The Conscious Eater

Amanda Breton, Ian Jackson, Mackenzie Looney

Intro and Aims

Overall, this project aims to implement an improved Braitenbug using new knowledge of the Winner Take All Circuit for a decision circuit and a Hopfield Network to create learning from weight changes. In Project 1, a "Picky Eater" Bug was created by team member Mackenzie Looney. Here, we seek to improve the "Picky Eater" by turning it into a "Conscious Eater", by using a Hopfield Network to make the Braitenbug "think" about what kind of food it is looking at, and make a decision using the Winner Take All Circuit to decide whether to exhibit cowardly or aggressive behavior towards the food. More specifically, we aim to:

- Encode a food information array with a food piece when it is spawned on the bug map.
- Use a Hopfield Network to error correct the food information array to either "good food" or "bad food" for decision making.
- Use the Winner Take All Circuit to inhibit the food type behavior not chosen by the Hopfield Network.
- The winning neuron will inhibit the losing neuron and excite the motor neurons, to exhibit either aggressor behavior towards "good food" or coward behavior towards "bad food."
- Distance of the food will still be used as a stimulus.

Background/Preliminary Results

For the Braitenbug, a combination of food sensors and motor sensors will be used. There will be a left and right food sensor that will sense the distance of the food as a stimulus. There will be both coward and aggressor neurons, left and right for each. The food distance stimulus will be sent from the food sensor neurons to the aggressor and coward neurons. Additionally, when food is spawned on the "bug map", a "food information array" will also be generated. For example, this food array could be: [1 1 1 1], [-1 -1 -1 -1], or even [1, -1, 1, -1]. This food array will be the x_test in the Hopfield Network, and the Hopfield network will attempt to do an error correction on the x_test/food array to either [1 1 1 1] or [-1 -1 -1 -1], either "good food" or "bad food" respectively. Depending on what the Hopfield Network error corrects to, a K1 and K2 stimulus will be sent out, which will be used for the WTA Circuit. The WTA circuit will then determine whether Aggressor Behavior or Coward Behavior will be exhibited by the bug. The winning behavior will inhibit the losing behavior and the winning behavior will excite the motor neurons for the Braitenbug to move accordingly.

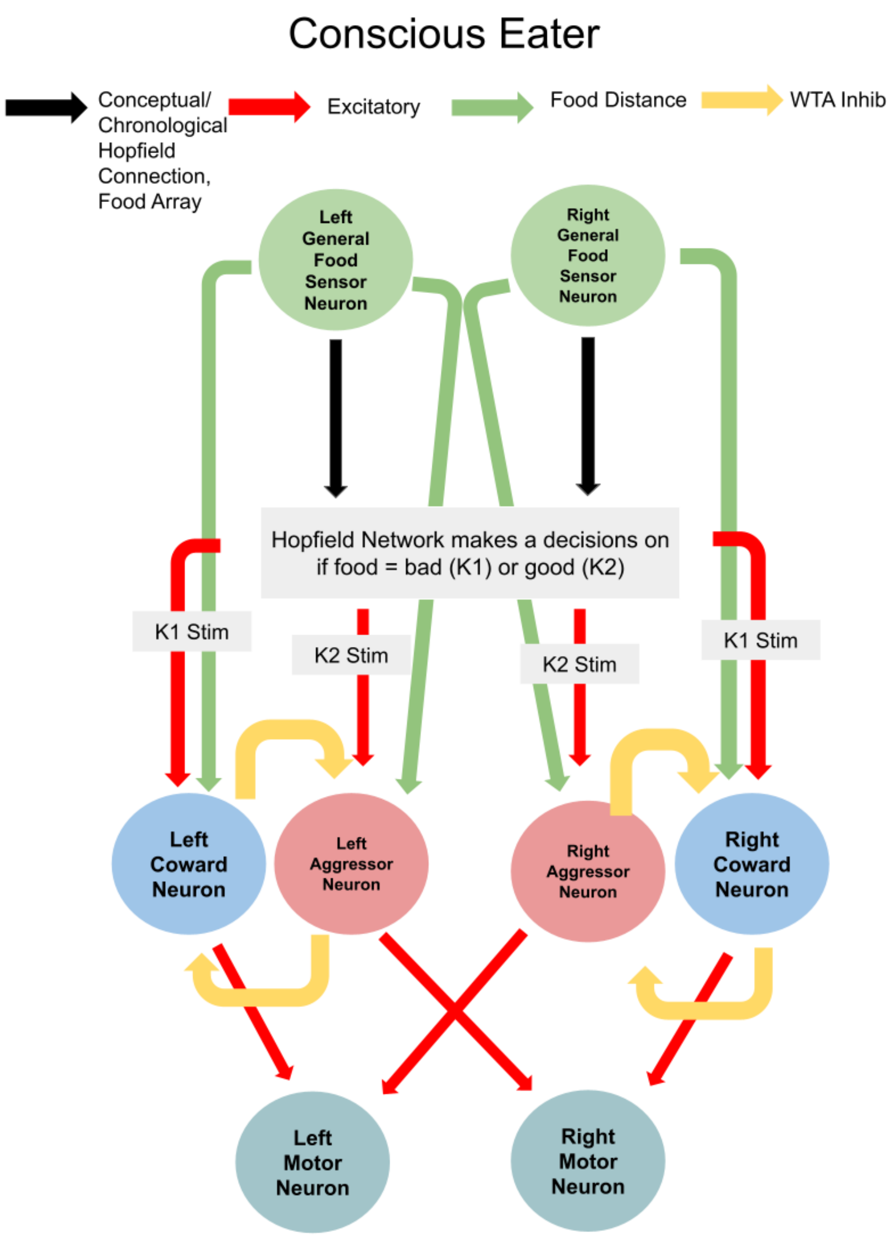

Figure 1. Circuit for "Conscious Eater" Braitenbug.

Approach

To create the "Conscious Eater", Low Spiking Threshold Izhikevich neurons will be used with alpha synapses for both the food sensor neurons and the locomotion neurons. We plan to create the Hopfield Network using a custom Python implementation based on the script we used in class for Exploration 4.

With a Hopfield Network trained on both good and bad food, we expect to observe behavior that changes based on the type of food and the distance of that food from the bug. For example, if a bug encounters a "good" food item that is close by, it's expected that the bug will move rapidly toward it. On the other hand, a close by "bad" food item would cause the bug to quickly move away from it. This behavior should change when the Hopfield Network is trained on only one type of food, creating a biased bug. An example of this would be a Hopfield Network given memory of only good food, causing the output of the network to more likely converge on a state that causes aggressor behavior. Thus, even if the food is slightly bad, the instability of this state would cause the bug to eventually still run toward the food. Of course, regardless of the type of food, we expect the distance to have a positive relationship with the strength of stimulation and, therefore, speed of movement.

Milestones:

1. Build base network of sensory and motor neurons with cross-inhibition for WTA → **November 18th**
2. Implement Hopfield Network → **November 18th**
3. Implement extensions/modifications → **December 2nd**
4. Implement environmental sensors and temperature gradient → **December 2nd**