Original Research

Characterization of multiscale logic operations in the neural circuits

JunHyuk Woo[1,2,†], Kiri Choi[3,†], Soon Ho Kim[1,†], Kyungreem Han[1,*], MooYoung Choi[2,*]

[1] *Laboratory of Computational Neurophysics, Convergence Research Center for Brain Science, Brain Science Institute, Korea Institute of Science and Technology, 02792 Seoul, Republic of Korea,* [2] *Department of Physics and Astronomy and Center for Theoretical Physics, Seoul National University, 08826 Seoul, Republic of Korea,* [3] *School of Computational Sciences, Korea Institute for Advanced Study, 02455 Seoul, Republic of Korea*

TABLE OF CONTENTS

1. Abstract

Background: Ever since the seminal work by McCulloch and Pitts, the theory of neural computation and its philosophical foundation known as ‘computationalism’ have been central to brain-inspired artificial intelligence (AI) technologies. The present study describes neural dynamics and neural coding approaches to understand the mechanisms of neural computation. The primary focus is to characterize the multiscale nature of logic computations in the brain, which might occur at a single neuron level, between neighboring neurons via synaptic transmission, and at the neural circuit level. **Results**: For this, we begin the analysis with simple neuron models to account for basic Boolean logic operations at a single neuron level and then move on to the phenomenological neuron models to explain the neural computation from the viewpoints of neural dynamics and neural coding. The roles of synaptic transmission in neural computation are investigated using biologically realistic multi-compartment neuron models: two representative computational entities, CA1 pyramidal neuron in the hippocampus and Purkinje fiber in the cerebellum, are analyzed in the information-theoretic framework. We then construct two-dimensional mutual information maps, which demonstrate that the synaptic transmission can process not only basic AND/OR Boolean logic operations but also the linearly non-separable XOR function. Finally, we provide an overview of the evolutionary algorithm and discuss its benefits in automated neural circuit design for logic operations. **Conclusions**: This study provides a comprehensive perspective on the multiscale logic operations in the brain from both neural dynamics and neural coding viewpoints. It should thus be beneficial for understanding computational principles of the brain and may help design biologically plausible neuron models for AI devices.

Submitted: 28 June 2021 Revised: 24 July 2021 Accepted: 5 August 2021 Published: 30 October 2021

2. Introduction

Neural computation is a popular concept in neuroscience [1–4]. It claims that the brain operates like a computer: a neuron is considered as the basic computational unit while local and global neural circuits are the infrastructures that may account for higher-level computations. This concept is rooted in the philosophical tradition known as computationalism [5–7]. The first mathematical interpretation given by Warren S. McCulloch and Walter Pitts in 1943 [8] suggests that neuronal activity is computational and thus small networks of artificial (model) neurons can mimic the cognitive function of the brain. Their idea was introduced into philosophy by Hilary Putnam in 1961 [7]. Ever since these seminal works, it has further developed to provide a framework for investigating the underlying principles of brain function and developing artificial intelligence technologies, including brain-inspired algorithms and neuromorphic devices.

Neural coding and neural dynamics are two complementary approaches to understanding the principles of neural computation [9, 10]. The neural coding approach, where a neuron is regarded as an information processing unit, focuses on explaining how the information is encoded, decoded, and transferred by the neuron. On the other hand, one may also consider a neuron as a dynamical system [11, 12] that changes its state over time; this leads to the neural dynamics approach. The dynamics is described typically by coupled differential equations involving time derivatives of variables representing relevant biological quantities. Their solutions are obtained either numerically via computer simulations or analytically. Although there have been skeptical perspectives on this approach as a valid basis for theories of brain function [13], it provides a useful tool to characterize the nonlinear behaviors of neurons that are essential for multimodal logic operations at single neuron and circuit levels [14–16]. Throughout this paper, we use the term ‘neural coding’ in a general sense (i.e., the neural representation of information) that normally permeates neuroscience, rather than in mentioning specifically the coding mechanism such as ‘rate coding’ and ‘temporal coding’.

Neurons have highly specialized structures with a variety of physical properties to facilitate information processing. Therefore their demand for cellular energy (e.g., adenosine triphosphate; ATP) is extraordinarily high [17, 18]. In the process of evolution, neurons are likely to have been optimized in the direction of minimizing the energy consumption for information coding. For survival, animals require to have highly energy-efficient information processing machinery [17–20]. In this context, the law of information is of primary importance to understand the design principles and functions of neurons, which naturally lend themselves to be explained with information theory. In the field of computational neurophysics, several metrics based on Shannon’s classical information theory have been used to characterize neural information processing [21]: mutual information measures the overlapping information between neurons (via synaptic transmission) or within a neuron [20, 22]. Transfer entropy quantifies the directional information flow [23]. Partial information decomposition (PID) allows measuring unique, shared, and synergistic contributions of multiple neuronal inputs to the output [24]. These information-theoretic measures are applicable to multiscale levels ranging from a single neuron and two neurons connected via a synapse to local and global neural circuits.

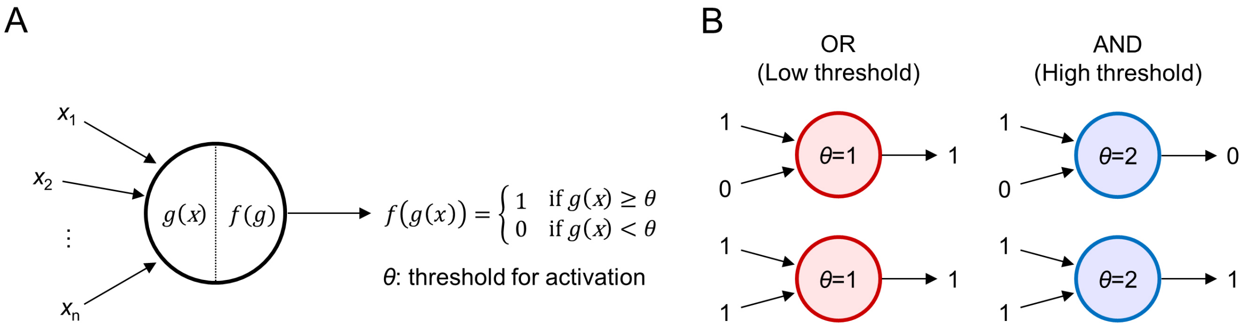
This study investigates the information-theoretic approaches to characterizing logic operations in model neurons, synapses, and neural circuits. This paper consists of six sections: In Section 3, simple neuron models (including McCulloch and Pitts model, linear-threshold model, and firing-rate model) and phenomenological models (integrate-and-fire models and Izhikevich model) are analyzed. Section 4 begins with the analysis of Hodgkin-Huxley type multi-compartment models for a cornu Ammonis 1 (CA1) pyramidal neuron in the hippocampus and Purkinje fiber (PF) in the cerebellum, which is extended to the cooperative and competitive computations of these neurons via homo and heterosynaptic transmissions. Section 5 examines methods to find a logic backbone at the neural circuit level. Finally, Section 6 discusses the results and concludes the paper.

3. Logic operations at single neuron level

3.1 Simple neuron models

Since the 1940s, simple artificial neurons have been developed as the building blocks of artificial intelligence (AI) technologies, including brain-inspired AI algorithms and neuromorphic devices [25, 26]. The artificial neurons are simple mathematical models conceived as a model of biological neurons; in general, they are designed to perform only basic arithmetical and Boolean logic operations [27–29]. Traditionally, only the basic dynamics and coding properties of biological neurons have been considered in developing simple neuron models. Specifically, details of individual synaptic currents and distinct dynamics of different types of spines are often disregarded because a single excitatory postsynaptic potential is typically much smaller in amplitude than the threshold for an action potential. The underlying notion of the simple neuron models is that a neuron can fire only when a sufficiently large number of excitatory synapses are activated simultaneously to drive its voltage over the threshold.

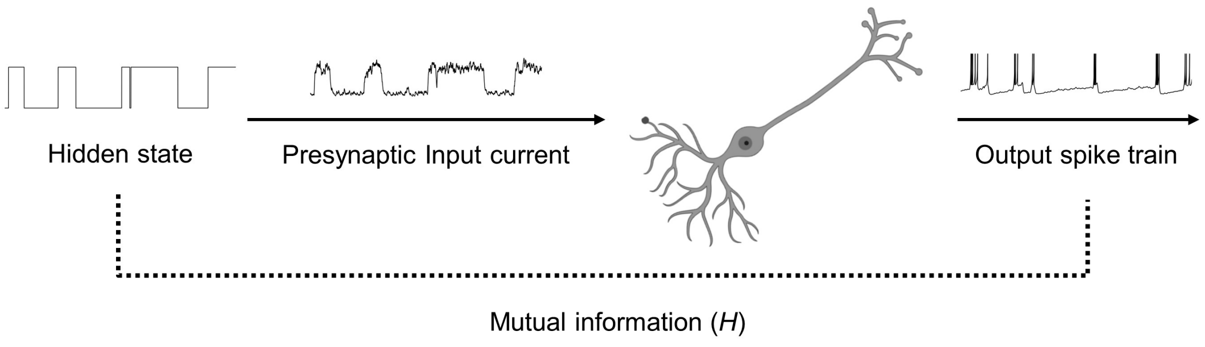
In 1943 Warren McCulloch and Walter Pitts developed the first mathematical neuron model [8], which takes multiple binary inputs and produces a single binary output. The neuron is characterized by the parameter θ denot-


A
x_1
x_2
⋮
x_n
$g(x)$
$f(g)$
$f(g(x)) = \begin{cases} 1 & \text{if } g(x) \geq \theta \\ 0 & \text{if } g(x) < \theta \end{cases}$
θ: threshold for activation
B
OR
(Low threshold)
AND
(High threshold)
1
0
θ=1
1
1
1
θ=1
1
1
0
θ=2
0
1
1
θ=2
1

C

Model	Input x	Output y	$y = f(g(x))$
McCulloch-Pitts	Binary	Binary	$y = H\left(\sum_i x_i - \theta\right)$
Linear-threshold	Continuously graded	Binary	$y = H\left(\sum_i w_i x_i - \theta\right)$
Firing-rate	Continuously graded	Continuously graded	$\tau \frac{dy}{dt} = -y + F\left(\sum_i w_i x_i\right)$

Fig. 1. Boolean logic operations in simple neuron models. (A) Schematic diagram of a simple neuron model with binary output upon synaptic inputs x_i (i = 1, 2, …, n). Functions g and f describe the integration of the inputs and the neuronal output. Integration of inputs $g(x)$ is compared with threshold θ to determine the neuronal output. (B) OR (left, colored in red) and AND (right, blue) operations are illustrated. The result of the operation with a single input is presented in the first row, followed by the result when both inputs are given (the second row). Depending on whether the threshold is low (θ = 1) or high (θ = 2), the information processing of the neuron models are mapped to OR and AND functions, respectively. (C) Description of simple neuron models. Here, H is the Heaviside step function defined by $H(u)$ = 1 for u $\geq$0 and $H(u)$ = 0; w_i denotes the strength or weight of the ith synapse; θ is the threshold of the neuron. The time constant τ represents the temporal response properties of the system as a whole, including the effects of both membrane and synaptic time constants. For constant current, the relationship between the total synaptic current I that a neuron receives and its firing rate is given in terms of a firing-rate function: $r = F(I)$.



$$H(X; I_{0\to t}) = \sum_{x, I_{0\to t}} p(x; I_{0\to t}) \log \frac{p(x; I_{0\to t})}{p(x)p(I_{0\to t})} = S(X) - S(X|I_{0\to t})$$

Fig. 2. Illustration of mutual information calculation. The binary hidden state triggers the presynaptic input current I. The mutual information $H(X; I_{0\to t})$ measures the overlapping information between the hidden state $X \equiv \{x\}$ and the history of postsynaptic spike trains $I_{0\to t}$ (for details see Eqns. 3,4,5).

ing the minimum number of excitatory synapses that can generate an action potential: if the number of synapses is greater than or equal to θ, the neuron is active (labeled as "1"); otherwise, it is inactive ("0"). This simple mathematical treatment also allows the Boolean logic operations, in which the binary values 1 and 0 correspond to "true" and "false". They remarked that the combination of such simple neuron models is capable of universal logic computations; this seminal work laid the foundations of developing brain-inspired digital electronic circuits for AI systems. Analyzing the McCulloch-Pitts (MP) neuron, one can establish basic Boolean logic operations (i.e., AND and OR operations) at the single neuron level (Fig. 1). With the threshold parameter θ set at a high value (e.g., equal to the total number of inputs), the neuron is active only if all the synaptic inputs are active, leading to the logical AND operation (Fig. 1A). Alternatively, with the threshold set at a low value, only a small portion of active synaptic inputs is enough to fire; this corresponds to the logical OR operation (Fig. 1B).

Similarly, the dynamics of linear-threshold (LT) [30] and firing-rate (FR) [31] models can be interpreted as AND/OR Boolean operations, with the given threshold θ (Fig. 1). The LT model neuron employs continuously graded input values to describe different contributions of synaptic inputs to the neuronal activation. Each synaptic input is assigned a weight (according to the relative contribution); the weighted sum of all the inputs is compared with θ to decide whether or not to activate the neuron. In the FR model, not only the input but also the output is treated as a continuously graded quantity. While MP and LT neuron models describe the integration of synaptic inputs using the Heaviside step function defined by $H(u)$ = 1 for u $\geq$0 and $H(u)$ = 0, the FR model is based on a differential equation for the firing rate with a continuous-time domain.

3.2 Integrate and fire models

The integrate-and-fire (IF) model is the most widely used simple spiking neuron model in artificial neural network algorithms [25, 32–35]. It is a single-compartment model describing the dynamics of membrane potential, with the morphologies of dendrite branches and axons not explicitly included.

The one variable IF models describe the relationship between the time-dependent voltage $V(t)$ and current $I(t)$:

$$\tau_{\mathrm{m}} \frac{\mathrm{d}V}{\mathrm{d}t} = \frac{R_{\mathrm{m}}}{A} I + f(V), \tag{1}$$

where τ_m and A denote the membrane time constant and the effective surface area, respectively, and $f(V)$ describes the leak and spike-generating currents as a function of V. If the voltage $V(t)$ exceeds $\tilde{V}$, which stands for the cutoff voltage V_c or threshold voltage V_T (depending on whether or not the spike-generating part of $f(V)$ exists), the voltage $V(t_+)$ at time t_+ right after spiking becomes equal to the resetting voltage V_r. After the membrane potential crosses $\tilde{V}$, it is reset to V_r and is inactivated for a brief time corresponding to the absolute refractory period t_{ref} of the neuron.

Two-variable IF models include the additional time-dependent adaptive variable u:

$$\begin{aligned} \tau_{\mathrm{m}} \frac{\mathrm{d}V}{\mathrm{d}t} &= \frac{R_{\mathrm{m}}}{A} I + f(V) - u \\ \tau_u \frac{\mathrm{d}u}{\mathrm{d}t} &= a\left(V - V_{\mathrm{L}}\right) - u \end{aligned} \tag{2}$$

with constant a controlling the adaptation to voltage and V_L denoting the leak reversal potential. If the voltage $V(t)$ exceeds $\tilde{V}$, the voltage $V(t_+)$ right after spiking reduces to V_r, similarly to the case of Eqn. 1; in addition, the adaptive variable $u(t_+)$ increases by the amount b controlling the magnitude of the spike event, namely, $u(t_+)$ is set equal to $u(t_-)+b$ with t_- being the time just before spiking.

Table 1 lists the five models, i.e., LIF (leaky integrate-and-fire); QIF (quadratic integrate-and-fire) with and without an adaptive variable; EIF (exponential integrate-and-fire) with and without an adaptive variable.

The information transfer capabilities are assessed in the information-theoretic framework originally suggested by Denève and colleagues [37–41]. Fig. 2 illustrates the framework used in the present study. In brief, the binary hidden state triggers a presynaptic neuron. Then the presynaptic neuron fires a spike train via a Poisson process with the firing rate q_{on} or q_{off}, depending on the hidden state. Synaptic input current I is generated by convolving the spike train with the double exponential kernel: $k\,(t) = exp\,(-t/\tau_1) - exp\,(-t/\tau_0)$ with τ_0 = 0.2 ms and τ_1 = 2 ms, followed by multiplying by the synaptic weight, which is modified to control the average input current $\bar{I}$. In general, mutual information measures the overlapping information of two random variables. The mutual information $H\left(X; I_{0\to t}\right)$ between the hidden state $X \equiv \{x\}$ and the history of postsynaptic spike trains $I_{0\to t}$ is defined as

$$\begin{aligned} H\left(X; I_{0\to t}\right) &= \sum\nolimits_{x, I_{0\to t}} p\left(x; I_{0\to t}\right) \log \frac{p\left(x; I_{0\to t}\right)}{p(x) p\left(I_0 \to t\right)} \\ &= S(X) - S\left(X \mid I_{0\to t}\right), \end{aligned} \tag{3}$$

where $p\left(x; I_{0\to t}\right)$ is the joint probability, and $S(X)$ and $S\left(X \mid I_{0\to t}\right)$ denote the information entropy of X and the conditional entropy of X given $I_{0\to t}$, respectively. They are given by

$$\begin{aligned} S(X) &\equiv -\sum\nolimits_x p(x) \log p(x) \\ &= -\langle x\rangle \log\langle x\rangle - (1 - \langle x\rangle) \log(1 - \langle x\rangle) \end{aligned} \tag{4}$$

$$\begin{aligned} &S\left(X \mid I_{0\to t}\right) = \\ &-\left\langle x \log p\left(x = 1 \mid I_{0\to t}\right)\right\rangle - \left\langle (1 - x) \log\left(1 - p\left(x = 1 \mid I_{0\to t}\right)\right)\right\rangle, \end{aligned} \tag{5}$$

Table 1. Summary of integrate-and-fire neuron models.

Models	$f(V)$	Adaptation	Parameters
LIF	$-(V - V_\mathrm{L})$	No	t_{ref} = 1.966 ms, V_r = −61.72 mV
QIF	$\frac{(V-V_\mathrm{T})^2}{2\Delta_\mathrm{T}} - \frac{R_\mathrm{m} I_\mathrm{T}}{A}$	No	t_{ref} = 2.473 ms, V_r = −57.56 mV, Δ_T = 0.4090 mV
QIF* (Izhikevich)	$\frac{(V-V_\mathrm{T})^2}{2\Delta_\mathrm{T}} - \frac{R_\mathrm{m} I_\mathrm{T}}{A}$	Yes	t_{ref} = 0 ms, V_r = −65 mV, Δ_T= 0.8333 mV, a = 0.2, b = 0, τ_u = 50 ms
EIF	$\Delta_\mathrm{T} \exp\left(\frac{V-V_\mathrm{T}}{\Delta_\mathrm{T}}\right) - (V - V_\mathrm{L}) - \frac{R_\mathrm{m} I_0}{A}$	No	t_{ref} = 10.85 ms, V_r = −58.84 mV, Δ_T = 0.1666 mV
EIF* (AdEx)	$\Delta_\mathrm{T} \exp\left(\frac{V-V_\mathrm{T}}{\Delta_\mathrm{T}}\right) - (V - V_\mathrm{L}) - \frac{R_\mathrm{m} I_0}{A}$	Yes	t_{ref} = 10.85 ms, V_r = −58.84 mV, Δ_T = 0.1666 mV, a = 0, b = 0.1, τ_u = 100 ms

Abbreviations: LIF, leaky integrate-and-fire; QIF, quadratic integrate-and-fire; EIF, exponential integrate-and-fire. Asterisk (*) denotes the model with an adaptive variable.

Parameters: V_L, leak reversal potential; t_{ref}, refractory period; V_r, resetting voltage; Δ_T, spike slope factor; R_m, membrane resistance; *A*, effective surface area; *a*, constant controlling the adaptation to voltage; *b*, constant controlling the adaptation to the spike event. The threshold point (V_T, I_T) satisfying $f(V_T)$ + (R_m/*A*) I_T= 0 and $f\,'$ (V_T) = 0 is identified to be (–57.28 mV, 65 pA), which agrees with that of the biophysical model [29, 36]. The EIF model has an additional fitting parameter I_0 = [Δ_T – (V_T – V_L) + (R_m/*A*) I_T]/(R_m/*A*). Other parameters are given by R_m = 40000 Ω·cm^2, τ_m = 30 ms, and V_L = –70 mV for all models.

where the average, defined to be taken with respect to the probability measure *p*(*x*), may be estimated as the time average.

The conditional probability $p\left(x=1|I_{0\to t}\right)$ is equivalent with posterior log-likelihood of the hidden state, $L(t) = \log_2 \frac{p(x=1|I_{0\to t})}{p(x=0|I_{0\to t})}$, where $p\left(x=1, t \mid I_{0\to t}\right)$ is the conditional probability of on-state $(x = 1)$ at time *t*, given the history $I_{0\to t} \equiv (I_0, I_1, \ldots, I_t)$ of the input current from time 0 to *t*. The log-odds ratio can be estimated via the following differential equation:

$$\frac{dL}{\mathrm{d}t} = r_\mathrm{on}\left(1+e^{-L}\right) - r_\mathrm{off}\left(1+e^{L}\right) + w\delta\left(I_t - 1\right) - \theta, \tag{6}$$

where $w \equiv \log\left(q_\mathrm{on}/q_\mathrm{off}\right)$ and $\theta \equiv q_\mathrm{on} - q_\mathrm{off}$ with the mean postsynaptic firing rates q_{on} and q_{off} for *x* = 1 and 0, respectively. The Dirac delta function δ produces a discontinuous jump when the postsynaptic neuron fires.

The information-theoretic framework is used for comparing the neural dynamics and coding properties of the five IF neuron models (Table 1). The train of hidden state *X* is presented to each of the neuron models to induce the presynaptic input current *I* and the resulting postsynaptic spike train $I_{0\to t}$. The time evolution of the hidden state and the postsynaptic spike is used to calculate the mutual information $H\left(X; I_{0\to t}\right)$ as a measure of information transfer by a neuron model.

Fig. 3 displays the current-rate (*I*-*f*) curves (the left column) and the time evolutions of the hidden state *x* and output spike trains (the right column). The *I*-*f* curve, which expresses the relationship between the applied current to a neuron and the firing rate (i.e., the frequency of output spikes), is used as the basic measure for characterizing neural dynamics. The firing rate *f* of the LIF model is the highest, followed by QIF and EIF. The models possessing adaptive variables (QIF* and EIF*) exhibit reduced

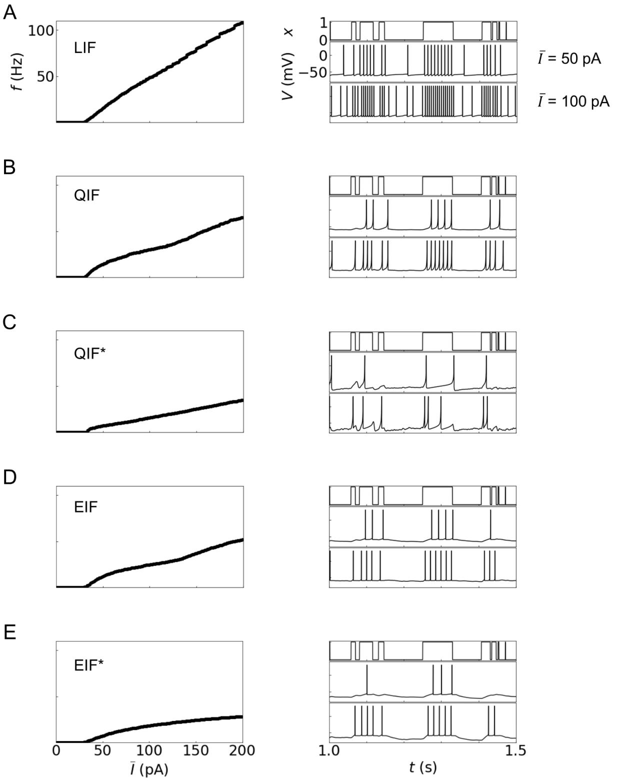


Fig. 3. Dynamics of integrate-and-fire (IF) models: (A) LIF, (B) QIF, (C) QIF* (QIF with an adaptive variable), (D) EIF, (E) EIF* (EIF with an adaptive variable). The left column shows the *I*-*f* curves, where $\bar{I}$ denotes the average input current and *f* the firing rate. The binary hidden state *x* triggers the spike train via a Poisson process with the firing rate q_{on}= 100 and q_{off} = 1 for *x* = 1 and *x* = 0, respectively. In the right column, time evolutions of the hidden state *x* (top) and spike output *V* (mV) of the neuron at $\bar{I}$ = 50 pA (middle) and $\bar{I}$ = 100 pA (bottom) are displayed.

firing rates compared with the corresponding models without adaptive variables. The right column displays the time evolutions of the hidden states (the first row) and resulting output spikes upon $\bar{I}$ = 50 pA (the second row) and 100 pA (the third row). In all models, the timing of spikes is generally well-matched with the hidden state "1"; however, the spike events do not reflect the fast transitions of hidden states between "0" and "1".

The dynamics and information processing of IF models are mapped to Boolean operations in Fig. 4. At a given threshold for firing rate f or that for mutual information H, if f or H is greater than or equal to the threshold, then the neuron is active ("1" or "true"); if it is under the threshold, the neuron is considered as inactive ("0" or "false"). Both OR and AND operations occur as a function of the fold change of the difference between the threshold voltage V_T and the resetting voltage V_r with respect to the default value $V_d/V_d{}^{(0)}$, where V$_d$ = V_T – V_r and $V_d{}^{(0)}$ = $V_T{}^{(0)}$ – $V_r{}^{(0)}$ (with superscript "$^{(0)}$" denoting the default parameters). V_d may indicate the voltage required for generating subsequent action potentials: a smaller value of V$_d$corresponds to the increased membrane excitability (i.e., greater tendency to fire). Several biological contexts, giving rise to the decrease of V_d, include (1) depolarization of the resting membrane potential, (2) reduction in GABAergic inhibition, (3) increased neuronal responsiveness to subthreshold input, and (4) increased conductance that dictates the rate of action potential firing [42]. The OR operations (red shaded regions) arise when both weak (e.g., $\bar{I}$ = 50 pA) and strong ($\bar{I}$ = 100 pA) inputs activate the neuron; on the other hand, AND operations (blue shaded regions) occur only when strong presynaptic input (e.g., $\bar{I}$ = 100 pA) can activate the neuron. These Boolean operations correspond to the schematic illustrations in Fig. 1B: the input current $\bar{I}$ = 50 pA may denote the active input '1', and $\bar{I}$ = 100 pA thus corresponds to two active inputs. Depending on the thresholds for f and H, the regions can be mapped to OR or AND operations.

Although these IF models are simplified versions of the biophysically realistic multi-compartment neuron models (which are explored in the following section), they appear to characterize the neural logic operations successfully. In particular, the neural dynamics and coding properties of exponential integrate-and-fire models (EIF and EIF*) are overall similar to the biophysical models, compared with other IF models [29]. The neural dynamics of IF models vary, depending on the mathematical form of the leak and spike-generating currents. In brief, the leak current term is necessary for responding to the changes of the hidden states in a timely manner; the IF models without the term usually fire in response to inactive hidden states ('0') because depolarization of the membrane potential during previous active hidden states ('1') is maintained during inactive states. The spike-generating currents [Eqn. 1 and Table 1] determine the speed of spiking: the IF models except EIF and EIF* exhibit much slower spike generation, compared with the biophysical model [29].

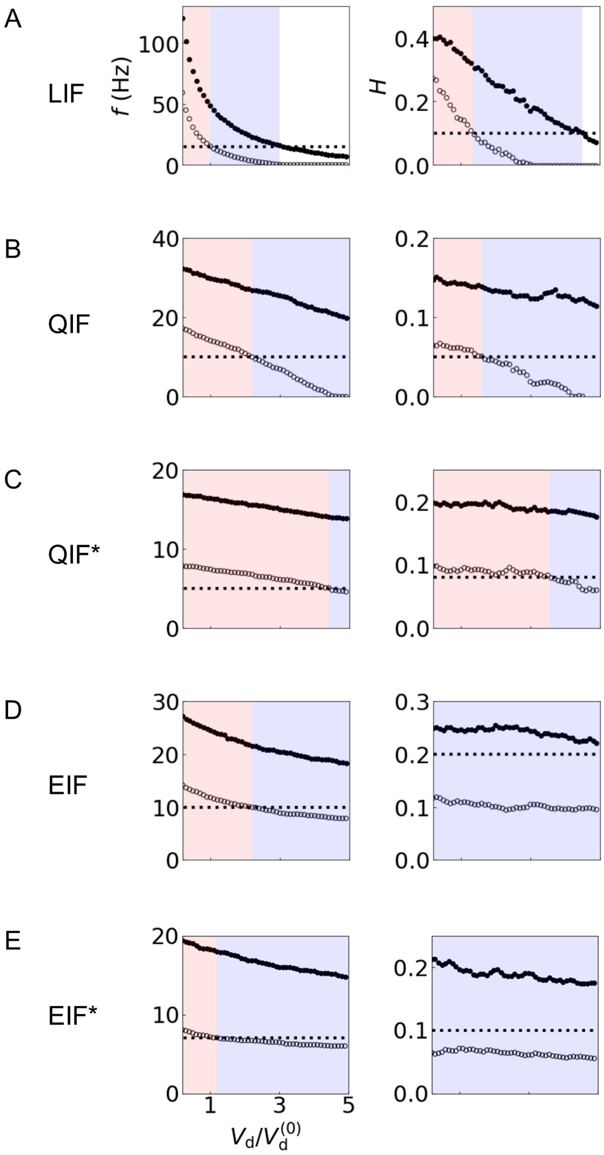


Fig. 4. Mapping dynamics and information processing of IF neurons to Boolean logic operation. (A) LIF, (B) QIF, (C) QIF* (QIF with an adaptive variable), (D) EIF, (E) EIF* (EIF with an adaptive variable). The left and right columns display the firing rate f and mutual information H between the hidden state and output spike train, respectively. The results for $\bar{I}$ = 100 pA and $\bar{I}$ = 50 pA are marked with filled and unfilled circles, respectively. The horizontal axis of each panel represents the fold change of the difference between the threshold voltage V_T and the resetting voltage V_r (i.e., V$_d$ = V_T – V_r) with $V_d{}^{(0)}$ = $V_T{}^{(0)}$ – $V_r^{(0)}$ denoting the default value. Displayed are f and H versus $V_d/V_d{}^{(0)}$ ranging from 0.2 to 5. Red and blue shaded colors indicate OR and AND operations, respectively, at given thresholds for f and H (dotted lines).

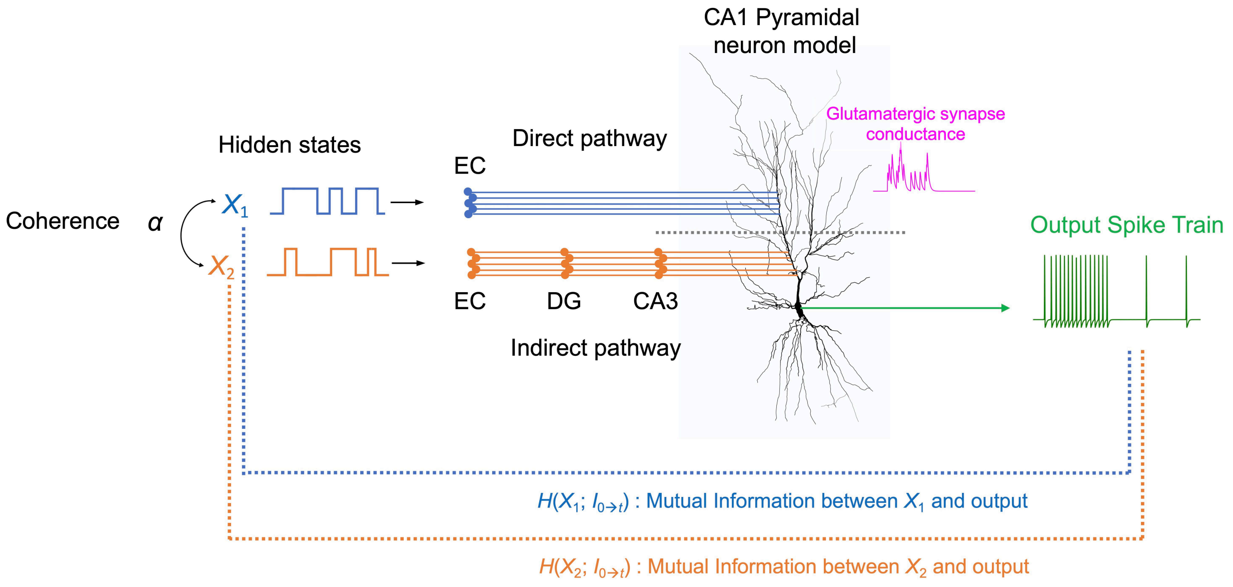


Fig. 5. Schematic of the information-theoretic framework for multimodal synaptic transmissions in the biophysical model of the CA1 pyramidal neuron. Each of the two hidden states X_1 and X_2 with coherence α triggers a set of presynaptic neurons which form glutamatergic synapses with the CA1 pyramidal neuron. The apical dendrite of the CA1 neuron is divided (black dotted line) into sections distal and proximal to the soma, which receive inputs from the direct pathway (X_1) and the indirect pathway (X_2), respectively. The mutual information $H(X_i; I_{0\to t})$ is measured between X_i (for i = 1, 2) and the postsynaptic spike train emerging in the soma.

4. Synaptic logic gate

4.1 Biophysical model

This section describes the characterization of information processing of the biophysically realistic multicompartment neuron models, which describe how action potentials are initiated and propagated, based on Hodgkin-Huxley (HH) type conductance models for ion channels [43, 44]. Containing the axon and dendrites explicitly, the model has highly realistic structures via three-dimensional morphological reconstruction of biological neurons [45].

Two representative neuron models for neural computation are compared: the pyramidal neuron in CA1 in the hippocampal circuit (ModelDB accession 7907) [46] and PF in the cerebellum (ModelDB accession 7907) [46]. In the CA1 pyramidal neuron model, all dendrites are divided into compartments with a maximum length of 7 mm. Spines are incorporated where appropriate by scaling membrane capacitance and conductance [47, 48]. Two Hodgkin–Huxley-type conductances (g$_{Na}$ and g$_K$) are inserted into the soma and dendrites at uniform densities. The model is tuned by attaching a synthetic axon. The uniform passive parameters of the model are given by R_i = 150 Ω·cm, C_m = 1 mF/cm^2, and R_m = 12 kΩ·cm^2. The standard values for g_{Na} and g_K are 35 and 30 pS/mm^2, respectively. For the Purkinje cell, we use the morphology of a 21-day-old Wistar rat PF [49]. The model consists of an axon, a soma, smooth dendrites, and spiny dendrites. The model has passive parameters as follows: R_m = 12 kΩ·cm^2, R_i = 150 Ω·cm, and C_m = 1 μF/cm^2. To compensate for the absence of spines in the reconstructed morphology, we scale the conductance of passive current and C_m by a factor of 5.34 in the spiny dendrite and 1.2 in the smooth dendrite. Two Hodgkin–Huxley-type conductances (g$_{Na}$ and g$_K$) are inserted into the soma and dendrites at uniform densities. The model is tuned by attaching a synthetic axon. The standard values for g_{Na} and g_K are 35 and 30 pS/mm^2, respectively.

The information-theoretic framework is similar to that used in Section 3 for IF models, except that the stimulus sites are carefully chosen based on biological knowledge and two inputs are provided simultaneously, with their competitive and cooperative effects assessed [29, 50, 51]. Fig. 5 displays the schematic of the framework. Each of the two hidden states X_1 and X_2 triggers a set of presynaptic neurons connected to the postsynaptic neuron via synapses. The synapses from each set (corresponding to X_1 or X_2) are colored blue or red, respectively. The coherence between X_1 and X_2 is measured by parameter α in the range [0, 1]: α vanishes for the two states behaving independently while it is equal to unity for the two fully synchronized. The mutual information $H\left(X_i; I_{0\to t}\right)$ between each hidden state X_i and the output spike train is measured as in Eqn. 3 (with X_i replacing X).

In the rest of this section, we explore the information processing of homosynaptic plasticity (Section 4.1) and analyze Boolean logic operations triggered by one hidden state. Then we examine heterosynaptic transmission in the two hidden state scheme, which allows us to assess the synaptic cooperation and competition (Section 4.2).

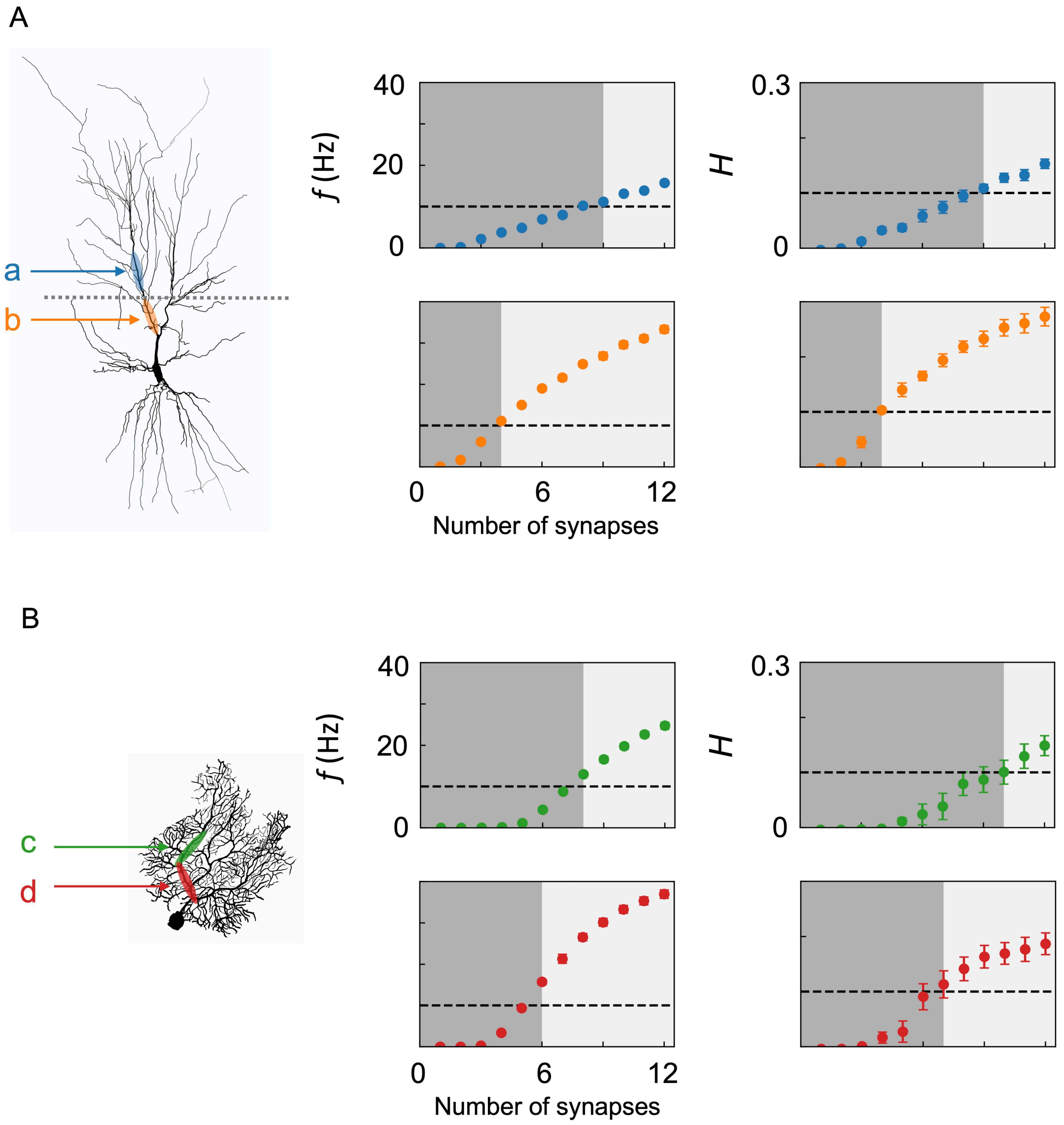


Fig. 6. Boolean logic mapping dynamics and information processing of neurons: (A) CA1 pyramidal neuron and (B) Purkinje fiber. The morphologies of the neurons are illustrated on the left panels. The neurons are given synaptic inputs stimulated by a hidden state. For the CA1 pyramidal neuron, regions a and b denote locations of synaptic inputs placed relatively distally and proximally from the soma, respectively. For the Purkinje cell, locations of c and d correspond to relatively distal and proximal positions, respectively. The panels in the middle column show the postsynaptic firing frequency f as a function of the number of synapses in each condition. Those on the right show the mutual information H between the hidden state and the postsynaptic spike train. The outputs are mapped to Boolean operations using threshold values of 10 Hz and 0.1, indicated by the dashed line. When f or H is below and above the threshold, the output is zero (dark gray region) and unity (light gray region), respectively. Error bars represent the standard errors of five independent simulations with different random seeds for hidden state generation.

4.2 Homosynaptic plasticity

Homosynaptic transmission refers to the specific modification of a synapse by the activity of the corresponding presynaptic and postsynaptic neurons. The most widely used realization of this concept, first proposed by Hebb in 1949 [52], is the spike-timing-dependent plasticity (STDP) rule [53], which is often adopted in spike-based artificial neural networks. This rule states that a presynaptic stimulus immediately followed by a postsynaptic spike results in potentiation of the synapse while the opposite results in depression. Another well-known synaptic plasticity rule is the Bienenstock, Cooper, and Munro (BCM) rule [54, 55], according to which modification of the synapse depends on the instantaneous postsynaptic firing rate. In the original formulation of the rule, depression occurs when the postsynaptic firing rate is below a threshold while potentiation occurs when the rate is above the threshold. In particular, the threshold separating depression and potentiation is itself a slow variable that changes as a function of the postsynaptic activity.

Here, we implement a learning rule in which the STDP rule is combined with the BCM rule [50, 56–58]. According to the STDP rule, weight changes occur for each pair of presynaptic and postsynaptic spikes separated by time interval Δt in the following way:

$$\begin{aligned}\Delta w_{\mathrm{p}}(\Delta t) &= A_{\mathrm{p}}(t)\exp\left(-\Delta t/\tau_{\mathrm{p}}\right) \text{ for } \Delta t > 0\\ \Delta w_{\mathrm{d}}(\Delta t) &= A_{\mathrm{d}}(t)\exp\left(\Delta t/\tau_{\mathrm{d}}\right) \text{ for } \Delta t < 0,\end{aligned} \tag{7}$$

where the subscripts p and d label potentiation and depression, respectively, and $\Delta t \equiv\ t_{post} - t_{pre}$ is the time difference between the presynaptic and postsynaptic spikes. The sign of Δt determines whether the presynaptic spike precedes the postsynaptic spike (positive) or follows (negative). The BCM rule is implemented by allowing the amplitudes A_p and A_d to vary with the postsynaptic firing rate $\langle c\rangle$ according to

$$\begin{aligned}A_{\mathrm{p}}(t) &= \langle c\rangle^{-1}A_{\mathrm{p}}(0)\\ A_{\mathrm{d}}(t) &= \langle c\rangle A_{\mathrm{d}}(0),\end{aligned} \tag{8}$$

where $\langle c\rangle$ is a weighted sum of postsynaptic spikes, given by

$$\langle c\rangle = \frac{\alpha_c}{\tau_c}\int_{-\infty}^{t} dt' c\left(t'\right)\exp\left(-\frac{t-t'}{\tau_c}\right). \tag{9}$$

The BCM rule has a balancing effect that allows robust synaptic learning [55, 56]. The parameters for the learning rule are as follows: τ_p = 20 ms; τ_d = 70 ms; $A_p(0)$ = 0.006 μS; $A_d(0)$ = 0.002 μS; τ_c = 1500 ms; α_c = 62.5.

Boolean logic mappings of the dynamics and information processing of the biophysical neurons are illustrated in Fig. 6. Note that the Boolean operations depend on the location of the synaptic input. For the CA1 pyramidal neuron, synapses are placed in the distal section or proximal section of the apical dendrite (a or b in Fig. 6A) in one setting or the other. These locations are selected based on neuroanatomical knowledge: The apical dendrite of the CA1 pyramidal has a long, extended structure to receive inputs from distinctly organized regions. Its distal region directly receives input via the perforant pathway from the entorhinal cortex while the proximal region indirectly receives via the granule cell and CA3 pyramidal neuron [59] (Fig. 5).

Presynaptic neurons fire at an average rate of 100 Hz, provided that the hidden state is on. When the stimulus is given to section a, nine synaptic inputs are required to cross the firing threshold of 10 Hz and the mutual information threshold of 0.1. In the case of the stimulus given to section b, the required number of synapses for the same threshold is just four. This manifests that the Boolean logic operations occurring with synaptic transmission depend on the location on the dendrite: in the distal region, the operation is closer to AND, with many concurrent inputs required to exceed the threshold, while in the proximal region, it is closer to OR, with only a few required.

The Purkinje cell has a highly branching, flattened structure built to receive inputs from up to 200,000 parallel fibers that pass orthogonally through the dendritic arbor [60]. In addition, each Purkinje cell receives input from one climbing fiber, which enwraps the dendrite and forms a vast number of synapses [61]. Unlike pyramidal neurons, there is no need for a vertical extension. Fig. 6B shows the logic operations performed by the Purkinje cell. Two inputs are given to two sections c and d along a main dendritic branch. The presynaptic neurons are assigned to a firing rate of 200 Hz when the hidden state is on. For the relatively distal section c, the firing rate threshold is crossed at eight synapses and the mutual information threshold at ten; for the relatively proximal section d the thresholds are exceeded at six and seven. Namely, there is a less drastic difference in the two sections compared with the CA1 case.

4.3 Heterosynaptic plasticity

The CA1 pyramidal neuron and the Purkinje cell form well-established functional neural circuits that integrate multiple inputs from distinct sources. In particular, they can perform complex computation via homo and heterosynaptic mechanisms [50]. Heterosynaptic transmission refers to the modification of synaptic strength by unspecific presynaptic stimuli. The activity of a presynaptic neuron alters the strength of a synapse of the postsynaptic neuron not directly connected to the presynaptic neuron in action [62]. Unlike the homosynaptic case, there is no widely accepted computational model for heterosynaptic transmission.

The CA1 pyramidal neuron receives inputs mainly from two sources, one directly from the entorhinal cortex and one from the CA3 region [63]. The processing of these inputs is a critical step in the role of the hippocampus in memory, which is postulated to play a comparator role [64]. Synaptic plasticity in the CA1 neuron is well studied and exhibits rich heterosynaptic plasticity mechanisms [65, 66]. The Purkinje cell provides the sole output from the cerebellar cortex. It receives inputs from parallel fibers, axons of granule cells, and just one climbing fiber originating from the inferior olive, which nevertheless comprises about 1500 synapses. The Purkinje cell is believed to play a key role in motor learning, yet the synaptic mechanism remains elusive [67]. It was suggested in the early theories of learning in the cerebellum that heterosynaptic interactions between the parallel and climbing fibers play a key role [68, 69]. These neuronal systems have in common that heterosynaptic interactions between multiple inputs, which can be cooperative or competitive, play a crucial role in their computations [70, 71]. To understand the processing of these multiple inputs, we study them in our information-theoretic framework (Fig. 5).

Fig. 7 displays the Boolean logical operations of the neuron models at given two multimodal inputs. For

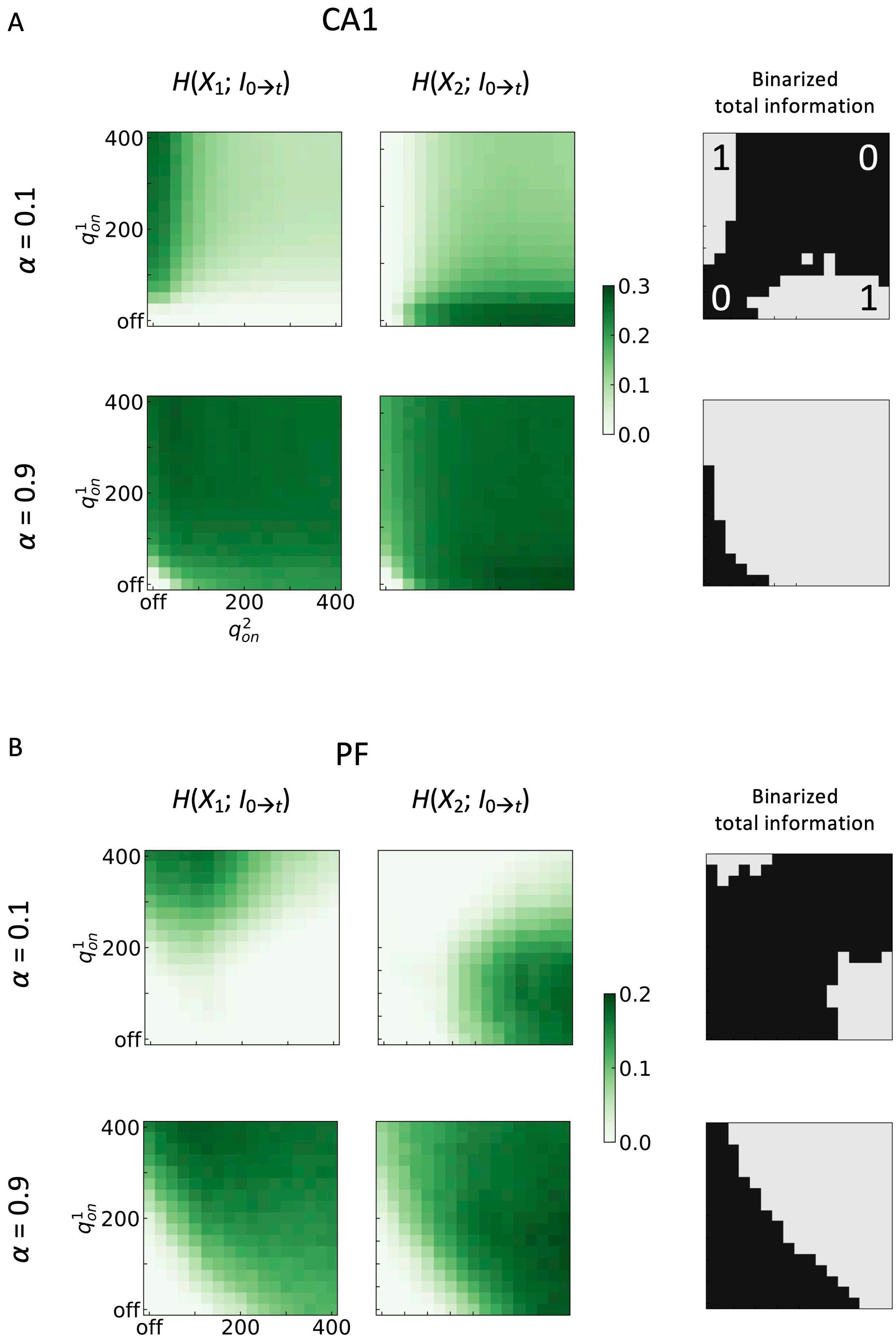


Fig. 7. Information processing and Boolean logic operations of multimodal synaptic transmissions in the (A) CA1 pyramidal neuron and (B) Purkinje fiber (PF). The mutual information maps for $H(X_1; I_{0\to t})$ and $H(X_2; I_{0\to t})$ (first and second columns) are drawn on the plane of firing rates $q^1{}_{on}$ and $q^2{}_{on}$ of the presynaptic neurons triggered by X_1 and X_2. The total mutual information, binarized with a threshold (80% of maximum), is shown in the third column. The first and second rows indicate results for coherence α = 0.1 and 0.9, respectively.

the CA1 neuron, X_1 stimulates twelve synapses in section a and X_2 stimulates eight synapses in b (see Fig. 6A); for the PF, X_1 stimulates five synapses in c and X_2 stimulates five synapses in d (see Fig. 6B). Varying the firing rates of the two sets of presynaptic neurons, $q^1{}_{on}$ and $q^2{}_{on}$, we compute the mutual information $H\left(X_1; I_{0\to t}\right)$ and $H\left(X_2; I_{0\to t}\right)$ and show the resulting maps in the first and second columns, respectively. The third column presents the total information transmitted, $H\left(X_1; I_{0\to t}\right) + H\left(X_2; I_{0\to t}\right)$, with the threshold set to 80% of the maximum. For both CA1 and PF, when the coherence is low (α = 0.1; first row), synaptic competition occurs. In case that just one input is on, the total mutual information exceeds the threshold and the output becomes unity; in the case of both on, the output is zero. It is remarkable that the resulting Boolean operation is the exclusive OR (XOR) operation. On the other hand, when the coherence is high (α = 0.9; first row), the two inputs exhibit cooperation, which results in OR and AND-like operations.

5. Approaches to designing logic backbones of neural circuits

So far, we have discussed the characteristics and corresponding design principles for single neuron models and operators derived from small networks of neurons. In this section, we discuss the strategies to implement these models to a system of an even larger scale. Neuron models and logic gates play a crucial role in the dynamics of neural circuits. Algorithms inspired from neural circuits, such as artificial neural networks and variations, are also relevant despite its high-level representation of brain circuits, as the individual elements are technically an abstracted version of neuron models. As the scale and complexity increase, designing a performant and efficient circuit becomes more and more demanding. At its core, these systems can be interpreted as graphs with different types of nodes and edges. Then, the problem becomes to find the optimal topology best suited for a specific purpose. Often one might believe that finding the optimal topology is either unnecessary or inconsequential due to effectively no limitations available in the software representations of these systems. But this is not true for two reasons: First, the optimal topology is crucial for hardware design. Second, our brain does have limited resources, spaces, and connections with much more advanced motifs than those a typical learning problem might attempt to implement in software.

Optimizing the topological aspect of the model is a difficult problem and has garnered a lot of interest in various fields for quite some time. Here, we review two distinct fields that have made progress in applying computational algorithms to the physical systems to reconstruct or find the optimal topology. The two are the fields of systems biology and chip design: Systems biology, at its core, studies biochemical reaction networks inside a cell comprised of various enzymes and ligands. A computer chip, on the other hand, is a complex amalgamation of modules, subsystems, and logic systems. Once visualized, both can be represented using graphs and thus share similarities with neural networks and circuits. Both systems can be modeled and simulated as the basic dynamics of each type of building block are known. Due to the scale of the model and the diversity of elements and possible interactions between them, building a biochemical reaction network from scratch is difficult. Similarly, engineers have utilized various toolsets to aid the design process due to their complexity.

Current attempts at automated network topology design rely on one of several different techniques. Most prevalent is the inference techniques [72], with Bayesian inference being the most common of the bunch [73, 74]. Machine learning has become another popular choice, with examples such as optimization of biochemical reaction networks through machine learning [75], deep learning for regulatory networks [76], chip design using reinforcement learning [77], and sparse network identification using Bayesian learning [78]. There are also various information-theoretic approaches [79] using an ensemble of logic models [80], regression approaches [81], other heuristic, meta-heuristic, and hybrid methods [82–85], and many more.

Another approach, relatively less well-adopted, is the evolutionary algorithm (EA); it is a type of optimization algorithm that is heuristic and population-based. At its core, EA is characterized by concepts inspired by nature, with processes defined in correspondence to selection, reproduction, mutation, and recombination. An adaptation of EA may proceed as follows: (1) A population of individuals is initialized. (2) Fitness of individuals is evaluated. (3) Individuals are selected for reproduction, based on fitness. (4) Offsprings are generated with some probability of mutation and crossover. (5) The next generation is established with the number of the least fitted individuals reduced. (6) Repeat the process until the termination criteria are met. Many variations of the algorithm exist, most notably differential evolution and the particle swarm algorithm. While typically the algorithm is used for numerical optimization problems, it can also apply to inference and topology searching problems. We believe EA may provide several benefits over other approaches for automated designs of neural networks and neural circuits. The algorithm has garnered a lot of interest over the years on automated and optimal designs of artificial neural networks for learning tasks [86–88]. We suggest EA-based algorithms to be a good alternative for topological search and output optimization of neural circuit designs. Here, we give a short demonstration of finding an optimal set of functions to recreate a target output signal from the given input signals, examining the feasibility of circuit design automation via an EA-like algorithm (see Fig. 8). Technically, this version of the problem is not looking for different topologies per se but instead looking for the optimal set of transfer functions under a given topological

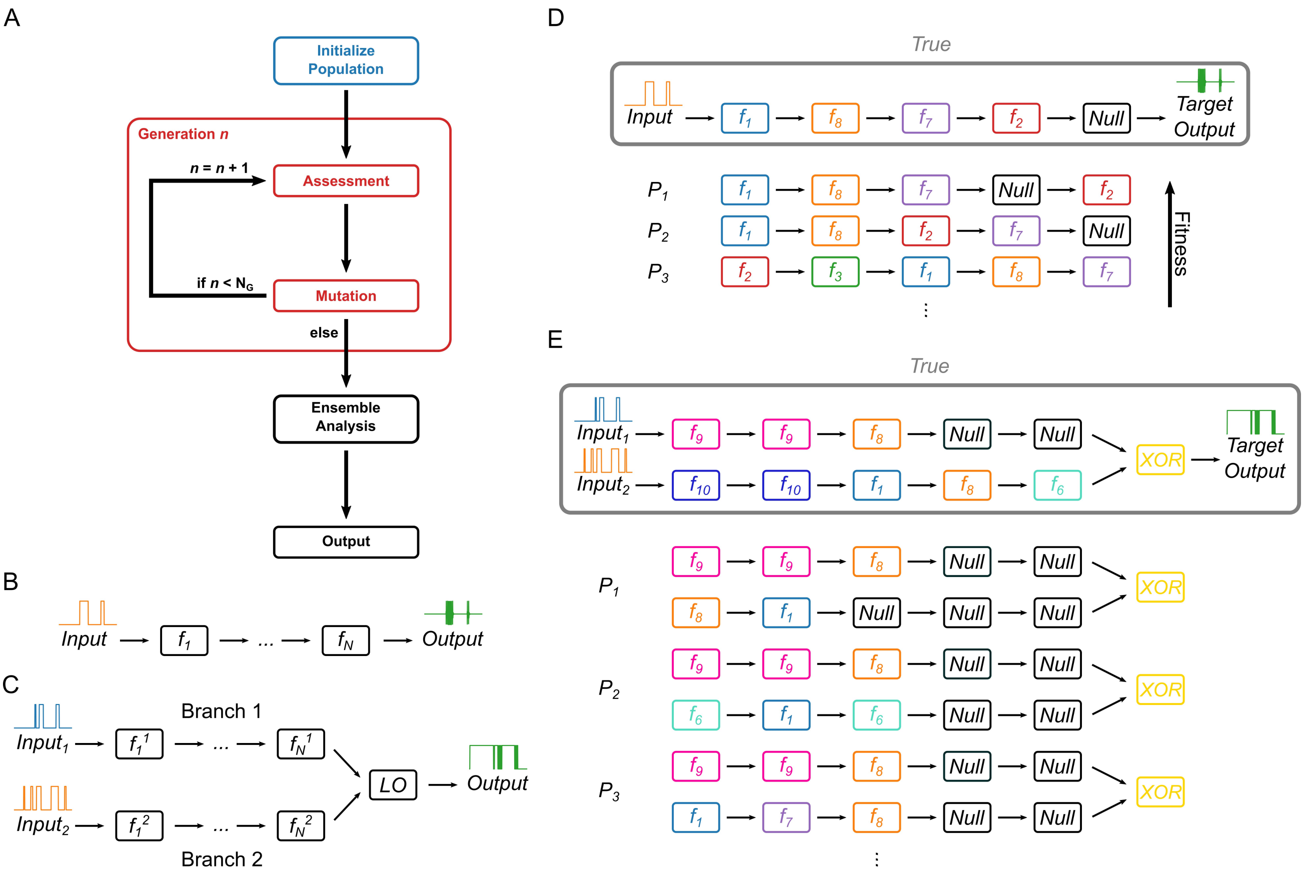


Fig. 8. The demonstration of a framework for designing logic backbones. (A) The overview of evolutionary algorithm. (B) Scenario 1, where a sequential single-chain layer of functions akin to neural networks is used. (C) Scenario 2, where two sequences of functions (branch 1 and branch 2) converge with a logical operator simulating a multimodal neural circuit are used. (D) Result of scenario 1, where the output of the true sequence is given to the algorithm as the target. The top three models with the highest fitness score are shown (P_1, P_2, P_3). The algorithm recovers the original sequence. 'Null' indicates an empty placeholder where the algorithm decides not to populate with a function. (E) Result of scenario 2, where the output of the true sequence is given to the algorithm as the target. Three of the 23 models that fully reproduce the true output are shown. The algorithm recovers the original sequence for branch 1 and LO but not for branch 2. 'Null' indicates an empty placeholder where the algorithm decides not to populate with a function.

constraint, which is imposed to make sure our example is analogous to neural circuits or neural networks. The workflow can easily incorporate topology modification with the adoption of an adjacency matrix-like representation of the topology. In this example, various predefined functions and logic operators are available for our algorithm to look for a model under some topological constraint that can mimic the output as much as possible. While we have used a set of elementary functions as the building blocks for demonstrative purposes, in more complex applications, they can always be replaced by more complicated classifiers, operators, layers, or even neuron models with detailed physiology specified. The contents of the building blocks can be of various scopes and levels of detail as long as an appropriate fitness function is chosen to reflect the changes.

For this study, we have created two different scenarios: one with a sequential single-chain layer of functions conceptually akin to typical neural networks (Fig. 8B) and the other with two sequences of functions converging with a logical operator simulating a multimodal neural circuit (Fig. 8C). In each case, we generate a synthetic model from which the target output is collected. For the neural network example, we use the population size $N_P = 200$ and the number of generations $N_G = 50$. For the neural circuit example, the population size and the number of generations are given by $N_P = 500$ and $N_G = 1000$, respectively. A total of ten different functions are available (spike generation, convolution, high and lowpass filters, differentiation and integration, Fourier and inverse Fourier transforms, forward and backward shifting), together with three additional logical operators (AND, OR, XOR) for the neural circuit example. The length of a function sequence has a maximum but no minimum. The algorithm could decide not to populate the backbone with a function that is represented as 'Null' in Fig. 8. We use the EA-inspired algorithm illustrated in Fig. 8A with only the input and the target output given. Our model is represented by a vector of integers where each integer corresponds to a specific building block. The fitness is determined by the sum of the residuals. The top half of the fittest population is selected, and offsprings are generated with a point mutation in such a way that a single function is randomly chosen to be replaced by another while the other half is discarded.

For scenario 1, we find that the algorithm recovered the original model nicely and collected similar models with comparable outputs (Fig. 8D). The multimodal neural circuit example is of particular interest since we believe neural circuit hardware design will benefit the most from the suggested approach. In the second scenario, multiple different models (N = 23) reproduce the given output. After analyzing the models, we notice that the differences are uniquely present in branch 2 while the algorithm recovers the correct logic operator and the sequence of functions for branch 1. We believe this is due to the binary nature of logical operators, where part of the information gets discarded. For a topology like Fig. 8C, where a logic operator integrates outputs from multiple sources, the content of the upstream elements might be tuned and simplified while keeping the desired dynamics, reducing the cost, therefore increasing the efficiency and potentially the scalability of the hardware implementation. Instead, the fitness function may be tweaked for parsimony, e.g., penalize a larger, more complex model, to achieve a similar goal.

The adjustment of edges (i.e., connections) between different nodes is crucial for the circuit design. Elements such as skip connections are known to have a profound impact on performance. A proper encoding strategy is necessary to achieve this goal. A directed graph representation is the most conceptually straightforward, with binary values indicating the connectivity between two nodes. For compartmental models, where the spatial aspect of synaptic plasticity may be studied, a continuous variable defining the position of synapses may be used instead. However, as many EA-based applications to neural network optimization for learning problems have demonstrated, a much more compact encoding is possible and recommended, as the initialization, mutation, and crossover can be performed much more efficiently. Support for variable-length encoding and artificial physical constraint are few other advantages. Another important factor in determining the performance of the algorithm is the evolving strategy. Crossover, in particular, is difficult to conceptualize for topology search. There have been algorithms such as NEAT [89] to address this problem. On top of mutation and crossover, artificially increasing the evolutionary pressure through the implementation of extinction/migration of individuals may be helpful for topological search, as the search space is discrete. On the same note, the application of generalized island models [90] may provide another interesting perspective to the problems with population diversity.

From the model engineering perspective, an EA-based algorithm like this is beneficial for two reasons. First, the algorithm, albeit optimized for the topology, is meta-heuristic and therefore flexible enough to incorporate mechanistic models. Supporting mechanistic models indicates that detailed biophysics can be implemented in the bottom-up approach where both the models and the results are comprehensible for further analysis. This aspect of the algorithm is particularly valuable for the hardware design, where the implementation and debugging as much more straightforward. If an abstract byproduct of the model is used for the fitness score, an algorithm like this can provide a good balance between bottom-up and top-down strategies. Second, population-based optimization raises the possibility of collecting model ensembles. With a large population running for a long time, the algorithm can collect various models utilizing different strategies to achieve the same goal. Further, we can analyze an ensemble of different but equally good models to gain insight into the system. Additional constraints can be applied to obtain a reduced ensemble or a single model suitable for specific use cases. Another benefit of population-based algorithms is scalability, where massive parallelization is possible based on individual genealogy.

6. Discussion and conclusions

This study has investigated multiscale mechanisms of neural computation via computer simulations and information-theoretic analysis. We have first reviewed the operation mechanisms of three representative simple mathematical neuron models (i.e., MP, LT, and FR models) to introduce the basic concept of neural logic operations that might explain simple Boolean operations such as AND and OR gates. Next, IF models have been analyzed and the neural computations interpreted from the viewpoints of neural dynamics and neural coding approaches. These two complementary approaches allow a more comprehensive understanding of neural computation at the single-cell level. The analysis has extended to the biophysically realistic multi-compartment neuron models, which is adequate for evaluating versatile information processing through homo and heterosynaptic transmissions. We have then compared two representative multimodal neurons (i.e., the pyramidal neuron in CA1 in the hippocampal circuit and PF in the cerebellum), and finally, investigated the logic operations at the neural circuit level.

The simple neuron models (i.e., MP, LT, and FR models) are indeed beneficial for understanding the basic concepts of neural computations at the single-cell level. They successfully reproduce the basic single neuron behavior: neurons may have their own intrinsic thresholds for determining whether to fire or not, and this can be described with simple mathematical treatment. By introducing the Heaviside step function (for MP and LT models) or a differential equation (for FR model), these simple models can perform the basic Boolean logic operations such as AND and OR gates, in which the binary values 1 and 0 correspond to “true” and “false” (Fig. 1). Next, five IF models have been subjected to extensive simulations combined with the information-theoretic framework (Fig. 2). It has been manifested that both neural dynamics and neural coding approaches support the computational capability of neurons (Figs. 3 and 4). We have then investigated the role of synaptic transmission in neural computation through biologically realistic multi-compartment neuron models, and analyzed two representative computational entities, CA1 pyramidal neuron in the hippocampus and PF in the cerebellum in the information-theoretic framework. For single-input modalities, synapses proximal to the soma have turned out to act as OR gates whereas those distal to be closer to AND. This is particularly relevant in the CA1 pyramidal neuron, whose extended apical dendrites reach fibers from different sources. We have further assessed heterosynaptic competition and cooperation of the neurons at given multimodal inputs. Both AND/OR-like operations have been observed in the CA1 and PF for inputs with high coherence. On the other hand, when the coherence is low, both neurons exhibit the linearly non-separable XOR operation. This hints that complex computation can occur in single neurons, which may not be properly described by the simple neuron models.

For more complex circuits, algorithms that can design the optimal models for given requirements and constraints would be highly beneficial. For systems like neural circuits and neural networks, the optimization is performed in only a few orthogonal search spaces, e.g., parameter, dynamics, and topology. Network topology optimization is often overlooked, although it can have a profound impact on how the circuit performs. Note that in general different fields and subjects have different goals and limitations requiring different strategies. Systems biology, for example, is often bottlenecked by experimental limitations. Thus, constructing and validating against sparse data often presents a big challenge for systems biologists. We have demonstrated an example of an automated network design algorithm based on EA with two distinct design scenarios, and shown that despite the same algorithm and building blocks in both cases, the specificity of the ensemble differs vastly. While the linear chain example simply recovers the original model, the branched example demonstrates that multiple versions may satisfy our criteria in reconstructing the output from the given input. The population-based nature of EA can create an ensemble of equally good choices, from which the best is chosen based on the overall priority of the design principle.

The information-theoretic analysis used in this study is based on the method originally proposed by Denève and colleagues [37–41], which measures the mutual information between a hidden state triggering presynaptic inputs and the postsynaptic output spike train. This framework provides an ideal means to measure the information processing of a single neuron. Extending the method, we have included two hidden states to characterize the computation performed by a neuron receiving inputs from two information sources [29, 50, 51]. Since the seminal work of MacKay and McCulloch in 1952 [91] that first quantified the information contained in a spike train, numerous measures based on the classical information theory [21] have been devised to quantify information processing in single neurons and between neurons through synaptic transmission. Mutual information is a fundamental and versatile measure for the overlapping information between two quantities (e.g., presynaptic input and postsynaptic output) [17]. In our extended framework, two mutual information values are calculated for each input modality, allowing us to assess synaptic competition and cooperation.

The multiscale approach to neural computations presented in this study may provide a starting point for the design of biologically plausible neuron and synapse models in AI technologies. While most existing neuron models are designed as simple integrators of unimodal synaptic inputs based on the “dumb” neuron concept in the 1940s and 50s, recent experiments have hinted towards developing “smart” neuron models with potential applications in artificial neural network algorithms. In particular, linearly non-separable functions such as the XOR operation were traditionally thought to require multiple neuron layers and summing junctions [92]. A recent experimental study has shown that damping behaviors of the dendritic action potential in the neocortical layer 2/3 pyramidal neuron can perform XOR operations [15]. This result supports theoretical work that argued for complex computations at the level of single neurons [93, 94]. Moreover, there is growing evidence that such nonlinear functions at the single neuron level may provide an essential computational resource in neural networks [95–98]. Large-scale deep learning algorithms have begun to explore complex operations at the single neuron level, such as the mirror neuron (for MirrorBot) [99], and multimodal neurons in the CLIP (Contrastive Language-Image Pre-training) algorithm [100].

In conclusion, this study describes simulations together with information-theoretic treatment of the multiscale logic operations in the brain. Both neural dynamics and information processing in biophysically realistic neuron models and phenomenological IF type models have been successfully mapped to the Boolean logic computation. Remarkably, neuronal information maps not only to basic AND/OR functions but also to the linearly non-separable XOR function, depending on the neuron type. Computational analysis on the multiscale nature of neural computation may be beneficial for understanding the computational principles of the brain and lay the foundation for developing brain-inspired advanced computational models.

7. Author contributions

Conceptualization, KH and MYC; modeling and simulations, JHW, KC, and SHK; analysis, SHK, KC, JHW, KH, and MYC; writing—original draft preparation, KC, SHK, JHW, and KH; writing—review and editing, MYC and KH; supervision, MYC and KH. All authors have read and agreed to the published version of the manuscript.

8. Ethics approval and consent to participate

Not applicable.

9. Acknowledgment

Not applicable.

10. Funding

This research was funded by Korea Institute of Science and Technology (KIST) Institutional Program (Project No. 2E30951, 2Z06588, and 2K02430) and National R&D Program through the National Research Foundation of Korea (NRF) funded by Ministry of Science and ICT (2021M3F3A2A01037808). KC was supported by the KIAS Individual Grants (Grant No. CG077001). MYC acknowledges the support from the NRF through the Basic Science Research Program (Grant No. 2019R1F1A1046285).

11. Conflict of interest

The authors declare no conflict of interest.

12. References

[1] Bugmann G. Biologically plausible neural computation. Biosystems. 1997; 40: 11–19.

[2] Hinton GE. Computation by neural networks. Nature Neuroscience. 2000; 3: 1170.

[3] Zador AM. The basic unit of computation. Nature Neuroscience. 2000; 3: 1167.

[4] Piccinini G, Scarantino A. Information processing, computation, and cognition. Journal of Biological Physics. 2011; 37: 1–38.

[5] Horst S. The computational theory of mind. In: Stanford Encyclopedia of Philosophy. Stanford, CA: Stanford University. 2005.

[6] Craik KJW. The nature of explanation. London: Cambridge University Press. 1952.

[7] Putnam H. Brains and behavior. In: originally read as part of the program of the American Association for the Advancement of Science, Section L (History and Philosophy of Science). 1961. Printed in Black N, editor. Readings in the philosophy of psychology. Cambridge, MA: Harvard University Press. 1980.

[8] McCulloch WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics. 1943; 5: 115–133.

[9] Ermentrout GB, Galán RF, Urban NN. Relating neural dynamics to neural coding. Physical Review Letters. 2007; 99: 248103.

[10] Eurich CW. Neural Dynamics and Neural Coding: Two Complementary Approaches to an Understanding of the Nervous System. Bremen: Universität Bremen. 2003.

[11] Grebogi C, Ott E, Yorke JA. Chaos, strange attractors, and fractal basin boundaries in nonlinear dynamics. Science. 1987; 238: 632–638.

[12] EM Izhikevich. Dynamical systems in neuroscience. Cambridge, MA: MIT press. 2007.

[13] Brette R. Is coding a relevant metaphor for the brain? Behavioral and Brain Sciences. 2019; 42: e215.

[14] Li C, Gulledge AT. NMDA Receptors Enhance the Fidelity of Synaptic Integration. Eneuro. 2021; 8: ENEURO.0396–ENEU20.2020.

[15] Gidon A, Zolnik TA, Fidzinski P, Bolduan F, Papoutsi A, Poirazi P, *et al*. Dendritic action potentials and computation in human layer 2/3 cortical neurons. Science. 2020; 367: 83–87.

[16] Sharif B, Ase AR, Ribeiro-da-Silva A, Séguéla P. Differential Coding of Itch and Pain by a Subpopulation of Primary Afferent Neurons. Neuron. 2020; 106: 940–951.e4.

[17] Stone JV. Principles of Neural Information Theory: Computational Neuroscience and Metabolic Efficiency. Sheffield, SYK: Sebtel Press. 2018.

[18] Laughlin SB, de Ruyter van Steveninck RR, Anderson JC. The metabolic cost of neural information. Nature Neuroscience. 1998; 1: 36–41.

[19] Borst A, Theunissen FE. Information theory and neural coding. Nature Neuroscience. 1999; 2: 947–957.

[20] Timme NM, Lapish C. A Tutorial for Information Theory in Neuroscience. ENeuro. 2018; 5: ENEURO.0052-18.2018.

[21] Shannon CE. A Mathematical Theory of Communication. Bell System Technical Journal. 1948; 27: 623–656.

[22] Stone JV. Information theory: a tutorial introduction. Sheffield, SYK: Sebtel Press. 2015.

[23] Schreiber T. Measuring Information Transfer. Physical Review Letters. 2000; 85: 461–464.

[24] Wibral M, Priesemann V, Kay JW, Lizier JT, Phillips WA. Partial information decomposition as a unified approach to the specification of neural goal functions. Brain and Cognition. 2017; 112: 25–38.

[25] Schuman CD, Potok TE, Patton RM, Birdwell JD, Dean ME, Rose GS, *et al*. A Survey of Neuromorphic Computing and Neural Networks in Hardware. arXiv. 2017. (in press)

[26] Hassabis D, Kumaran D, Summerfield C, Botvinick M. Neuroscience-Inspired Artificial Intelligence. Neuron. 2017; 95: 245–258.

[27] Blomfield S. Arithmetical operations performed by nerve cells. Brain Research. 1974; 69: 115–124.

[28] Silver RA. Neuronal arithmetic. Nature Reviews Neuroscience. 2010; 11: 474–489.

[29] Woo J, Kim SH, Han K, Choi M. Characterization of dynamics and information processing of integrate-and-fire neuron models. 2021 (Under Review).

[30] ROSENBLATT F. The perceptron: a probabilistic model for information storage and organization in the brain. Psychological Review. 1958; 65: 386–408.

[31] Wilson HR, Cowan JD. Excitatory and inhibitory interactions in localized populations of model neurons. Biophysical Journal. 1972; 12: 1–24.

[32] Indiveri G, Liu S. Memory and Information Processing in Neuromorphic Systems. Proceedings of the IEEE. 2015; 103: 1379–1397.

[33] Yaghini Bonabi S, Asgharian H, Safari S, Nili Ahmadabadi M. FPGA implementation of a biological neural network based on the hodgkin-huxley neuron model. Frontiers in Neuroscience. 2014; 8: 1–12.

[34] Rice KL, Bhuiyan MA, Taha TM, Vutsinas CN, Smith MC. FPGA Implementation of Izhikevich Spiking Neural Networks for Character Recognition. In: Prasanna V, Torres L, Cumplido R, editors. 2009 International Conference on Reconfigurable Computing and FPGAs. Cancun, Mexico: IEEE. 2009; 451–456.

[35] Millner S, Grübl A, Meier K, Schemmel J, Schwartz MO. A VLSI implementation of the adaptive exponential integrate-and-fire neuron model. In: Lafferty J, Williams C, Shawe-Taylor J, Zemel R, Culotta A, editors. Advances in Neural Informa-

tion Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. San Diego, CA: NIPS. 2010; 1642–1650.

[36] Wilmes KA, Sprekeler H, Schreiber S. Inhibition as a Binary Switch for Excitatory Plasticity in Pyramidal Neurons. PLOS Computational Biology. 2016; 12: e1004768.

[37] Deneve S. Bayesian spiking neurons II: learning. Neural Computation. 2008; 20: 118–145.

[38] Deneve S. Bayesian spiking neurons i: inference. Neural Computation. 2008; 20: 91–117.

[39] Deneve S. Bayesian inference in spiking neurons. In: Advances in neural information processing systems. MA: MIT Press Cambridge. 2005.

[40] Lochmann T, Denève S. Information transmission with spiking Bayesian neurons. New Journal of Physics. 2008; 10: 55019.

[41] Zeldenrust F, de Knecht S, Wadman WJ, Denève S, Gutkin B. Estimating the Information Extracted by a Single Spiking Neuron from a Continuous Input Time Series. Frontiers in Computational Neuroscience. 2017; 11: 49.

[42] Rosenkranz JA, Venheim ER, Padival M. Chronic stress causes amygdala hyperexcitability in rodents. Biological Psychiatry. 2010; 67: 1128–1136.

[43] Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. Journal of Physiology. 1952; 117: 500–544.

[44] Pospischil M, Toledo-Rodriguez M, Monier C, Piwkowska Z, Bal T, Frégnac Y, *et al*. Minimal Hodgkin-Huxley type models for different classes of cortical and thalamic neurons. Biological Cybernetics. 2008; 99: 427–441.

[45] Herz AVM, Gollisch T, Machens CK, Jaeger D. Modeling single-neuron dynamics and computations: a balance of detail and abstraction. Science. 2006; 314: 80–85.

[46] Vetter P, Roth A, Häusser M. Propagation of Action Potentials in Dendrites Depends on Dendritic Morphology. Journal of Neurophysiology. 2001; 85: 926–937.

[47] Holmes WR. The role of dendritic diameters in maximizing the effectiveness of synaptic inputs. Brain Research. 1989; 478: 127–137.

[48] Shelton DP. Membrane resistivity estimated for the Purkinje neuron by means of a passive computer model. Neuroscience. 1985; 14: 111–131.

[49] Roth A, Häusser M. Compartmental models of rat cerebellar Purkinje cells based on simultaneous somatic and dendritic patch-clamp recordings. the Journal of Physiology. 2001; 535: 445–472.

[50] Kim SH, Woo J, Choi K, Choi M, Han K. Modulation of neural information processing by multimodal synaptic transmission. 2021 (Under Review).

[51] Woo J, Kim SH, Choi K, Choi M, Han K. The structural aspects of neural dynamics and computation: simulations and information-theoretic analysis. 2021 (To be Submitted).

[52] Hebb DO. The organization of behavior; a neuropsycholocigal theory. New York, NY: Wiley. 1949.

[53] Caporale N, Dan Y. Spike timing-dependent plasticity: a Hebbian learning rule. Annual Review of Neuroscience. 2008; 31: 25–46.

[54] Bienenstock EL, Cooper LN, Munro PW. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. Journal of Neuroscience. 1982; 2: 32–48.

[55] Cooper LN, Bear MF. The BCM theory of synapse modification at 30: interaction of theory with experiment. Nature Reviews. Neuroscience. 2012; 13: 798–810.

[56] Abraham WC, Mason-Parker SE, Bear MF, Webb S, Tate WP. Heterosynaptic metaplasticity in the hippocampus in vivo: a BCM-like modifiable threshold for LTP. Proceedings of the National Academy of Sciences of the United States of America. 2001; 98: 10924–10929.

[57] Benuskova L, Abraham WC. STDP rule endowed with the BCM sliding threshold accounts for hippocampal heterosynaptic plasticity. Journal of Computational Neuroscience. 2007; 22: 129–133.

[58] Jedlicka P, Benuskova L, Abraham WC. A Voltage-Based STDP Rule Combined with Fast BCM-Like Metaplasticity Accounts for LTP and Concurrent “Heterosynaptic” LTD in the Dentate Gyrus In Vivo. PLOS Computational Biology. 2015; 11: e1004588.

[59] Witter MP, Naber PA, van Haeften T, Machielsen WC, Rombouts SA, Barkhof F, *et al*. Cortico-hippocampal communication by way of parallel parahippocampal-subicular pathways. Hippocampus. 2000; 10: 398–410.

[60] Tyrrell T, Willshaw D. Cerebellar cortex: its simulation and the relevance of Marr’s theory. Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences. 1992; 336: 239–257.

[61] Simpson JI, Wylie DR, De Zeeuw CI. On climbing fiber signals and their consequence(s) Behavioral and Brain Sciences. 1996; 19: 384–398.

[62] Bailey CH, Giustetto M, Huang YY, Hawkins RD, Kandel ER. Is heterosynaptic modulation essential for stabilizing Hebbian plasticity and memory? Nature Reviews. Neuroscience. 2001; 1: 11–20.

[63] Kandel ER, Schwartz JH, Jessell TM, Siegelbaum S, Hudspeth AJ, Mack S. Principles of neural science. New York: McGraw-hill. 2000.

[64] Vinogradova OS. Hippocampus as comparator: role of the two input and two output systems of the hippocampus in selection and registration of information. Hippocampus. 2001; 11: 578–598.

[65] Staubli UV, Ji ZX. The induction of homo- vs. heterosynaptic LTD in area CA1 of hippocampal slices from adult rats. Brain Research. 1996; 714: 169–176.

[66] Oh WC, Parajuli LK, Zito K. Heterosynaptic structural plasticity on local dendritic segments of hippocampal CA1 neurons. Cell Reports. 2015; 10: 162–169.

[67] Jörntell H, Hansel C. Synaptic memories upside down: bidirectional plasticity at cerebellar parallel fiber-Purkinje cell synapses. Neuron. 2006; 52: 227–238.

[68] Marr D. A theory of cerebellar cortex. Journal of Physiology. 1969; 202: 437–470.

[69] Albus JS. A theory of cerebellar function. Mathematical Biosciences. 1971; 10: 25–61.

[70] Miller KD. Synaptic economics: competition and cooperation in synaptic plasticity. Neuron. 1996; 17: 371–374.

[71] Ramiro-Cortés Y, Hobbiss AF, Israely I. Synaptic competition in structural plasticity and cognitive function. Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences. 2014; 369: 20130157.

[72] McGoff KA, Guo X, Deckard A, Kelliher CM, Leman AR, Francey LJ, *et al*. The Local Edge Machine: inference of dynamic models of gene regulation. Genome Biology. 2016; 17: 214.

[73] Oates CJ, Dondelinger F, Bayani N, Korkola J, Gray JW, Mukherjee S. Causal network inference using biochemical kinetics. Bioinformatics. 2014; 30: i468–i474.

[74] Daniels BC, Nemenman I. Efficient inference of parsimonious phenomenological models of cellular dynamics using S-systems and alternating regression. PLoS ONE. 2015; 10: e0119821.

[75] Yan J, Deforet M, Boyle KE, Rahman R, Liang R, Okegbe C, *et al*. Bow-tie signaling in c-di-GMP: Machine learning in a simple biochemical network. PLoS Computational Biology. 2017; 13: e1005677.

[76] Fisher J, Woodhouse S. Program synthesis meets deep learning for decoding regulatory networks. Current Opinion in Systems Biology. 2017; 4: 64–70.

[77] Mirhoseini A, Goldie A, Yazgan M, Jiang JW, Songhori E, Wang S, *et al*. A graph placement methodology for fast chip design. Nature. 2021; 594: 207–212.

[78] Jin J, Yuan Y, Pan W, Tomlin C, Webb AA, Goncalves J. Identification of nonlinear sparse networks using sparse Bayesian learning. In: 2017 IEEE 56th Annual Conference on Decision and Control (CDC). Melbourne, Australia: IEEE. 2017; 6481-6486.

[79] Zoppoli P, Morganella S, Ceccarelli M. TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. BMC Bioinformatics. 2010; 11: 154.

[80] Henriques D, Villaverde AF, Rocha M, Saez-Rodriguez J, Banga JR. Data-driven reverse engineering of signaling pathways using ensembles of dynamic models. PLoS Computational Biology. 2017; 13: e1005379.

[81] Bonneau R, Reiss DJ, Shannon P, Facciotti M, Hood L, Baliga NS, *et al*. The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. Genome Biology. 2006; 7: R36.

[82] Pan W, Yuan Y, Ljung L, Goncalves J, Stan G. Identification of Nonlinear State-Space Systems from Heterogeneous Datasets. IEEE Transactions on Control of Network Systems. 2017; 5: 737–747.

[83] Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, *et al*. Predicting network activity from high throughput metabolomics. PLoS Computational Biology. 2013; 9: e1003123.

[84] Fakhfakh M, Cooren Y, Sallem A, Loulou M, Siarry P. Analog circuit design optimization through the particle swarm optimization technique. Analog Integrated Circuits and Signal Processing. 2010; 63: 71–82.

[85] Torun HM, Swaminathan M, Kavungal Davis A, Bellaredj MLF. A Global Bayesian Optimization Algorithm and its Application to Integrated System Design. IEEE Transactions on very Large Scale Integration (VLSI) Systems. 2018; 26: 792–802.

[86] Stanley KO, Clune J, Lehman J, Miikkulainen R. Designing neural networks through neuroevolution. Nature Machine Intelligence. 2019; 1: 24–35.

[87] Sun Y, Xue B, Zhang M, Yen GG, Lv J. Automatically Designing CNN Architectures Using the Genetic Algorithm for Image Classification. IEEE Transactions on Cybernetics. 2020; 50: 3840–3854.

[88] Gao S, Zhou M, Wang Y, Cheng J, Yachi H, Wang J. Dendritic Neuron Model with Effective Learning Algorithms for Classification, Approximation, and Prediction. IEEE Transactions on Neural Networks and Learning Systems. 2019; 30: 601–614.

[89] Stanley KO, Miikkulainen R. Evolving neural networks through augmenting topologies. Evolutionary Computation. 2002; 10: 99–127.

[90] Izzo D, Ruciński M, Biscani F. The Generalized Island Model. The generalized island model. In: Parallel Architectures and Bioinspired Algorithms. Berlin, Heidelberg: Springer. 2012.

[91] MacKay DM, McCulloch WS. The limiting information capacity of a neuronal link. the Bulletin of Mathematical Biophysics. 1952; 14: 127–135.

[92] Minsky M, Papert S. Perceptrons. Cambridge, MA: MIT Press. 1969.

[93] Fromherz P, Gaede V. Exclusive-or function of single arborized neuron. Biological Cybernetics. 1993; 69: 337–344.

[94] Cazé RD, Humphries M, Gutkin B. Passive dendrites enable single neurons to compute linearly non-separable functions. PLoS Computational Biology. 2013; 9: e1002867.

[95] Moldwin T, Kalmenson M, Segev I. The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. PLOS Computational Biology. 2021; 17: e1009015.

[96] Jones IS, Kording KP. Might a Single Neuron Solve Interesting Machine Learning Problems through Successive Computations on its Dendritic Tree? Neural Computation. 2021; 33: 1554–1571.

[97] Chavlis S, Poirazi P. Drawing inspiration from biological dendrites to empower artificial neural networks. Current Opinion in Neurobiology. 2021; 70: 1–10.

[98] Stöckel A, Eliasmith C. Passive Nonlinear Dendritic Interactions as a Computational Resource in Spiking Neural Networks. Neural Computation. 2021; 33: 96–128.

[99] Thill S, Svensson H, Ziemke T. Modeling the Development of Goal-Specificity in Mirror Neurons. Cognitive Computation. 2011; 3: 525–538.

[100] Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, *et al*. Learning transferable visual models from natural language supervision. arXiv. 2021. (in press)

Abbreviations: AI, artificial intelligence; ATP, adenosine triphosphate; CA, cornu Ammonis; EA, evolutionary algorithm; EIF, exponential integrate, and, fire; FR, firing, rate; IF, integrate and fire; LIF, leaky integrate-and-fire; LT, linear threshold, MP, McCulloch and Pitts; PF, Purkinje fiber; PID, Partial information decomposition; QIF, quadratic integrate-and-fire.

Keywords: Neural dynamics; Neural coding; Logic operation; Synaptic transmission; Information theory

Send correspondence to:

Kyungreem Han, Laboratory of Computational Neurophysics, Convergence Research Center for Brain Science, Brain Science Institute, Korea Institute of Science and Technology, 02792 Seoul, Republic of Korea, E-mail: khan@kist.re.kr

MooYoung Choi, Department of Physics and Astronomy and Center for Theoretical Physics, Seoul National University, 08826 Seoul, Republic of Korea, E-mail: mychoi@snu.ac.kr

† These authors contributed equally.