Practical Lab Session: From Continuous Gene Regulatory Networks to Discrete Boolean Models

Sergio Peignier

Abstract

This practical session bridges continuous and discrete modeling paradigms in systems biology. Students will implement a 3-node continuous Gene Regulatory Network (GRN) using Hill kinetics in PyTorch, transition it into a discrete Boolean framework with synchronous and asynchronous updating schemes, and analyze State Transition Graphs (STGs). In the second part, students will scale up to a biologically realistic 12-variable mammalian cell cycle network (Traynard et al., 2016) using both custom NumPy solvers and the advanced `PyBoolNet` library.

1 3-Node GRN (Continuous to Discrete)

1.1 Continuous Simulation with PyTorch

Implement a 3-gene regulatory circuit where gene A is activated by B, B is activated by C, and C is activated by A or B using Hill activation kinetics:

$$H^+(x; K, n) = \frac{x^n}{K^n + x^n} \tag{1}$$

The continuous update follows Euler integration:

$$\frac{dx_i}{dt} = \text{Production}_i - \gamma x_i \tag{2}$$

Where Production_i is simply the equation involving the activators and inhibitors of x_i. Choose one initial condition and plot some trajectories. Change some of the the parameters and analyze the impact.

1.2 Boolean Discretization and State Transition Graphs

Convert the continuous states into binary variables (0 or 1) and program:

- **Synchronous updates:** All nodes update simultaneously based on current states.
- **Asynchronous updates:** Single-node updates (or priority-constrained updates) capturing stochastic biological time delays.
- **State Transition Graph Generation:** Mapping all 2^3 or 2^{12} states to their successors (using a synchronous update).
- **Trajectory tracing**: to detect terminal attractors.

2 Exercise 2: Mammalian Cell Cycle Network (Traynard et al., 2016)

Analyze the mammalian cell cycle network consisting of 12 regulatory variables (including cyclins `CycD, CycE, CycA, CycB`, checkpoints `Cdc20, Cdh1`, and inhibitors like `p27` and `Rb`).

- Paper Reference: Traynard et al. (2016), *Descriptive and predictive model of mammalian cell cycle regulation*, `https://pubmed.ncbi.nlm.nih.gov/27013506/`
- Using the `bnet2primes` function from `pyboolnet.file_exchange` load the network into pyboolnet.
- Use the `attractors` function from `pyboolnet.attractors` to obtain and analyze the attractors.
- Use the `stg` function from `pyboolnet.state_transition_graphs` to simulate a trajectory path from a given initial condition, and plot it as a heatmap.
- Analyze environmental perturbation, by studying attractors under growth factor presence (`CycD = 1`) versus serum starvation (`CycD = 0`).
- How would you perform a knock-out experiment? Test different experiments
- Take a look at the `repo` object from `pyboolnet.repository`, load other state-of-the-art boolean networks and analyze them