

# Concurrency Theory

Winter 2025/26

## Lecture 9: Properties of Hennessy-Milner Logic

Thomas Noll, Peter Thiemann  
Programming Languages Group  
University of Freiburg

<https://proglang.github.io/teaching/25ws/ct.html>

Thomas Noll, Peter Thiemann

Winter 2025/26

# Outline of Lecture 9

- 1 Closure under Negation
- 2 HML and Process Traces
- 3 HML and Strong Bisimilarity
- 4 HML and Weak Bisimilarity

# Closure under Negation I

**Observation:** **Negation** is *not* one of the HML constructs.

**Reason:** HML is **closed under complement**.

# Closure under Negation I

**Observation:** **Negation** is *not* one of the HML constructs.

**Reason:** HML is **closed under complement**.

## Lemma 9.1

*For every  $F \in \text{HMF}$  there exists  $F^c \in \text{HMF}$  such that  $\llbracket F^c \rrbracket = S \setminus \llbracket F \rrbracket$  for every LTS  $(S, \text{Act}, \longrightarrow)$ .*

# Closure under Negation I

**Observation:** **Negation** is *not* one of the HML constructs.

**Reason:** HML is **closed under complement**.

## Lemma 9.1

For every  $F \in \text{HMF}$  there exists  $F^c \in \text{HMF}$  such that  $\llbracket F^c \rrbracket = S \setminus \llbracket F \rrbracket$  for every LTS  $(S, \text{Act}, \longrightarrow)$ .

## Proof.

Definition of  $F^c$ :

$$\begin{array}{ll} \text{tt}^c := \text{ff} & \text{ff}^c := \text{tt} \\ (F_1 \wedge F_2)^c := F_1^c \vee F_2^c & (F_1 \vee F_2)^c := F_1^c \wedge F_2^c \\ (\langle \alpha \rangle F)^c := [\alpha] F^c & ([\alpha] F)^c := \langle \alpha \rangle F^c \end{array}$$

# Closure under Negation II

Proof (Lemma 9.1; continued).

We show  $\llbracket F^c \rrbracket = S \setminus \llbracket F \rrbracket$  by induction on the structure of  $F \in \mathbf{HMF}$ :

- $F = \mathbf{tt}$  ( $F = \mathbf{ff}$  analogously):

$$\llbracket F^c \rrbracket = \llbracket \mathbf{ff} \rrbracket \stackrel{\text{Def. 8.2}}{=} \emptyset = S \setminus S \stackrel{\text{Def. 8.2}}{=} S \setminus \llbracket \mathbf{tt} \rrbracket = S \setminus \llbracket F \rrbracket$$

# Closure under Negation II

Proof (Lemma 9.1; continued).

We show  $\llbracket F^c \rrbracket = S \setminus \llbracket F \rrbracket$  by induction on the structure of  $F \in \text{HMF}$ :

- $F = F_1 \wedge F_2$  ( $F = F_1 \vee F_2$  analogously):

$$\begin{aligned}\llbracket F^c \rrbracket &= \llbracket F_1^c \vee F_2^c \rrbracket \\ &\stackrel{\text{Def. 8.2}}{=} \llbracket F_1^c \rrbracket \cup \llbracket F_2^c \rrbracket \\ &\stackrel{\text{ind. hyp.}}{=} S \setminus \llbracket F_1 \rrbracket \cup S \setminus \llbracket F_2 \rrbracket \\ &\stackrel{\text{de Morgan}}{=} S \setminus (\llbracket F_1 \rrbracket \cap \llbracket F_2 \rrbracket) \\ &\stackrel{\text{Def. 8.2}}{=} S \setminus \llbracket F \rrbracket\end{aligned}$$

# Closure under Negation II

Proof (Lemma 9.1; continued).

We show  $\llbracket F^c \rrbracket = S \setminus \llbracket F \rrbracket$  by induction on the structure of  $F \in \mathbf{HMF}$ :

- $F = \langle \alpha \rangle F_0$  ( $F = [\alpha] F_0$  analogously):

$$\begin{aligned} \llbracket F^c \rrbracket &= \llbracket [\alpha] F_0^c \rrbracket \\ &\stackrel{\text{Def. 8.2}}{=} [\cdot \alpha \cdot](\llbracket F_0^c \rrbracket) \\ &\stackrel{\text{Def. 8.2}}{=} \{s \in S \mid \forall s' \in S : s \xrightarrow{\alpha} s' \Rightarrow s' \in \llbracket F_0^c \rrbracket\} \\ &\stackrel{\text{ind. hyp.}}{=} \{s \in S \mid \forall s' \in S : s \xrightarrow{\alpha} s' \Rightarrow s' \in S \setminus \llbracket F_0 \rrbracket\} \\ &= \{s \in S \mid \forall s' \in \llbracket F_0 \rrbracket : s \not\xrightarrow{\alpha} s'\} \\ &= S \setminus \{s \in S \mid \exists s' \in \llbracket F_0 \rrbracket : s \xrightarrow{\alpha} s'\} \\ &\stackrel{\text{Def. 8.2}}{=} S \setminus \langle \cdot \alpha \cdot \rangle(\llbracket F_0 \rrbracket) \\ &\stackrel{\text{Def. 8.2}}{=} S \setminus \llbracket F \rrbracket \end{aligned}$$





# Outline of Lecture 9

- 1 Closure under Negation
- 2 HML and Process Traces**
- 3 HML and Strong Bisimilarity
- 4 HML and Weak Bisimilarity

## Lemma 9.2 (HML and process traces)

Let  $(S, Act, \longrightarrow)$  be an LTS, and let  $s, t \in S$  satisfy the same HMF (i.e., for all  $F \in HMF$ :  $s \models F \iff t \models F$ ). Then  $Tr(s) = Tr(t)$ .

## Lemma 9.2 (HML and process traces)

Let  $(S, Act, \longrightarrow)$  be an LTS, and let  $s, t \in S$  satisfy the same HMF (i.e., for all  $F \in HMF$ :  $s \models F \iff t \models F$ ). Then  $Tr(s) = Tr(t)$ .

## Proof.

Let  $s, t \in S$  such that for all  $F \in HMF$ :  $s \models F \iff t \models F$ .

Assumption:  $Tr(s) \neq Tr(t)$ .

Then there exists  $n \geq 1$  and  $w = \alpha_1 \dots \alpha_n \in Act^+$  with  $w \in Tr(s) \setminus Tr(t)$  (or vice versa).

Hence, for  $F := \langle \alpha_1 \rangle \dots \langle \alpha_n \rangle tt \in HMF$ :  $s \models F$  but  $t \not\models F$ .  $\downarrow$



**Remark:** The converse does *not* hold.

## Example 9.3

- Let
  - $P := a.(b.\text{nil} + c.\text{nil}) \in \text{Prc}$  and
  - $Q := a.b.\text{nil} + a.c.\text{nil} \in \text{Prc}$ .
- Then  $\text{Tr}(P) = \text{Tr}(Q) = \{\varepsilon, a, ab, ac\}$ .

**Remark:** The converse does *not* hold.

## Example 9.3

- Let
  - $P := a.(b.\text{nil} + c.\text{nil}) \in \text{Prc}$  and
  - $Q := a.b.\text{nil} + a.c.\text{nil} \in \text{Prc}$ .
- Then  $\text{Tr}(P) = \text{Tr}(Q) = \{\varepsilon, a, ab, ac\}$ .
- Let  $F := [a](\langle b \rangle \text{tt} \wedge \langle c \rangle \text{tt}) \in \text{HMF}$ .
- Then  $P \models F$  but  $Q \not\models F$ .

**Remark:** The converse does *not* hold.

## Example 9.3

- Let
  - $P := a.(b.\text{nil} + c.\text{nil}) \in \text{Prc}$  and
  - $Q := a.b.\text{nil} + a.c.\text{nil} \in \text{Prc}$ .
- Then  $\text{Tr}(P) = \text{Tr}(Q) = \{\varepsilon, a, ab, ac\}$ .
- Let  $F := [a](\langle b \rangle \text{tt} \wedge \langle c \rangle \text{tt}) \in \text{HMF}$ .
- Then  $P \models F$  but  $Q \not\models F$ .
- Thus: HML can distinguish **branching behaviour** of processes (just as bisimulation can...).

# Outline of Lecture 9

- 1 Closure under Negation
- 2 HML and Process Traces
- 3 HML and Strong Bisimilarity
- 4 HML and Weak Bisimilarity

# Strong Bisimilarity and HML

- Strong bisimilarity (and observation congruence) are based on mutual mimicking of processes.
- They possess the required properties of behavioural equivalences.
- In particular,  $\sim$  and  $\approx^c$  are deadlock-sensitive CCS congruences.
- Hennessy-Milner Logic (HML) is a logic for expressing properties of processes.



# Strong Bisimilarity and HML

- Strong bisimilarity (and observation congruence) are based on mutual mimicking of processes.
- They possess the required properties of behavioural equivalences.
- In particular,  $\sim$  and  $\approx^c$  are deadlock-sensitive CCS congruences.
- Hennessy-Milner Logic (HML) is a logic for expressing properties of processes.

## Aim

Study the connection between strong bisimilarity and satisfaction of HML formulae.

# Finitely Branching Transition Systems

## Definition 9.4 (Finitely branching LTS)

- A process  $P \in \text{Prc}$  is **finitely branching** if the set  $\{P' \in \text{Prc} \mid P \xrightarrow{\alpha} P'\}$  is finite for every  $\alpha \in \text{Act}$ .
- A labelled transition system is **finitely branching** if each state is finitely branching.

# Finitely Branching Transition Systems

## Definition 9.4 (Finitely branching LTS)

- A process  $P \in \text{Prc}$  is **finitely branching** if the set  $\{P' \in \text{Prc} \mid P \xrightarrow{\alpha} P'\}$  is finite for every  $\alpha \in \text{Act}$ .
- A labelled transition system is **finitely branching** if each state is finitely branching.

## Example 9.5

- (1) The process  $A_{rep} = a.nil \parallel A_{rep}$  (“ $A$  replicated”) is not finitely branching.  
By induction on  $n$ , one can prove that for each  $n \in \mathbb{N}$ :

$$A_{rep} \xrightarrow{a} \underbrace{a.nil \parallel \dots \parallel a.nil}_{n \text{ times}} \parallel nil \parallel A_{rep}$$

# Finitely Branching Transition Systems

## Definition 9.4 (Finitely branching LTS)

- A process  $P \in \text{Prc}$  is **finitely branching** if the set  $\{P' \in \text{Prc} \mid P \xrightarrow{\alpha} P'\}$  is finite for every  $\alpha \in \text{Act}$ .
- A labelled transition system is **finitely branching** if each state is finitely branching.

## Example 9.5

- (1) The process  $A_{\text{rep}} = a.\text{nil} \parallel A_{\text{rep}}$  (“ $A$  replicated”) is not finitely branching.  
By induction on  $n$ , one can prove that for each  $n \in \mathbb{N}$ :

$$A_{\text{rep}} \xrightarrow{a} \underbrace{a.\text{nil} \parallel \cdots \parallel a.\text{nil}}_{n \text{ times}} \parallel \text{nil} \parallel A_{\text{rep}}$$

- (2) Also the “process”  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  with  $a^0 = \text{nil}$  and  $a^{i+1} = a.a^i$  is not finitely branching:  
for each  $i \geq 1$ ,  $A^{<\omega} \xrightarrow{a} a^{i-1}$ .

# Relationship Between HML and Strong Bisimilarity I

## Theorem 9.6 (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF : (s \models F \iff t \models F)$ .

# Relationship Between HML and Strong Bisimilarity I

## Theorem 9.6 (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF$  :  $(s \models F \iff t \models F)$ .

## Proof.

“ $\Rightarrow$ ”: Assume  $s \sim t$  and  $s \models F$  for some  $F \in HMF$ .  
We show  $t \models F$  by structural induction on  $F$ .

# Relationship Between HML and Strong Bisimilarity I

## Theorem 9.6 (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF : (s \models F \iff t \models F)$ .

## Proof.

“ $\Rightarrow$ ”: Assume  $s \sim t$  and  $s \models F$  for some  $F \in HMF$ .

We show  $t \models F$  by structural induction on  $F$ . Interesting cases:

- $F = \langle \alpha \rangle F'$ :
  - Since  $s \models F$ , there ex.  $s' \in S$  such that  $s \xrightarrow{\alpha} s'$  and  $s' \models F'$ .
  - Since  $s \sim t$ , there ex.  $t' \in S$  such that  $t \xrightarrow{\alpha} t'$  and  $s' \sim t'$ .
  - By induction hypothesis,  $t' \models F'$ . Thus,  $t \models \langle \alpha \rangle F' = F$ .

# Relationship Between HML and Strong Bisimilarity I

## Theorem 9.6 (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF : (s \models F \iff t \models F)$ .

## Proof.

“ $\Rightarrow$ ”: Assume  $s \sim t$  and  $s \models F$  for some  $F \in HMF$ .

We show  $t \models F$  by structural induction on  $F$ . Interesting cases:

- $F = \langle \alpha \rangle F'$ :
  - Since  $s \models F$ , there ex.  $s' \in S$  such that  $s \xrightarrow{\alpha} s'$  and  $s' \models F'$ .
  - Since  $s \sim t$ , there ex.  $t' \in S$  such that  $t \xrightarrow{\alpha} t'$  and  $s' \sim t'$ .
  - By induction hypothesis,  $t' \models F'$ . Thus,  $t \models \langle \alpha \rangle F' = F$ .
- $F = [\alpha] F'$ : Assume that  $t \xrightarrow{\alpha} t'$  for some  $t' \in S$ .
  - Since  $s \sim t$ , there ex.  $s' \in S$  such that  $s \xrightarrow{\alpha} s'$  and  $s' \sim t'$ .
  - Since  $s \models [\alpha] F'$ , also  $s' \models F'$ .
  - By induction hypothesis,  $t' \models F'$ . Thus,  $t \models [\alpha] F' = F$ .



# Relationship Between HML and Strong Bisimilarity II

## Theorem (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF$  :  $(s \models F \iff t \models F)$ .

## Proof.

“ $\Leftarrow$ ”: Define  $u \equiv v$  iff  $(u \models F \iff v \models F)$  for every  $F \in HMF$ , and let  $s \equiv t$ .

# Relationship Between HML and Strong Bisimilarity II

## Theorem (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF : (s \models F \iff t \models F)$ .

## Proof.

“ $\Leftarrow$ ”: Define  $u \equiv v$  iff  $(u \models F \iff v \models F)$  for every  $F \in HMF$ , and let  $s \equiv t$ .

We prove  $s \sim t$  by showing that  $\equiv$  is a strong bisimulation.

To this aim, let  $u \equiv v$  and  $u \xrightarrow{\alpha} u'$  for some  $u' \in S$ .

We have to show that ex.  $v' \in S$  with  $v \xrightarrow{\alpha} v'$  and  $u' \equiv v'$ .

# Relationship Between HML and Strong Bisimilarity II

## Theorem (Hennessy-Milner Theorem)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:  
 $s \sim t$  iff for every  $F \in HMF$  :  $(s \models F \iff t \models F)$ .

## Proof.

“ $\Leftarrow$ ”: Define  $u \equiv v$  iff  $(u \models F \iff v \models F)$  for every  $F \in HMF$ , and let  $s \equiv t$ .

We prove  $s \sim t$  by showing that  $\equiv$  is a strong bisimulation.

To this aim, let  $u \equiv v$  and  $u \xrightarrow{\alpha} u'$  for some  $u' \in S$ .

We have to show that ex.  $v' \in S$  with  $v \xrightarrow{\alpha} v'$  and  $u' \equiv v'$ .

- Assume that there is no such  $v'$ .
- Let  $\{v' \in S \mid v \xrightarrow{\alpha} v'\} = \{v'_1, \dots, v'_n\}$  with  $n \in \mathbb{N}$  (finitely branching!).
- By the previous assumption,  $u' \not\equiv v'_i$  for each  $i \in [n]$ .
- Thus, for each  $i \in [n]$  there ex.  $F_i \in HMF$  with  $u' \models F_i$  and  $v'_i \not\models F_i$ .  
(If  $u' \not\models F_i$  and  $v'_i \models F_i$ , then consider  $F_i^c$  [Lemma 9.1] instead.)
- Hence, for  $G := \langle \alpha \rangle (F_1 \wedge \dots \wedge F_n)$ :  $u \models G$  but  $v \not\models G$ , which contradicts  $u \equiv v$ . □

# Proving Non-Bisimilarity

## Proving non-bisimilarity

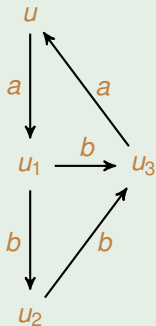
Showing  $P \not\sim Q$  thus amounts to finding a single HML-formula  $F$  with  $P \models F$  and  $Q \not\models F$ .

# Proving Non-Bisimilarity

## Proving non-bisimilarity

Showing  $P \not\sim Q$  thus amounts to finding a single HML-formula  $F$  with  $P \models F$  and  $Q \not\models F$ .

### Example 9.7

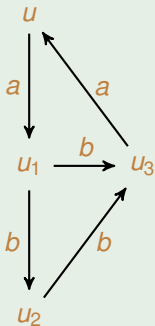


# Proving Non-Bisimilarity

## Proving non-bisimilarity

Showing  $P \not\sim Q$  thus amounts to finding a single HML-formula  $F$  with  $P \models F$  and  $Q \not\models F$ .

### Example 9.7



#### Distinguishing formulae

(satisfied by row state / violated by column state):

	$u$	$u_1$	$u_2$	$u_3$
$u$	—	$\langle a \rangle \text{tt}$	$\langle a \rangle \text{tt}$	$\langle a \rangle \langle b \rangle \text{tt}$
$u_1$	$[a] \text{ff}$	—	—	$[a] \text{ff}$
$u_2$	$[a] \text{ff}$	—	—	$[a] \text{ff}$
$u_3$	$\langle a \rangle \langle a \rangle \text{tt}$	$\langle a \rangle \text{tt}$	$\langle a \rangle \text{tt}$	—

# Counterexample for Non-Finitely Branching Processes

## Lemma 9.8

Let  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  (see Example 9.5) and  $A^\omega = a.A^\omega$ . Then  $A^{<\omega}$  and  $A^{<\omega} + A^\omega$

- (1) are not strongly bisimilar, but
- (2) satisfy the same HML formulae.

# Counterexample for Non-Finitely Branching Processes

## Lemma 9.8

Let  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  (see Example 9.5) and  $A^\omega = a.A^\omega$ . Then  $A^{<\omega}$  and  $A^{<\omega} + A^\omega$

- (1) are not strongly bisimilar, but
- (2) satisfy the same HML formulae.

## Proof.

- (1) Assume that  $A^{<\omega} \sim A^{<\omega} + A^\omega$ . Then  $A^{<\omega} + A^\omega \xrightarrow{a} A^\omega$  must be mimicked by  $A^{<\omega} \xrightarrow{a} a^{i-1}$  for some  $i \geq 1$ . But obviously  $A^\omega \not\sim a^{i-1}$ .



# Counterexample for Non-Finitely Branching Processes

## Lemma 9.8

Let  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  (see Example 9.5) and  $A^\omega = a.A^\omega$ . Then  $A^{<\omega}$  and  $A^{<\omega} + A^\omega$

- (1) are not strongly bisimilar, but
- (2) satisfy the same HML formulae.

## Proof.

- (1) Assume that  $A^{<\omega} \sim A^{<\omega} + A^\omega$ . Then  $A^{<\omega} + A^\omega \xrightarrow{a} A^\omega$  must be mimicked by  $A^{<\omega} \xrightarrow{a} a^{i-1}$  for some  $i \geq 1$ . But obviously  $A^\omega \not\sim a^{i-1}$ .
- (2) By structural induction on  $F \in \text{HML}$ , using the following lemma. □

# Counterexample for Non-Finitely Branching Processes

## Lemma 9.8

Let  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  (see Example 9.5) and  $A^\omega = a.A^\omega$ . Then  $A^{<\omega}$  and  $A^{<\omega} + A^\omega$

- (1) are not strongly bisimilar, but
- (2) satisfy the same HML formulae.

## Proof.

- (1) Assume that  $A^{<\omega} \sim A^{<\omega} + A^\omega$ . Then  $A^{<\omega} + A^\omega \xrightarrow{a} A^\omega$  must be mimicked by  $A^{<\omega} \xrightarrow{a} a^{i-1}$  for some  $i \geq 1$ . But obviously  $A^\omega \not\sim a^{i-1}$ .
- (2) By structural induction on  $F \in \text{HMF}$ , using the following lemma. □

## Lemma 9.9

For every  $F \in \text{HMF}$ ,  $A^\omega \models F$  iff  $a^k \models F$ , where  $k$  is the modal depth<sup>a</sup> of  $F$ .

---

<sup>a</sup>the maximal number of nested occurrences of modal operators in  $F$

# Counterexample for Non-Finitely Branching Processes

## Lemma 9.8

Let  $A^{<\omega} = \sum_{i \in \mathbb{N}} a^i$  (see Example 9.5) and  $A^\omega = a.A^\omega$ . Then  $A^{<\omega}$  and  $A^{<\omega} + A^\omega$

- (1) are not strongly bisimilar, but
- (2) satisfy the same HML formulae.

## Proof.

- (1) Assume that  $A^{<\omega} \sim A^{<\omega} + A^\omega$ . Then  $A^{<\omega} + A^\omega \xrightarrow{a} A^\omega$  must be mimicked by  $A^{<\omega} \xrightarrow{a} a^{i-1}$  for some  $i \geq 1$ . But obviously  $A^\omega \not\sim a^{i-1}$ .
- (2) By structural induction on  $F \in \text{HMF}$ , using the following lemma. □

## Lemma 9.9

For every  $F \in \text{HMF}$ ,  $A^\omega \models F$  iff  $a^k \models F$ , where  $k$  is the modal depth<sup>a</sup> of  $F$ .

<sup>a</sup>the maximal number of nested occurrences of modal operators in  $F$

# Outline of Lecture 9

- 1 Closure under Negation
- 2 HML and Process Traces
- 3 HML and Strong Bisimilarity
- 4 HML and Weak Bisimilarity

# Recap: Weak Bisimulation

Definition (Weak transition relation; Definition 5.8)

For  $\alpha \in \text{Act}$ ,  $\Longrightarrow_\alpha \subseteq \text{Prc} \times \text{Prc}$  is given by

$$\Longrightarrow_\alpha := \begin{cases} \left( \xrightarrow{\tau} \right)^* \circ \xrightarrow{\alpha} \circ \left( \xrightarrow{\tau} \right)^* & \text{if } \alpha \neq \tau \\ \left( \xrightarrow{\tau} \right)^* & \text{if } \alpha = \tau. \end{cases}$$

where  $\left( \xrightarrow{\tau} \right)^*$  denotes the reflexive and transitive closure of relation  $\xrightarrow{\tau}$ .

Definition (Weak bisimulation; Definition 5.9)

(Milner 1989)

A binary relation  $\rho \subseteq \text{Prc} \times \text{Prc}$  is a **weak bisimulation** whenever for every  $(P, Q) \in \rho$  and  $\alpha \in \text{Act}$  (including  $\alpha = \tau$ ):

- (1) if  $P \xrightarrow{\alpha} P'$ , then there exists  $Q' \in \text{Prc}$  such that  $Q \Longrightarrow_\alpha Q'$  and  $P' \rho Q'$ , and
- (2) if  $Q \xrightarrow{\alpha} Q'$ , then there exists  $P' \in \text{Prc}$  such that  $P \Longrightarrow_\alpha P'$  and  $P' \rho Q'$ .

# Recap: Weak Bisimulation

Definition (Weak transition relation; Definition 5.8)

For  $\alpha \in \text{Act}$ ,  $\Longrightarrow_{\alpha} \subseteq \text{Prc} \times \text{Prc}$  is given by

$$\Longrightarrow_{\alpha} := \begin{cases} \left( \xrightarrow{\tau} \right)^* \circ \xrightarrow{\alpha} \circ \left( \xrightarrow{\tau} \right)^* & \text{if } \alpha \neq \tau \\ \left( \xrightarrow{\tau} \right)^* & \text{if } \alpha = \tau. \end{cases}$$

where  $\left( \xrightarrow{\tau} \right)^*$  denotes the reflexive and transitive closure of relation  $\xrightarrow{\tau}$ .

Definition (Weak bisimulation; Definition 5.9)

(Milner 1989)

A binary relation  $\rho \subseteq \text{Prc} \times \text{Prc}$  is a **weak bisimulation** whenever for every  $(P, Q) \in \rho$  and  $\alpha \in \text{Act}$  (including  $\alpha = \tau$ ):

- (1) if  $P \xrightarrow{\alpha} P'$ , then there exists  $Q' \in \text{Prc}$  such that  $Q \Longrightarrow_{\alpha} Q'$  and  $P' \rho Q'$ , and
- (2) if  $Q \xrightarrow{\alpha} Q'$ , then there exists  $P' \in \text{Prc}$  such that  $P \Longrightarrow_{\alpha} P'$  and  $P' \rho Q'$ .

# Introducing Weak Modalities

**Goal:** Modify HML by turning strong modal operators into weak ones  
(see J. Parrow et al.: *Weak Nominal Modal Logic*, FORTE 2017 )

# Introducing Weak Modalities

## Definition 9.10 (Syntax and semantics of weak modalities)

- $wHMF$  is obtained from  $HMF$  of (Definition 8.1) by replacing  $\langle \alpha \rangle$  and  $[\alpha]$  with

$$\langle \langle \alpha \rangle \rangle F \quad \text{and} \quad [[\alpha]] F$$

for  $\alpha \in Act$  and  $F \in wHMF$ .



# Introducing Weak Modalities

## Definition 9.10 (Syntax and semantics of weak modalities)

- $wHMF$  is obtained from  $HMF$  of (Definition 8.1) by replacing  $\langle \alpha \rangle$  and  $[\alpha]$  with

$$\langle\langle \alpha \rangle\rangle F \quad \text{and} \quad [[\alpha]] F$$

for  $\alpha \in Act$  and  $F \in wHMF$ .

- Modifying Definition 8.2, for an LTS  $(S, Act, \longrightarrow)$  and  $F \in wHMF$  we let

$$\llbracket \langle\langle \alpha \rangle\rangle F \rrbracket := \langle\langle \cdot \alpha \cdot \rangle\rangle (\llbracket F \rrbracket) \qquad \llbracket [[\alpha]] F \rrbracket := [[\cdot \alpha \cdot]] (\llbracket F \rrbracket)$$

where  $\langle\langle \cdot \alpha \cdot \rangle\rangle, [[\cdot \alpha \cdot]] : 2^S \rightarrow 2^S$  are given by

$$\begin{aligned} \langle\langle \cdot \alpha \cdot \rangle\rangle (T) &:= \{s \in S \mid \exists s' \in T : s \xrightarrow{\alpha} s'\} \\ [[\cdot \alpha \cdot]] (T) &:= \{s \in S \mid \forall s' \in S : s \xrightarrow{\alpha} s' \text{ implies } s' \in T\} \end{aligned}$$

Again, we write  $s \models F$  iff  $s \in \llbracket F \rrbracket$ , and  $F, G \in wHMF$  are **equivalent** (written  $F \equiv G$ ) iff they are satisfied by the same processes in every LTS.

# Relationship Between HML with Weak Modalities and Weak Bisimilarity

## Theorem 9.11 (Hennessy-Milner Theorem for weak bisimulation)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:

$$s \approx t \quad \text{iff} \quad \text{for every } F \in wHMF : (s \models F \iff t \models F).$$

## Proof.

see J. Parrow et al.: *Weak Nominal Modal Logic*, FORTE 2017



# Relationship Between HML with Weak Modalities and Weak Bisimilarity

## Theorem 9.11 (Hennessy-Milner Theorem for weak bisimulation)

Let  $(S, Act, \{\xrightarrow{a} \mid a \in Act\})$  be a finitely branching LTS and  $s, t \in S$ . Then:

$$s \approx t \quad \text{iff} \quad \text{for every } F \in wHMF : (s \models F \iff t \models F).$$

## Proof.

see J. Parrow et al.: *Weak Nominal Modal Logic*, FORTE 2017



## Example 9.12 (Counterexample to congruence of weak bisimilarity)

- Lecture 6:  $\tau.a.\text{nil} + b.\text{nil} \not\approx a.\text{nil} + b.\text{nil}$
- Distinguishing *wHMF* formula:  $\langle\langle\tau\rangle\rangle[[b]]\text{ff}$
- Check using CAAL