Parallel Algorithm for bifurcation diagram of the Hodgkin–Huxley model

Magdalena Zlatanova

June 8, 2023

Abstract

Bifurcation refers to a qualitative change in the behavior of a dynamic system due to parameter variation, with implications for various physical, biological, and engineering systems. This paper presents a parallel algorithm for generating the bifurcation diagram of the Hodgkin–Huxley model, a mathematical model used to understand neural networks and action potentials in neurons. The algorithm's efficiency and scalability make it valuable for neuroscience and medicine, enabling comprehensive analysis, aiding in the development of targeted treatments, and enhancing our understanding of complex neural dynamics. The Hodgkin–Huxley model describes the movement of ions across cell membranes, involving nonlinear ordinary differential equations and gating variables. The parallel implementation allows for faster computation by distributing the simulation across multiple processors or threads. The solution architecture divides the applied current values and assigns subsets to worker processes for concurrent computation. After completing the calculations, the results are aggregated to analyze and visualize the bifurcation diagram.

1 Introduction

Bifurcation occurs when the system undergoes a sudden shift or branching in its equilibrium points, periodic orbits, or other stable states. In the context of nonlinear systems and dynamical systems theory, bifurcation theory studies how the qualitative behavior of a system changes as a parameter is continuously varied. It explores the critical values of the parameter at which qualitative changes occur and characterizes the associated changes in stability, periodicity, or attractor structure. Bifurcations are important because they can lead to the emergence of new dynamic patterns, complex behaviors, and transitions between different states or regimes. They are often associated with the occurrence of self-organization, pattern formation, chaos, or the onset of oscillations in various physical, biological, and engineering systems. Bifurcations play a significant role in various diseases and disorders, including Parkinson's disease, epilepsy, and pathological heart rhythms. Scientists and researchers have conducted extensive studies and experiments to understand and treat these conditions. Bifurcation control, specifically through electrical stimulation, is a promising approach in the field of neuromodulation. By applying targeted electrical stimulation to specific regions or circuits in the brain or heart, it is possible to influence the bifurcation behavior and restore normal physiological patterns. Numerous studies have investigated the use of bifurcation control techniques for therapeutic purposes. These studies explore how electrical stimulation can be used to induce desired bifurcation transitions, stabilize unstable states, or disrupt abnormal oscillatory patterns associated with diseases. The goal is to restore the system to a stable and healthy state. Bifurcation control techniques offer potential avenues for developing innovative treatments and interventions for diseases characterized by abnormal bifurcation dynamics. By understanding the underlying bifurcation mechanisms and developing targeted stimulation strategies, researchers aim to advance therapeutic approaches and improve the quality of life for individuals affected by these disorders. This paper provides a parallel algorithm for the generation of the bifurcation diagram for a mathematical model that describes the generation and propagation of action potentials, or electrical impulses, in neurons, so called the Hodgkin-Huxley model. This parallel implementation because of its efficiency and scalability might find applications in Neuroscience and Medicine, facilitating more comprehensive analysis, aiding in the development of targeted treatments and advancing our understanding of complex neural dynamics.

2 Related work

Several studies have focused on the parallelization and computational efficiency of bifurcation point detection and analysis. In the paper by Hasegawa et al. [1], the authors present the parallelization of Nested Layer Particle Swarm Optimization (NLPSO) for bifurcation point detection. By utilizing CUDA, the computational time for bifurcation point identification with NLPSO was significantly improved, achieving a speedup of up to 24 times compared to previous approaches. This work highlights the benefits of parallel computing and particle swarm optimization techniques for efficient bifurcation analysis. Cherfils [2] introduces a parallel and adaptive continuation method for the following of solution branches of parametrized semilinear elliptic problems. The proposed method combines mesh adaptation strategy with parallel programming on a computer network to reduce the computation time for accurate solutions. By distributing the computational load among multiple computers, the method achieves efficient continuation on refined meshes, thus enhancing the performance of bifurcation analysis. The paper by Pala and Machaczek [3] explores parallel computing techniques, specifically Nvidia CUDA and OpenMP, for the computation of two-parameter bifurcation diagrams of an electric arc model. The authors demonstrate that the complex bifurcations, including periodic and chaotic responses, exhibited by the electric arc model can be efficiently analyzed using parallel computing technologies. The utilization of both CPU and GPU processors allows for parallel and massively parallel computations, enabling faster and more accurate analysis of the bifurcation diagrams. Continillo et al. [4] propose a set of parallel tools for the bifurcation analysis of large-scale chemically reactive dynamical systems. The authors focus on systems where the right-hand side can be represented by a time numerically calculated evolution operator. By conducting the analysis in parallel and computing the eigenvalues of the Jacobian matrix, significant computational time is saved, enhancing the efficiency of the analysis. The presented examples, solution lines, and performance diagrams demonstrate the effectiveness of the proposed parallel tools for bifurcation analysis. The work by the authors of LOCA software library [5] introduces a set of bifurcation tracking algorithms specifically designed for large-scale applications. These algorithms, such as turning point , pitchfork, and Hopf bifurcation tracking, are implemented based on Newton's method and aim for scalability and ease of implementation with new application codes. The study focuses on the robustness and scalability of the algorithms, showcasing their successful application to a 1.6 million unknown model of 3D Rayleigh–Bénard convection. The results demonstrate the ability of the algorithms to handle large-scale problems while providing accurate solutions. These related works highlight various approaches to parallelization, adaptive continuation methods, and efficient computation techniques for bifurcation analysis. They demonstrate the effectiveness of parallel computing, particle swarm optimization, mesh adaptation, and specialized algorithms in improving the computational efficiency and accuracy of bifurcation point detection and analysis in different application domains.

3 Hodgkin-Huxley model

The Hodgkin-Huxley model is one of the best tools we have to understand the behavior of neural networks and the generation of action potentials (nerve impulses). It was developed by scientists Alan Hodgkin and Andrew Huxley in 1952 for which they earned the Nobel Prize in Physiology or Medicine in 1963. The mathematical model treats the components of the excitable cell as electrical elements (Figure 1). The model consists of nonlinear ordinary differential equations describing the movement of ions (charged particles) across the membrane of the cell, which is governed by the principles of electrochemical gradients and the properties of ion channels. Four channels are considered: sodium (Na+), potassium (K+), and leak channels, each of which can be open or closed. Since at each moment they could be open or closed, the model introduces gating variables Gating (m, h, n) representing the probability of the channels being open. The rate at which the channels transition between the open and closed states is determined by the voltage-dependent rate constants, alpha and beta. Equations [1], [2], and [3] govern these processes. The flow of ions through the channels influences the neuron's membrane potential (electrical charge across the membrane). The membrane potential (V) determines the neuron's excitability and whether an action potential is generated. In turn, the probability of a channel being open is based on the membrane potential.

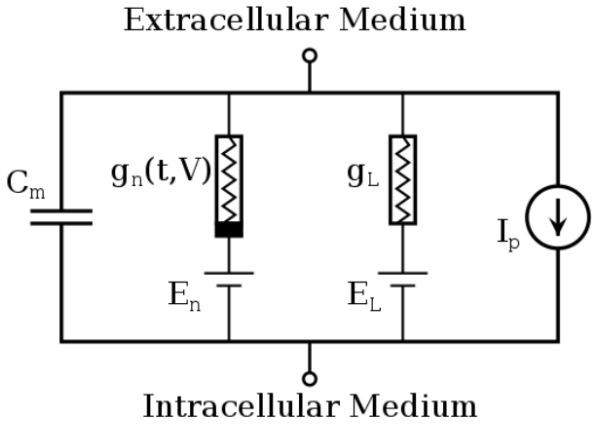

Figure 1: Hodgkin–Huxley model

1. Sodium channel activation variable (m):

$$\alpha_m(V) = \frac{2.5 - 0.1(V + 65)}{\exp(2.5 - 0.1(V + 65)) - 1}$$

$$\beta_m(V) = 4 \exp\left(\frac{-(V + 65)}{18}\right)$$

2. Sodium channel inactivation variable (h):

$$\alpha_h(V) = 0.07 \exp\left(\frac{-(V + 65)}{20}\right)$$

$$\beta_h(V) = \frac{1}{\exp(3.0 - 0.1(V + 65)) + 1}$$

3. Potassium channel activation variable (n):

$$\alpha_n(V) = \frac{0.1 - 0.01(V + 65)}{\exp(1 - 0.1(V + 65)) - 1}$$

$$\beta_n(V) = 0.125 \exp\left(\frac{-(V + 65)}{80}\right)$$

4 Challenges

The HH model is a set of nonlinear differential equations. The equations governing the dynamics of membrane voltage and ion channel conductances have nonlinear terms, making it challenging to find analytical solutions. Additionally, the variables in the HH model, such as membrane voltage and ion channel gating variables, are coupled to each other. Changes in one variable affect the behavior of others, leading to intricate interactions. This coupling makes it difficult to isolate individual variables and solve the equations independently. To make matters worse, multiple processes occur at different time scales. For example, the opening and closing of ion channels may happen on a faster time scale compared to the changes in membrane voltage. This presents a difficulty to numerical integration. A main concept explored in this paper is the sensitivity to changes in parameters. The HH model contains numerous parameters that govern the behavior of ion channels and neuronal dynamics. These parameters determine the characteristics of action potentials and neuronal excitability. However, small changes in parameter values can lead to significant changes in the dynamics, making the system highly sensitive.

5 Our implementation

The simulations presented in this paper used fixed-step numerical integration approach with a constant time step (dt). In the code, the update of the membrane potential (V) and the gating variables (m, h, n) was performed using explicit formulas derived from the HH model equations. The code iterated over a fixed number of time steps, and at each step, it updated the values of V, m, h, and n based on their respective update formulas. The update formulas in the code were based on the HH model's differential equations, where the time derivatives of V, m, h, and n are expressed in terms of voltage-dependent rate functions, conductances, and currents. The conductance of ion channels is modulated by three gating variables: m, h, and n. The update equations are as follows:

$$
\begin{aligned}
m[t+1] &= m[t] + \Delta t \left(\alpha_m(V[t])(1 - m[t]) - \beta_m(V[t])m[t]\right) \\
h[t+1] &= h[t] + \Delta t \left(\alpha_h(V[t])(1 - h[t]) - \beta_h(V[t])h[t]\right) \\
n[t+1] &= n[t] + \Delta t \left(\alpha_n(V[t])(1 - n[t]) - \beta_n(V[t])n[t]\right)
\end{aligned}
$$

6 The door for parallelization

The HH model involves simulating the dynamics of multiple neurons for different applied current values. By parallelizing the computation across multiple processors or threads, the simulation time can be significantly reduced. Each processor or thread can handle a subset of the neurons or current values independently, thereby exploiting parallel computing resources and achieving faster execution. As the size of the parameter space or the number of neurons increases, parallel computing allows for efficient utilization of computational resources. By adding more processors or threads, the computation can be further distributed, ensuring that the simulation scales with the increasing complexity of the problem.

7 Solution architecture

The range of applied current values is divided into smaller chunks and distributed among multiple worker processes. Each worker process independently computes the HH model for its assigned subset of current values. This parallel distribution allows for concurrent computation, reducing the overall computation time compared to performing the calculations sequentially. Each worker process independently evaluates the HH model equations for its assigned current values. This includes updating the membrane potential (V) and gating variables (m, h, n) over time using the differential equations of the HH model. By evaluating the model equations concurrently, the parallel implementation leverages the available computational resources to speed up the simulation. Each worker process maintains its own memory space for storing the computed membrane potentials (V) and gating variables (m, h, n). This isolation ensures that each process can perform its computations independently without interfering with the memory of other processes. The membrane potential represents the electrical potential difference across the neuronal membrane and changes over time in response to various stimuli, such as applied currents or synaptic inputs, thus each worker thread can be thought of as producing a signal. After enough iterations, this signal can either converge to a certain membrane potential value, oscillate between values or exhibit completely chaotic behaviour. Once all worker processes complete their computations, each process finds the peaks and troughs of the membrane potential signal and stores them in a list. If there is just one value in this list, then it is obvious that our system converged for the given value of the current. Otherwise, if it jumps between values we can conclude the membrane potential oscillates. Lastly, if there are very many peaks and troughs, the behaviour of the system for the given current is unpredictable. Since we know from the existing literature that the Hodgkin–Huxley model exhibits a Hopf Bifurcation Diagram (see Figure 2), we know that the system either converges or oscillates between two values. Therefore, it is sufficient to take only the minimum and maximum values of the signal. This aggregation step brings together the computed values from each process, allowing for further analysis or visualization of the bifurcation diagram.

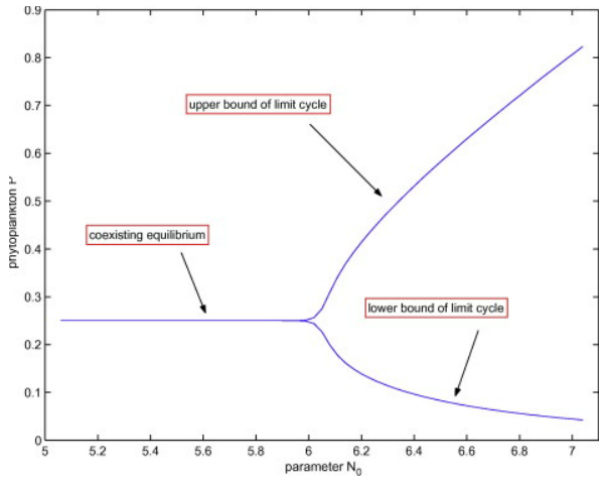

Figure 2: Hopf bifurcation

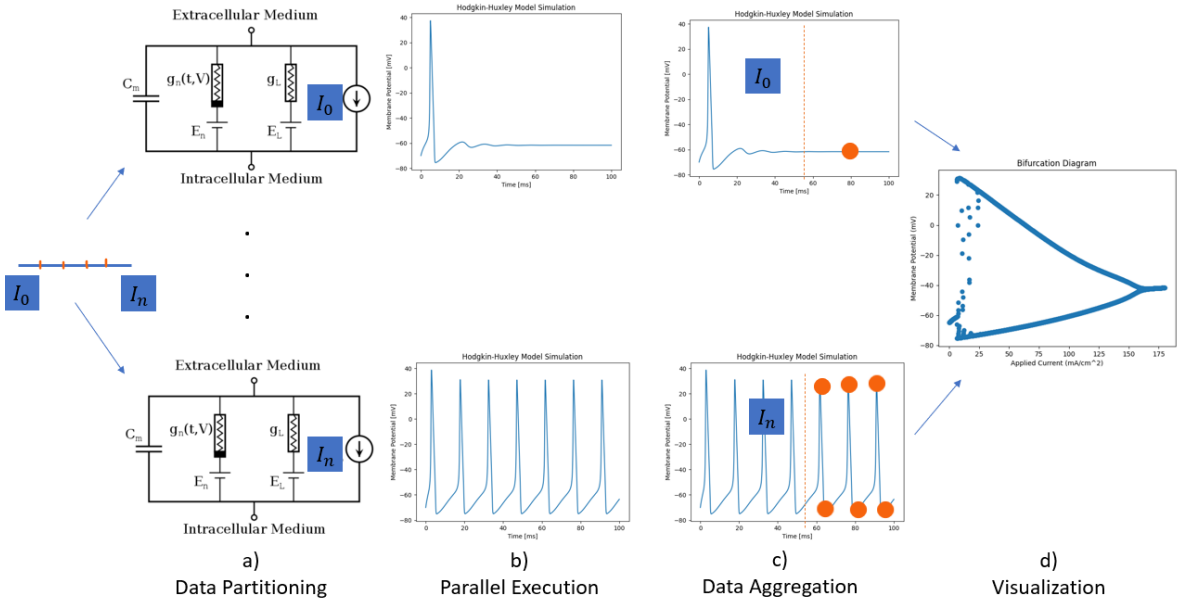

Figure 3: Solution architecture

index	I_values_start	I_values_end	Parallel Time	Sequential Time
0	0	80	30.234762	104.570596
0	0	180	30.508144	105.016344
0	0	220	33.480147	101.117105

Figure 4: Comparison of the sequential and parallel algorithms

8 Results

The experiment we conducted involves simulating the Hodgkin-Huxley model, which describes the electrical activity of neurons. The goal of the experiment is to analyze the impact of different ranges of applied current values (I) on the behavior of the model. Additionally, the experiment compares the execution times between parallel and sequential processing to evaluate the effectiveness of parallelization. We divided the range of applied current values into three intervals: (0 to 80), (0 to 180), and (0 to 220). For each interval, we generated a set of 1000 equally spaced current values within that range. To analyze the impact of applied current (I), we simulated the Hodgkin-Huxley model for each current value and recorded the maximum and minimum values of the membrane potential (V) during the last 1000 time steps. These values provide insights into the behavior of the model under different levels of applied current. By using the multiprocessing module, specifically the Pool class and its map method, we were able to divide the task into smaller subtasks and distribute the simulation tasks across multiple processes. Each process handled a subset of the current values, allowing for parallel execution of the simulations. This approach indeed leads to improved performance and reduced execution time compared to sequential processing. For the first range of applied current values (0 to 80), the parallel processing took approximately 30.23 seconds, while the sequential processing took around 104.57 seconds. This indicates a significant speedup with parallel processing compared to sequential processing. For the second range of applied current values (0 to 180), the parallel processing took approximately 30.51 seconds, and the sequential processing took around 105.02 seconds. Similar to the first range, parallel processing shows a significant speedup compared to sequential processing. For the third range of applied current values (0 to 220), the parallel processing took approximately 33.48 seconds, and the sequential processing took around 101.12 seconds. Again, parallel processing outperforms sequential processing in terms of execution time. Overall, the results indicate that parallel processing provides a substantial speedup compared to sequential processing for all three ranges of applied current values. The execution times for parallel processing are consistently lower than those for sequential processing. The experiment's findings emphasize the importance of parallelization techniques in optimizing the performance of computationally intensive tasks, such as simulations or data processing, where the workload can be divided into independent subtasks. Fortunately, such is the case for many bifurcation-based analyses. The experiments were performed on Intel(R) Core(TM) i5-11300H Processor with 4 cores. Figure 4 displays the results.

9 Conclusion

In conclusion, the presented parallel algorithm offers an efficient and scalable solution for generating the bifurcation diagram of the Hodgkin-Huxley model. By leveraging parallel computation, the algorithm significantly reduces computation time, enabling comprehensive analysis of the model's behavior. The results demonstrate that the algorithm accurately captures the bifurcation points and the qualitative behavior of the Hodgkin-Huxley model, allowing researchers to explore the impact of parameter variations on the system's dynamics. This advancement has significant implications for the fields of neuroscience and medicine. It enhances our understanding of complex neural dynamics and aids in the development of targeted treatments. By efficiently computing the bifurcation diagram, researchers can explore how neuronal networks respond to different stimuli and conditions, shedding light on the underlying mechanisms of neural disorders, optimizing stimulation protocols, and designing interventions tailored to individual patients. Looking ahead, there are several areas for future work to further enhance the algorithm and expand its capabilities. Firstly, the exploration of a wider range of parameter values would provide a more comprehensive understanding of the model's behavior. Additionally, incorporating experimental data into the algorithm can improve its predictive capabilities and validate the model's accuracy. Further research can also focus on exploring alternative paralleliza-

tion strategies to optimize the algorithm's performance. Techniques such as task-based parallelism or GPU acceleration could be investigated to achieve even greater computational efficiency. Moreover, extending the algorithm to incorporate multi-scale modeling would enable the investigation of emergent phenomena and network-level behaviors. This would provide a more holistic understanding of neural dynamics, bridging the gap between cellular-level dynamics and large-scale network dynamics. Additionally, the development of real-time parallel algorithms for the Hodgkin-Huxley model would have valuable applications in neuroprosthetics and brain-machine interfaces. Real-time simulations would enable interactive experiments, facilitating closed-loop control and feedback-based studies. By addressing these avenues for future work, researchers can further advance our understanding of neural dynamics and their implications in various domains. This would ultimately contribute to the development of improved treatments for neurological disorders and brain-related conditions. In summary, the presented parallel algorithm offers a powerful tool for studying the Hodgkin-Huxley model, and future research can build upon this foundation to explore new frontiers in neuroscience and medicine.

References

[1] Matsushita, H., H. Kurokawa, and T. Kousaka. "Saddle-node bifurcation parameter detection strategy with nested-layer particle swarm optimization." Chaos, Solitons Fractals 119 (2019): 126-134.

[2] Cherfils, L. (1998). A parallel and adaptive continuation method for semilinear bifurcation problems. Computer methods in applied mechanics and engineering, 163(1-4), 247-259.

[3] Pala, A., Machaczek, M. (Year). Parallel computing of 2-D bifurcation diagrams in circuits with electric arcs. Przeglad Elektrotechniczny, Volume 95, 2019.

[4] Gaetano Continillo, Artur Grabski, Erasmo Mancusi, Lucia Russo, Parallel tools for the bifurcation analysis of large-scale chemically reactive dynamical systems, Computers Chemical Engineering, Volume 38, 2012. b5Salinger, Andrew G., et al. "Bifurcation tracking algorithms and software for large scale applications. International Journal of Bifurcation and Chaos 15.03 (2005): 1015-1032.

[5] K. Cliffe, A. Spence and S. Tavener, The Numerical Analysis of Bifurcation with Application to Fluid Mechanics, Acta Numerica (Cambridge University Press, 2000) pp. 39–131.

[6] W. Govaerts , Numerical Methods for Bifurcations of Dynamic Equilibria (SIAM , Philadelphia, PA , 2000).

[7] HODGKIN AL, HUXLEY AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol. 1952 Aug;117(4):500-44. doi: 10.1113/jphysiol.1952.sp004764. PMID: 12991237; PMCID: PMC1392413.

[8] Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting, Eugene M. Izhikevich, The MIT Press, 2006

[9] E. M. Izhikevich, "Simple model of spiking neurons," in IEEE Transactions on Neural Networks, vol. 14, no. 6, pp. 1569-1572, Nov. 2003, doi: 10.1109/TNN.2003.820440.

[10] Guckenheimer, J., Holmes, P. (1986). Nonlinear oscillations, dynamical systems, and bifurcations of vector fields. Springer.