Boolean Networks

Krzysztof Zabłocki, Sarbel Fadel, Ryan John

2024-05-29

Exercise simple network

```r
# Load the network
simple <- loadNetwork("simplenetwork.txt")

# Check the contents of the variable 'simple'
print(simple)
```

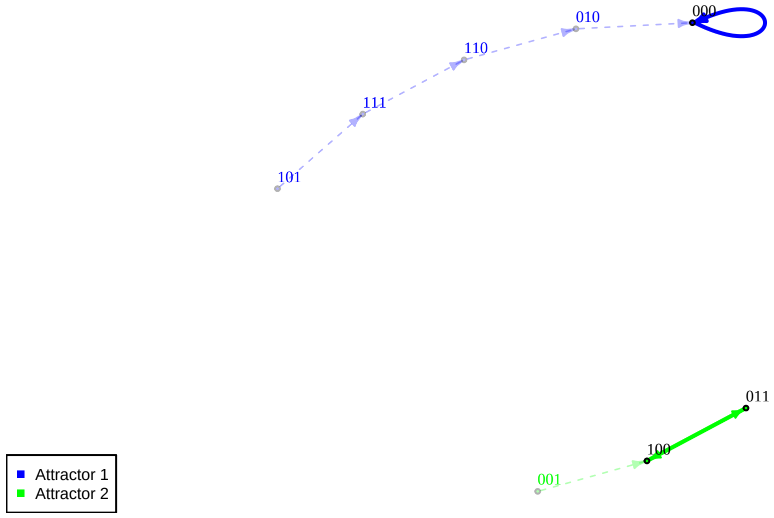
```
## Boolean network with 3 genes
##
## Involved genes:
## a b c
##
## Transition functions:
## a = c
## b = a
## c = a & !b
```

```r
# Get the attractors
attr <- getAttractors(simple)
```

```r
# Check the contents of the variable 'attr'
print(attr)
```

```
## Attractor 1 is a simple attractor consisting of 1 state(s) and has a basin of 5 state(s):
##
##  |--<--|
##  V     |
##  000   |
##  V     |
##  |-->--|
##
##
## Genes are encoded in the following order: a b c
##
## Attractor 2 is a simple attractor consisting of 2 state(s) and has a basin of 3 state(s):
##
##  |--<--|
##  V     |
##  100   |
##  011   |
##  V     |
##  |-->--|
##
##
## Genes are encoded in the following order: a b c
```

```r
# Display the network using plotStateGraph
plotStateGraph(attr, drawLabels = TRUE)
```



Question 1: what are the various dots and lines that you see?

The lines are steps in the trajectories, the dots are the states of the network.

Question 2: how many basins of attraction are there? Why?

1. (Fixed-Point) Attractor 1 has 5 basin of attraction
2. (Cyclic) Attractor 2 has 3 basin of attraction There are 8 attractors because there are 8 (2ˆ3) possible states in the network.

Exercise insulin network

```
# Load the insulin regulatory network
insulin <- loadNetwork("insulin.txt")

# Check the contents of the variable 'insulin'
print(insulin)
```

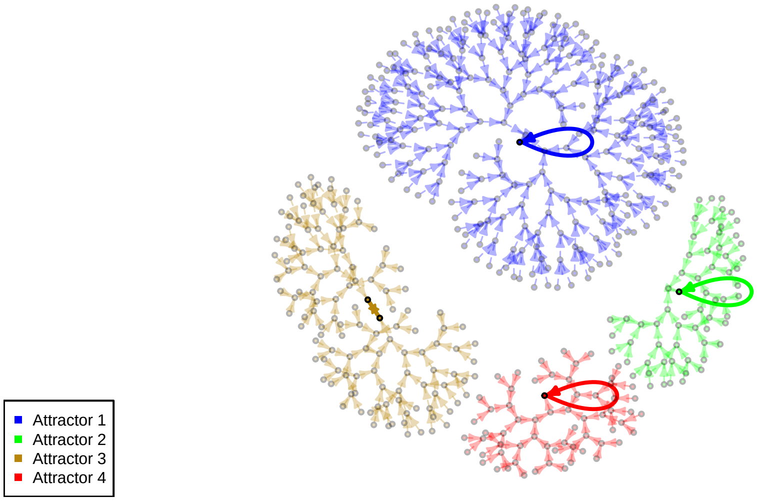
```
## Boolean network with 9 genes
##
## Involved genes:
## extracellular_glucose GLUT2 cellular_glucose cellular_ATP K_channel cellular_K calcium_channel cellul
##
## Transition functions:
## extracellular_glucose = extracellular_glucose
## GLUT2 = extracellular_glucose & cellular_glucose
## cellular_glucose = GLUT2
## cellular_ATP = cellular_glucose
## K_channel = !cellular_ATP
## cellular_K = !K_channel
## calcium_channel = cellular_K
## cellular_Ca = calcium_channel
## insulin_release = cellular_Ca
```

```
# Find the attractors of the network
attr <- getAttractors(insulin)
```

```r
# Check the contents of the variable 'attr'
print(attr)
```

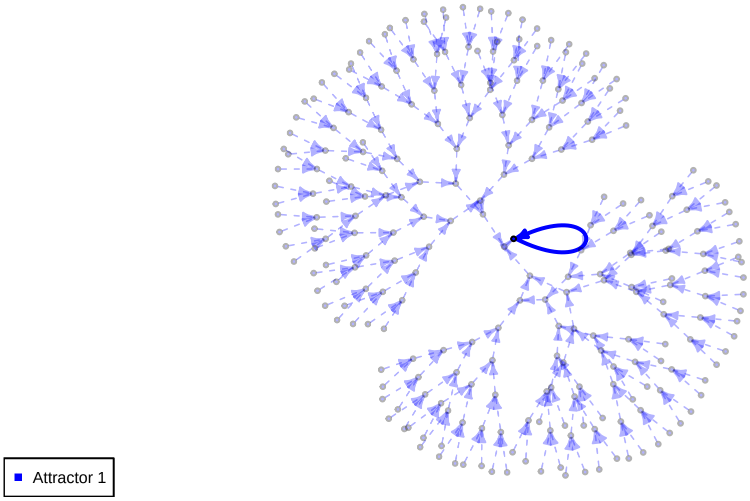
```
## Attractor 1 is a simple attractor consisting of 1 state(s) and has a basin of 256 state(s):
##
##  |--<--------|
##  V           |
##  000010000   |
##  V           |
##  |-->--------|
##
##
## Genes are encoded in the following order: extracellular_glucose GLUT2 cellular_glucose cellular_ATP
##
## Attractor 2 is a simple attractor consisting of 1 state(s) and has a basin of 64 state(s):
##
##  |--<--------|
##  V           |
##  100010000   |
##  V           |
##  |-->--------|
##
##
## Genes are encoded in the following order: extracellular_glucose GLUT2 cellular_glucose cellular_ATP
##
## Attractor 3 is a simple attractor consisting of 1 state(s) and has a basin of 64 state(s):
##
##  |--<--------|
##  V           |
##  111101111   |
##  V           |
##  |-->--------|
##
##
## Genes are encoded in the following order: extracellular_glucose GLUT2 cellular_glucose cellular_ATP
##
## Attractor 4 is a simple attractor consisting of 2 state(s) and has a basin of 128 state(s):
##
##  |--<--------|
##  V           |
##  110111010   |
##  101000101   |
##  V           |
##  |-->--------|
##
##
## Genes are encoded in the following order: extracellular_glucose GLUT2 cellular_glucose cellular_ATP
```

```r
# Plot the state transition graph to visualize the basin of attraction
plotStateGraph(attr, drawLabels = FALSE)
```



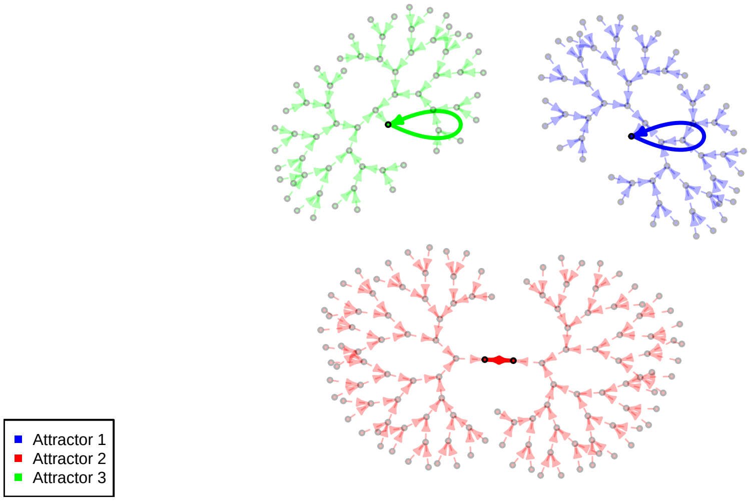
```r
# Force the network into states without extracellular glucose
off <- fixGenes(insulin, "extracellular_glucose", 0)

# Find and plot attractors for the modified network
attr_off <- getAttractors(off)
plotStateGraph(attr_off, drawLabels = FALSE)
```



```
# Force the network into states with always extracellular glucose
on <- fixGenes(insulin, "extracellular_glucose", 1)

# Find and plot attractors for the modified network
attr_on <- getAttractors(on)
plotStateGraph(attr_on, drawLabels = FALSE)
```


Attractor 1
Attractor 2
Attractor 3

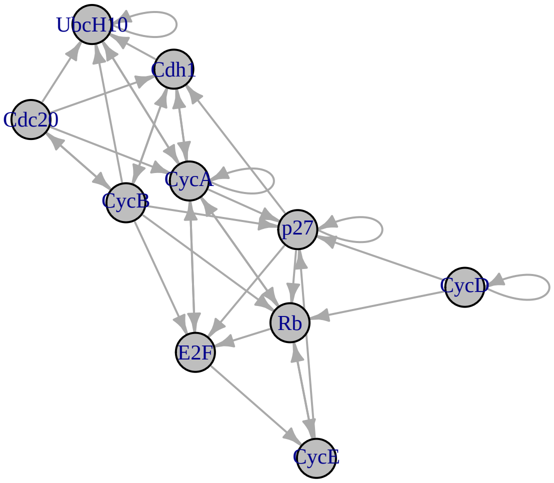
Question 3: What do you think is “wrong” or unrealistic in the model?

1. **GLUT2 Expression**:
   - **Model**: `GLUT2` is expressed when both `extracellular_glucose` and `cellular_glucose` are present (`GLUT2, extracellular_glucose & cellular_glucose`).
   - **Reality**: GLUT2 is primarily regulated by extra-cellular glucose levels and is responsible for the uptake of glucose into the cell. Its expression is not directly dependent on the presence of cellular glucose.
2. **K Channel Regulation**:
   - **Model**: The potassium channel (`K_channel`) is only influenced by cellular ATP levels (`K_channel, !cellular_ATP`).
   - **Reality**: Potassium channels in pancreatic beta cells are influenced by multiple factors, including ATP levels, but also other signaling molecules and cellular states. The model oversimplifies this regulation.
3. **Calcium Channel Activation**:
   - **Model**: Calcium channels are activated by the presence of cellular potassium (`calcium_channel, cellular_K`).
   - **Reality**: While depolarization due to potassium efflux is a trigger for calcium channel opening, the model should include the intermediate step of membrane depolarization rather than directly linking potassium to calcium channel activation.
4. **Insulin Release**:
   - **Model**: Insulin release is solely dependent on intra-cellular calcium levels (`insulin_release, cellular_Ca`).
   - **Reality**: Insulin release is a complex process involving multiple signaling pathways, including calcium influx but also various other factors such as glucose metabolism, signaling molecules, and intra-cellular vesicle trafficking mechanisms.

The cell cycle network

```
# Load the cell cycle network data from the BoolNet package
data(cellcycle)

# Visualize the network wiring
plotNetworkWiring(cellcycle)
```

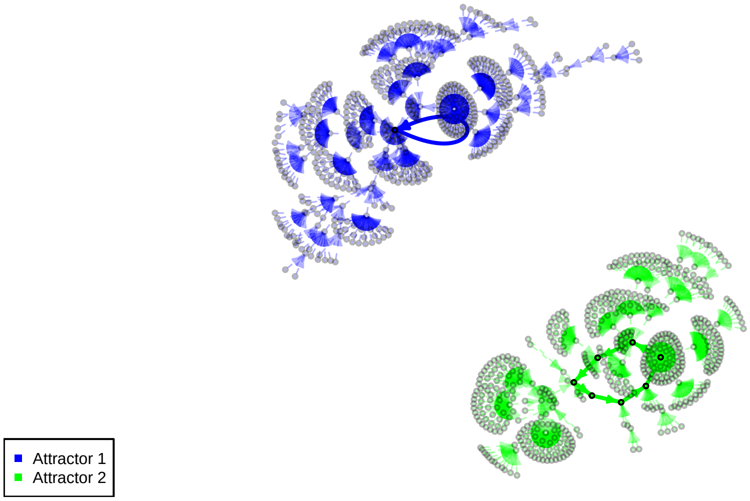


```
# Get the attractors of the network
attr <- getAttractors(cellcycle)
```

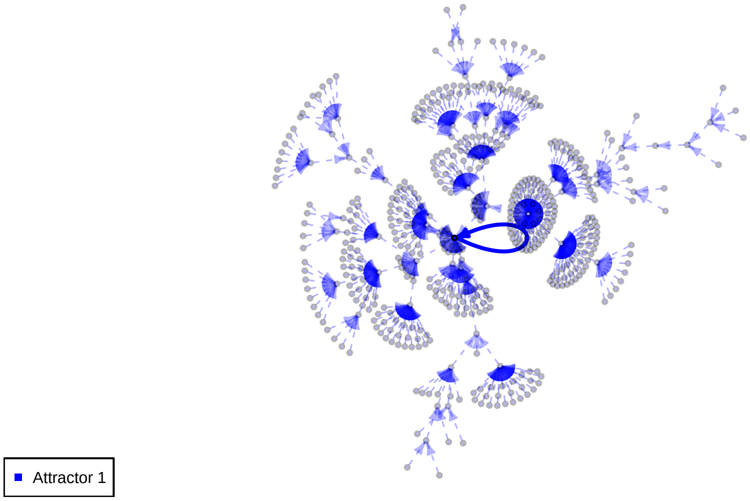
```r
# Check the contents of the variable 'attr'
print(attr)
```

```
## Attractor 1 is a simple attractor consisting of 1 state(s) and has a basin of 512 state(s):
##
##  |--<---------|
##  V            |
##  0100010100   |
##  V            |
##  |-->---------|
##
##
## Genes are encoded in the following order: CycD Rb E2F CycE CycA p27 Cdc20 Cdh1 UbcH10 CycB
##
## Attractor 2 is a simple attractor consisting of 7 state(s) and has a basin of 512 state(s):
##
##  |--<---------|
##  V            |
##  1001100000   |
##  1000100011   |
##  1000101011   |
##  1000001110   |
##  1010000110   |
##  1011000100   |
##  1011100100   |
##  V            |
##  |-->---------|
##
##
## Genes are encoded in the following order: CycD Rb E2F CycE CycA p27 Cdc20 Cdh1 UbcH10 CycB
```

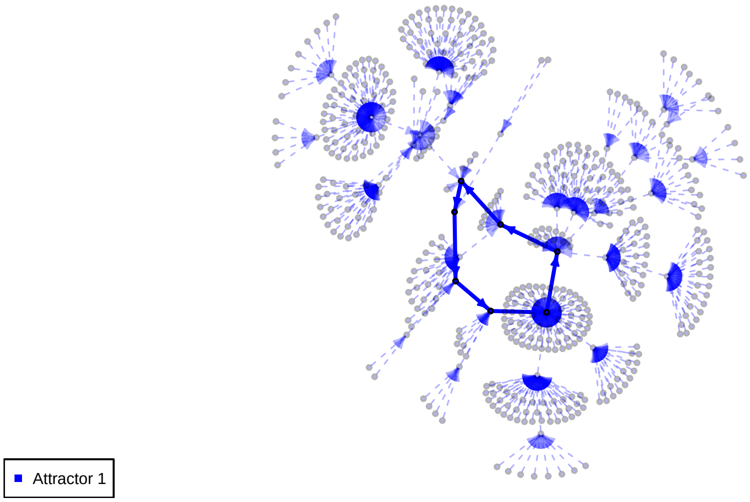
```r
# Plot the state transition graph to visualize the basin of attraction
plotStateGraph(attr, drawLabels = FALSE)
```



```
# Force the network into states with different inputs (e.g., CycD as an input, constant). This allows u
off <- fixGenes(cellcycle, "CycD", 0)
attr_off <- getAttractors(off)
plotStateGraph(attr_off, drawLabels = FALSE)
```



```r
on <- fixGenes(cellcycle, "CycD", 1)
attr_on <- getAttractors(on)
plotStateGraph(attr_on, drawLabels = FALSE)
```



Question 4: Show this graph. What are the dots and lines (solid and dashed) in this graph?

The lines are steps in the trajectories, the dots are the states of the network. The dashed lines represent the transitions between unstable states, while the solid lines represent the self-loops or cycles how we say in BIO.

Question 5: What do you think the “attractors” of this network represent in biological terms?

They represent some signaling proteins.

Question 6: Show the two state graphs.

Done above :))

Question 7: What is the biological interpretation of the two state graph plots?

The one without `CycD` as an input shows that the network is in a stable state, while the one with `CycD` as an input shows that the network is in an cycling. This means that `CycD` is a crucial factor for the cell cycle to start and proceed.

Exercise simple cycle

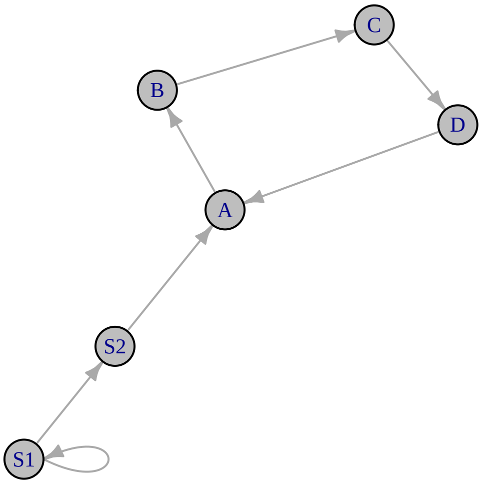
```
# Load the network
simpleC <- loadNetwork("simplecycle.txt")
```

```
## Warning in loadNetwork("simplecycle.txt"): There is no transition function for
## gene "S1"! Assuming an input!
```

```
# Check the contents of the variable 'simple'
print(simpleC)
```

```
## Boolean network with 6 genes
##
## Involved genes:
## S2 A B C D S1
##
## Transition functions:
## S2 = S1
## A = D & !S2
## B = A
## C = B
## D = C
## S1 = S1
```

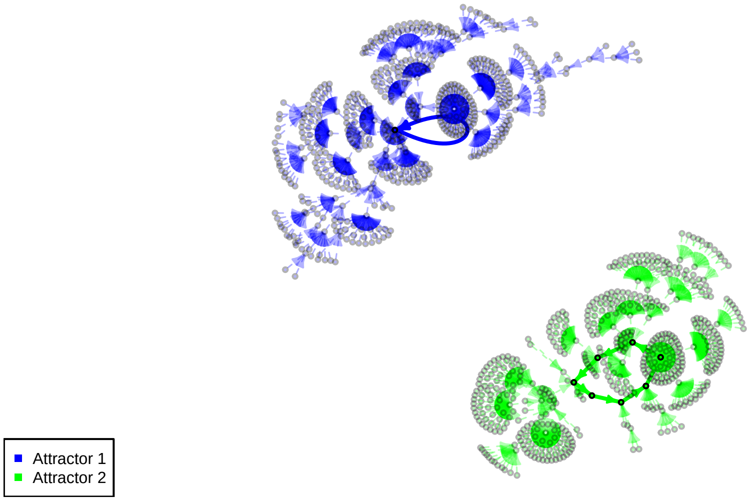
```
# print the network wiring
plotNetworkWiring(simpleC)
```


C
B
D
A
S2
S1

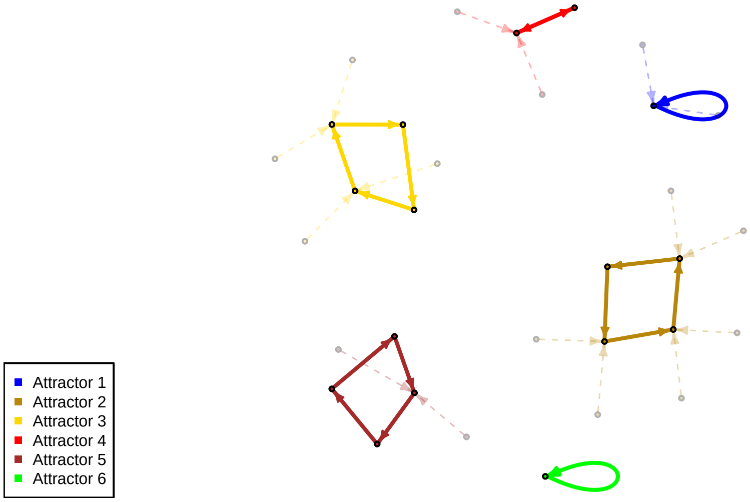
```r
# Check the contents of the variable 'attr'
print(attr)
```

```
## Attractor 1 is a simple attractor consisting of 1 state(s) and has a basin of 512 state(s):
##
##  |--<---------|
##  V            |
##  0100010100   |
##  V            |
##  |-->---------|
##
##
## Genes are encoded in the following order: CycD Rb E2F CycE CycA p27 Cdc20 Cdh1 UbcH10 CycB
##
## Attractor 2 is a simple attractor consisting of 7 state(s) and has a basin of 512 state(s):
##
##  |--<---------|
##  V            |
##  1001100000   |
##  1000100011   |
##  1000101011   |
##  1000001110   |
##  1010000110   |
##  1011000100   |
##  1011100100   |
##  V            |
##  |-->---------|
##
##
## Genes are encoded in the following order: CycD Rb E2F CycE CycA p27 Cdc20 Cdh1 UbcH10 CycB
```

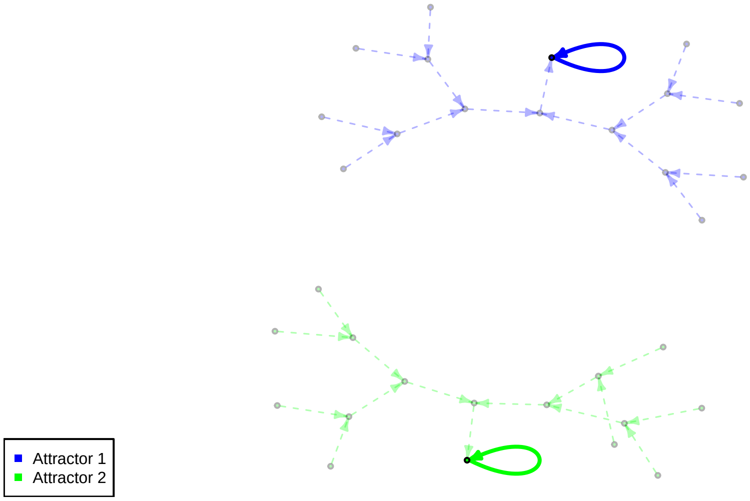
```r
# Display the network using plotStateGraph
plotStateGraph(attr, drawLabels = FALSE)
```



```
offC <- fixGenes(simpleC, "S1", 0)
attr_offC <- getAttractors(offC)
plotStateGraph(attr_offC, drawLabels = FALSE)
```



```r
onC <- fixGenes(simpleC, "S2", 1)
attr_onC <- getAttractors(onC)
plotStateGraph(attr_onC, drawLabels = FALSE)
```


■ Attractor 1
■ Attractor 2