The Quantitative Linear-Time–Branching-Time Spectrum

Uli Fahrenberg Axel Legay Claus Thrane

IRISA/INRIA Rennes, France / Aalborg University, Denmark

CISS Jan 2012

Upshot

Generalize process equivalences and preorders
to a quantitative setting

- trace equivalence $\rightsquigarrow$ trace distance
- simulation preorder $\rightsquigarrow$ simulation distance
- bisimulation equivalence $\rightsquigarrow$ bisimulation distance
- etc.

Upshot

Problem: For processes with quantities, lots of different ways to measure distance

- point-wise
- accumulating
- limit-average
- discounting
- maximum-lead
- etc

Upshot

Two ideas:

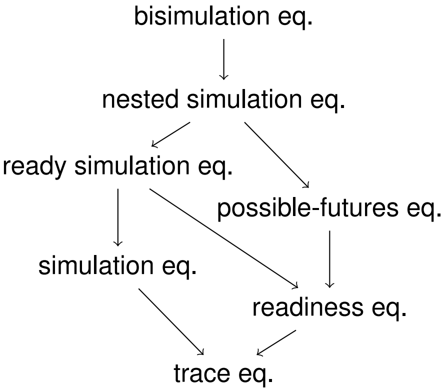
- For an application, it is easiest to define distance between system traces (executions)
- Use games to convert this *linear* distance to *branching* distances

  Or:

- **If you give us a distance between strings, we give you back a bunch of distances between systems.**

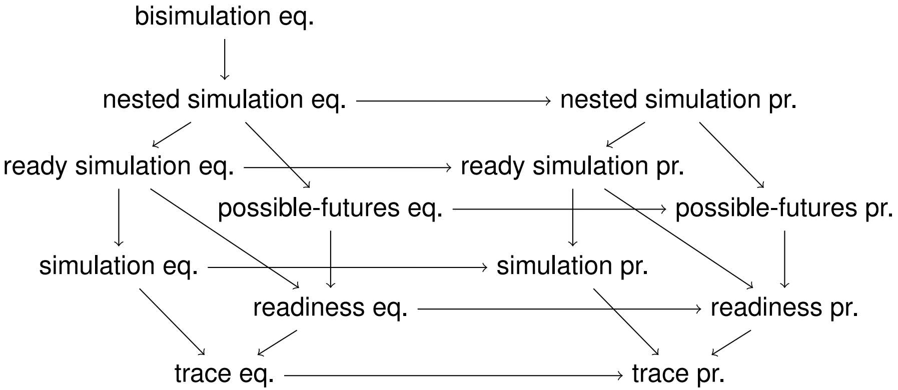
The Linear-Time–Branching-Time Spectrum

van Glabbeek, 2001 (excerpt):



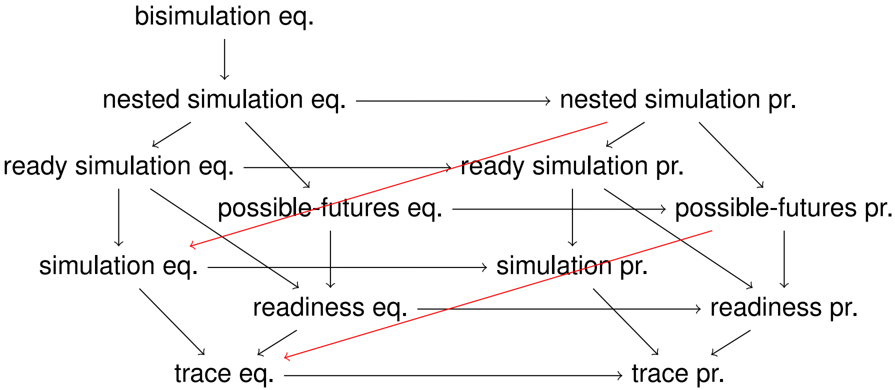
The Linear-Time–Branching-Time Spectrum

van Glabbeek, 2001 (excerpt):

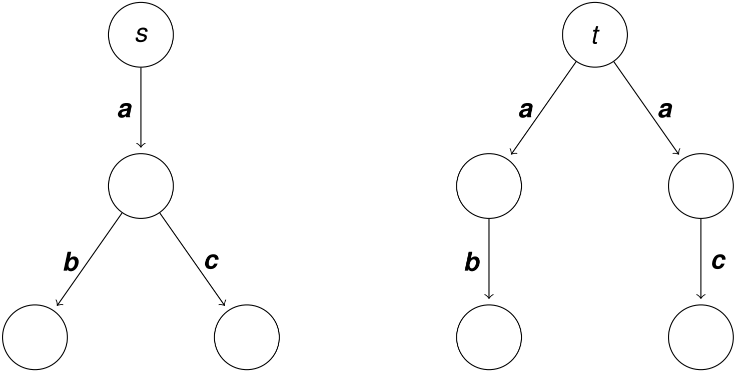


The Linear-Time–Branching-Time Spectrum

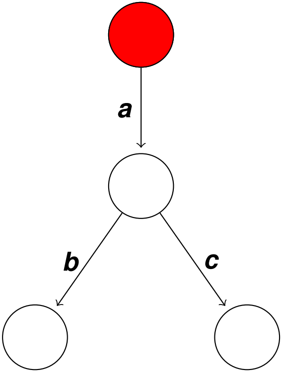
van Glabbeek, 2001 (excerpt):



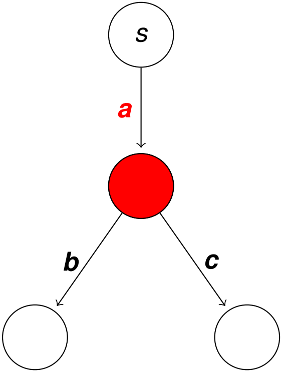
The Simulation Game



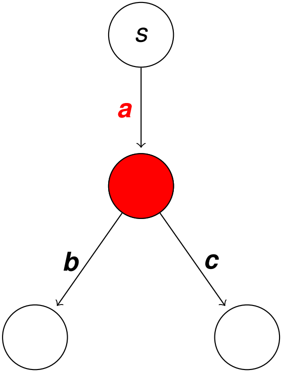
The Simulation Game



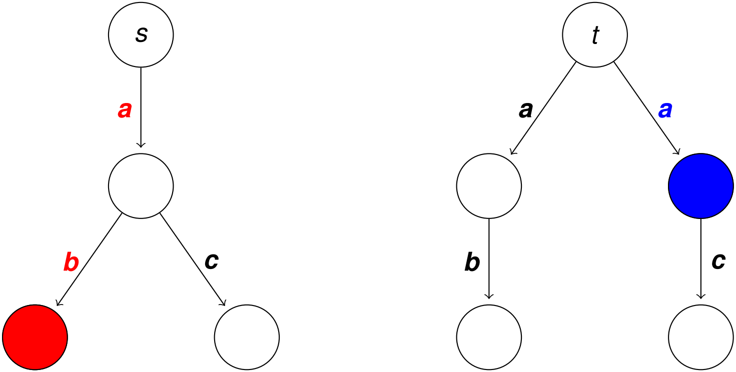
The Simulation Game



The Simulation Game



The Simulation Game

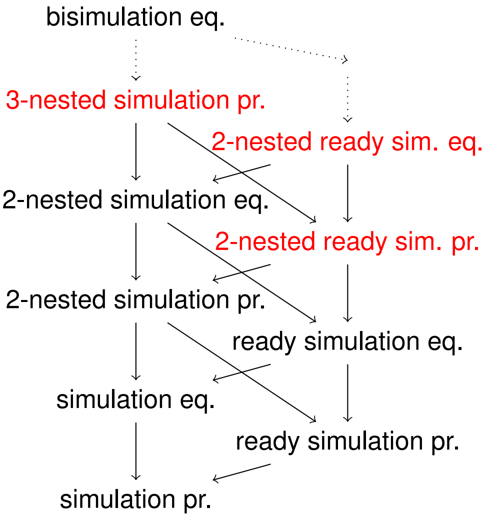


The Simulation Game

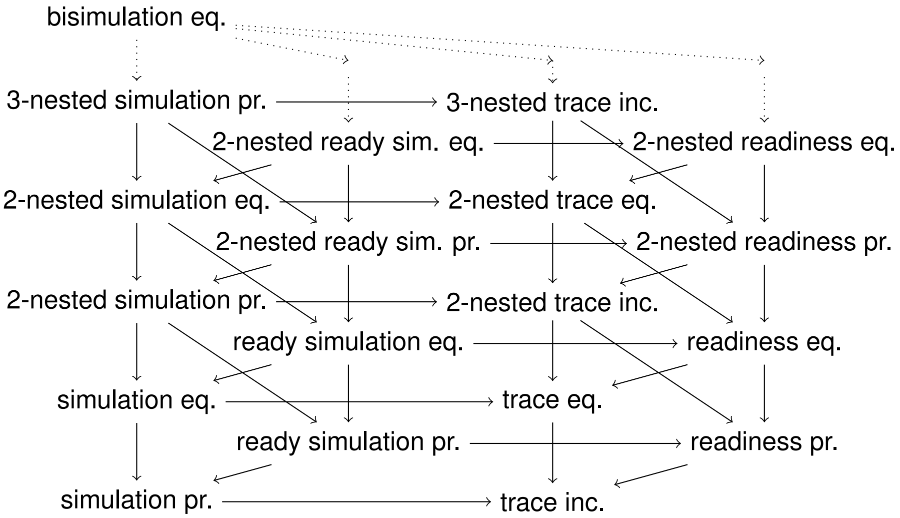
1. Player 1 ("Spoiler") chooses edge from s (leading to s')
2. Player 2 ("Duplicator") chooses matching edge from t (leading to t')
3. Game continues from configuration s', t'

ω. If Player 2 can always answer: YES, t simulates s.
Otherwise: NO

The Linear-Time–Branching-Time Spectrum, Reordered



The Linear-Time–Branching-Time Spectrum, Reordered



The Simulation Game, Revisited

1. Player 1 chooses edge from s (leading to s')
2. Player 2 chooses matching edge from t (leading to t')
3. Game continues from configuration s', t'

ω. If Player 2 can always answer: YES, t simulates s.
Otherwise: NO

Or, as an Ehrenfeucht-Fraïssé game:

1. Player 1 chooses edge from s (leading to s')
2. Player 2 chooses edge from t (leading to t')
3. Game continues from new configuration s', t'

ω. At the end (maybe after infinitely many rounds!), compare the chosen traces:
If the trace chosen by t matches the one chosen by s: YES
Otherwise: NO

Quantitative Ehrenfeucht-Fraïssé Games

The quantitative setting:

- Assume we have a way, possibly application-determined, to measure distances of traces
- Hence a (hemi)metric $d_T : (\sigma, \tau) \mapsto d_T(\sigma, \tau) \in \mathbb{R}_{\geq 0} \cup \{\infty\}$

The quantitative Ehrenfeucht-Fraïssé game:

1. Player 1 chooses edge from s (leading to s')
2. Player 2 chooses edge from t (leading to t')
3. Game continues from new configuration s', t'

ω. At the end, compare the chosen traces σ, τ:
The simulation distance from s to t is defined to be $d_T(\sigma, \tau)$

This can be done for all the games in the LTBT spectrum.

Quantitative EF Games: The Gory Details – 1

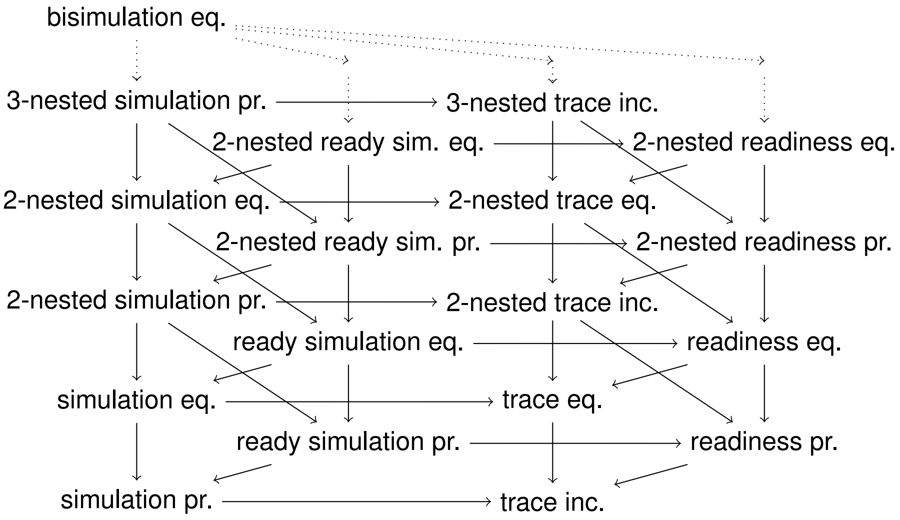
- Configuration of the game: (π, ρ): π the Player-1 choices up to now; ρ the Player-2 choices
- Strategy: mapping from configurations to next moves
  - Θ_i: set of Player-i strategies
- Simulation strategy: Player-1 moves allowed from end of π
- Bisimulation strategy: Player-1 moves allowed from end of π or end of ρ
  - (hence π and ρ are generally not paths – "mingled paths")
- Pair of strategies $\Longrightarrow$ (possibly infinite) sequence of configurations
- Take the limit; unmingle $\Longrightarrow$ pair of (possibly infinite) traces (σ, τ)
- Bisimulation distance: $\sup_{\theta_1 \in \Theta_1} \inf_{\theta_2 \in \Theta_2} d_T(\sigma, \tau)$
- Simulation distance: $\sup_{\theta_1 \in \Theta_1^0} \inf_{\theta_2 \in \Theta_2} d_T(\sigma, \tau)$ (restricting Player 1's capabilities)

Quantitative EF Games: The Gory Details – 2

- Blind Player-1 strategies: depend only on the end of ρ
  - ("cannot see Player-2 moves")
  - $\tilde{\Theta}_1$: set of blind Player-1 strategies
- Trace inclusion distance: $\sup_{\theta_1 \in \tilde{\Theta}_1^0} \inf_{\theta_2 \in \Theta_2} d_T(\sigma, \tau)$
- For nesting: count the number of times Player 1 choses edge from end of ρ
  - Θ_1^k: k choices from end of ρ allowed
- Nested simulation distance: $\sup_{\theta_1 \in \Theta_1^1} \inf_{\theta_2 \in \Theta_2} d_T(\sigma, \tau)$
- Nested trace inclusion distance: $\sup_{\theta_1 \in \tilde{\Theta}_1^1} \inf_{\theta_2 \in \Theta_2} d_T(\sigma, \tau)$
- For ready: allow extra "I'll see you" Player-1 transition from end of ρ

The Quantitative Linear-Time–Branching-Time Spectrum

For any trace distance $d : (\sigma, \tau) \mapsto d(\sigma, \tau) \in \mathbb{R}_{\geq 0} \cup \{\infty\}$:



Further Results

Transfer Principle:

- Given two equivalences or preorders in the *qualitative* setting for which there is a *counter-example* which separates them, then the two corresponding distances are topologically inequivalent
- (for any reasonable trace distance $d : (\sigma, \tau) \mapsto d(\sigma, \tau)$)
- (And the proof uses precisely the same counter-example)

Further Results

Recursive characterization:

- If the trace distance $d : (\sigma, \tau) \mapsto d(\sigma, \tau)$ has a decomposition $d = g \circ f : \text{Tr} \times \text{Tr} \to L \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ through a complete lattice L,
- and f has a recursive formula
- *i.e.* such that $f(\sigma, \tau) = F(\sigma_0, \tau_0, f(\sigma^1, \tau^1))$ for some $F : \Sigma \times \Sigma \times L \to L$ (which is *monotone* in the third coordinate)
- (where $\sigma = \sigma_0 \cdot \sigma^1$ is a split of σ into first element and tail)
- then all distances in the QLTBT are given as least fixed points of some clever functionals using F

All trace distances we know can be expressed recursively like this.

Conclusion & Further Work

- We show how to convert any (typically application-given) distance on system traces to any type of branching distance in the LTBT spectrum
- "*In doing this, they avoid many future papers on many possible variations — just for that, this paper deserves to be published!*"
  – an anonymous reviewer
- "Adding an extra dimension to the LTBT spectrum"
- Application to different scenarios (How does it work in concrete cases? Do we get sensible algorithms? Approximations?)
- Application to real-time and hybrid systems
- Quantitative LTBT with silent moves?
- What about probabilistic systems?