Autapses enable temporal pattern recognition in spiking neural networks

Muhammad Yaqoob[1,2,*], **Volker Steuber**[1,+], **and Borys Wróbel**[2,3,+]

[1]University of Hertfordshire, Hatfield, United Kingdom
[2]Laboratory of Bioengineering, Institute of Molecular Biology and Biotechnology, Adam Mickiewicz University in Poznan, Poland
[3]European Institute for Brain Research, Amstelveen, The Netherlands
[*]m.yaqoob3@herts.ac.uk
[+]these authors contributed equally to this work

ABSTRACT

Most sensory stimuli are temporal in structure. How action potentials encode the information incoming from sensory stimuli remains one of the central research questions in neuroscience. Although there is evidence that the precise timing of spikes represents information in spiking neuronal networks, information processing in spiking networks is still not fully understood. One feasible way to understand the working mechanism of a spiking network is to associate the structural connectivity of the network with the corresponding functional behaviour. This work demonstrates the structure-function mapping of spiking networks evolved (or handcrafted) for a temporal pattern recognition task. The task is to recognise a specific order of the input signals so that the *Output* neurone of the network spikes only for the correct placement and remains silent for all others. The minimal networks obtained for this task revealed the twofold importance of autapses in recognition; first, autapses simplify the switching among different network states. Second, autapses enable a network to maintain a network state, a form of memory. To show that the recognition task is accomplished by transitions between network states, we map the network states of a functional spiking neural network (SNN) onto the states of a finite-state transducer (FST, a formal model of computation that generates output symbols, here: spikes or no spikes at specific times, in response to input, here: a series of input signals). Finally, based on our understanding, we define rules for constructing the topology of a network handcrafted for recognising a subsequence of signals (pattern) in a particular order. The analysis of minimal networks recognising patterns of different lengths (two to six) revealed a positive correlation between the pattern length and the number of autaptic connections in the network. Furthermore, in agreement with the behaviour of neurones in the network, we were able to associate specific functional roles of 'locking,' 'switching,' and 'accepting' to neurones.

Introduction

In biological neural networks, the problem of temporal pattern recognition refers to identifying a sequence of input signals or spikes that carry information. The brain processes such sequences elegantly and responds quickly[1–3]. Biological nervous systems can efficiently differentiate spike patterns distributed across time and space[4]. However, little is known about how individual neurones contribute to processing temporal signals. Interpreting a sequence of spikes generated by a neurone (or a group of neurones) to determine the spatial or temporal structure of a stimulus is a fundamental problem in neuroscience[5]. Analogue sensory information from different modalities, including olfactory, auditory, and visual input, is encoded in the form of spikes[6], and processed precisely by the brain with incredible speed[7]. The timing of the spikes captures the varying transient intensities of the stimulus, and even a single spike represents remarkable information after stimulus onset[8,9]. These results suggest that temporal features of spikes can precisely represent a stimulus and convey information in biological and artificial spiking neural networks[10]. In the last 15 years, it has been established that sensory information is represented by the precise timing of spikes in the somatosensory, auditory, visual, and olfactory systems[9,11–17]. Furthermore, several studies have shown that information processing in the nervous system is linked to transitions from one spiking behaviour to another[5,18–23]. However, despite extensive research, the association between structural connectivity and the functional behaviour of neural systems remains unclear. Understanding how spiking neural networks process information to gain insight into the computational capabilities of a biological brain is one of the most challenging problems in computational neuroscience[24,25]. A possible way to study the complex computational processing capabilities of the brain is to artificially produce networks with biologically plausible neurones that can perform a computational task.

From a computational perspective, processing temporal spike patterns is a general computational task performed by the

brain[26, 27]. There are two common ways of learning to recognise temporal patterns: (i) adjusting conduction delays[28–31], (ii) selecting conduction delays from a spectrum of existing delays[20, 32, 33]. Spiking neural networks (SNNs) are documented to differentiate temporal patterns by exploiting different time delays and pathways in the network[6]. In neural systems, time delays can be adjusted at the level of synapse, axon, or soma. Adjusting these delays at one or more levels in the network can uncover features of a signal[34]. On the other hand, it is possible that signals produced at different timings arrive together at the readout neurone, generating a maximal response. This phenomenon is used for edge detection in the visual system[35]. Moreover, delays in the network can also be used to identify keywords in continuous speech[36]. This work provides a new way of learning to recognise temporal patterns by evolving (constructing following the rules borrowed from the natural evolution of population and selection) the topology and connection weights of a population of SNNs, without changing or selecting conduction delays.

Artificial spiking neural networks can process a large amount of data using an efficient spiking communication mechanism among neurones such that information is transmitted only when a neurone spikes[9]. Due to their similarities to biological networks, SNNs are widely used to model information processing in animal brains. SNNs have been shown to be computationally more powerful, especially in terms of speed and accuracy, and can solve problems more efficiently than non-spiking neural networks[37, 38]. Furthermore, SNNs are considered robust to noise and damage, and their functionality degrades gracefully[39, 40]. To understand the learning and information processing mechanism of SNNs, we first need to obtain networks that can perform a specific computational task. Considering the fact that information received from the majority of sensory modalities is temporal in structure, in this work we evolved the structural topology and connection weights of a population of SNNs using a genetic algorithm to recognise temporal patterns. This task can be accomplished by recurrent connections (a form of memory) or delays in the network[29, 41]. In general, a recurrent connection can be of any size. However, the shortest possible recurrent connection observed in the nervous system is a self-connection. Self-connections (autaptic connections or autapses) are recurrent synaptic connections between the axon and dendrites or soma of a single neurone (either excitatory or inhibitory). Discovered five decades ago[42], autapses can be observed in the mammalian brain in the neocortex, hippocampus, and cerebellum[43]. Recent studies suggest possible roles of autapses in the synchronisation of networks[44], flexible working memory networks[45], and coherence resonance[46]. However, their functional role remains unknown[47].

The computational task for SNNs in this work is to recognise a given subsequence of signals in a continuous stream of input signals. A single *Output* neurone spikes for the correct input pattern while remaining silent for other input patterns. We show the importance of autapses by revealing their possible functional role in state maintenance—a form of memory in the network. Furthermore, SNNs with autaptic connections tend to evolve a simplified switching mechanism to recognise patterns of lengths three and four[48]. Consistent with the evolved networks, we define rules for constructing the topology of a network by hand to recognise patterns up to length six with six interneurones, demonstrating a perfect linear relation between the number of signals in the pattern and the number of interneurones in the network. We show that autapses are crucial for switching the network between states and maintaining a network state. Furthermore, we indicate that the pattern length to be recognised correlates positively with the number of autaptic connections in the network. More specifically, recognising a pattern of n signals requires a network of n interneurones with n-1 autaptic connections. Finally, we demonstrate that, in addition to the other neurones (N1, $N2$, . . .), a successful recogniser network must have three specialised neurones: a *Lock*, a *Switch*, and an *Accept* neurone. The activity of *Lock* prevents *Output* from spiking, except when the network receives the second to last correct input signal and allows the *Output* neurone to spike in response to the correct last input. The *Switch* neurone is responsible for transitions between the network start state and inter-signal network states. The *Accept* neurone forces spike(s) in the *Output* neurone if the lock is released by the penultimate signal in the pattern to be recognised and switches the network back to the *Start* state by activating the *Switch* neurone.

Methods

The SNNs in this work consist of adaptive exponential integrate and fire neurones (AdEx) with a set of parameters that result in tonic spiking when a constant input current is injected into the neurone[49]. Each AdEx neurone has four state variables: membrane potential V, excitatory conductance gE, and inhibitory conductance gI, and adaptation w, along with 14 parameters (Table 1):

Out of the 14 parameters, four bifurcation parameters are responsible for the spiking behaviour: the adaptation conductance a, the spike-triggered adaptation b, the adaptation time constant τ_w, and the resting potential V_r. The remaining scaling parameters are: the total capacitance C, the total leak conductance g_l, the effective rest potential E_l, the inhibitory E_{in} and the excitatory E_{ex} reversal potential, the threshold slope factor Δ_T, the effective threshold potential V_T, and the two time constants for excitatory synapses τ_{ex} and inhibitory synapses τ_{in}.

$$\frac{dV}{dt} = \frac{1}{C}(g_E(E_E - V) + g_I(E_I - V) - w) + \frac{1}{\tau_m}(E_L - V + \Delta_T e^{(\frac{V-V_T}{\Delta_T})}) \tag{1}$$

$$\frac{dw}{dt} = \frac{1}{\tau_w}a(V - E_L) - w \tag{2}$$

$$\frac{dg_E}{dt} = \frac{-g_E}{\tau_E} \tag{3}$$

$$\frac{dg_I}{dt} = \frac{-g_I}{\tau_I} \tag{4}$$

The exponential term in equation 1 defines the spike generation mechanism and the ascent of the action potential. In the mathematical description of the model, a spike is fired at time t^f when the membrane potential crosses an arbitrary firing threshold value (larger than V_T, say +30 mV). When this happens, the integration of the differential equations (Equations 1 to 4) is stopped, the spike time t^f is recorded, and the voltage is reset to a fixed value V_r. This reset describes the descent of the action potential, given by:

at $\quad t = t^f \quad$ reset $\quad V \longrightarrow V_r$

Simultaneously, when a spike is recorded at time t^f, the adaptation current w increases by an amount b:

at $\quad t = t^f \quad$ reset $\quad w \longrightarrow w + b$

The interaction between the differential equations of the AdEx model and the above two discrete resets can generate a variety of spiking behaviours[49]. In this work, we use the parameters (Table 1) for producing tonic spiking when a step current is injected into a neurone. The state variables are integrated with a time step of 1 ms using Euler integration (using a time step of 0.1 ms or more precise integration algorithms did not affect the results in preliminary experiments, and was not practical from the point of view of computational time, even when using high throughput computing resources made available for this project). The membrane potential is affected by noise; at each time step, a random value is added to it, drawn from a normal distribution with a mean of 0 and a standard deviation of 1 mV.

Parameter		Value
E_l	effective rest potential	-70 mV
E_{in}	inhibitory reversal potential	-70 mV
E_{ex}	excitatory reversal potential	0 mV
V_r	reset voltage	-58 mV
V_T	effective threshold potential	-50mV
V_{th}	spike detection threshold	0 mV
Δ_T	threshold slope factor	2 mV
C	total capacitance	0.2 nF
g_l	total leak conductance	10 nS
a	adaptation conductance	2 nS
b	spike-triggered adaptation	0 pA
τ_w	adaptation time constant	30 ms
τ_{ex}	excitatory time constant	5 ms
τ_{in}	inhibitory time constant	5 ms

Table 1. AdEx parameters for tonic spiking

Evolution of networks for recognising a pattern of length 3. To obtain networks that recognise a pattern of three signals in a particular order, we used a genetic algorithm originally developed for evolving gene regulatory networks[50,51], where the topologies of the networks in the population are encoded as linear genomes. Each genome contains a list of genetic elements such that each element has three attributes: a type (input I, output O, dendrite D, or axon terminal A), a sign (+, -), and coordinates (x, y) (Figure 1). A sequence of D elements followed by a sequence of A elements encodes one neurone in the network. To determine the total synaptic strength (weight) of a connection between two neurones in the network, we aggregate the affinity between all A elements of the presynaptic neurone and all D elements of the postsynaptic neurone. Each element

has (x,y) coordinates in an abstract affinity space; the smaller the Euclidean distance between two elements i and j, the larger the contribution to the synaptic strength given by $s_i s_j \frac{2(5-d_{i,j})}{10d_{i,j}+1}$. If the signs ($s_i$ and s_j) are the same (different), the contribution is positive (negative). The pre-post connection is established only if the absolute value of the sum of the weight contributions is above a cut-off threshold (to prevent full connectivity); a positive (negative) sum results in an excitatory (inhibitory) connection. In this work, a given presynaptic neurone can excite some of its targets and inhibit others – violating Dale's principle[52]). However, any violating neurone in the evolved network can be transformed to follow Dale's principle by dividing it into two parts, one excitatory and one inhibitory (each having the same number of incoming connections, with the same weights, as the original neurone), so that network performance is not compromised[48].
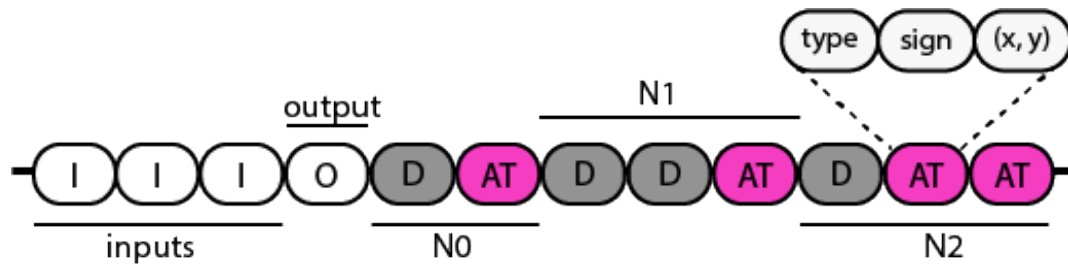

Figure 1. The structure of a random linear genome in the population, encoding a network with three interneurones, three input nodes and one *Output* neurone. A sequence of D elements (dendrites) followed by a sequence of A elements (axon terminals) is considered one neurone. Each element has a type (I, O, D, A), a sign (+, -) and (x, y) coordinates in 2D space. The strength of a connection between two elements is defined as an inverse function of the Euclidean distance between their coordinates

We use the genetic algorithm[50] with a constant population size of 300. The initial population is created with random topologies encoded as linear genomes. Subsequent generations are created with a size two tournament selection and an elite count of 10 individuals. Four genetic operators, point mutation, deletion, duplication, and crossover, are described in a previous work[40], and their probabilities can be found therein. A recurrent SNN for recognising a pattern of three signals was found to require at least three interneurones, one *Output* neurone and three input nodes. The inputs are not allowed to connect to *Output* directly, and only the interneurones are allowed to have self-loops (autaptic connections). During evolution, the genetic algorithm can add/remove connections by moving the (x,y) coordinates associated with each genetic element.

The task of the networks is to recognise a pattern of three signals in a continuous stream of signals in which all signals (A, B, and C) occur with equal probability. In an input sequence, the length (duration) of a signal is 6 ms, followed by a silence interval of 24 ms. During evolution, each network in the population is evaluated for six sequences of signals. Four out of these six sequences are generated randomly with an equiprobable occurrence of the three signals (A, B, and C). The remaining two sequences are created by concatenating hard-to-differentiate patterns (ABA, ABB, ABC, BBC) in random order. The fitness function rewards the spiking of the *Output* neurone in the correct inter-stimulus interval after the occurrence of the correct pattern ABC and penalises spikes in all other intervals. The inter-stimulus interval is defined as the interval between the onsets of two consecutive stimuli.

$$f_{fitness} = 1 - (R - kP) \tag{5}$$

where R is the normalised reward given by the number of inter-stimulus intervals after C in which the *Output* neurone spikes after receiving the correct pattern ABC divided by the total number of correct patterns in the input sequence. The penalty P is the number of inter-stimulus intervals in which the *Output* neurone spikes incorrectly, divided by the total number of inter-stimulus intervals in the sequence (which is one less than the total number of signals in the sequence). As this denominator is large, the normalised penalty P is amplified by a constant $k = 4$.

Evolution tends to produce superfluous connections that can be pruned without impairing the performance of the network[48]. To aid the network analysis, we prune the evolved networks by removing a random connection and testing if the performance of the network is compromised. If it is, the connection is put back and labelled vital. Otherwise, the excessive connection is removed from the network. This process is repeated until only vital connections remain in the network.

Pruning revealed structural similarities between the networks obtained from different independent evolutionary runs; thus, the networks that recognised a pattern of three signals were either equal or isomorphic. Moreover, the recognition of a pattern depends on the transitions between the network states. Therefore, the states of an evolved network can be mapped onto the states of the corresponding finite state transducer (FST), accepting a string of three letters.

Handcrafting networks for recognising patterns of length four and above. Understanding the working mechanism of three-signal networks allowed handcrafting network topologies that recognise longer patterns. For example, a network topology recognising a pattern of three signals can be extended so that it recognises a pattern of four signals by adding a new input, an inter-neurone and six synaptic connections (Figure 2b-c). In the extended network, the input stimuli are renamed *ABCD*. The newly added input A and the neurone *N*4 connect to the existing network; input A connects to *N*4 with an excitatory connection, input *B* (previously named input *A* and connected to neurone *N*3) excites *N*2 (the switch neurone), and input *B* also connects to the newly added neurone *N*4 with an inhibitory connection. The neurone *N*4 connects to *N*3 and to itself with excitatory connections, and the neurone *N*4 and *N*2 (the switch) inhibit each other and thus are mutually exclusive in terms of their activation state. In networks recognising longer patterns, two more connections are essential for seamless switching back to the start state after receiving the last input signal. Therefore, these connections are introduced in networks recognising patterns of length four and above: (i) the last input excites the switch neurone, and (ii) the lock (*N*3) neurone inhibits the *Accept* (*N*1) neurone. These general rules can be used to extend the topology of a four-signal network to obtain the topology for recognising patterns of lengths five and six (Figure 2d-e).

The structural topology and the sign of the connections are fixed in the handcrafted networks. We use a genetic algorithm to optimise only the weights of connections. Here, an individual's genome is the adjacency list of the handcrafted network. In the initial generation, a population of 100 individuals sharing the same structural topology is created. The weights of the excitatory (inhibitory) connections are drawn from a uniform distribution U[0, 10] (U[-10, 0]). The population size is kept constant during evolution with 10 individuals in the elite. The only mutation operator is adding a random number drawn from a normal distribution N[mean=0, SD=1] to the connection weight chosen with a probability of 0.1. Subsequent generations are created with a size two tournament selection and the fitness function is identical to the one for evolving both the topology and
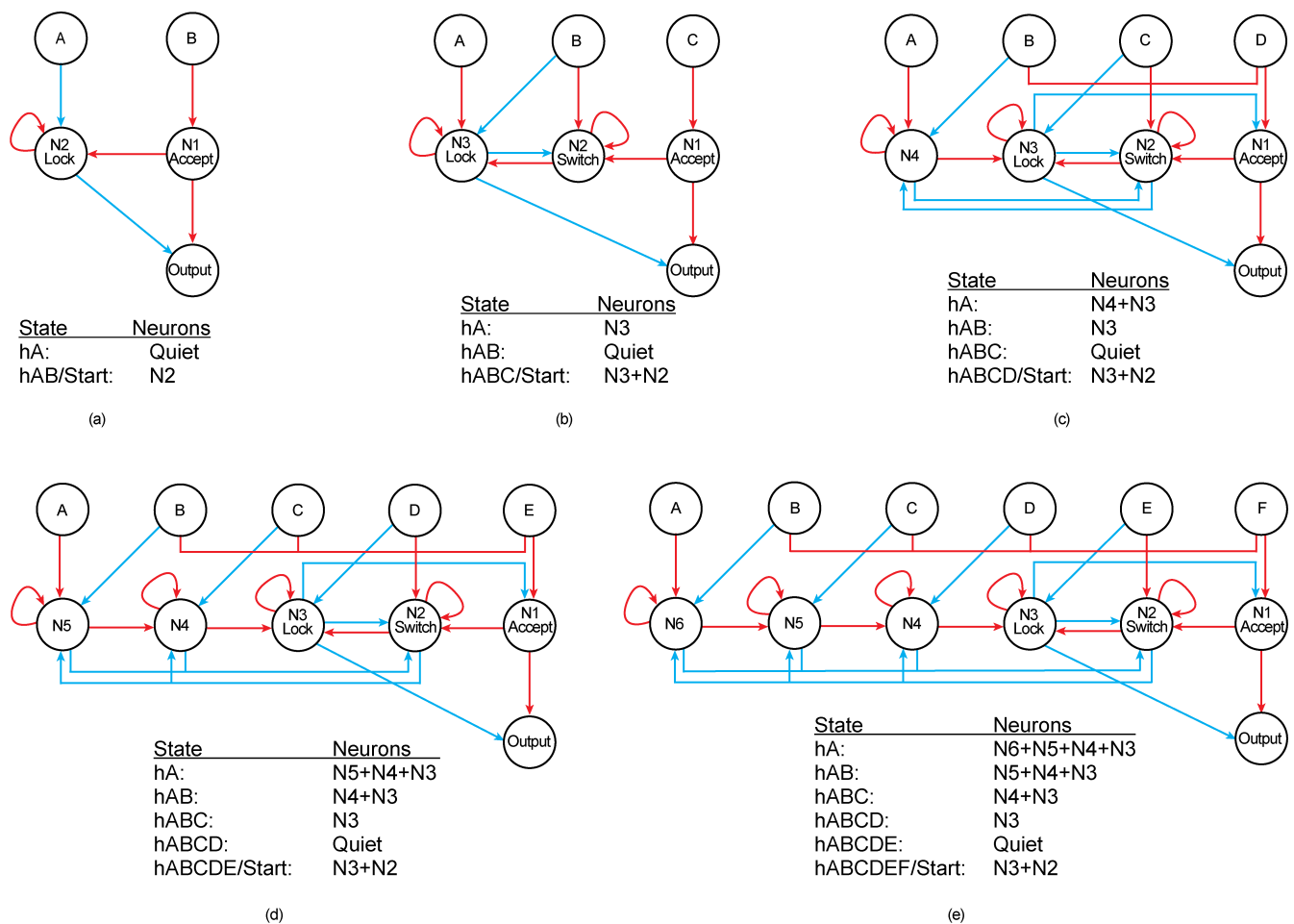

Figure 2. Networks recognising patterns of lengths 2 to 6. The first two networks in panels a and b were obtained with artificial evolution. The table below each network shows transitions between network states, represented by the number of active neurones. The topology is extended by hand to recognise patterns of lengths 4, 5, and 6 (c-e).

the connection weights (Equation 5), rewarding spikes in the correct inter-stimulus interval and penalising spikes elsewhere.

The number of possible patterns increases exponentially with the length of the pattern. There are n^n possible orderings for n signals. For example, a pattern of four signals has $4^4 = 256$ (AAAA to DDDD) possible permutations, and a pattern of six signals has $6^6 = 46656$ (AAAAAA to FFFFFF). This means that when n is large, a given permutation will occur less frequently in a random input stream of signals, and some patterns may not occur at all (so the networks cannot be penalised for recognising them). We have therefore designed a genetic algorithm that runs in two stages. Consider evolving networks for recognising a pattern of length four in a given order. In the first stage, the networks are optimised only for patterns similar to the target pattern ABCD. These similar patterns have the form AXXX, XXXD, where XXX in AXXX (XXXD) is replaced by all 27 possible patterns of BCD (ABC). A sequence of 10,000 signals is created by randomly concatenating the 54 patterns (ABBB to ADDD, AAAD to CCCD). Once the genetic algorithm converges for the first one (finds a network that only responds to the correct pattern ABCD and remains silent for all other patterns), the second stage begins by identifying hard-to-recognise patterns. This requires evaluating the network for a large random sequence with an equiprobable occurrence of signals A, B, C and D. A pattern is considered hard if the network responds incorrectly to multiple occurrences (at least 40% of the total number of occurrences) of a given pattern in a random sequence. Evolution continues with the fitness function in which the penalty coefficient is equal to 50 and the input sequence of 10,000 signals is created by randomly concatenating the 54 patterns (AXXX + XXXD) and hard patterns with equal probability. The second stage (identification of hard patterns, evolution) repeats until the algorithm converges and no hard patterns remain.

Results

States of the network correspond to the states of a finite state transducer. To understand how the evolved networks work, we show that the activity of a network can be mapped onto the states of a finite state transducer (FST)[39,48]. An FST is a finite state machine generally used for analysing time-structured data[53]. For example, an FST that accepts a string of three letters needs to have four distinct states (Figure 3b), and so does an SNN evolved to recognise a pattern of three signals (Figure 3a). We established an association between the network states of a pruned network and the states of the corresponding minimal FST. This association can also be obtained for an evolved network, but the mapping is more complex due to superfluous connections[48,54]. The *Start* state of the network corresponds to the continuous spiking of the neurones $N3$ and $N2$ (Figure 3c). Once the network receives the first correct symbol A, it goes into the *hA* (for "had A") state, maintained by continuous spiking of the $N3$ neurone only. The presence of an excitatory loop on $N3$ preserves the network state until the next signal (A, B, or C) is received. If the network receives another A, it remains in the *hA* state. However, if the network receives a B, it transforms to the *hAB* state retained by no activity in the network (all neurones in the network are quiescent). If C rather than B arrives after A, the network goes back to the *Start* state from the *hA* state.

Suppose that the network is in the *hAB* state and a signal C arrives. As a result, the *Accept* neurone spikes several times and activates the *Output* neurone. The *Accept* neurone also activates the *Switch* neurone to transform the network back into the *Start* state. On the other hand, receiving a signal B when the network is in the *hAB* state activates the neurone $N2$ (explained by the weak excitatory connection from input B to $N2$), which in turn activates $N3$, transforming the network into the *Start* state, represented by continuous spiking of $N2$ and $N3$. Furthermore, if the network receives an A while in the *hAB* state, the network switches back to the *hA* state characterised by continuous spiking of $N3$.

Specialised role of neurones in the network and the importance of autapses. The pruned networks obtained from independent evolutionary runs for recognising a pattern of three signals were equal or isomorphic[39,48], allowing us to discover their working mechanism. All resulting networks accomplished the task with four neurones (three interneurones and one *Output* neurone) and 11 connections (three inhibitory and eight excitatory). Furthermore, networks capable of maintaining network states exhibited self-excitatory loops on at least two interneurones[39,48]. To describe the working mechanism of the networks, first, the role of self-excitatory loops (autapses) is identified. Then, specific roles are associated with the interneurones on the basis of their spiking behaviour. Finally, we demonstrate the contribution of each connection to pattern recognition. The existence of self-excitatory loops enables the evolved networks to maintain network states irrespective of the silent intervals between signals. These recurrent connections permit three active states: the *hA* state by persistent spiking of the *Lock* neurone, and the *hABC/Start* states by causing tonic spiking of the *Lock* and *Switch* neurones (Figure 3a). The only difference between the *Start* and the *hABC* state is the intermittent spiking of the *Output* neurone. During the evolutionary process, both the *Lock* and the *Switch* neurones form self-excitatory loops with sufficient weights to prevent spiking activity from dying out in the absence of input activity (Figure 3d). In summary, the ability of the network to maintain a state for longer silent intervals requires the formation of autapses in these minimal networks[40,48].

The interneurones have specialised roles of *locking*, *switching* and *accepting*. Self-excitation of the *Lock* neurone prevents the *Output* neurone from spiking, except when the second to last correct input signal (for example, B in ABC) shuts down the
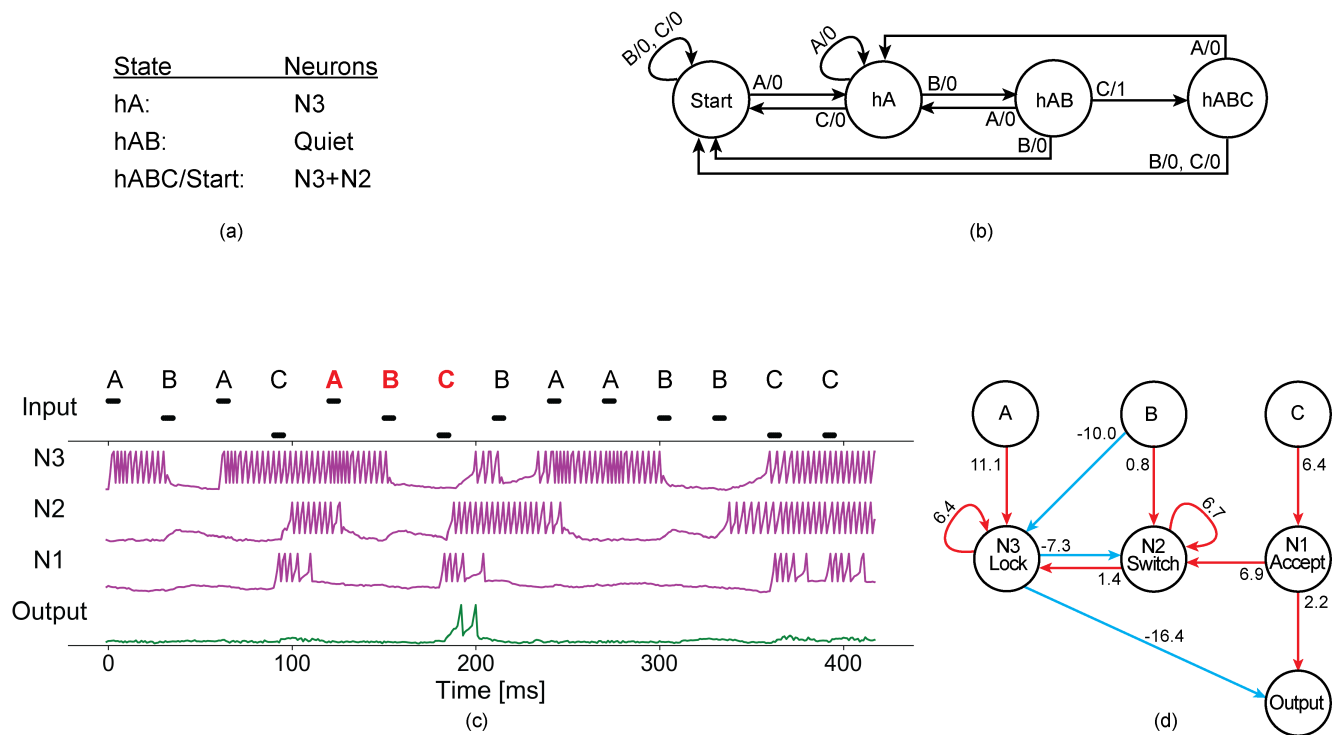
Figure 3. The working mechanism of a network evolved in the presence of membrane potential noise. (a) The states of the network and the corresponding activity of the neurones in the pruned network. (b) The minimal finite-state transducer that recognises ABC. (c) The activity of the network. (d) The topology of the pruned evolved network.

Lock neurone, enabling the *Output* neurone to spike for the last correct input signal (C in ABC). If the penultimate correct input signal releases the *Lock*, the *Accept* neurone activates the *Output* neurone on receiving the last correct input. The *Accept* neurone also sends a signal to the *Switch* neurone, transforming the network back to the *Start* state. The *Switch* neurone is responsible for the transitions between the *Start* state (when *Switch* is active) and the intermediate states hA and hAB (when *Switch* is quiescent).

Contribution of each connection to pattern recognition. The contribution of each synaptic connection to the recognition of input patterns is determined by its effect on the behaviour of postsynaptic neurones. The excitatory connection from the input A to the neurone $N3$ ($A \to N3$) activates $N3$, $N3$ excites itself with an autaptic connection $N3 \to N3$ and inhibits $N2$ with an inhibitory connection $N3 \to N2$, thus putting the network in the hA state in which only the neurone $N3$ spikes continuously (Figure 3c-d). The continuous spiking of $N3$ also prevents the *Output* neurone from spiking (due to an inhibitory connection $N3 \to Output$). When signal A is followed by B, the inhibitory connection ($B \to N3$) forces $N3$ (which maintains the state hA) to stop spiking, and the network goes into a quiescent state hAB. The input B is connected to $N2$ with a weak excitatory connection. This connection prevents an incorrect response of the network to a repeated signal B in the correct pattern ABC. The weight of this connection is adjusted so that the first B cannot activate the *Switch* neurone due to the continued inhibition of $N3$ ($N3 \to N2$). However, when $N3$ is released by the first B in the correct order, the second B can activate the *Switch* neurone, which in turn activates the neurone $N3$. As a result, the network goes back into the *Start* state (continuous spiking of both neurones $N3$ and $N2$). The positive connection from the input C to $N1$ triggers several spikes in the neurone $N1$, and $N1$ passes the activity to the *Output* neurone. Consequently, the *Output* spikes in response to the last correct input if the lock has been released by the second to last correct symbol. $N1$ also activates the $N2$/*Switch* neurone, which in turn activates the $N3$/*Lock* neurone, thus transforming the network back into the *Start* state.

Performance of the handcrafted network for patterns of six signals. The analysis of a network handcrafted for recognising a pattern of six signals shows that the network accomplishes the task of recognising the pattern with seven well-defined network states (Figure 4a). When the network receives the first correct input signal A around 180 ms (Figure 4c), the network goes into the hA (had A) state represented by four active neurones, $N6$, $N5$, $N4$, and $N3$. The input A is connected directly to $N6$ through a strong excitatory connection, which causes $N6$ to spike on arrival of A (Figure 4b). The

self-excitatory loop on N6 makes it spike continuously. This spiking activity, in turn, activates N5 and prevents N2 (the *Switch* neurone) from spiking. The spiking activity of N5 persists (due to the self-excitatory loop on N5), and N5 passes the activity to N4, which in turn activates N3 (the *Lock* neurone) in a similar way (Figure 4b-c). Meanwhile, when an input signal B is received, the inhibitory connection from the input B to N6 shuts down N6, transforming the network into the state hAB, which is characterised by continuous spiking of N5, N4 and N3. The input B also excites the *Switch* neurone with a precise connection weight such that only a second B signal can activate the *Switch* neurone and transform the network back to the *Start* state. Similarly, suppose that AB is followed by the third input signal C in the correct order. In that case, the inhibitory connection from the input C to N5 shuts down N5, transforming the network into the state hABC, represented by the continuous spiking of neurones N4 and N3. Next, if the network receives an input signal D in the correct order, it switches off N4 through an inhibitory connection from D to N4, transforming the network into the state hABCD, maintained by continuous spiking of N3 (the *Lock* neurone) only (Figure 4b-c). These four states from hA to hABCD are actively maintained by persistent spiking of four, three, two, and one neurone(s), respectively. It is important to note that N3 (the *Lock* neurone) is always active, except when the network receives the second to last input signal E in the correct order, allowing the *Output* neurone to spike for the correct last input signal F. The strong inhibitory connection from N3 to the *Output* neurone prevents *Output* from spiking for incorrect patterns. If the network is in the state hABCD (represented by continuous spiking of the *Lock* neurone) and it receives the second to the last input signal E in the correct order, it switches to a quiescent state by releasing the lock from *Output* through an inhibitory connection from E to the *Lock* neurone (N3), which enables the *Output* neurone to spike in response to the last correct input signal F (Figure 4b-c). The input F activates N1 through a positive connection, which in turn passes the activity to both the *Output* and the *Switch* neurones. The *Output* spikes in response to receiving the correct pattern ABCDEF, and the *Switch* neurone activates the *Lock* neurone. This transforms the network back into the *Start* state where the *Switch* (N2) and the *Lock* neurone (N3) spike continuously.

The performance of the handcrafted networks is evaluated in terms of precision and sensitivity. Precision is defined as the
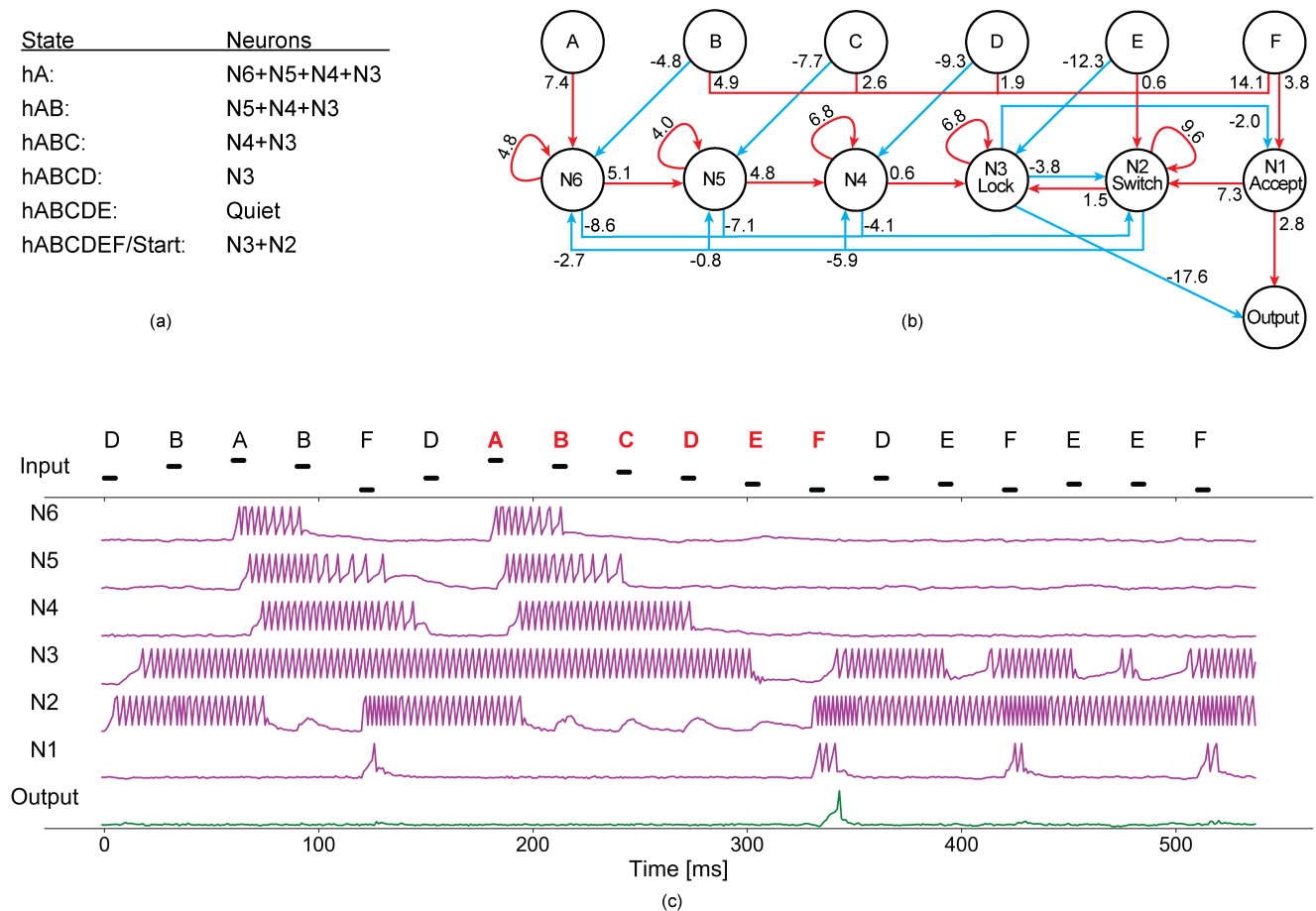

Figure 4. (a) The network states with corresponding active neurones. (b) The handcrafted network for recognising a pattern of length six. (c) The behaviour of each neurone in the network.
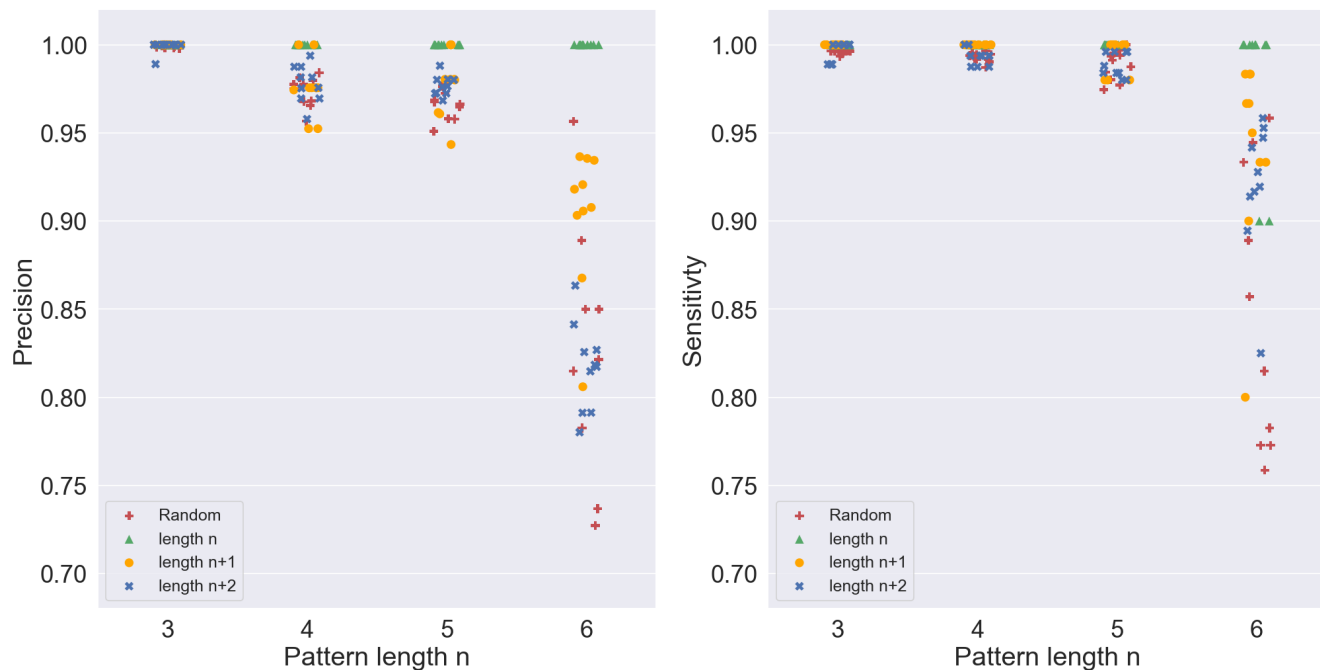
Figure 5. Performance degradation of handcrafted networks with increasing pattern length. The precision (left) and sensitivity (right) of the top 10 networks for each pattern size (3 to 6) are evaluated for a random sequence of length 1 million and all possible patterns of length n, n+1, and n+2.

fraction of *Output* spikes that are correct, that is, the number of true positives divided by the total number of times the *Output* spiked, while sensitivity is the fraction of correctly classified target patterns, that is, the number of true positives divided by the actual number of correct patterns in the sequence. The top 10 networks for recognising patterns of lengths three, four, five, and six are re-evaluated for a random sequence of one million signals (Figure 5). The performance of the networks degrades with increasing length of the pattern. Our results show that the precision of the top 10 networks obtained for recognising *ABCDEF* is between 0.73 and 0.96, while for length five (ABCDE) and below the precision is always above 0.94. The networks are then tested with all possible permutations of six signals in six positions with replacement $^6P_6$ (from AAAAAA to FFFFFF), $^6P_7$ (from AAAAAAA to FFFFFFF), and $^6P_8$ (from AAAAAAAA to FFFFFFFF). The testing of the networks with all possible patterns of lengths seven and eight ensures that the performance of the networks is not affected by the proceeding signals, that is, by history. Similarly, all possible permutations of input patterns of lengths three, four, and five are evaluated for up to two preceding signals (Figure 5).

To demonstrate that all false positives were caused by history or noise, each pattern was presented to the network 10 times. All perfect networks responded to the correct pattern XXABCDEF at least eight out of 10 times, while responding at most four out of 10 times sporadically for a very small number of false positives. This clear margin indicated that handcrafted networks could perfectly recognise correct patterns with up to six signals.

Discussion

In the present study, we have explored how spiking neural networks can perform temporal pattern recognition tasks without updating or selecting conduction delays. Understanding how the brain processes temporal information is not only an important research topic in computational neuroscience, but also crucial for reproducing artificial networks with brain-like computational capabilities. The neural circuitry of the brain performs a large number of computational tasks, including signal processing, memory, classification, decision-making, etc. It is important to link functional behaviours to the corresponding neural connectivity of the brain. However, the diverse set of computational capabilities, the enormous number of neurones, and the complexity of the brain make a principled mapping of neural connectivity to computational function difficult. In order to make inroads towards a basic understanding of information processing in neural systems, we evolved minimal artificial spiking neural networks for the computational task of temporal pattern recognition.

We found a clear association between structural connectivity and the functional properties of neural networks. We observed that the complexity of the networks increased with the length of the temporal patterns that could be recognised. The number

of possible temporal patterns increased exponentially with the length of the patterns, making network training more difficult. Intuitively, the recognition of longer patterns required a larger number of neurones in the network, exponentially increasing the number of connections in the network. Larger and more interconnected networks were harder to train due to the computational cost and larger search space. Consequently, the artificial evolution could not produce an optimal minimal network for recognising a pattern of length four and above, and we resorted to handcrafting the topology of networks for longer patterns (and using artificial evolution only to optimise the weights in these networks).

One of the most salient observations is the emergence of autapses in our spiking neural networks. Although autapses are widely observed in the brain, their functional role in neural information processing is not yet fully clear. Autapses have previously been shown to play a significant role in memory maintenance[55], oscillatory activity of a single neurone[56], switching between different network states[57], synchronisation of network activity[44], and signal detection[58]. Here, we show the importance of excitatory autaptic connections for network state transitions and network state maintenance. In our work, all active network states are maintained by excitatory autaptic connections, whereas state transitions occur when an excitatory autaptic connection of an active neurone is silenced by a strong inhibitory connection. Furthermore, we examined the relationship between the number of excitatory autapses in the network and the length of temporal patterns that can be recognised. The number of autapses in a network constrains and is equal to the number of distinct actively maintained states in the network. We observed that a minimal network recognising a pattern of length n requires at least n-1 autaptic connections. This is a necessary condition for a network to be able to maintain network states. A sufficient condition is that the weights of the autaptic connections are strong enough to keep the spiking activity from dying out in the presence of longer silent intervals.

Redundant connections in a network have a potential role in learning temporal sequences[59–61]. Although the aim of the present work was to obtain minimal networks for recognising patterns, some redundancy is essential in the network. For example, the extended network that recognises a pattern of length four requires two redundant connections (Figure 2c). If we remove input *A* and neurone *N*4, the four-signal network is reduced to a three-signal network with two redundant connections; an excitatory connection from input *D* to the *Switch* (*N*2) neurone and an inhibitory connection from the *Lock* (*N*3) to the *Accept* (*N*1) neurone. The redundant connection from *D* to *N*2 is required in a four-signal network to activate the switch neurone as soon as the network receives the last correct signal, while the redundant inhibitory connection from *N*3 to *N*1 prevents the *Output* neurone from spiking when the last symbol D is received in a wrong order. Although the *Output* neurone is prevented from spiking directly by the *Lock* (*N*3) neurone, this connection reduces the activity that reaches the *Output* neurone when the lock is active.

Information processing in neural systems is affected by the presence of noise. Noise in the nervous system has previously been documented to play a computational role[62–67]. We previously showed[54] that networks evolved in the presence of noise are robust to intrinsic (perturbation of parameters) and extrinsic (variation of silent intervals) disturbances. In fact, noise enables networks to maintain network states (a form of memory) when the silent interval between signals is increased during evolution. One of the key findings is that the introduction of noise during evolution simplifies the networks instead of further complicating them. The resulting networks are robust to the removal of connections. Thus, we could prune excessive connections without impairing the network's performance. As a result, the simplified (evolved and pruned) networks are more efficient and easier to understand. In contrast, the networks that evolved in the absence of noise were fragile and a slight variation of neuronal parameters or network topology can completely alter the functionality of these networks[54].

The approach of handcrafting the network topology provides insight into neural information processing and the interplay between network connectivity and functional behaviour. However, this work has several limitations that could be addressed in future studies. In particular, the maximum number of active neurones maintaining a network state increases with an increasing length of the patterns that can be recognised. As the number of connections in the network grows, their weights become more difficult to optimise. Due to this limitation, further scaling of the topology produces suboptimal networks. One possible solution to this problem is to build a larger network by connecting two or more three-signal (perfect) networks to recognise more extended patterns. However, the approach of hierarchically connecting smaller networks to build a larger network is not straightforward. We have investigated several ways to interconnect smaller networks to recognise more extended patterns of length 5 and 6, but none of them could outperform the sequential extension of the topologies presented in Figure 2. In future studies, it would be interesting to explore other possible topologies where the maximum number of active neurones (autapses) representing a network state is invariant to the pattern length.

An AdEx neurone can generate a wide range of spiking patterns in response to a step current, depending on the choice of initial parameters[49]. In this study, we are using the parameters for the simplest type of spiking pattern, that is, tonic or regular spiking[49]. A standard leaky integrate-and-fire model (LIF) can also generate this behaviour in response to a step current. In the scope of this study, we do not take advantage of the rich spiking behaviours that an AdEx neurone can have. The neurones in our networks are limited to three possible states: active (tonic spiking), intermittently active, and nonactive (quiet). We observed that an activated neurone may speed up or slow down after receiving an input signal, but we did not take these firing rate changes into account when analysing network behaviour. Identifying network states based on the spiking behaviour of a

single neurone would require longer intervals between input signals to notice the difference between their responses. Studying the state transition at the level of a single neurone in the network could result in more efficient solutions. For example, different neuronal behaviours, such as bursting, tonic, and irregular spiking, may represent distinct network states. Moreover, different spiking/bursting frequencies could also represent different network states.

Finally, in this study, we use a genetic algorithm to optimise connection weights in a handcrafted topology. Since all connections in the handcrafted topology are defined according to the state transition table and both the topology and network states are known, a more systematic approach like backpropagation could be employed instead of the genetic algorithm to optimise the weights in the network. This could be done by (i) calculating the errors between the current and the desired network states and (ii) adjusting the connection weights to reach the desired states.

Conclusions

The present study used a novel combination of evolving and handcrafting spiking neural networks to explore potential mechanisms for temporal pattern recognition in neural systems. Our work demonstrates a link between structural network connectivity and functional network behaviour. A systematic analysis of the resulting networks indicates that excitatory autaptic connections can play a key role in memory maintenance and network state transitions. We predict that the presence of autapses in a neural system implies that the system can perform temporal pattern recognition. A systematic investigation of links between autaptic connections and temporal coding should be an exciting topic for future experimental work.

Acknowledgements

Author contributions statement

All authors contributed equally to the planning of the experiments, the analysis of the results, the editing and the review of the manuscript. M.Y. conducted the experiments using a software platform developed initially by B.W. and other members of his research group.

Data availability

All data generated or analysed during this study are included in this published article.

References

1. Isaacson, J. S. Odor representations in mammalian cortical circuits. *Curr. Opin. Neurobiol.* **20**, 328–331 (2010).

2. Joris, P. & Yin, T. C. A matter of time: internal delays in binaural processing. *Trends Neurosci.* **30**, 70–78 (2007).

3. Thorpe, S., Fize, D. & Marlot, C. Speed of processing in the human visual system. *Nature* **381**, 520 (1996).

4. Johansson, R. & Birznieks, I. First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nat. neuroscience* **7**, 170–7 (2004).

5. Ahissar, E. & Arieli, A. Figuring space by time. *Neuron* **32**, 185–201 (2001).

6. Hopfield, J. J. Pattern recognition computation using action potential timing for stimulus representation. *Nature* **376**, 33–36 (1995).

7. Kayser, C., Montemurro, M. A., Logothetis, N. K. & Panzeri, S. Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. *Neuron* **61**, 597–608 (2009).

8. Gollisch, T. & Meister, M. Rapid neural coding in the retina with relative spike latencies. *science* **319**, 1108–1111 (2008).

9. VanRullen, R., Guyonneau, R. & Thorpe, S. J. Spike times make sense. *Trends Neurosci.* **28**, 1–4 (2005).

10. Kempter, R., Gerstner, W., Van Hemmen, J. & Wagner, H. Temporal coding in the sub-millisecond range: Model of barn owl auditory pathway. *Adv. neural information processing systems* **8** (1995).

11. Johansson, R. & Birznieks, I. First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nat. neuroscience* **7**, 170–7 (2004).

12. Swadlow, H. & Gusev, A. G. Receptive-field construction in cortical inhibitory interneurons. *Nat. Neurosci.* **5**, 403–404 (2002).

13. Perez-Orive, J. *et al.* Oscillations and sparsening of odor representations in the mushroom body. *Science* **297**, 359 – 365 (2002).

14. Meister, M. & Berry, M. J. The neural code of the retina. *Neuron* **22**, 435–450 (1999).

15. Heil, P. Auditory cortical onset responses revisited. i. first-spike timing. *J. neurophysiology* **77 5**, 2616–41 (1997).

16. Steveninck, R., Lewen, G., Strong, S., Koberle, R. & Bialek, W. Reproducibility and variability in neural spike trains. *Sci. (New York, N.Y.)* **275**, 1805–8 (1997).

17. Wang, X., Lu, T. & Liang, L. Temporal and rate representations of time-varying signals in auditory cortex. In *Plasticity and Signal Representation in the Auditory System*, 71–83 (Springer, 2005).

18. Bialek, W., Rieke, F., van Steveninck, R. R. d. R., Warland, D. *et al.* Reading a neural code. In *Neural Information Processing Systems*, 36–43 (1989).

19. Decharms, R. C. & Zador, A. Neural representation and the cortical code. *Annu. Rev. Neurosci.* **23**, 613–647 (2000).

20. Gerstner, W., Kempter, R., van Hemmen, J. L. & Wagner, H. A neuronal learning rule for sub-millisecond temporal coding. *Nature* **383**, 76–78 (1996).

21. Huxter, J., Burgess, N. & O'keefe, J. Independent rate and temporal coding in hippocampal pyramidal cells. *Nature* **425**, 828–832 (2003).

22. Laurent, G. Dynamical representation of odors by oscillating and evolving neural assemblies. *Trends Neurosci.* **19**, 489–496 (1996).

23. Rieke, F., Warland, D., de Ruyter van Steveninck, R. & Bialek, W. *Spikes: Exploring the Neural Code* (MIT Press, 1999).

24. Bertschinger, N. & Natschläger, T. Real-time computation at the edge of chaos in recurrent neural networks. *Neural computation* **16**, 1413–36 (2004).

25. Chialvo, D. Emergent complex neural dynamics. *Nat. Phys.* **6**, 744–750 (2010).

26. du Castel, B. Pattern activation/recognition theory of mind. *Front. computational neuroscience* **9**, 90 (2015).

27. de Paiva, G. Pattern recognition theory of mind. *arXiv preprint arXiv:0907.4509* (2009).

28. Natschläger, T. & Ruf, B. Spatial and temporal pattern analysis via spiking neurons. *Network: Comput. Neural Syst.* **9**, 319 (1998).

29. Steuber, V. & Willshaw, D. J. Adaptive leaky integrator models of cerebellar Purkinje cells can learn the clustering of temporal patterns. *Neurocomputing* **26**, 271–276 (1999).

30. Steuber, V. & Willshaw, D. A biophysical model of synaptic delay learning and temporal pattern recognition in a cerebellar purkinje cell. *J. computational neuroscience* **17**, 149–164 (2004).

31. Steuber, V., Willshaw, D. & Van Ooyen, A. Generation of time delays: Simplified models of intracellular signalling in cerebellar purkinje cells. *Network: Comput. Neural Syst.* **17**, 173–191 (2006).

32. Fiala, J. C., Grossberg, S. & Bullock, D. Metabotropic glutamate receptor activation in cerebellar purkinje cells as substrate for adaptive timing of the classically conditioned eye-blink response. *J. Neurosci.* **16**, 3760–3774 (1996).

33. Medina, J. F., Garcia, K. S., Nores, W. L., Taylor, N. M. & Mauk, M. D. Timing mechanisms in the cerebellum: testing predictions of a large-scale computer simulation. *J. Neurosci.* **20**, 5516–5525 (2000).

34. Tank, D. W. & Hopfield, J. Neural computation by concentrating information in time. *Proc. Natl. Acad. Sci.* **84**, 1896–1900 (1987).

35. Reichardt, W. Autokorrelations-auswertung als funktionsprinzip des zentralnervensystems. *Zeitschrift für Naturforschung B* **12**, 448–457 (1957).

36. Unnikrishnan, K., Hopfield, J. J. & Tank, D. W. Connected-digit speaker-dependent speech recognition using a neural network. *IEEE Transactions on Signal Process.* **39** (1991).

37. Maass, W. Networks of spiking neurons: The third generation of neural network models. *Neural Networks* **10**, 1659–1671 (1997).

38. Ponulak, F. & Kasiński, A. Introduction to spiking neural networks: Information processing, learning and applications. *Acta neurobiologiae experimentalis* **71**, 409–33 (2011).

39. Yaqoob, M. & Wróbel, B. Very small spiking neural networks evolved for temporal pattern recognition and robust to perturbed neuronal parameters. In *Artificial Neural Networks and Machine Learning – ICANN*, 322–331 (2018).

40. Yaqoob, M. & Wróbel, B. Robust very small spiking neural networks evolved with noise to recognize temporal patterns. In *ALIFE 2018: Proceedings of the 2018 Conference on Artificial Life – MIT Press*, 665–672 (2018).

41. Maex, R. & Steuber, V. The first second: Models of short-term memory traces in the brain. *Neural Networks* **22**, 1105–1112 (2009).

42. Van Der Loos, H. & Glaser, E. M. Autapses in neocortex cerebri: synapses between a pyramidal cell's axon and its own dendrites. *Brain Res.* **48**, 355 – 360 (1972).

43. Bacci, A. & Huguenard, J. Enhancement of spike-timing precision by autaptic transmission in neocortical inhibitory interneurons. *Neuron* **49**, 119–130 (2006).

44. wei Fan, H., Wang, Y., Wang, H., Lai, Y.-C. & Wang, X. Autapses promote synchronization in neuronal networks. *Sci. Reports* **8** (2017).

45. Kim, R. & Sejnowski, T. Strong inhibitory signaling underlies stable temporal dynamics and working memory in spiking neural networks. *Nat. Neurosci.* **24**, 1–11 (2021).

46. Yilmaz, E., Ozer, M., Baysal, V. & Perc, M. Autapse-induced multiple coherence resonance in single neurons and neuronal networks. *Sci. Reports* **6** (2016).

47. Wiles, L., Gu, S., Pasqualetti, F., Bassett, D. & Meaney, D. Autaptic connections shift network excitability and bursting. *Sci. Reports* **7** (2016).

48. Yaqoob, M., Steuber, V. & Wróbel, B. The importance of self-excitation in spiking neural networks evolved to recognize temporal patterns. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Theoretical Neural Computation*, 758–771 (2019).

49. Naud, R., Marcille, N., Clopath, C. & Gerstner, W. Firing patterns in the adaptive exponential integrate-and-fire model. *Biol. Cybern.* **99** (2008).

50. Joachimczak, M. & Wróbel, B. Open ended evolution of 3d multicellular development controlled by gene regulatory networks. In *Artificial Life XIII: Proceedings of the Thirteenth International Conference on the Simulation and Synthesis of Living Systems*, 67–74 (MIT Press, Cambridge, MA, 2012).

51. Wróbel, B., Abdelmotaleb, A. & Joachimczak, M. Evolving networks processing signals with a mixed paradigm, inspired by gene regulatory networks and spiking neurons. In *International Conference on Bio-Inspired Models of Network, Information, and Computing Systems*, 135–149 (Springer, 2012).

52. Burnstock, G. Autonomic neurotransmission: 60 years since sir henry dale. *Annu. Rev. Pharmacol. Toxicol.* **49**, 1–30 (2009).

53. Sipser, M. *Introduction to the Theory of Computation* (International Thomson Publishing, 1996).

54. Yaqoob, M. & Wróbel, B. Very small spiking neural networks evolved to recognize a pattern in a continuous input stream. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI) – IEEE*, 3496–3503 (2017).

55. Seung, H. S., Lee, D., Reis, B. & Tank, D. The autapse: A simple illustration of short-term analog memory storage by tuned synaptic feedback. *J. Comput. Neurosci.* **9**, 171–185 (2004).

56. Herrmann, C. & Klaus, A. Autapse turns neuron into oscillator. *Int. J. Bifurc. Chaos* **14**, 623–633 (2004).

57. Wang, H., Ma, J., Chen, Y. & Chen, Y. Effect of an autapse on the firing pattern transition in a bursting neuron. *Commun. Nonlinear Sci. Numer. Simul.* **19**, 3242 – 3254 (2014).

58. Yilmaz, E. & Ozer, M. Delayed feedback and detection of weak periodic signals in a stochastic hodgkin–huxley neuron. *Phys. A: Stat. Mech. its Appl.* **421**, 455 – 462 (2015).

59. Zakharov, A. V. Information capacity of a neural network with redundant connections between neurons. In *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 1, 16–19 (2017).

60. Maass, W., Natschläger, T. & Markram, H. Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Comput.* **14**, 2531–2560 (2002).

61. Burgess, N. & Hitch, G. Memory for serial order: A network model of the phonological loop and its timing. *Psychol. Rev.* **106** (1999).

62. Florian, R. V. Biologically inspired neural networks for the control of embodied agents. *Cent. for Cogn. Neural Stud. (Cluj-Napoca, Romania), Tech. Rep. Coneural-03-03* (2003).

63. Faisal, A. A., Selen, L. P. & Wolpert, D. M. Noise in the nervous system. *Nat. Rev. Neurosci.* **9**, 292–303 (2008).

64. Jacobson, G. *et al.* Subthreshold voltage noise of rat neocortical pyramidal neurones. *The J. Physiol.* **564**, 145–60 (2005).

65. Destexhe, A. & Paré, D. Impact of network activity on the integrative properties of neocortical pyramidal neurons in vivo. *J. Neurophysiol.* **81**, 1531–1547 (1999).

66. Stacey, W. & Durand, D. Stochastic resonance improves signal detection in hippocampal neurons. *J. Neurophysiol.* **83**, 1394–402 (2000).

67. Wiesenfeld, K. & Moss, F. Stochastic resonance and the benefits of noise: From ice ages to crayfish and squids. *Nature* **373**, 33–36 (1995).