CN510 Assignment 1: the Leaky Integrator

Massimiliano Versace

23 September 2001

1. Introduction

A leaky integrator (LI) is the simplest neuron model able to incorporate the idea of a dynamic membrane potential without including the complexity of ion channels modeling. The LI is defined by the rate of change of the activation *x* which may be interpreted as the membrane potential of a neuron axon at time *t*. Let *I* be the total post-synaptic potential generated by the afferent input to the neuron, *A* (A ≥ 0) a rate constant which depends on some cell membrane parameters (capacitance, resistance), than the change in x activation is given by (1):

$$\frac{d}{dt}x = -Ax + I \tag{1}$$

Equation (1) can thus be interpreted as a simple way to model STM in biological neural networks. The target of this assignment is to study the behavior of such a system.

2. Method

Equation (1) was numerically integrated using Euler's method, given that a "…differential equation can be approximated by a difference equation in the time steps are "small enough" for a good approximation" [1]. This method can be used to estimate *x(t)* at times *t* greater than 0, given values for *A* and *I(t)* and an initial condition *x(0)*. The accuracy of Euler's method can be improved by decreasing the time steps. For this simulation, a time step of 0.01 was used.

3. Results

In the first simulation $A = 1$, $x(0) = 0$, *I* is a step input of 10000 msec, with $I = 5$ from time $t = 1000$ to $t = 6000$, and $I = 0$ at all other times (Figure 1a). Figure 1b illustrates the results of this first simulation. The asymptote of the curve with $A = 1$ is given by solving equation (1) at equilibrium by setting $\frac{d}{dt}x = 0$:

$$0 = -Ax + I \tag{2}$$

$$x = \frac{I}{A} \tag{3}$$

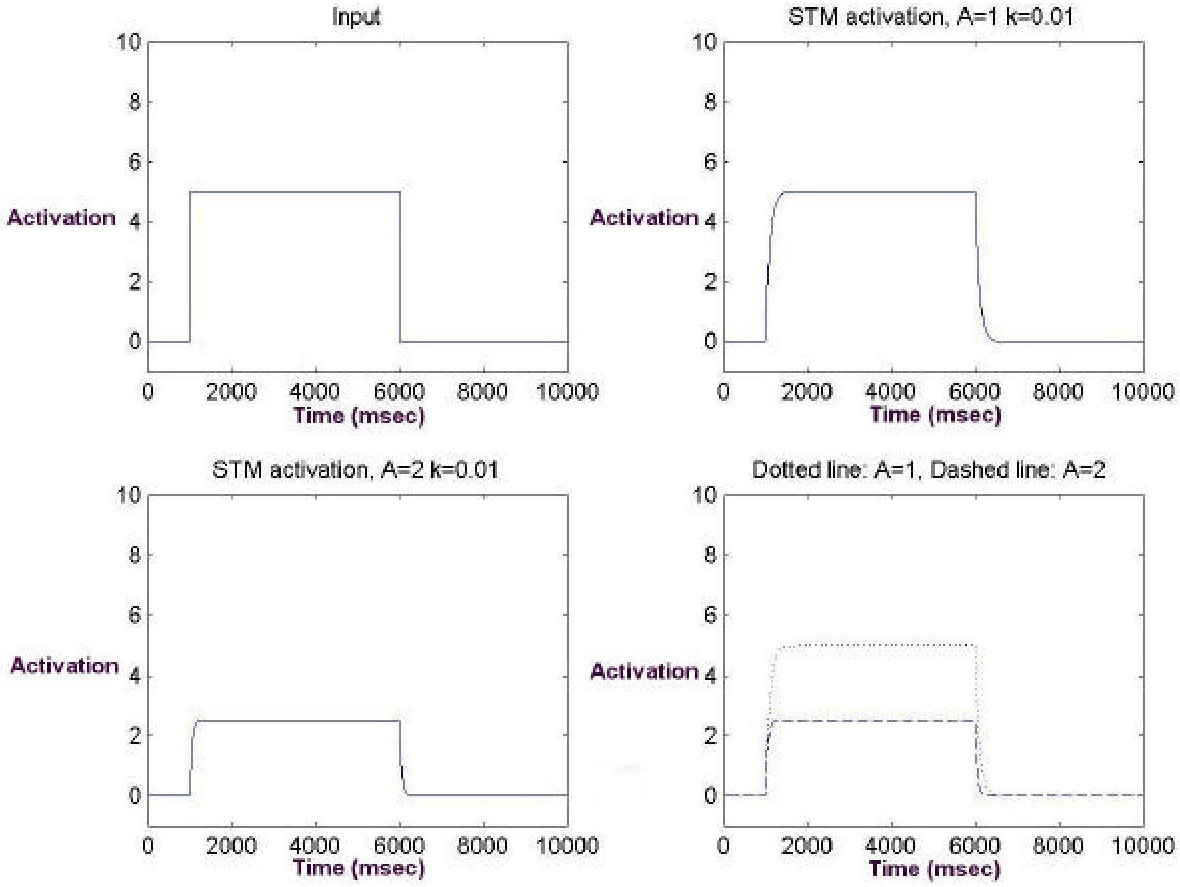

Figure 1. Activity of Leaky Integrator over 10 seconds. a) Input. b) STM activation over time, decay rate A = 1, time step k = 0.01. c) STM activation over time, decay rate A = 2. c) Comparison of STM activation of the same unit with different decay rate (dashed line, A =1, dotted line, A = 2). Time is shown in milliseconds.

In the second simulation the value of A was changed to $A = 2$. The effect of the new decay rate is shown in Figure 1c. In Figure 1d the results of the two simulations are compared in order to appreciate the change in behavior of the same neuron when A is varied. The asymptote of the curve is now changed from 5 to 2.5 (as can be inferred from equation (3)), and the decay to $x = 0$ occurs at a steeper rate with respect to the first simulation. The asymptote of the two activations is consistent with equation (3).

The analytic solution to equation (1) can be found in the following way:

$$\frac{d}{dt}x = -Ax + I \tag{4}$$

$$\frac{d}{dt}x + Ax = I \tag{5}$$

$$e^{At}\frac{d}{dt}x + e^{At}Ax = e^{At}I \quad (6)$$

For the chain rule, $e^{At}\frac{d}{dt}x + e^{At}Ax = \frac{d(xe^{At})}{dt}$:

$$dxe^{At} = Ie^{At}dt \quad (7)$$

$$\int dxe^{At} = \int Ie^{At}dt \quad (8)$$

$$\int dxe^{At} = I\int e^{At}dt \quad (9)$$

$$xe^{At} = I\frac{e^{At}}{A} + C \quad (10)$$

$$xe^{At} = \frac{I}{A}e^{At} + C \quad (11)$$

$$x = \frac{I}{A} + Ce^{-At} \quad (12)$$

Solving C for the initial condition $x = 0$ at time $t = 0$, knowing that $e^0 = 1$, we obtain

$$C = -\frac{I}{A} \quad (13)$$

$$x = \frac{I}{A} + (-\frac{I}{A})e^{-At} \quad (14)$$

$$x = \frac{I}{A}(1 - e^{-At}) \quad (15)$$

4. Discussion

This simulation has shown how Euler's method can be used to numerically solve differential equations. This method is relatively easy to implement and represents a valuable tool for many neural network simulations that include simple network dynamics and do not require a great precision in the analysis of their behavior.

LI is a popular way of introducing dynamics into a model neuron because of its biological plausibility. Moreover, there exist a precise analogy between well-studied physical systems and the LI. LI thus represents a good tradeoff between complicated compartmental models and too simplistic neural networks.

5. References

1. **Levine, D.S.** (2000). Introduction to Neural and Cognitive Modeling, 2nd Edition. Hillsdale, NJ: Erlbaum, pp. 355:377

MATLAB code

```
% CN 510 - Simulation Assigment 1
% -------------------------------
% -----The Leaky Integrator------
% -------------------------------
%      Massimiliano Versace
% Room 110 - versace@cns.bu.edu
% The Leaky integrator model of neural activity is so called because the neuron is assumed
% to integrate its input overtime while decayng, or leaking, at a rate proportional to its activity.
% The leaky integrator is represented by the following equation: dx/dt = -Ax + I

%%%%% Parameters initialization %%%%%
% A = decay rate
A = 1;

% Unit STM
x = 0;

% STM activation vector over time
Actx = zeros(1,10000);

% Integration step (numerical integration) delta-t
del_t = 0.01;

% STM change
dx_dt = 0;

% I = input vector, defined for 10,000 msec
I = zeros(1,10000);

% Actx1 and Actx2 are two variables that store x data in the two simulations
Actx1 = [];
Actx2 = [];

%%%%% Simulation 1: step input, A = 1%%%%%
I(1000:6000) = 5;
for i = 2:10000
    dx_dt = -A*Actx(i-1) + I(i);
    Actx(i) = Actx(i-1) + del_t*dx_dt;
end
Actx1 = Actx;
```

```
title1 = strcat('STM activation, A= ', num2str(A), ' k= ', num2str(del_t));


%%%%% Simulation 2: step input, A = 2%%%%%
Actx = zeros(1,10000);
A = 2;
I(1000:6000) = 5;
for i = 2:10000
    dx_dt = -A*Actx(i-1) + I(i);
    Actx(i) = Actx(i-1) + del_t*dx_dt;
end
Actx2 = Actx;
title2 = strcat('STM activation, A= ', num2str(A), ' k= ', num2str(del_t));


subplot(2,2,1),plot(I),title('Input'), axis([0 10000 -1 10])
subplot(2,2,2),plot(Actx1),title(title1), axis([0 10000 -1 10])
subplot(2,2,3),plot(Actx2, 'm'),title(title2), axis([0 10000 -1 10])
subplot(2,2,4),plot(Actx1),hold on, plot(Actx2,'m'), title('Blue: A=1, Magenta: A=2 '), axis([0 10000 -1 10])
```