Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Efficient digital design of ganglion cells in the retinal pathway

Soheila Nazari

Faculty of Electrical Engineering, Shahid Beheshti University, Tehran, Iran

ARTICLE INFO

INDEX TERMS:
Retinal ganglion cell (RGC)
Linear retinal ganglion cell model (LRGC)
Digital design
Retinal pathway
Image reconstruction

ABSTRACT

Spiking networks, the third generation of neural networks, are presented as low-power consumption machines with higher cognitive ability, one of the main concerns in intelligence machines. In fact, neuromorphic systems are hardware implementations of spiking networks with minimum resource, area, and power consumption while preserve maximum working frequency. Here, the focus is on the digital implementation of Retinal Ganglion Cell (RGC) based on the linear approximation of non-linear terms which is called Linear Retinal Ganglion Cell (LRGC). The low-cost hardware design of biological cells is acceptable when the digital model of the cell has the same phase and time domain behavior as the original model and follows the dynamic behavior of the original model accurately, which is discussed and confirmed with different analyzes in this paper. The low-cost hardware design of biological cells allows the optimal implementation of a neural population on the hardware, provided that the collective behavior of the digital model matches the original model which is approved by the large-scale simulation of RGC and LRGC models. Cognitive processes are performed in the nervous system at a very low cost, which neuromorphic systems are trying to achieve this important. In this regard, the behavior of RGC and LRGC models in the reconstruction of the image through the retina pathway was examined and a high agreement between the performance of the two models was achieved. Finally, the high functional compatibility of RGC, LRGC models proves that the proposed model is a good candidate of the main model in neuromorphic systems with low hardware cost.

1. Introduction

In the complex world of interaction of biological cells in the nervous system, neurons form a larger and more important part [1]. Neurotransmitters are transmitted through excitatory and inhibitory synapses that form the connection between neurons and create neural interactions in learning and cognitive processes [2]. The amazing thing is that even during complicated cognitive operations, the nervous system uses only power of 10–20 W, while having a vast number of neurons and intricate connections between them [3]. Spiking networks strive to do this by inspiring neural system computations, which transform the nervous system's distinct low-power consumption characteristics to modern intelligence robots [4]. This significance has led to the consideration of mimicking the functional and computational process of the nervous system in the hardware design of biological cells, including neurons [5], astrocytes [6,7], and synapses [5]. Efficient hardware design of biological cells offers the option of delivering a brain-inspired neural population for machine learning applications and cognitive processes, despite the limited hardware resources and the size of the created neuromorphic devices [8,9].

Owing to the significant importance of developing frameworks with spike-like computations, many efforts have focused on

E-mail address: soheilanazari@aut.ac.ir.

https://doi.org/10.1016/j.heliyon.2024.e36673
Received 7 February 2024; Received in revised form 15 August 2024; Accepted 20 August 2024
Available online 27 August 2024

computational models of brain-like structures, particularly neurons, in the past few years. In the field of computational neuroscience, various neuron models have been presented, including the leaky integrate and fire neuron (LIF) [10], Izhikevich [11], Hindmarsh-Rose (HR) [12], Morris–Lecar [13], …, Hodgkin–Huxley (HH) neuron model [14]; additionally, retinal neurons have been studied, including bipolar cells [15], Ganglion cells [16], and amacrine cells [17]. Some of these models, like LIF, have low biological realization and computational cost, while others, like HH and Retinal Ganglion Cell (RGC), have high biological realization and computational cost. But because the HH and RGC models are grounded in biological characteristics and empirical findings, they are valuable tools for developing highly parallel neuromorphic processors for bio-inspired computer systems.

Three approaches to hardware neural model creation are possible: mixed mode analog/digital, analog design, and digital design. Even if analog design is more skilled, it is given less attention than digital design because it requires more time to plan, is more susceptible to noise, and is less flexible [18]. Within the applications of the neuromorphic framework plan, FPGA (Field-Programmable Gate Array) has been able to garner greater attention among the hardware boards for digital implementation [19]. Because of their availability, flexibility, and capacity to provide a scalable pool of digital gates for the building of large-scale spiking networks, FPGAs are an excellent fit for the application examined in this research. Many works have been presented in the topic of digital neuromorphic implementation on FPGA. The FPGA design of neural models has employed several approaches, some of which are worth mentioning: nonlinear estimates based on LUT [20], piecewise linear approximation [21], Single Constant Multiply (SCM) [5], and CORDIC based approximations [22]. Upon examining prior designs of neuromorphic digital neurons, it becomes evident how important it is to digitally realize biological cells like retinal neurons as a component of neuromorphic frameworks that enable the use of spiking systems for cognitive applications on an electronic chip. The ganglion cell and the HH neuron share computational relationships; however, the ganglion cell has five ion channels whereas the HH neuron only has two. This indicates the ganglion cell's greater biological realism and computational complexity. Although the ganglion cell has not yet been implemented in hardware, a small number of studies have concentrated on the HH neuron's hardware implementation. As a result, it may be useful to study the earlier digital implementation techniques for HH neurons. First, a quite simple neural network implementation has been suggested [23]. This approach's clear drawback is that it requires a lot of hardware resources, which makes large-scale implementation impracticable. Other methods have suggested performing exponential terms using the CORDIC calculation and look-up tables (LUTs) [24]. However, even with this improvement, there was still a high resource use rate. Furthermore, other approaches concentrated on streamlining the HH neuron's dynamic equations, which were linked to the issue of restricting the input range and inaccuracy [25–29]. Consequently, it appears that the hardware implementation of the HH neuron, which preserves the model's accuracy and dynamic reactivity while streamlining the equations, may be a workable approach. In this paper, we focus on the hardware implementation of the ganglion cell by simplifying the dynamic equations, since the hardware technique has not yet been applied to dynamic ganglion cells. This allows the designed digital cell's complex and dynamic behaviors to closely align with the original model.

A linear approximation of the original RGC model is called the Linear Retinal Ganglion Cell (LRGC). In comparison to the RGC model, the LRGC requires less area, less resource utilization, and higher speed, all of which translate into lower power consumption. The LRGC model's effective digital implementation, which uses little resources, enables the digital realization of a large-scale population of bio-inspired neurons.

The suggested LRGC model should be able to faithfully replicate the dynamic behavior of the original RGC model while requiring less hardware than the original model. The spiking response in the time domain, the behavior of the phase space, and the movement of the trajectories in the nullcline space are compared, and the exact imitation of the LRGC from RGC model was endorsed in order to ensure that the functional and behavioral performance of the proposed LRGC model matches to the RGC model.

From the perspective of single cell behavior, the suggested LRGC model acts exactly like the original model. It also has the advantage of being able to be created for a significantly lower hardware cost than the RGC model. Therefore, by modeling a population of ganglion cells of the original and approximate model, it was shown that the collective behavior of the two models is completely consistent. Considering that the application of ganglion cells is in the visual system, the performance of the proposed model should be confirmed in this structure. Consequently, the LRGC model has been applied to the biological retinal pathway design in picture reconstruction; the outcomes of this phase also validate the precise functionality of the suggested LRGC model. Lastly, the innovations and achievements are categorized as follows:

The innovations.

1. For the first time, the ganglion cells which are the main way of connection between the retina and the nervous system, have been implemented as hardware.
2. The proposed approach in digital implementation (linear approximation based on search algorithm and low-cost choice of registers size) is in a way that provides high working frequency and low hardware error while consuming low power and hardware resources.

The achievements.

1. The collective behavior of the proposed cells in a large-scale neural population is similar to the original model.
2. The error between the image reconstruction from the spike trains generated from the ganglion cells and the reconstruction using the proposed Linear Ganglion cells is zero.

The rest of paper is organized as follows:

Sections II and III examine the RGC and LRGC computational models. Section IV discusses large-scale retinal ganglion cell modeling. Sections V and VI contain the hardware implementation and discussion, and section VII wraps up the work.

2. Retinal ganglion cell model

In 1990, Fohlmeister et al. presented one of the most important models of retinal neurons called Retinal Ganglion Cell [16]. Based on the retinal pathway structure, retinal ganglion cells are the main way of connection between the retina and the nervous system [30]. The Ganglion cells modulate information in the form of frequency spikes. According to the laboratory findings on rat [31] and salamander [32], the number of ion channels in ganglion cells is at least 5, which shows the higher computational complexity of these cells compared to high-cost Hodgkin-Huxley neurons. Fohlmeister's proposed model is a biological computational model of a ganglion cell with 5 nonlinear ion currents, which has spike responses very similar to experimental records [16]. five ion currents considered in this model include: sodium current (Na^{+}), calcium current (Ca^{2+}), Non inactivating potassium current (K^{+}), activating potassium current (A type), and Ca^{2+} activated K^{+}. The first four ion currents are voltage-gated while the last ion current ($g_{K,Ca}$) is calcium gated. The differential equation of the ganglion cell membrane potential was obtained according to Eq. (1):

$$C_m\frac{dE}{dt} = -\widetilde{g_{Na}}m^3h(E-E_{Na}) - \widetilde{g_{Ca}}C^3(E-E_{Ca}) - \widetilde{g_K}n^4(E-E_K) - \widetilde{g_A}a^3h_A(E-E_K) - g_{K,Ca}(E-E_K) + I_{Stim} \tag{1}$$

The equations of five ion currents, which include four voltage-gated ion currents and one current-gated ion current, are presented as follows through Eq. (2) to Eq. (9):

$$\frac{d\left[Ca^{2+}\right]}{dt} = -0.000015\,I_{Ca} - (0.02)\left(\left[Ca^{2+}\right] - 0.0001\right) \tag{2}$$

$$g_{K,Ca} = \widetilde{g_{K,Ca}}\frac{\left(\frac{\left[Ca^{2+}\right]}{Ca^{2+}{}_{diss}}\right)^2}{1+\left(\frac{\left[Ca^{2+}\right]}{Ca^{2+}{}_{diss}}\right)^2} \tag{3}$$

$$\frac{dm}{dt} = \alpha_m - (\alpha_m + \beta_m)m \tag{4}$$

$$\frac{dh}{dt} = \alpha_h - (\alpha_h + \beta_h)h \tag{5}$$

$$\frac{dC}{dt} = \alpha_C - (\alpha_C + \beta_C)C \tag{6}$$

$$\frac{dn}{dt} = \alpha_n - (\alpha_n + \beta_n)n \tag{7}$$

$$\frac{da}{dt} = \alpha_a - (\alpha_a + \beta_a)a \tag{8}$$

$$\frac{dh_A}{dt} = \alpha_{h_A} - \left(\alpha_{h_A} + \beta_{h_A}\right)h_A \tag{9}$$

Also, the auxiliary equations used are according to Eq. (10) to Eq. (21):

$$\alpha_m = \frac{-0.6(E+30)}{e^{-0.1(E+30)} - 1} \tag{10}$$

$$\beta_m = 20e^{\frac{-(E+55)}{18}} \tag{11}$$

$$\alpha_h = 0.4e^{\frac{-(E+50)}{20}} \tag{12}$$

$$\beta_h = \frac{6}{e^{-0.1(E+20)} + 1} \tag{13}$$

$$\alpha_C = \frac{-0.3(E+13)}{e^{-0.1(E+13)} - 1} \tag{14}$$

$$\beta_C = 10e^{\frac{-(E+38)}{18}} \tag{15}$$

$$\alpha_n = \frac{-0.02(E+40)}{e^{-0.1(E+40)} - 1} \tag{16}$$

$$\beta_n = 0.4e^{\frac{-(E+50)}{80}} \tag{17}$$

$$\alpha_a = \frac{-0.006(E+90)}{e^{-0.1(E+90)} - 1} \tag{18}$$

$$\beta_a = 0.1e^{\frac{-(E+30)}{10}} \tag{19}$$

$$\alpha_{h_A} = 0.04e^{\frac{-(E+70)}{20}} \tag{20}$$

$$\beta_{h_A} = \frac{0.6}{e^{-0.1(E+40)} + 1} \tag{21}$$

The values of the constant parameters are summarized in Table 1.

The RGC model contains a variety of nonlinear terms, including fractional terms, exponential functions, and expressions with powers of two, three, and four (see Table 1). This is seen from the equations. Large-scale neural implementation is made more difficult by non-linear terms and the usage of multiplier, divider, and exponential functions, which also raise hardware implementation costs [5]. By substituting effectively linear approximation blocks for the nonlinear elements, the Linear Retinal Ganglion Cell (LRGC) model has been presented as a solution to this problem. As a result, the approximation model can be used to create inexpensive digital circuits, while the LRGC model fully protects the dynamic properties of the RGC model.

3. Linear Retinal Ganglion Cell model

In order to achieve a dense digital circuit with the lowest possible resource consumption, maximum operating frequency, and area, the nonlinear terms in the RGC model need to be made simpler. The linear approximation approach has been used in this paper to simplify the nonlinear elements of the RGC. A search algorithm is described to determine the optimum linear fitting for the RGC model's non-linear terms. In the process of finding the best linear approximation, first two random points are selected on the nonlinear curve and these two random points are shifted in half of the horizontal axis changes with a step length of $\frac{1}{20} * \textit{Horizontal axis changes} = \frac{1}{\textit{number of point}*10} * \textit{Horizontal axis changes}$ and the error criteria NMAE (Eq. (41)) was calculated between approximated linear functions and nonlinear curve. If the minimum calculated errors are less than the threshold limit, the search ends, and otherwise, three random points are selected, each of which is in one third of the horizontal axis. These three points are shifted in third of the horizontal axis changes with a step length of $\frac{1}{30} * \textit{Horizontal axis changes} = \frac{1}{\textit{number of point}*10} * \textit{Horizontal axis changes}$ and the error criteria NMAE was calculated between approximated linear functions and nonlinear curve. If the minimum calculated errors are less than the threshold limit, the search ends, and otherwise, 4, 5, and up to 11 random points are selected on the nonlinear curve and the previous steps are repeated. Determining the value of the threshold error depends on the effect value of the desired nonlinear function in the production of the cell membrane potential. So that for functions $\widetilde{g_{Na}}m^3$, $\widetilde{g_{Ca}}C^3$, $\widetilde{g_A}a^3$ and $\widetilde{g_K}n^4$, the minimum error is 10^{-6}, and for functions α_m, β_m, …, α_{h_A}, β_{h_A} the minimum error is 0.3.

The comparison results that will be discussed further show that the process of finding the best linear fitting has been done properly, so that the approximate model LRGC follows the behavior of the original RGC model in the various designed tests. Finally, the LRGC model is summarized with Eq. (22) to Eq. (40):

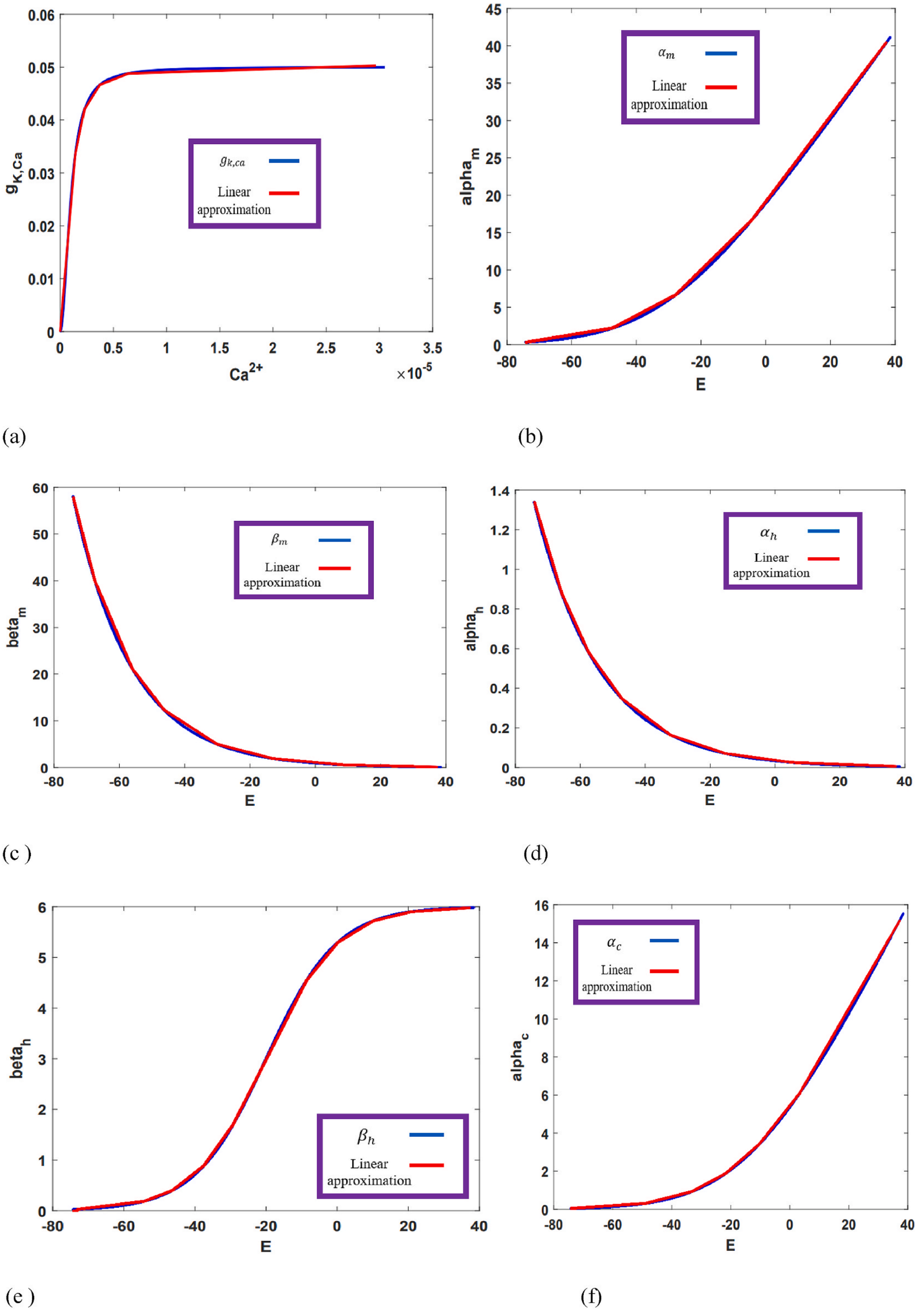
$$\frac{d\left[Ca^{2+}\right]}{dt} = -0.000015\, I_{Ca} - (0.02)\left(\left[Ca^{2+}\right] - 0.0001\right) \tag{22}$$

Table 1
The constant parameters values of the RGC model.

Parameter	Value	Parameter	Value
C_m	10ˆ-3	$\widetilde{g_{Na}}$	40
$\widetilde{g_{Ca}}$	2	$\widetilde{g_K}$	12
$\widetilde{g_A}$	36	E_{Na}	35
E_K	−75	E_{Ca}	30
I_{Ca}	10ˆ-2	Ca^{2+}_{diss}	10ˆ-6

Table 2
Constant value of equations (23)–(39).

Interval of changes Ca^{2+}	X_1	Y_1	Interval of changes E	X_2	Y_2	Interval of changes E	X_8	Y_8
$[0, 1.421 * 10^{-6}]$	23539.76	0	[− ∞, − 47.29]	0.0717	5.6305	[− ∞, − 52.15]	0.0036	0.2918
$[1.421, 2.303] * 10^{-6}$	9773.2426	0.019563	[− 47.29, − 27.84]	0.2279	13.0162	[− 52.15 , − 32.38]	0.0093	0.5858
$[2.303, 3.744] * 10^{-6}$	3192.227	0.04362	[− 27.84, − 4.308]	0.4258	18.5242	[− 32.38, − 9.434]	0.0154	0.7831
$[3.744, 6.417] * 10^{-6}$	808.4624	0.04362	[− 4.308, ∞]	0.5713	19.1510	[− 9.434, ∞]	0.0194	0.8216
$[6.417 * 10^{-6}, \infty]$	62.596	0.0484						
Interval of changes E	X_3	Y_3	**Interval of changes E**	X_4	Y_4	**Interval of changes E**	X_6	Y_6
[, − 67.46]	− 2.6882	− 141.3830	[, − 65.77]	− 0.0546	− 2.7078	[, − 48.96]	0.0105	0.8204
[− 67.46 , − 56]	− 1.6422	− 70.8251	[− 65.77 , − 57.39]	− 0.0360	− 1.4861	[− 48.96 , − 33.12]	0.0396	2.2438
[− 56, − 46.33]	− 0.9080	− 29.7059	[− 57.39, − 47.24]	− 0.0227	− 0.7229	[− 33.12, − 21.67]	0.0832	3.6890
[− 46.33, − 30.17]	− 0.4533	− 8.6404	[− 47.24, − 32.38]	− 0.0123	− 0.2326	[− 21.67, − 10.19]	0.1356	4.8240
[− 30.17, − 13.09]	− 0.1807	− 0.4161	[− 32.38, − 15.04]	− 0.0055	− 0.0136	[− 10.19, 3.432]	0.1959	5.4386
[− 13.09, 8.68]	− 0.0628	1.1267	[− 15.04, 5.543]	− 0.0022	+ 0.0369	[3.432,]	0.2675	5.1931
[8.68,]	− 0.0159	0.7196	[5.543,]	− 0.0006	0.0284			
Interval of changes E	X_5	Y_5	**Interval of changes E**	X_7	Y_7	**Interval of changes E**	X_9	Y_9
[, − 54.22]	0.0085	0.6492	[, − 67.6]	− 3.4725	− 182.9694	[, − 61.05]	− 0.0063	0.0755
[− 54.22 , − 46.24]	0.0271	1.6565	[− 67.6 , − 61.67]	− 2.4165	− 111.5872	[− 61.05 , − 47.05]	− 0.0053	0.1370
[− 46.24, − 37.57]	0.0550	2.9506	[− 61.67, − 55.01]	− 1.7583	− 70.9918	[− 47.05, − 33.05]	− 0.0044	0.1768
[− 37.57, − 29.33]	0.0983	4.5772	[− 55.01, − 47.2]	− 1.1601	− 38.0844	[− 33.05, − 19.05]	− 0.0037	0.2003
[− 29.33, − 8.594]	0.1376	5.7298	[− 47.2, − 37.77]	− 0.7211	− 17.3661	[− 19.05, − 5.05]	− 0.0031	0.2117
[− 8.594, 0.2297]	0.0852	5.2794	[− 37.77, − 24.83]	− 0.3910	− 4.8965	[− 5.05, 8.95]	− 0.0026	0.2143
[0.2297, 10.15]	0.0424	5.2893	[− 24.83, − 9.875]	− 0.1815	+ 0.3032	[8.95, 22.95]	− 0.0022	0.2106
[10.15, 20.86]	0.0169	5.5485	[− 9.875,]	− 0.0415	+ 1.6863	[22.95,]	− 0.0018	0.2026
[20.86,]	0.0048	5.8						
Interval of changes E	X_{11}	Y_{11}	**Interval of changes E**	X_{12}	Y_{12}	**Interval of changes E**	X_{13}	Y_{13}
[, − 69.92]	− 0.6720	− 41.5699	[, − 65.62]	− 0.0020	− 0.0992	[, − 63.43]	0.0032	0.2543
[− 69.92 , − 65.54]	− 0.4381	− 25.2189	[− 65.62 , − 57.06]	− 0.0013	− 0.0533	[− 63.43 , − 56.08]	0.0065	0.4626
[− 65.54, − 60.22]	− 0.2714	− 14.2934	[− 57.06, − 48.41]	− 0.0009	− 0.0304	[− 56.08, − 49.47]	0.0102	0.6736
[− 60.22, − 54.12]	− 0.1536	− 7.1982	[− 48.41, − 39.03]	− 0.0005	− 0.0110	[− 49.47, − 29.17]	0.0138	0.8515
[− 54.12, − 46.49]	− 0.0780	− 3.1054	[− 39.03, − 25.12]	− 0.0003	− 0.0032	[− 29.17, − 22.22]	0.0093	0.7207
[− 46.49, − 36.09]	− 0.0323	− 0.9829	[− 25.12, − 11.07]	− 0.1522e − 03	0.4157e − 03	[− 22.22, − 12.85]	0.0053	0.6308
[− 36.09, − 19.62]	− 0.0090	− 0.1413	[− 11.07, 10.38]	− 0.00007	0.0014	[− 12.85, − 1.063]	0.0021	0.5903
[− 19.62,]	− 0.0006	0.0234	[10.38,]	− 0.0202e − 03	0.9282e − 03	[− 1.063, 18.13]	0.0005	0.5886
						[18.13,]	0.0001	0.5968
Interval of changes m	X_{14}	Y_{14}	**Interval of changes C**	X_{15}	Y_{15}	**Interval of changes n**	X_{17}	Y_{17}
[, 0.209]	0.087	**0**	[, 0.209]	1.7474	0	[, 0.205]	0.1034	**0**
[0.209 , 0.329]	0.441	− 0.0739	[0.209 , 0.329]	8.8233	− 1.4789	[0.205 , 0.301]	0.8053	− 0.1439
[0.329, 0.504]	1.056	− 0.2763	[0.329, 0.504]	21.1257	− 5.5264	[0.301, 0.385]	1.9655	− 0.4931
[0.504, 0.604]	1.8465	− 0.6745	[0.504, 0.604]	36.9300	− 13.4917	[0.385, 0.45]	3.5154	− 1.0898
[0.604, 0.748]	2.752	− 1.22	[0.604, 0.748]	55.0417	− 24.4312	[0.45, 0.506]	5.2607	− 1.8752
[0.748, 0.869]	3.9297	− 2.1	[0.748, 0.869]	78.5950	− 42.0491	[0.506,]	8.1022	− 3.3130
[0.869,]	5.248	− 3.248	[0.869,]	104.9618	− 64.9618			
Interval of changes E	X_{10}	Y_{10}	**Interval of changes a**	X_{16}	Y_{16}			
[, − 53.25]	0.0051	0.4957	[, 0.209]	1.5727	0			
[− 53.25 , − 31.45]	0.0058	0.5340	[0.209 , 0.329]	7.941	− 1.3310			
[− 31.45,]	0.0060	0.5397	[0.329,]	19.0131	− 4.9738			

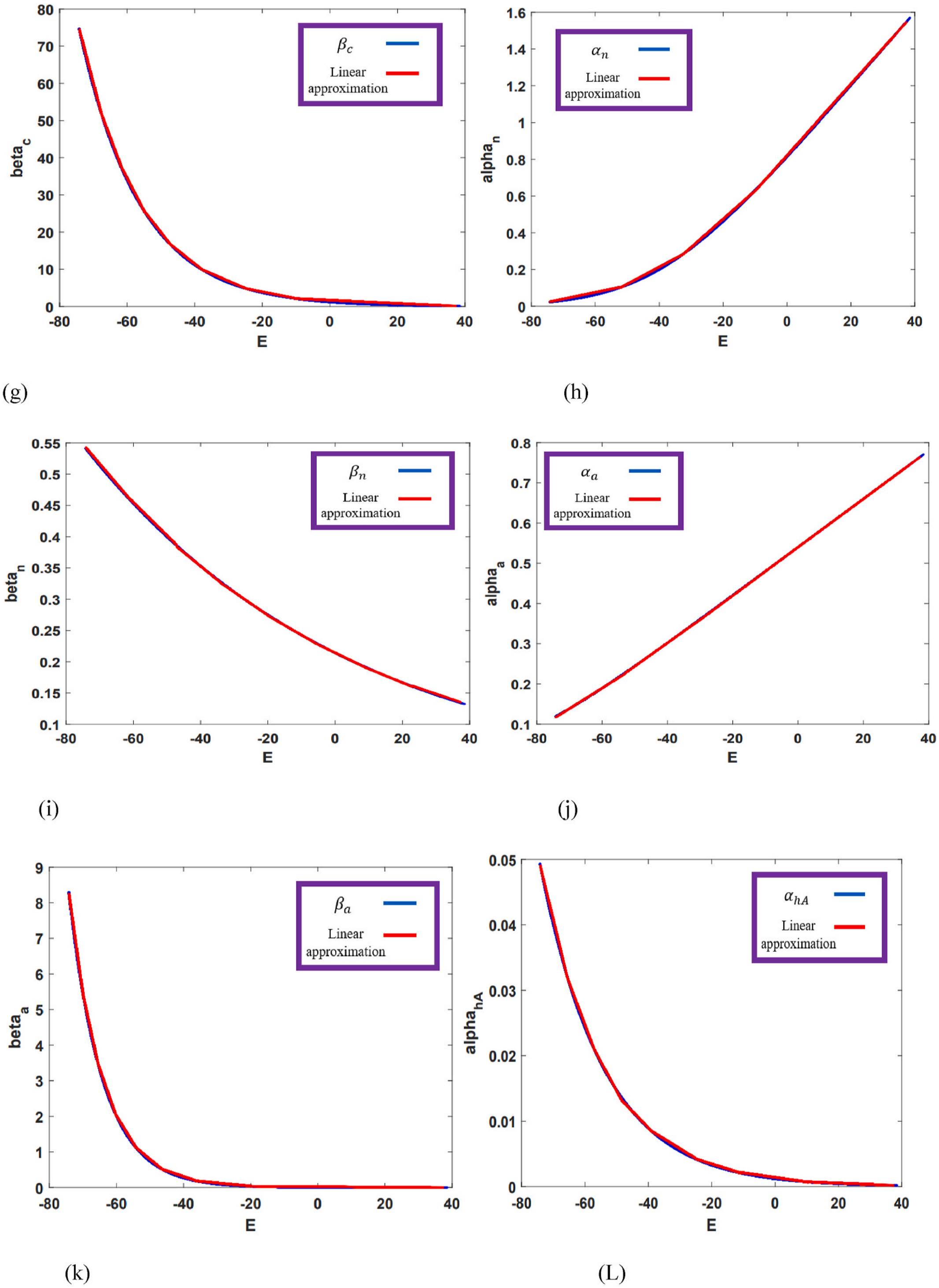


(a) (b)

(c) (d)

(e) (f)

Fig. 1. Linear approximations used to simplify the nonlinear terms of RGC model. In all the figures, the blue color shows the nonlinear curve and the red color shows the linear approximation. The linear approximation of $g_{K,Ca}$, α_m, β_m, α_h, β_h, α_c, β_c, α_n, β_n, α_a, β_a, α_{h_A}, β_{h_A}, $\widetilde{g_{Na}}m^3$, $\widetilde{g_K}n^4$ was shown in panels (a) to (p), respectively.

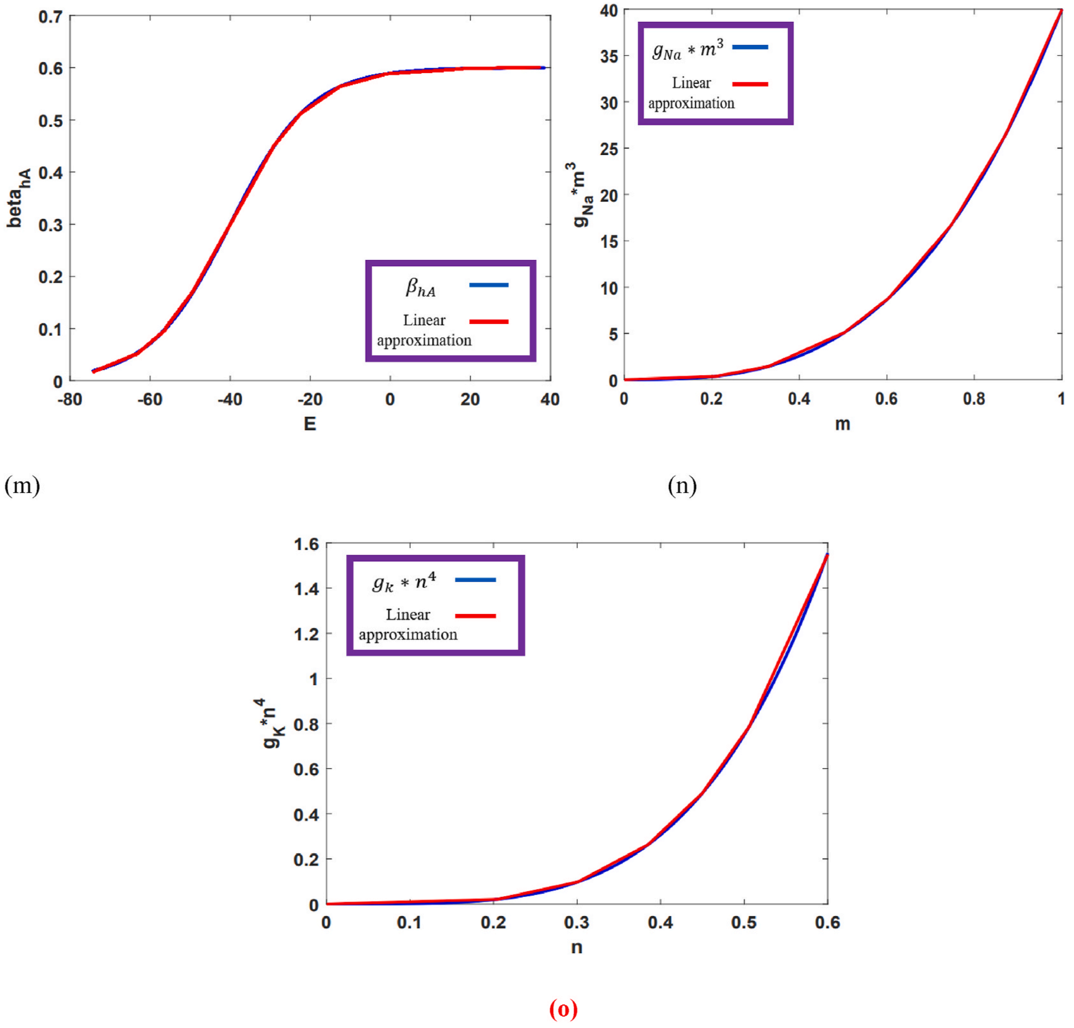


(g) (h)

(i) (j)

(k) (L)

Fig. 1. (*continued*).



(m)

(n)

(o)

Fig. 1. (*continued*).

$$g_{K,Ca} = \widetilde{g_{K,Ca}} \frac{\left(\frac{[Ca^{2+}]}{Ca^{2+}{}_{diss}}\right)^2}{1+\left(\frac{[Ca^{2+}]}{Ca^{2+}{}_{diss}}\right)^2} = X_1 Ca^{2+} + Y_1 \tag{23}$$

$$\alpha_m = \frac{-0.6(E+30)}{e^{-0.1(E+30)}-1} = X_2E + Y_2 \tag{24}$$

$$\beta_m = 20e^{\frac{-(E+55)}{18}} = X_3E + Y_3 \tag{25}$$

$$\alpha_h = 0.4e^{\frac{-(E+50)}{20}} = X_4E + Y_4 \tag{26}$$

$$\beta_h = \frac{6}{e^{-0.1(E+20)}+1} = X_5E + Y_5 \tag{27}$$

$$\alpha_C = \frac{-0.3(E+13)}{e^{-0.1(E+13)}-1} = X_6E + Y_6 \tag{28}$$

$$\beta_C = 10e^{\frac{-(E+38)}{18}} = X_7E + Y_7 \tag{29}$$

$$\alpha_n = \frac{-0.02(E+40)}{e^{-0.1(E+40)} - 1} = X_8 E + Y_8 \tag{30}$$

$$\beta_n = 0.4e^{\frac{-(E+50)}{80}} = X_9 E + Y_9 \tag{31}$$

$$\alpha_a = \frac{-0.006(E+90)}{e^{-0.1(E+90)} - 1} = X_{10} E + Y_{10} \tag{32}$$

$$\beta_a = 0.1e^{\frac{-(E+30)}{10}} = X_{11} E + Y_{11} \tag{33}$$

$$\alpha_{h_A} = 0.04e^{\frac{-(E+70)}{20}} = X_{12} E + Y_{12} \tag{34}$$

$$\beta_{h_A} = \frac{0.6}{e^{-0.1(E+40)} + 1} = X_{13} E + Y_{13} \tag{35}$$

$$\widetilde{g_{Na}} m^3 = X_{14} m + Y_{14} \tag{36}$$

$$\widetilde{g_{Ca}} C^3 = X_{15} C + Y_{15} \tag{37}$$

$$\widetilde{g_A} a^3 = X_{16} a + Y_{16} \tag{38}$$

$$\widetilde{g_K} n^4 = X_{17} n + Y_{17} \tag{39}$$

$$\begin{aligned} C_m \frac{dE}{dt} = & -(X_{14}m + Y_{14})h(E - E_{Na}) - (X_{15}C + Y_{15})(E - E_{Ca}) - (X_{17}n + Y_{17})(E - E_K) - (X_{16}a + Y_{16})h_A(E - E_K) \\ & - \left(X_1 Ca^{2+} + Y_1\right)(E - E_K) + I_{Stim} - 0.8 \end{aligned} \tag{40}$$

The constant values of equations (23)–(39) are reported in Table 2.

Fig. 1(a)–(p) shows the linear approximations presented in equations (23)–(39 (see Table 2)). The process of linear approximation is done using the search algorithm in such a way that there is a compromise between the number of lines and the error.

In **Equations 41 and 42**, error criteria *NMAE* and *NRMSE* are introduced to calculate the error between nonlinear curves and their equivalent linear approximations and Table 3 reports the calculated errors. Unfortunately, no simplification or approximation has been provided on RGC model, so it is not possible to compare with previous works (see Table 4).

$$\text{Normalized Mean Absolute Error (NMAE)} = \frac{1}{n \times Max|R_{Nonlnear\ Curve}|} \sum_{i=1}^{n} |R_{Nonlinear\ Curve} - R_{Lnear\ Curve}| \tag{41}$$

Table 3
The error criteria NMAE and NRMSE between nonlinear curves and their equivalent linear approximations.

Approximate Equation	NMAE	NRMSE
$g_{K,Ca}$	0.0077	0.0108
α_m	0.1767	0.3106
β_m	0.3033	0.4168
α_h	0.2916	0.4019
β_h	0.2776	0.4842
α_c	0.1555	0.2829
β_c	0.2901	0.4003
α_n	0.1852	0.3201
β_n	0.2030	0.3016
α_A	0.1852	0.2938
β_A	0.2866	0.3923
α_{h_A}	0.2905	0.4028
β_{h_A}	0.2845	0.4676
$\widetilde{g_{Na}} m^3$	9.1912e-07	9.4119e-04
$\widetilde{g_{Ca}} C^3$	9.5358e-07	9.7648e-04
$\widetilde{g_A} a^3$	7.6236e-07	7.8067e-04
$\widetilde{g_K} n^4$	9.2497e-07	9.4718e-04

$$\text{Normalized Root Mean Square Error (NRMSE)} = \frac{1}{Max(R_{Nonlnear\ Curve})}\sqrt{\frac{\sum_{i=1}^{n}(R_{Nonlinear\ Curve} - R_{Lnear\ Curve})^2}{n}} \tag{42}$$

In the hardware implementation of biological models, there is a compromise between error and resources consumption, so that more hardware resources should be used for higher accuracy. By consuming less hardware resources, the error can be increased to the extent that the system dynamics is maintained. The following validation performed to ensure the preservation of the dynamics of the model in exchange for the reported error.

3.1. Validation of LRGC model

This subsection delves into the individual compatibility of two LRGC and RGC models inside the spiking reactions of the phase space, nullcline space, and attractors' rejection and pull in.

A. Examining the LRGC Model's Time Domain Behavior

Four error criteria have been proposed in the form of equations (43)–(46) in order to examine the concordance of the time space spiking response of the RGC and LRGC model:

$$\text{Normalized Mean Absolute Error (NMAE)} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{R_{RGC\ model} - R_{LRGC\ model}}{max(R_{RGC\ model} - R_{LRGC\ model})}\right| \tag{43}$$

$$\text{Correlation} = \frac{cov(R_{RGC\ model}, R_{LRGC\ model})}{\sigma_{R_{LRGC\ model}}\sigma_{R_{RGC\ model}}} \tag{44}$$

$$\text{Normalized Root Mean Square Error (NRMSE)} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(\frac{R_{RGC\ model} - R_{LRGC\ model}}{max(R_{RGC\ model} - R_{LRGC\ model})}\right)^2} \tag{45}$$

$$\text{Normalized Frequency Difference (NFD)} = \frac{|Mean\ of\ frequnce\ RGS\ responce - Mean\ of\ frequnce\ LRGS\ responce|}{Maximum\ frequence\ RGS\ responce} \tag{46}$$

In Eqs.(43)–(46), $R_{RGC\ model}$, $R_{LRGC\ model}$ can be any of the variables, E, n, m, …in the model of RGC and LRGC, respectively. The mean of absolute value of error, mean square error, correlation and similarity of responses, and the difference in frequency of spikes are reported as error criteria NMAE, NRMSE, Correlation, and NFD, respectively, in Table 4. Error criteria between all RGC and LRGC model variables are presented for input current $I = 2$ in Table 4. Considering that the approximate model of ganglion cells has not been presented so far, it is not possible to compare the reported errors with previous works (see Table 4).

Table 4's results make it clear that the suggested LRGC model closely mimics the behavior of the original RGC model in the time domain, with minimal inaccuracy, particularly when it comes to spike frequency. Previous study [33] confirmed that ganglion cells modulate information in the frequency of spikes and the results of Table 4 show that the frequency error of the LRGC model is very low, which is a strong point for the proposed model.

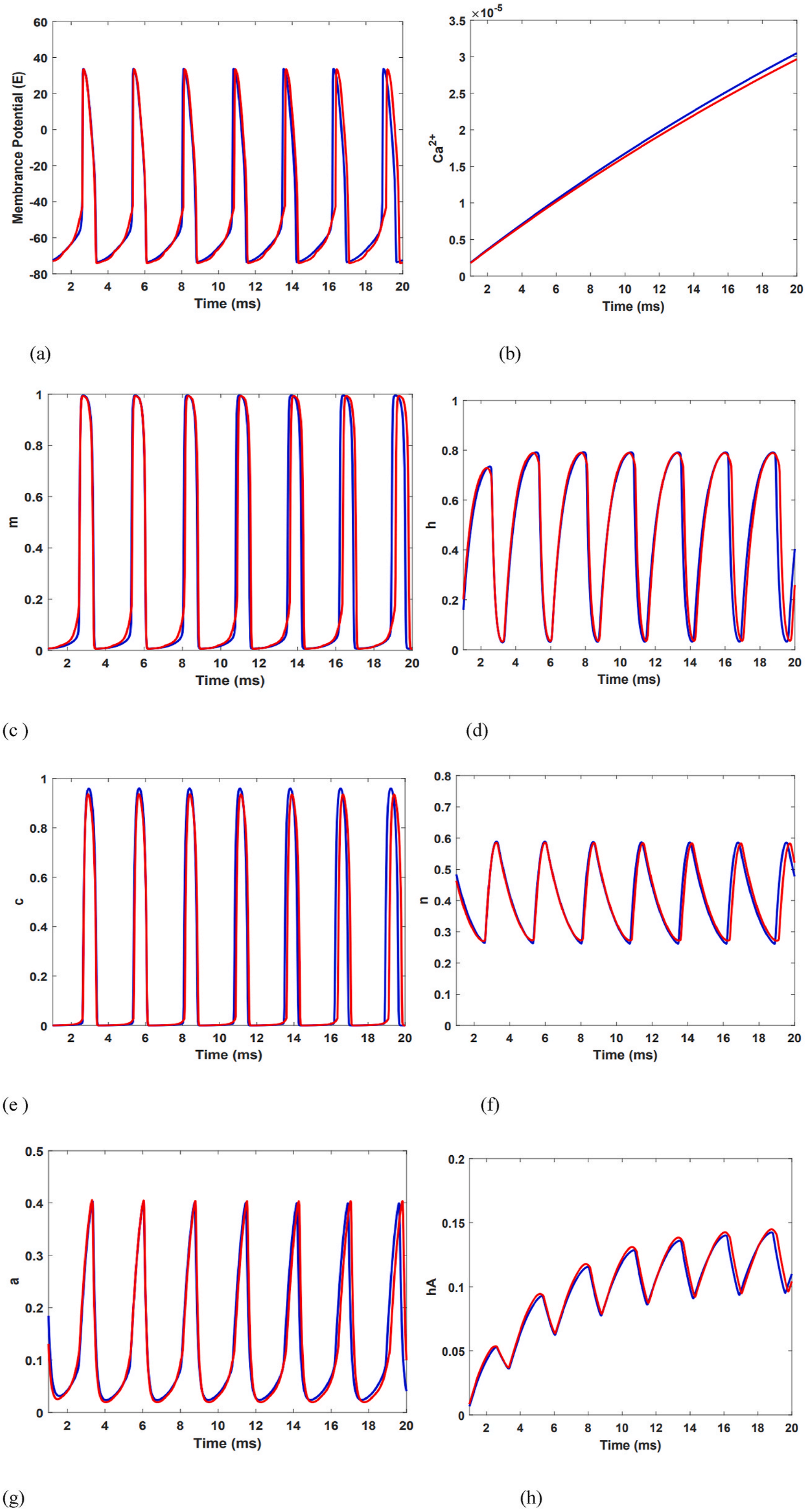
Fig. 2(a)–(h), which displays all ganglion cell's variables for input currents $I = 2$, demonstrates how closely the suggested LRGC model coordinates with the original RGC model within the reaction of the time space.

B Examining The LRGC Model's Dynamic Behavior

The coordinating of the dynamic behavior of models should be examined after confirming the high concordance of the LRGC model with the RGC inside the time space spike response. As a result, Figs. 3–5 depict, respectively, the behavior of trajectories in nullcline

Table 4
Comparison of the time space response of LRGC model compared to the RGC model.

Object variable	NMAE	Correlation	NRMSE	NFD
E	0.0846	0.8888	0.2074	0.0001
Ca^{2+}	0.5332	1	0.6062	–
M	0.0725	0.8980	0.2103	0.0002
H	0.0814	0.9581	0.1641	0.0000
C	0.0689	0.8978	0.1869	0.0002
N	0.1050	0.9592	0.1832	0.0001
A	0.0724	0.9403	0.1497	0.0001
h_A	0.1921	0.9958	0.2653	0.0000



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 2. The time response of the variables *E, Ca^{2+}, m, h, c, n, a, h_A* of the LRGC (red color) and RGC models (blue color) for input currents *I*=*2* was shown respectively in Panes (a) to (h).

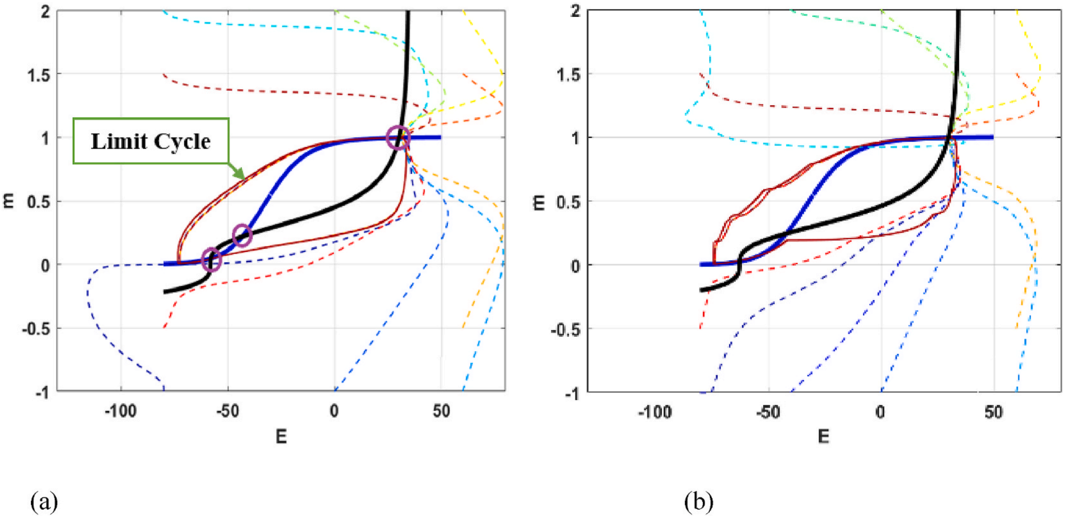


Fig. 3. The E-m nullclines in the RGC model (a) and LRGC model (b). It is clear that both the original and suggested models have the same quantity and kind of equilibrium points.

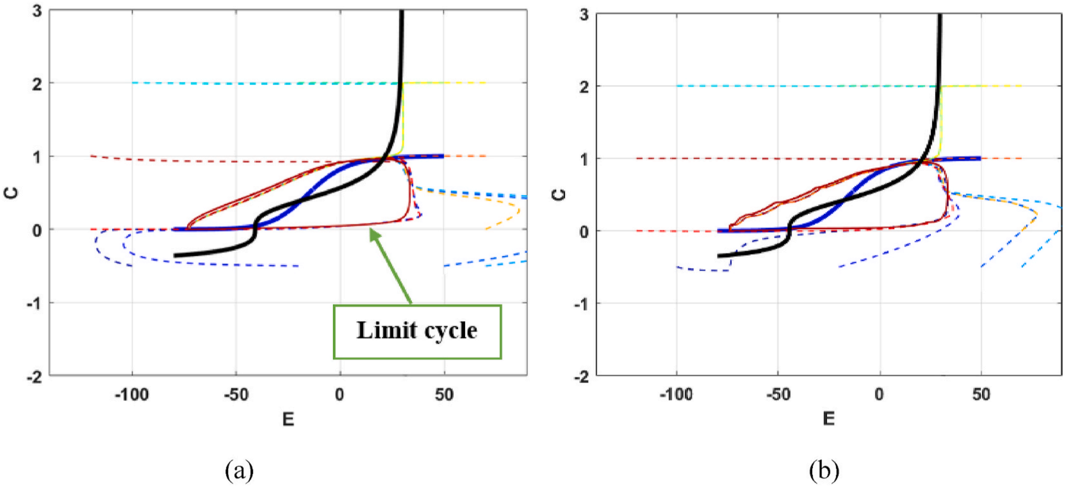
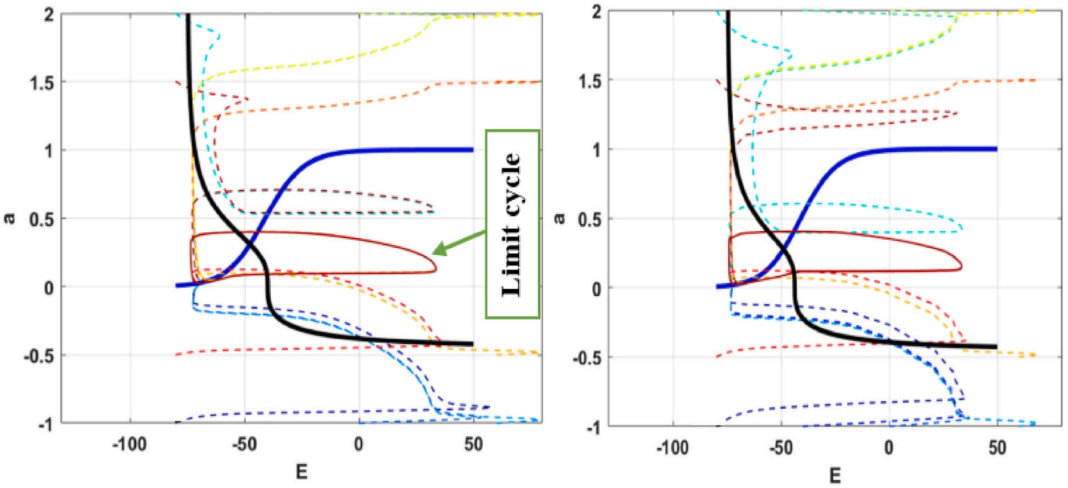


Fig. 4. The E-c nullclines in the RGC model (a) and LRGC model (b). The behavior of the trajectories in the suggested model and the original model exactly match.

space E-m, E-c, and E-a for the LRGC and RGC models. The similarity of the motion direction of trajectories in the nullcline space indicates that the number and kind of equilibrium points in the RGC and LRGC models are the same. A dynamic system's stability is largely dependent on the quantity and kind of equilibrium points, hence it is crucial to maintain these throughout the approximation process [5]. The collision focuses of nullclines, which are represented by a circle in Fig. 3(a), correspond to the equilibrium points of the dynamical system. The locations of the equilibrium points for the RGC and LRGC models are equal, as shown in Figs. 3–5. Additionally, the final behavior of the trajectories within the nullcline space can be used to determine the type of equilibrium points. As shown in Fig. 3 (E-m nullcline space), 4 (E-c nullcline space), and 5 (E-a nullcline space), the behavior of the trajectories for the RGC and LRGC models is completely coordinated. Figs. 3–5 depict the movement of the trajectories for 12 beginning locations in the three nullcline spaces (E-m, E-c, and E-a) of the original and approximation model.

The first E-nullcline and m-nullcline for the RGC (Eq. (47)) and LRGC models (Eq. (48)) have been computed in the following.



(a) (b)

Fig. 5. The E-a nullclines in the RGC model (a) and LRGC model (b).

$$RGC\ Model: \begin{cases} E-nullcline: \dfrac{dE}{dt}=0 \\ m-nullcline: \dfrac{dm}{dt}=0 \end{cases} \rightarrow \begin{cases} m=\sqrt[3]{\dfrac{-\widetilde{g_{Ca}}C^3(E-E_{Ca})-\widetilde{g_K}n^4(E-E_K)-\widetilde{g_A}a^3h_A(E-E_K)-g_{K,Ca}(E-E_K)+2}{\widetilde{g_{Na}}h(E-E_{Na})}} \\ m=\dfrac{\dfrac{-0.6(E+30)}{e^{-0.1(E+30)}-1}}{\left(\dfrac{-0.6(E+30)}{e^{-0.1(E+30)}-1}+20e^{\frac{-(E+55)}{18}}\right)} \end{cases} \tag{47}$$

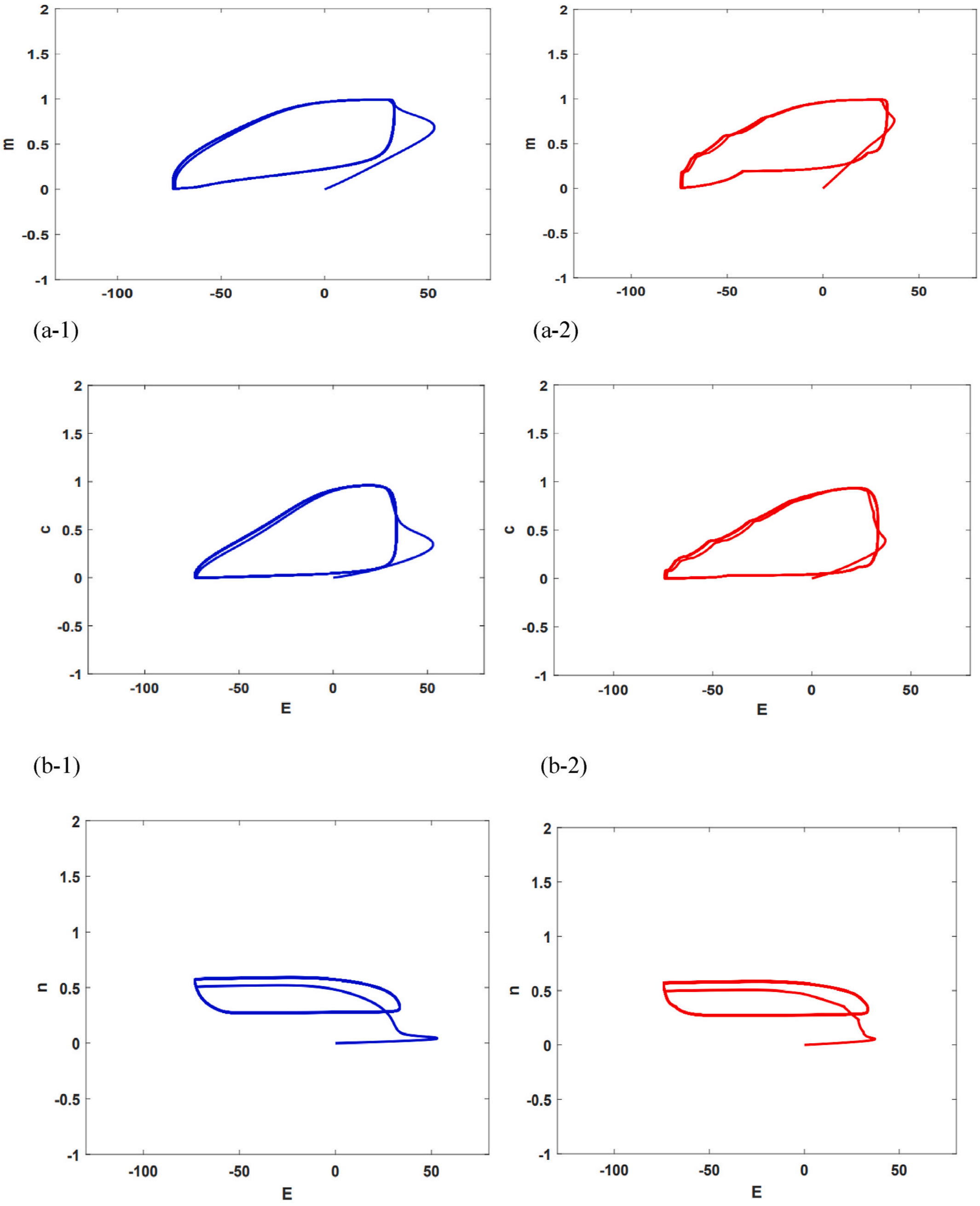
$$LRGC\ Model: \begin{cases} E-nullcline: \dfrac{dE}{dt}=0 \\ m-nullcline: \dfrac{dm}{dt}=0 \end{cases}$$
$$\rightarrow \begin{cases} m=\sqrt[3]{\dfrac{-(X_{15}E+Y_{15})(E-E_{Ca})-(X_{17}E+Y_{17})(E-E_K)-(X_{16}E+Y_{16})h_A(E-E_K)-\left(X_1Ca^{2+}+Y_1\right)(E-E_K)+2}{(X_{14}E+Y_{14})h(E-E_{Na})}} \\ m=\dfrac{X_2E+Y_2}{X_2E+Y_2+X_3E+Y_3} \end{cases} \tag{48}$$

Other variables set to a fixed value to indicate the two-dimensional nullcline space (E−nullcline (blue curve) and m-nullcline (black curve)) and to improve the transparency of the motion direction of all trajectories. Fig. 3 shows the E-nullcline and m-nullcline in the RGC (Fig. 3(a)) and LRGC (Fig. 3 (b)) models. As it is evident, the trajectories of all 12 paths eventually incline towards the specified limit cycle. In fact, in the E-m nullcline space of the RGC and LRGC model, there is a stable limit cycle that absorbs the trajectories. Because the trajectories in Fig. 3(a) and (b) behave similarly, the equilibrium points of the RGC and LRGC models produced by the collision of nullclines are equal.

$$\begin{aligned}&Equilibrium\ \text{points of RGC}\\ &\text{Model}=\begin{cases} point1=(30.3,1.006) \\ point2=(-43.2,0.2174) \\ point3=(-57.9,0.0418) \end{cases}\end{aligned}$$

$$\begin{aligned}&Equilibrium\ \text{points of LRGC}\\ &\text{Model}=\begin{cases} point1=(30.25,1.005) \\ point2=(-43.22,0.2184) \\ point3=(-57.83,0.042) \end{cases}\end{aligned}$$

Subsequently, the RGC (Eq. (49)) and LRGC (Eq. (50)) models' E− and c-nullclines have been computed.



(a-1)

(a-2)

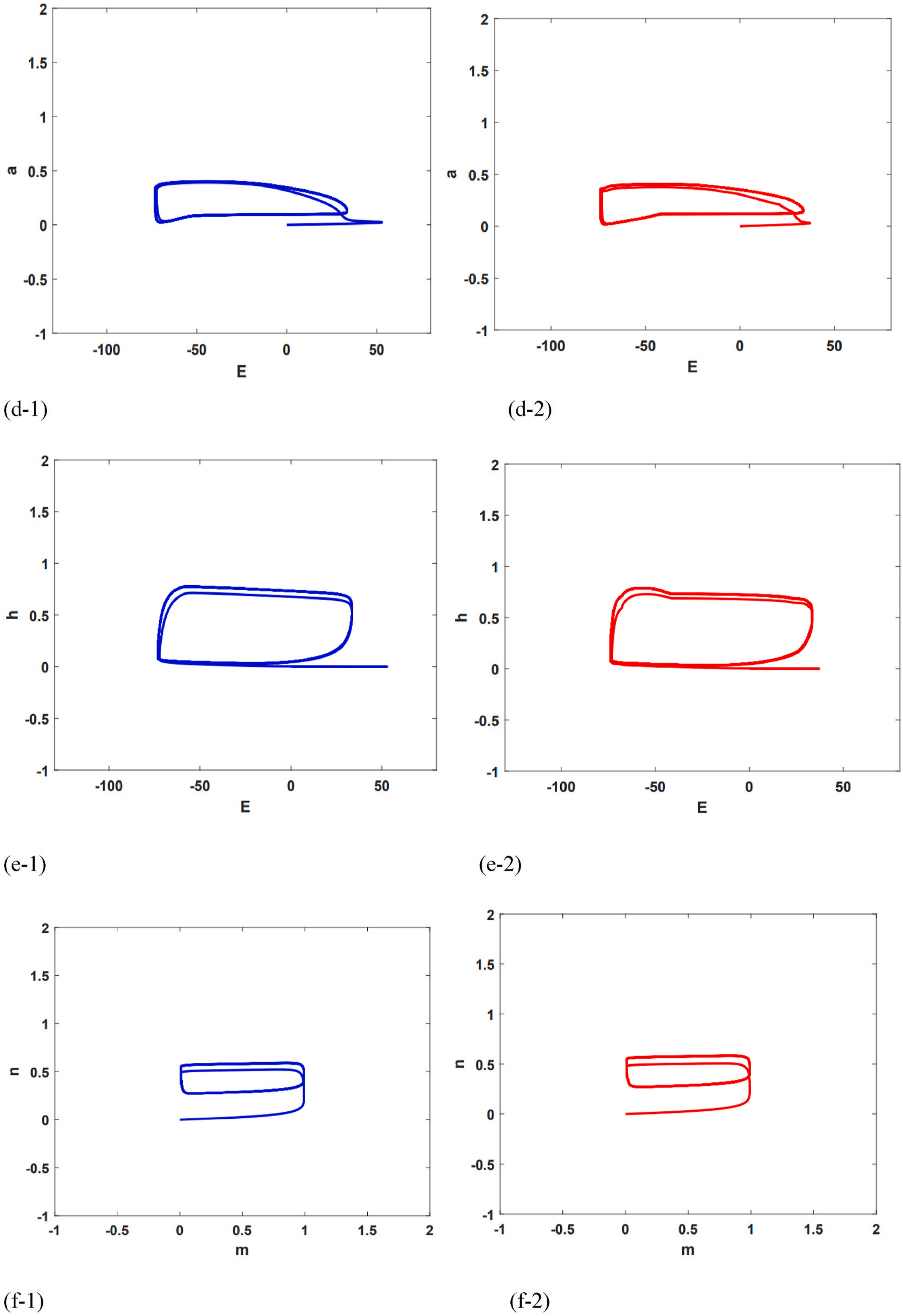
(b-1)

(b-2)

(c-1)

(c-2)

Fig. 6. Show the phase space of the RGC and LRGC model in blue and red color, respectively. Plotting the phase space is done for different pairs of variables: (a) *E*-*m*, (b) *E*-*c*, (c) *E*-*n*, (d) *E*-*a*, (e) *E*-*h*, (f) *m*-*n*, (g) *m*-*c*, (h) *m*-*a*, (I) *n*-*a*, (J) *n*-*c*.

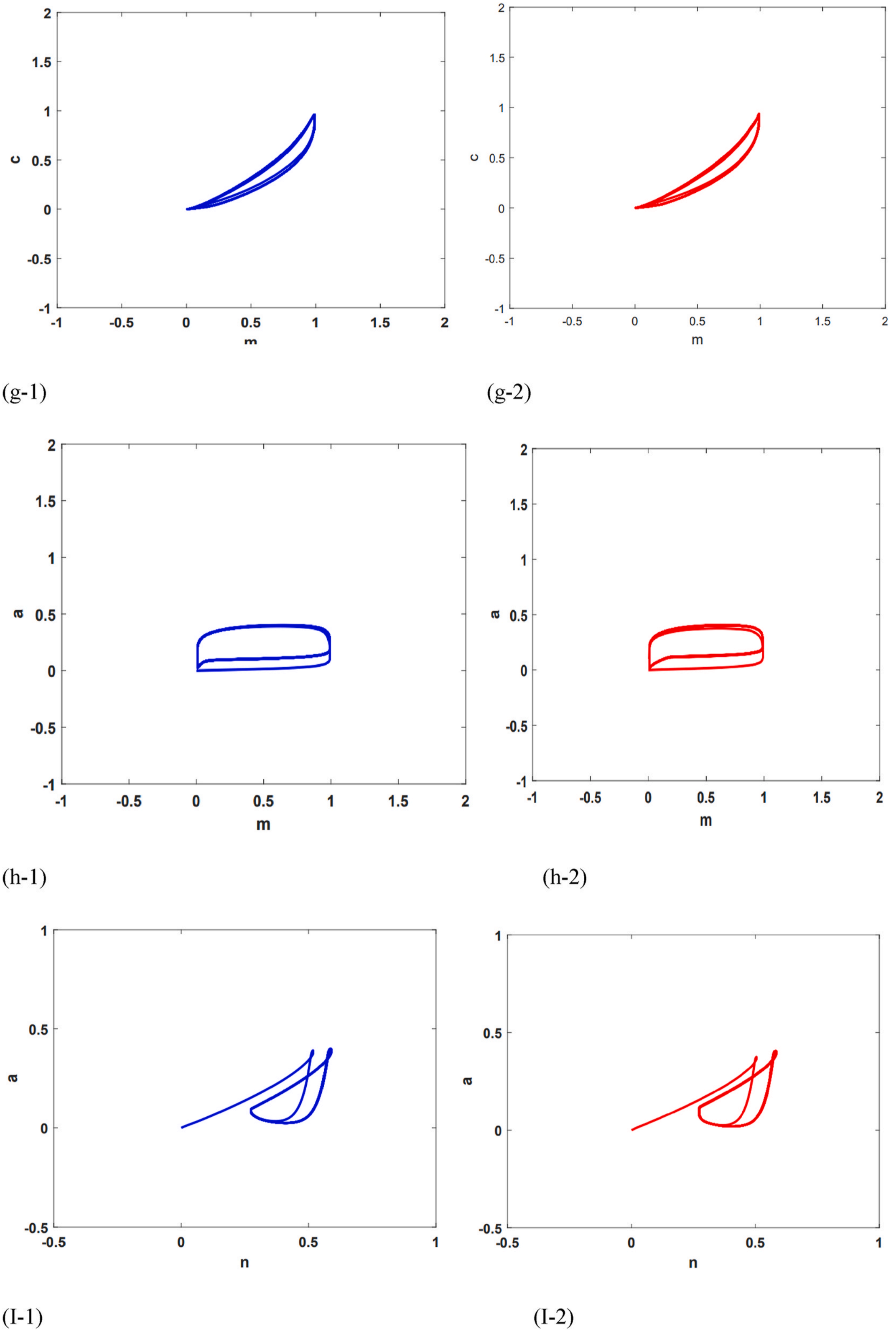


(d-1) (d-2)

(e-1) (e-2)

(f-1) (f-2)

Fig. 6. (*continued*).

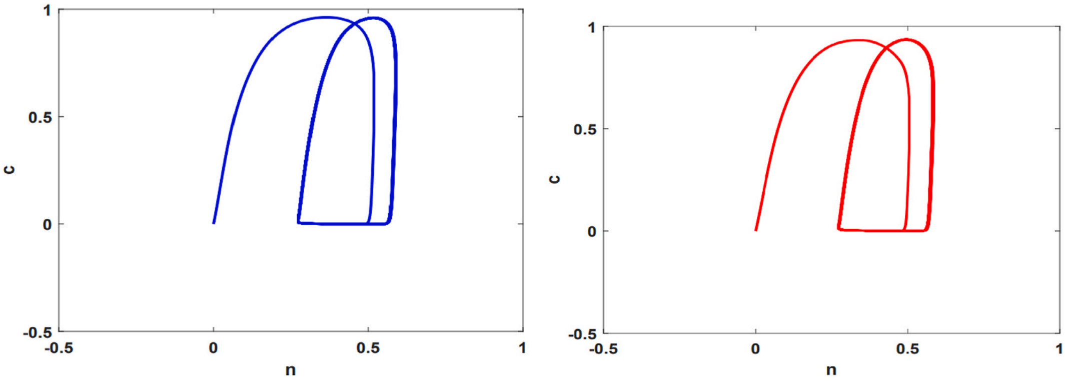


(g-1) (g-2)

(h-1) (h-2)

(I-1) (I-2)

Fig. 6. (*continued*).



(J-1) (J-2)

Fig. 6. (*continued*).

$$RGC\ Model: \begin{cases} E - nullcline: \dfrac{dE}{dt} = 0 \\ c - nullcline: \dfrac{dc}{dt} = 0 \end{cases}$$

$$\rightarrow \begin{cases} c = \sqrt[3]{\dfrac{-\widetilde{g_{Na}}m^3h(E-E_{Na}) - \widetilde{g_K}n^4(E-E_K) - \widetilde{g_A}a^3h_A(E-E_K) - g_{K,Ca}(E-E_K) + 2}{\widetilde{g_{Ca}}(E-E_{Ca})}} \\ c = \dfrac{\dfrac{-0.3(E+13)}{e^{-0.1(E+13)}-1}}{\dfrac{-0.3(E+13)}{e^{-0.1(E+13)}-1} + 10e^{\frac{-(E+38)}{18}}} \end{cases} \tag{49}$$

$$LRGC\ Model: \begin{cases} E - nullcline: \dfrac{dE}{dt} = 0 \\ c - nullcline: \dfrac{dc}{dt} = 0 \end{cases}$$

$$\rightarrow \begin{cases} c = \sqrt[3]{\dfrac{-(X_{14}E+Y_{14})h(E-E_{Na}) - (X_{17}E+Y_{17})(E-E_K) - (X_{16}E+Y_{16})h_A(E-E_K) - \left(X_1Ca^{2+}+Y_1\right)(E-E_K) + 2}{(X_{15}E+Y_{15})(E-E_{Ca})}} \\ c = \dfrac{X_6E+Y_6}{X_6E+Y_6+X_7E+Y_7} \end{cases} \tag{50}$$

Additionally, other variables are set to a fixed value to indicate the two-dimensional nullcline space (E−nullcline and c-nullcline) and to increase the transparency of the motion direction of all trajectories. The RGC (Fig. 4 (a)) and LRGC (Fig. 4(b)) models' E-nullcline (blue curve) and c-nullcline (black curve) are depicted in Fig. 4. In this instance as well, all 12 pathways' trajectories finally absorb in the direction of the designated stable limit cycle. The following are the RGC and LRGC models' equilibrium points:

$$\begin{array}{l} E\text{quilibrium points of RGC} \\ \text{Model} = \begin{cases} point1 = (21, 0.97) \\ point2 = (-19.1, 0.39) \\ point3 = (-40.8, 0.0485) \end{cases} \end{array}$$

$$\begin{array}{l} E\text{quilibrium points of LRGC} \\ \text{Model} = \begin{cases} point1 = (21.07, 0.975) \\ point2 = (-19.07, 0.0.38) \\ point3 = (-40.82, 0.0485) \end{cases} \end{array}$$

Additionally, the RGC (Eq. (51)) and LRGC (Eq. (52)) models' E− and a-nullclines have been computed.

$$RGC\ Model:\begin{cases}E-nullcline:\dfrac{dE}{dt}=0\\[2ex] a-nullcline:\dfrac{da}{dt}=0\end{cases}\rightarrow\begin{cases}a=\sqrt[3]{\dfrac{-\widetilde{g_{Na}}m^3h(E-E_{Na})-\widetilde{g_K}n^4(E-E_K)-\widetilde{g_{Ca}}C^3(E-E_{Ca})-g_{K,Ca}(E-E_K)+2}{\widetilde{g_A}h_A(E-E_K)}}\\[3ex] a=\dfrac{\dfrac{-0.006(E+90)}{e^{-0.1(E+90)}-1}}{\dfrac{-0.006(E+90)}{e^{-0.1(E+90)}-1}+0.1e^{\frac{-(E+30)}{10}}}\end{cases}\tag{51}$$

$$LRGC\ Model:\begin{cases}E-nullcline:\dfrac{dE}{dt}=0\\[2ex] a-nullcline:\dfrac{da}{dt}=0\end{cases}$$
$$\rightarrow\begin{cases}a=\sqrt[3]{\dfrac{-(X_{14}E+Y_{14})h(E-E_{Na})-(X_{17}E+Y_{17})(E-E_K)-(X_{15}E+Y_{15})(E-E_{Ca})-\left(X_1Ca^{2+}+Y_1\right)(E-E_K)+2}{(X_{16}E+Y_{16})h_A(E-E_K)}}\\[3ex] a=\dfrac{X_{10}E+Y_{10}}{X_{10}E+Y_{10}+X_{11}E+Y_{11}}\end{cases}\tag{52}$$

Other variables are assigned to a fixed value in order to display the two-dimensional nullcline space (E−nullcline and a-nullcline). The RGC (Fig. 5(a)) and LRGC (Fig. 5(b)) models' E-nullcline (blue curve) and a-nullcline (black curve) are displayed in Fig. 5. In the nullcline space of the RGC and LRGC models, there is a stable limit cycle that attracts all trajectories. The equilibrium points of the RGC and LRGC models is as follows:

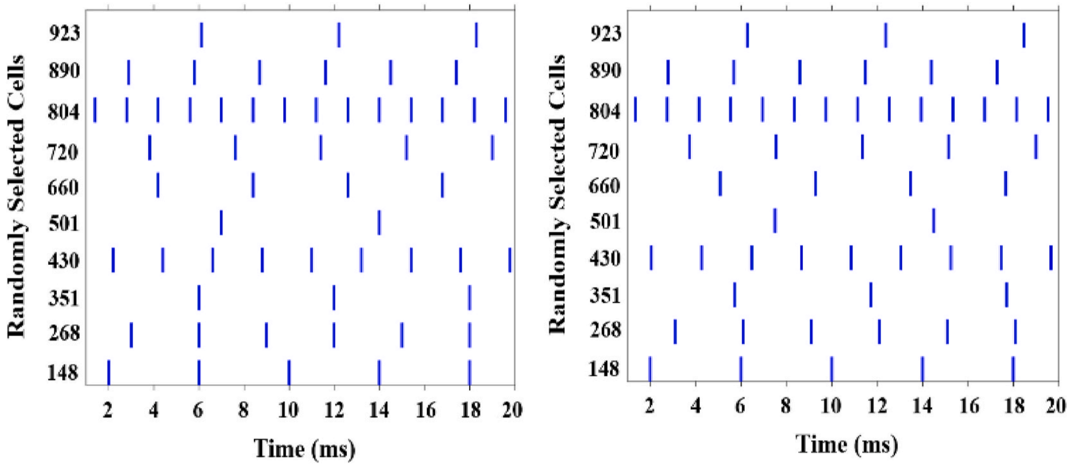
$$Equilibrium\ \text{point of RGC Model} = (-47.3, 0.304)$$

$$Equilibrium\ \text{point of LRGC Model} = (-47.31, 0.302)$$

The findings displayed in Figs. 3–5 verify that the LRGC model fully retains the dynamic characteristics of the RGC model. As a result, the suggested approximation can closely resemble the behavior of the original cell model in both time and nullclines space, while also reducing hardware costs.

C Examining the LRGC Model's Phase Space Behavior

Phase space analysis may be an incredibly useful tool for examining a system's dynamic behavior. The LRGC model is further supported by examining the behavior of the phase space of the primary variables (*E, c, m, n, a, h, hA*) in the original and proposed ganglion cell models. Fig. 6(a)–(j) shows the phase space with the input stimulus current of 2 for the LRGC model drawn in red and the RGC model drawn in blue. The test involves graphing the changes of the primary variables in relation to one another; the blue and red graph matching validates the proposed model's correctness.



(a) (b)

Fig. 7. Raster plot shows the spiking response of ten randomly chosen RGC_network (a) and LRGC_network (b) cells.

The behavior of the approximate model's phase space in various pairings of variables is consistent with the original model, as seen in Fig. 6.

Up to this point, the approximation model has followed the original model in three different tests. A much more thorough investigation into the conformance of the collective behavior of the RGC and LRGC models is presented in the section that follows. This involves the proposal of a network consisting of 1000 cells, with a connection chance of 0.2 between any two cells.

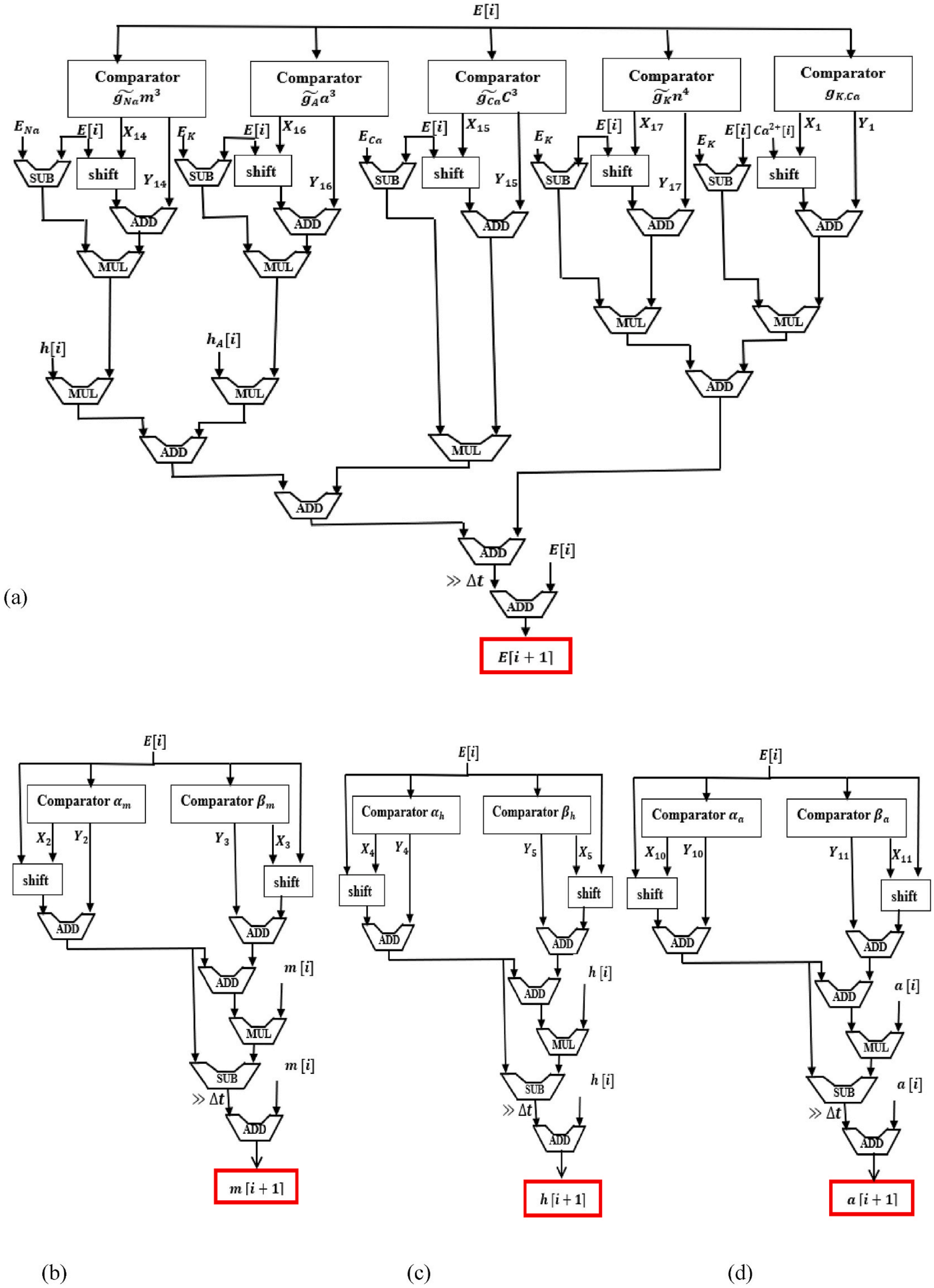


Fig. 8. Scheduling diagram of the variables *E, Ca^{2+}, m, n, C, h, a, h_A* in the proposed LRGC model was shown respectively in panels (a) to (h).

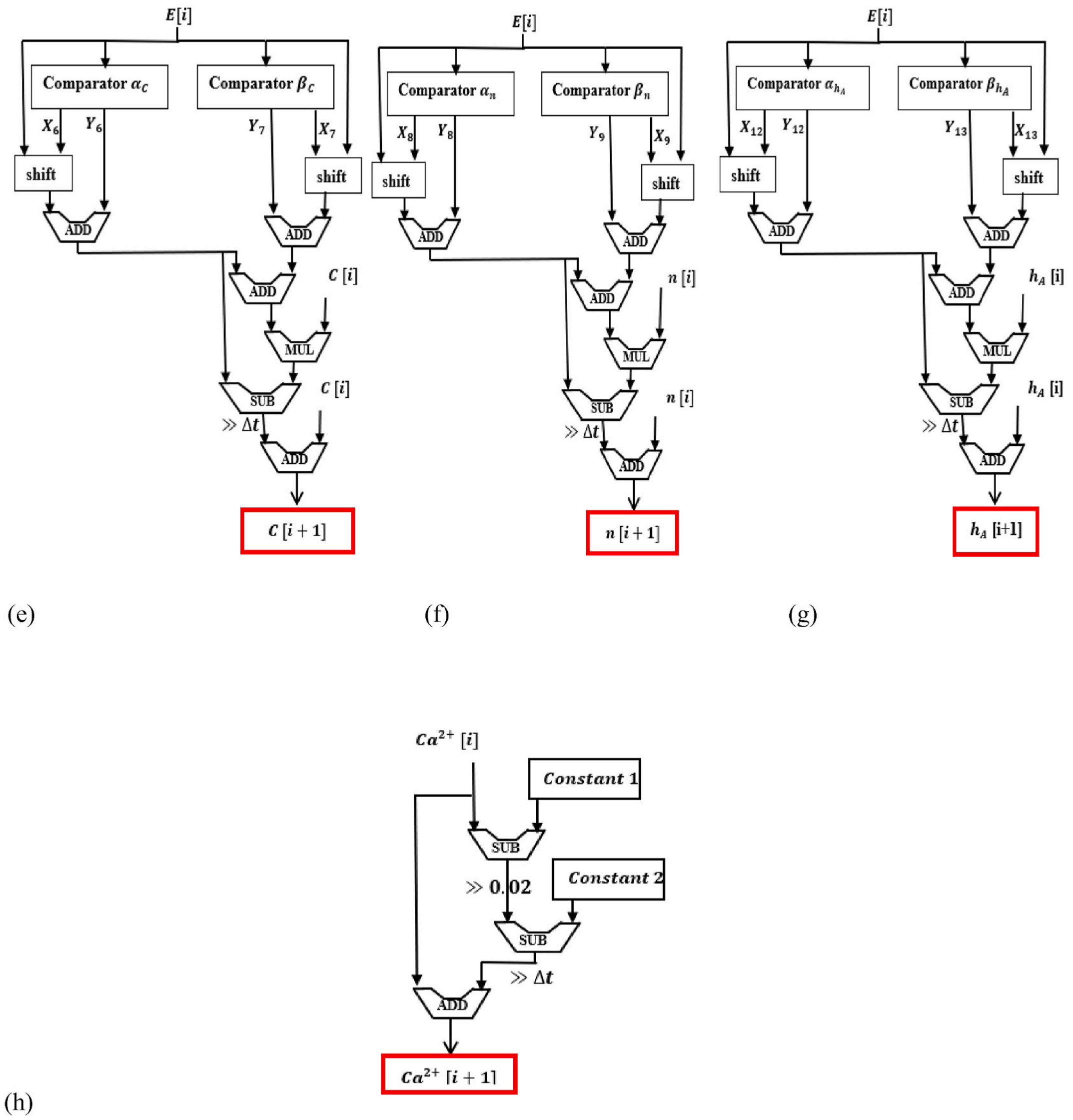


Fig. 8. (*continued*).

3.1.1. *Large scale network of LRGC model*

Considering that the objective of low-cost digital design of a biological cell is the feasibility of hardware implementation of a neural population, the compatibility of the collective behavior of the LRGC and RGC models should be considered. In this way, two populace of 1000 cells were created, which one utilizing RGC model (RGC_network) and the other utilizing LRGC models (LRGC_network). In both systems, cells are randomlly associated with a likelihood of 0.2. In the RGC_network and LRGC_network, each cell is approximately connected to 200 cells, and each network often contains about 200,000 synapses. In both systems, synapses that connect neurons are regarded as weighted synapses, with a constant weight of 1. Fig. 7(a) and (b) show the raster plot of the spiking reaction of 10 randomly chosen cells from the RGC_network (a) and LRGC_network (b), assuming that input current *I* is set to 2 for both networks.

The overall behavior of the randomly connected RGCs in RGC_network is evidently entirely consistent with that of the LRGCs in LRGC_network, as seen by Fig. 7. Hence, if the digital design of a ganglion cell is done with minimal resources and maximum speed, there is an assurance that the collective hardware behavior of the proposed ganglion cells is highly consistent with the original model.

4. Hardware design of LRGC model

This section discusses the effective hardware design of the suggested LRGC model as the RGC model contains nonlinear elements that necessitate the use of multipliers and dividers in the digital design. The linear approximation has taken the place of the nonlinear variables in the LRGC model, and addition, subtraction, and shift can be used to create further parts of the digital circuit. In comparison to the RGC model, the digital architecture of the LRGC requires reduced resource use due to the approximation of non-linear components.

A. The Procedure of Selecting Bit-Width and Parameter

In this strategy, shift and addition are used in place of the multiplier to multiply the fixed parameters in the variables. As a result, only shift can be used to perform delta multiplication when choosing parameters like Δt.

Since reducing hardware costs is the main goal of this design, all computations have been defined as fixed-point operations. The range of changes in the dynamic system variables is taken into consideration when designing a fixed-point operation. As a result, in the worst scenario, variables and parameters are represented by 7 bits for the integer part, 24 bits for the fractional part, and 1 bit as a sign bit. This indicates that 32 bits is the maximum length for a register. Least bit length is taken into consideration for fixed point calculations of each variable during the digital design phase in order to reduce the use of hardware resources. The hardware model's registers are really not all 32 bits; just 20 % of them are, and 40 % of them are 24 bits. The remaining registers, on the other hand, are fewer than 20 bits.

B. Discretization of Differential Equations

Differential equations can be solved in a variety of ways [34–36], although the iterative approaches using equation discretization are the main focus of this work. For the purpose of designing digital circuits, all of the continuous differential equations in the suggested LRGC model should be discretized. There are several other discretization procedures, such as Runge–Kutta and Euler; the first order Euler strategy is used since it is accurate and straightforward. The LRGC model's discretized equations are computed as follows through Eq. (53) to Eq. (60):

$$\begin{aligned}[i+1] = E[i] + \frac{\Delta t}{C_m}\Big[&-(X_{14}E[i]+Y_{14})h[i](E[i]-E_{Na}) - (X_{15}E[i]+Y_{15})(E[i]-E_{Ca}) - (X_{17}E[i]+Y_{17})(E[i]-E_K) \\ &-(X_{16}E[i]+Y_{16})h_A[i](E[i]-E_K) - \left(X_1 Ca^{2+}[i]+Y_1\right)(E[i]-E_K) + I_{Stim}\Big]\end{aligned} \tag{53}$$

$$Ca^{2+}[i+1] = Ca^{2+}[i] + \Delta t\left[-0.000015\, I_{Ca} - (0.02)\left(\left[Ca^{2+}[i]\right] - 0.0001\right)\right] \tag{54}$$

$$m[i+1] = m[i] + \Delta t[X_2E[i]+Y_2 - (X_2E[i]+Y_2+X_3E[i]+Y_3)m[i]] \tag{55}$$

$$h[i+1] = h[i] + \Delta t[X_4E[i]+Y_4 - (X_4E[i]+Y_4+X_5E[i]+Y_5)h[i]] \tag{56}$$

$$C[i+1] = C[i] + \Delta t[X_6E[i]+Y_6 - (X_6E[i]+Y_6+X_7E[i]+Y_7)C[i]] \tag{57}$$

$$n[i+1] = n[i] + \Delta t[X_8E[i]+Y_8 - (X_8E[i]+Y_8+X_9E[i]+Y_9)n[i]] \tag{58}$$

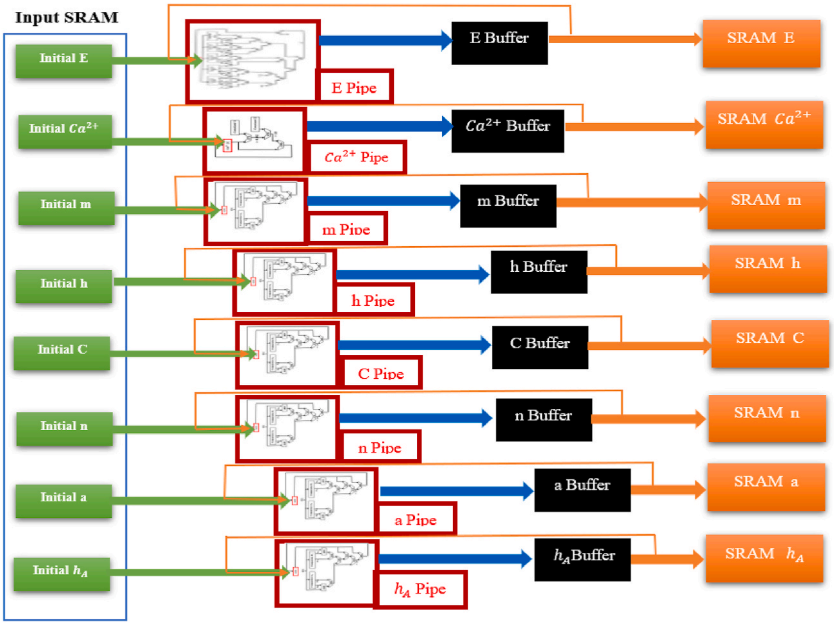


Fig. 9. Overall hardware diagram of LRGC digital circuit.

Table 5
Resource utilization of digital circuits of LRGC and RGC models on FPGA virtex II XC2VP30 board.

Digital model	Slice Flip Flop	4-in-LUT	Speed MHz	Multiplier	Adder	Subtractor	power consumption (W)
RGC model	18,210	24,550	22	10	15	12	0.54
LRGC model	1760	11,200	110	2	5	5	0.31

$$a[i+1] = a[i] + \Delta t[X_{10}E[i] + Y_{10} - (X_{10}E[i] + Y_{10} + X_{11}E[i] + Y_{11})a[i]] \tag{59}$$

$$h_A[i+1] = h_A[i] + \Delta t[X_{12}E[i] + Y_{12} - (X_{12}E[i] + Y_{12} + X_{13}E[i] + Y_{13})h_A[i]] \tag{60}$$

Δt, the discretization step in all equations, is conveniently multiplied by only 9 times shift right because it is thought to be 1/256.

C. Scheduling Schematics

The digital circuit for the suggested LRGC model's dynamic equations is shown in this section. The hardware synthesized circuits for the discrete equations of the LRGC model, numbered 53 through 60, are displayed in Fig. 8(a)–(h). Non-linear terms have been removed from this digital architecture and replaced with comparator blocks that correspond to linear approximations; shift and addition operations have been used in place of multipliers.

D An Overview of the LRGC Digital Circuit Structure

Fig. 9 shows the overall hardware diagram for the LRGC model. The advantage of the suggested model is that it can be implemented on low-cost hardware, as was covered in the previous sections. In many tests, the proposed LRGC model demonstrated good accuracy and low error, following the behavior of the RGC model. The scheduling diagram indicates that the majority of the LRGC model's digital blocks can only be implemented by using inexpensive digital modules like shift, comparator, adder, and subtractor blocks. The general hardware diagram is set up so that the digital blocks *E, Ca2+, m, n, C, h, a,* and h_A are supported with the constant parameters and initial values of the modules called from the associated memory resource. The execution times of cell computations are thought to be sped up by pipes *E, Ca2+, m, n, C, h, a,* and h_A; nonetheless, the estimated parallelization causes resource use to increase only somewhat.

E. Cost of Digital Design

Low-cost computing blocks based on linear approximation have replaced non-linear terms in the proposed LRGC model in order to reduce the cost of hardware implementation. A comparison of the resources used in the RGC and LRGC models' digital designs is presented in Table 5. In comparison to the original RGC model, the proposed LRGC model has the lowest hardware resource use and the highest frequency, as indicated by the findings shown in Table 5. On the other hand, considering that LRGC model has a high functional match with the RGC model from the point of view of spiking reaction, dynamic behavior in the phase space, and collective behavior, it can be a suitable candidate for the realization of neural population on hardware platforms (see Table 6).

The reason for using LUTs is that piecewise linear approximations are synthesized and implemented through LUTs. But the advantage of using LUTs is that the speed and working frequency of the circuit is much higher than when the DSP is used. The comparison of the power consumption of RGC model (0.54 W) with LRGC model (0.31 W) shows that the proposed approach in digital implementation has reduced the power consumption by 57 %. Ganglion cell model is approximated for the first time in this paper and its hardware is presented. Therefore, it is not possible to compare hardware resources, power consumption and approximation accuracy of LRGC with previous methods. However, a comparison with previous methods in hardware implementation of Hodgkin Huxley neuron has been considered. The reason for choosing the Hodgkin-Huxley neuron is that it has closer dynamics to the ganglion cell compared to other neurons. Although the ganglion cell has more complex dynamics than the Hodgkin-Huxley neuron.

As seen by Table 6's results, the approximation method based on linear approximation provides a high working frequency, while the ganglion cell has three more ion channels than the Hodgkin-Huxley neuron and subsequently much more computational complexity. In terms of hardware resources, it is not fair to compare a ganglion cell to a Hodgkin-Huxley neuron because a ganglion cell has much more complex dynamics. But the comparison confirms the superiority of the approach and method proposed in this paper. Sadly, hardware accuracy is lacking from most neural approximation works. However, the suggested model, although hardware efficient, has a good functional adaptation of the original model in terms of time space, trajectories movement in nullcline space, phase space, and large scale cell population.

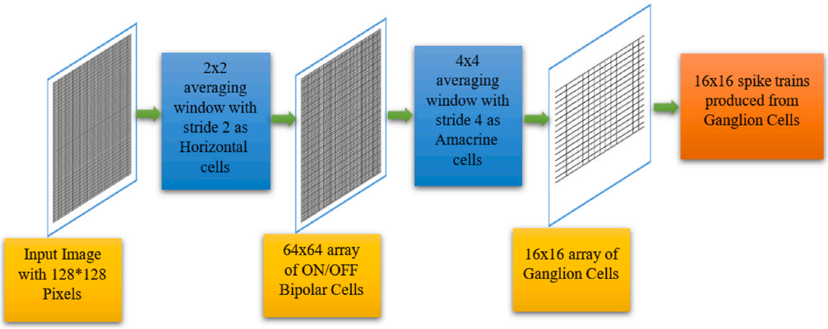
On the other hand, due to the fact that ganglion cells modulate information in the frequency of spikes, therefore, the amplitude error is not very important and the frequency error is very important. The results of Table 4 show the very low frequency error of the proposed LRGC model and the original model.

5. Discussion

Digital implementation of biophysical cells has gained attention in recent years, since it enables the implementation of biologically

Table 6
Comparison of resource utilization, speed, and error between digital implementation of ganglion cell with HH neuron.

method	Digital design method	LUT	RAM	FFs	DSP	Bit	Max Speed	power consumption	Error
Hodgkin–Huxley [28]	CORDIC	8457	32	7571	45	32	100 MHz	N/A	N/A
Hodgkin–Huxley [29]	Function approximation	2103	N/A	N/A	56	32	71.4 MHz	N/A	N/A
Hodgkin–Huxley [24]	CORDIC/LUT	23514	N/A	7231	99	32	38.4 MHz	N/A	N/A
Hodgkin–Huxley [27]	CORDIC	86032	14	30528	1112	32	63.3 MHz	N/A	N/A
Hodgkin–Huxley [26]	CORDIC	5660	N/A	2840	0	40	85 MHz	N/A	N/A
Hodgkin–Huxley [25]	CORDIC	4735	N/A	1552	16	N/A	100 MHz	N/A	N/A
LRGC (Proposed)	Linear approximation	11200	68	1760	2	32	110 MHz	0.31 W	0.393 (NRMSE)



2x2 averaging window with stride 2 as Horizontal cells
4x4 averaging window with stride 4 as Amacrine cells
16x16 spike trains produced from Ganglion Cells
Input Image with 128*128 Pixels
64x64 array of ON/OFF Bipolar Cells
16x16 array of Ganglion Cells

Fig. 10. The conversion of input image into the spike trains through retinal pathway.

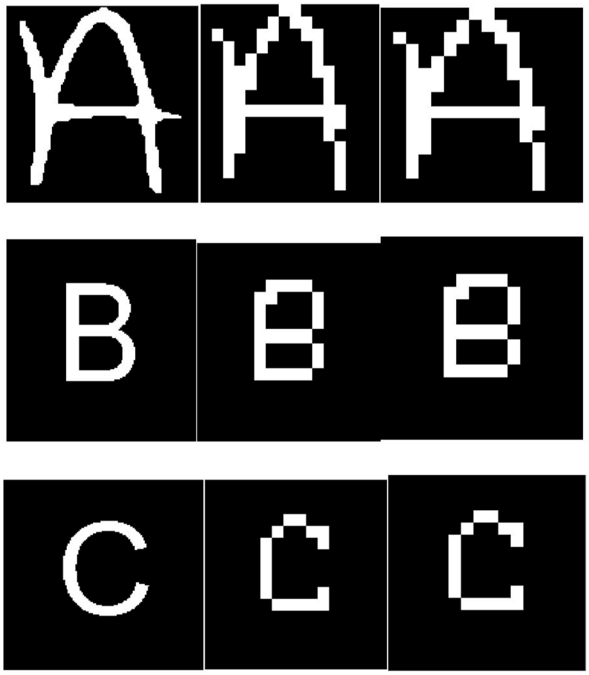


Fig. 11. Converting the spike trains generated from RGC and LRGC models within the retinal pathway into an image with 16 x 16 pixels. The first column shows the 128x128 pixel images given as input to the retina path. The second column is reconstructed images with 16 x 16 pixels from 16 x 16 spike trains which these spike trains are generated from the original ganglion cell model (RGC). The third column is reconstructed images with 16 x 16 pixels from 16 x 16 spike trains which these spike trains are generated from the proposed ganglion cell model (LRGC).

derived processing systems in the form of massive neuronal populations on hardware. Even if the processing capability of today's computers is enormous, it is crucial to create processors that can mimic the nervous system's reflexes in order to boost the machines' cognitive capacities. The brain's neurons work together to develop the sophisticated cognitive abilities that are the shortcoming of today's clever technologies. In order to achieve this, we can generate populations of biological hardware cells that require less resources than current machines. Eventually, these systems will be able to perform cognitive tasks that surpass those of current computers. Providing effective, affordable digital hardware for biological cell models, like RGCs, is the first part of this work's goal. With this hardware, an approximate model (LRGC) can mimic the behavior of the original model (RGC). Next, we simulated a network of 1000 cells to demonstrate that the collective behavior of LRGCs is comparable to that of RGCs. The performance of the suggested LRGC model has been tested in image processing applications since it has a collective behavior that is comparable to RGC and closely resembles the behavior of the original RGC model up until this point. At this level, a test for retinal pathway-based image reconstruction using RGC and the proposed LRGC model was developed to ensure accurate imitation of RGC by the LRGC model in practical applications.

In the retinal pathway, the input image passes through horizontal cells where it is averaged with a 2x2 window and a step size of 2 to produce an electrical signal of size 64x64. These electrical signals are passed through bipolar cells [15] and then they are further averaged by amacrine cells using 4×4 window with 4 pixels steps to produce electrical signals of size 16×16. Finally, ganglion cells [16] convert these electrical signals from amacrine cells into spike trains of 16×16 each. An overview of this process is shown in Fig. 10.

As shown in Fig. 10, the input 128x128 image is transformed into a 16x16 or 256 spike trains by passing through the retinal pass. The interesting result is that each spike train generated from the ganglion cells was mapped to an image pixel and reconstruct the input image. To interpret spike trains into image pixels, a mapping defined in a previous study [17] is used. Considering the fact that ganglion cells modulate information with the frequency of spikes [33], the reconstruction of images from the spike trains is performed using the mapping of the frequency interval [0-max frequency of spike trains] into the pixel brightness [0–255] [17]. Finally, the reconstructed gray scale image is converted to a binary image that can be better compared with the original image.

Therefore, in this study, two retinal routes are presented according to Fig. 10, one uses the original model of ganglion cells (RGC) and the other uses the proposed model of ganglion cells (LRGC), and then using the introduced coding, image with 16x16 pixels is generated from the 16x16 spike trains and compared with the original image with 128*128 pixels in Fig. 11. According to the results presented in Fig. 11, the reconstructed image of spike trains produced by LRGC model is completely identical to the reconstructed image of spike trains of RGC model. Thus, as reported in Fig. 11 and the previous results, the LRGC model has followed the performance of the RGC model with high accuracy in various tests, while LRGC model has a much lower hardware cost.

Considering that the frequency error of the LRGC model is very low and image reconstruction is performed using the mapping of the frequency into the pixel brightness, the error between the image reconstruction of retinal pathway with RGC model and LRGC model is zero. This point confirmed the effectiveness of the proposed LRGC circuit which is a strong point for the proposed model.

Limiting the range of changes of dynamic variables for approximation is one of the work's constraints; this is, of course, also taken into account by other hardware approximation techniques. Additionally, the frequency of FPGA boards and hardware resource limitations restrict the potential of hardware implementation in large-scale neural populations that are similar to the brain in real time. However, these limitations can be partially solved by building a corresponding neuromorphic chip.

6. Conclusion

The digital implemented circuit of the suggested LRGC model, an efficient digital circuit for the RGC model, was proposed in this research. The RGC model's nonlinear terms have been swapped out for effective linear approximation blocks in the LRGC model. The proposed LRGC circuit uses less power than the RGC model, it takes up less area and operates at a higher frequency. The response of the two models within the time space, the trajectory motion within the nullcline space, and their phase space behavior have been compared in order to examine the imitation accuracy of the LRGC model in comparison to the original RGC model, which was also confirmed that the LRGC model was highly adaptable. This work takes into consideration that the collective behavior of approximated ganglion cells (LRGC) should correspond with ganglion cells (RGC), since the hardware implementation of the neural cell population is a prerequisite for the digital design of a biological cell. In order to reconstruct images from spike trains, a retinal pathway based on the suggested LRGC model was created, and a significant degree of agreement between the RGC and LRGC model was discovered. Thus, the effective digital design of the LRGC model presented in this research can be applied as a low-cost neuromorphic platform for machine vision applications.

Funding

This study was not funded.

Data availability

Data would be available through corresponding author with reasonable request.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

CRediT authorship contribution statement

Soheila Nazari: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] I.B. Levitan, L.K. Kaczmarek, The Neuron: Cell and Molecular Biology, Oxford University Press, USA, 2002.
[2] Y. Humeau, D. Choquet, The next generation of approaches to investigate the link between synaptic plasticity and learning, Nat. Neurosci. 22 (10) (2019) 1536–1543.
[3] S. Yang, J. Wang, X. Hao, H. Li, X. Wei, B. Deng, K.A. Loparo, BiCoSS: toward large-scale cognition brain with multigranular neuromorphic architecture, IEEE Transact. Neural Networks Learn. Syst. 33 (7) (2021) 2801–2815.
[4] K. Roy, A. Jaiswal, P. Panda, Towards spike-based machine intelligence with neuromorphic computing, Nature 575 (7784) (2019) 607–617.
[5] M. Balubaid, O. Taylan, M.T. Yilmaz, E. Eftekhari-Zadeh, E. Nazemi, M. Alamoudi, Central nervous system: overall considerations based on hardware realization of digital spiking silicon neurons (DSSNs) and synaptic coupling, Mathematics 10 (6) (2022) 882.
[6] Z. Han, A.N. Islam, A. Sengupta, Astromorphic self-repair of neuromorphic hardware systems, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, 2023, June, pp. 7821–7829, 6.
[7] M. Amiri, S. Nazari, M. Janahmadi, Digital configuration of astrocyte stimulation as a new technique to strengthen the impaired astrocytes in the tripartite synapse network, J. Comput. Electron. 17 (2018) 1382–1398.
[8] Y. Zeng, D. Zhao, F. Zhao, G. Shen, Y. Dong, E. Lu, W. Bi, BrainCog: a spiking neural network based, brain-inspired cognitive intelligence engine for brain-inspired AI and brain simulation, Patterns 4 (8) (2023).
[9] J.P. Thivierge, É. Giraud, M. Lynn, Toward a brain-inspired theory of artificial learning, Cognitive Computation (2023) 1–8.
[10] D. Tal, E.L. Schwartz, Computing with the leaky integrate-and-fire neuron: logarithmic computation and multiplication, Neural Comput. 9 (2) (1997) 305–318.
[11] E.M. Izhikevich, Simple model of spiking neurons, IEEE Trans. Neural Network. 14 (6) (2003) 1569–1572.
[12] J.L. Hindmarsh, R.M. Rose, A model of neuronal bursting using three coupled first order differential equations, in: Proceedings of the Royal Society of London. Series B. Biological Sciences, 1984, pp. 87–102, 221(1222).
[13] C. Morris, H. Lecar, Voltage oscillations in the barnacle giant muscle fiber, Biophys. J. 35 (1) (1981) 193–213.
[14] A.L. Hodgkin, A.F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve, The Journal of physiology 117 (4) (1952) 500.
[15] P. Werginz, H. Benav, E. Zrenner, F. Rattay, Modeling the response of ON and OFF retinal bipolar cells during electric stimulation, Vis. Res. 111 (2015) 170–181.
[16] J.F. Fohlmeister, P.A. Coleman, R.F. Miller, Modeling the repetitive firing of retinal ganglion cells, Brain Res. 510 (2) (1990) 343–345.
[17] J.K. Eshraghian, K. Cho, C. Zheng, M. Nam, H.H.C. Iu, W. Lei, K. Eshraghian, Neuromorphic vision hybrid rram-cmos architecture, IEEE Trans. Very Large Scale Integr. Syst. 26 (12) (2018) 2816–2829.
[18] V. Kakkar, Comparative study on analog and digital neural networks, Int. J. Comput. Sci. Netw. Secur 9 (7) (2009) 14–21.
[19] A. Siddique, M.I. Vai, S.H. Pun, A low-cost, high-throughput neuromorphic computer for online SNN learning, Cluster Comput. (2023) 1–18.
[20] Y. Wang, O. Taylan, A.S. Alkabaa, I. Ahmad, E. Tag-Eldin, E. Nazemi, H.S. Alqabbaa, An optimization on the neuronal networks based on the ADEX biological model in terms of LUT-state behaviors: digital design and realization on FPGA platforms, Biology 11 (8) (2022) 1125.
[21] M. Amiri, S. Nazari, K. Faez, Digital realization of the proposed linear model of the H odgkin-H uxley neuron, Int. J. Circ. Theor. Appl. 47 (3) (2019) 483–497.
[22] J. Wang, Z. Peng, Y. Zhan, Y. Li, G. Yu, K.S. Chong, C. Wang, A high-accuracy and energy-efficient CORDIC based izhikevich neuron with error suppression and compensation, IEEE Transactions on Biomedical Circuits and Systems 16 (5) (2022) 807–821.
[23] T. Levi, F. Khoyratee, S. Saïghi, Y. Ikeuchi, Digital implementation of Hodgkin–Huxley neuron model for neurological diseases studies, Artif. Life Robot. 23 (2018) 10–14.
[24] S.Y. Bonabi, H. Asgharian, R. Bakhtiari, S. Safari, M.N. Ahmadabadi, FPGA implementation of Hodgkin-Huxley neuron model, in: International Conference on Neural Computation Theory and Applications, vol. 2, SciTePress, 2012, October, pp. 522–528.
[25] F. Khoyratee, F. Grassia, S. Saïghi, T. Levi, Optimized realtime biomimetic neural network on FPGA for bio-hybridization, Frontiers Neurosci. 13 (2019) 377 [Online], https://www.frontiersin.org/article/10.3389/fnins.2019.00377.
[26] F. Shama, S. Haghiri, M.A. Imani, FPGA realization of Hodgkin–Huxley neuronal model, IEEE Trans. Neural Syst. Rehabil. Eng. 28 (5) (2020) 1059–1068.
[27] S. Yaghini Bonabi, H. Asgharian, S. Safari, M. Nili Ahmadabadi, FPGA implementation of a biological neural network based on the Hodgkin-Huxley neuron model, Front. Neurosci. 8 (2014) 379.
[28] Z. Yi, W. Fu, L. Cao, Implementation of Hodgkin–Huxley spiking neuron model using FPGA, in: Proc. IEEE 19th Int. Conf. Commun. Technol. (ICCT), 2019, pp. 1535–1539.
[29] K. Akbarzadeh-Sherbaf, B. Abdoli, S. Safari, A.H. Vahabie, A scalable FPGA architecture for randomly connected networks of Hodgkin-Huxley neurons, Front. Neurosci. 12 (2018) 698.
[30] A.I. Cohen, Vertebrate retinal cells and their organization, Biol. Rev. 38 (4) (1963) 427–459.
[31] S.A. Lipton, D.L. Tauck, Voltage-dependent conductances of solitary ganglion cells dissociated from the rat retina, J. Physiol. 385 (1) (1987) 361–391.
[32] P. Lukasiewicz, F. Werblin, A slowly inactivating potassium current truncates spike activity in ganglion cells of the tiger salamander retina, J. Neurosci. 8 (12) (1988) 4470–4481.
[33] K. Koch, J. McLean, M. Berry, P. Sterling, V. Balasubramanian, M.A. Freed, Efficiency of information transmission by retinal ganglion cells, Curr. Biol. 14 (17) (2004) 1523–1530.
[34] O.A. Arqub, Z. Abo-Hammour, Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm, Inf. Sci. 279 (2014) 396–415.
[35] B. Maayah, A. Moussaoui, S. Bushnaq, O. Abu Arqub, The multistep Laplace optimized decomposition method for solving fractional-order coronavirus disease model (COVID-19) via the Caputo fractional approach, Demonstr. Math. 55 (1) (2022) 963–977.
[36] B. Maayah, O.A. Arqub, S. Alnabulsi, H. Alsulami, Numerical solutions and geometric attractors of a fractional model of the cancer-immune based on the Atangana-Baleanu-Caputo derivative and the reproducing kernel scheme, Chin. J. Phys. 80 (2022) 463–483.