Fairness in LTL

Lecture #15 of Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

E-mail: `katoen@cs.rwth-aachen.de`

December 10, 2008

What did we treat so far?

- LTL semantics: for words, states and transition systems
- LTL equivalence: idempotence, duality, absorption, and expansion
- Dual operators to until: weak until and release
- Expansion law as characteristic equation for until and weak until
- Positive normal form
  - for weak until: exponential blow-up of formula
  - for release: linear transformation
- LTL is a specification formalism for LT properties

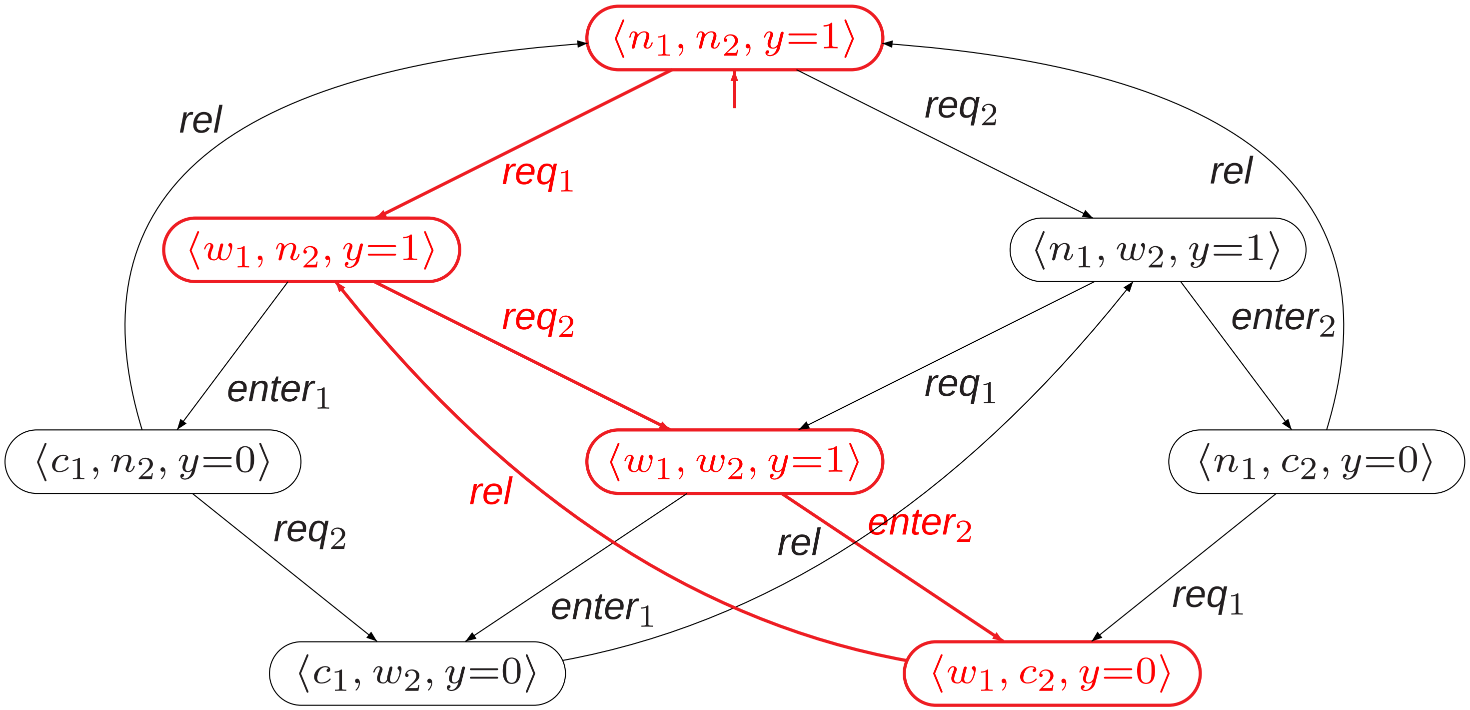
what about fairness in LTL?

Overview Lecture #15

⇒ Repetition: action-based fairness

- State-based fairness in LTL
- Action-based versus state-based fairness
- LTL with fairness constraints

Process one starves



Fairness

- Starvation freedom is often considered under process fairness

$\Rightarrow$ there is a fair scheduling of the execution of processes

- Fairness is typically needed to prove liveness
  - not for safety properties!
  - to prove some form of progress, progress needs to be possible
- Fairness is concerned with a fair resolution of nondeterminism
  - such that it is not biased to consistently ignore a possible option
- Problem: liveness properties constrain infinite behaviours
  - but some traces—that are unfair—refute the liveness property

Summary of fairness

- *Fairness constraints* rule out unrealistic executions
  - by putting constraints on the actions that occur along infinite executions
- Unconditional, strong, and weak fairness constraints
  - unconditional $\Rightarrow$ strong fair $\Rightarrow$ weak fair
  - weak fairness rules out the least number of runs; unconditional the most
- *Fairness assumptions* allow distinct constraints on distinct action sets
- (Realizable) fairness assumptions are irrelevant for safety properties
  - important for the verification of liveness properties

Action-based fairness constraints

For set A of actions and infinite run ρ:

- *Unconditional fairness*

  some action in A occurs infinitely often along ρ

- *Strong fairness*

  if actions in A are *infinitely often* enabled (not necessarily always!)
  then some action in A has to occur infinitely often in ρ

- *Weak fairness*

  if actions in A are *continuously enabled* (no temporary disabling!)
  then it has to occur infinitely often in ρ

Action-based fairness constraints

For $\mathit{TS} = (S, \mathit{Act}, \rightarrow, I, \mathit{AP}, L)$ without terminal states, $A \subseteq \mathit{Act}$,

and infinite execution fragment $\rho = s_0 \xrightarrow{\alpha_0} s_1 \xrightarrow{\alpha_1} \ldots$ of TS:

1. ρ is *unconditionally A-fair* whenever: $\underbrace{\forall k \geqslant 0.\, \exists j \geqslant k.\, \alpha_j \in A}_{\text{infinitely often } A \text{ is taken}}$

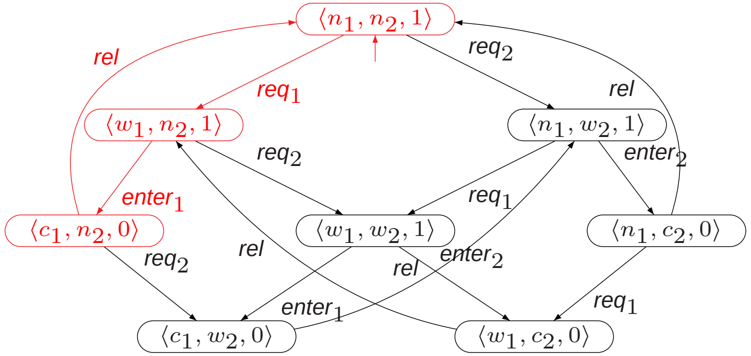
2. ρ is *strongly A-fair* whenever:

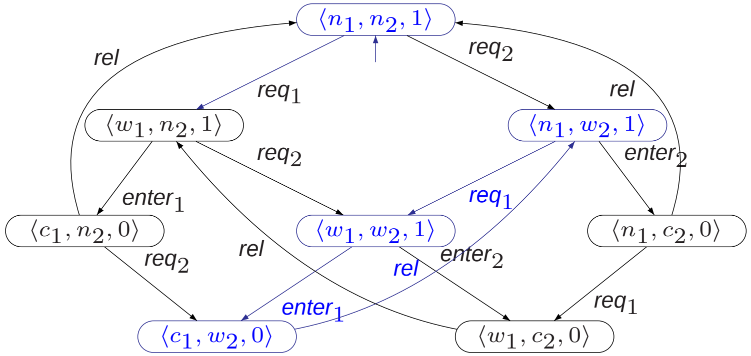
$$\underbrace{(\forall k \geqslant 0.\, \exists j \geqslant k.\, \mathit{Act}(s_j) \cap A \neq \varnothing)}_{\text{infinitely often } A \text{ is enabled}} \implies \underbrace{(\forall k \geqslant 0.\, \exists j \geqslant k.\, \alpha_j \in A)}_{\text{infinitely often } A \text{ is taken}}$$

3. ρ is *weakly A-fair* whenever:

$$\underbrace{(\exists k \geqslant 0.\, \forall j \geqslant k.\, \mathit{Act}(s_j) \cap A \neq \varnothing)}_{A \text{ is eventually always enabled}} \implies \underbrace{(\forall k \geqslant 0.\, \exists j \geqslant k.\, \alpha_j \in A)}_{\text{infinitely often } A \text{ is taken}}$$

Examples





- Run $\langle n_1, n_2, 1\rangle \xrightarrow{\textit{req}_1} \langle w_1, n_2, 1\rangle \xrightarrow{\textit{enter}_1} \langle c_1, n_2, 0\rangle \xrightarrow{\textit{rel}} \langle n_1, n_2, 1\rangle \xrightarrow{\textit{req}_1} \ldots$
  - is not unconditionally A-fair for $A = \{\, \textit{enter}_2 \,\}$
  - but strongly A-fair, as in no ρ-state, the action $\textit{enter}_2$ is enabled

- Run $\langle n_1, n_2, 1\rangle \xrightarrow{\textit{req}_2} \langle n_1, w_2, 1\rangle \xrightarrow{\textit{req}_1} \langle w_1, w_2, 1\rangle \xrightarrow{\textit{enter}_1} \langle c_1, w_2, 0\rangle \xrightarrow{\textit{rel}} \langle n_1, w_2, 1\rangle \ldots$
  - is not strongly A-fair: along ρ, $\textit{enter}_2$ is infinitely often enabled but never taken
  - but weakly A-fair, since $\textit{enter}_2$ is always not enabled along ρ

Fairness assumptions

- A *fairness assumption* for *Act* is a triple

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

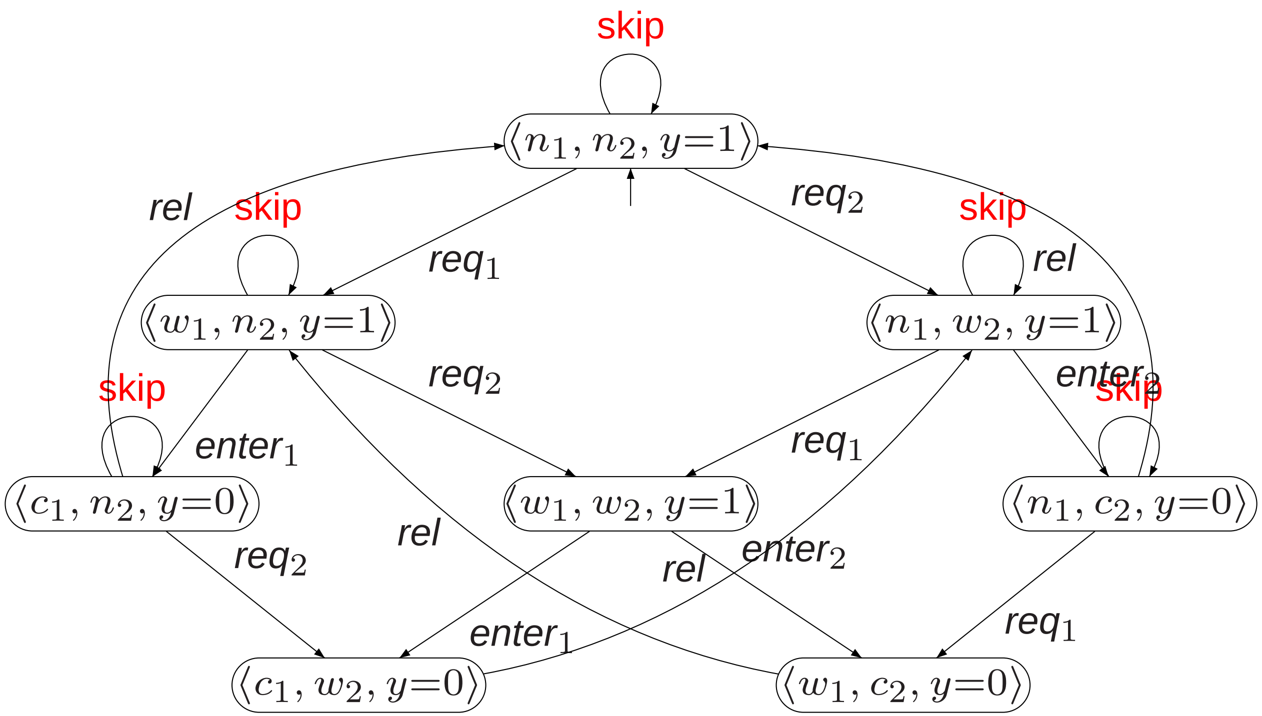
  with $\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak} \in 2^{\mathit{Act}}$.

- Execution ρ is $\mathcal{F}$-fair if:
  - it is unconditionally A-fair for all $A \in \mathcal{F}_{ucond}$, and
  - it is strongly A-fair for all $A \in \mathcal{F}_{strong}$, and
  - it is weakly A-fair for all $A \in \mathcal{F}_{weak}$

- $\mathcal{F}$ is *realizable* for *TS* if for any $s \in \mathit{Reach}(\mathit{TS})$: $\mathit{FairPaths}_{\mathcal{F}}(s) \;\neq\; \varnothing$

fairness assumption $(\varnothing, \mathcal{F}', \varnothing)$ denotes strong fairness; $(\varnothing, \varnothing, \mathcal{F}')$ weak, etc.

Example: fairness assumption for mutual exclusion



$$\mathcal{F}' = \Big(\varnothing, \underbrace{\big\{\{\,\textit{enter}_1\,\},\{\,\textit{enter}_2\,\}\big\}}_{\mathcal{F}_{strong}}, \underbrace{\big\{\{\,\textit{req}_1\,\},\{\,\textit{req}_2\,\}\big\}}_{\mathcal{F}_{weak}}\Big)$$

in any $\mathcal{F}'$-fair execution each process infinitely often requests access

Fair paths and traces

- Let fairness assumption $\mathcal{F} = (\mathcal{F}_{ucond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$
- Path $s_0 \longrightarrow s_1 \longrightarrow s_2 \ldots$ is *$\mathcal{F}$-fair* if
  - there exists an $\mathcal{F}$-fair execution $s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \ldots$
  - $\mathit{FairPaths}_{\mathcal{F}}(s)$ denotes the set of $\mathcal{F}$-fair paths that start in s
  - $\mathit{FairPaths}_{\mathcal{F}}(\mathit{TS}) = \bigcup_{s \in I} \mathit{FairPaths}_{\mathcal{F}}(s)$
- Trace σ is *$\mathcal{F}$-fair* if there exists an $\mathcal{F}$-fair execution ρ with $\mathit{trace}(\rho) = \sigma$
  - $\mathit{FairTraces}_{\mathcal{F}}(s) = \mathit{trace}(\mathit{FairPaths}_{\mathcal{F}}(s))$
  - $\mathit{FairTraces}_{\mathcal{F}}(\mathit{TS}) = \mathit{trace}(\mathit{FairPaths}_{\mathcal{F}}(\mathit{TS}))$

Fair satisfaction

- *TS* *satisfies* LT-property P:

$$\mathit{TS} \models P \quad \text{if and only if} \quad \mathit{Traces}(\mathit{TS}) \subseteq P$$

- *TS* *fairly satisfies* LT-property P wrt. fairness assumption $\mathcal{F}$:

$$\mathit{TS} \models_{\mathcal{F}} P \quad \text{if and only if} \quad \mathit{FairTraces}_{\mathcal{F}}(\mathit{TS}) \subseteq P$$

  – *TS* satisfies the LT property P if *all* its *fair* observable behaviors are admissible

Overview Lecture #15

- Repetition: action-based fairness

$\Rightarrow$ State-based fairness in LTL

- Action-based versus state-based fairness

- LTL with fairness constraints

LTL fairness constraints

Let Φ and Ψ be propositional logic formulas over *AP*.

1. An *unconditional LTL fairness constraint* is of the form:

$$ufair \;=\; \Box\Diamond\Psi$$

2. A *strong LTL fairness condition* is of the form:

$$sfair \;=\; \Box\Diamond\Phi \;\longrightarrow\; \Box\Diamond\Psi$$

3. A *weak LTL fairness constraint* is of the form:

$$wfair \;=\; \Diamond\Box\Phi \;\longrightarrow\; \Box\Diamond\Psi$$

Φ stands for “something is enabled”; Ψ for “something is taken”

LTL fairness assumption

- *LTL fairness assumption* = conjunction of LTL fairness constraints
  - the fairness constraints are of any arbitrary type
- Strong fairness assumption: $sfair = \bigwedge_{0<i\leqslant k} \left(\Box\Diamond\Phi_i \longrightarrow \Box\Diamond\Psi_i \right)$
  - compare this to an action-based strong fairness constraint over A with $|A| = k$
- General format: $fair = ufair \wedge sfair \wedge wfair$
- Rules of thumb:
  - strong (or unconditional) fairness assumptions are useful for solving contentions
  - weak fairness suffices for resolving nondeterminism resulting from interleaving

Fair satisfaction

For state s in transition system *TS* (over *AP*) without terminal states, let

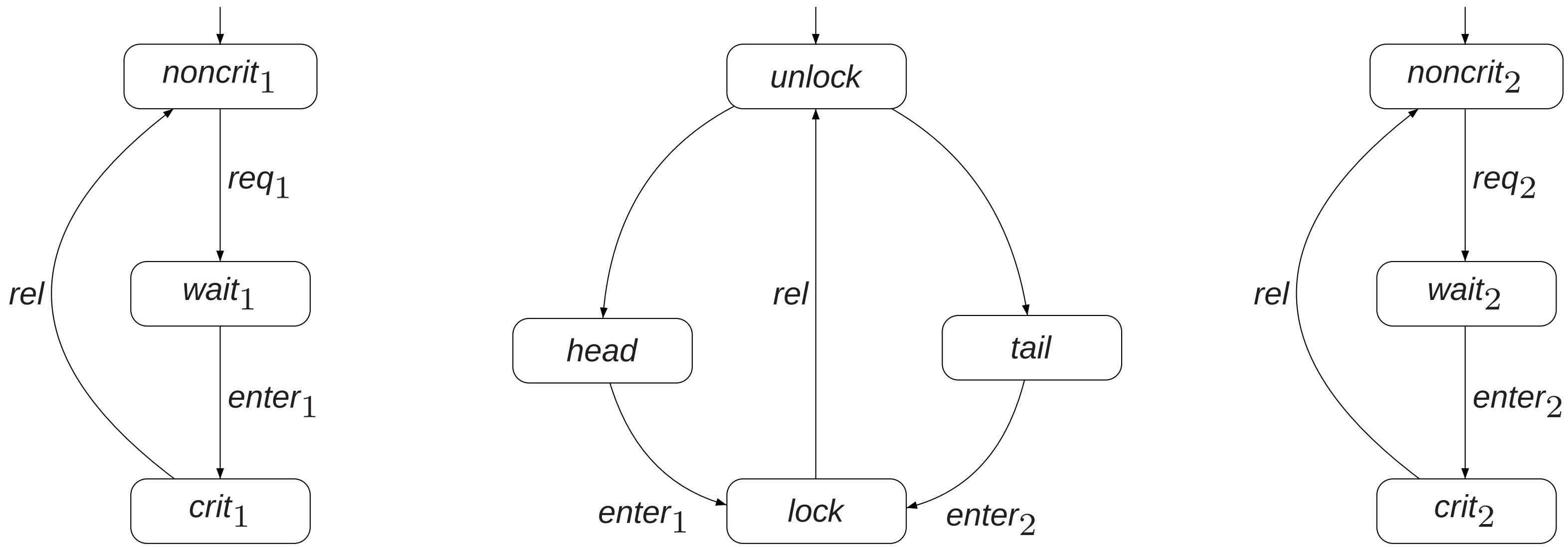
$$\begin{aligned} \mathit{FairPaths}_{fair}(s) &= \{\, \pi \in \mathit{Paths}(s) \mid \pi \models fair \,\} \\ \mathit{FairTraces}_{fair}(s) &= \{\, \mathit{trace}(\pi) \mid \pi \in \mathit{FairPaths}_{fair}(s) \,\} \end{aligned}$$

For LTL-formula φ, and LTL fairness assumption $fair$:

$$\begin{aligned} s \models_{fair} \varphi \quad &\text{if and only if} \quad \forall \pi \in \mathit{FairPaths}_{fair}(s).\, \pi \models \varphi \quad \text{and} \\ TS \models_{fair} \varphi \quad &\text{if and only if} \quad \forall s_0 \in I.\, s_0 \models_{fair} \varphi \end{aligned}$$

$\models_{fair}$ is the *fair satisfaction relation* for LTL; $\models$ the standard one for LTL

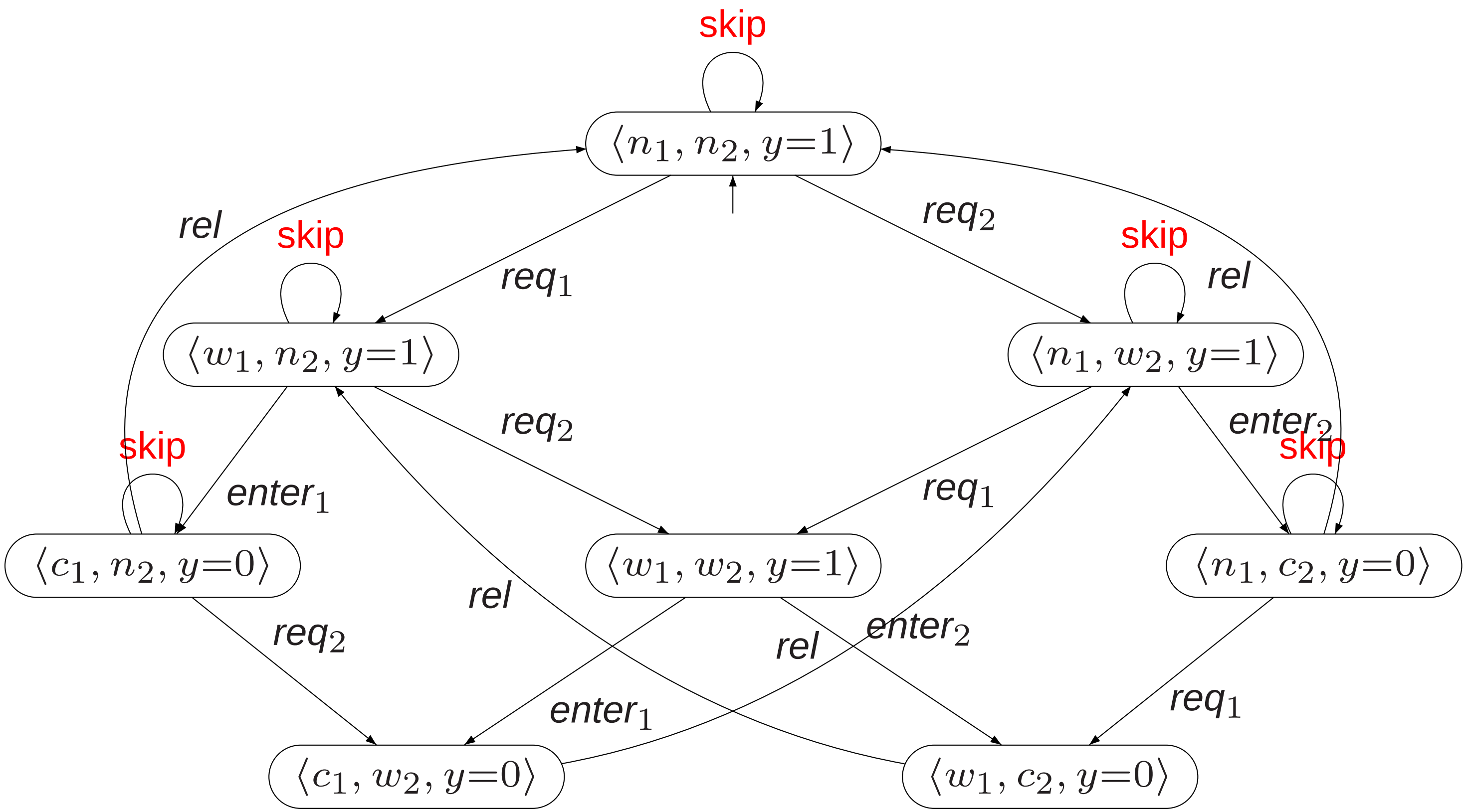
Randomized arbiter



$$TS_1 \parallel Arbiter \parallel TS_2 \;\not\models\; \Box\Diamond\, crit_1$$

But: $TS_1 \parallel Arbiter \parallel TS_2 \models_{fair} \Box\Diamond crit_1 \wedge \Box\Diamond crit_2$ with $fair = \Box\Diamond head \wedge \Box\Diamond tail$

Semaphore-based mutual exclusion



on black board: some action- versus state-based fairness assumptions

State- versus action-based fairness

- From action-based to (state-based) LTL fairness assumptions:
  - premise: deduce from state label, the possible enabled actions
  - conclusion: deduce from state label, the just executed actions

- General scheme:
  - copy each non-initial state s and keep track of action used to enter s
  - copy $\langle s, \alpha \rangle$ means s has been entered via action α

$\Rightarrow$ Any action-based fairness assumption can be transformed into an equivalent LTL fairness assumption

  - the reverse, however, does not hold

Turning action-based into state-based fairness

For $TS = (S, Act, \rightarrow, I, AP, L)$ let $TS' = (S', Act \cup \{ \, begin \, \}, \rightarrow', I', AP', L')$ with:

- $S' = I \times \{ \, begin \, \} \cup S \times Act$ and $I' = I \times \{ \, begin \, \}$
- $\rightarrow'$ is the smallest relation satisfying:

$$\frac{s \xrightarrow{\alpha} s'}{\langle s, \beta \rangle \xrightarrow{\alpha}{}' \langle s', \alpha \rangle} \quad \text{and} \quad \frac{s_0 \xrightarrow{\alpha} s \quad s_0 \in I}{\langle s_0, begin \rangle \xrightarrow{\alpha}{}' \langle s, \alpha \rangle}$$

- $AP' = AP \cup \left\{ \, enabled(\alpha), taken(\alpha) \mid \alpha \in Act \, \right\}$
- labeling function:
  - $L'(\langle s_0, begin \rangle) = L(s_0) \cup \left\{ enabled(\beta) \mid \beta \in Act(s_0) \right\}$
  - $L'(\langle s, \alpha \rangle) = L(s) \cup \left\{ \, taken(\alpha) \, \right\} \cup \left\{ \, enabled(\beta) \mid \beta \in Act(s) \, \right\}$

it follows: $Traces_{AP}(TS) = Traces_{AP}(TS')$

State- versus action-based fairness

- Strong A-fairness is described by the LTL fairness assumption:

$$sfair_A = \Box\Diamond \bigvee_{\alpha \in A} enabled(\alpha) \rightarrow \Box\Diamond \bigvee_{\alpha \in A} taken(\alpha)$$

- The fair traces of *TS* and its action-based variant TS' are equal:

$$\begin{aligned} & \left\{ \mathit{trace}_{\mathit{AP}}(\pi) \mid \pi \in \mathit{Paths}(\mathit{TS}), \pi \text{ is } \mathcal{F}\text{-fair} \right\} \\ = & \left\{ \mathit{trace}_{\mathit{AP}}(\pi') \mid \pi' \in \mathit{Paths}(\mathit{TS}'), \pi' \models fair \right\} \end{aligned}$$

- For every LT-property P (over *AP*): $TS \models_{\mathcal{F}} P$ iff $TS' \models_{fair} P$

Example

Reducing $\models_{fair}$ to $\models$

For:

- transition system *TS* without terminal states
- LTL formula φ, and
- LTL fairness assumption *fair*

it holds:

$$\boxed{TS \models_{fair} \varphi \qquad \text{if and only if} \qquad TS \models (fair \rightarrow \varphi)}$$

verifying an LTL-formula under a fairness assumption can be done using standard verification algorithms for LTL