Biological Computation – Final Project - ex1

In order to find all of the monotonic regulation conditions of the reasoning engine, we created 9 pairs considering the cases whether none, some or all of the activators /repressors are present (which encapsulate all of the possible cases). Meaning, we considered the following 9 cases (All for Activators, R for repressors):

Index	0	1	2	3	4	5	6	7	8
Case	None A None R	Some A None R	All A None R	None A Some R	Some A Some R	All A Some R	None A All R	Some A All R	All A All R

In total, there are $2^{3\cdot3}$ functions, as there are 9 pairs, and each can make the target (Result) activated or deactivated.

In a for loop that runs 512 times, we create a new 'functions' list for each of these 512 functions: 'functions' maps each of the 9 pairs of activators and repressors to a binary value '1' or '0', which represents 'Activated' or 'Deactivated' respectively. For example, if the list holds [0, 0, 1, 0, 0, 0, 0, 0, 0] it means that the target is activated only when all activators are active, and no repressors.

Filtering

We want to find the functions that meet the following conditions and filter out the others:

- The target (Result) is activated when all of its activators are present and none of its repressors are present. Similarly, a target is deactivated when all of the repressors are present and none of the activators are present.
- The regulation condition must be monotonic: For example, if the target requires only one activator to be activated, then any greater number of activators will also activate that target (assuming no change in the state of the repressors).

The first condition is simple: We simply filter out all of the functions that don't have '1' in their second index (all activators, none Repressors) and don't have '0' in their sixth index (none activators, all repressors). After filtering these out we are left with 128 functions. We filter using the function 'fully_activated_deactivated()'.

For the second condition we must consider the monotonicity of the function. There are a few cases to consider:

- If the target is activated even when no activators are present, it must also be activated when some or all activators are present.
- If the target is activated when some activators are present, it must also be activated when all activators are present.

Using the function 'monotonic_activators_check()' and an input of 3 indices, we only keep the functions that hold these conditions:

$$function[0] = 1 \Longrightarrow function[1] = 1 \; AND \; function[2] = 1$$

$$function[1] = 1 \Longrightarrow function[2] = 1$$

$$function[3] = 1 \Longrightarrow function[4] = 1 \; AND \; function[5] = 1$$

$$function[4] = 1 \Longrightarrow function[5] = 1$$

$$function[6] = 1 \Longrightarrow function[8] = 1 \; AND \; function[8] = 1$$

$$function[8] = 1 \Longrightarrow function[9] = 1$$

Similarly:

- If the target is deactivated even when no repressors are present, it must also be deactivated when some or all repressors are present.
- If the target is deactivated when some repressors are present, it must also be deactivated when all repressors are present.

Using the function 'monotonic_repressors_check()' and an input of 3 indices, we only keep the functions that hold these conditions:

$$function[0] = 0 \Longrightarrow function[3] = 0 \; AND \; function[6] = 0$$

$$function[3] = 0 \Longrightarrow function[6] = 0$$

$$function[1] = 0 \Longrightarrow function[4] = 0 \; AND \; function[7] = 0$$

$$function[4] = 0 \Longrightarrow function[7] = 0$$

$$function[2] = 0 \Longrightarrow function[5] = 0 \; AND \; function[8] = 0$$

$$function[5] = 0 \Longrightarrow function[8] = 0$$

Results

We are left with 18 functions:

1. (AllActivators AND NoneRepressors)

2. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors)

3. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors)

4. (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors)

5. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors)

6. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors)

7. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors)

8. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors)

9. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (NoneActivators AND SomeRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors)

10. (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

11. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

12. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

13. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

14. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

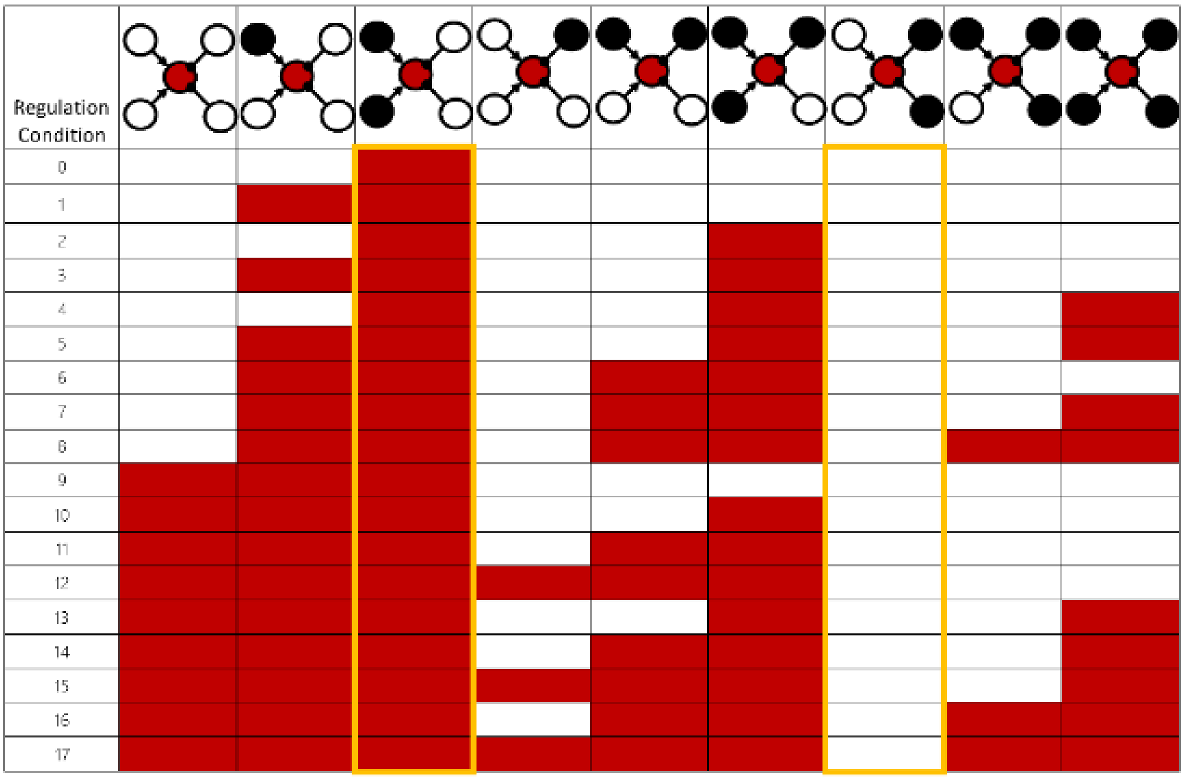
15. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (NoneActivators AND SomeRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (AllActivators AND AllRepressors)

16. (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (SomeActivators AND AllRepressors) OR (AllActivators AND AllRepressors)

17. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (SomeActivators AND AllRepressors) OR (AllActivators AND AllRepressors)

18. (NoneActivators AND NoneRepressors) OR (SomeActivators AND NoneRepressors) OR (AllActivators AND NoneRepressors) OR (NoneActivators AND SomeRepressors) OR (SomeActivators AND SomeRepressors) OR (AllActivators AND SomeRepressors) OR (SomeActivators AND AllRepressors) OR (AllActivators AND AllRepressors)

These functions indeed correspond to the monotonic regulation conditions that are required in the exercise as can be compared below (the functions are in the same order):



Regulation Condition									
0			🟥						
1		🟥	🟥						
2			🟥			🟥			
3		🟥	🟥			🟥			
4			🟥			🟥			🟥
5		🟥	🟥			🟥			🟥
6		🟥	🟥		🟥	🟥			
7		🟥	🟥		🟥	🟥			🟥
8		🟥	🟥		🟥	🟥		🟥	🟥
9	🟥	🟥	🟥						
10	🟥	🟥	🟥			🟥			
11	🟥	🟥	🟥		🟥	🟥			
12	🟥	🟥	🟥	🟥	🟥	🟥			
13	🟥	🟥	🟥			🟥			🟥
14	🟥	🟥	🟥		🟥	🟥			🟥
15	🟥	🟥	🟥	🟥	🟥	🟥			🟥
16	🟥	🟥	🟥		🟥	🟥		🟥	🟥
17	🟥	🟥	🟥	🟥	🟥	🟥		🟥	🟥