

Expressing Fairness in LTL

Luca Tesei

Reactive Systems Verification

MSc in Computer Science

University of Camerino

Topics

- Fairness in LTL. State-based approach.
- Action-based LTL fairness. Ad-hoc and general construction.

Material

Reading:

Chapter 5 of the book, pages 225-243.

More:

The slides in the following pages are taken from the material of the course “Introduction to Model Checking” held by Prof. Dr. Ir. Joost-Pieter Katoen at Aachen University.

Recall: action-based fairness

LTLSF3.1-38

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, \text{AP}, L)$:

$$\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$$

where $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \subseteq 2^{\text{Act}}$

\mathcal{F}_{ucond} unconditional fairness assumption

\mathcal{F}_{strong} strong fairness assumption

\mathcal{F}_{weak} weak fairness assumption

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, \text{AP}, L)$:

$$\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$$

where $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \subseteq 2^{\text{Act}}$

execution $\mathcal{S}_0 \xrightarrow{\alpha_1} \mathcal{S}_1 \xrightarrow{\alpha_2} \mathcal{S}_2 \xrightarrow{\alpha_3} \dots$ \mathcal{F} -fair if

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, \text{AP}, L)$:

$$\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$$

where $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \subseteq 2^{\text{Act}}$

execution $\mathcal{S}_0 \xrightarrow{\alpha_1} \mathcal{S}_1 \xrightarrow{\alpha_2} \mathcal{S}_2 \xrightarrow{\alpha_3} \dots$ \mathcal{F} -fair if

- for all $A \in \mathcal{F}_{ucond}$: $\exists i \geq 1. \alpha_i \in A$

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, \text{AP}, L)$:

$$\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$$

where $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \subseteq 2^{\text{Act}}$

execution $s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \xrightarrow{\alpha_3} \dots$ \mathcal{F} -fair if

- for all $A \in \mathcal{F}_{ucond}$: $\exists^{\infty} i \geq 1. \alpha_i \in A$
- for all $A \in \mathcal{F}_{strong}$:

$$\exists^{\infty} i \geq 1. A \cap \text{Act}(s_i) \neq \emptyset \implies \exists^{\infty} i \geq 1. \alpha_i \in A$$

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, \text{AP}, L)$:

$$\mathcal{F} = (\mathcal{F}_{\text{ucond}}, \mathcal{F}_{\text{strong}}, \mathcal{F}_{\text{weak}})$$

where $\mathcal{F}_{\text{ucond}}, \mathcal{F}_{\text{strong}}, \mathcal{F}_{\text{weak}} \subseteq 2^{\text{Act}}$

execution $\mathcal{S}_0 \xrightarrow{\alpha_1} \mathcal{S}_1 \xrightarrow{\alpha_2} \mathcal{S}_2 \xrightarrow{\alpha_3} \dots$ \mathcal{F} -fair if

- for all $A \in \mathcal{F}_{\text{ucond}}$: $\exists i \geq 1. \alpha_i \in A$
- for all $A \in \mathcal{F}_{\text{strong}}$:

$$\exists i \geq 1. A \cap \text{Act}(\mathcal{S}_i) \neq \emptyset \implies \exists i \geq 1. \alpha_i \in A$$
- for all $A \in \mathcal{F}_{\text{weak}}$:

$$\forall i \geq 1. A \cap \text{Act}(\mathcal{S}_i) \neq \emptyset \implies \exists i \geq 1. \alpha_i \in A$$

fairness assumption for TS $\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \mathcal{S}_0, AP, L)$:

$$\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$$

where $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \subseteq 2^{\text{Act}}$

satisfaction relation for LT-properties under fairness:

$$\mathcal{T} \models_{\mathcal{F}} E \quad \text{iff} \quad \text{for all } \mathcal{F}\text{-fair paths } \pi \text{ of } \mathcal{T}: \\ \text{trace}(\pi) \in E$$

$$\varphi ::= \mathit{true} \mid a \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \mathbf{U} \varphi_2$$

eventually $\diamond\varphi \stackrel{\text{def}}{=} \mathit{true} \mathbf{U} \varphi$

always $\square\varphi \stackrel{\text{def}}{=} \neg\diamond\neg\varphi$

infinitely often $\square\diamond\varphi$

eventually forever $\diamond\square\varphi$

$$\varphi ::= \mathit{true} \mid a \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \mathbf{U}\varphi_2$$

eventually $\diamond\varphi \stackrel{\text{def}}{=} \mathit{true} \mathbf{U}\varphi$

always $\square\varphi \stackrel{\text{def}}{=} \neg\diamond\neg\varphi$

infinitely often $\square\diamond\varphi$

eventually forever $\diamond\square\varphi$

e.g., unconditional fairness $\square\diamond\mathit{crit}_i$

strong fairness $\square\diamond\mathit{wait}_i \rightarrow \square\diamond\mathit{crit}_i$

$$\varphi ::= \mathbf{true} \mid \mathbf{a} \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \mathbf{U} \varphi_2$$

eventually $\diamond\varphi \stackrel{\text{def}}{=} \mathbf{true} \mathbf{U} \varphi$

always $\square\varphi \stackrel{\text{def}}{=} \neg\diamond\neg\varphi$

infinitely often $\square\diamond\varphi$

eventually forever $\diamond\square\varphi$

e.g., unconditional fairness $\square\diamond\mathbf{crit}_i$

strong fairness $\square\diamond\mathbf{wait}_i \rightarrow \square\diamond\mathbf{crit}_i$

weak fairness $\diamond\square\mathbf{wait}_i \rightarrow \square\diamond\mathbf{crit}_i$

... are **conjunctions** of LTL formulas of the form:

- unconditional fairness $\Box\Diamond\phi$
- strong fairness $\Box\Diamond\phi_1 \rightarrow \Box\Diamond\phi_2$
- weak fairness $\Diamond\Box\phi_1 \rightarrow \Box\Diamond\phi_2$

where ϕ_1, ϕ_2, ϕ are propositional formulas

... are **conjunctions** of LTL formulas of the form:

- unconditional fairness $\Box\Diamond\phi$
- strong fairness $\Box\Diamond\phi_1 \rightarrow \Box\Diamond\phi_2$
- weak fairness $\Diamond\Box\phi_1 \rightarrow \Box\Diamond\phi_2$

where ϕ_1, ϕ_2, ϕ are propositional formulas

If *fair* is a LTL fairness assumption, *s* a state in a TS, and φ an LTL formula then

... are **conjunctions** of LTL formulas of the form:

- unconditional fairness $\Box\Diamond\phi$
- strong fairness $\Box\Diamond\phi_1 \rightarrow \Box\Diamond\phi_2$
- weak fairness $\Diamond\Box\phi_1 \rightarrow \Box\Diamond\phi_2$

where ϕ_1, ϕ_2, ϕ are propositional formulas

If **fair** is a LTL fairness assumption, **s** a state in a TS, and φ an LTL formula then

$s \models_{\text{fair}} \varphi$ iff for all $\pi \in \text{Paths}(s)$:
if $\pi \models_{\text{fair}}$ then $\pi \models \varphi$

... are conjunctions of **LTL formulas** of the form:

- unconditional fairness $\Box\Diamond\phi$
- strong fairness $\Box\Diamond\phi_1 \rightarrow \Box\Diamond\phi_2$
- weak fairness $\Diamond\Box\phi_1 \rightarrow \Box\Diamond\phi_2$

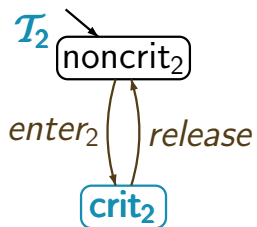
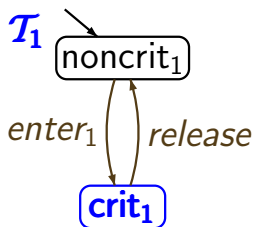
where ϕ_1, ϕ_2, ϕ are propositional formulas

If **fair** is a LTL fairness assumption, **s** a state in a TS, and φ an LTL formula then

$$\begin{aligned} s \models_{\text{fair}} \varphi & \text{ iff for all } \pi \in \text{Paths}(s): \\ & \text{if } \pi \models \text{fair} \text{ then } \pi \models \varphi \\ & \text{iff } s \models \text{fair} \rightarrow \varphi \end{aligned}$$

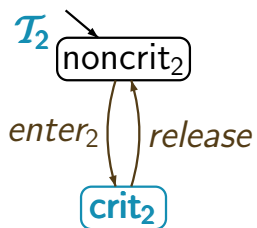
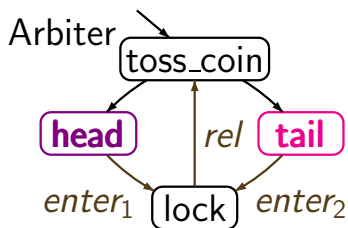
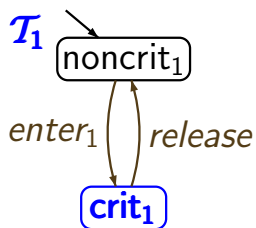
Randomized arbiter for MUTEX

LTLSF3.1-40



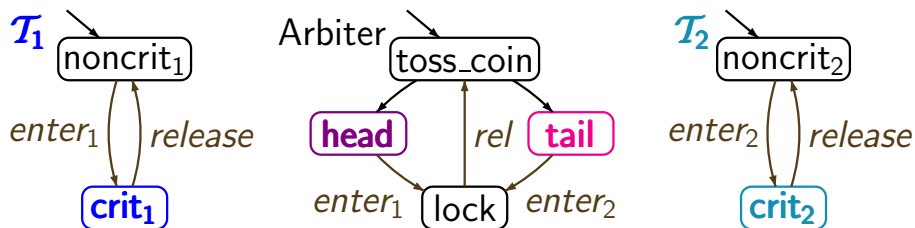
Randomized arbiter for MUTEX

LTLSF3.1-40

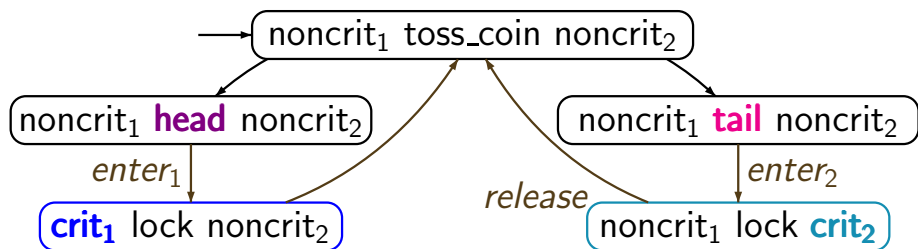


Randomized arbiter for MUTEX

LTLSF3.1-40

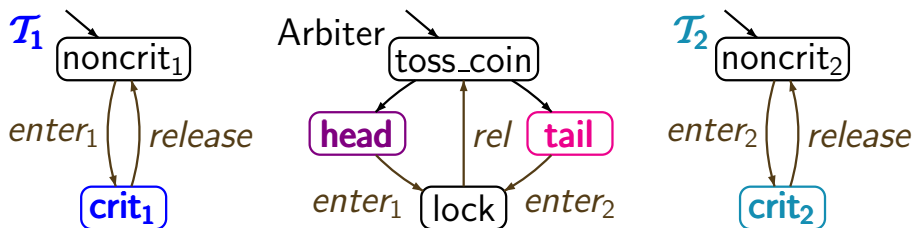


$(\mathcal{T}_1 \parallel \mathcal{T}_2) \parallel \text{Arbiter}$

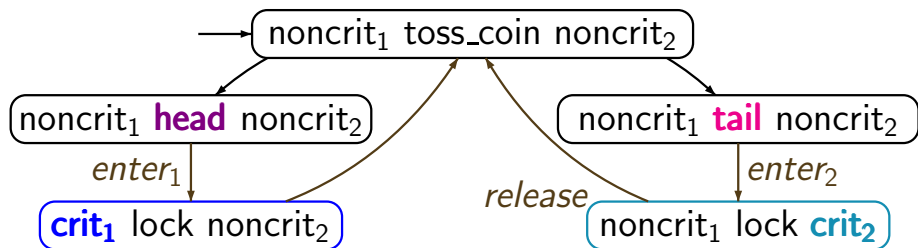


Randomized arbiter for MUTEX

LTLSF3.1-40

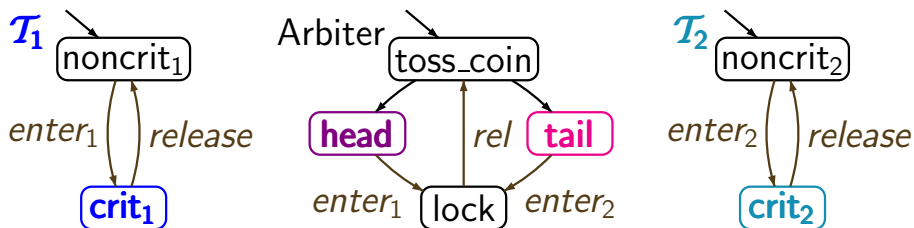


$$(\mathcal{T}_1 \parallel \mathcal{T}_2) \parallel \text{Arbiter} \not\models \square \diamond \text{crit}_1 \wedge \square \diamond \text{crit}_2$$



Randomized arbiter for MUTEX

LTLSF3.1-40

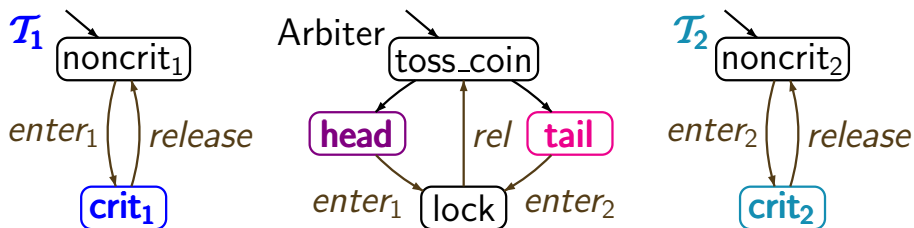


unconditional LTL-fairness:

$$\text{fair} = \square \diamond \text{head} \wedge \square \diamond \text{tail}$$

Randomized arbiter for MUTEX

LTLSF3.1-40



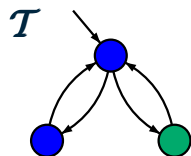
unconditional LTL-fairness:

$$\text{fair} = \square \diamond \text{head} \wedge \square \diamond \text{tail}$$

$$(\mathcal{T}_1 \parallel \mathcal{T}_2) \parallel \text{Arbiter} \models_{\text{fair}} \square \diamond \text{crit}_1 \wedge \square \diamond \text{crit}_2$$

Correct or wrong?

LTLSF3.1-41

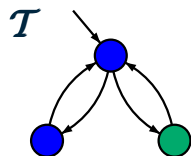


LTL fairness assumption
fair = $\diamond \Box a \rightarrow \Box \diamond b$

● $\hat{=} \{a\}$ ● $\hat{=} \{b\}$

Correct or wrong?

LTLSF3.1-41



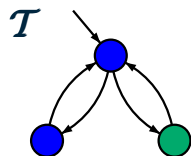
LTL fairness assumption
 $fair = \diamond \Box a \rightarrow \Box \diamond b$

● $\hat{=} \{a\}$ ● $\hat{=} \{b\}$

$\mathcal{T} \models_{fair} \bigcirc b$?

Correct or wrong?

LTLSF3.1-41



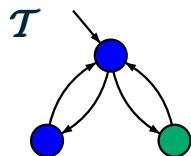
LTL fairness assumption
 $fair = \diamond \Box a \rightarrow \Box \diamond b$

$\bullet \hat{=} \{a\}$ $\bullet \hat{=} \{b\}$

$\mathcal{T} \not\models_{fair} \bigcirc b$ as $\bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \dots$ is fair

Correct or wrong?

LTLSF3.1-41



LTL fairness assumption
 $fair = \diamond \Box a \rightarrow \Box \diamond b$

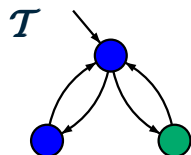
$\bullet \hat{=} \{a\}$ $\bullet \hat{=} \{b\}$

$\mathcal{T} \not\models_{fair} \bigcirc b$ as $\bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \dots$ is fair

$\mathcal{T} \models_{fair} a \cup b$?

Correct or wrong?

LTLSF3.1-41



LTL fairness assumption
 $fair = \diamond \Box a \rightarrow \Box \diamond b$

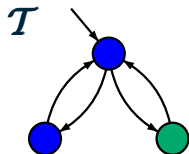
$\bullet \hat{=} \{a\}$ $\bullet \hat{=} \{b\}$

$\mathcal{T} \not\models_{fair} \bigcirc b$ as $\bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \dots$ is fair

$\mathcal{T} \models_{fair} a \cup b$ \checkmark

Correct or wrong?

LTLSF3.1-41



LTL fairness assumption
fair = $\diamond \Box a \rightarrow \Box \diamond b$

● $\hat{=}\{a\}$ ● $\hat{=}\{b\}$

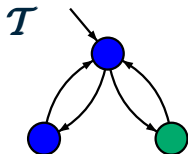
$\mathcal{T} \not\models_{\text{fair}} \bigcirc b$ as ● \rightarrow ● \rightarrow ● \rightarrow ● \rightarrow ● \rightarrow ● \rightarrow ... is fair

$\mathcal{T} \models_{\text{fair}} a \cup b$ ✓

$\mathcal{T} \models_{\text{fair}} a \cup \Box (b \leftrightarrow \bigcirc a)$?

Correct or wrong?

LTLSF3.1-41



LTL fairness assumption

$$\text{fair} = \diamond \Box a \rightarrow \Box \diamond b$$

$$\bullet \hat{=} \{a\} \quad \bullet \hat{=} \{b\}$$

$\mathcal{T} \not\models_{\text{fair}} \bigcirc b$ as $\bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \dots$ is fair

$\mathcal{T} \models_{\text{fair}} a \cup b$ \checkmark

$\mathcal{T} \not\models_{\text{fair}} a \cup \Box (b \leftrightarrow \bigcirc a)$

as $\bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \dots$ is fair

- can be necessary to **prove liveness properties**, e.g., mutual exclusion with arbiter/semaphore

$$\mathcal{I}_{sem} \not\models \square \diamond crit_1 \wedge \square \diamond crit_2$$

$$\mathcal{I}_{sem} \models_{fair} \square \diamond crit_1 \wedge \square \diamond crit_2$$

for appropriate fairness condition

- can be necessary to **prove liveness properties**, e.g., mutual exclusion with arbiter/semaphore

$$\mathcal{T}_{sem} \not\models \square \diamond crit_1 \wedge \square \diamond crit_2$$

$$\mathcal{T}_{sem} \models_{fair} \square \diamond crit_1 \wedge \square \diamond crit_2$$

for appropriate fairness condition, e.g.,

$$fair = \bigwedge_{i=1,2} ((\square \diamond wait_i \rightarrow \square \diamond crit_i) \wedge (\diamond \square noncrit_i \rightarrow \square \diamond wait_i))$$

- can be necessary to prove liveness properties, e.g., mutual exclusion with arbiter/semaphore

$$\mathcal{T}_{sem} \not\models \square \diamond crit_1 \wedge \square \diamond crit_2$$

$$\mathcal{T}_{sem} \models_{fair} \square \diamond crit_1 \wedge \square \diamond crit_2$$

for appropriate fairness condition

- can be verifiable system properties

e.g., Peterson algorithm guarantees strong fairness

$$\mathcal{T}_{Pet} \models \square \diamond wait_1 \rightarrow \square \diamond crit_1$$

- can be necessary to prove liveness properties, e.g.,

$$\mathcal{T}_{sem} \not\models \square\Diamond crit_1 \wedge \square\Diamond crit_2$$

$$\mathcal{T}_{sem} \models_{fair} \square\Diamond crit_1 \wedge \square\Diamond crit_2$$

for appropriate fairness condition

- can be verifiable system properties, e.g.,

$$\mathcal{T}_{Pet} \models \square\Diamond wait_1 \rightarrow \square\Diamond crit_1$$

- are irrelevant for verifying safety properties

$$\mathcal{T} \models \varphi_{safe} \quad \text{iff} \quad \mathcal{T} \models_{fair} \varphi_{safe}$$

if *fair* is realizable

Each strong **LTL** fairness assumption

$$\mathit{fair} = \Box\Diamond a \rightarrow \Box\Diamond b$$

is **realizable** for each TS over $AP = \{a, b, \dots\}$.

Each strong **LTL** fairness assumption

$$\mathit{fair} = \Box\Diamond a \rightarrow \Box\Diamond b$$

is **realizable** for each TS over $AP = \{a, b, \dots\}$.

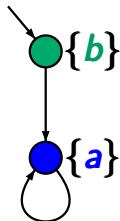
recall: a fairness condition is called **realizable**
if for each reachable state **s** there exists
a fair path starting in **s**

Each strong **LTL** fairness assumption

$$\textit{fair} = \Box\Diamond a \rightarrow \Box\Diamond b$$

is **realizable** for each TS over $AP = \{a, b, \dots\}$.

wrong



$$\textit{fair} = \Box\Diamond a \rightarrow \Box\Diamond b$$

is not realizable

Action-based fairness \rightsquigarrow LTL-fairness

LTLSF3.1-43

idea: use new atomic propositions *enabled(A)* and *taken(A)* and extend the labeling function:

enabled(A) $\in L(s)$ iff $s \xrightarrow{\alpha} \dots$ for some $\alpha \in A$

taken(A) $\in L(s)$ iff for all transitions $\dots \xrightarrow{\alpha} s$:
 $\alpha \in A$

idea: use new atomic propositions **enabled(A)** and **taken(A)** and extend the labeling function:

$$\begin{aligned} \mathit{enabled}(A) \in L(s) & \text{ iff } s \xrightarrow{\alpha} \dots \text{ for some } \alpha \in A \\ \mathit{taken}(A) \in L(s) & \text{ iff for all transitions } \dots \xrightarrow{\alpha} s: \\ & \alpha \in A \end{aligned}$$

- unconditional **A**-fairness: $\Box \Diamond \mathit{taken}(A)$
- strong **A**-fairness: $\Box \Diamond \mathit{enabled}(A) \rightarrow \Box \Diamond \mathit{taken}(A)$
- weak **A**-fairness: $\Diamond \Box \mathit{enabled}(A) \rightarrow \Box \Diamond \mathit{taken}(A)$

idea: use new atomic propositions **enabled(A)** and **taken(A)** and extend the labeling function:

$$\begin{aligned} \text{enabled}(A) \in L(s) & \text{ iff } s \xrightarrow{\alpha} \dots \text{ for some } \alpha \in A \\ \text{taken}(A) \in L(s) & \text{ iff for } \boxed{\text{all}} \text{ transitions } \dots \xrightarrow{\alpha} s: \\ & \alpha \in A \end{aligned}$$

problem: each state **s** can have several incoming transitions

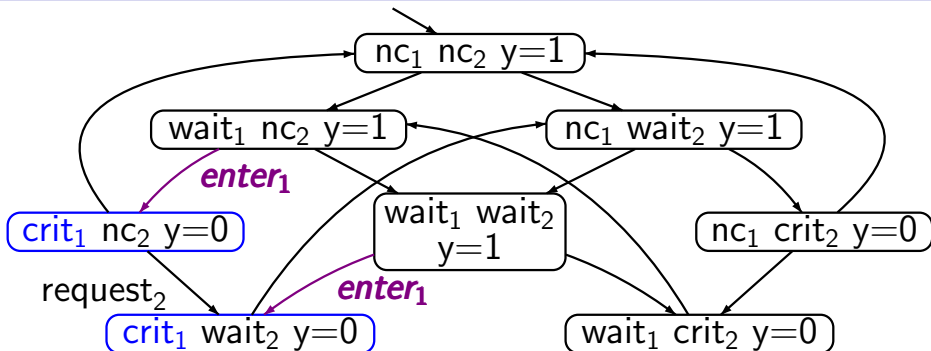
$$t \xrightarrow{\alpha} s, \quad u \xrightarrow{\beta} s, \quad \dots$$

idea: use new atomic propositions **enabled(A)** and **taken(A)** and extend the labeling function:

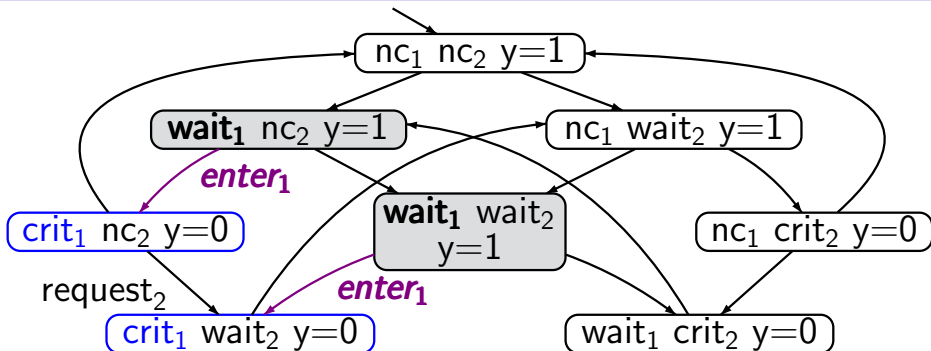
enabled(A) $\in L(s)$ iff $s \xrightarrow{\alpha} \dots$ for some $\alpha \in A$
taken(A) $\in L(s)$ iff for **all** transitions $\dots \xrightarrow{\alpha} s$:
 $\alpha \in A$

alternative 1: ad-hoc choice of “**taken**-predicate”

alternative 2: modify the given transition system by adding an action component to the states

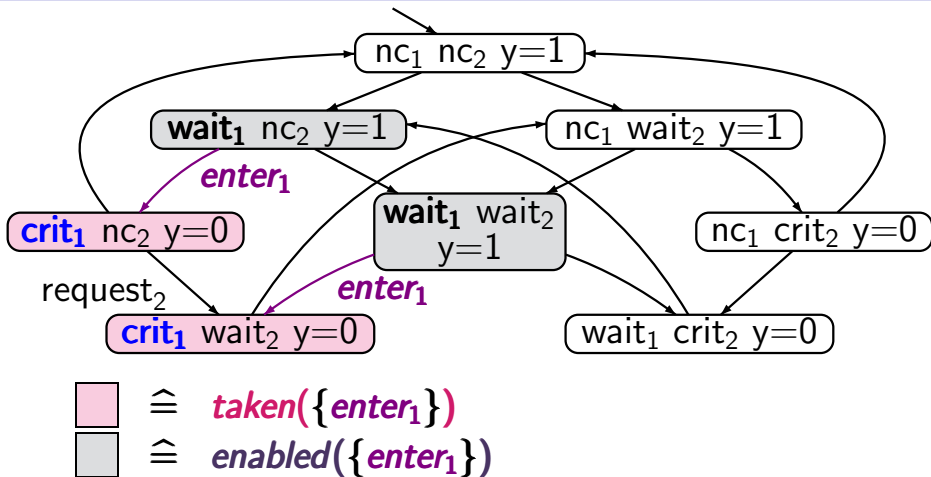


TS for mutual exclusion with semaphore

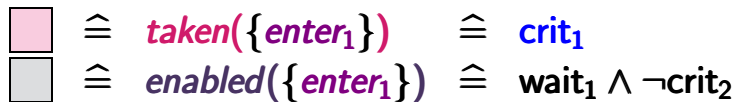
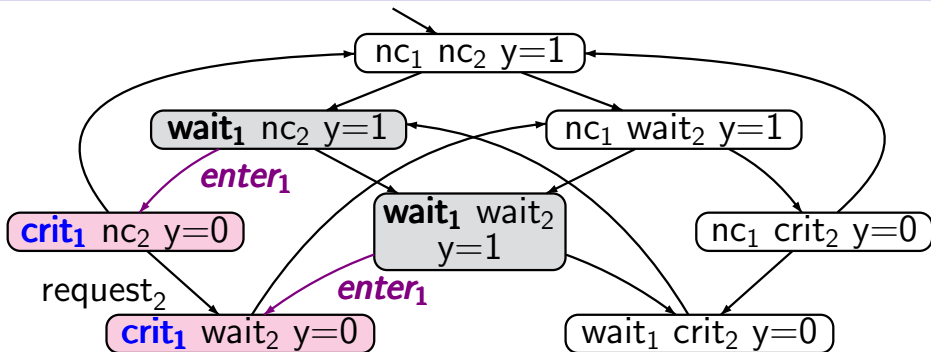


 $\hat{=}$ $enabled(\{enter_1\})$

TS for mutual exclusion with semaphore

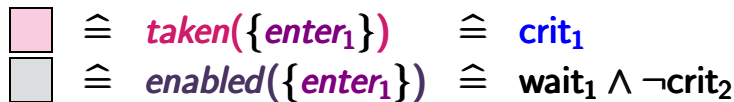
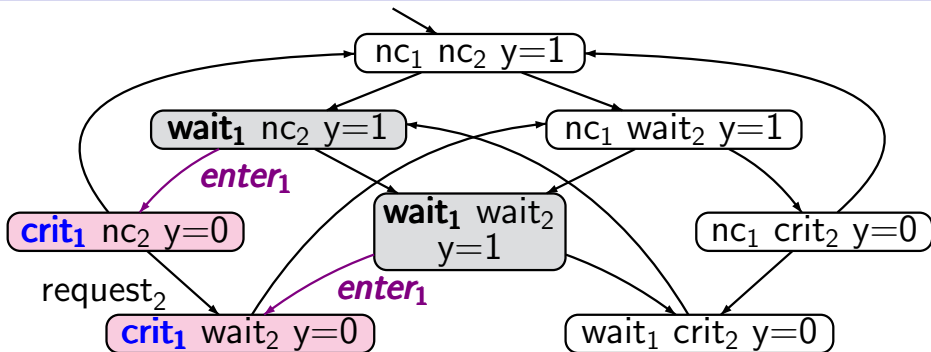


TS for mutual exclusion with semaphore



Ad-hoc: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-44

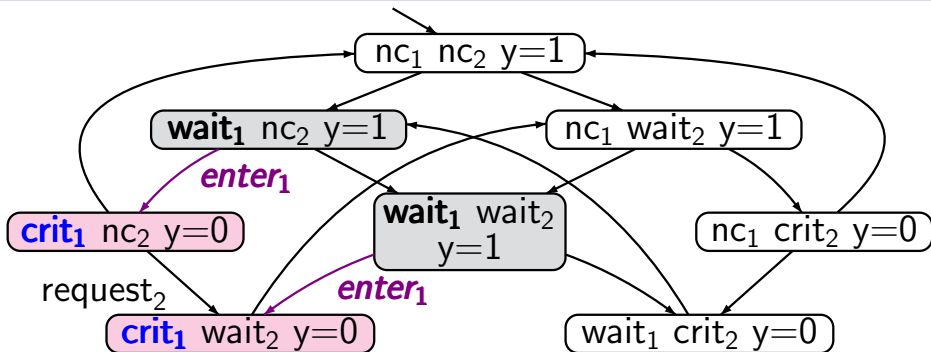


strong *{enter₁}*-fairness: LTL formula

$$\square \diamond \textit{enabled}(\{enter_1\}) \rightarrow \square \diamond \textit{taken}(\{enter_1\})$$

Ad-hoc: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-44



$\hat{=}$ *taken*(*enter₁*) $\hat{=}$ **crit₁**
 $\hat{=}$ *enabled*(*enter₁*) $\hat{=}$ **wait₁ \wedge \neg crit₂**

$\square \diamond \textit{enabled}(\{enter_1\}) \rightarrow \square \diamond \textit{taken}(\{enter_1\})$

$\hat{=}$ $\square \diamond (\textit{wait}_1 \wedge \neg \textit{crit}_2) \rightarrow \square \diamond \textit{crit}_1$

idea: use new atomic propositions **enabled(A)** and **taken(A)** and extend the labeling function:

enabled(A) $\in L(s)$ iff $s \xrightarrow{\alpha} \dots$ for some $\alpha \in A$

taken(A) $\in L(s)$ iff for all transitions $\dots \xrightarrow{\alpha} s$:
 $\alpha \in A$

alternative 1: **ad-hoc choice** of “**taken**-predicate”

alternative 2: modify the given transition system by adding an action component to the states

idea: use new atomic propositions **enabled(A)** and **taken(A)** and extend the labeling function:

enabled(A) $\in L(s)$ iff $s \xrightarrow{\alpha} \dots$ for some $\alpha \in A$

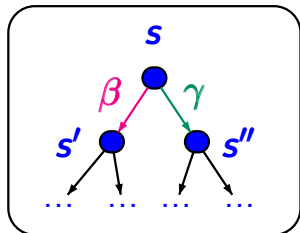
taken(A) $\in L(s)$ iff for all transitions $\dots \xrightarrow{\alpha} s$:
 $\alpha \in A$

alternative 1: ad-hoc choice of “**taken**-predicate”

alternative 2: modify the given transition system by **adding an action component** to the states

transition system

$$\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \dots)$$

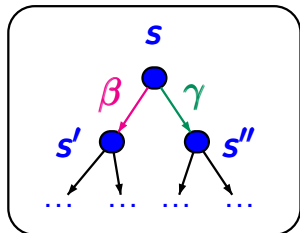


Action-based fairness \rightsquigarrow LTL-fairness

LTLSP3.1-47

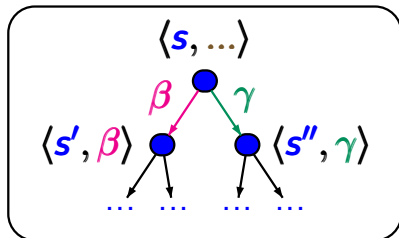
transition system

$$\mathcal{T} = (\mathcal{S}, \text{Act}, \rightarrow, \dots)$$



transition system

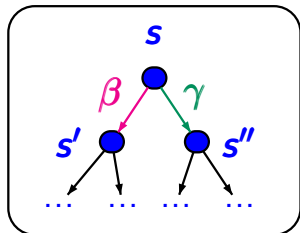
$$\mathcal{T}' = (\mathcal{S} \times \text{Act}, \dots, \text{AP}', L')$$



Action-based fairness \rightsquigarrow LTL-fairness

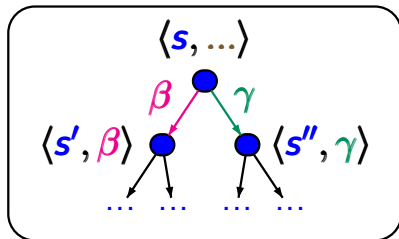
LTLSP3.1-47

transition system
 $\mathcal{T} = (\mathcal{S}, \mathbf{Act}, \rightarrow, \dots)$



strong **A**-fairness
 for $A \subseteq \mathbf{Act}$

transition system
 $\mathcal{T}' = (\mathcal{S} \times \mathbf{Act}, \dots, \mathbf{AP}', L')$

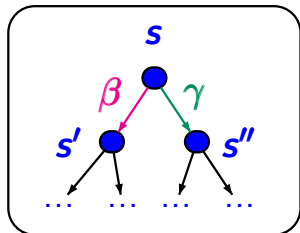


strong **LTL**-fairness
 $\Box \Diamond \mathbf{enabled}(A) \rightarrow \Box \Diamond \mathbf{taken}(A)$

Action-based fairness \rightsquigarrow LTL-fairness

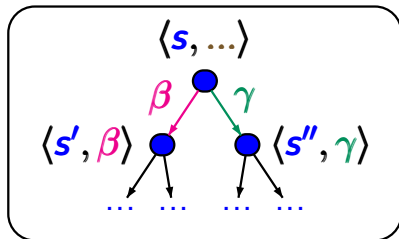
LTLSP3.1-47

transition system
 $\mathcal{T} = (\mathcal{S}, \mathbf{Act}, \rightarrow, \dots)$



strong **A**-fairness
 for $A \subseteq \mathbf{Act}$

transition system
 $\mathcal{T}' = (\mathcal{S} \times \mathbf{Act}, \dots, \mathbf{AP}', L')$



strong **LTL**-fairness
 $\Box \Diamond \mathbf{enabled}(A) \rightarrow \Box \Diamond \mathbf{taken}(A)$

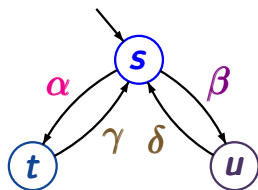
$\mathbf{enabled}(A) \in L'(\langle s, \alpha \rangle)$ iff $s \xrightarrow{\beta} \dots$ for some $\beta \in A$

$\mathbf{taken}(A) \in L'(\langle s, \alpha \rangle)$ iff $\alpha \in A$

Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow LTL-fairness

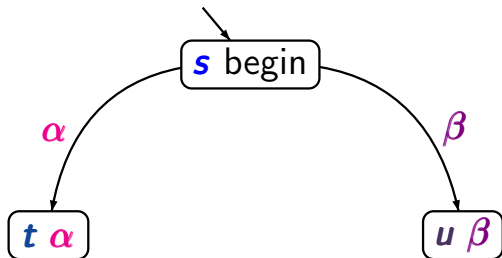
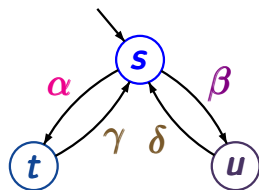


Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow

LTL-fairness

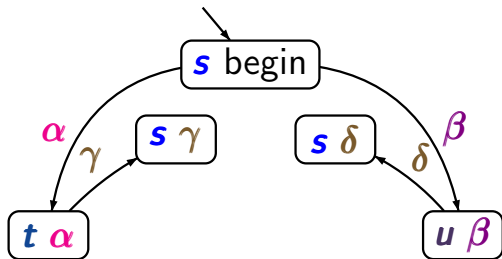
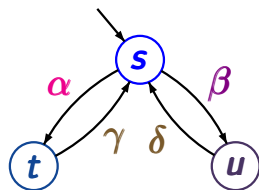


Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow

LTL-fairness

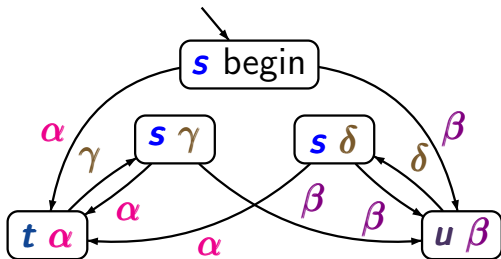
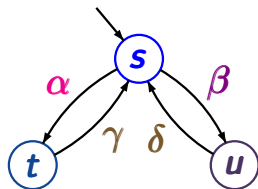


Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow

LTL-fairness

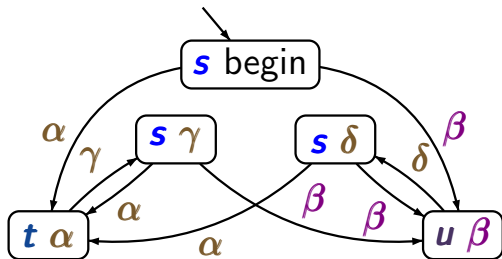
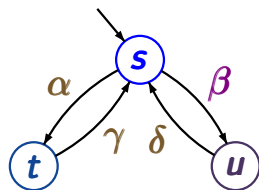


Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow

LTL-fairness

strong fairness for $\{\beta\}$:

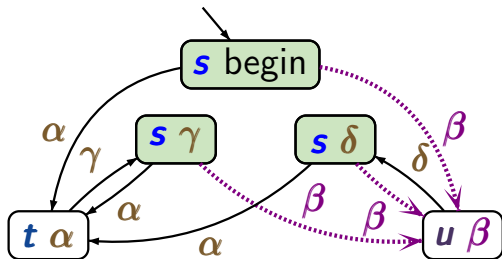
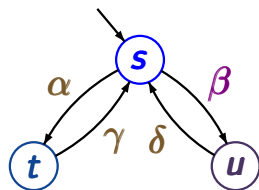
$$\square \diamond \textit{enabled}(\beta) \rightarrow \square \diamond \textit{taken}(\beta)$$

Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow

LTL-fairness

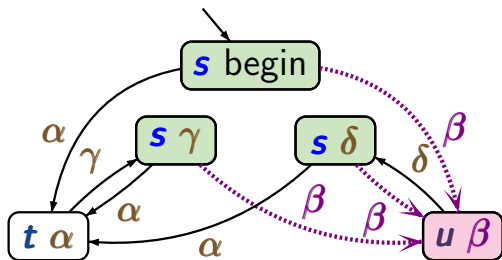
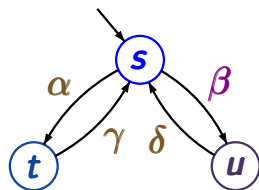
strong fairness for $\{\beta\}$:

$$\square \diamond \text{enabled}(\beta) \rightarrow \square \diamond \text{taken}(\beta)$$

Example: action fairness \rightsquigarrow LTL-fairness

LTLSF3.1-48

action-based fairness \rightsquigarrow LTL-fairness

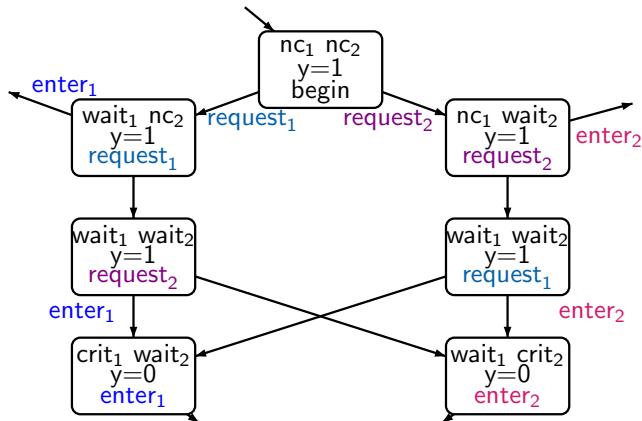


strong fairness for $\{\beta\}$:

$$\square \diamond \text{enabled}(\beta) \rightarrow \square \diamond \text{taken}(\beta)$$

Example: mutual exclusion with semaphore

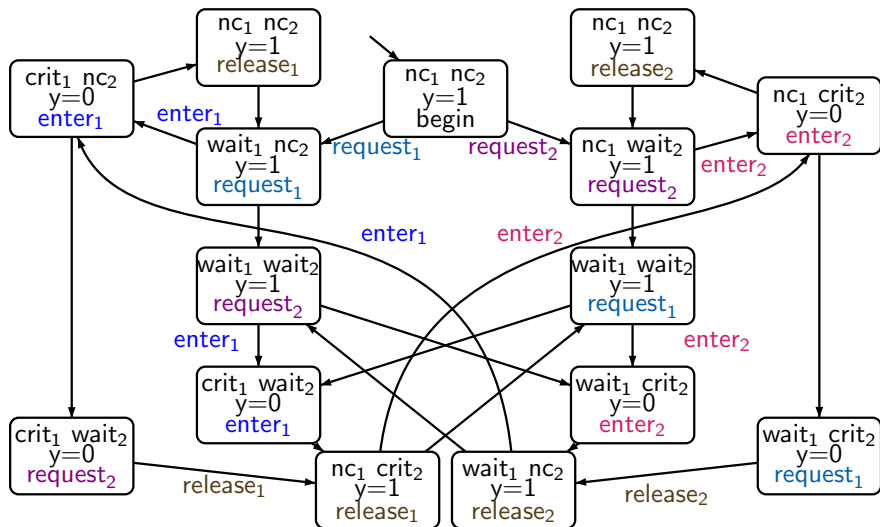
add additional variable `last_action` with domain $\text{Act} \cup \{\text{begin}\}$



Example: mutual exclusion with semaphore

LTLSF3.1-49

add additional variable `last_action` with domain $\text{Act} \cup \{\text{begin}\}$



Example: mutual exclusion with semaphore

LTL3.1-49

add additional variable `last_action` with domain $\text{Act} \cup \{\text{begin}\}$

