# Evolving spiking neural networks for spatio and spectro-temporal pattern recognition

Kasabov, N

Abstract: This paper provides a survey on the evolution of the evolving connectionist systems (ECOS) paradigm, from simple ECOS introduced in 1998 to evolving spiking neural networks (eSNN) and neurogenetic systems. It presents methods for their use for spatio-and spectro temporal pattern recognition. Future directions are highlighted.

DOI: https://doi.org/10.1109/IS.2012.6335110



Originally published at:
Kasabov, N (2012). Evolving spiking neural networks for spatio and spectro-temporal pattern recognition. In: 2012 6th IEEE International Conference Intelligent Systems (IS), Sofia, Bulgaria, 6 September 2012 - 8 September 2012, 27-32.
DOI: https://doi.org/10.1109/IS.2012.6335110

Paper accepted for **IEEE Proceedings** of the 2012 6th Int. IEEE Conference *Intelligent Systems* ,
September 6-8, 2012, Sofia, Bulgaria

# Evolving Spiking Neural Networks for Spatio- and Spectro-Temporal Pattern Recognition

Nikola Kasabov, FIEEE, FRSNZ

Knowledge Engineering and Discovery Research Institute - KEDRI,
Auckland University of Technology,
and Institute for Neuroinformatics - INI, ETH and University of Zurich
nkasabov@aut.ac.nz, www.kedri.info, ncs.ethz.ch/projects/evospike

## Abstract

This paper provides a survey on the evolution of the evolving connectionist systems (ECOS) paradigm, from simple ECOS introduced in 1998 to evolving spiking neural networks (eSNN) and neurogenetic systems. It presents methods for their use for spatio-and spectro temporal pattern recognition. Future directions are highlighted.

**Keywords** Evolving Connectionist Systems (ECOS); Evolving Spiking Neural Networks (eSNN); Computational Neurogenetic Systems (CNGS); spatio-temporal pattern recognition; spectro-temporal pattern recognition; quantum inspired SNN.

## 1. Evolving Connectionist Systems (ECOS)

Evolving connectionist systems (ECOS) are modular connectionist based systems that evolve their structure and functionality in a continuous, self-organised, on-line, adaptive, interactive way from incoming information [22-25]. They can process both data and knowledge in a supervised and/or unsupervised way [20]. ECOS learn local models from data through clustering of the data and associating a local output function for each cluster represented in a connectionist structure. They can learn incrementally single data items or chunks of data and also incrementally change their input features [40,41]. Elements of ECOS have been proposed as part of the classical NN models, such as SOM, RBF, FuzzyARTMap, Growing neural gas, neuro-fuzzy systems, RAN (see [25]). Other ECOS models, along with their applications, have been reported in [27,20].

Here we will briefly illustrate the concepts of ECOS on two implementations, EFuNN [23] and DENFIS [24]. Examples of EFuNN and DENFIS are shown in Fig. 1 and Fig. 2 respectively. In ECOS clusters of data are created based on similarity between data samples either in the input space (this is the case in some of the ECoS models, *e.g.* the dynamic neuro-fuzzy inference system DENFIS [24], or in both the input and output space (this is the case *e.g.* in the EFuNN models [23]. Samples that have a distance to an existing node (cluster center, rule node) less than a certain threshold are allocated to the same cluster. Samples that do not fit into existing clusters, form new clusters. Cluster centers are continuously adjusted according to new data samples, and new clusters are created incrementally.

ECOS learn from data and automatically create or update a local output function for each cluster, the function being represented in the connection weights, thus creating local models. Different functionalities of ECOS have been introduced and studied in [21,25,27] such as: on-line or off-line neuron aggregation and pruning; "sleep"-learning; fuzzy rule extraction and rule adaptation; using SVM as local models; evolutionary optimisation of features and parameters of ECOS; and others.

Applications of ECOS span across domain areas [25,27 ], *e.g.* bioinformatics; speech and image processing; multimodal audio-visual information processing; ecological modelling; robot control; personalised modelling [28,32]; neuroinformatics and brain study; financial applications [54]. Software environment Neucom (www.theneucom.com) has been developed to include some of the ECOS methods. A detailed survey on ECOS can be found in [27,53].

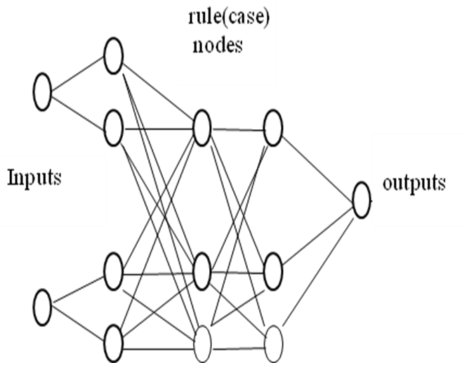



Fig.1. An example of EFuNN model [25]

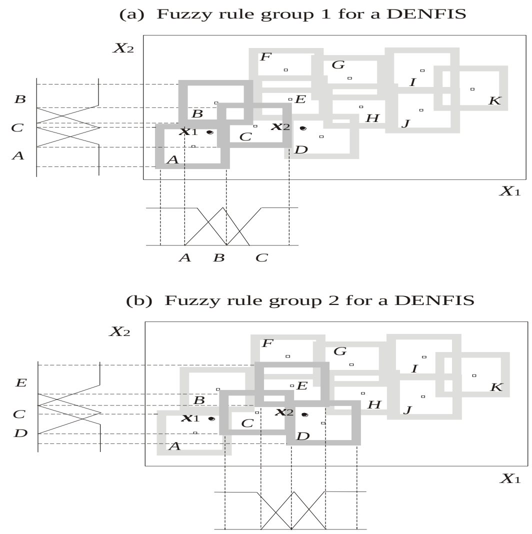



Fig.2. An example of DENFIS local modelling [24]

A special direction of ECOS was transductive reasoning and personalised modelling. Instead of building a set of local models (e.g. prototypes) to cover the whole problem space and then use these models to classify/predict any new input vector, in transductive modelling for every new input vector a new model is created based on selected nearest neighbour vectors from the available data. Such ECOS models are NFI and TWNFI [27,28]. In TWNFI for every new input vector the neighbourhood of closets data vectors is optimised using both the distance between the new vector and the neighbouring ones and the weighted importance of the input variables so that the error of the model is minimised in the neighbourhood area.

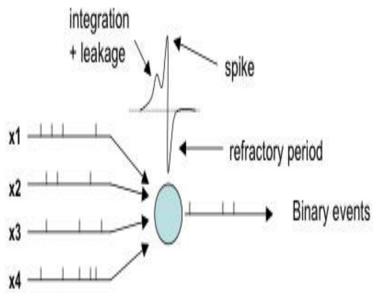
While the classical ECOS use a simple McCulloch and Pitts model of a neuron, the further developed evolving spiking neural network (eSNN) architecture use a spiking neuron model, but same or similar ECOS principles and applications are applicable.

## 2 Evolving Spiking Neural Networks

Based on the ECOS principles, an evolving spiking neural network architecture (eSNN) was proposed in [29,27,55] which was initially designed as a visual pattern recognition system. The first eSNNs were based on the Thorpe's neural model [49], in which the importance of early spikes (after the onset of a certain stimulus) is boosted called rank-order coding and learning. Synaptic plasticity is employed by a fast supervised one-pass learning algorithm that is explained as part of this section. Following eSNN architectures used both rank-order and time-based learning methods to account for spatio-temporal data [10].

### 2.1. Single spiking neuron models

A single biological neuron and the associated synapses is a complex information processing machine that involves short term information processing, long term information storage, and evolutionary information stored as genes in the nucleus of the neuron. Some of the-state-of-the-art models of a spiking neuron include: early models by Hodgkin and Huxley [16] 1952; more recent models by Maas, Gerstner, Kistler, Izhikevich and others, e.g.: Spike Response Models (SRM); Integrate-and-Fire Model (fig.3); Izhikevich models; adaptive IFM; probabilistic IFM (fig.4) [6,7,12,13,15,33,34,35,38,31].



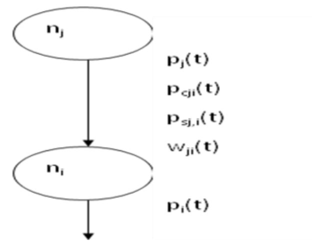




Fig.3. The structure of the LIFM

Fig.4. A probabilistic spiking neuron model [31]

An important part of a model of a neuron is the model of the synapses. Most of the neuronal models assume scalar synaptic efficacy parameters that are subject to learning, either on-line or off-line (batch mode). The total post synaptic potential PSPi(t) of the probabilistic spiking neuron $n_i$ is calculated using the following formula [31]:

$$PSP_i(t) = \sum_{p=t_0,.,t} \left(\sum_{j=1,..,m} e_j f_1(p_{cj,i}(t-p)) f_2(p_{sj,i}(t-p)) w_{j,i}(t) + \eta(t-t_0)\right) \quad (1)$$

where: $e_j$ is 1, if a spike has been emitted from neuron $n_{j,}$ and 0 otherwise; $f_1(p_{cj,i}(t))$ is 1 with a probability $p_{cji}(t)$, and 0 otherwise; $f_2(p_{sj,i}(t))$ is 1 with a probability $p_{sj,i}(t)$, and 0 otherwise; $t_0$ is the time of the last spike emitted by $n_i$; $\eta(t-t_0)$ is an additional term representing decay in the $PSP_i$.

A neurogenetic model of a neuron is proposed in [29] and studied in [5,30]. It utilises information about how some proteins and genes affect the spiking activities of a neuron such as *fast excitation, fast inhibition, slow excitation, and slow inhibition*. An important part of the model is a dynamic gene/protein regulatory network (GRN) model of the dynamic interactions between genes/proteins over time that affect the spiking activity of the neuron.

### 2.2. Learning and memory in a spiking neuron

A learning process has an effect on the synaptic efficacy of the synapses connected to a spiking neuron and on the information that is memorized. Memory can be:

- Short-term, represented as a changing PSP and temporarily changing synaptic efficacy;
- Long-term, represented as a stable establishment of the synaptic efficacy;
- Genetic (evolutionary), represented as a change in the genetic code and the gene/ protein expression level as a result of the above short-term and long term memory changes and evolutionary processes.

Learning in SNN can be: Unsupervised - there is no desired output signal provided; Supervised – a desired output signal is provided; Semi-supervised. Several biologically plausible learning rules have been introduced so far, depending on the type of the information presentation: Rate-order learning, that is based on the average spiking activity of a neuron over time [ ]; Temporal learning, that is based on precise spike times [17,12,14,48]; Rank-order learning, that takes into account the order of spikes across all synapses connected to a neuron [49,50]. Rate-order information representation is typical for cognitive information processing. Temporal spike learning is observed in the auditory, the visual and the motor control information processing of the brain. Its use in neuro-prosthetics is essential, along with applications for a fast, real-time recognition and control of sequence of related processes.

Temporal coding accounts for the precise time of spikes and has been utilised in several learning rules, most popular being Spike-Time Dependent Plasticity (STDP) [48] and Spike Dependent Synaptic Plasticity (SDSP) [12]. Temporal coding of information in SNN makes use of the exact time of spikes (e.g. in milliseconds). Every spike matters and its time matters too.

### 2.3. Evolving Spiking Neural Networks

eSNN evolve their structure and functionality in an on-line manner, from incoming information. For every new input pattern, a new neuron is dynamically allocated and connected to the input neurons (feature neurons). The neurons connections are established for the neuron to recognise this pattern (or a similar one) as a positive example. The neurons represent centres of clusters in the space of the synaptic weights. In some implementations similar neurons are merged [25,27]. That makes it possible to achieve a very fast learning in an eSNN (only one pass may be necessary), both in a supervised and in an unsupervised mode (fig. 5).

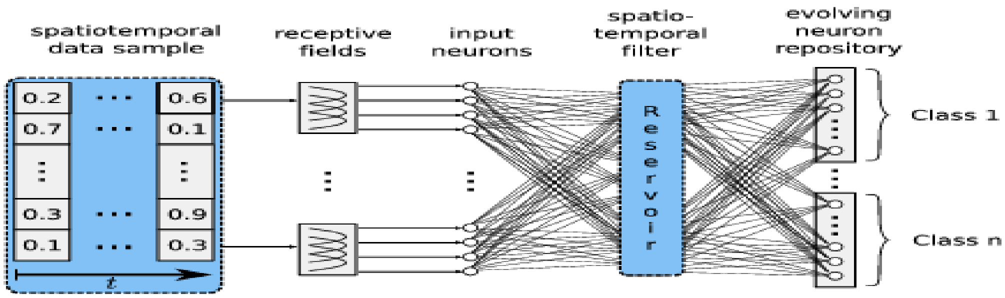



Fig.5. eSNN used as a spatio-temporal pattern classifier. A reservoir filter is used to store a larger 'chunks' of spatio-temporal input data streams (from [26, ]).

## 3. eSNN for Spatio/Spectro-Temporal Pattern Recognition (STPR)

**3.1. Spatio- and specro temporal data and the pattern recognition problem**

Most problems in nature require spatio- or/and spectro-temporal data (SSTD) that include measuring spatial or/and spectral variables over time. It is important for a computational model to capture and learn *whole* spatio- and spectro-temporal patterns from data streams in order to predict most accurately future events for new input data. Examples of problems involving SSTD are: brain cognitive state evaluation based on spatially distributed EEG electrodes; fMRI data; moving object recognition from video data [26]; spoken word recognition based on spectro-temporal audio data; evaluating risk of disease, e.g. heart attack; evaluating response of a disease to treatment based on clinical and environmental variables, e.g. stroke [1]; prognosis of outcome of cancer; modelling the progression of a neuro-degenerative disease, such as Alzheimer's Disease [30]; modelling and prognosis of the establishment of invasive species in ecology [27]. The prediction of events in geology, astronomy, economics and many other areas also depend on accurate SSTD modelling.

The commonly used models for dealing with temporal information based on Hidden Markov Models (HMM) [43] and traditional artificial neural networks (ANN) [20] have limited capacity to achieve the integration of complex and long temporal spatial/spectral components because they usually either ignore the temporal dimension or over-simplify its representation. A new trend in machine learning is currently emerging and is known as *deep machine learning* [4].

The human brain has the amazing capacity to learn and recall patterns from SSTD at different time scales, ranging from milliseconds to years and possibly to millions of years (e.g. genetic information, accumulated through evolution). Thus the brain is the ultimate inspiration for the development of new machine learning techniques for STPR. Indeed, brain-inspired Spiking Neural Networks (SNN) have the potential to learn SSTD by using trains of spikes (binary temporal events) transmitted among spatially located synapses and neurons. Both spatial and temporal information can be encoded in an SNN as locations of synapses and neurons and time of their spiking activity respectively. Spiking neurons send spikes via connections that have a complex dynamic behaviour, collectively forming an SSTD memory.

**3. 2. STPR in a single neuron**

In contrast to the distributed representation theory and to the widely popular view that a single neuron cannot do much, some recent results showed that a single neuronal model can be used for complex STPR. A single LIF neuron, for example, with simple synapses can be trained with the STDP unsupervised learning rule to discriminate a repeating pattern of synchronised spikes on certain synapses from noise. Single neuron models have been introduced for STPR, such as: Temportron [14];

Chronotron [11]; ReSuMe [42]; SPAN [36,37]. Each of them can learn to emit a spike or a spike pattern (spike sequence) when a certain STP is recognised. Some of them can be used to recognise multiple STP per class and multiple classes [36,37].

### 3.3. The EvoSpike architecture for STPR

A general framework of eSNN for STPR is shown in fig.6. It consists of the following blocks: Input data encoding block; Machine learning block (consisting of several sub-blocks); Output block.

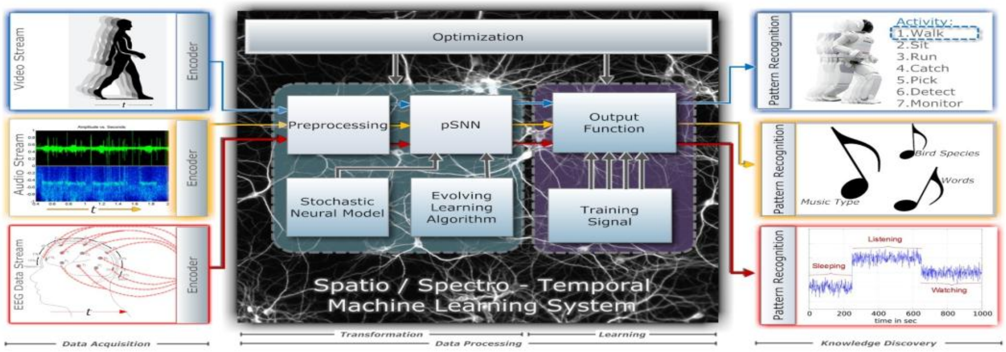



Fig.6. The EvoSpike framework for STPR (from: http://ncs.ethz.ch/projects/evospike)

In the input block continuous value input variables are transformed into spikes. Different approaches can be used: population rank coding [ ]; thresholding the input value, so that a spike is generated if the input value (e.g. pixel intensity) is above a threshold; Address Event Representation (AER) - thresholding the difference between two consecutive values of the same variable over time as it is in the artificial cochlea [ ] and artificial retina devices [ ]. The input information is entered either on-line (for on-line, real time applications) or as a batch data. The *time* of the input data is in principal different from the internal SNN *time* of information processing.

The framework from fig.6 supports the creation of a multi-modular integrated system, where different modules, consisting of different neuronal types and genetic parameters, represent different functions (e.g.: vision; sensory information processing; sound recognition; motor-control) and the whole system works in an integrated mode. Different realisation of the framework from fig.6 have been explored so far. One of them uses a reservoir filter [52] and an eSNN classifier (fig.5). Other adaptive classifiers can be explored for the classification of the reservoir state into one of the output classes, including: statistical techniques, e.g. regression techniques; MLP; nearest-neighbour techniques; incremental LDA [41]. State vector transformation, before classification can be done with the use of adaptive incremental transformation functions, such as incremental PCA [40].

Another implamentation of the framework from fig.5 is based on the computationla neurogenetic modelling (CNGM) approach and the neurogenetic types of neuron – fig.7 [5,30]. It uses as a main principle the analogy with biological facts about the relationship between spiking activity and gene/protein dynamics in order to control the learning and spiking parameters in a SNN when SSTP are learned. Biological support of this can be found in numerous publications.

A major problem with the CNGM from fig.7 is how to optimize the numerous parameters of the model. One solution could be using evolutionary computation, such as the recently proposed quantum inspired evolutionary computation technique [8,44,45] and quantum inspired PSO [39] . The latter can deal with a very large dimensional space as each quantum-bit chromosome represents the whole space, each point to certain probability. Such algorithms are faster and lead to a close solution to the global optimum in a very short time.

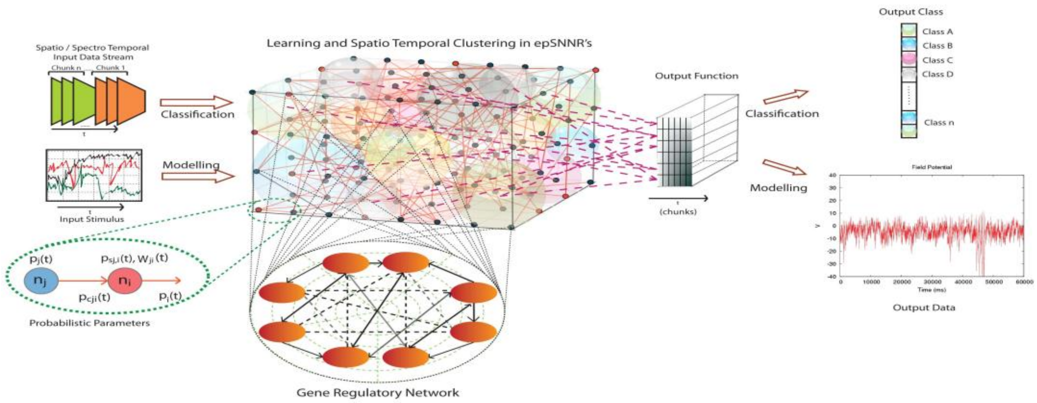



Fig.7. A schematic diagram of a CNGM framework, consisting of: input encoding module; output function for SNN state evaluation; output classifier; GRN (optional module). The framework can be used to create concrete models for STPR or data modelling (from [30]).

## 4. Current and Future Applications of eSNN for STPR

Numerous are the applications of eSNN for STPR. Here only few of them are listed:

- Moving object recognition [26];
- EEG and fMRI data modelling and pattern recognition [27, 47] (fig.8);
- Robot control through EEG signals (www.kedri.info, fig.9) and robot navigation [2];
- Sign language gesture recognition (e.g. the Brazilian sign language [ ];
- Risk of event evaluation, e.g. prognosis of establishment of invasive species [27]; stroke occurrence [1], etc.
- Cognitive and emotional robotics [3] and neuro-rehabilitation robots [30];
- Modelling the progression or the response to treatment of neurodegenerative diseases [30];
- Modelling financial and economic problems (neuro-economics) [54];
- Personalized modeling [32];
- Neuromorphic computation [18,9,51].

### ACKNOWLEDGMENT

The work on this paper is sponsored by the Knowledge Engineering and Discovery Research Institute, KEDRI (www.kedri.info) and the EU FP7 Marie Curie International Incoming Fellowship project PIIF-GA-2010-272006 *EvoSpike*, hosted by the Institute for Neuroinformatics, University of Zurich and ETH Zurich (http://ncs.ethz.ch/projects/evospike).

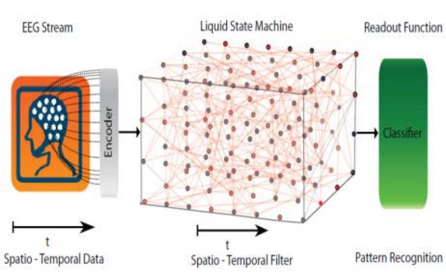



Fig.8. EEG based BCI

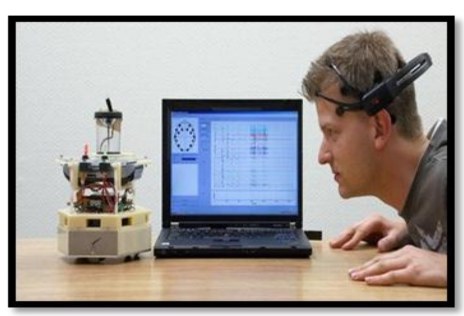


Fig.9. Robot control and navigation (www.kedri.info)

## References

1. Barker-Collo, S., Feigin, V. L., Parag, V., Lawes, C. M. M., & Senior, H. (2010). Auckland Stroke Outcomes Study. Neurology, 75(18), 1608-1616.
2. Belatreche, A., Maguire, L. P., and McGinnity, M. Advances in Design and Application of Spiking Neural Networks. Soft Comput. 11, 3, 239-248, 2006
3. Bellas, F., R. J. Duro, A. Faiña, D. Souto, MDB: Artificial Evolution in a Cognitive Architecture for Real Robots, IEEE Transactions on Autonomous Mental Development, vol. 2, pp. 340-354, 2010
4. Bengio, Y. (2009) Learning Deep Architectures for AI, Found. Trends. Mach. Learning, vol.2, 1, 1-127.
5. Benuskova, L, and N.Kasabov, Computational neuro-genetic modelling, Springer, New York, 2007.
6. Bohte, S.M., The evidence for neural information processing with precise spike-times: A survey. NATURAL COMPUTING, 3:2004, 2004.
7. Brette R., et al, (2007). Simulation of networks of spiking neurons: a review of tools and strategies. J. Comput. Neurosci. 23, 349–398.
8. Defoin-Platel, M., S.Schliebs, N.Kasabov, Quantum-inspired Evolutionary Algorithm: A multi-model EDA, IEEE Trans. Evolutionary Computation, vol.13, No.6, Dec.2009, 1218-1232
9. Delbruck, T., jAER open source project, 2007, http://jaer.wiki.sourceforge.net.
10. Dhoble, K., N.Nuntalid, G.Indivery and N.Kasabov, Online Spatio-Temporal Pattern Recognition with Evolving Spiking Neural Networks utilising Address Event Representation, Rank Order, and Temporal Spike Learning, Proc. IJCNN 2012, Brisbane, June 2012, IEEE Press
11. Florian, R.V., The chronotron: a neuron that learns to fire temporally-precise spike patterns.
12. Fusi, S., M. Annunziato, D. Badoni, A. Salamon, and D. Amit, Spike-driven synaptic plasticity: theory, simulation, VLSI implementation, Neural Computation, vol. 12, no. 10, pp. 2227–2258, 2000.
13. Gerstner, W. (1995) Time structure of the activity of neural network models, Phys. Rev 51: 738-758.
14. Gutig, R., and H. Sompolinsky. The tempotron: a neuron that learns spike timing-based decisions. Nat Neurosci, 9(3):420–428, Mar. 2006.
15. Hebb, D. (1949). The Organization of Behavior. New York, John Wiley and Sons.
16. Hodgkin, A. L. and A. F. Huxley (1952) A quantitative description of membrane current and its application to conduction and excitation in nerve. Journal of Physiology, 117: 500-544.
17. Hopfield, J., Pattern recognition computation using action potential timing for stimulus representation. Nature, 376:33–36, 1995.
18. Indiveri, G; Chicca, E; Douglas, R J (2009). Artificial cognitive systems: From VLSI networks of spiking neurons to neuromorphic cognition. Cognitive Computation, 1(2):119-127.
19. Izhikevich, E. M. (2004) Which model to use for cortical spiking neurons? IEEE TNN,15(5)1063-70.
20. Kasabov, N. Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering. Cambridge, Massachussets, MIT Press (1996) 550p
21. Kasabov N, Kim J S, Watts M, Gray A (1997) FuNN/2- A Fuzzy Neural Network Architecture for adaptive learning and Knowledge Acquisition, Information Sciences - Applications, 101(3-4): 155-175
22. Kasabov N (1998) Evolving Fuzzy Neural Networks - Algorithms, Applications and Biological Motivation. In: Yamakawa, T. and G.Matsumoto (eds) Methodologies for the conception, design and application of soft computing, World Scientific, pp 271 – 274
23. Kasabov N (2001) Evolving fuzzy neural networks for on-line supervised/unsupervised, knowledge–based learning. IEEE Trans. SMC/ B, Cybernetics 31(6): 902-918.
24. Kasabov N, Song Q (2002) DENFIS: Dynamic, evolving neural-fuzzy inference systems and its application for time-series prediction. IEEE Trans. on Fuzzy Systems, vol. 10:144 – 154
25. Kasabov N (2003) Evolving connectionist systems: Methods and Applications in Bioinformatics, Brain study and intelligent machines. Springer Verlag, London, New York, Heidelberg
26. Kasabov, N., Dhoble, K., Nuntalid, N., & Mohemmed, A. (2011). Evolving probabilistic spiking neural networks for spatio-temporal pattern recognition: A preliminary study on moving object recognition. In 18th Int. Conf. Neural Information Processing, ICONIP 2011, Shanghai, China, Springer LNCS .
27. Kasabov, N., Evolving connectionist systems: The knowledge engineering approach, Springer, 2007.
28. Kasabov, N., Global, local and personalised modelling and profile discovery in Bioinformatics: An integrated approach, Pattern Recognition Letters, Vol. 28, Issue 6 , April 2007, 673-685
29. Kasabov, N., L.Benuskova, S.Wysoski, A Computational Neurogenetic Model of a Spiking Neuron, IJCNN 2005 Conf. Proc., IEEE Press, 2005, Vol. 1, 446-451
30. Kasabov, N., R.Schliebs, H.Kojima (2011) Probabilistic Computational Neurogenetic Framework: From Modelling Cognitive Systems to Alzheimer's Disease. IEEE Trans. Autonomous Mental Development, vol.3, No.4,2011, 1-12.

31. Kasabov, N., To spike or not to spike: A probabilistic spiking neuron model, Neural Netw., 23(1), 16–19, 2010.
32. Kasabov, N., and Y. Hu (2010) Integrated optimisation method for personalised modelling and case study applications, Int. Journal of Functional Informatics and Personalised Medicine, vol.3,No.3,236-256.
33. Kistler, G., and W. Gerstner, Spiking Neuron Models - Single Neurons, Populations, Plasticity, Cambridge Univ. Press, 2002.
34. Maass, W., and A. M. Zador, Computing and learning with dynamic synapses, In: Pulsed Neural Networks, pages 321–336. MIT Press, 1999.
35. Maass, W., T. Natschlaeger, H. Markram, Real-time computing without stable states: A new framework for neural computation based on perturbations, Neural Computation, 14(11), 2531–2560, 2002.
36. Mohemmed, A., Schliebs, S., Matsuda, S., Kasabov, N., Evolving Spike Pattern Association Neurons and Neural Networks, Neurocomputing, Elsevier, in print.
37. Mohemmed, A., Schliebs, S., Matsuda, S., Kasabov, N., SPAN: Spike Pattern Association Neuron for Learning Spatio-Temporal Sequences, International Journal of Neural Systems, in print, 2012.
38. Natschläger, T., W. Maass, Spiking neurons and the induction of finite state machines, Theoretical Computer Science - Natural Computing, Vol. 287, Issue 1, pp.251-265, 2002.
39. Nuzly, H., N.Kasabov, S.Shamsuddin (2010) Probabilistic Evolving Spiking Neural Network Optimization Using Dynamic Quantum Inspired Particle Swarm Optimization, Proc. ICONIP 2010, Part I, LNCS, vol.6443.
40. Ozawa, S., S. Pang and N Kasabov, Incremental Learning of Chunk Data for On-line Pattern Classification Systems, IEEE Transactions of Neural Networks, vol.19, no.6, June 2008, 1061-1074,
41. Pang, S., S. Ozawa and N. Kasabov, Incremental Linear Discriminant Analysis for Classification of Data Streams, IEEE Trans. SMC-B, vol. 35, No. 5, 2005, 905 – 914
42. Ponulak, F., and A. Kasi´nski. Supervised learning in spiking neural networks with ReSuMe: sequence learning, classification, and spike shifting. Neural Computation, 22(2):467–510, Feb. 2010. PMID:19842989.
43. Rabiner, L.R., A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE, vol. 77, no. 2, pp. 257 - 285, 1989.
44. Schliebs, S., Kasabov, N., and Defoin-Platel, M. (2010), “On the Probabilistic Optimization of Spiking Neural Networks,” *International Journal of Neural Systems*, 20:6, pp. 481-500.
45. Schliebs,S., M.Defoin-Platel, S.Worner, N.Kasabov, Integrated Feature and Parameter Optimization for Evolving Spiking Neural Netw.: Exploring Heterogeneous Probabilistic Models, Neural Netw.,22,623-632, 2009.
46. Soltic and S. Kasabov, N. (2010), “Knowledge extraction from evolving spiking neural networks with rank order population coding ,” *International Journal of Neural Systems*, 20:6, pp. 437-445.
47. Sona, D., H.Veeramachaneni, E.Olivetti, P.Avesani, Inferring cognition from fMRI brain images, Proc. of IJCNN, 2011, IEEE Press.
48. Song, S., K. Miller, L. Abbott et al., Competitive hebbian learning through spike-timing-dependent synaptic plasticity, Nature Neuroscience, vol. 3, pp. 919–926, 2000.
49. Thorpe, S., and J. Gautrais, Rank order coding, Computational neuroscience: Trends in research, vol. 13, pp. 113–119, 1998.
50. Thorpe,S., Delorme, A., et al. (2001). Spike-based strategies for rapid processing. Neural Netw.,14(6-7), 715-25.
51. van Schaik, A., L. Shih-Chii Liu, AER EAR: a matched silicon cochlea pair with address event representation interface, in: Proc. of ISCAS - IEEE Int. Symp. Circuits and Systems, pp. 4213- 4216, vol. 5, 23-26 May 2005.
52. Verstraeten, D., B. Schrauwen, M. D’Haene, and D. Stroobandt, An experimental unification of reservoir computing methods, Neural Networks, 20(3):391 – 403, 2007.
53. Watts, M. (2009) A Decade of Kasabov’s Evolving Connectionist Systems: A Review, IEEE Trans. Systems, Man and Cybernetics- Part C: Applications and Reviews, vol.39, no.3, 253-269.
54. Widiputra, H., Pears, R., & Kasabov, N. (2011). Multiple time-series prediction through multiple time-series relationships profiling and clustered recurring trends. Springer LNAI 6635 (PART 2), 161-172.
55. Wysoski, S., L. Benuskova, N. Kasabov, Evolving spiking neural networks for audiovisual information processing, Neural Networks, vol 23, 7, pp 819-835, 2010.