Formal Methods in Algorithmic Cheminformatics and Systems Biology

Juri Kolčák

[YURI KOLCHA:K]

Wednesday 9th October, 2024

About this Course

Two hours lecture + two hours seminar weekly.

The slides will be made available online:
`www.jurikolcak.github.io/teaching.html`

There are no other official materials, but you are free to consult some books or articles:

E. De Maria. *Systems Biology Modelling and Analysis: Formal Bioinformatics Methods and Tools.*
Wiley, 2022

M. Bernardo, P. Degano, and G. Zavattaro, editors. *Formal Methods for Computational Systems Biology.*
Lecture Notes in Computer Science. Springer Berlin, Heidelberg, May 2008

C. Baier and J. P. Katoen. *Principles of Model Checking.*
The MIT Press. MIT Press, 2008

F. Bause and P. S. Kritzinger. *Stochastic Petri Nets: An Introduction to the Theory.*
Vieweg+Teubner Verlag, 1996

J. M. Bower and H. Bolouri. *Computational Modeling of Genetic and Biochemical Networks.*
Bradford book. MIT Press, 2001

The course is concluded with an oral exam.

Algorithmic Cheminformatics

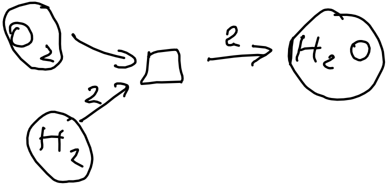
My group!
https://www.uni-bielefeld.de/fakultaeten/technische-fakultaet/arbeitsgruppen/algorithmische-chemieinformatik/

Head of Group: Prof. Daniel Merkle

Main analysis tools: graph transformation, Petri nets.

Examples:

- Chemical Reaction Networks
- Metabolic Networks
- Metabolic/Synthesis Pathways
- Mass Spectra Analysis



Formal Methods

from software verification

Mathematical Logic, Automata, Formal Languages, Type Theory, Control Theory, Game Theory, . . .

- **formal specification** (specification, modelling, design);
- **formal verification** (verification, analysis);

Partial observation/specification – The application area or setting of the problem is only partially known/described/measurable.

→ specific to live Sciences

Blurs the line between specification and verification as verification methods bleed into the specification by means of model or parameter inference.

"Trying to infer the source code from sample runs of the program."

Systems Biology

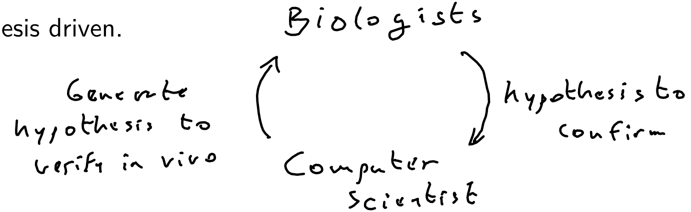
Complex systems in **biology**.

Mathematical and computational study of systems as a whole, with focus on complex interactions between the components and the arising emergent properties/behaviours, which are not observable by study of the components individually.

Holistic approach: “A system is more than the sum of its parts.”

The dynamics (behaviour) of the systems and dynamical properties are of interest.

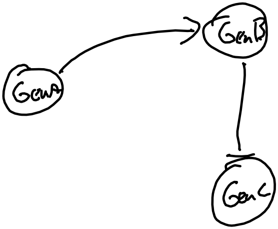
Hypothesis driven.



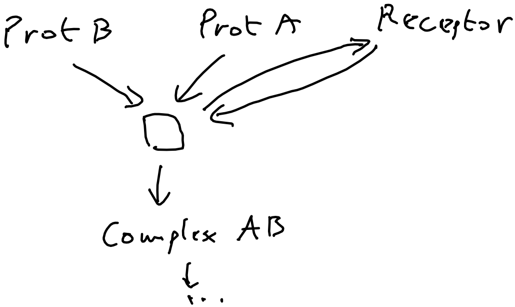
Examples from Biology and Chemistry

1. Gene Regulatory Networks
2. Signalling Pathways
3. Metabolic Networks
4. Population Models
5. Cells in a Tissue/Tissues in an Organism
6. Chemical Reaction Networks

1. GRNs



2.



Brief History

Biology has been dominated by reductionist approaches since inception, owning to the popularisation of the approach in post-Newtonian physics.

Several biologists started to advocate for the importance of a systems-approach in the first-half of the 20th century.

It wasn't until the turn of 20th century when the technological advances in measurement technologies enabled the accumulation of large amounts of data.

Data

* OMICS

1. Genomics
2. Epigenomics
3. Transcriptomics
4. Proteomics
5. Metabolomics
6. Phenomics
7. Lipidomics

→ Changes of the above ⇒ Dynamical properties

Need for Formal Methods in Systems Biology

Motivation comes from scale.

- Need to characterise complex interactions, which are highly combinatorial.
- Need to express and reason about dynamical properties.
- Large amounts of input data to consider.

Formal Specification (Modelling)

Specification = Syntax + Semantics

$x ::= v \mid True \mid False$

$\varphi ::= x \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \neg \varphi \ \ldots$

Sem.:

$A \wedge B$ is true iff A is true and B true

Modelling Pipeline

1. Identify the problem of interest.
2. Gather sufficient background knowledge and data.
3. Choose a suitable model type.
4. Assemble the model.

Greater precision leads to larger size.

Larger size leads to higher complexity.

Models are tailored to the analysis problems. (KISS)

Deterministic vs Non-deterministic Models

Deterministic: The evolution in any configuration is fixed. There is only one option.

Non-Deterministic: The evolution in some (all) configurations is not fixed. There may be more than one option.

Stochastic: The evolution in any configuration is governed by a random variable sampled from a predefined probability distribution.

Deterministic:

ODE (ordinary differential equations)

Boolean Networks (with synchronous updating)

Non-Deterministic:

Boolean Networks (with asynchronous updating)

Petri Nets

Stochastic:

Stochastic Petri Nets

Kappa (site graph rewriting)

Qualitative vs Quantitative Models

Qualitative Used to answer "yes/no" questions.

Quantitative Used to answer questions about "measurable" properties.

Qualitative:

Boolean networks

(safe) Petri nets

Qualitative:

ODEs

Petri nets (incl. stochastic)

Kappa

Discrete vs Continuous Models

Discrete The evolution happens in discrete time chunks (not necessarily defined intervals).

Continuous The evolution happens continuously, no time steps.

Hybrid The evolution happens in continuously until a boundary is hit, then a discrete step occurs.

Discrete:

Boolean Networks,

Petri nets (incl. stochastic)

Kappa

Continuous:

ODEs

Agent-Based vs Generative Models

Agent-Based All "actors" in the model are predefined explicitly. Their internal structure is not necessarily important.

Generative The "actors" in the model are generated on the fly based on a set of rules that apply based on their internal structure. (predictive aspect)

Agent-Based:

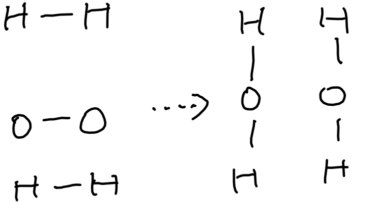
Petri nets (stochastic)

Boolean networks

ODEs

Generative:

Kappa



Formal Verification (Analysis)

"Categories"

- Static Analysis → is blind to dynamics
- Dynamic Analysis → Transition system
- Causal Analysis

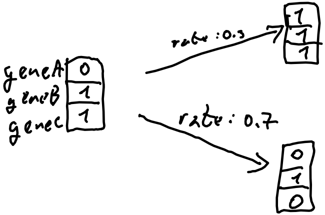
Concrete methods

- Simulation
- Model Checking
- Abstract Interpretation
- Unfolding

(Discrete)

Model Dynamics

(Labelled) Transition Systems.



Configurations of the system as vertices

Directed (labelled) edges represent the possible transitions between configurations.

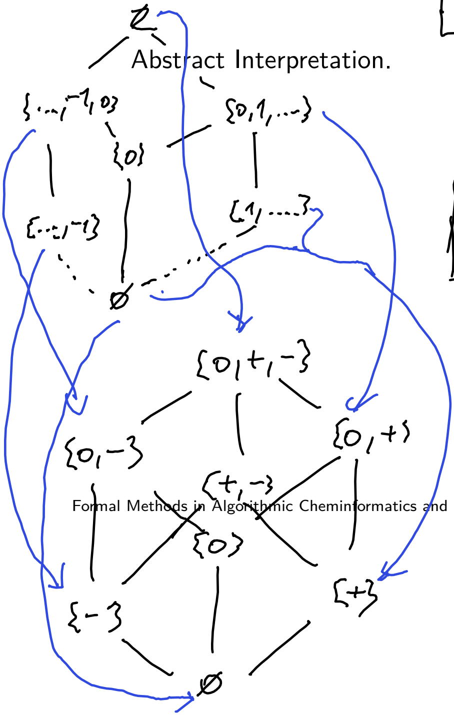
A (labelled) transition system is in general exponentially larger than the model, if not infinite.

Static Analysis

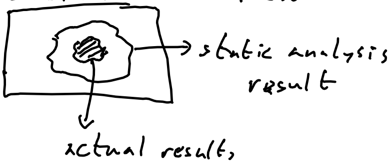
Exploits the links between structure and dynamics of the model.

Approximate results.

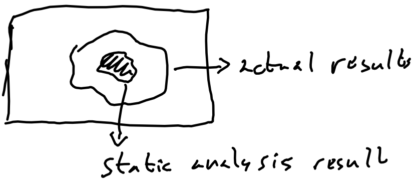
Abstract Interpretation.



OVERAPPROXIMATION



UNDERAPPROXIMATION



Model Checking

Properties specified in logic. In systems biology/chemistry almost exclusively a temporal logic.

The configurations of the system are traversed one-by-one to check the validity of the formula.

Symbolic model checking (formulas, BDDs, etc.)

Causal Analysis

Of interest are causal “cause-and-effect” relationships between transitions.

Models dedicated to causality and concurrency analysis (process algebras, **Petri nets**).

Event structures.

Simulation

“Execution” of the model. The result is a single trace, witness of one possible behaviour.

For deterministic models, simulation gives a full account of dynamics (for the given initial state).

Stochastic simulation (Gillespie algorithm).

Deterministic simulation.

Outline

- Petri nets;
- Boolean networks;
- stochastic Petri nets and Markov chains;
- stochastic simulation (Gillespie algorithm);
- Kappa site graph transformation;
- model checking;
- abstract interpretation;
- unfolding;