Seminar
Concurrency Theory
Initial Meeting

Luis María Ferrer Fioriti Hernán Baró Graf Hassan Hatefi
Dr. Martin Neuhäußer Dr. Andrea Turrini

Chair for Dependable Systems & Software
Saarland University
neuhaeusser@cs.uni-saarland.de
http://depend.cs.uni-sb.de/index.php?concurrency/

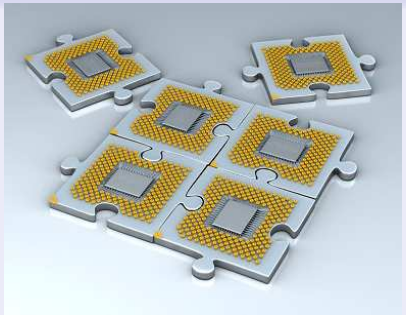
Updated version! November 14, 2011

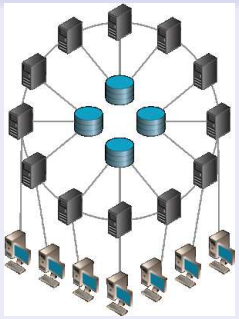
Deadlines have been shifted!

Why concurrency theory?

Example of concurrency

- Multi-core processors
- Distributed computing



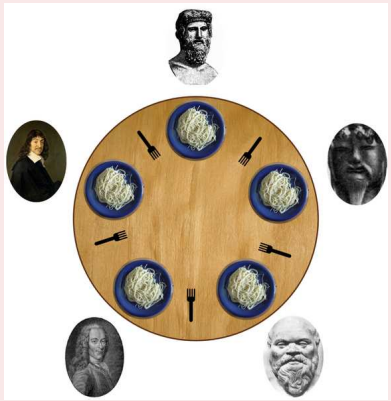


Challenges

Parallelism induces **uncertainty**.

How to describe concurrent behavior?

- Interleaving semantics
- True concurrency



Learning objectives

Learn about concurrency

What theoretical approaches exist to reason about concurrency?

- Which formalism is best suited to model a specific aspect of a system?
- Different ways to think about concurrency.
- **Decidability** results.

Learn how to work scientifically

- Grasp the essence of a research topic and understand it thoroughly.
- Find references and additional papers in the scope of your topic.
- Write a scientific seminar paper on your topic.
- Give a talk and present scientific results comprehensibly.

Learning objectives

Learn about concurrency

What theoretical approaches exist to reason about concurrency?

- Which formalism is best suited to model a specific aspect of a system?
- Different ways to think about concurrency.
- **Decidability** results.

Learn how to work scientifically

- Grasp the essence of a research topic and understand it thoroughly.
- Find references and additional papers in the scope of your topic.
- Write a **scientific** seminar paper on your topic.
- Give a talk and present scientific results **comprehensibly**.

Topics in concurrency theory

Interleaving concurrency

One step at a time, **completely ordered**.

versus

True concurrency (non-interleaving concurrency)

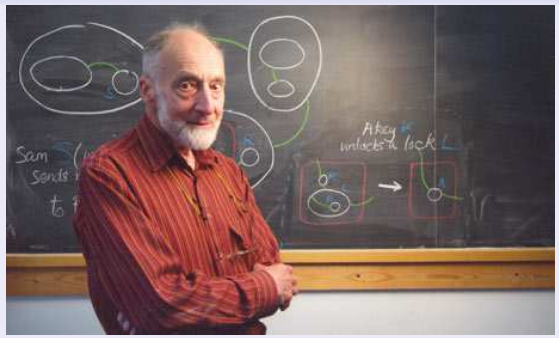
Order of events (transitions, steps, etc.) remains **partial**, **causality**.

Process calculi: CCS

A complete axiomatisation of observational congruence

- Strong bisimulation is too fine grained.
- Weak bisimulation is too coarse.

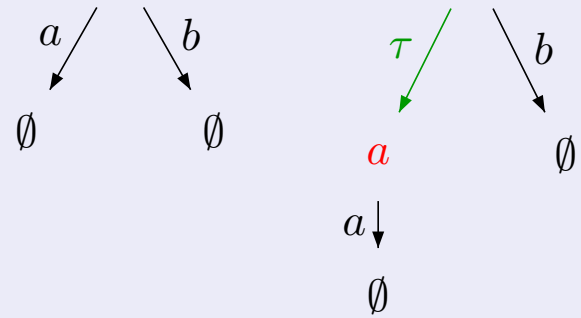
$\Rightarrow$ **Observational congruence!**



This topic is on a complete axiomatisation of observational congruence for finite CCS processes.

Weak bisimulation:
$a \approx_{weak} \tau.a$ and $b \approx_{weak} b \ldots$

But: $a + b \not\approx_{weak,obs} \tau.a + b$



Observational congruence: $a \not\approx_{obs} \tau.a$

Process calculi: CCS

Barbed bisimulation

- **Reduction** vs. **interaction**.
- Another derivation of strong bisimulation.
- Inspired by reduction in term rewrite systems.
- τ-steps and reduction.

Process calculi: CCS

Comparing recursion, replication, and iteration in process calculi

Extend CCS to infinite behaviors:
Recursion, Replication, and Iteration.

$$P ::= \emptyset \mid \alpha.P \mid P_1 + P_2 \mid P_1 \,||\, P_2 \mid P \setminus L \mid A(x_1, \ldots, x_n) \mid !P \mid P^*$$

Questions that arise in this topic are:

- How to extend CCS with iteration?
- How are recursion, replication and iteration related?
- Can one encode the other?
- Decidability results for convergence and weak bisimilarity.

Process calculi: CCS

Decidable subsets of CCS

- Strong bisimilarity **undecidable** for full CCS.
- Which **subsets** of CCS make it decidable?

$$P ::= \emptyset \mid \alpha.P \mid P_1 + P_2 \mid P_1 \,||\, P_2 \mid P \setminus L \mid A\,(x_1, x_2, \ldots, x_n)$$

$$\Delta = \{X_i \equiv P_i \mid 1 \leq i \leq k\}$$

Process calculi: π-calculus

Foundations of the π-calculus

- CCS' communication structure is **static**.
- π-calculus for **mobile** processes.
- Two entities: **names** and **processes**.

Topics: **Theoretical foundations** of the π-calculus.
Classical notions of **bisimulation** in the π-calculus.

$$
\begin{aligned}
P ::= \; & \emptyset \mid \\
& x(y).P \mid \\
& \bar{x}\langle y \rangle P \mid \\
& [x == b]\, P \mid \\
& P_1 \,||\, P_2 \mid \\
& (\nu x)P \mid \\
& !P
\end{aligned}
$$

Process calculi: π-calculus

A theory of bisimulation for the π-calculus

Classical bisimulations do not preserve all π-calculus operators.

- Ground bisimulation (not preserved under parallel composition)

  $P \xrightarrow{a} P' \Longrightarrow Q \xrightarrow{a} Q'$ for some Q' s.t. $P' \sim_g Q'$

- Late bisimulation (not preserved under prefixing)

  $P \xrightarrow{a(x)} P' \Longrightarrow Q \xrightarrow{a(x)} Q'$ for some Q' s.t. for all y. $P'\,[y/x] \sim_l Q'\,[y/x]$

Topic: **A better bisimulation** which

- preserves all π-calculus operators.
- Bisimulation for CCS + X.

$$\begin{array}{ccc} a(x).\,[x == b]\,.\bar{b}b & \begin{array}{c}\sim_g \\ \not\sim_l\end{array} & a(x).\emptyset \\ \Big\downarrow a(x) & & \Big\downarrow a(x) \\ [x == b]\,.\bar{b}b & \begin{array}{c}\sim_g \\ \sim_l\end{array} & \emptyset \end{array}$$

$$a(x).\,[x == b]\,.\bar{b}b||\bar{a}b \;\not\sim_g\; a(x).\emptyset||\bar{a}b$$

Process calculi: The spi-calculus

Introduction of the spi calculus

- Extension of the π-calculus
  to model **cryptographic protocols**.
- Extend π-calculus with notions of
  - encryption
  - decryption

Process calculi: Psi-calculus

The theory of the Psi-calculus

- General extension the π-calculus
- With nominal data
- Data terms, conditions and assertions

Process calculi: Mobile ambients

Mobile ambients

- Mobility: **Computing nodes** and **data**!
- Ambient: Where computation happens.
- Agents: Processes.
- Ambients have associated processes.
- Ambients can be nested and move.

Interface theories

Modal transition systems and interface theories

- **Modal transition systems**
- Modalities (necessary and admissible behavior)
- Refinement.
- Extend process algebra towards specification logic!
- **Interface theories**

Input/Output Automata

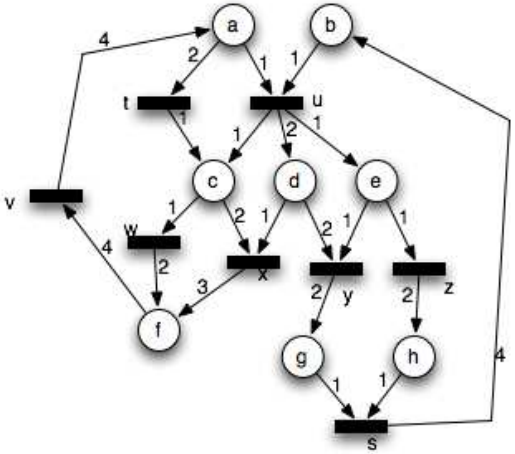
I/O automata

- Automata-based interface language
- Assumptions about environment
- Guarantees about component's behavior
- Extend interface automata to modal I/O automata.

Non-interleaving models of concurrency

Undecidability of bisimilarity for Petri nets

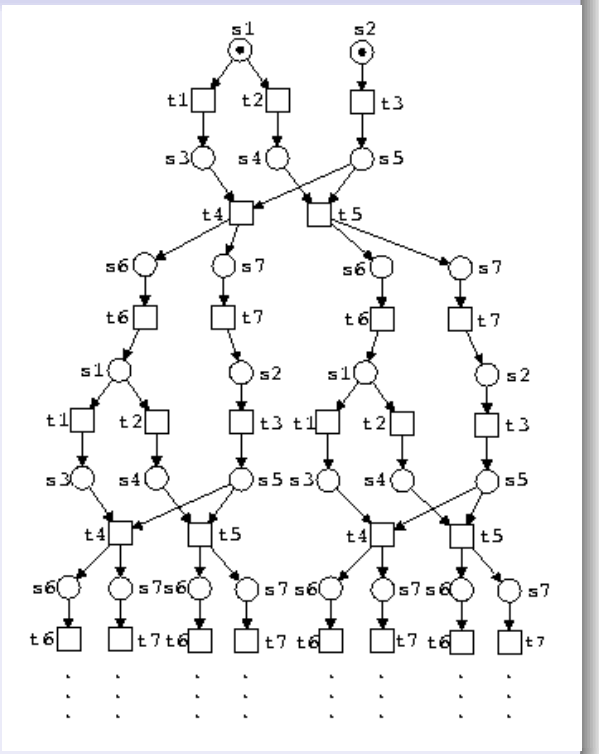
- Petri nets as a true-concurrency model
- **Undecidability of bisimilarity** in Petri nets
- Proof by reduction of the halting problem of **counter machines**.
- Further decidability results



Non-interleaving models of concurrency

Branching processes of Petri nets

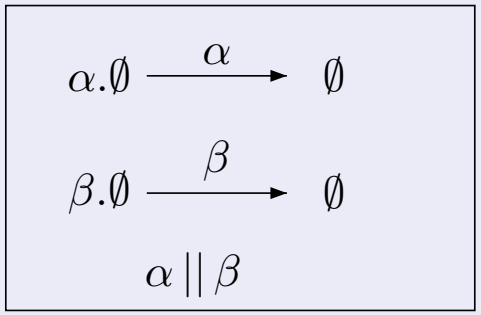
- Branching process = **initial run** of a Petri net
- **Partial order** on branching processes
- Unfolding of Petri nets
- Basis for **event structures**.

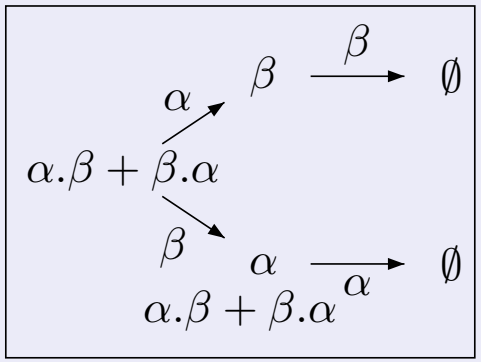


Non-interleaving models of concurrency

The relation between CCS and Petri nets

- Usually, CCS has a **interleaving semantics**.
- $\alpha \,||\, \beta$ and $\alpha.\beta + \beta.\alpha$ coincide in classical CCS.
- But not here:
  - CCS semantics defined by Petri nets
  - No **linear** but a **causal** order on events

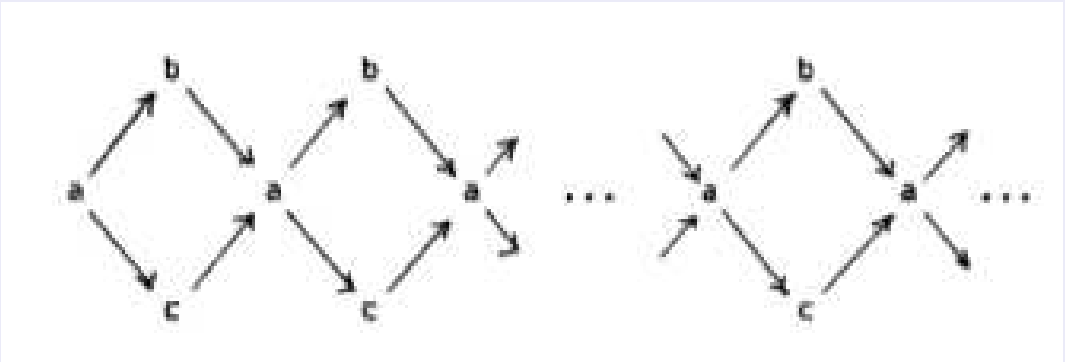




Non-interleaving models of concurrency

Trace theory and Mazurkiewicz traces

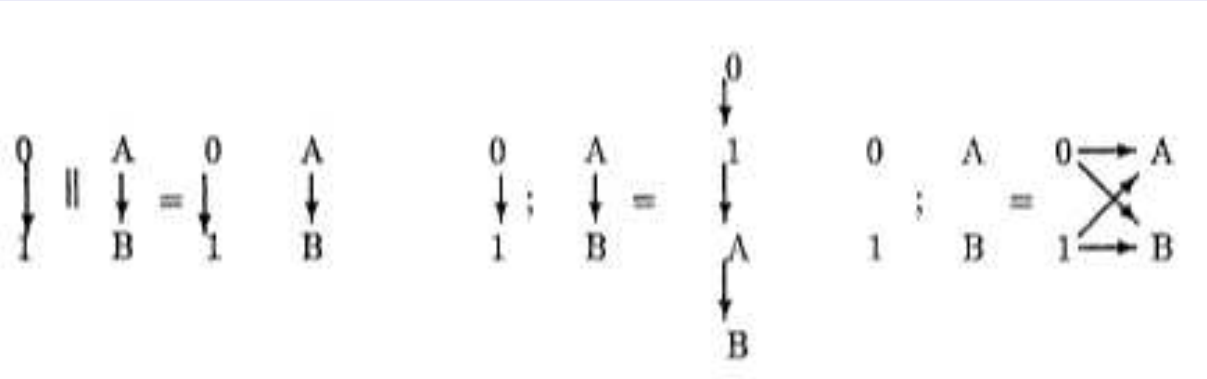
- Traces describe processes
- Not sequential, but partially ordered
- **Non-interleaving semantics**
- Strings vs. traces



Non-interleaving models of concurrency

Modeling concurrency with partial orders

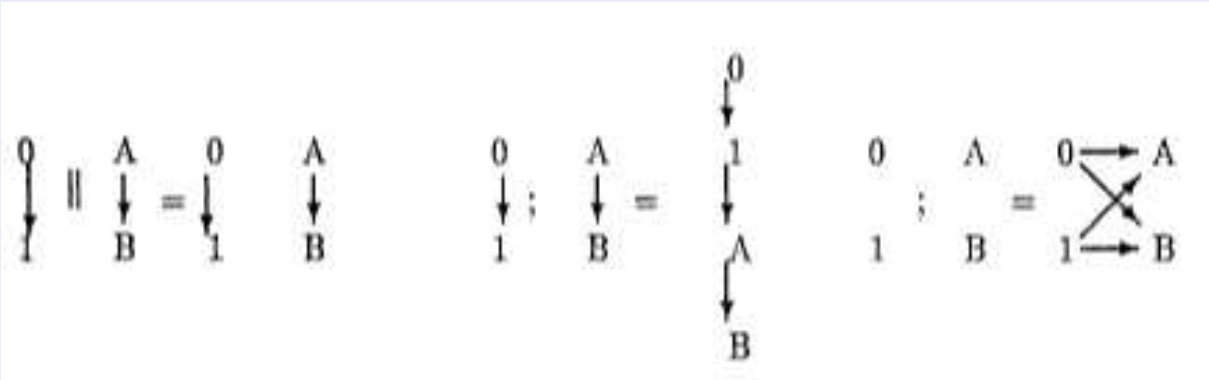
- Partial string: **partially ordered multiset (Pomset)**
- Set of partial strings: Processes



Non-interleaving models of concurrency

Comparing Pomsets and Mazurkiewicz traces

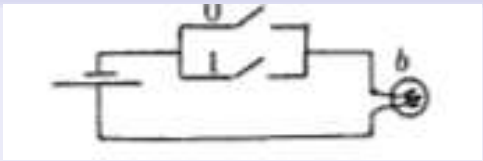
- Trade-off between Pomsets and Mazurkiewicz traces:
- Mazurkiewicz traces are restricted subclass of Pomsets
- **Pomsets more general**
- **Mazurkiewicz traces have more structure**

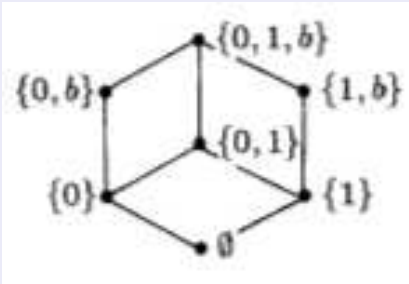


Non-interleaving models of concurrency

Event structures

- Processes execute **events**.
  - consistent (events may prevent others)
  - enabling: (sequence of events)
- **Configurations** are the state of a process,
- they are partially ordered: **progress**.





Expectations

Seminar paper

- Prepare a seminar paper of **max. 20 pages**.
- Include a complete list of **references** and **cite correctly**.
- **Plagiarism**:
  Copying content from other sources without citation
  leads to disqualification.
- Font size 11pt with the usual page dimensions.
- Language is **English**.
- We expect **proper language** if you hand in your drafts:
  $\geq$ 10 typos per page $\Longrightarrow$ no further corrections

The final seminar papers are distributed **at least a week before** the talks.
$\Longrightarrow$ Read other participants' papers!

Expectations

Seminar talk

- You give a scientific talk of **30 minutes**.
- Present your topic according to the audience.
  Goal: Participants should be able to follow you!
- **Well structured** slides:
  - Not more than ~ 10 lines of text per slide.
  - Sensible use of colors
  - Select the important aspects and present them formally.
  - Use examples and figures to motivate theory.
- The talk will be given in **English**.

Be prepared to ask questions and to discuss about the seminar's topics.

Organisation

Schedule for the seminar paper (Attention: Schedule changed!)

November 21:	Last possibility to withdraw from the seminar.
December 5:	Hand in the structure of your paper.
	Be prepared to discuss about your topic!
January 11:	Hand in the seminar paper.
February 1:	Final version of seminar paper due.
February 15:	Hand in preliminary version of slides.
March 7:	Final version of slides due.
March 12 & 13:	"Mini conference" with seminar talks.

Handling of deadlines

- All **deadlines** in this seminar are **strict**.
  Missing a deadline disqualifies from further participation.
- Participation during the talks is **mandatory**.

Supervision

Supervisors

- Luis María Ferrer Fioriti
  ferrer@depend.cs.uni-saarland.de
- Hernán Baró Graf
  hernanbg@cs.uni-saarland.de
- Hassan Hatefi
  hhatefi@depend.cs.uni-saarland.de
- Dr. Martin Neuhäußer
  neuhaeusser@cs.uni-saarland.de
- Dr. Andrea Turrini
  turrini@cs.uni-saarland.de