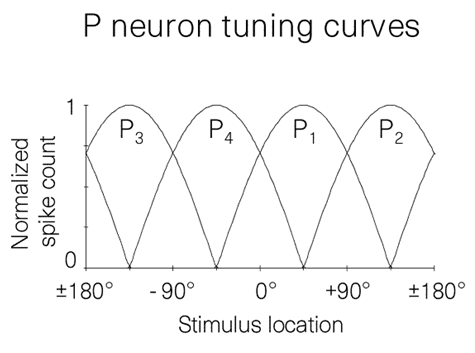
**Graded assignment 1 – 4 credit supplementary**

At this time, you all have implemented leaky integrate and fire neurons and used them to run simple simulations. In this assignment you will use your coding experience and what you have learned about population coding to implement a simple network similar to the leech bend-away-from-touch network that we designed in class.

1. Create 4 sensory neurons with sensory receptive fields in response to touch which span 360 degrees so that each stimulus is covered. The figure below is an example but you don’t have to copy this. You don’t need to implement real neurons, just a representation of spikes per unit time as a function of angle. Most probably these will look more triangular than sinusoidal and that is fine. These four neurons represent four “input lines” to your network of motor neurons. Use a 1 second touch stimulus to create the spike trains. Plot spike trains of one neuron in response to at least 6 angles to show how this works.



1. **First** implement a population vector calculation that allows you to predict the angle of any stimulus from the firing rates of your sensory neurons. Test your population vector calculation on three different angles as well as a situation in which two angles spanned by separate sensory neurons are touched at the same time (similar to what was described in the paper). You can do this what I call “by hand”, just write it all out or use vector algebra.
2. **Second** create a network of 12 postsynaptic neurons receiving spiking inputs from these four sensory neurons and which have receptive fields with a better resolution than the sensory neurons. You want to try and keep their receptive fields as narrow as possible. These should be LIF neurons and should all have the same time constant and threshold but receive inputs from these four sensory neurons with specific synaptic weights. Hint: I would figure out first on a piece of paper or with just one neuron how to choose the amplitude of the incoming spikes, the weights and time constant etc to make this work. If you want narrow receptive fields you want the neurons to have a relatively high threshold for. Use the information you gathered in HW3 and in the written portion of the assignment to guide you in parameter choice.
3. **Third**, as output, plot the responses (spikes per unit of stimulus application) as a function of angle for each of the motor neurons.
4. **Fourth**, now you want to modify your network in such a manner that only one motor neuron per stimulus is active such as to not confuse the animal. Use inhibitory synapses to do this.
5. Please submit your code as well as a detailed explanation of what you did !