Final project (homework 5) — assigned on Friday 20 April, due 11:59PM on Friday 4 May

General instructions

Total number of points available: 200 pts.

This is a more open-ended assignment that involves using ODE modeling and Python-based simulation techniques to determine the behavior of a DNA “seesaw” logic gate, which were covered in lectures.

For questions that do not require a programming answer, please include your answer in your submitted file in some appropriate format, e.g., as a separate text file.

Note that if you use any Python programming techniques beyond those covered in class, you must supply comments demonstrating that you understand how and why your code works.

Also note that points will be deducted for bad software engineering practices, in particular, large-scale duplication of code blocks where an iterative solution would be preferable.

How to submit

Your submission should consist of the following parts:

- A commented source file that contains your name and your code. The comments should identify which code addresses each part of the assignment, and should provide any necessary explanation of the code’s operation.
- Appropriate and legible image files containing any plots produced by your code.
- Written answers to any non-programming questions.

All submissions must be submitted via UNM Learn. Please archive your files into a zipfile and submit the zipfile.

5.1 Modeling a seesaw logic gate (200 pts)

Figure 1 below presents the chemical reaction network for a DNA “seesaw” logic gate. The circuit accepts three input species, labeled “INPUT1”, “INPUT2”, and “INPUT3” in Figure 1, and its output is read via a single output species, labeled “OUTPUT” in Figure 1. The intent is that the inputs and outputs are implemented using a “presence/absence” logic encoding, as outlined below.

To complete the final project, carry out the following tasks based on this model:

(a) (50 pts) Write down an ODE model of the chemical reaction network from Figure 1.

(b) (50 pts) Implement your model in Python, including code for running and plotting the results of simulations, given arbitrary starting conditions.

(c) (100 pts) Simulate the system behavior for various combinations of the logical inputs, with appropriate simulation lengths and numbers of datapoints. You should use the following initial concentrations for the species:

- Each input species: 0 if that input is false or $1x$ if that input is true, as appropriate.
- Sum gate: $3x$ nM
- Threshold: $1.5x$ nM
- Signal gate: $1x$ nM
- Fuel: $2x$ nM
- Reporter: $1.5x$ nM
- Output: 0 nM
- All species unnamed in Figure 1: 0 nM

where x is a scale factor of your choice. The following are appropriate values for the rate constants:

- $k = 5 \times 10^{-4}\,\text{nM}^{-1}\,\text{s}^{-1}$
- $kFast = 2 \times 10^{-2}\,\text{nM}^{-1}\,\text{s}^{-1}$

Use the results from your simulations to determine what logic function the system computes. Present your results in a **500-1000 word report** that summarizes your deductive process and your findings, and **explains how the features of the circuit design enable it to compute that function.** You should include all relevant plots and references to the corresponding Python code.

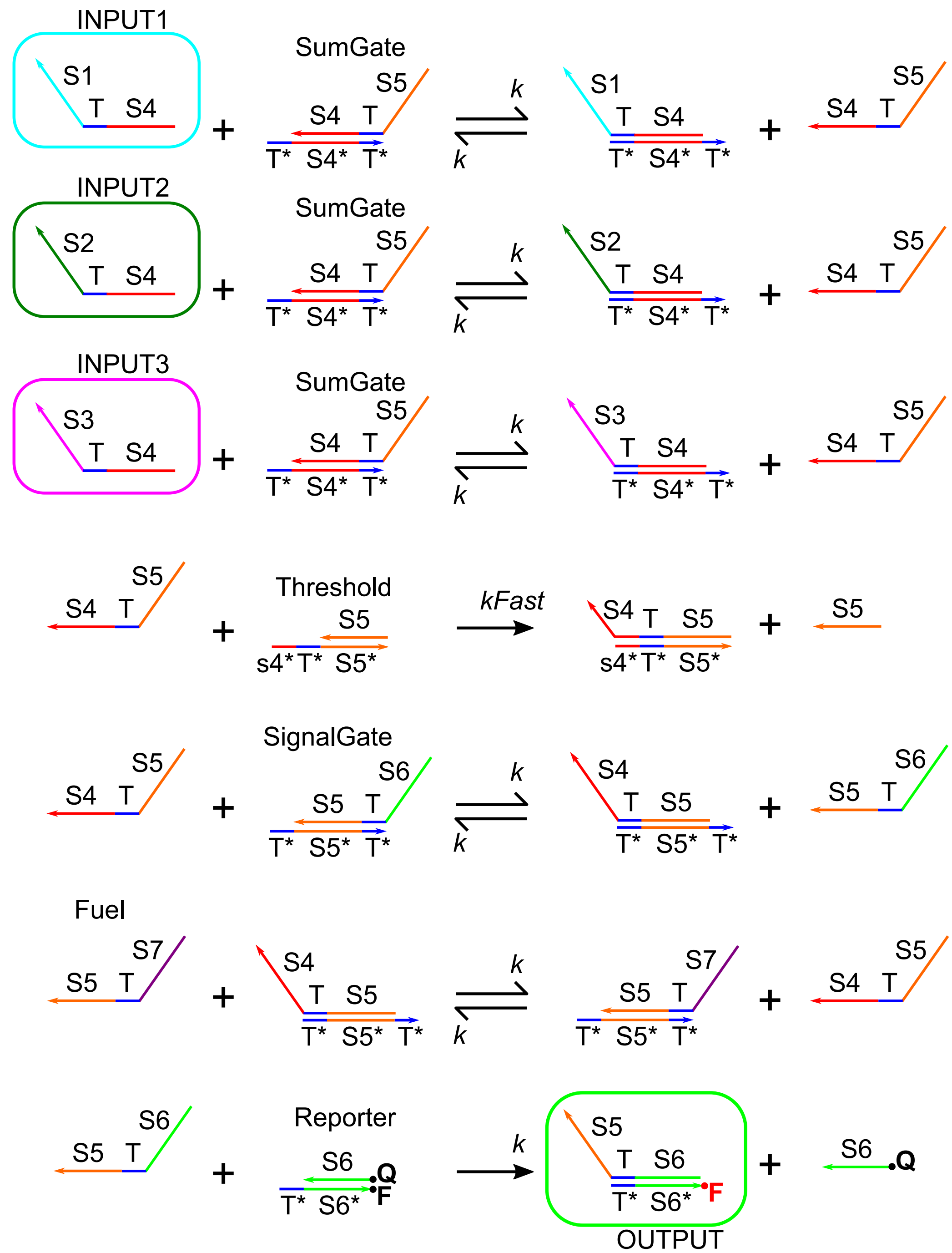

Figure 1: Reactions forming a 3-input, 1-output seesaw gate logic circuit. Species of note for observing the inputs and outputs from the system, and for setting up the initial conditions, are labeled.