CIS6930/4930 Intro to Computational Neuroscience Fall 2008
Home Work Assignment 5:
Due Thursday 12/11/08 in my office

The purpose of the first part of the assignment is to appreciate the complexity that can result from very simple dynamical systems.

1. Simulate the time course of the following parameterized class of discrete dynamical systems. The dynamics is that of a single variable $x \in [0,1]$ that is updated by the equation $f(x_{n+1}) = a * x_n * (1 - x_n)$, where a lies in the range $[0,4]$. Note that as long as $a \in [0,4]$, $f(x) \in [0,1]$ if $x \in [0,1]$. Hence the sequence of points $x_0, f(x_0), f(f(x_0)), ...$ never leaves the unit interval.

   Your job is to plot the non-wandering (also called recurring) set for the dynamical system for a range of values of the parameter a.

   Start with $a = 0$, choose a random point $x_0 \in [0,1]$, and run the dynamical system for $5,000$ points. Throw away the first $1,000$ points (which are presumably the transient points until the system settles onto the non-wandering set) and plot the rest on the y-axis. Do this for several random initializations of x_0. Next increment a by 0.001, choose a new random starting point x_0, and do the same, and so on and so forth until you reach $a = 4.0$. (a is plotted along the x-axis).

   One would assume that the non-wandering points for successive values of a should look similar. Do they? In case you find something interesting happening in a certain range for a, zoom into that range and change a by smaller increments.

This next project is not for the faint of heart. There will be *no* deduction of points if you can not complete it. It is about first constructing a random cortical network of spiking neurons and stabilizing its activity using external input in the range 0-50 Hz. In the second part, you computationally ascertain whether the dynamics of the system is *sensitive to initial conditions*.

1. Begin by constructing a network of 1000 neurons, 80% of which are randomly chosen to be excitatory neurons and the remaining inhibitory neurons. Now assign 100 synapses (each neuron *receives* inputs from 100 neurons) to each neuron and randomly choose the pre-synaptic neurons for each such post-synaptic neuron.

   Next set up 100 excitatory input neurons that spike Poisson spike trains at a rate of 10 Hz (the neurons spike with a certain fixed probability at each time step). Assign 10 additional synapses on each of the 1000 neurons in the network and randomly choose input neurons to innervate these synapses.

   Next assign 50 variables to each neuron to store the times at which the neuron spiked in the past 100 msec (Naturally, the time bound for the horizon is 100 msec.)

   Finally, build a spike response model for each neuron as follows. The total potential at the soma of a neuron is the sum of the excitatory and inhibitory post-synaptic potentials generated by each spike at each synapse, and the after-hyperpolarization potentials generated by the spikes emitted by the neuron itself. Randomly assign to each synapse a time in the range [0.4,0.9] msec that accounts for the time it takes for a spike to travel down the axon and reach that synapse.

   Use the following functions to model the postsynaptic potentials at the synapses

   Excitatory :

$$\frac{Q}{\alpha\sqrt{t}} exp(-\alpha^2/t) exp(-t/\tau)$$

set Q in the range [5,10], set α in the range [1,2] and $\tau = 20$ msec.

Inhibitory :

$$\frac{-Q}{\alpha\sqrt{t}} exp(-\alpha^2/t) exp(-t/\tau)$$

set Q in the range [30,60], set α in the range [1,1.1] and $\tau = 20$ msec.

AHP:

$$-1000 * exp(-t/1.2)$$

Start with different thresholds (5 mV for excitatory and 10 mV for inhibitory) for each neuron and play with the thresholds so as to stabilize the system to sustained recurrent activity (for at least 10 sec). The time step for your simulations should be no larger than 0.2 msec.

Plot the spike trains for 250 neurons in the population, as well as the total number of spikes generated in the system over the past 100 msec.

2. Now simulate the dynamics of the same system with the same exact initial configuration of spikes **except for** *one* randomly chosen spike perturbed by *one* msec, driven by the *exact same* input drive. Generate a spike raster plot with the following color codes. Color spikes as red if they are generated at the same time by the two systems (original and perturbed). Otherwise color them green/blue depending on which system the spike belongs to. Based on the results of your experiment report whether the dynamics of your system is *sensitive to initial conditions*.