Exercise 6, Jan 12, 2017 to be handed in January 19

The Amari oscillator is a system of an excitatory neuron, u, coupled to an inhibitory neuron, v as follows:

$$\begin{aligned}\tau\dot{u}(t) &= -u(t) + h_u + c_{uu}\,\sigma(u(t)) + c_{uv}\,\sigma(v(t))\\ \tau\dot{v}(t) &= -v(t) + h_v + c_{vu}\,\sigma(u(t))\end{aligned}$$

In words: The excitatory neuron is self-excitatory ($c_{uu} > 0$) and excites the inhibitory neuron ($c_{vu} > 0$), while the inhibitory neuron inhibits the excitatory neuron ($c_{uv} < 0$). The sigmoidal function can be approximated as a step-function:

$$\sigma(u) = \begin{cases} 1 & \text{for } u > 0 \\ 0 & \text{for } u < 0 \end{cases}$$

You can read up on this oscillator in the paper by Amari, S. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics* **27**:7787. The relevant section is 8.1 there (the rest of the paper discusses the peak solution of neural fields and may be of interest as well).

1. Analyze this dynamics exploiting the piece-wise linear nature of the vector-field (in each of the four quadrants, the equation is linear). By writing down these four linear equations, you can determine the fixed point of the linear dynamics valid in each quadrant (the fixed point may lie outside that quadrant, see next).

2. According to Amari, an oscillation occurs if for each quadrant, the fixed point of the dynamics lies in the neighboring quadrant along a counter-clockwiser order. This will drive the activation state into the neighboring quadrant so that the trajectory will go counter-clockwise from quadrant to quadrant (Figure 8 in the paper). Derive conditions for the resting levels and coupling strengths that put the fixed points into the right quadrant.

3. Simulate the Amari oscillator using `launcherTwoNeuronSimulator` from the COSIVINA package. You can use the left hand panel to visualize the fixed points. The parameters must be chosen in the top right slider panel, so that the fixed points lie in the right quadrant. (The sigmoid is not a step function, you can make it steeper under "advanced"). Try to make the oscillations larger or smaller in amplitude. Making them faster is harder.

Bonus Can you make the system generate *active transients*? Use Amari's paper to understand what that is.