Models of cellular Processes - Modelle zellulärer Prozess

Matthias König
https://mcp.readthedocs.io

livermetabolism.com
 @konigmatt

Abstraction steps in modeling

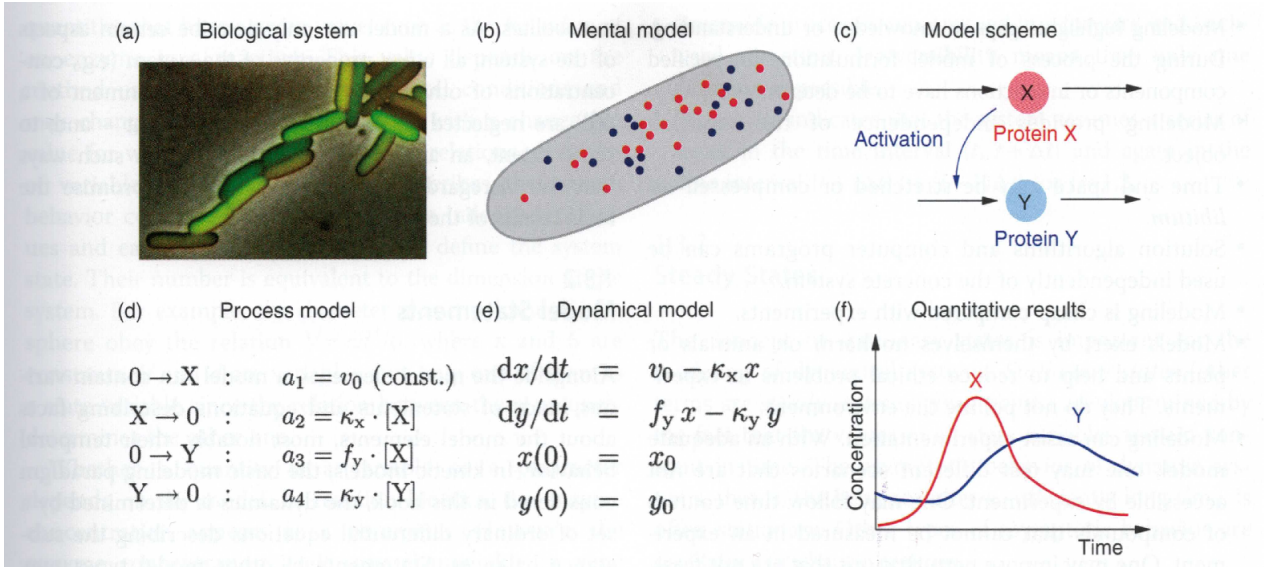
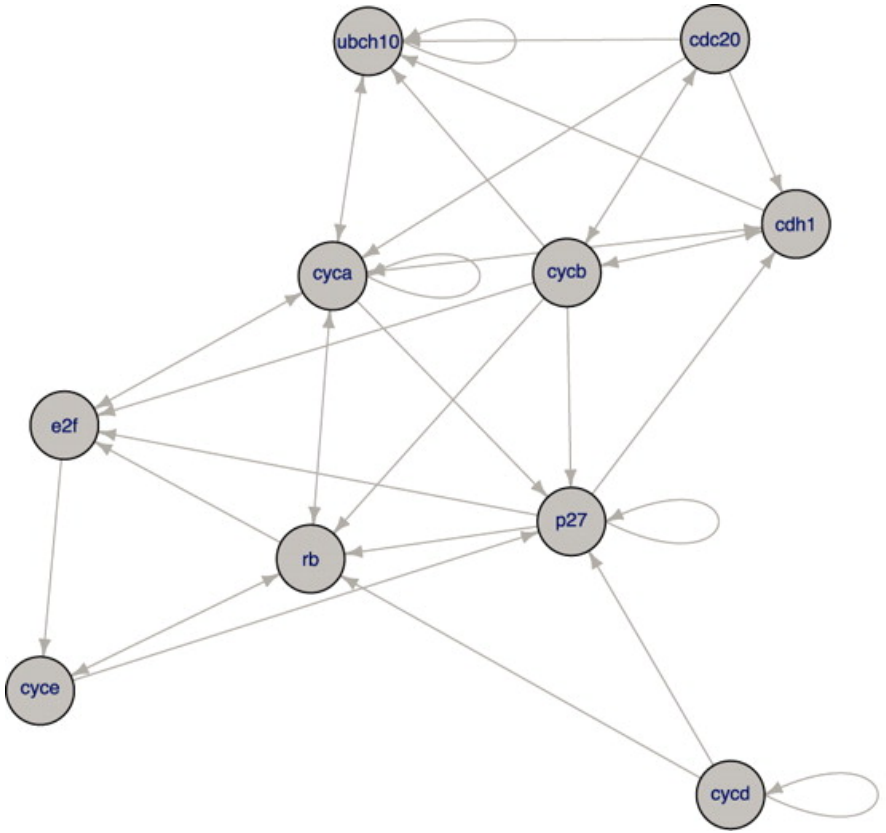
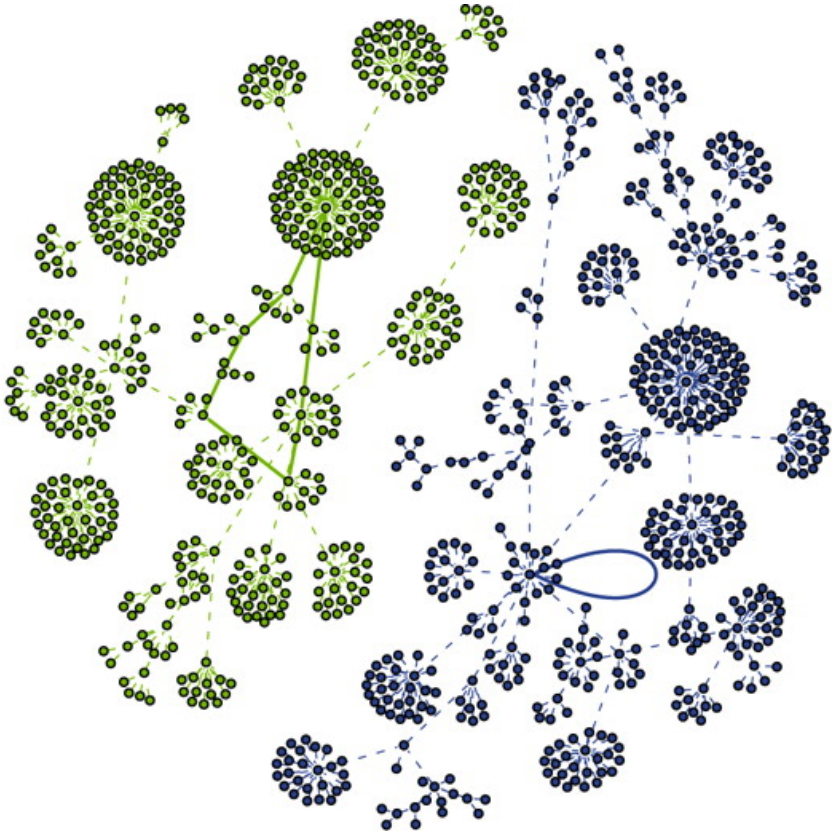


Figure 1.2 Typical abstraction steps in mathematical modeling. (a) *E. coli* bacteria produce thousands of different proteins. If a specific protein type is labeled with a fluorescent marker, cells glow under the microscope according to the concentration of this marker. (Courtesy of M. Elowitz.) (b) In a simplified mental model, we assume that cells contain two enzymes of interest, X (red) and Y (blue), and that the molecules (dots) can freely diffuse within the cell. All other substances are disregarded for the sake of simplicity. (c) The interactions between the two protein types can be drawn in a wiring scheme: each protein can be produced or degraded (black arrows). In addition, we assume that proteins of type X can increase the production of protein Y. (d) All individual processes to be considered are listed together with their rates a (occurrence per time). The mathematical expressions for the rates are based on a simplified picture of the actual chemical processes. (e) The list of processes can be translated into different sorts of dynamic models, in this case, deterministic rate equations for the protein concentrations x and y. (f) By solving the model equations, predictions for the time-dependent concentrations can be obtained. If the predictions do not agree with experimental data, this indicates that the model is wrong or too much simplified. In both cases, the model has to be refined.

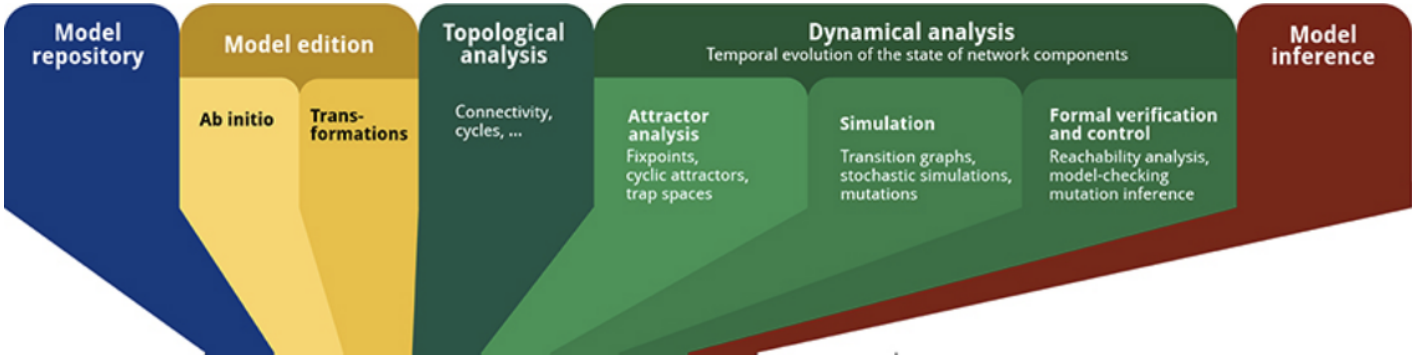
Systems Biology, A Textbook, Second Edition
Klipp, Liebermeister, Kowald

Boolean Networks





Software



Software	Technology	Specificity
CellCollective	Javascript	Web application
GINsim	Java	Graphical and script interface
bioLQM	Java, ASP	Command line and script interface
MaBoSS	C++	Stochastic simulations
Pint	OCaml, ASP	Large-scale analysis; formal inference of mutations
NuSMV	C	General purpose symbolic model-checker
BoolSim	C++	Scalable cyclic attractors identification
BooleanNet	Python	Discrete and hybrid semantics
pyBoolNet	Python, ASP	Boolean network Python toolbox
BoolNet	R	Simulation, attractors, network reverse-engineering
CellNOptR	R	Network optimisation from steady-state/time-series
CaspoTS	ASP	Exhaustive inference from multiplex time-series

Advanced Boolean networks

asynchronous Boolean networks

- asynchronous update schemas: a random node is selected in each time point and updated
- repeated simulation of the same network with identical start conditions can provide an average behavior of the network

multiple responses per state

- discrete levels: 0, 1, 2, 3
- more complicated rules but the principles remain the same

random Boolean networks

- generalization of boolean networks
- update rules are choosen randomly during construction (remain constant over time)

probabilistic boolean networks

- assign with a certain probability update rules to nodes at each time step