Overview: Model Checking

1. Introduction
2. Modelling parallel systems
3. Linear Time Properties
4. Regular Properties
5. Linear Temporal Logic
6. Computation Tree Logic
7. Equivalences and Abstraction
8. **Partial Order Reduction**
9. Timed Automata
10. Probabilistic Systems

Basic idea of partial order reduction

- for asynchronous systems

Basic idea of partial order reduction

- for asynchronous systems
- analyze representatives of path equivalence classes

Basic idea of partial order reduction

- for asynchronous systems
- analyze representatives of path equivalence classes that represent the same the same behavior up to the interleaving order

Basic idea of partial order reduction

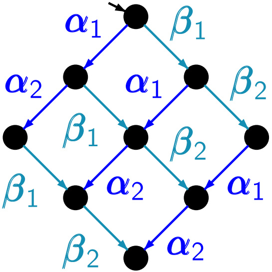
- for asynchronous systems
- analyze representatives of path equivalence classes that represent the same the same behavior up to the interleaving order

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

Basic idea of partial order reduction

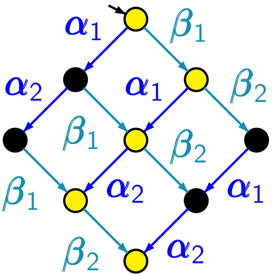
- for asynchronous systems
- analyze representatives of path equivalence classes that represent the same the same behavior up to the interleaving order

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

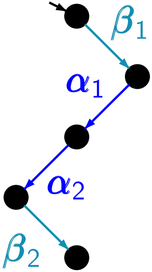


- for asynchronous systems
- analyze representatives of path equivalence classes that represent the same the same behavior up to the interleaving order

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

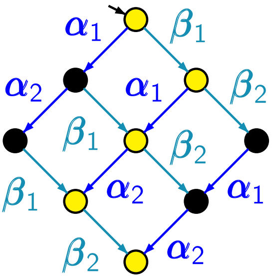


$\mathcal{T}_{\text{red}}$

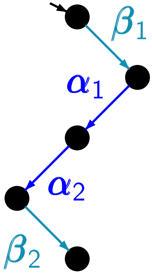


Partial order reduction for $\text{LTL}_{\setminus\bigcirc}$ specifications

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

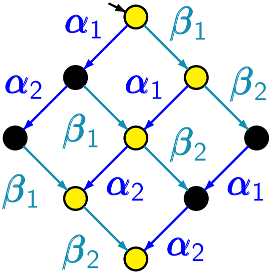


$\mathcal{T}_{\text{red}}$

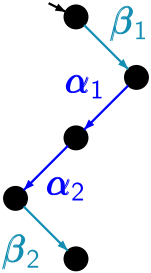


Partial order reduction for $\mathsf{LTL}_{\setminus\bigcirc}$ specifications

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$



$\mathcal{T}_{\text{red}}$

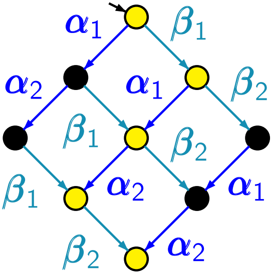


requirement: for all $\mathsf{LTL}_{\setminus\bigcirc}$ formulas φ:

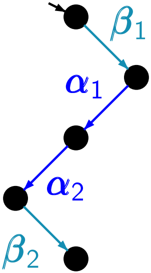
$$\mathcal{T} \models \varphi \quad \text{iff} \quad \mathcal{T}_{\text{red}} \models \varphi$$

Partial order reduction for $\text{LTL}_{\setminus\bigcirc}$ specifications

$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$



$\mathcal{T}_{\text{red}}$



requirement: for all $\text{LTL}_{\setminus\bigcirc}$ formulas φ:

$$\mathcal{T} \models \varphi \quad \text{iff} \quad \mathcal{T}_{\text{red}} \models \varphi$$

hence: ensure that the reduction yields $\mathcal{T} \overset{\Delta}{=} \mathcal{T}_{\text{red}}$

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\text{red}}$
by selecting action-sets $\text{ample}(s) \subseteq \text{Act}(s)$

The ample set method [Peled '93]

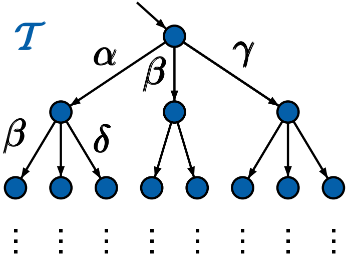
given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\mathsf{red}}$
by selecting action-sets $\mathsf{ample}(s) \subseteq \mathsf{Act}(s)$
and expanding only the α-successors of s
where $\alpha \in \mathsf{ample}(s)$

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

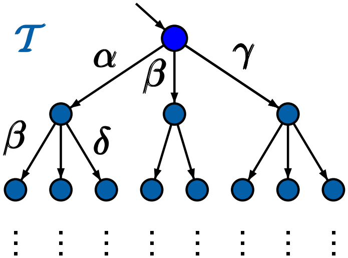
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

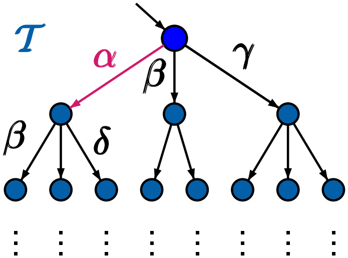
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

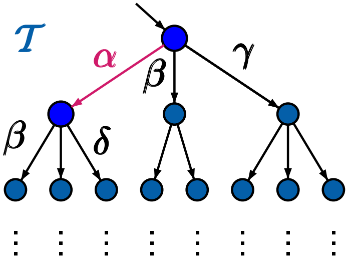
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

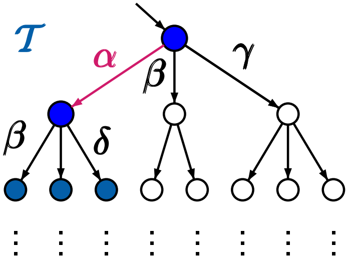
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

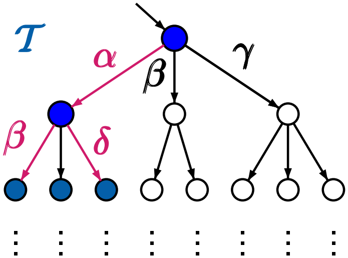
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

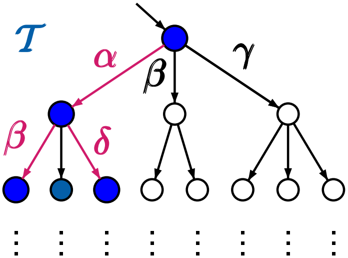
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets ample(s) $\subseteq$ Act(s)
and expanding only the α-successors of s
where $\alpha \in$ ample(s)



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

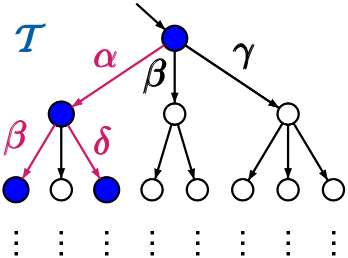
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets ample(s) $\subseteq$ Act(s)
and expanding only the α-successors of s
where $\alpha \in$ ample(s)



The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

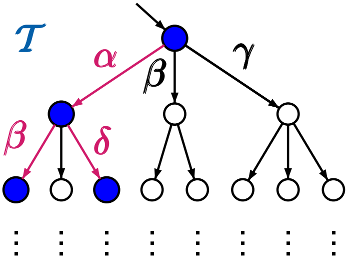
goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets ample(s) $\subseteq$ Act(s)
and expanding only the α-successors of s
where $\alpha \in$ ample(s)

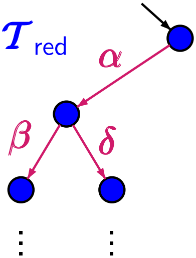


The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\text{red}}$
by selecting action-sets ample(s) ⊆ Act(s)
and expanding only the α-successors of s
where $\alpha \in$ ample(s)





The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$

requirements:

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\mathsf{red}}$
by selecting action-sets $\mathsf{ample}(s) \subseteq \mathsf{Act}(s)$
and expanding only the α-successors of s
where $\alpha \in \mathsf{ample}(s)$

requirements:

- stutter trace equivalence: $\mathcal{T} \triangleq \mathcal{T}_{\mathsf{red}}$

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{red}$
by selecting action-sets $ample(s) \subseteq Act(s)$
and expanding only the α-successors of s
where $\alpha \in ample(s)$

requirements:

- stutter trace equivalence: $\mathcal{T} \triangleq \mathcal{T}_{red}$
  hence: $\mathcal{T}, \mathcal{T}_{red}$ are $\mathbf{LTL}_{\setminus\bigcirc}$ equivalent

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\mathsf{red}}$
by selecting action-sets $\mathsf{ample}(s) \subseteq \mathsf{Act}(s)$
and expanding only the α-successors of s
where $\alpha \in \mathsf{ample}(s)$

requirements:

- stutter trace equivalence: $\mathcal{T} \triangleq \mathcal{T}_{\mathsf{red}}$
  hence: $\mathcal{T}, \mathcal{T}_{\mathsf{red}}$ are $\mathbf{LTL}_{\setminus \bigcirc}$ equivalent
- $\mathcal{T}_{\mathsf{red}}$ is smaller than $\mathcal{T}$

The ample set method [Peled '93]

given: syntactical representation of processes of TS $\mathcal{T}$

goal: on-the-fly construction of a fragment $\mathcal{T}_{\text{red}}$
by selecting action-sets $\text{ample}(s) \subseteq \text{Act}(s)$
and expanding only the α-successors of s
where $\alpha \in \text{ample}(s)$

requirements:

- stutter trace equivalence: $\mathcal{T} \triangleq \mathcal{T}_{\text{red}}$
  hence: $\mathcal{T}$, $\mathcal{T}_{\text{red}}$ are $\mathbf{LTL}_{\setminus \bigcirc}$ equivalent
- $\mathcal{T}_{\text{red}}$ is smaller than $\mathcal{T}$
- efficient construction of $\mathcal{T}_{\text{red}}$ is possible

The reduced transition system $\mathcal{T}_{red}$

is a fragment of $\mathcal{T}$ that results from $\mathcal{T}$ by

- a DFS-based on-the-fly analysis and
- choosing ample sets $ample(s) \subseteq Act(s)$ for each expanded state,
- expanding only the α-successors of s where $\alpha \in ample(s)$

The reduced transition system $\mathcal{T}_{red}$

is a fragment of $\mathcal{T}$ that results from $\mathcal{T}$ by

- a DFS-based on-the-fly analysis and
- choosing ample sets $ample(s) \subseteq Act(s)$ for each expanded state,
- expanding only the α-successors of s where $\alpha \in ample(s)$

transition relation $\Rightarrow$ of $\mathcal{T}_{red}$ is given by:

$$\frac{s \xrightarrow{\alpha} s' \wedge \alpha \in ample(s)}{s \overset{\alpha}{\Longrightarrow} s'}$$

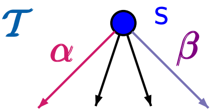
The reduced transition system $\mathcal{T}_{red}$

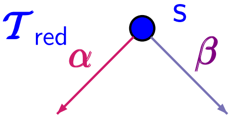
is a fragment of $\mathcal{T}$ that results from $\mathcal{T}$ by

. . . choosing ample sets $ample(s) \subseteq Act(s)$

transition relation $\Rightarrow$ of $\mathcal{T}_{red}$ is given by:

$$\frac{s \xrightarrow{\alpha} s' \ \wedge \ \alpha \in ample(s)}{s \overset{\alpha}{\Longrightarrow} s'}$$





$ample(s) = \{\alpha, \beta\}$

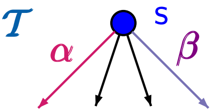
The reduced transition system $\mathcal{T}_{red}$

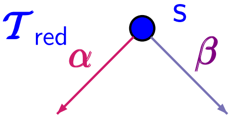
is a fragment of $\mathcal{T}$ that results from $\mathcal{T}$ by

... choosing ample sets $ample(s) \subseteq Act(s)$

transition relation $\Rightarrow$ of $\mathcal{T}_{red}$ is given by:

$$\frac{s \xrightarrow{\alpha} s' \wedge \alpha \in ample(s)}{s \overset{\alpha}{\Longrightarrow} s'}$$





$ample(s) = \{\alpha, \beta\}$

state space S_{red} of $\mathcal{T}_{red}$: all states that are reachable from the initial states in $\mathcal{T}$ via $\Rightarrow$

Action-determinism

LTL3.4-11A

Action-determinism

Let $\mathcal{T} = (\mathbf{S}, Act, \rightarrow, \mathbf{S}_0, \mathrm{AP}, \mathrm{L})$ be a transition system.

Action-determinism

Let $\mathcal{T} = (\mathbf{S}, Act, \rightarrow, \mathbf{S}_0, \mathrm{AP}, \mathrm{L})$ be a transition system.

For state $\mathbf{s}$:

$$Act(\mathbf{s}) \;=\; \{\, \alpha \in Act \;:\; \exists \mathbf{t} \in \mathbf{S} \text{ s.t. } \mathbf{s} \xrightarrow{\alpha} \mathbf{t} \,\}$$

Action-determinism

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

For state s:

$$Act(s) \; = \; \{ \alpha \in Act \; : \; \exists t \in S \text{ s.t. } s \xrightarrow{\alpha} t \}$$

$\mathcal{T}$ is called action-deterministic iff for all states s and all actions $\alpha \in Act(s)$:

$$| \{ t \in S \; : \; s \xrightarrow{\alpha} t \} | \; \leq \; 1$$

Action-determinism

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

For state s:

$$Act(s) \;=\; \{\, \alpha \in Act \;:\; \exists t \in S \text{ s.t. } s \xrightarrow{\alpha} t \,\}$$

$\mathcal{T}$ is called action-deterministic iff for all states s and all actions $\alpha \in Act(s)$:

$$|\{ t \in S \;:\; s \xrightarrow{\alpha} t \}| \;\leq\; 1$$

notation: if $\alpha \in Act(s)$ then

$$\alpha(s) = \text{unique state } t \text{ s.t. } s \xrightarrow{\alpha} t$$

Independence of actions

LTL3.4-11

Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

Independence of actions

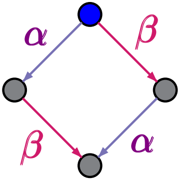
Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set *Act*, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

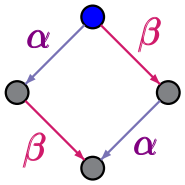


Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

1. $\beta \in Act(\alpha(s))$

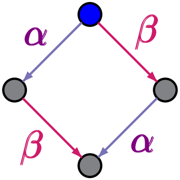


Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

1. $\beta \in Act(\alpha(s))$
2. $\alpha \in Act(\beta(s))$

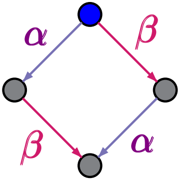


Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

1. $\beta \in Act(\alpha(s))$
2. $\alpha \in Act(\beta(s))$
3. $\beta(\alpha(s)) = \alpha(\beta(s))$

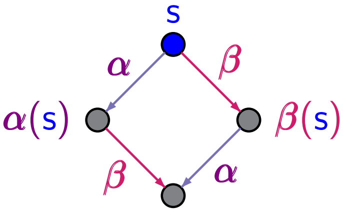


Independence of actions

Let $\mathcal{T}$ be an action-deterministic transition system with action-set Act, and $\alpha, \beta \in Act$.

α, β are called independent in $\mathcal{T}$ if for all states s s.t. $\alpha, \beta \in Act(s)$:

1. $\beta \in Act(\alpha(s))$
2. $\alpha \in Act(\beta(s))$
3. $\beta(\alpha(s)) = \alpha(\beta(s))$



$\beta(\alpha(s)) = \alpha(\beta(s))$

Conditions for ample sets

(A1) nonemptiness condition

$$\emptyset \neq \mathsf{ample}(\mathsf{s}) \subseteq Act(\mathsf{s})$$

Conditions for ample sets

(A1) nonemptiness condition

$$\emptyset \neq \mathsf{ample}(\mathsf{s}) \subseteq Act(\mathsf{s})$$

(A2) dependency condition

Conditions for ample sets

(A1) nonemptiness condition

$$\emptyset \neq \text{ample}(s) \subseteq Act(s)$$

(A2) dependency condition

for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is dependent from ample(s)
there is some $i < n$ with

$$\beta_i \in \text{ample}(s)$$

Conditions for ample sets

(A1) nonemptiness condition

$$\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$$

(A2) dependency condition

for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1}\xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

(A1) nonemptiness condition

$\emptyset \neq \mathsf{ample}(\mathsf{s}) \subseteq Act(\mathsf{s})$

(A2) dependency condition

for each execution fragment in $\mathcal{T}$

$$\mathsf{s} \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(\mathsf{s})$ there is some $\mathsf{i} < \mathsf{n}$ with $\beta_{\mathsf{i}} \in \mathsf{ample}(\mathsf{s})$

(A3) stutter condition

Conditions for ample sets

(A1) nonemptiness condition

$\emptyset \neq \text{ample}(s) \subseteq Act(s)$

(A2) dependency condition

for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from ample(s) there is some $i < n$ with $\beta_i \in \text{ample}(s)$

(A3) stutter condition

if $\text{ample}(s) \neq Act(s)$ then all actions in ample(s) are stutter actions

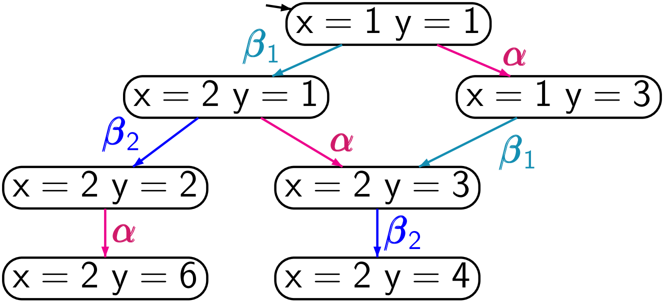
Example

$$\underbrace{x := 2 \cdot x}_{\beta_1} \;\; ; \;\; \underbrace{y := y + 1}_{\beta_2} \;\; ||| \;\; \underbrace{y := 3 \cdot y}_{\alpha}$$

Example

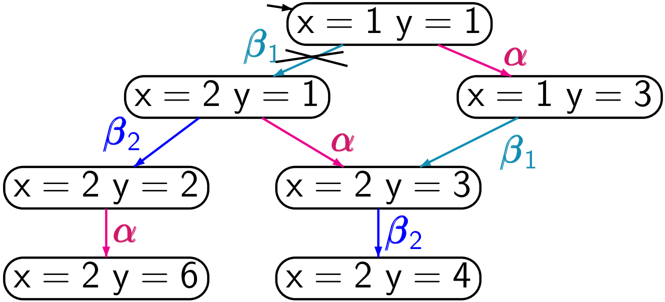
LTL3.4-23

$$\underbrace{x := 2 \cdot x}_{\beta_1} \;\; ; \;\; \underbrace{y := y + 1}_{\beta_2} \;\; ||| \;\; \underbrace{y := 3 \cdot y}_{\alpha}$$



Example

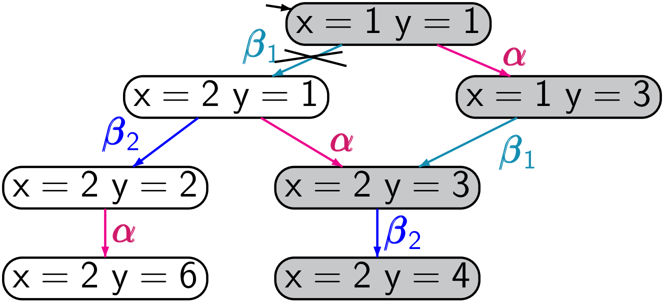
$$\underbrace{x := 2 \cdot x}_{\beta_1} \quad ; \quad \underbrace{y := y + 1}_{\beta_2} \quad ||| \quad \underbrace{y := 3 \cdot y}_{\alpha}$$



Example

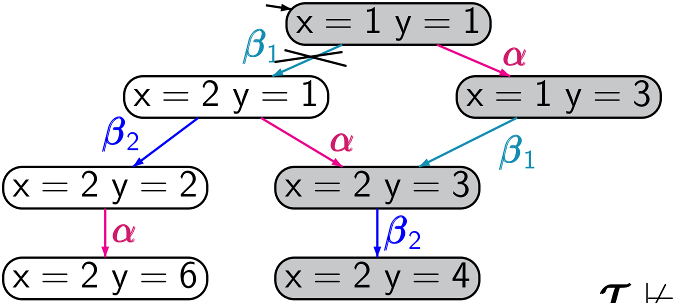
LTL3.4-23

$$\underbrace{x := 2 \cdot x}_{\beta_1} \;;\; \underbrace{y := y + 1}_{\beta_2} \;|||\; \underbrace{y := 3 \cdot y}_{\alpha}$$



Example

$$\underbrace{x := 2 \cdot x}_{\beta_1} \;\; ; \;\; \underbrace{y := y + 1}_{\beta_2} \;\; ||| \;\; \underbrace{y := 3 \cdot y}_{\alpha}$$

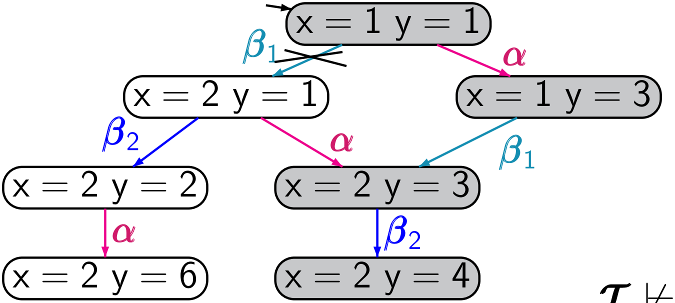


$$\mathcal{T} \not\models \Box(y \neq 6)$$

$$\mathcal{T}_{\text{red}} \models \Box(y \neq 6)$$

Example

$$\underbrace{x := 2 \cdot x}_{\beta_1} \;;\; \underbrace{y := y + 1}_{\beta_2} \;\;|||\;\; \underbrace{y := 3 \cdot y}_{\alpha}$$



$\mathcal{T} \not\models \Box(y \neq 6)$

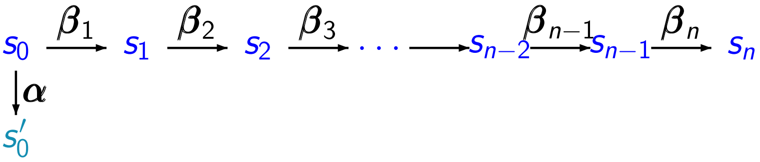
$\mathcal{T}_{red} \models \Box(y \neq 6)$

(A2) violated as β_2, α dependent

Conditions (A2) and (A3)

Suppose

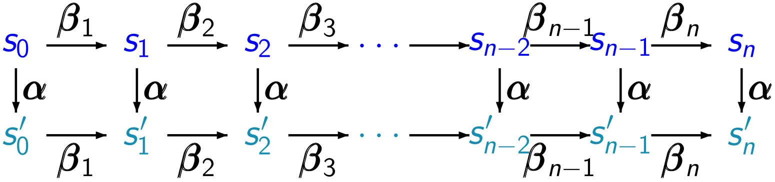
- $\alpha \in$ ample(s_0), $\beta_i \notin$ ample(s_0)
- α stutter action



Conditions (A2) and (A3)

Suppose

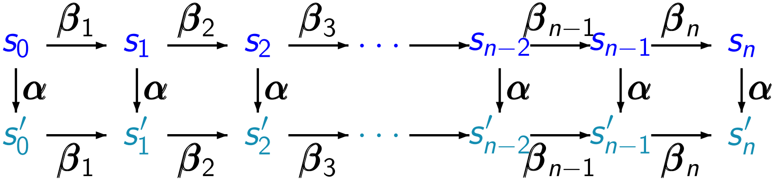
- $\alpha \in \text{ample}(s_0)$, $\beta_i \notin \text{ample}(s_0)$
- α stutter action



Conditions (A2) and (A3)

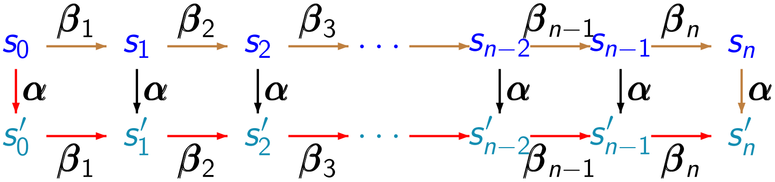
Suppose

- $\alpha \in$ ample(s_0), $\beta_i \notin$ ample(s_0)
- α stutter action $\Rightarrow$ $L(s_i) = L(s_i')$, $i = 0, 1, 2, \ldots$

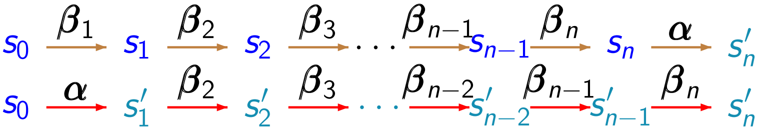


Conditions (A2) and (A3)

- $\alpha \in \mathrm{ample}(s_0)$, $\beta_i \notin \mathrm{ample}(s_0)$
- α stutter action $\Rightarrow L(s_i) = L(s_i')$, $i = 0, 1, 2, \ldots$

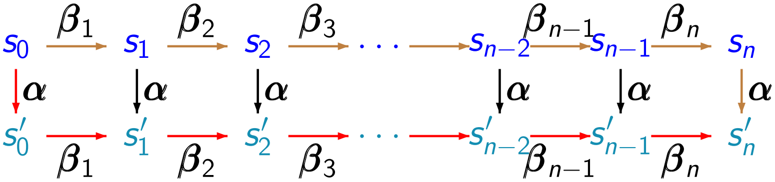


case 1:

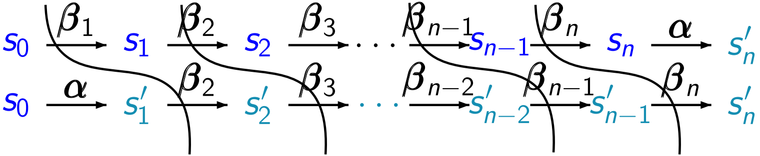


Conditions (A2) and (A3)

- $\alpha \in \text{ample}(s_0)$, $\beta_i \notin \text{ample}(s_0)$
- α stutter action $\Rightarrow$ $L(s_i) = L(s_i')$, $i = 0, 1, 2, \ldots$

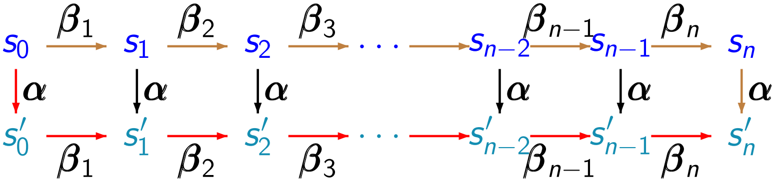


case 1:

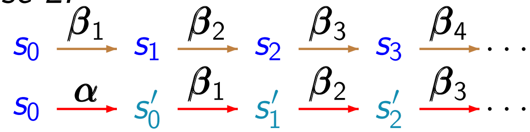


Conditions (A2) and (A3)

- $\alpha \in ample(s_0)$, $\beta_i \notin ample(s_0)$
- α stutter action $\Rightarrow L(s_i) = L(s'_i)$, $i = 0, 1, 2, \ldots$

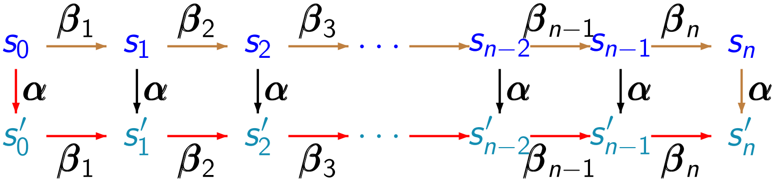


case 2:

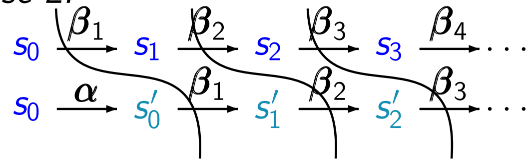


Conditions (A2) and (A3)

- $\alpha \in \text{ample}(s_0)$, $\beta_i \notin \text{ample}(s_0)$
- α stutter action $\Rightarrow$ $L(s_i) = L(s_i')$, $i = 0, 1, 2, \ldots$



case 2:



Conditions (A1), (A2), (A3) are not sufficient

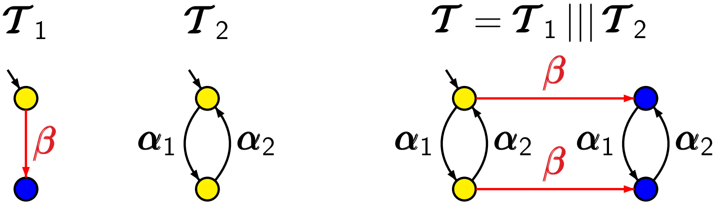
LTL3.4-30

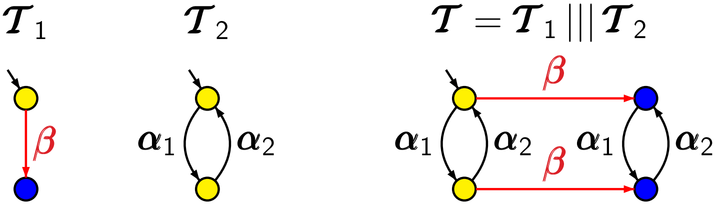
Conditions (A1), (A2), (A3) are not sufficient

There exists a finite, action-deterministic transition system $\mathcal{T}$ and ample sets for $\mathcal{T}$ such that

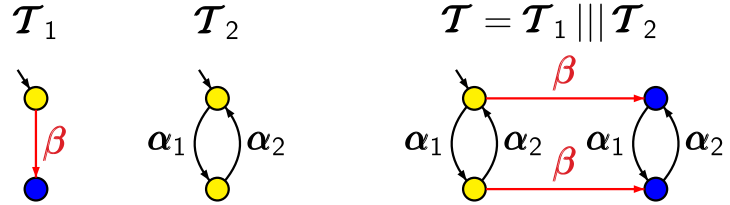
$$\mathcal{T} \not\triangleq \mathcal{T}_{\text{red}}$$

remind: $\triangleq$ stutter trace equivalence


$\mathcal{T}_1$
β
$\mathcal{T}_2$
α_1
α_2
$\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$
β
α_1
α_2
β
α_1
α_2

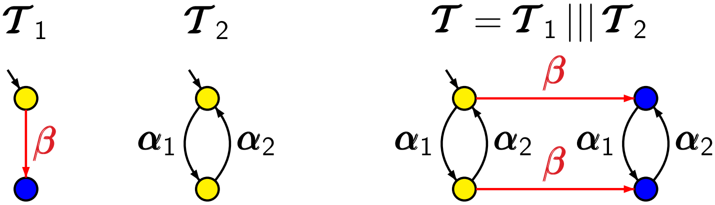


β, α_i independent



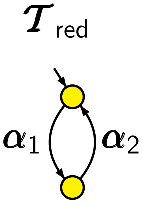
β, α_i independent

α_1, α_2 stutter actions

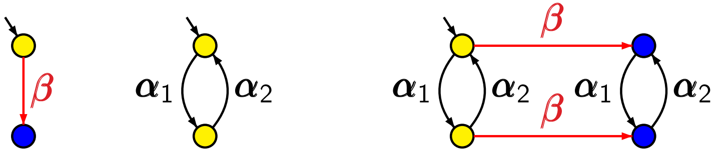


β, α_i independent

α_1, α_2 stutter actions



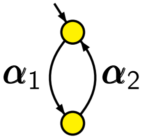
$\mathcal{T}_1$ $\mathcal{T}_2$ $\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$



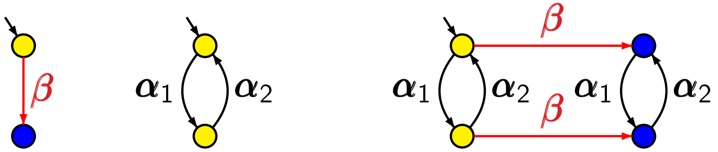
$\boldsymbol{\beta}, \boldsymbol{\alpha}_i$ independent

$\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ stutter actions

$\mathcal{T}_{\text{red}}$ satisfies (A1), (A2), (A3)



$\mathcal{T}_1$ $\mathcal{T}_2$ $\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

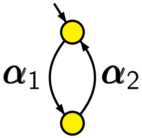


$\mathcal{T} \not\models \Box\neg$blue

β, α_i independent

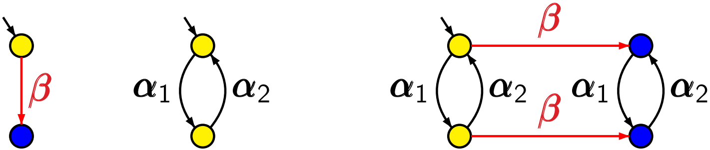
α_1, α_2 stutter actions

$\mathcal{T}_{\text{red}}$ satisfies (A1), (A2), (A3)



$\mathcal{T}_{\text{red}} \models \Box\neg$blue

$\mathcal{T}_1$ $\mathcal{T}_2$ $\mathcal{T} = \mathcal{T}_1 \,|||\, \mathcal{T}_2$

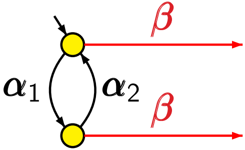


$\mathcal{T} \not\models \Box\neg\text{blue}$

β, α_i independent

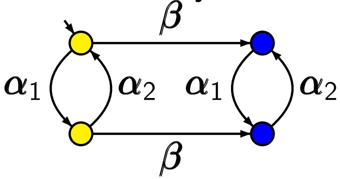
α_1, α_2 stutter actions

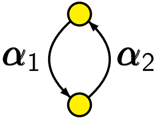
$\mathcal{T}_{\text{red}}$ satisfies (A1), (A2), (A3)

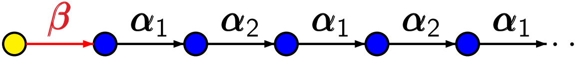


$\mathcal{T}_{\text{red}} \models \Box\neg\text{blue}$

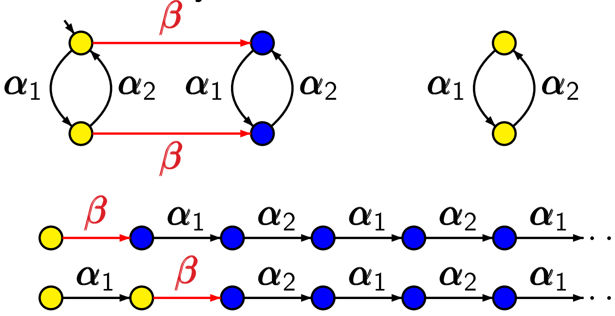
transition system $\mathcal{T}$ reduced TS $\mathcal{T}_{\text{red}}$



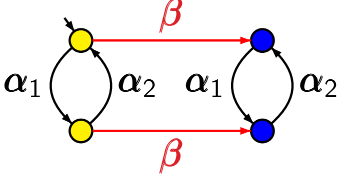


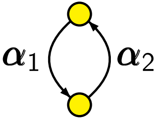


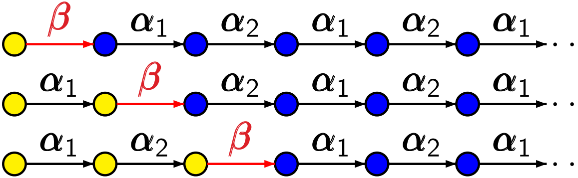
transition system $\mathcal{T}$ reduced TS $\mathcal{T}_{red}$

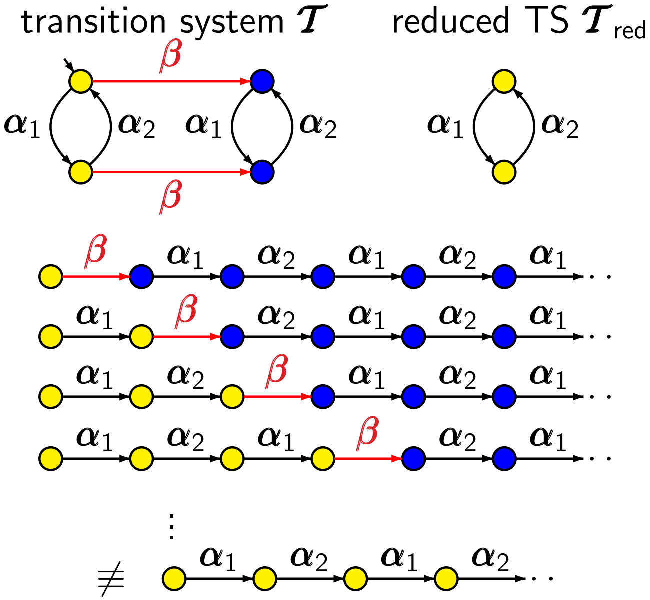


transition system $\mathcal{T}$ reduced TS $\mathcal{T}_{red}$

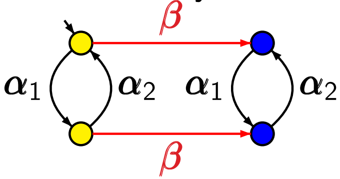


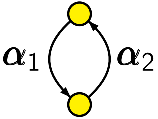


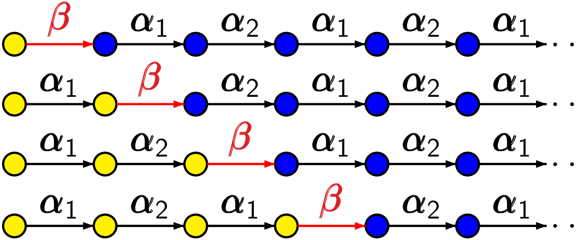



transition system $\mathcal{T}$
reduced TS $\mathcal{T}_{red}$
β
α_1
α_2
$\ne$

transition system $\mathcal{T}$ reduced TS $\mathcal{T}_{\text{red}}$

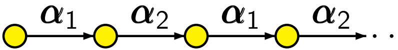






$\vdots$

$\neq$



$=$ the unique execution of $\mathcal{T}_{\text{red}}$

4 conditions for ample sets LTL3.4-A4

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

4 conditions for ample sets

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

4 conditions for ample sets

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1}\xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

(A3) if $\mathsf{ample}(s) \neq Act(s)$ then all actions in $\mathsf{ample}(s)$ are *stutter actions*

4 conditions for ample sets

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

(A3) if $\mathsf{ample}(s) \neq Act(s)$ then all actions in $\mathsf{ample}(s)$ are *stutter actions*

(A4) cycle condition

4 conditions for ample sets LTL3.4-A4

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

(A3) if $\mathsf{ample}(s) \neq Act(s)$ then all actions in $\mathsf{ample}(s)$ are *stutter actions*

(A4) for each *cycle* $s_0 \Rightarrow s_1 \Rightarrow \ldots \Rightarrow s_n$ in $\mathcal{T}_{red}$ and each action

$$\beta \in \bigcup_{0 \leq i < n} Act(s_i)$$

there is some $i \in \{1, \ldots, n\}$ with $\beta \in \mathsf{ample}(s_i)$

4 conditions for ample sets

(A1) $\emptyset \neq \mathsf{ample}(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1}\xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $\mathsf{ample}(s)$ there is some $i < n$ with $\beta_i \in \mathsf{ample}(s)$

(A3) if $\mathsf{ample}(s) \neq Act(s)$ then all actions in $\mathsf{ample}(s)$ are *stutter actions*

(A4) for each *cycle* $s_0 \Rightarrow s_1 \Rightarrow \ldots \Rightarrow s_n$ in $\mathcal{T}_{red}$ and each action

$$\beta \in \bigcup_{0 \leq i < n} Act(s_i)$$

there is some $i \in \{1, \ldots, n\}$ with $\beta \in \mathsf{ample}(s_i)$

4 conditions for ample sets

(A1) $\emptyset \neq ample(s) \subseteq Act(s)$

(A2) for each execution fragment in $\mathcal{T}$

$$s \xrightarrow{\beta_1} \xrightarrow{\beta_2} \ldots \xrightarrow{\beta_{i-1}} \xrightarrow{\beta_i} \xrightarrow{\beta_{i+1}} \ldots \xrightarrow{\beta_{n-1}} \xrightarrow{\beta_n}$$

such that β_n is *dependent* from $ample(s)$ there is some $i < n$ with $\beta_i \in ample(s)$

(A3) if $ample(s) \neq Act(s)$ then all actions in $ample(s)$ are *stutter actions*

(A4) for each *cycle* $s_0 \Rightarrow s_1 \Rightarrow \ldots \Rightarrow s_n$ in $\mathcal{T}_{red}$ and each action

$$\beta \in \bigcup_{0 \leq i < n} Act(s_i)$$

there is some $i \in \{1, \ldots, n\}$ with $\beta \in ample(s_i)$

Soundness of conditions (A1), (A2), (A3), (A4)

Soundness of conditions (A1), (A2), (A3), (A4)

Let $\mathcal{T}$ be a finite, action-deterministic transition system.

Soundness of conditions (A1), (A2), (A3), (A4)

Let $\mathcal{T}$ be a finite, action-deterministic transition system.

If the ample sets ample(s) satisfy conditions (A1), (A2), (A3), (A4) then

$$\mathcal{T} \triangleq \mathcal{T}_{\text{red}}$$

remind: $\triangleq$ stutter trace equivalence

Soundness of conditions (A1), (A2), (A3), (A4)

Let $\mathcal{T}$ be a finite, action-deterministic transition system.

If the ample sets ample(s) satisfy conditions (A1), (A2), (A3), (A4) then

$$\mathcal{T} \triangleq \mathcal{T}_{\mathsf{red}}$$

hence: for all $\mathbf{LTL}_{\setminus\bigcirc}$ formulas φ:

$$\mathcal{T} \models \varphi \quad \text{iff} \quad \mathcal{T}_{\mathsf{red}} \models \varphi$$