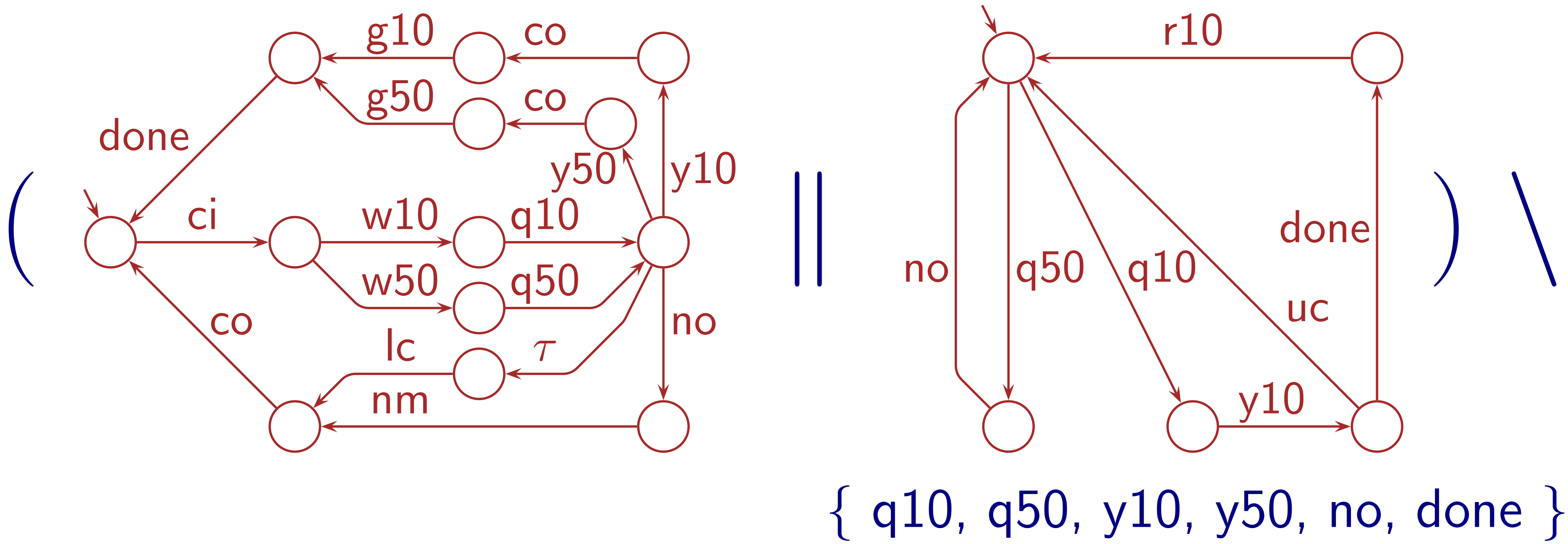
All Linear-Time Congruences for Familiar Operators

Part I: Finite LTSs **ACSD 2012, Hamburg**

Part II: Infinite LTSs **CONCUR 2012, Newcastle**

Antti Valmari
Tampere University of Technology, FINLAND

1 Systems and Congruences



Systems

- labelled transition systems (LTS): τ, $(S, \Sigma, \Delta, \hat{s})$
- operators for composing systems from LTSs: (later)

Notion of "equivalent behaviour"

- numerous in the literature

Congruence

$$L_1 \cong L_1' \wedge \cdots \wedge L_n \cong L_n' \;\Rightarrow\; f(L_1, \ldots, L_n) \cong f(L_1', \ldots, L_n')$$

2 Why *All* Congruences (in Some Region)?

Equivalences in verification

- let φ be deadlock-freedom or guarantee of eventual service or ...
- "$\cong$" *preserves* φ if and only if for every L and L',
  $L \cong L'$ implies that either none or both of L and L' satisfy φ
- if "$\cong$" preserves φ, L is complicated, L' is simple, and $L \cong L'$, then
  it is correct and advantageous to check $L \models \varphi$ by checking $L' \models \varphi$

Compositional methods

$$L \;=\; (\;(L_1 \| L_2) \backslash A_{12} \;\|\; (L_3 \| L_4) \backslash A_{34}\;) \setminus A$$
$$L' \;=\; \mathrm{reduce}(\;(\;\mathrm{reduce}((L_1 \| L_2) \backslash A_{12}) \;\|\; \mathrm{reduce}((L_3 \| L_4) \backslash A_{34})\;) \setminus A\;)$$

- reduce preserves "$\cong$", "$\cong$" must be a congruence
- many advanced variants exist
- the weaker (i.e., coarser) "$\cong$" is, the better are reduction results

So the *weakest* congruence that preserves φ gives best reduction results

But weakest congruences are hard to find!

- what is the weakest congruence that distinguishes $\circ \xrightarrow{a} \circ$ from $\circ \xleftarrow{\tau} \circ \xrightarrow{a} \circ$?

Another reason: curiousity

3 Which Operators? (1/2)

More operators $\Rightarrow$ fewer (or the same) congruences

- the fewer operators we use, the stronger are our results ...
- ... but we need enough operators for the proofs to go through
- use of common operators is justified

Parallel composition $L_1 \| L_2$

- Application of Concurrency to System Design, Concurrency Theory
- we use this variant:

$$\frac{L_1 -a\rightarrow M_1 \wedge a \notin \Sigma_2}{L_1 \| L_2 -a\rightarrow M_1 \| L_2}, \text{ 1-2-symmetric, } \frac{L_1 -a\rightarrow M_1 \wedge L_2 -a\rightarrow M_2 \wedge a \neq \tau}{L_1 \| L_2 -a\rightarrow M_1 \| M_2}$$

- associative and commutative
- allows 3-way synchronization
- simplest in a complexity-theoretic sense (Valmari & Kervinen Concur 2002)

Hiding $L \backslash A$

$$\frac{L -a\rightarrow M \wedge a \notin A}{L \backslash A -a\rightarrow M \backslash A} \qquad \frac{L -a\rightarrow M \wedge a \in A}{L \backslash A -\tau\rightarrow M \backslash A}$$

- important for LTS reduction

3 Which Operators? (2/2)

Relational renaming (multiple renaming) $L\Phi$

- Φ is any set of pairs (a, b) such that $a \neq \tau \neq b$

$$\frac{L \xrightarrow{a} M \wedge (a,b) \in \Phi}{L\Phi \xrightarrow{b} M\Phi} \qquad \frac{L \xrightarrow{a} M \wedge \forall b : (a,b) \notin \Phi}{L\Phi \xrightarrow{a} M\Phi}$$

- all non-τ actions should be equal $\Rightarrow$ functional renaming is natural to require
- CSP has relational renaming
- simulation of CCS $|$ uses relational renaming
- proofs of some ACSD results use relational renaming
- the alphabet result provably needs relational renaming

Action prefix $a.L$

$$\frac{a \neq \tau}{a.L \xrightarrow{a} L}$$

- $\tau.L$ is obtained as $(a.L) \setminus \{a\}$, where $\tau \neq a \notin \Sigma(L)$
- some results provably need action prefix

No choice, no interrupt

- future work (spoiler: ...)

4 The Alphabet Result (ACSD)

The *dullest congruence* is the one that has $L \cong L'$ for every L and L'

We only consider congruences that are implied by (strong) bisimilarity "$\equiv$"

- very weak and acceptable assumption
- otherwise we would have clearly irrelevant congruences such as $L \cong L'$ if and only if their initial states have the same name

Theorem All other congruences than the dullest preserve Σ.

Proof

- if "$\cong$" does not preserve Σ, there are $M_1 \cong M_2$ and a such that $a \in \Sigma_1 \setminus \Sigma_2$
- let $f(M_i) = (c.M_i \| \mathord{\searrow\!\circ}^{\{c\}}) \setminus (\{c\} \cup \Sigma_2 \cup \Sigma_1 \setminus \{a\})$, where $\tau \neq c \neq a$
- we have $\mathord{\searrow\!\circ}^{\{a\}} \equiv f(M_1) \cong f(M_2) \equiv \mathord{\searrow\!\circ}^{\emptyset}$
- $\Phi = \{(a,b) \mid b \in \Sigma\}$ yields $\mathord{\searrow\!\circ}^{\Sigma} \equiv \mathord{\searrow\!\circ}^{\{a\}}\Phi \cong \mathord{\searrow\!\circ}^{\emptyset}\Phi \equiv \mathord{\searrow\!\circ}^{\emptyset}$
- for any L, let L' be τ-part of L and L'' be $L'[\frac{a}{\tau}]$, then $L \equiv L \| \mathord{\searrow\!\circ}^{\emptyset} \cong L \| \mathord{\searrow\!\circ}^{\Sigma} \equiv L' \| \mathord{\searrow\!\circ}^{\Sigma} \cong L' \| \mathord{\searrow\!\circ}^{\emptyset} \equiv L' = L'' \setminus \{a\} \equiv (L'' \| \mathord{\searrow\!\circ}^{\emptyset}) \setminus \{a\} \cong (L'' \| \mathord{\searrow\!\circ}^{\{a\}}) \setminus \{a\} \equiv \mathord{\searrow\!\circ}^{\emptyset}$

Without relational renaming, the result would not hold (Concur)

- the following would be a congruence: $L \cong L'$ if and only if both $\Sigma(L) \setminus \Sigma(L')$ and $\Sigma(L') \setminus \Sigma(L)$ are finite

5 Linear Time (As We Use the Term)

Stuttering-insensitive linear temporal logic (Manna & Pnueli 1992)

- *stuttering-insensitive*
  $\Rightarrow \tau$ is not directly observable
- observations on *complete* executions
  - *infinite traces* $\mathit{Inf}(L) \;=\; \{\xi \in \Sigma^\omega \mid \hat{s} =\xi\Rightarrow\}$
  - *deadlocking* and *divergence traces* $D\ell(L) \cup \mathit{Div}(L)$

$$\mathit{Div}(L) \;=\; \{\sigma \in \Sigma^* \mid \exists s : \hat{s} =\sigma\Rightarrow s \wedge s -\tau^\omega\rightarrow\}$$

- we strengthen a bit assuming deadlock can be distinguished from divergence

Congruence w.r.t. $\|$ implies that *stable failures* must be preserved

$$\mathit{Sf}(L) \;=\; \{(\sigma, A) \in \Sigma^* \times 2^\Sigma \mid \exists s : \hat{s} =\sigma\Rightarrow s \wedge \forall a \in A \cup \{\tau\} : \neg(s -a\rightarrow)\}$$

So we define *the strongest abstract linear-time congruence* by $L \doteq L'$ if and only if

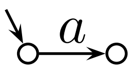
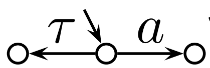
$$\Sigma(L) = \Sigma(L'),\ \mathit{Sf}(L) = \mathit{Sf}(L'),\ \mathit{Div}(L) = \mathit{Div}(L'),\ \text{and } \mathit{Inf}(L) = \mathit{Inf}(L')$$

- also called *Chaos-Free Failures Divergences Equivalence* or *CFFD*
- $D\ell$ is not needed, because $D\ell(L) = \{\sigma \mid (\sigma, \Sigma) \in \mathit{Sf}(L)\}$
- furthermore, $\mathit{Tr}(L) = \mathit{Div}(L) \cup \{\sigma \mid (\sigma, \emptyset) \in \mathit{Sf}(L)\}$
- if L is finite, then $\mathit{Inf}(L) = \{a_1 a_2 \cdots \in \Sigma^\omega \mid \forall i : a_1 \cdots a_i \in \mathit{Tr}(L)\}$

6 The Results

The picture shows *all* congruences that are weaker than or the same as "$\doteq$" (i.e., CFFD)

- path from "$\cong_1$" down to "$\cong_2$" means that "$\cong_1$" is stronger than "$\cong_2$"
- find Tr, CSP, NDFD, nF

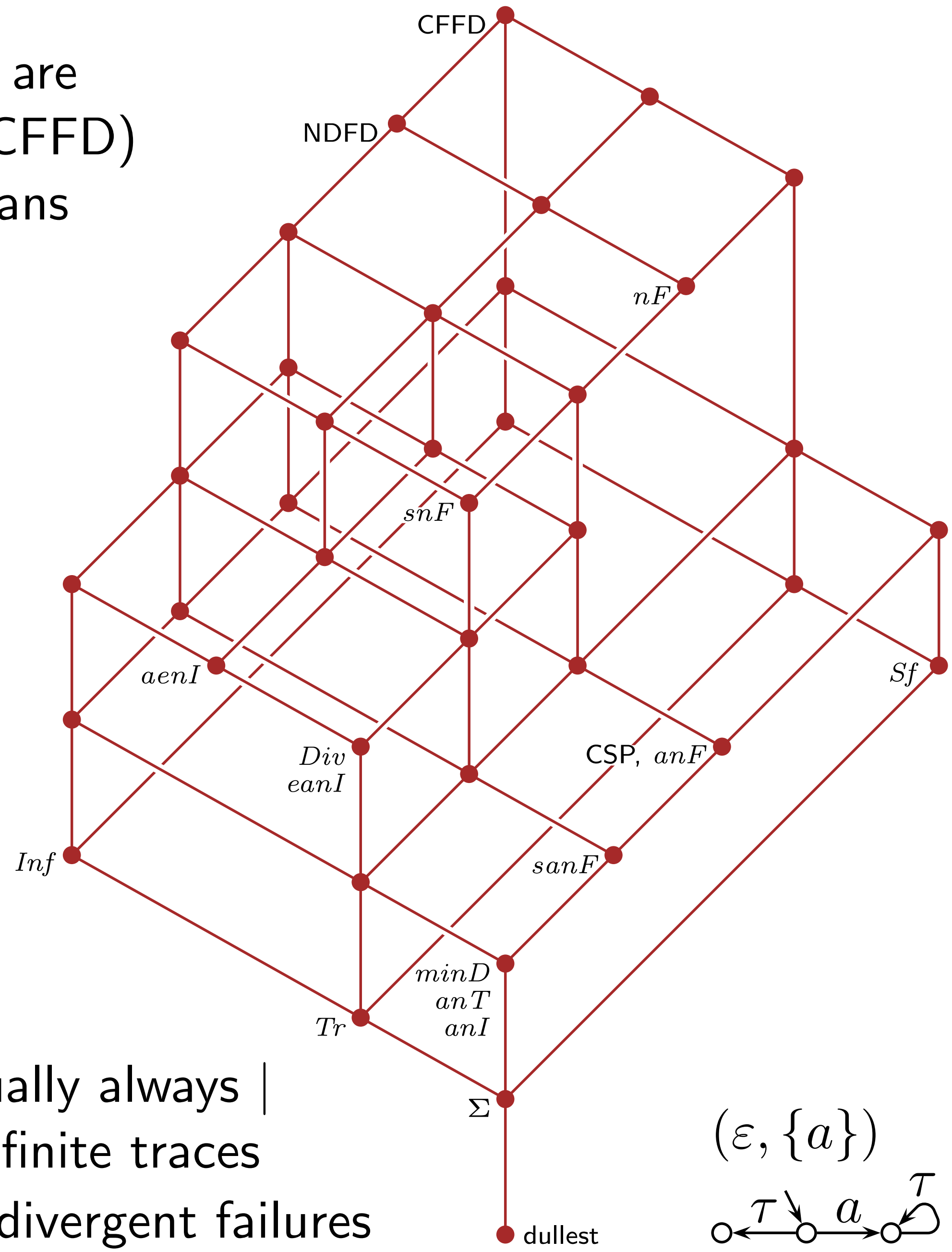
What is the weakest that distinguishes  from ?

- Σ-Tr-Div-Inf does not
- Σ-Sf does
- Σ-$sanF$-…does

$\Rightarrow$ two solutions

What are $minD$, anT, and so on?

- $minD$: minimal divergence traces
- anT: always nondivergent traces
- anI, $eanI$, $aenI$: (always | eventually always | always eventually) nondivergent infinite traces
- $[s][a]nF$: [strongly][always] nondivergent failures



$(\varepsilon, \{a\})$

7 Proof Technique (ACSD)

Lemma If (for every L)

- "$\cong$" is an equivalence
- "$\doteq$" implies "$\cong$"
- "$\cong$" preserves Σ and $X_1, \ldots, X_k$
- $L \cong f(L)$
- $\mathit{Sf}(f(L))$, $\mathit{Div}(f(L))$, $\mathit{Inf}(f(L))$ are functions of $\Sigma(L)$, $X_1(L)$, ..., $X_k(L)$

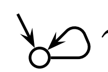
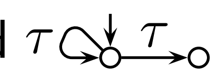
then $L \cong L' \;\Leftrightarrow\; \Sigma(L) = \Sigma(L') \wedge X_1(L) = X_1(L') \wedge \cdots \wedge X_k(L) = X_k(L')$.

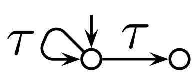
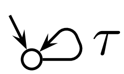
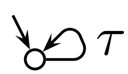
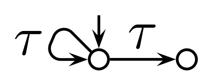
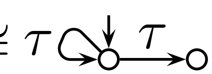
Proof

$\Rightarrow$: immediate

$\Leftarrow$: $L \cong f(L) \doteq f(L') \cong L'$, because "$\doteq$" of $f(L)$ only depends on $\Sigma(L)$, etc.

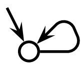
Second trick

- if $\Sigma = \emptyset$, there are only three "$\doteq$"-equivalence classes: , , and 
- study in turn each of the cases  $\cong$ 

   $\cong$  $\not\cong$   $\not\cong$  $\not\cong$  $\not\cong$   $\not\cong$  $\cong$ 

8 Example Proof (Concur)

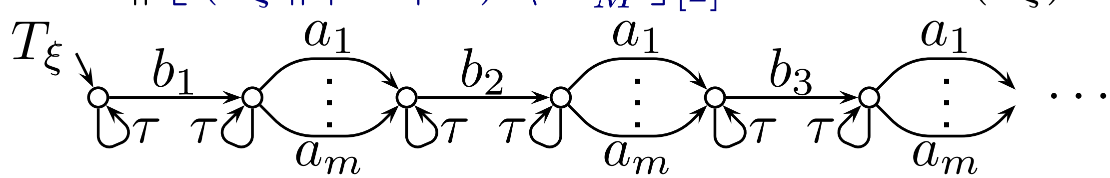
Theorem If

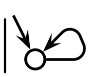
- "$\cong$" is a congruence
- "$\doteq$" implies "$\cong$"
- "$\cong$" preserves Tr but not Inf
-  $\cong$  τ

then $L \cong L' \;\Leftrightarrow\; \Sigma(L) = \Sigma(L') \wedge \mathit{Tr}(L) = \mathit{Tr}(L')$.

Proof

- there is $\xi \in \mathit{Inf}(M_1) \setminus \mathit{Inf}(M_2)$, where $M_1 \cong M_2$
- let $b_1 b_2 \cdots = \xi^{[1]}$ and $\{a_1, \ldots, a_m\} = \Sigma(L)^{[2]}$
- let $f'(L,M) \;=\; L \,\|\, \lfloor (T_\xi \,\|\, \lceil M \rceil^{[1]}) \setminus \Sigma_M^{[1]} \rfloor_{[2]}$, where $\Sigma(T_\xi) = \Sigma_M^{[1]} \cup \Sigma(L)^{[2]}$



- $\mathit{Tr}(f'(L,M_i)) = \mathit{Tr}(L)$, $\mathit{Inf}(f'(L,M_1)) = \mathit{Inf}(L)$, and $\mathit{Inf}(f'(L,M_2)) = \emptyset$
- $L \equiv L \|$  $\cong L \|$  $\tau \doteq f'(L,M_1) \cong f'(L,M_2) = f(L)$
- $\mathit{Sf}(f(L)) = \emptyset$, $\mathit{Div}(f(L)) = \mathit{Tr}(L)$, and $\mathit{Inf}(f(L)) = \emptyset$ — use the lemma

9 Discussion

Also other ideas were used in the proofs

- lemmas like "any congruence that preserves Div also preserves Tr and $eanI$"
- any congruence implied by "$\doteq$" such that $\tau \xrightarrow{\tau} \not\cong \tau$ preserves Sf
- any congruence implied by "$\doteq$" such that $\tau \xrightarrow{\tau} \not\cong$ preserves $minD$
- if $\hat{s} = \sigma \Rightarrow s$, $\hat{s} = \rho \Rightarrow s$, and σ and ρ need different processing, use $L \,\|\, \mathrm{Det}(L)$
- composing f from many "$\cong$"-preserving information-destroying functions
  - nF: ν preserves Div and Inf, but $Sf(\nu(L)) = Sf(L) \cup (Div(L) \times 2^{\Sigma(L)})$

The value of the result is in proving the *absence* of more congruences

A nontrivial range was covered, without using more exotic operators than Φ

- cf. similar results by Roscoe for CSP

Without $a.L$, the following (and many similar) would be a congruence

- $L \cong L'$ if and only if $L \doteq L'$ or $\varepsilon \in Div(L) \cap Div(L')$

Future dream: extending results to some — just any — part of branching time

THANK YOU FOR ATTENTION!