Bisimulation vs. CTL* and CTL equivalence

Let TS be a finite transition system without terminal states, and let s_1, s_2 states in TS

The following statements are equivalent:

(1) $s_1 \sim_{TS} s_2$

(2) s_1 and s_2 are CTL-equivalent, i.e., $s_1 \equiv_{CTL} s_2$

(3) s_1 and s_2 are CTL*-equivalent, i.e., $s_1 \equiv_{CTL^*} s_2$

this is proven in three steps: $\equiv_{CTL} \subseteq \sim_{TS} \subseteq \equiv_{CTL^*} \subseteq \equiv_{CTL}$

equivalence is also obtained for any sub-logic containing $\neg$, $\wedge$ and EX

$\equiv_{CTL} \subseteq \sim_{TS}$

Let TS be a finite transition system without terminal states, and let s_1, s_2 be states in TS:

$s_1 \equiv_{CTL} s_2$ implies $s_1 \sim_{TS} s_2$.

Proof: We show that $\mathcal{R} = \{(s_1, s_2) \mid s_1 \equiv_{CTL} s_2\}$ is a bisimulation on TS.

- For any $(s_1, s_2) \in \mathcal{R}$, $L(s_1) = L(s_2)$.
  Consider the following CTL state formula Φ over AP:

$$\Phi = \bigwedge_{a \in L(s_1)} a \ \wedge \bigwedge_{a \in AP \setminus L(s_1)} \neg a$$

  Since $s_1 \models \Phi$ and $s_1 \equiv_{CTL} s_2$, it follows that $s_2 \models \Phi$. Hence, $L(s_2) = L(s_1)$.

$\equiv_{CTL} \subseteq \sim_{TS}$ (continued)

- if $s_1' \in Post(s_1)$ then there exists an $s_2' \in Post(s_2)$ with $(s_1', s_2') \in \mathcal{R}$
  - Let $[s_1']$ be the equivalence class of s_1 with respect to $\mathcal{R}$.
    We construct a CTL-formula $\Phi_{[s_1']}$ with $Sat(\Phi_{[s_1']}) = [s_1']$.
    For any pair of equivalence classes $(C, D) \in S/\mathcal{R}$,
    let $\Phi_{C,D}$ be a CTL-formula such that
    $C \subseteq Sat(\Phi_{C,D})$ and $D \cap Sat(\Phi_{C,D}) = \emptyset$.
    Then $\Phi_{[s_1']} = \bigwedge_{D \in S/\mathcal{R}, D \neq [s_1']} \Phi_{[s_1'],D}$.
  - Since $s_1' \in Post(s_1)$, we have $s_1 \models \mathrm{EX}\, \Phi_{[s_1']}$.
  - Since $s_1 \equiv_{CTL} s_2$, we get $s_2 \models \mathrm{EX}\, \Phi_{[s_1']}$.
  - Thus, there is a state $s_2' \in Post(s_2)$ with $s_2' \models \Phi_{[s_1']}$.
  - Hence, $s_2' \in [s_1']$, and therefore $(s_1', s_2') \in \mathcal{R}$.

- if $s_2' \in Post(s_2)$ then there exists an $s_1' \in Post(s_1)$ with $(s_1', s_2') \in \mathcal{R}$
  analogous ($\mathcal{R}$ is an equivalence relation).

$\sim_{TS} \subseteq \equiv_{CTL^*}$

Let TS be a transition system without terminal states, let s_1, s_2 be states in TS, and π_1, π_2 be infinite path fragments in TS:

(a) If $s_1 \sim_{TS} s_2$, then for any CTL* state formula Φ: $s_1 \models \Phi$ iff $s_2 \models \Phi$
(b) If $\pi_1 \sim_{TS} \pi_2$, then for any CTL* path formula φ: $\pi_1 \models \varphi$ iff $\pi_2 \models \varphi$

Proof: By **induction** over the structure of the formula.

- $\Phi = a \in AP$:

$$s_1 \models a \quad \text{iff} \quad a \in L(s_1) \quad \text{iff} \quad a \in L(s_2) \quad \text{iff} \quad s_2 \models a$$

- $\Phi = \neg\Psi$:

$$s_1 \models \neg\Phi \quad \text{iff} \quad s_1 \not\models \Phi \quad \text{iff} \quad s_2 \not\models \Phi \quad \text{iff} \quad s_2 \models \neg\Phi$$

- ...

$\sim_{TS} \subseteq \equiv_{CTL^*}$ (continued)

- $\Phi = \mathsf{E}\,\varphi$:
  - Assume $s_1 \models \mathsf{E}\,\varphi$. Then there exists path π starting in s_1 that satisfies φ.
  - Then there exists a path π_2 starting in s_2 such that $\pi_1 \sim_{TS} \pi_2$.
  - From the induction hypothesis, it follows that $\pi_2 \models \varphi$, and therefore $s_2 \models \mathsf{E}\,\varphi$
- $\varphi = \mathsf{X}\,\psi$:

  $\pi_1 \models \mathsf{X}\,\psi \quad$ iff $\quad \pi_1[1..] \models \psi \quad$ iff $\quad \pi_2[1..] \models \psi \quad$ iff $\quad \pi_2 \models \mathsf{X}\,\psi$
- ...

Computing bisimulation quotients

The importance of this result

- CTL and CTL* equivalence coincide
  - despite the fact that CTL* is more expressive than CTL
- Bisimilar transition systems preserve the same CTL* formulas
  - and thus the same LTL formulas
- Non-bisimilarity can be shown by a single CTL (or CTL*) formula
  - $TS_1 \models \Phi$ and $TS_2 \not\models \Phi$ implies $TS_1 \not\sim TS_2$
- You even do not need to use an until-operator!
- To check $TS \models \Phi$, it suffices to check $TS/\!\sim \models \Phi$

Computing bisimulation quotients

- A **partition** $\Pi = \{B_1, \ldots, B_k\}$ of S is a set of nonempty ($B_i \neq \emptyset$) and pairwise disjoint **blocks** B_i that decompose S ($S = \biguplus_{i=1,\ldots k} B_i$).
- A **partition** Π defines an **equivalence relation** $\sim$ ($(q, q') \in \sim \Leftrightarrow \exists B_i \in \Pi.\ q, q' \in B_i$).
- Likewise, an **equivalence relation** $\sim$ defines a **partition** $\Pi = S/\!\sim$.
- A nonempty union $C = \biguplus_{i \in I} B_i$ of blocks is called a **superblock**.
- A block B_i of a partition Π is called **stable** w.r.t. a set B if either $B_i \cap \mathit{Pre}(B) = \emptyset$, or $B_i \subseteq \mathit{Pre}(B)$.
  ($\mathit{Pre}(B) = \{q \in S \mid \mathit{Post}(q) \cap B \neq \emptyset\}$)
- A partition Π is called **stable** w.r.t. a set B if all blocks of Π are stable w.r.t. B.

Stable partitions and bisimulation

Lemma 1. A partition Π with consistently labeled blocks is stable with respect to all of its (super)blocks iff it defines a bisimulation relation.

"$\Rightarrow$"

- Let $s_1 \sim s_2$, and $B = [s_1]_\Pi = [s_2]_\Pi$.
- Let $s_1' \in \mathit{Post}(s_1)$ and $C = [s_1']_\Pi$.
- Since $s_1 \in B \cap \mathit{Pre}(C)$, $B \subseteq \mathit{Pre}(C)$.
- Hence, $s_2 \in \mathit{Pre}(C)$.
- Hence, there is a state $s_2' \in \mathit{Post}(s_2) \cap C$.
- Since $s_2' \in C$, $s_2' \sim s_1'$.

Stable partitions and bisimulation (cont'd)

Lemma 1. A partition Π with consistently labeled blocks is stable with respect to all of its (super)blocks iff it defines a bisimulation relation.

"$\Leftarrow$"

- Let B, C be blocks of Π.
- We assume that $B \cap \mathit{Pre}(C) \neq \emptyset$ and show that $B \subseteq \mathit{Pre}(C)$.
- Since $B \cap \mathit{Pre}(C) \neq \emptyset$ there exists a state $s_1 \in B$ and a state $s_1' \in \mathit{Post}(s_1) \cap C$.
- Let s_2 be an arbitrary state in B. We show that $s_2 \in \mathit{Pre}(C)$.
- Since $\sim$ is a bisimulation, there exists a transition $s_2 \to s_2'$ such that $s_2' \in C$.
- Hence, $s_2 \in \mathit{Pre}(C)$.

Partition refinement

For two partitions $\Pi = \{B_1, \ldots, B_k\}$ and $\Pi' = \{B_1', \ldots, B_j'\}$ of S, we say that Π is **finer** than Π' iff every block of Π' is a superblock of Π.

For a given partition $\Pi = \{B_1, \ldots, B_k\}$, we call a (super)block C of Π a **splitter** of a block B_i / the partition Π if B_i / Π is not stable w.r.t. C.

$\mathrm{Refine}(B_i, C)$ denotes $\{B_i\}$ if B_i is **stable** w.r.t. C, and $\{B_i \cap \mathit{Pre}(C), B_i \setminus \mathit{Pre}(C)\}$ if C is a **splitter** of B_i.
$\mathrm{Refine}(\Pi, C) = \biguplus_{i=1,\ldots,k} \mathrm{Refine}(B_i, C)$.

Lemma 2. $\mathrm{Refine}(\Pi, C)$ is finer than Π.

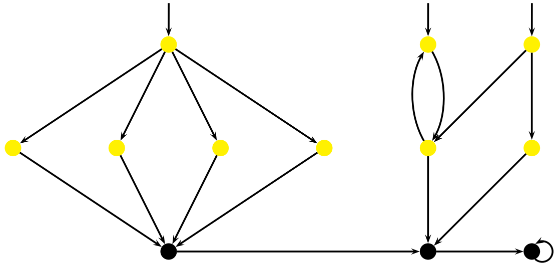
An algorithm for bisimulation quotienting

Input: Transition system $(S, \mathit{Act}, \to, I, AP, L)$
Output: Bisimulation quotient

1. $\Pi = S/{\sim_{AP}}$ $\qquad (q, q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π
   loop invariant: Π is coarser than $S/{\sim_{TS}}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \mathrm{Refine}(\Pi, B)$
3. return Π

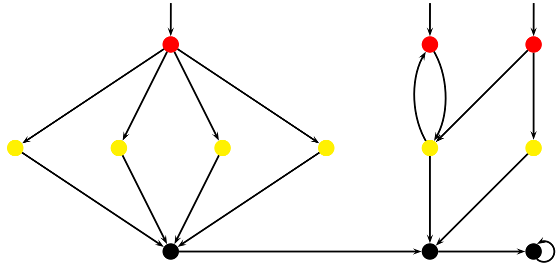
Example

1. $\Pi = S/{\sim_{AP}}$ $(q, q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/{\sim_{TS}}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π



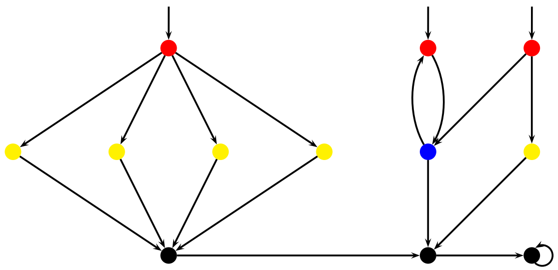
Example

1. $\Pi = S/{\sim_{AP}}$ $(q, q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/{\sim_{TS}}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π



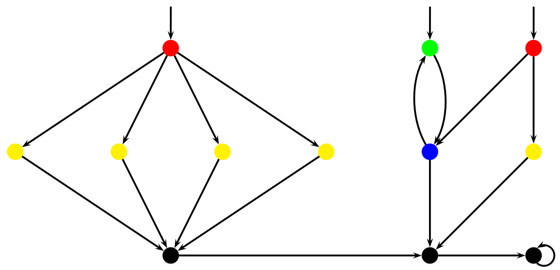
Example

1. $\Pi = S/{\sim_{AP}}$ $(q, q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/{\sim_{TS}}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π



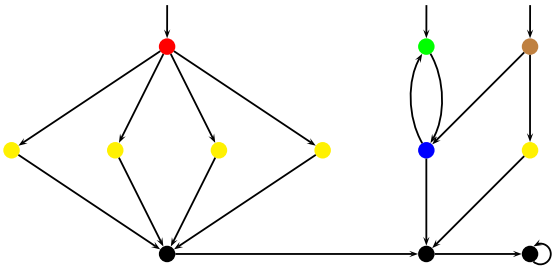
Example

1. $\Pi = S/{\sim_{AP}}$ $(q, q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/{\sim_{TS}}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π



Example

1. $\Pi = S/\sim_{AP}$ $(q, q') \in \sim_{AP} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/\sim_{TS}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π



Correctness and termination

1. $\Pi = S/\sim_{AP}$ $(q, q') \in \sim_{AP} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $S/\sim_{TS}$
   2.1 pick a block B that is a splitter of Π
   2.2 $\Pi = \text{Refine}(\Pi, B)$
3. return Π

Lemma 3. The algorithm terminates.
Lemma 4. The loop invariant holds initially.
Lemma 5. The loop invariant is preserved.

Theorem. The algorithm returns the quotient $S/\sim_{TS}$ of the coarsest bisimulation $\sim_{TS}$.