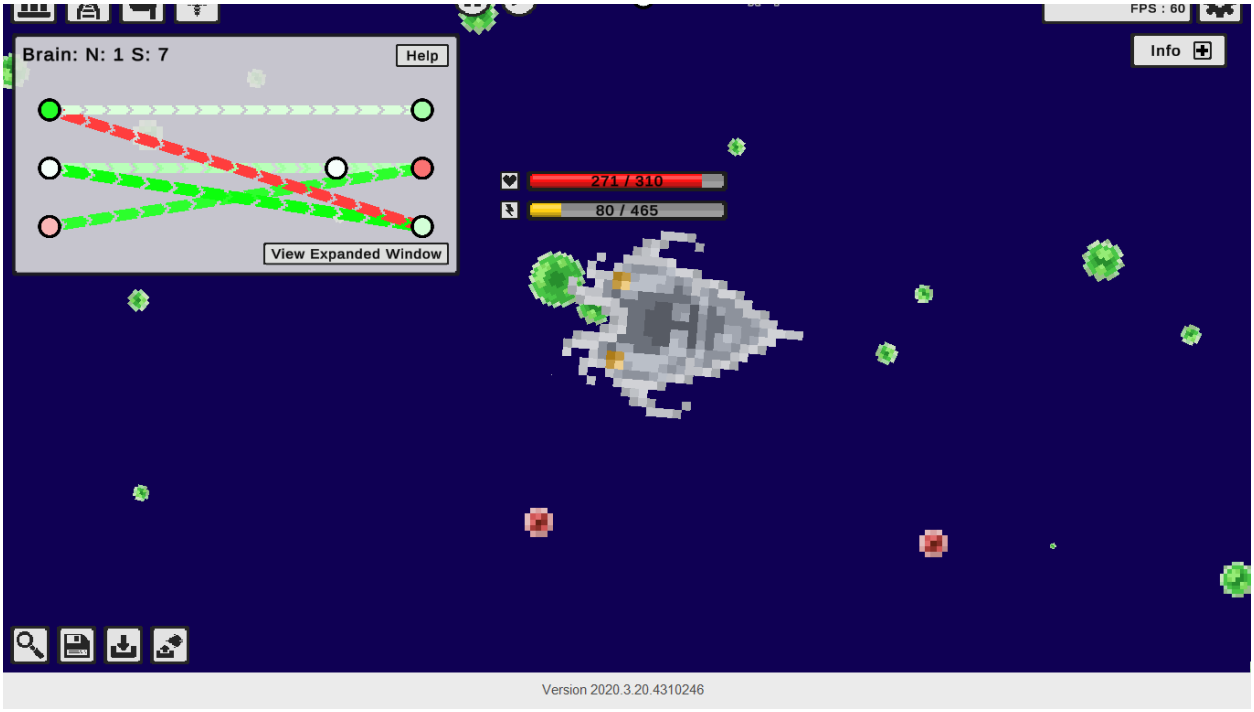
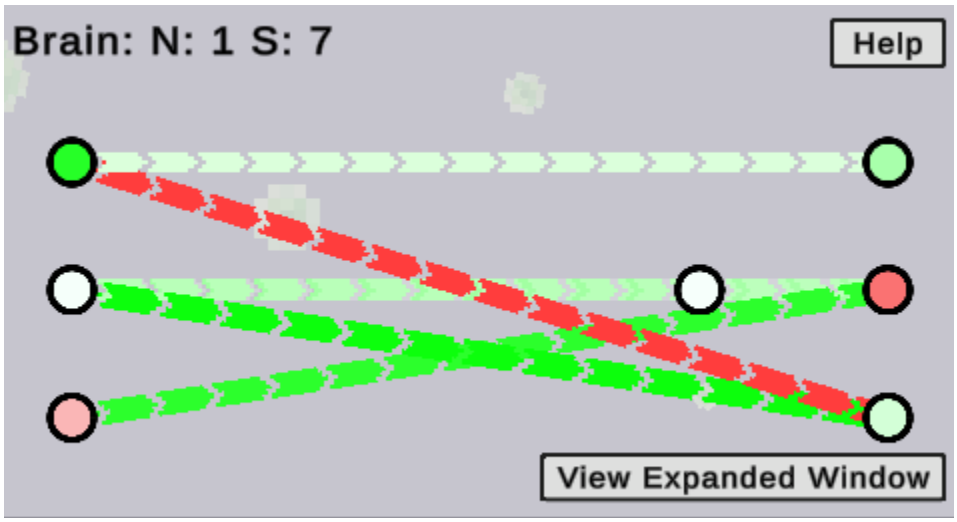
So you want to understand how the Bibites think, Eh? Well tough luck, I'm really bad at making understandable things. I'll still try my best to give you a clearer picture

So let's go, here's the full screen :



Oh god, so much stuff

For now, we'll focus on that small area



Still A lot, eh ? Let me annotate it and we'll go through it procedurally.

In order :

1. **Life**:

How much health it has
[Present health] / [Max health]

Is it sick? Is it in good shape? This metric will tell you!
This bibite is not in good shape :(

2. **Energy** :

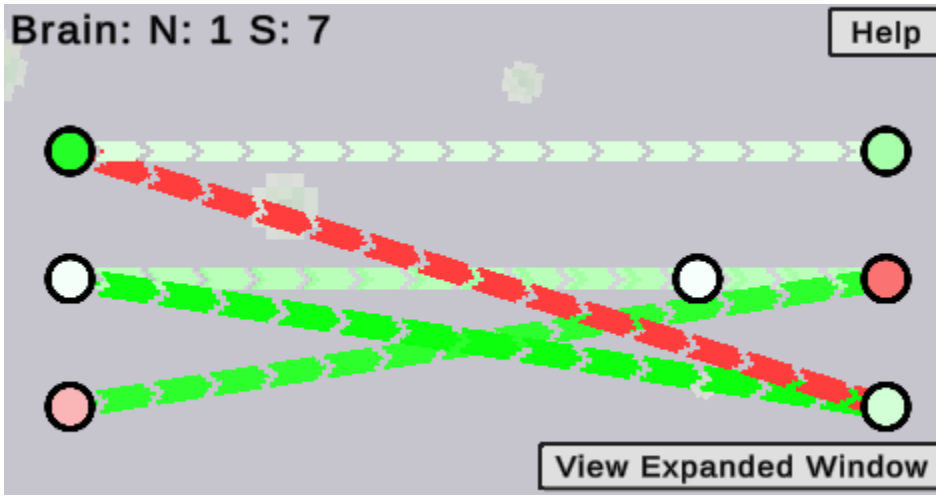
How much Energy this bibite has left. Energy is used by everything. Moving, thinking, laying an egg, you name it!
When a bibite doesn't have enough energy to do something, it will

use health instead. +E([Energy amount])

3. Generation:

It simply tells us how many ancestors it has.

4. Brain structure:



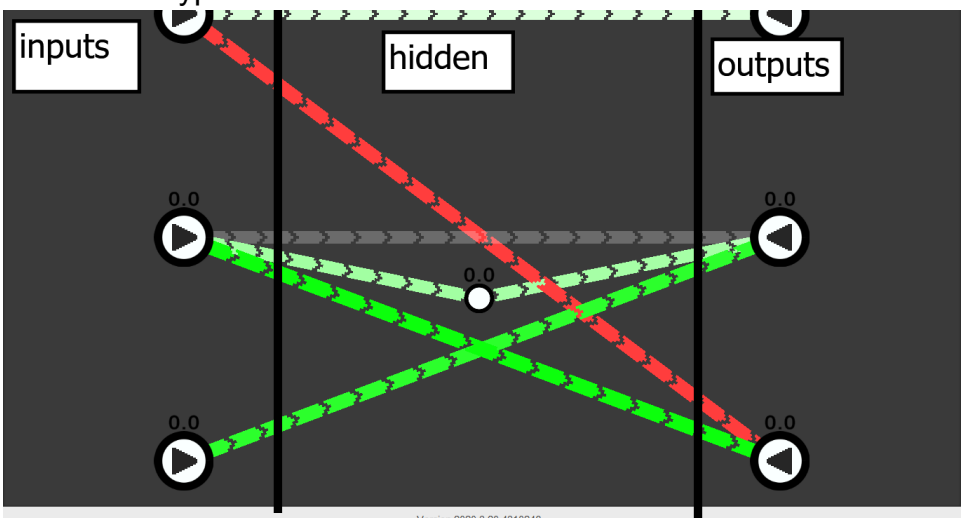
It tells us the numbers of hidden neurons and synapses in the brain.

Brain : N: [Number of hidden neurons] S: [number of synapses]

But what are Synapses and Neurons ??

5. Neurons :

There are 3 types of neurons :



* First of all, you only see the neurons which have connections and are relevant *

Input Neurons are the senses of our Bibites and are fixed (all bibites have the same 23) : They go as follows :

1. Constant: always equal to one. Useful for constant behavior
2. Energy: How much energy the bibite has
3. Maturity: How mature the bibite is (1 when can reproduce)
4. Liferatio: How full the health of the bibite is
5. Spee: How fast the bibite is going
6. Dist2bibite: How close the closest bibite it sees is going
7. Angle2bibite: The angle to the closest bibite it sees
8. Dist2pellet: How close the closest pellet sees is going
9. Angle2pellet: The angle to the closest pellet it sees
10. nBibite: Number of bibites it sees
11. nPellet: Number of pellets it sees
12. Rbibite: The Red gradient of the closest bibite it sees

13. Gbibite: The Green gradient of the closest bibite it sees
14. Bbibite: The Blue gradient of the closest bibite it sees
15. clkTic: Internal timer (1s on, 1s off (actual period decided by genes)) 16. clkMinut: kind of like a chronometer, counts time, gets reset by an output neuron 17. clkTimeAlive: Time passed since the bibite was born.
18. PheroSense1: How much Red pheromones is it sensing.
19. PheroSense2: How many Green pheromones is it sensing.
20. PheroSense3: How many Blue pheromones is it sensing.
21. AngleToPhero1: Angle to the Red pheromones it is sensing.
22. AngleToPhero2: Angle to the Green pheromones it is sensing
23. AngleToPhero3: Angle to the Blue pheromones it is sensing

PFIEW...

Now the Output neurons (the actions a bibite can do, again all bibites have the same 16) : * DISCLAIMER: All output neurons (unless specified) have a default value of 0.5 *

1. Forward: How much does it want to go forward?
2. Backward: How much does it want to go backward?
3. turnLeft: How much does it want to turn Left?
4. turnRight: How much does it want to turn Right?
5. Want2lay: Does it want to lay an egg? (>=0.5 is yes)
6. Want2eat: Does it want to eat the pellets it touches? (>=0.5 is yes)
7. Want2sex: Not used anymore (but was used for sexual reproduction)
8. Grab: Does it want to grab (with its mouth) what it touches? (>=0.75 is yes) 9. clkReset: Does it want to reset its internal chronometer? (>=0.75 is yes)
10. Pherout1: How many red pheromones is it producing? (default is 0)
11. Pherout2: How many green pheromones is it producing? (default is 0) 12. Pherout3: How many blue pheromones is it producing? (default is 0) 13. Want2grow: How much does it want to grow (mature)?
14. want2Heal: How much does it want to heal?
15. want2Attac: How violently does it want to attack others?

God my fingers hurts

Finally, there are Hidden neurons, but these are just neurons that can be evolved in the brain, those are specific to every individual (or species) and are not directly used by the body.

The name of that neuron can be seen when you hover over it **:**

They each hold a value, and their color will change

according to the value: From red (negative)  to

Green  (positive values) All that is good but how does

a neuron get attributed a value?

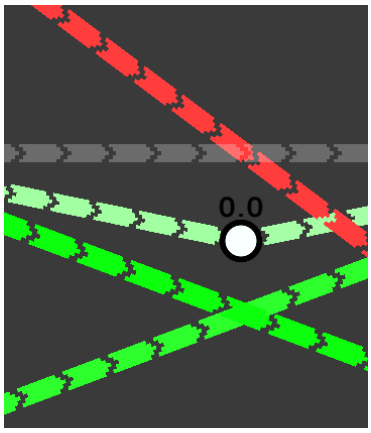
That's where synapses come into play :

6. Synapses :

Synapses are the stringy bits that link neurons together.

They also have a value that is visible when you hover over it through the color. Again, red is a negative value, and green a positive value. In synapses, that value is called **Strength**

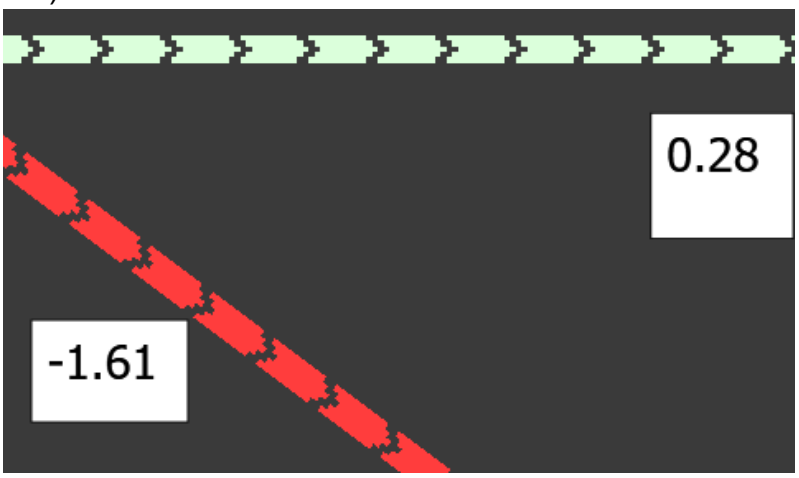
Synapse can also be disabled, they are just ghosts and are not useful for the bibite (but can easily be mutated on or off by the bibites children). They look transparent:



And well it’s simple, a neuron’s value is equal to the SUM of all the synapse’s strength coming into it multiplied by the neuron value it’s from.

Sounds complicated? It’s not that much, I’m just bad with words.

Here’s an example : (the direction of a synapse can always be seen through with its text)



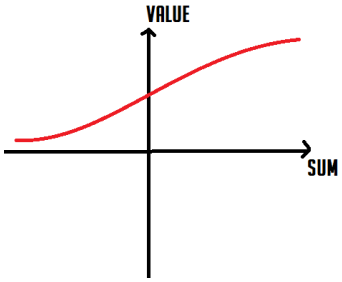
8. Neuron functions :

Before setting their actual value, a neuron will apply its activation function to the previously calculated sum. The different functions are :

Before setting their actual value, a neuron will apply its activation function to the previously calculated sum. The different functions are :

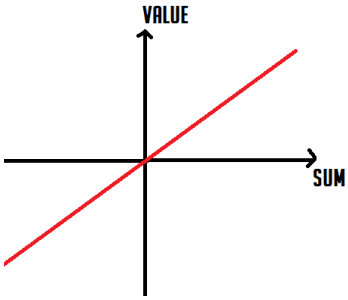
1. INP: Input
   Value is set by one of the body's systems.

2. SIG: Sigmoid (that's why the value is 0.5 by default, most outputs are sigmoid)



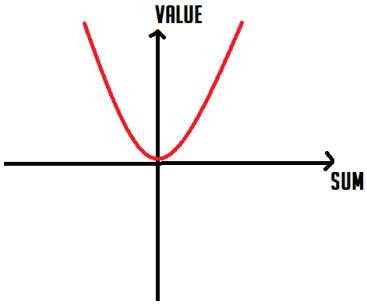
Useful to chose between 0 and 1

3. LIN: Linear



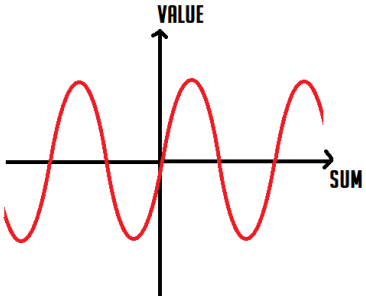
Useful when somethings can have any value

4. SQR: Square



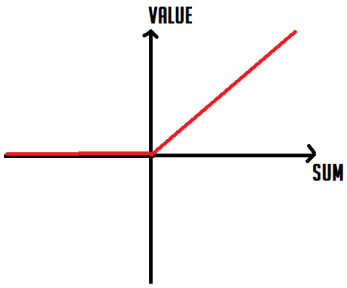
Useful to draw out differences between low and high signal

5. SIN: Sinusoidal



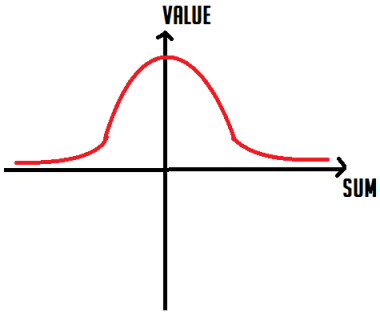
Useful for cyclical behaviors

6. REL: Rectilinear



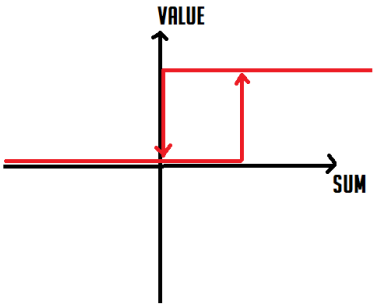
Useful when something can't have a negative value

7. GAU: Gaussian



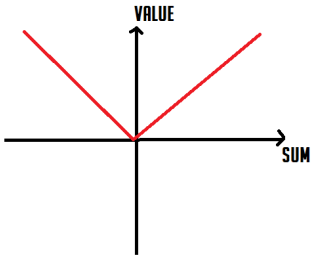
Useful when you want to be selective about something

8. LAT: Latch (kinda memory)



Useful to memory lol

9. ABS: Absolute



You’re starting to get it

PFIEEWW^2, that was something...

But at least now you should understand how it works a little better? Maybe?