$s_1 \ \preceq_{\mathcal{T}} \ s_2$ iff for all formulas $\Phi \in \mathbb{L}$:
$s_2 \models \Phi$ implies $s_1 \models \Phi$

observation: $\mathbb{L}$ <u>cannot</u> be closed under negation

The universal fragment ∀CTL* of CTL*

CTL* formulas in positive normal form, without ∃

Syntax of ∀CTL*

∀CTL* state formulas:

$$\Phi \quad ::= \quad \mathit{true} \;\mid\; \mathit{false} \;\mid\; a \;\mid\; \neg a \;\mid$$
$$\Phi_1 \wedge \Phi_2 \;\mid\; \Phi_1 \vee \Phi_2 \;\mid\; \forall \varphi$$

∀CTL* path formulas:

$$\varphi \quad ::= \quad \Phi \;\mid\; \varphi_1 \wedge \varphi_2 \;\mid\; \varphi_1 \vee \varphi_2 \;\mid\; \bigcirc \varphi \;\mid$$
$$\varphi_1 \,\mathsf{U}\, \varphi_2 \;\mid\; \varphi_1 \,\mathsf{W}\, \varphi_2$$

Syntax of ∀CTL*

∀CTL* state formulas:

$$\Phi \;::=\; \mathit{true} \;\mid\; \mathit{false} \;\mid\; a \;\mid\; \neg a \;\mid\; \Phi_1 \wedge \Phi_2 \;\mid\; \Phi_1 \vee \Phi_2 \;\mid\; \forall \varphi$$

∀CTL* path formulas:

$$\varphi \;::=\; \Phi \;\mid\; \varphi_1 \wedge \varphi_2 \;\mid\; \varphi_1 \vee \varphi_2 \;\mid\; \bigcirc \varphi \;\mid\; \varphi_1 \,\mathsf{U}\, \varphi_2 \;\mid\; \varphi_1 \,\mathsf{W}\, \varphi_2$$

eventually: $\Diamond \varphi \stackrel{\text{def}}{=} \mathit{true}\,\mathsf{U}\,\varphi$

Syntax of $\forall$CTL*

$\forall$CTL* state formulas:

$$\Phi ::= \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \forall \varphi$$

$\forall$CTL* path formulas:

$$\varphi ::= \Phi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc \varphi \mid \varphi_1 \mathsf{U} \varphi_2 \mid \varphi_1 \mathsf{W} \varphi_2$$

eventually: $\Diamond \varphi \stackrel{\text{def}}{=} \mathit{true} \, \mathsf{U} \, \varphi$

always: $\Box \varphi \stackrel{\text{def}}{=} \varphi \, \mathsf{W} \, \mathit{false}$

Embedding of LTL in ∀CTL*

∀CTL* state formulas:

$$\Phi \;::=\; \mathit{true} \;\mid\; \mathit{false} \;\mid\; a \;\mid\; \neg a \;\mid\; \Phi_1 \wedge \Phi_2 \;\mid\; \Phi_1 \vee \Phi_2 \;\mid\; \forall \varphi$$

∀CTL* path formulas:

$$\varphi \;::=\; \Phi \;\mid\; \varphi_1 \wedge \varphi_2 \;\mid\; \varphi_1 \vee \varphi_2 \;\mid\; \bigcirc \varphi \;\mid\; \varphi_1 \,\mathsf{U}\, \varphi_2 \;\mid\; \varphi_1 \,\mathsf{W}\, \varphi_2$$

for all **LTL** formulas φ in **PNF**:

Embedding of LTL in ∀CTL*

∀CTL* state formulas:

$$\Phi \ ::= \ \mathit{true} \ | \ \mathit{false} \ | \ a \ | \ \neg a \ | \ \Phi_1 \wedge \Phi_2 \ | \ \Phi_1 \vee \Phi_2 \ | \ \forall \varphi$$

∀CTL* path formulas:

$$\varphi \ ::= \ \Phi \ | \ \varphi_1 \wedge \varphi_2 \ | \ \varphi_1 \vee \varphi_2 \ | \ \bigcirc \varphi \ | \ \varphi_1 \, \mathsf{U} \, \varphi_2 \ | \ \varphi_1 \, \mathsf{W} \, \varphi_2$$

for all **LTL** formulas φ in **PNF**:

$$s \models_{\mathsf{LTL}} \varphi \quad \text{iff} \quad s \models_{\forall\mathsf{CTL}^*} \forall \varphi$$

Embedding of LTL in ∀CTL*

∀CTL* state formulas:

$$\Phi \; ::= \; \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \forall \varphi$$

∀CTL* path formulas:

$$\varphi \; ::= \; \Phi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc \varphi \mid \varphi_1 \, \mathsf{U} \, \varphi_2 \mid \varphi_1 \, \mathsf{W} \, \varphi_2$$

for all **LTL** formulas φ in **PNF**:

$$s \models_{\mathsf{LTL}} \varphi \quad \text{iff} \quad s \models_{\forall\mathsf{CTL}^*} \forall \varphi$$

but $\forall\Diamond\forall\Box a$ cannot be expressed in **LTL**

The universal fragments of CTL* and CTL

syntax of $\forall$**CTL***:

$$\Phi \;::=\; \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \forall\varphi$$

$$\varphi \;::=\; \Phi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \,\mathsf{U}\, \varphi_2 \mid \varphi_1 \,\mathsf{W}\, \varphi_2$$

$\forall$**CTL**: sublogic of $\forall$**CTL***

- no Boolean operators for paths formulas
- the arguments of the temporal modalities $\bigcirc$, **U** and **W** are state formulas

syntax of $\forall$**CTL***:

$$\Phi ::= \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \forall \varphi$$

$$\varphi ::= \Phi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc \varphi \mid \varphi_1 \, \mathsf{U} \, \varphi_2 \mid \varphi_1 \, \mathsf{W} \, \varphi_2$$

$\forall$**CTL**: sublogic of $\forall$**CTL***

syntax of $\forall$**CTL**:

$$\Phi ::= \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \forall \bigcirc \Phi \mid \forall(\Phi_1 \, \mathsf{U} \, \Phi_2) \mid \forall(\Phi_1 \, \mathsf{W} \, \Phi_2)$$

Logical characterization of simulation

Logical characterization of simulation

Let $\mathcal{T}$ be a finite TS without terminal states. Then, for all states s_1 and s_2 in $\mathcal{T}$, the following statements are equivalent:

Logical characterization of simulation

Let $\mathcal{T}$ be a finite TS without terminal states. Then, for all states s_1 and s_2 in $\mathcal{T}$, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

Logical characterization of simulation

Let $\mathcal{T}$ be a finite TS without terminal states. Then, for all states s_1 and s_2 in $\mathcal{T}$, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$**CTL** state formulas Φ:

$$\text{if } s_2 \models \Phi \text{ then } s_1 \models \Phi$$

Logical characterization of simulation

Let $\mathcal{T}$ be a finite TS without terminal states. Then, for all states s_1 and s_2 in $\mathcal{T}$, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$**CTL** state formulas Φ:

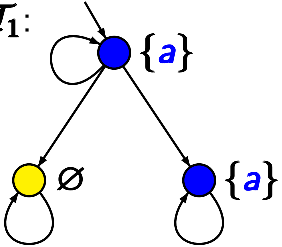
if $s_2 \models \Phi$ then $s_1 \models \Phi$

(3) for all $\forall$**CTL*** state formulas Φ:

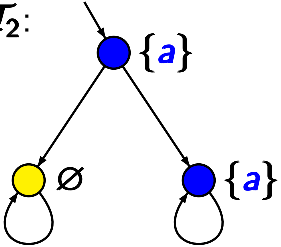
if $s_2 \models \Phi$ then $s_1 \models \Phi$

∀CTL and simulation

$\mathcal{T}_1$:

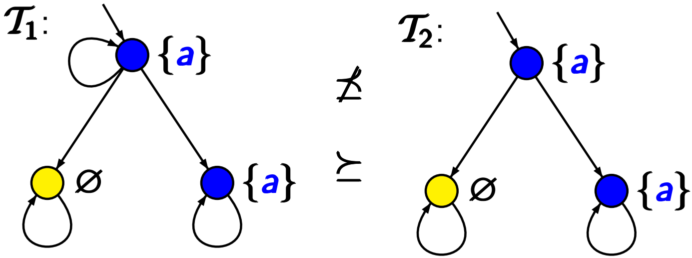


$\mathcal{T}_2$:



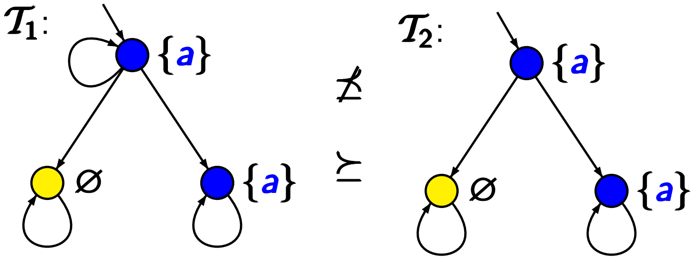
$AP = \{a\}$

∀CTL and simulation



$AP = \{a\}$

∀CTL and simulation

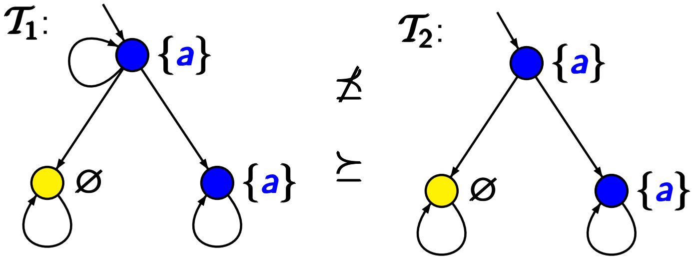


$AP = \{a\}$

e.g., $\mathcal{T}_1 \not\models \forall\bigcirc(\forall\bigcirc\neg a \;\vee\; \forall\bigcirc a)$

$\mathcal{T}_2 \models \forall\bigcirc(\forall\bigcirc\neg a \;\vee\; \forall\bigcirc a)$

∀CTL and simulation



$AP = \{a\}$

e.g., $\mathcal{T}_1 \not\models \forall\bigcirc(\forall\bigcirc\neg a \ \vee\ \forall\bigcirc a)$

$\mathcal{T}_2 \models \forall\bigcirc(\forall\bigcirc\neg a \ \vee\ \forall\bigcirc a)$

$\mathcal{T}_1 \not\models \forall\Diamond(\forall\Box\neg a \ \vee\ \forall\Box a)$

$\mathcal{T}_2 \models \forall\Diamond(\forall\Box\neg a \ \vee\ \forall\Box a)$

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all **∀CTL** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all **∀CTL*** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1)	$s_1 \preceq_{\mathcal{T}} s_2$
(2)	for all **$\forall$CTL** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$
(3)	for all **$\forall$CTL*** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) $\Longrightarrow$ (2): obvious as **$\forall$CTL** is a sublogic of **$\forall$CTL***

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$**CTL** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all $\forall$**CTL*** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) $\Longrightarrow$ (2): obvious as $\forall$**CTL** is a sublogic of $\forall$**CTL***

(1) $\Longrightarrow$ (3): holds for arbitrary (possibly infinite) TS without terminal states

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$CTL formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all $\forall$CTL* formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) $\Longrightarrow$ (2): obvious as $\forall$CTL is a sublogic of $\forall$CTL*

(1) $\Longrightarrow$ (3): holds for arbitrary (possibly infinite) TS without terminal states

↑

proof by structural induction

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$**CTL** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all $\forall$**CTL*** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(1) $\Longrightarrow$ (3): show by structural induction:

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$CTL formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all $\forall$CTL* formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(1) $\Longrightarrow$ (3): show by structural induction:

(i) for all $\forall$CTL* state formulas Φ and states s_1, s_2:
if $s_1 \preceq_{\mathcal{T}} s_2$ and $s_2 \models \Phi$ then $s_1 \models \Phi$

∀CTL/∀CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all ∀CTL formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all ∀CTL* formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(1) $\Longrightarrow$ (3): show by structural induction:

(i) for all ∀CTL* state formulas Φ and states s_1, s_2:
if $s_1 \preceq_{\mathcal{T}} s_2$ and $s_2 \models \Phi$ then $s_1 \models \Phi$

(ii) for all ∀CTL* path formulas φ and paths π_1, π_2:
if $\pi_1 \preceq_{\mathcal{T}} \pi_2$ and $\pi_2 \models \varphi$ then $\pi_1 \models \varphi$

$\forall$CTL/$\forall$CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$
(2) for all $\forall$**CTL** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$
(3) for all $\forall$**CTL*** formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(2) $\Longrightarrow$ (1):

∀CTL/∀CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all ∀CTL formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all ∀CTL* formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(2) $\Longrightarrow$ (1): show that

$$\mathcal{R} = \{ (s_1, s_2) : \text{for all } \forall\text{CTL formulas } \Phi: s_2 \models \Phi \text{ implies } s_1 \models \Phi \}$$

is a simulation.

∀CTL/∀CTL* and the simulation preorder

For finite TS without terminal states, the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all ∀CTL formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(3) for all ∀CTL* formulas Φ: $s_2 \models \Phi$ implies $s_1 \models \Phi$

(2) $\Longrightarrow$ (1): show that for finite TS:

$$\mathcal{R} = \{ (s_1, s_2) : \text{for all } \forall\text{CTL formulas } \Phi: s_2 \models \Phi \text{ implies } s_1 \models \Phi \}$$

is a simulation.

The existential fragment ∃CTL* of CTL*

The existential fragment ∃CTL* of CTL*

dual to ∀**CTL***, i.e., **CTL*** formulas in **PNF**, without ∀

The existential fragment ∃CTL* of CTL*

∃CTL* (state) formulas:

$$\Psi \quad ::= \quad \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Psi_1 \wedge \Psi_2 \mid \Psi_1 \vee \Psi_2 \mid \exists \varphi$$

∃CTL* path formulas:

$$\varphi \quad ::= \quad \Psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc \varphi \mid \varphi_1 \, \mathsf{U} \, \varphi_2 \mid \varphi_1 \, \mathsf{W} \, \varphi_2$$

The existential fragment ∃CTL* of CTL*

∃CTL* (state) formulas:

$$\Psi ::= \textit{true} \mid \textit{false} \mid a \mid \neg a \mid \Psi_1 \wedge \Psi_2 \mid \Psi_1 \vee \Psi_2 \mid \exists\varphi$$

∃CTL* path formulas:

$$\varphi ::= \Psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \, \mathbf{U} \, \varphi_2 \mid \varphi_1 \, \mathbf{W} \, \varphi_2$$

analogous: **∃CTL**

Duality of $\forall$CTL* and $\exists$CTL*

$\exists$**CTL*** (state) formulas:

$$\Psi \quad ::= \quad \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Psi_1 \wedge \Psi_2 \mid \Psi_1 \vee \Psi_2 \mid \exists\varphi$$

$\exists$**CTL*** path formulas:

$$\varphi \quad ::= \quad \Psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \,\mathbf{U}\, \varphi_2 \mid \varphi_1 \,\mathbf{W}\, \varphi_2$$

analogous: $\exists$**CTL**

For each $\forall$**CTL*** formula Φ there is a $\exists$**CTL*** formula Ψ s.t. $\Phi \equiv \neg\Psi$

∃CTL* (state) formulas:

$$\Psi \;::=\; \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Psi_1 \wedge \Psi_2 \mid \Psi_1 \vee \Psi_2 \mid \exists \varphi$$

∃CTL* path formulas:

$$\varphi \;::=\; \Psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc \varphi \mid \varphi_1 \,\mathsf{U}\, \varphi_2 \mid \varphi_1 \,\mathsf{W}\, \varphi_2$$

analogous: **∃CTL**

For each **∀CTL*** formula Φ there is a **∃CTL*** formula Ψ s.t. $\Phi \equiv \neg\Psi$ (and vice versa)

Duality of ∀CTL* and ∃CTL*

∃CTL* (state) formulas:

$$\Psi \;::=\; \mathit{true} \mid \mathit{false} \mid a \mid \neg a \mid \Psi_1 \wedge \Psi_2 \mid \Psi_1 \vee \Psi_2 \mid \exists\varphi$$

∃CTL* path formulas:

$$\varphi \;::=\; \Psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \,\mathsf{U}\, \varphi_2 \mid \varphi_1 \,\mathsf{W}\, \varphi_2$$

analogous: ∃CTL

For each ∀CTL* formula Φ there is a ∃CTL* formula Ψ s.t. $\Phi \equiv \neg\Psi$ (and vice versa)

For each ∀CTL formula Φ there is a ∃CTL formula Ψ s.t. $\Phi \equiv \neg\Psi$ (and vice versa)

Logical characterization of simulation

GRM5.5-20A

If s_1 and s_2 are states in a finite TS then the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2) for all $\forall$CTL formulas Φ:

if $s_2 \models \Phi$ then $s_1 \models \Phi$

(3) for all $\forall$CTL* formulas Φ:

if $s_2 \models \Phi$ then $s_1 \models \Phi$

Logical characterization of simulation

If s_1 and s_2 are states in a finite TS then the following statements are equivalent:

(1) $s_1 \preceq_{\mathcal{T}} s_2$

(2∀) for all ∀**CTL** formulas Φ:

if $s_2 \models \Phi$ then $s_1 \models \Phi$

(3∀) for all ∀**CTL*** formulas Φ:

if $s_2 \models \Phi$ then $s_1 \models \Phi$

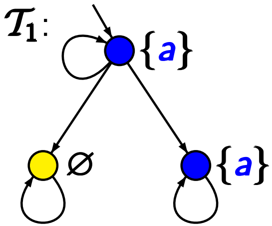
(2∃) for all ∃**CTL** formulas Ψ:

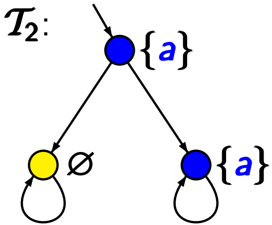
if $s_1 \models \Psi$ then $s_2 \models \Psi$

(3∃) for all ∃**CTL** formulas Ψ:

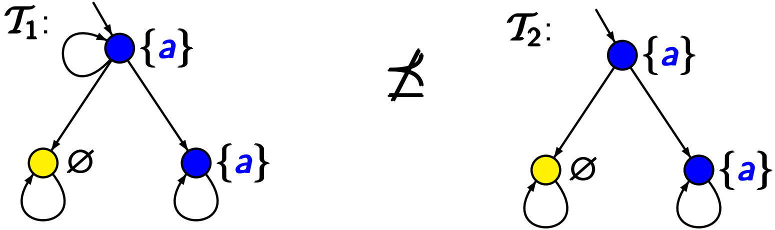
if $s_1 \models \Psi$ then $s_2 \models \Psi$

Example: ∀CTL/∃CTL and simulation

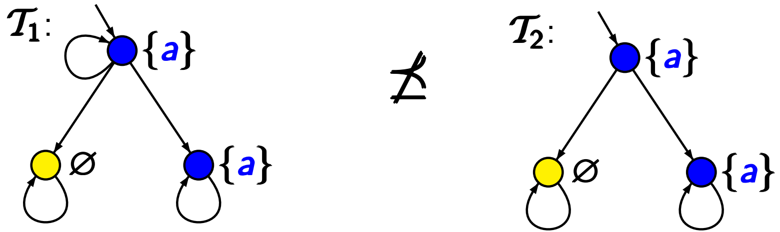




Example: ∀CTL/∃CTL and simulation



Example: ∀CTL/∃CTL and simulation

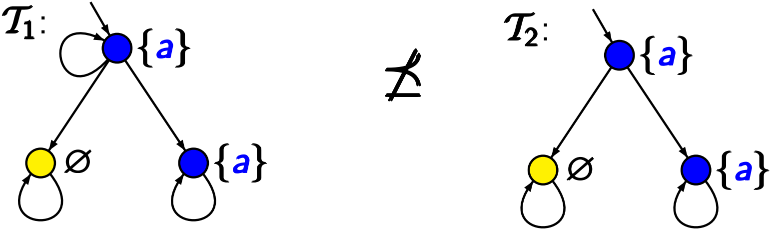


$\mathcal{T}_1 \not\models \forall\bigcirc(\forall\bigcirc\neg a \vee \forall\bigcirc a)$

$\mathcal{T}_2 \models \forall\bigcirc(\forall\bigcirc\neg a \vee \forall\bigcirc a)$

∀CTL formula

Example: ∀CTL/∃CTL and simulation



$\mathcal{T}_1 \not\models \forall\bigcirc(\forall\bigcirc\neg a \vee \forall\bigcirc a)$

$\mathcal{T}_2 \models \forall\bigcirc(\forall\bigcirc\neg a \vee \forall\bigcirc a)$

∀CTL formula

$\mathcal{T}_1 \models \exists\bigcirc(\exists\bigcirc\neg a \wedge \exists\bigcirc a)$

$\mathcal{T}_2 \not\models \exists\bigcirc(\exists\bigcirc\neg a \wedge \exists\bigcirc a)$

∃CTL formula

Characterizations of simulation equivalence

Characterizations of simulation equivalence

for finite TS without terminal states:

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∀CTL*** formulas

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∀CTL*** formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∀CTL** formulas

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∀CTL*** formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same ∀CTL formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∃CTL*** formulas

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∀CTL*** formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same ∀CTL formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same **∃CTL*** formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same ∃CTL formulas

Characterizations of simulation equivalence

for finite TS without terminal states:

$\mathcal{T}_1 \simeq \mathcal{T}_2$ iff $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same $\forall$CTL* formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same $\forall$CTL formulas

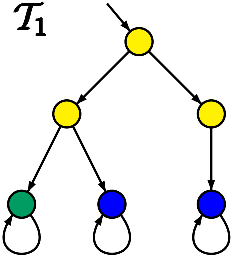
iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same $\exists$CTL* formulas

iff $\mathcal{T}_1, \mathcal{T}_2$ satisfy the same $\exists$CTL formulas

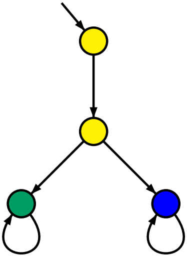
... even holds for $\forall\text{CTL*}_{\setminus \text{U,W}}$, $\forall\text{CTL}_{\setminus \text{U,W}}$, $\exists\text{CTL*}_{\setminus \text{U,W}}$, $\exists\text{CTL}_{\setminus \text{U,W}}$

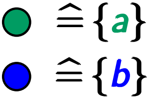
Simulation equivalence

$\mathcal{T}_1$

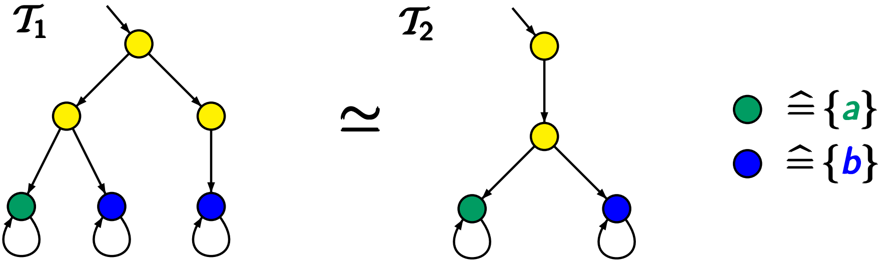


$\mathcal{T}_2$

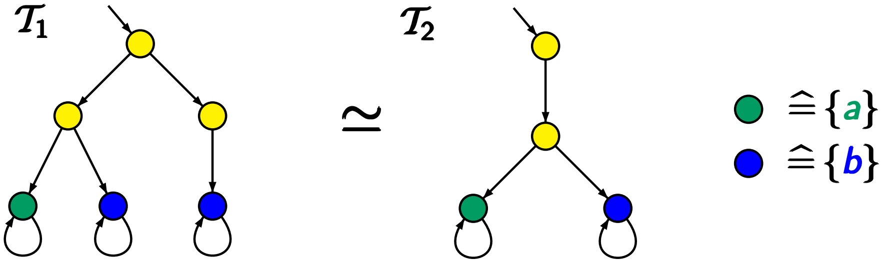




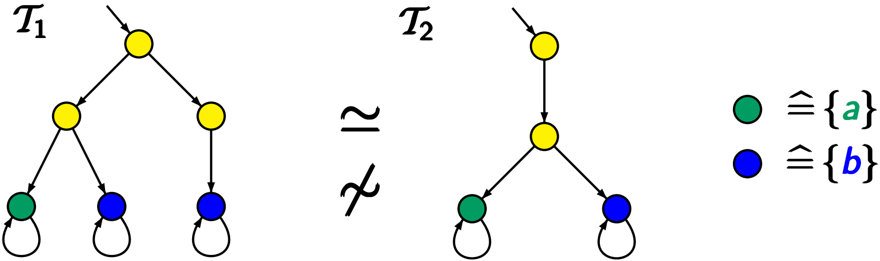
Simulation equivalence



Simulation equivalence

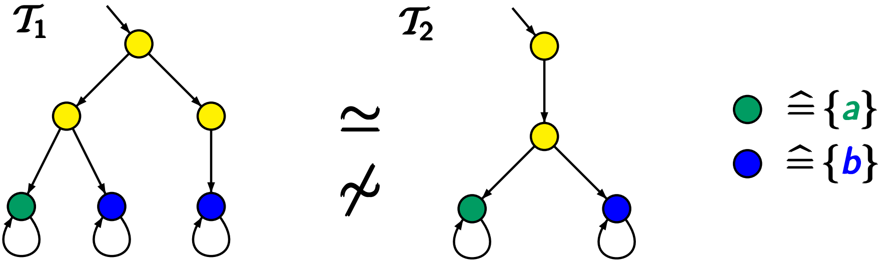


$\mathcal{T}_1$, $\mathcal{T}_2$ cannot be distinguished by the temporal logics
∀CTL, **∀CTL***, **∃CTL**, or **∃CTL***,



$\mathcal{T}_1$, $\mathcal{T}_2$ cannot be distinguished by the temporal logics
∀CTL, **∀CTL***, **∃CTL**, or **∃CTL***,

Simulation equivalence



$\mathcal{T}_1$, $\mathcal{T}_2$ cannot be distinguished by the temporal logics **∀CTL**, **∀CTL***, **∃CTL**, or **∃CTL***,

but by **CTL**:

$$\mathcal{T}_1 \not\models \forall\bigcirc(\exists\bigcirc a \wedge \exists\bigcirc b)$$

$$\mathcal{T}_2 \models \forall\bigcirc(\exists\bigcirc a \wedge \exists\bigcirc b)$$

Correct or wrong?

In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

Correct or wrong?

In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

correct

Correct or wrong?

In finite TS without terminal states:

If s_1, s_2 satisfy the same ∃CTL formulas
then s_1, s_2 satisfy the same LTL formulas

correct

$$\begin{aligned} & \exists\text{CTL equivalence} \\ = \; & \text{simulation equivalence} \\ = \; & \forall\text{CTL* equivalence} \end{aligned}$$

Correct or wrong?

In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

correct

$\exists$CTL equivalence
= simulation equivalence
= $\forall$CTL* equivalence

and LTL is a sublogic of $\forall$CTL*

Correct or wrong?

In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

correct

If s_1, s_2 satisfy the same LTL formulas
then s_1, s_2 satisfy the same $\forall$CTL formulas

In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

correct

If s_1, s_2 satisfy the same LTL formulas
then s_1, s_2 satisfy the same $\forall$CTL formulas

wrong

Correct or wrong?

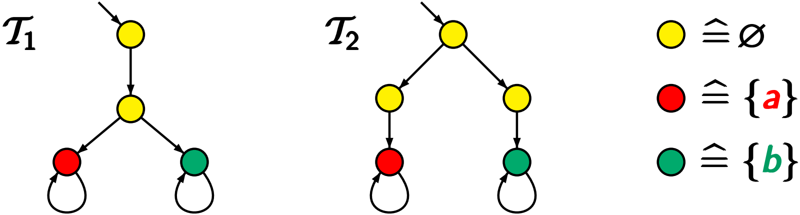
In finite TS without terminal states:

If s_1, s_2 satisfy the same $\exists$CTL formulas
then s_1, s_2 satisfy the same LTL formulas

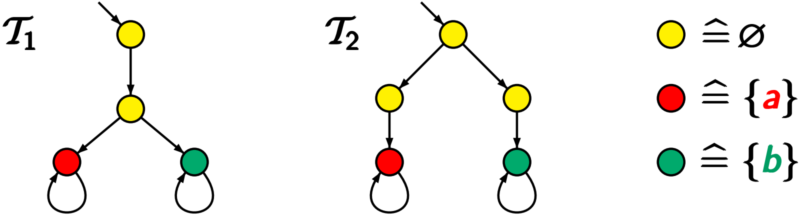
correct

If s_1, s_2 satisfy the same LTL formulas
then s_1, s_2 satisfy the same $\forall$CTL formulas

wrong, as trace equivalence does not imply simulation equivalence



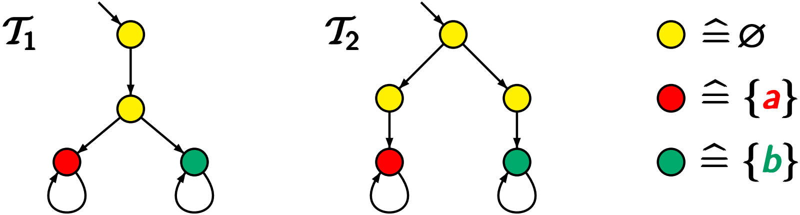
Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?



Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes

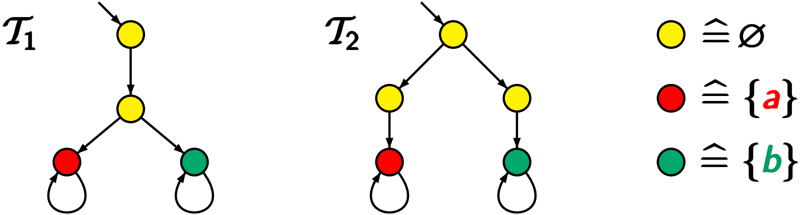
Does there exist ...?



Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes, as $\mathcal{T}_1 \not\preceq \mathcal{T}_2$

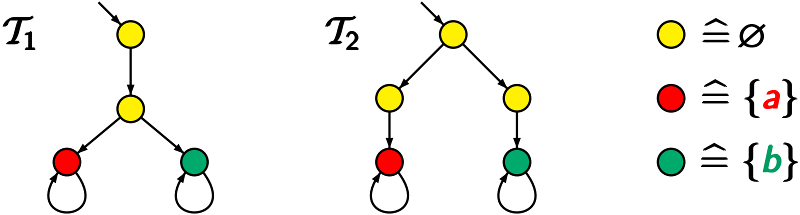
Does there exist ...?



Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes, as $\mathcal{T}_1 \not\preceq \mathcal{T}_2$, e.g., $\Phi = \exists\bigcirc(\exists\bigcirc a \ \wedge \ \exists\bigcirc b)$

Does there exist ...?

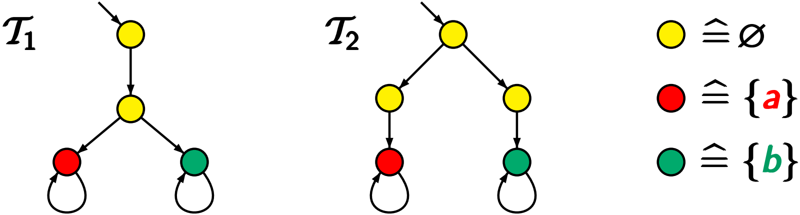


Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes, as $\mathcal{T}_1 \not\preceq \mathcal{T}_2$, e.g., $\Phi = \exists\bigcirc(\exists\bigcirc a \wedge \exists\bigcirc b)$

Does there exist a **∀CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

Does there exist ...?



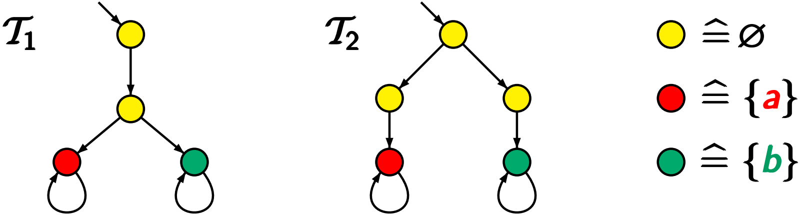
Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes, as $\mathcal{T}_1 \not\preceq \mathcal{T}_2$, e.g., $\Phi = \exists \bigcirc (\exists \bigcirc a \wedge \exists \bigcirc b)$

Does there exist a **∀CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

no

Does there exist ...?



Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

yes, as $\mathcal{T}_1 \not\preceq \mathcal{T}_2$, e.g., $\Phi = \exists\bigcirc(\exists\bigcirc a \ \wedge \ \exists\bigcirc b)$

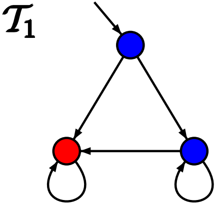
Does there exist a $\forall$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

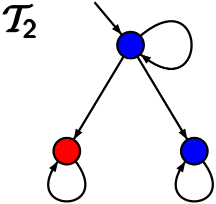
no, as $\mathcal{T}_2 \preceq \mathcal{T}_1$

Does there exist ...?

Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

Does there exist ...?



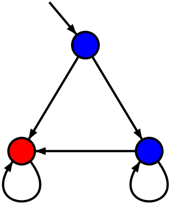


Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

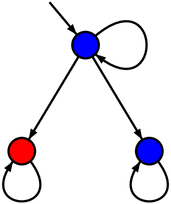
no

Does there exist ...?

$\mathcal{T}_1$

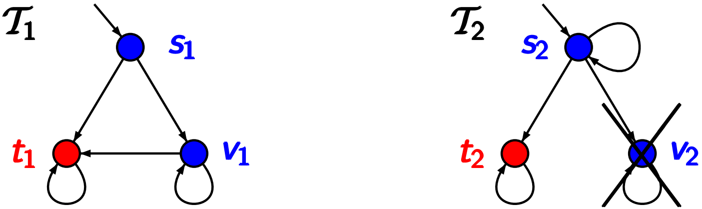


$\mathcal{T}_2$



Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

no, since $\mathcal{T}_1 \simeq \mathcal{T}_2$

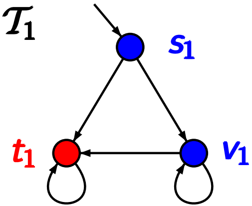


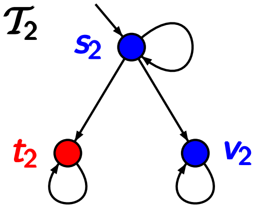
Does there exist a **∃CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

no, since $\mathcal{T}_1 \simeq \mathcal{T}_2$

simulation for $(\mathcal{T}_1, \mathcal{T}_2)$: $\{(s_1, s_2), (v_1, s_2), (t_1, t_2)\}$

Does there exist ...?





Does there exist a $\exists$**CTL** formula Φ s.t.
$\mathcal{T}_1 \models \Phi$ and $\mathcal{T}_2 \not\models \Phi$?

no, since $\mathcal{T}_1 \simeq \mathcal{T}_2$

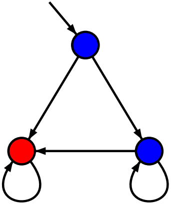
simulation for $(\mathcal{T}_1, \mathcal{T}_2)$: $\{(s_1, s_2), (v_1, s_2), (t_1, t_2)\}$

simulation for $(\mathcal{T}_2, \mathcal{T}_1)$:

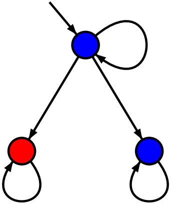
$\{(s_2, s_1), (s_2, v_1), (v_2, v_1), (t_1, t_2)\}$

Does there exist ...?

$\mathcal{T}_1$

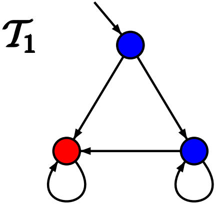


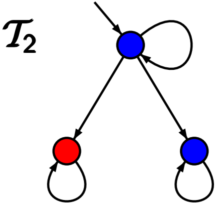
$\mathcal{T}_2$



Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

Does there exist ...?





Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

yes

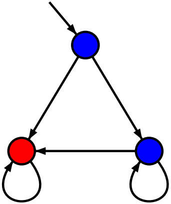
Does there exist ...?

Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

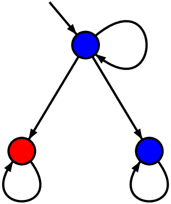
yes, as $\mathcal{T}_1 \not\sim \mathcal{T}_2$

Does there exist ...?

$\mathcal{T}_1$



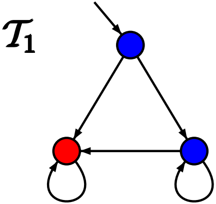
$\mathcal{T}_2$

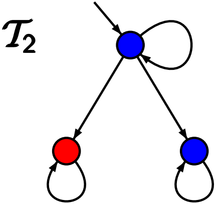


Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

yes, as $\mathcal{T}_1 \not\sim \mathcal{T}_2$, e.g., $\Phi = \exists \bigcirc \forall \Box \mathit{blue}$

Does there exist ...?





Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

yes, as $\mathcal{T}_1 \not\sim \mathcal{T}_2$, e.g., $\Phi = \exists\bigcirc\forall\Box \textit{blue}$

Does there exist a **LTL** formula φ s.t.
$\mathcal{T}_1 \not\models \varphi$ and $\mathcal{T}_2 \models \varphi$?

Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

yes, as $\mathcal{T}_1 \not\sim \mathcal{T}_2$, e.g., $\Phi = \exists\bigcirc\forall\Box \textit{blue}$

Does there exist a **LTL** formula φ s.t.
$\mathcal{T}_1 \not\models \varphi$ and $\mathcal{T}_2 \models \varphi$?

no

Does there exist ...?

Does there exist a **CTL** formula Φ s.t.
$\mathcal{T}_1 \not\models \Phi$ and $\mathcal{T}_2 \models \Phi$?

yes, as $\mathcal{T}_1 \not\sim \mathcal{T}_2$, e.g., $\Phi = \exists \bigcirc \forall \Box \textit{blue}$

Does there exist a **LTL** formula φ s.t.
$\mathcal{T}_1 \not\models \varphi$ and $\mathcal{T}_2 \models \varphi$?

no, as $\mathcal{T}_1$, $\mathcal{T}_2$ are simulation equivalent

Simulation quotient

Simulation quotient

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS.

simulation quotient $\mathcal{T}/\simeq$:

transition system that arises from $\mathcal{T}$ by collapsing all simulation equivalent states

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \ \stackrel{\text{def}}{=} \ (S/\simeq, Act', \rightarrow_{\simeq}, S'_0, AP', L')$$

Simulation quotient

Let $\mathcal{T} = (S, Act, \to, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\!\simeq \;\stackrel{\text{def}}{=}\; (S/\!\simeq, Act', \to_{\simeq}, S_0', AP', L')$$

- state space $S/\!\simeq$ $\longleftarrow$ set of all simulation equivalence classes

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \;\stackrel{\text{def}}{=}\; (S/\simeq, Act', \rightarrow_{\simeq}, S_0', AP', L')$$

- state space $S/\simeq$ ⟵ set of all simulation equivalence classes
- initial states: $S_0' = \{[s] : s \in S_0\}$

$$[s] = \{s' \in S : s \simeq_{\mathcal{T}} s'\}$$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\!\simeq \;\stackrel{\text{def}}{=}\; (S/\!\simeq, Act', \rightarrow_{\simeq}, S_0', AP', L')$$

- state space $S/\!\simeq$ ⟵ set of all simulation equivalence classes
- initial states: $S_0' = \{[s] : s \in S_0\}$
- labeling: $AP' = AP$ and $L'([s]) = L(s)$

$$[s] = \{s' \in S : s \simeq_{\mathcal{T}} s'\}$$

Simulation quotient

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \ \stackrel{\text{def}}{=} \ (S/\simeq, Act', \rightarrow_{\simeq}, S_0', AP', L')$$

- state space $S/\simeq$ ⟵ set of all simulation equivalence classes
- initial states: $S_0' = \{[s] : s \in S_0\}$
- labeling: $AP' = AP$ and $L'([s]) = L(s)$
- transition relation: $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

Simulation quotient

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\!\simeq \;\stackrel{\text{def}}{=}\; (S/\!\simeq, Act', \rightarrow_{\simeq}, S_0', AP', L')$$

- state space $S/\!\simeq$ ⟵ set of all simulation equivalence classes
- initial states: $S_0' = \{[s] : s \in S_0\}$
- labeling: $AP' = AP$ and $L'([s]) = L(s)$
- transition relation: $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

  action labels: irrelevant

Similarity of $\mathcal{T}$ and $\mathcal{T}/\simeq$

Similarity of $\mathcal{T}$ and $\mathcal{T}/\simeq$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \;=\; (S/\simeq, Act', \rightarrow_{\simeq}, S'_0, AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

$\mathcal{T}$ and $\mathcal{T}/\simeq$ are simulation equivalent, i.e.,

$$\mathcal{T} \preceq \mathcal{T}/\simeq \quad \text{and} \quad \mathcal{T}/\simeq \preceq \mathcal{T}$$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \; = \; (S/\simeq, Act', \rightarrow_{\simeq}, S'_0, AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

$\mathcal{T}$ and $\mathcal{T}/\simeq$ are simulation equivalent, i.e.,
$\mathcal{T} \preceq \mathcal{T}/\simeq$ and $\mathcal{T}/\simeq \; \preceq \mathcal{T}$

Proof. provide simulations for $(\mathcal{T}, \mathcal{T}/\simeq)$ and $(\mathcal{T}/\simeq, \mathcal{T})$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \;=\; (S/\simeq, Act', \rightarrow_{\simeq}, S_0', AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

$\mathcal{T}$ and $\mathcal{T}/\simeq$ are simulation equivalent, i.e.,
$\mathcal{T} \preceq \mathcal{T}/\simeq$ and $\mathcal{T}/\simeq \preceq \mathcal{T}$

Proof. provide simulations for $(\mathcal{T}, \mathcal{T}/\simeq)$ and $(\mathcal{T}/\simeq, \mathcal{T})$

simulation for $(\mathcal{T}, \mathcal{T}/\simeq)$: $\{(s, [s]) : s \in S\}$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\!\simeq \;=\; (S/\!\simeq, Act', \rightarrow_{\simeq}, S_0', AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

$\mathcal{T}$ and $\mathcal{T}/\!\simeq$ are simulation equivalent, i.e.,

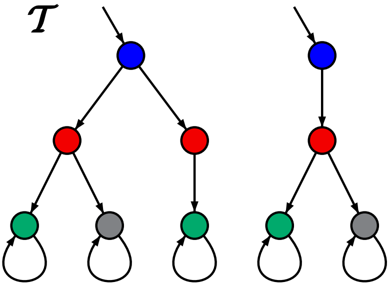
$$\mathcal{T} \preceq \mathcal{T}/\!\simeq \quad \text{and} \quad \mathcal{T}/\!\simeq \;\preceq \mathcal{T}$$

Proof. provide simulations for $(\mathcal{T}, \mathcal{T}/\!\simeq)$ and $(\mathcal{T}/\!\simeq, \mathcal{T})$

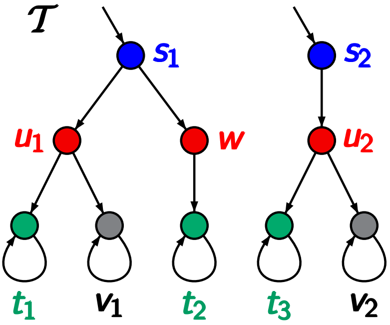
simulation for $(\mathcal{T}, \mathcal{T}/\!\simeq)$: $\{(s, [s]) : s \in S\}$

simulation for $(\mathcal{T}/\!\simeq, \mathcal{T})$: **?**

Example: simulation quotient



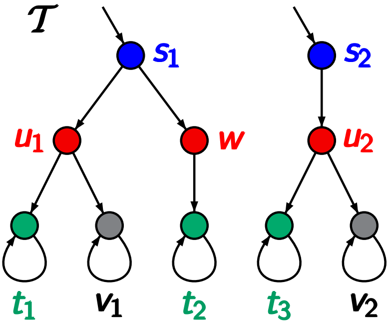
Example: simulation quotient



t_1, t_2, t_3 are simulation equivalent

v_1, v_2 are simulation equivalent

Example: simulation quotient

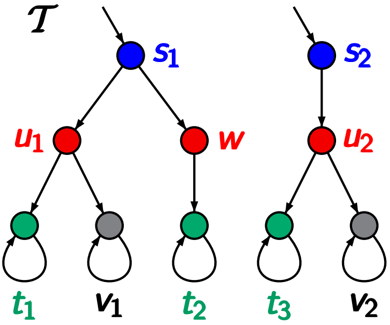


t_1, t_2, t_3 are simulation equivalent

v_1, v_2 are simulation equivalent

$u_1 \simeq u_2$,

Example: simulation quotient

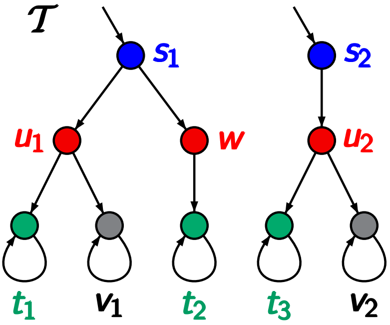


t_1, t_2, t_3 are simulation equivalent

v_1, v_2 are simulation equivalent

$u_1 \simeq u_2$, $\quad w \preceq u_1, u_2$, but $w \not\simeq u_1, u_2$

Example: simulation quotient



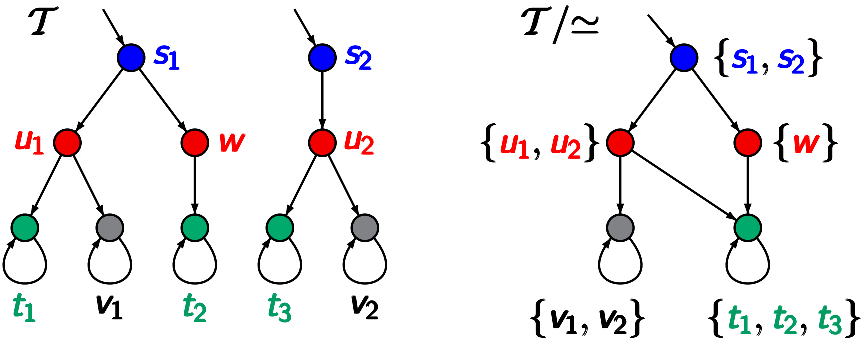
t_1, t_2, t_3 are simulation equivalent

v_1, v_2 are simulation equivalent

$u_1 \simeq u_2$, $\quad w \preceq u_1, u_2$, but $w \not\simeq u_1, u_2$

$s_1 \simeq s_2$

Example: simulation quotient



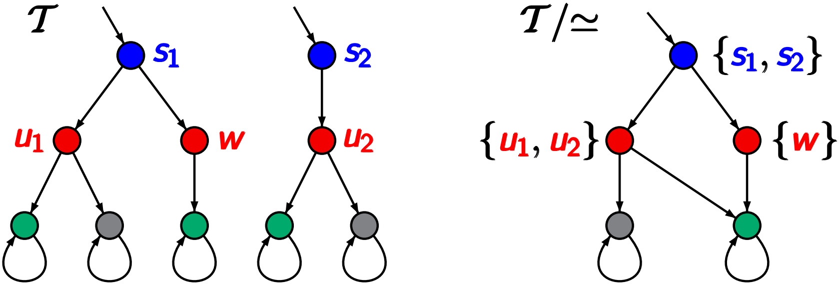
t_1, t_2, t_3 are simulation equivalent

v_1, v_2 are simulation equivalent

$u_1 \simeq u_2$, $\quad w \preceq u_1, u_2$, but $w \not\simeq u_1, u_2$

$s_1 \simeq s_2$

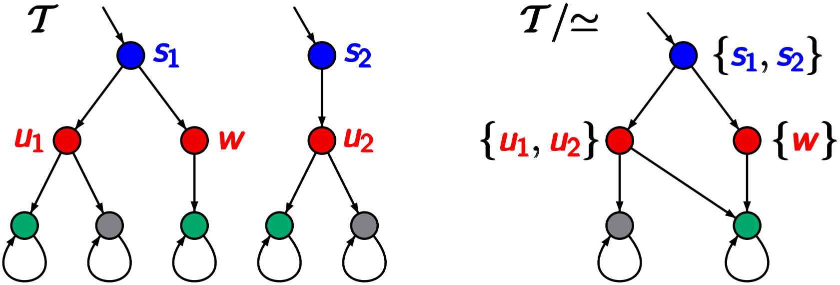
Example: simulation quotient



simulation for $(\mathcal{T}, \mathcal{T}/\simeq)$:

$$\{(s, [s]) : s \text{ is a state in } \mathcal{T}\}$$

Example: simulation quotient



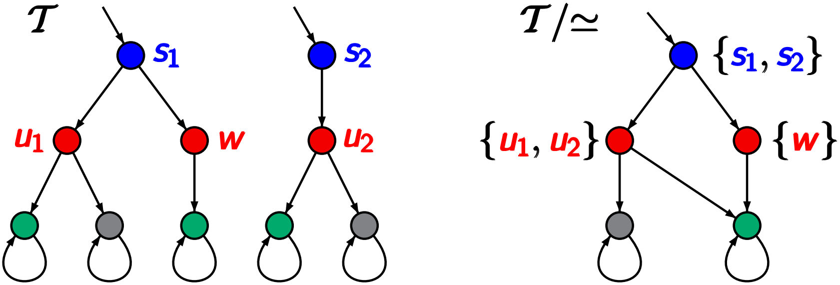
simulation for $(\mathcal{T}, \mathcal{T}/\simeq)$:

$$\{(s, [s]) : s \text{ is a state in } \mathcal{T}\}$$

but $\{([s], s) : s \text{ is a state in } \mathcal{T}\}$

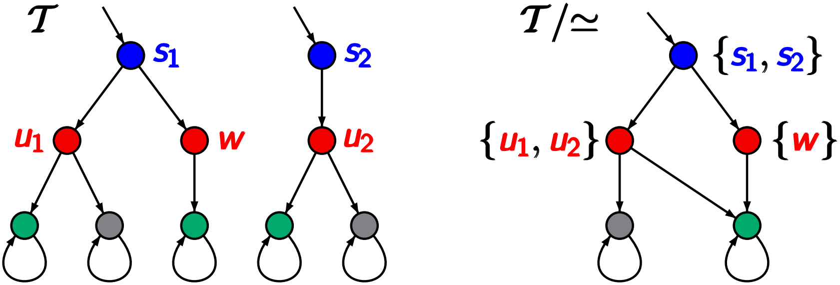
is <u>not</u> a simulation for $(\mathcal{T}/\simeq, \mathcal{T})$

Example: simulation quotient



show that $\mathcal{R} = \{([s], s) : s$ is a state in $\mathcal{T}\}$ is not a simulation for $(\mathcal{T}/\simeq, \mathcal{T})$

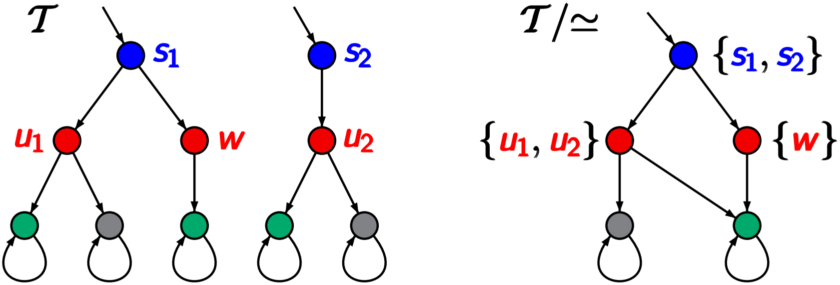
Example: simulation quotient



show that $\mathcal{R} = \{([s], s) : s$ is a state in $\mathcal{T}\}$ is not a simulation for $(\mathcal{T}/\simeq, \mathcal{T})$

regard $(\{s_1, s_2\}, s_2) \in \mathcal{R}$

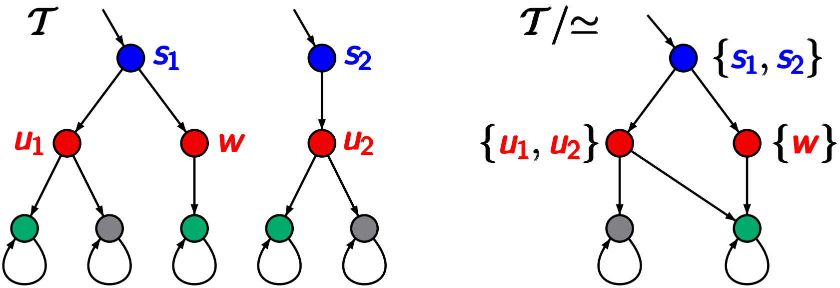
Example: simulation quotient



show that $\mathcal{R} = \{([s], s) : s \text{ is a state in } \mathcal{T}\}$ is <u>not</u> a simulation for $(\mathcal{T}/\simeq, \mathcal{T})$

regard $(\{s_1, s_2\}, s_2) \in \mathcal{R}$ and $\{s_1, s_2\} \rightarrow_{\simeq} \{w\}$

Example: simulation quotient



show that $\mathcal{R} = \{([s], s) : s \text{ is a state in } \mathcal{T}\}$ is not a simulation for $(\mathcal{T}/{\simeq}, \mathcal{T})$

regard $(\{s_1, s_2\}, s_2) \in \mathcal{R}$ and $\{s_1, s_2\} \rightarrow_{\simeq} \{w\}$

there is no transition $s_2 \rightarrow w'$ in $\mathcal{T}$ s.t. $(\{w\}, w') \in \mathcal{R}$

Similarity of $\mathcal{T}$ and $\mathcal{T}/\simeq$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \;=\; (S/\simeq, Act', \rightarrow_{\simeq}, S_0', AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

$\mathcal{T}$ and $\mathcal{T}/\simeq$ are simulation equivalent, i.e.,
$\mathcal{T} \preceq \mathcal{T}/\simeq$ and $\mathcal{T}/\simeq \preceq \mathcal{T}$

Proof. provide simulations for $(\mathcal{T}, \mathcal{T}/\simeq)$ and $(\mathcal{T}/\simeq, \mathcal{T})$

simulation for $(\mathcal{T}, \mathcal{T}/\simeq)$: $\{(s, [s]) : s \in S\}$

simulation for $(\mathcal{T}/\simeq, \mathcal{T})$: **?**

Similarity of $\mathcal{T}$ and $\mathcal{T}/\simeq$

Let $\mathcal{T} = (S, Act, \rightarrow, S_0, AP, L)$ be a TS. Then:

$$\mathcal{T}/\simeq \; = \; (S/\simeq, Act', \rightarrow_{\simeq}, S_0', AP, L')$$

where the transitions are given by $\dfrac{s \longrightarrow s'}{[s] \longrightarrow_{\simeq} [s']}$

> $\mathcal{T}$ and $\mathcal{T}/\simeq$ are simulation equivalent, i.e.,
> $\mathcal{T} \preceq \mathcal{T}/\simeq$ and $\mathcal{T}/\simeq \, \preceq \mathcal{T}$

Proof. provide simulations for $(\mathcal{T}, \mathcal{T}/\simeq)$ and $(\mathcal{T}/\simeq, \mathcal{T})$

simulation for $(\mathcal{T}, \mathcal{T}/\simeq)$: $\{(s, [s]) : s \in S\}$

simulation for $(\mathcal{T}/\simeq, \mathcal{T})$: $\{([s], t) : s \preceq_{\mathcal{T}} t\}$