

Equações de diferença: Uma aplicação à neurociência computacional

Difference equations: An application to computational neuroscience

André J. de Q. Padilha, andre.padilha@aluno.ufabc.edu.br
Rafael D. Vilela

Fundação Universidade Federal do ABC, Santo André, SP

Submetido em 01/09/2015

Revisado em 15/10/2015

Aprovado em 22/02/2016

Resumo: Muitos fenômenos em biologia podem ser enunciados em termos de mapas, *i.e.*, equações da forma $x_{n+1} = f(x_n)$, onde x_j e $f(x_j)$ são grandezas reais ou vetoriais e j é um inteiro positivo. Aqui, modelamos a atividade elétrica de neurônios usando mapas bidimensionais para investigar a possibilidade de ocorrência de atividade auto-sustentável numa rede de neurônio com topologia de mundo pequeno.

Palavras chave: Neurônios; Spikes; Mapas; Redes de mundo pequeno.

Abstract: Many biological phenomena can be stated in terms of maps, *i.e.*, equations of the form $x_{n+1} = f(x_n)$, where x_j and $f(x_j)$ are real-valued or vector quantities and j is a positive integer. Here, we model the electric activity of neurons using two-dimensional maps to investigate the possibility of occurrence of self-sustained activity in a small world neuron network.

Keywords: Neurons; Spikes; Maps; Small-world networks.

Introdução

Muitos fenômenos naturais envolvem grandezas que evoluem com o tempo. Entre inúmeros exemplos, podemos citar a temperatura em um ponto do espaço, o número de indivíduos de uma espécie biológica e a concentração de um reagente químico. Se supomos que a evolução temporal dessas grandezas é determinística (em oposição a aleatória), então podemos tentar construir um modelo baseado em *sistemas dinâmicos* (vide STROGATZ para uma abordagem detalhada sobre o assunto).

Um sistema dinâmico determina um conjunto de possíveis estados baseando-se em estados anteriores, por exemplo:

$$x_n = 2x_{n-1} \quad (1)$$

Onde n é um número natural. Lemos a eq. (1) da seguinte maneira: a n -ésima iterada da grandeza x é dada pelo dobro da sua $(n - 1)$ -ésima iterada.

Podemos classificar os sistemas dinâmicos em dois tipos:

- 1) O primeiro é correspondente aos *sistemas dinâmicos de tempo discreto*, precisamente porque n (variável temporal) pode assumir apenas valores inteiros. São também chamados de mapas ou ainda de equações de diferença. A equação (1) é um exemplo de mapa unidimensional.
- 2) O segundo corresponde aos *sistemas dinâmicos de tempo contínuo*, onde a variável temporal assume valores reais. Eles são geralmente governados e descritos por equações diferenciais.

Este trabalho estudou apenas sistemas dinâmicos do primeiro tipo, já que esses são acessíveis a um aluno ingressante na universidade, como era o caso do primeiro autor.

As diversas aplicações que sistemas dinâmicos possuem na ciência são temas contemporâneos de diversas pesquisas científicas. A compreensão do cérebro é uma tarefa de extrema importância para o avanço de áreas como a medicina e inteligência artificial, e o presente trabalho se situa nessa área (vide IZHEKEVICH para uma abordagem da atividade neuronal por sistemas dinâmicos).

Um problema atual e importante em neurociência teórica é o de como a atividade elétrica dos neurônios no cérebro subsiste mesmo na ausência de estímulos externos, e ao mesmo tempo não cresce explosivamente, o que poderia levar a um colapso desse órgão (vide ROXIN *et al.*). Em outras palavras, como o cérebro é capaz de manter atividade auto-sustentável (*i.e.*, que não se extingue) e limitada? Neste trabalho, propomos uma investigação desse problema modelando o cérebro como uma rede *de mundo pequeno* de neurônios cuja atividade elétrica é descrita por mapas unidimensionais.

Objetivos

O objetivo desse projeto foi estudar aspectos gerais de equações de diferença unidimensionais e aplicá-las à simulação (em linguagem C) de uma rede de mundo pequeno de neurônios. A partir das simulações, buscou-se avaliar a possibilidade de atividade auto-sustentável na rede.

Metodologia

Aspectos da teoria de mapas unidimensionais foram estudados no livro de ALLIGOOD *et al.*, e então se iniciou a modelagem da rede neuronal. O primeiro passo foi implementar as conexões de cada neurônio da rede.

Um neurônio é uma célula capaz de receber, processar e transmitir informações elétricas. A transmissão de informação se dá tipicamente por sinapses químicas, em que se conectam o axônio do neurônio transmissor (dito pré-sináptico) e os dendritos do neurônio receptor (dito pós-sináptico), da seguinte maneira: quando o potencial elétrico da célula pré-sináptica alcança certo limiar, seu valor se eleva bruscamente, fenômeno de curta duração (da ordem de 1 ms) conhecido como spike ou potencial de ação. O spike se propaga pela célula pré-sináptica e, quando chega ao axônio, induz a liberação de neurotransmissores. Esses neurotransmissores induzem a abertura de canais iônicos nos dendritos da célula pós-sináptica, o que acarreta uma corrente elétrica que altera o valor do potencial daquela célula.

As conexões entre cada neurônio foram representadas por meio de uma matriz de dimensão n^2 , onde n é o número de neurônios na rede. Essas conexões representam sinapses químicas entre neurônios e seguem um padrão lógico que explicamos a seguir.

Para a criação dessa matriz o seguinte algoritmo foi utilizado. A matriz de conexões w_{ij} era percorrida linha a linha, indo da primeira conexão da i -ésima linha até a última, dessa maneira percorrendo todas as i linhas e j colunas. Para todas as conexões que obrigatoriamente deveriam existir devido ao formato escolhido para a rede uma relação entre os parâmetros i e j foi encontrada; por exemplo, caso $i = j$ a entrada w_{ij} da matriz possuirá valor 0 já que um neurônio nunca está conectado a ele mesmo. Caso a saída do j -ésimo neurônio esteja ligada a entrada do i -ésimo neurônio então $w_{ij} = 1$.

Primeiramente implementamos as ligações bidirecionais com os primeiros vizinhos. Feito isso, procedemos à criação de clusters. Os clusters são regiões da rede neural com conexões e funcionamento similares. Essas regiões possuem uma alta densidade de conexões entre seus neurônios. Clusters possuem conexões de longo alcance com outros clusters da rede. Dessa maneira, regiões distantes dentro da rede podem trocar informação de maneira mais rápida.

Para a criação dos clusters, mais uma vez a matriz de conexões é percorrida linha a linha, passando-se por todos os elementos das linhas. Para isso os seguintes parâmetros foram utilizados:

- c_c , o parâmetro “cluster chance”, variando de 0 a 100. É a probabilidade de um neurônio pertencer a um cluster. Para cada neurônio da rede, o algoritmo sorteia um número aleatório distribuído uniformemente entre 0 e 100. Caso o valor obtido seja menor que c_c e exista espaço para o cluster, então o cluster será criado.
- c_a , o parâmetro “cluster activity”, variando também de 0 a 100. Com os clusters já criados, o algoritmo percorre todos os neurônios de cada cluster e para cada neurônio, sorteia um número aleatório distribuído uniformemente entre 0 e 100. Caso o valor obtido seja menor que c_a para certo neurônio de um cluster, então uma conexão será criada entre esse neurônio e um outro neurônio escolhido aleatoriamente entre aqueles pertencentes aos outros clusters da rede. É importante mencionar que a partir do momento em que o cluster (digamos, o cluster β) tenha um de seus neurônios com uma conexão direcionada para um neurônio de outro cluster (digamos, o cluster γ), todas as outras conexões de saída do

cluster β serão com o cluster γ . Porém, o cluster γ ainda poderá ter conexões de saída com um outro cluster.

Para o presente trabalho, a rede que foi mais exaustivamente testada foi a que possui o formato da superfície de um toro. É importante mencionar que um toro pode ser construído dobrando-se uma folha e unindo seus lados opostos. Depois disso uni-se os outros dois lados opostos. Sabendo disso, a “folha” inicial usada possuía 100x100 neurônios, totalizando uma rede de 10.000 neurônios.

Para a modelagem da dinâmica intrínseca de um neurônio, vamos introduzir inicialmente um modelo mais simples baseado em uma equação de diferença unidimensional, $x_n = f(x_{n-1})$, que descreve a evolução do potencial intracelular. A função f apresentada a seguir captura a essência da dinâmica de um neurônio isolado:

$$f(x) = \begin{cases} x^2, & \text{se } x < x_l \\ \theta, & \text{se } x \geq x_l \text{ e } x < \theta \\ x_r, & \text{se } x \geq \theta \end{cases} \quad (2)$$

Nessa equação, x_l corresponde ao valor limiar do potencial: se o potencial alcança valor maior ou igual ao limiar, então ele se eleva até o valor máximo θ (spike). O outro parâmetro, x_r (“x reset”), corresponde ao valor do potencial imediatamente após o spike. Note que $f(x)$ representa o valor do potencial do neurônio na próxima iterada da dinâmica.

Pelo gráfico 1 é possível notar a presença dos três intervalos da função. É importante ressaltar que os pontos onde $f(x) = x$ são os chamados pontos fixos da função f (nesse caso a função possui apenas dois pontos fixos). A derivada de f avaliada nesses pontos nos diz se esses pontos são sorvedouros (*i.e.*, um ponto fixo atrator) ou fontes (*i.e.*, um ponto fixo repulsor). Nesse caso, o primeiro ponto fixo, denotado por (σ_1) , é um sorvedouro e o segundo, denotado por (σ_2) , é uma fonte. Do Gráfico 1 vemos que, se o potencial de um neurônio ultrapassar o segundo ponto fixo, ele com certeza produzirá um spike em alguma das próximas iteradas.

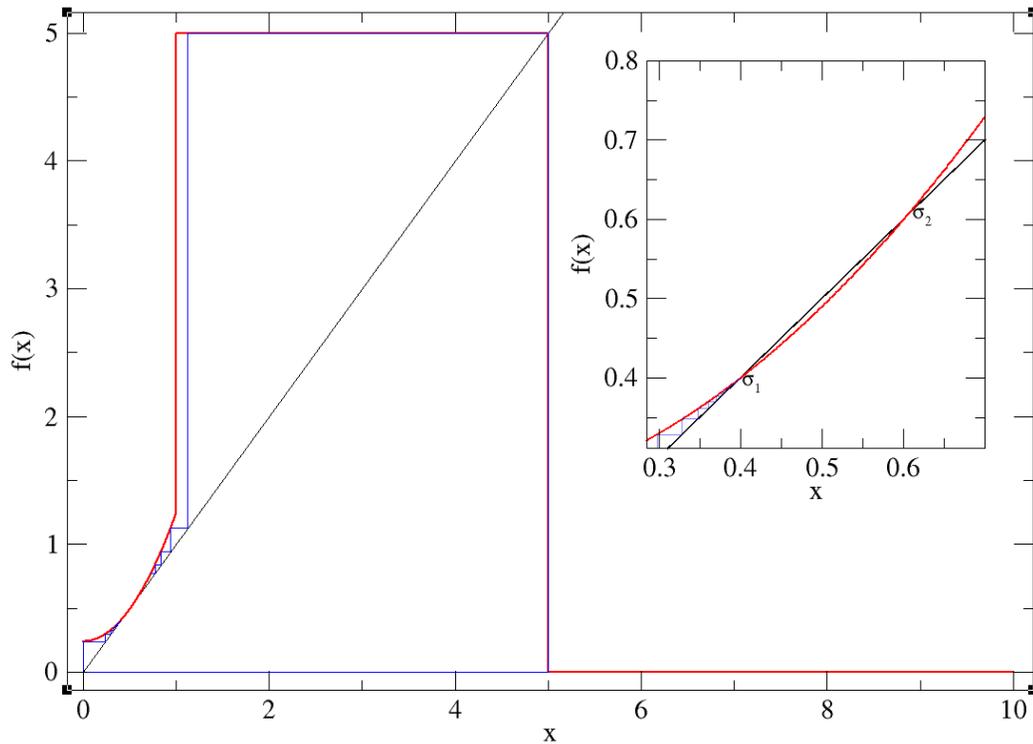


Gráfico 1: Em vermelho foi plotada a função e em azul conectamos os pontos iniciais de uma órbita com valor inicial acima do primeiro ponto fixo. Pontos consecutivos da órbita, representados na reta identidade $f(x)=x$, são conectados da seguinte forma: traçamos uma reta vertical até que seja alcançado do gráfico de $f(x)$ e, a seguir, traçamos uma reta horizontal até que se alcance novamente a reta identidade. Esse tipo de diagrama é chamado de cobweb plot. Os valores nos eixos são apenas ilustrativos e não correspondem aos reais valores utilizados nas simulações.

O modelo correspondente à eq. (2) pode ser incrementado para modelar a presença de correntes lentas, que estão presentes em muitas situações de interesse biofísico (vide THOMPSON *et al.*). Isso pode ser feito pela introdução de uma segunda variável dinâmica, denotada por y_n , o que implica um modelo bidimensional. Adotamos, assim, as seguintes equações para a modelagem de um neurônio individual:

$$\begin{cases} x_{n+1} = f(x_n) + \mu - y_n \\ y_{n+1} = \mu \delta_{\theta, x_n} + \frac{y_n}{\lambda} (1 - \delta_{\theta, x_n}) \end{cases} \quad (3)$$

Aqui temos alguns detalhes importantes: $\mu = 0.24$ para que $f(x) = x^2$ seja tal que torne σ_1 um sorvedouro. y_{n+1} , a função delay, é introduzida para que a parábola criada por f seja rebaixada uma iterada depois da ativação do neurônio e volte à sua posição original a uma taxa de $\lambda = 1.1$ por iterada. Isso é necessário para evitar a explosão de atividade na rede (já que aumenta a distância entre σ_1 e σ_2 fazendo com que mais estímulos sejam necessários para ativar o mesmo neurônio duas vezes seguidas). O δ corresponde ao delta de Kroenecker e caso x_n tenha produzido um spike, *i.e.*, $x_n = \theta$, então $\delta_{\theta, x_n} = 1$, caso contrário $\delta_{\theta, x_n} = 0$.

O comportamento que o parâmetro y_n leva o neurônio a ter é visto no Gráfico 2.

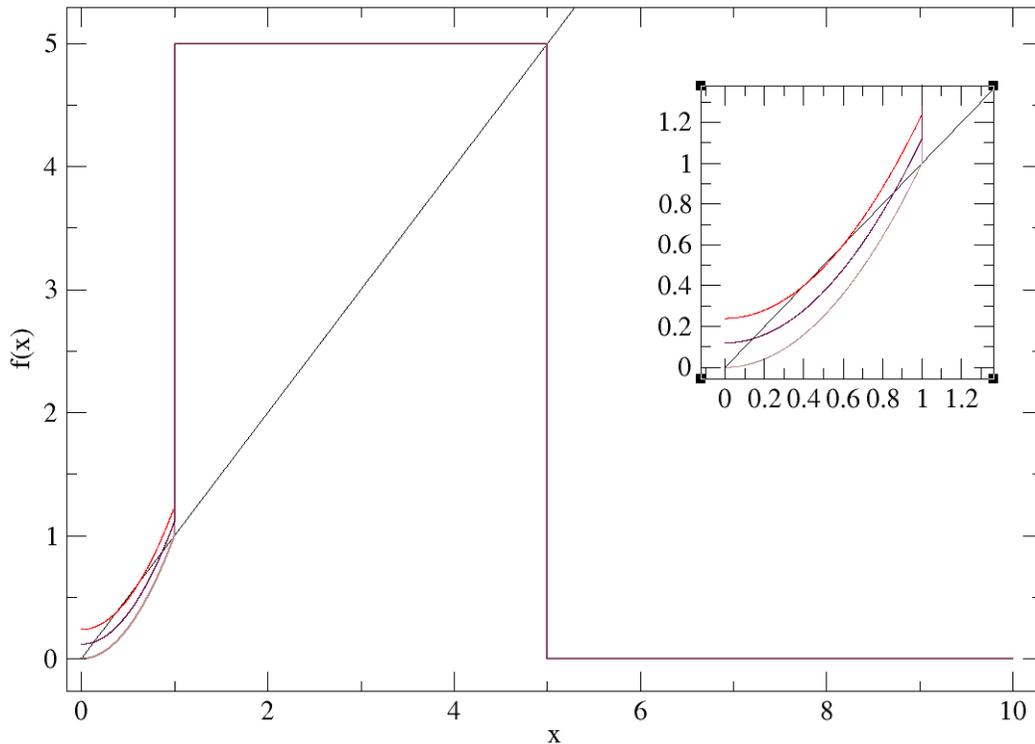


Gráfico 2: Note que a parte correspondente à parábola da função f possui três componentes nessa imagem. A primeira delas, em vermelho, é o estado inicial do neurônio, com uma certa distância entre seus pontos fixos. Em cinza, temos

o outro extremo, o caso onde o neurônio acabou de produzir uma excitação e teve sua parábola “rebaixada”, fazendo a distância entre os pontos fixos aumentar.

Finalmente, para o término da modelagem, introduzimos a interação dos neurônios dentro da rede, o que é feito na equação 4.

$$x_n^i = f(x_{n-1}^i) + \mu - y_n^i + g_{sin} \sum_{i \neq j} w_{ij} \delta_{\theta, x_{n-\tau}^j} \quad (4)$$

Nessa equação, x_n^i é o valor do potencial do i -ésimo neurônio na n -ésima iterada, w_{ij} é a matriz de conectividade da rede e possui valor $w_{ij} = 1$ quando o neurônio j pode enviar um sinal para o neurônio i ou $w_{ij} = 0$ quando ele não pode. Mais uma vez o delta de Kronecker aparece aqui, mas agora esse termo serve para indicar se o neurônio j enviou um potencial de ação ao neurônio i em alguma iterada $n - \tau$ anterior. O g_{sin} (“g sináptico”) é uma constante que altera o peso das excitações que chegam ao neurônio i provenientes da rede; dependendo do seu valor, um certo número de estímulos externos são necessários para excitar um neurônio da rede.

Inicialmente todos os neurônios começam com o potencial igual ao valor 0.4 e o sinal proveniente da rede e todos os delays (y) são iguais a zero.

Para simular a atividade dos neurônios, uma certa excitação inicial deveria ser aplicada na rede. Então, cada neurônio possui seu valor de potencial atualizado a cada iterada, baseado nos valores de potenciais de todos os neurônios da rede na iterada anterior. Os valores de potencial de cada neurônio, em cada iterada, são exportados para um arquivo de dados. Esses valores são então importados para o software *Xmgrace* para análise.

Para que um grande número de situações seja avaliada na busca por atividade auto-sustentável, variamos os valores de g_{sin} , da criação de clusters (c_c) entre 1% e 50%, do número de neurônios ativos dentro de um cluster (c_a), ou seja, a proporção de neurônios que enviam ligações de longo alcance (fixada

entre 50% e 80%), de σ_1 (que por sua vez depende do valor de μ) e do número e distribuição dos neurônios ativados na iterada zero.

O comportamento esperado é tal que um número não muito alto e nem muito baixo de neurônios sejam ativados por iterada, uma vez que os casos extremos não ocorrem na natureza. Um cérebro saudável não ativa quase todos os seus neurônios ou os mantém quase todos inativos.

Resultados

Diversas simulações foram realizadas, variando-se todos os parâmetros já mencionados. Em um primeiro momento, buscou-se encontrar a atividade auto-sustentável utilizando um modelo de neurônios sem a presença do termo de corrente lenta y_n^i e utilizando uma rede onde os neurônios se conectavam com seus oito vizinhos mais próximos. Porém, a atividade encontrada era sempre trivial, ou seja, se extinguiu ou era explosiva. Uma maneira de se manter atividade não explosiva na rede seria incluir neurônios inibitórios, isto é, cujo sinal tem o efeito de baixar o potencial nos neurônios pós-sinápticos. Como nosso objetivo era não utilizar neurônios inibitórios na rede, uma outra solução para o problema teve que ser encontrada.

Primeiro, a rede foi simplificada. Os neurônios agora só se conectavam com seus 4 vizinhos mais próximos (eliminando os vizinhos diagonais). Depois disso foi implantado o parâmetro que simula correntes lentas nos neurônios y_n^i . A rede então apresentou atividade auto-sustentável, como mostramos na figura a seguir, produzida utilizando-se a biblioteca SDL em C. Cada pequeno quadrado branco a seguir representa um neurônio na rede, que está ativado nessa exata iterada. As imagens são apresentadas numa ordem crescente de iteradas.

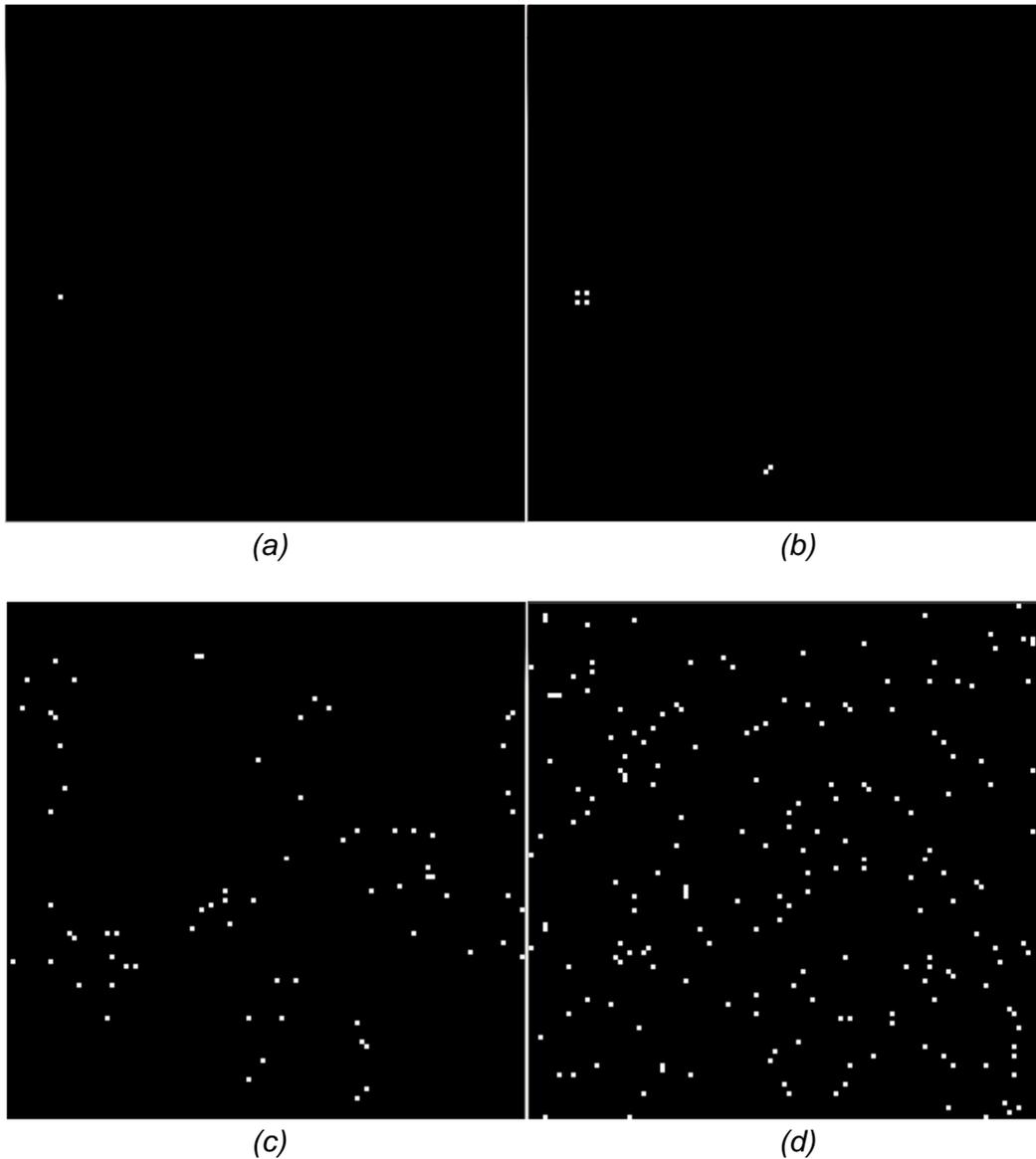


Figura 1: neurônios sendo ativados na rede na iteração (a) 1, (b) 5. (c) representa uma iteração próxima ao estado de equilíbrio e (d) já representa o próprio estado de equilíbrio.

Conclusões

É possível notar pelas imagens produzidas que a atividade realmente tornou-se auto-sustentável na rede, *i.e.*, a atividade manteve-se presente, com um número não muito alto nem muito baixo de neurônios sendo ativados por

iterada. Além disso, é importante observar que nenhum padrão de ativação periódico foi encontrado na rede.

Esse comportamento porém, como já mencionado, não foi encontrado numa rede sem a presença das correntes lentas sucedendo a ativação de um neurônio.

Sugestões de trabalhos futuros

Para trabalhos futuros é necessário criar um algoritmo que automatize as simulações. Em virtude do grande número de parâmetros, muitas possibilidades não foram simuladas. Desse modo, é possível que alguma escolha dos parâmetros, além da topologia da rede e da excitação inicial (utilizando um certo número de neurônios e/ou ativando neurônios em regiões diferentes da rede) possa criar atividade auto-sustentável na rede sem a inclusão da corrente lenta.

Referências

ROXIN, A.; RIECKE, H.; SOLLA, S. A. **Self-Sustained Activity in a Small-World Network of Excitable Neurons**. Evanstons: The American Physical Society, Phys. Rev. Lett., v. 92, n. 19, 2004.

IZHIKEVICH, E. M. **Dynamical Systems in Neuroscience: The geometry of Excitability and Bursting**. Cambridge: The MIT Press, 2007.

ALLIGOOD, K. T.; SAUER, T. D.; YORKE, J. A. **Chaos: An Introduction to Dynamical Systems**. Nova York: Springer, 1996.

STROGATZ, S. H. **Nonlinear Dynamics and Chaos**. Boulder: Westview Press, 2001.

THOMPSON, S.; SMITH, S. J.; JOHNSON, J. W. **Slow Outward Tail Currents in Molluscan Bursting Pacemaker Neurons: Two Components Differing in Temperature Sensitivity**. Pacific Grove: The Journal of Neuroscience, 1986.