

40. Evolving Connectionist Systems: From Neuro-Fuzzy-, to Spiking- and Neuro-Genetic

Nikola Kasabov

This chapter follows the development of a class of neural networks (NN) called evolving connectionist systems (ECOS). The term *evolving* is used here in its meaning of *unfolding, developing, changing, revealing* (according to the Oxford dictionary) rather than *evolutionary*. The latter represents processes related to populations and generations of them. An ECOS is a neural network-based model that evolves its structure and functionality through incremental, adaptive learning and self-organization during its *lifetime*. In principle, it could be a simple NN or a *hybrid* connectionist system. The latter is a system based on neural networks that also integrate other computational principles, such as linguistically meaningful explanation features of fuzzy rules, optimization techniques for structure and parameter optimization, quantum-inspired methods, and gene regulatory networks. The chapter includes definitions and examples of ECOS such as: evolving neuro-fuzzy and hybrid systems; evolving spiking neural networks, neurogenetic systems, quantum-inspired systems, which are all discussed from the point of view of the structural and functional development of a connectionist-based model and the knowledge that it represents. Applications for knowledge engineering across domain areas, such as in bioinformatics, brain study, and intelligent machines are presented.

40.1	Principles of Evolving Connectionist Systems (ECOS)	771
40.2	Hybrid Systems and Evolving Neuro-Fuzzy Systems	772
40.2.1	Hybrid Systems.....	772
40.2.2	Evolving Neuro-Fuzzy Systems....	773
40.2.3	From Local to Transductive (Individualized) Learning and Modeling	775
40.2.4	Applications.....	775
40.3	Evolving Spiking Neural Networks (eSNN)	775
40.3.1	Spiking Neuron Models	775
40.3.2	Evolving Spiking Neural Networks (eSNN)	775
40.3.3	Extracting Fuzzy Rules from eSNN	777
40.3.4	eSNN Applications	777
40.4	Computational Neuro-Genetic Modeling (CNGM)	778
40.4.1	Principles	778
40.4.2	The NeuroCube Framework.....	778
40.4.3	Quantum-Inspired Optimization of eSNN and CNGM.....	779
40.4.4	Applications of CNGM.....	779
40.5	Conclusions and Further Directions	779
	References	780

40.1 Principles of Evolving Connectionist Systems (ECOS)

Everything in Nature evolves, develops, unfolds, reveals, and changes in time. The brain is probably the ultimate evolving system, which develops during a lifetime, based on genetic information (Nature) and learning from the environment (nurture). Inspired by information principles of the developing brain, ECOS are adaptive, incremental learning and knowledge representation systems that evolve their structure and func-

tionality from incoming data through interaction with the environment, where in the core of a system is a connectionist architecture that consists of neurons (information processing units) and connections between them [40.1]. An ECOS is a system based on neural networks and the use of also other techniques of computational intelligence (CI), which operates continuously in time and adapts its structure and functionality through

continuous interaction with the environment and with other systems. The adaptation is defined through:

1. A set of evolving rules.
2. A set of parameters (*genes*) that are subject to change during the system operation.
3. An incoming continuous flow of information, possibly with unknown distribution.
4. Goal (rationale) criteria (also subject to modification) that are applied to optimize the performance of the system over time.

ECOS learning algorithms are inspired by brain-like information processing principles, e.g.:

1. They evolve in an open space, where the dimensions of the space can change.
2. They learn via incremental learning, possibly in an on-line mode.

3. They learn continuously in a lifelong learning mode.
4. They learn both as individual systems and as an evolutionary population of such systems.
5. They use constructive learning and have evolving structures.
6. They learn and partition the problem space locally, thus allowing for a fast adaptation and tracing the evolving processes over time.
7. They evolve different types of knowledge representation from data, mostly a combination of memory-based and symbolic knowledge.

Many methods, algorithms, and computational intelligence systems have been developed since the conception of ECOS and many applications across disciplines. This chapter will review only the fundamental aspects of some of these methods and will highlight some principal applications.

40.2 Hybrid Systems and Evolving Neuro-Fuzzy Systems

40.2.1 Hybrid Systems

A hybrid computational intelligent system integrates several principles of computational intelligence to enhance different aspects of the performance of the system. Here we will discuss only hybrid connectionist systems that integrate artificial neural networks (NN) with other techniques utilizing the adaptive learning features of the NN.

Early hybrid connectionist systems combined NN with rule-based systems such as production rules [40.3]

or predicate logic [40.4]. They utilized NN modules for a lower level of information processing and rule-based systems for reasoning and explanation at a higher level.

The above principle is applied when fuzzy rules are used for higher-level information processing and for approximate reasoning [40.5–7]. These are expert systems that combine the learning ability of NN with the explanation power of linguistically plausible fuzzy rules [40.8–11]. A block diagram of an exemplar system is shown in Fig. 40.1, where at a lower level a neural network (NN) module predicts the level of a stock index and at a higher level a fuzzy reasoning module combines the predicted values with some macro-economic variables representing the political and the economic situations using the following types of fuzzy rules [40.2]

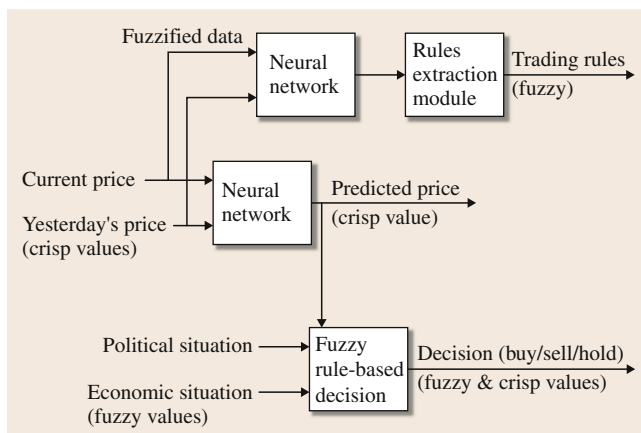


Fig. 40.1 A hybrid NN-fuzzy rule-based expert system for financial decision support (after [40.2])

$$\begin{aligned}
 & \text{IF } \langle \text{the predicted by the NN module stock value} \\
 & \quad \text{in the future is high} \rangle \text{ AND} \\
 & \quad \langle \text{the economic situation is good} \rangle \text{ AND} \\
 & \quad \langle \text{the political situation is stable} \rangle \\
 & \text{THEN } \langle \text{buy stock} \rangle . \quad (40.1)
 \end{aligned}$$

Along with the integration of NN and fuzzy rules for a better decision support, the system from Fig. 40.1 includes an NN module for extracting recent rules form data that can be used by experts to analyze the dy-

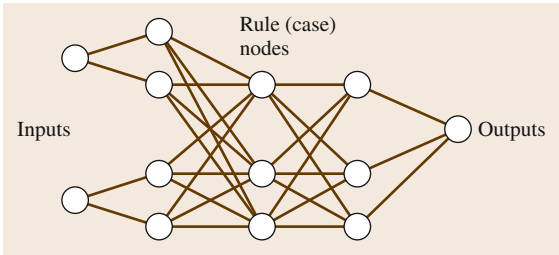


Fig. 40.2 A simple, feedforward EFuNN structure. The rule nodes evolve from data to capture cluster centers in the input space, while the output nodes evolve local models to learn and approximate the data in each of these clusters

namics of the stock and to possibly update the trading fuzzy rules in the fuzzy rule-based module. This NN module uses a fuzzy neural (FNN) network for the rule extraction.

Fuzzy neural networks (FNN) integrate NN and fuzzy rules into a single neuronal model tightly coupling learning and fuzzy reasoning rules into a connectionist structure. One of the first FNN models was initiated by Yamakawa and other Japanese scientists and promoted at a series of IIZUKA conferences in Japan [40.12, 13]. Many models of FNNs were developed based on these principles [40.2, 14, 15].

40.2.2 Evolving Neuro-Fuzzy Systems

The evolving neuro-fuzzy systems further extended the principles of hybrid neuro-fuzzy systems and the FNN, where instead of training a fixed connectionist structure,

the structure and its functionality evolve from incoming data, often in an on-line, one-pass learning mode. This is the case with evolving connectionist systems (ECOS) [40.1, 16–19].

ECOS are modular connectionist-based systems that evolve their structure and functionality in a continuous, self-organized, on-line, adaptive, and interactive way from incoming information [40.17]. They can process both data and knowledge in a supervised and/or unsupervised way. 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 [40.18].

Elements of ECOS have been proposed as part of the early, classical NN models, such as Kohonen’s self organising maps (SOM) [40.20], radial basis function(RBF) [40.21], FuzyARTMap [40.22] by Carpenter et al. and Fritzke’s growing neural gas [40.23], Platt’s resource allocation networks (RAN) [40.24].

Some principles of ECOS are:

- Neurons are created (evolved) and allocated as centers of (fuzzy) data clusters. Fuzzy clustering, as a means to create local knowledge-based systems, was stimulated by the pioneering work of Bezdek, Yager and Filev [40.27–30].
- Local models are evolved and updated in these clusters.

Here we will briefly illustrate the concepts of ECOS on two implementations: evolving fuzzy neu-

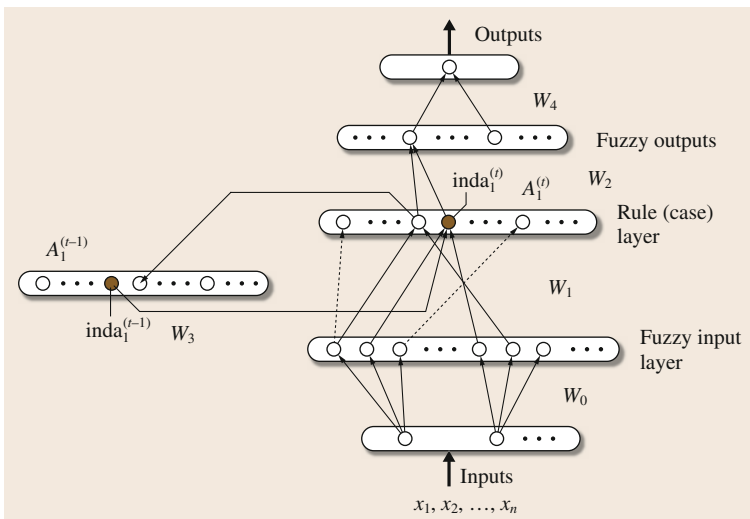


Fig. 40.3 An EFuNN structure with feedback connections (after [40.16])

tral networks (EFuNN) [40.16] and dynamic neuro-fuzzy inference systems (DENFIS) [40.25]. Examples of EFuNN are shown in Figs. 40.2 and 40.3 and of

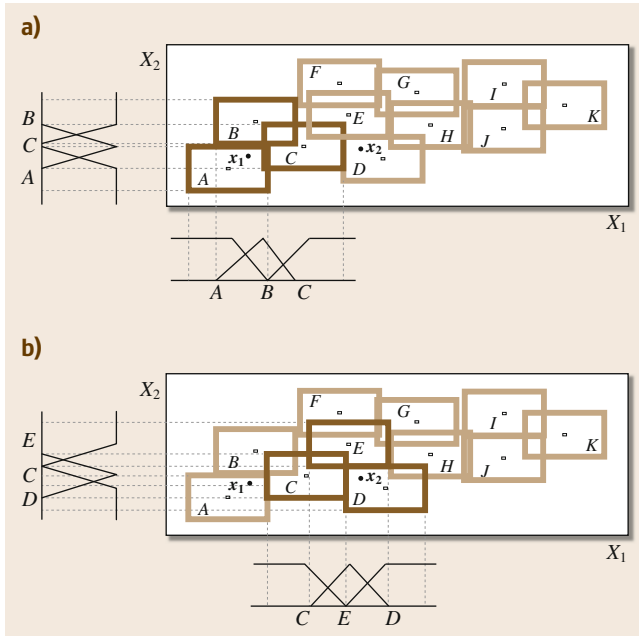


Fig. 40.4a,b Learning in DENFIS uses the evolving clustering method illustrated on a simple example of 2 inputs and 1 output and 11 data clusters evolved. The recall of the DENFIS for two new input vectors x_1 and x_2 is illustrated with the use of the 3 closest clusters to the new input vector (after [40.25]). (a) Fuzzy role group 1 for a DENFIS. (b) Fuzzy role group 2 for a DENFIS

DENFIS in Figs. 40.4 and 40.5. In ECOS, clusters of data are created (evolved) based on similarity between data samples (input vectors) either in the input space (this is the case in some of the ECOS models, e.g., DENFIS), or in both the input and output space (this is the case, e.g., in the EFuNN models). Samples that have a distance to an existing node (cluster center, rule node, neuron) less than a certain threshold are allocated to the same cluster. Samples that do not fit into existing clusters, form (generate, evolve) new clusters. Cluster centers are continuously adjusted according to new data samples, others are created incrementally. ECOS learn from data and automatically create or update a *local* (fuzzy) model/function in each cluster, e.g.,

$$\begin{aligned} & \text{IF } \langle \text{data is in a (fuzzy) cluster } C_i \rangle \\ & \text{THEN } \langle \text{the model is } F_i \rangle, \end{aligned} \tag{40.2}$$

where F_i can be a fuzzy value, a linear or logistic regression function (Fig. 40.5), or an NN model [40.25].

ECOS utilize evolving clustering methods. There is no fixed number of clusters specified a priori, but clusters are created and updated incrementally. Other ECOS that use this principle are: evolving self-organized maps (ESOM) [40.17], evolving classification function [40.18, 26], evolving spiking neural networks (Sect. [40.4]).

As an example, the following are the major steps for the training and recall of a DENFIS model:

Training:

1. Create or update a cluster from incoming data.
2. Create or update a Takagi–Sugeno fuzzy rule for each cluster:

$$\begin{aligned} & \text{IF } x \text{ is in cluster } C_j \text{ THEN } y_j = f_j(x), \\ & \text{where: } y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q. \end{aligned}$$

The function coefficients are incrementally updated with every new input vector or after a chunk of data. Recall – fuzzy inference for a new input vector:

1. For a new input vector $x = [x_1, x_2, \dots, x_q]$ DENFIS chooses m fuzzy rules from the whole fuzzy rule set for forming a current inference system.
2. The inference result is

$$y = \frac{\sum_{i=1,m} [\omega_i f_i(x_1, x_2, \dots, x_q)]}{\sum_{i=1,m} \omega_i}, \tag{40.3}$$

where i is the index of one of the m closest to the new input vector x clusters, $\omega_i = 1 - d_i$ is the weighted distance between this vector the cluster center, $f_i(x)$ is the calculated output for x according to the local model f_i for cluster i .

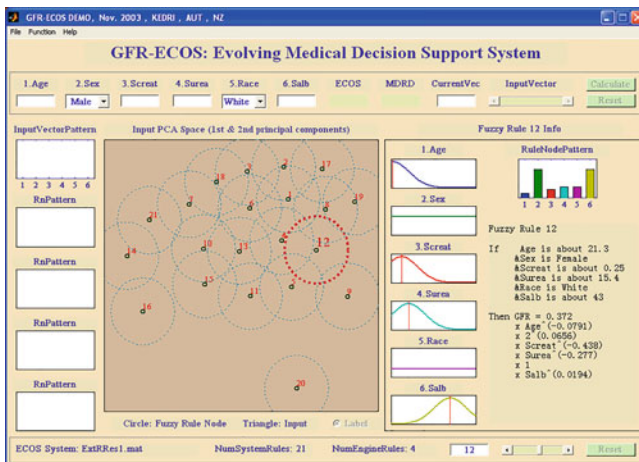


Fig. 40.5 An example of the DENFIS model (after [40.26]) for medical renal function evaluation

40.2.3 From Local to Transductive (Individualized) Learning and Modeling

A special direction of ECOS is transductive reasoning and personalized modeling. Instead of building a set of local models f_i (e.g., prototypes) to cover the whole problem space and then using these models to classify/predict any new input vector, in transductive modeling for every new input vector x a new model f_x is created based on selected nearest neighbor vectors from the available data. Such ECOS models are neuro-fuzzy inference systems (NFI) [40.31] and the transductive weighted neuro-fuzzy inference system (TWNFI) [40.32]. In TWNFI for every new input vector the neighborhood of the closest data vectors is optimized using both the distance between the new vector and the neighboring ones and the weighted importance of the input variables, so that the error of the model is minimized in the neighborhood area [40.33]. TWNFI is a further development of the weighted-weighted nearest neighbor method (WWKNN) proposed in [40.34]. The output for a new input vector is calculated based on the outputs of the k -nearest neighbors, where the weighting is based on both distance and a priori calculated importance for each variable using a ranking method such as signal-to-noise ratio or the t -test.

Other ECOS were been developed as improvements of EFuNN, DENFIS, or other early ECOS models by *Ozawa et al.* and *Watts* [40.35–37], including ensembles of ECOS [40.38]. A similar approach to ECOS

was used by *Angelov* in the development of the (ETS) models [40.39].

40.2.4 Applications

ECOS have been applied to problems across domain areas. It is demonstrated that local incremental learning or transductive learning are superior when compared to global learning models and when compared in terms of accuracy and new knowledge obtained. A review of ECOS applications can be found in [40.26]. The applications include:

- Medical decision support systems (Fig. 40.5)
- Bioinformatics, e.g., [40.40]
- Neuroinformatics and brain study, e.g., [40.41]
- Evolvable robots, e.g., [40.42]
- Financial and economic decision support systems, e.g., [40.43]
- Environmental and ecological modeling, e.g., [40.44]
- Signal processing, speech, image, and multimodal systems, e.g., [40.45]
- Cybersecurity, e.g., [40.46]
- Multiple time series prediction, e.g., [40.47].

While classical ECOS use a simple McCulloch and Pitts model of a neuron and the Hebbian learning rule [40.48], evolving spiking neural network (eSNN) architectures use a spiking neuron model, applying the same or similar ECOS principles.

40.3 Evolving Spiking Neural Networks (eSNN)

40.3.1 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. A spiking neuron model assumes input information represented as trains of spikes over time. When sufficient input spikes are accumulated in the membrane of the neuron, the neuron's post-synaptic potential exceeds a threshold and the neuron emits a spike at its axon (Fig. 40.6a,b). Some of the state-of-the-art models of spiking neurons include: early models by *Hodgkin* and *Huxley* [40.49], and *Hopfield* [40.50]; and more recent models by *Maass*, *Gerstner*, *Kistler*, *Izhikevich*, *Thorpe* and *van Ruller* [40.51–54]. Such models are spike response models (SRMs),

the leaky integrate-and-fire model (LIFM) (Fig. 40.6), *Izhikevich* models, adaptive LIFM, and probabilistic IFM [40.55].

40.3.2 Evolving Spiking Neural Networks (eSNN)

Based on the ECOS principles, an evolving spiking neural network architecture (eSNN) was proposed in [40.26], which was initially designed as a visual pattern recognition system. The first eSNNs were based on *Thorpe's* neural model [40.54], 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. An exemplar eSNN for multimodal auditory-visual information processing on

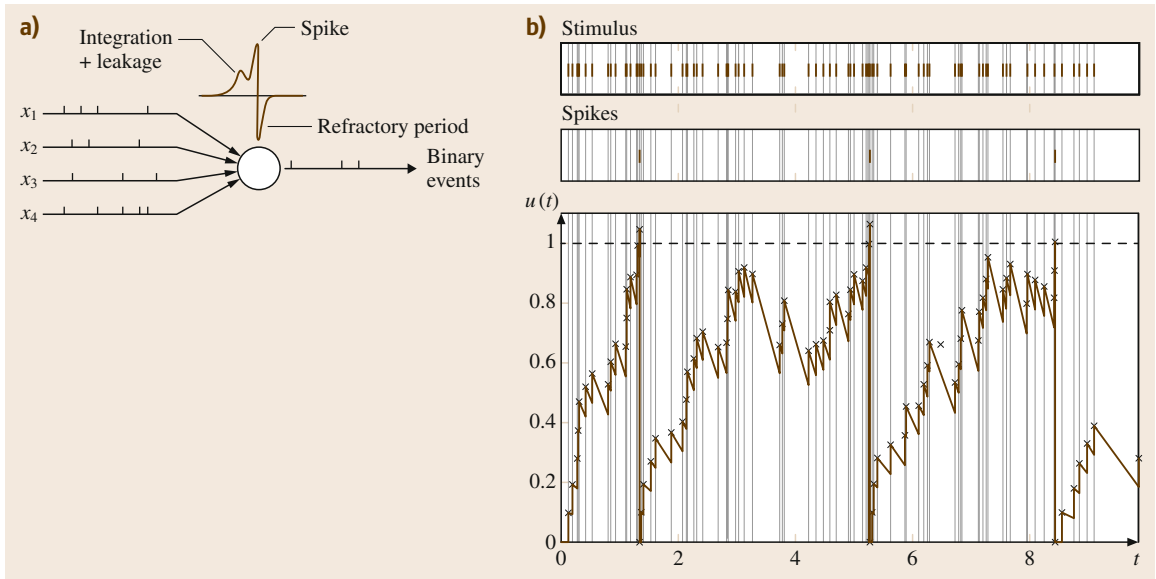


Fig. 40.6 (a) LIFM of a spiking neuron. (b) The LIFM increases its membrane potential $u(t)$ with every incoming spike at time t until the potential reaches a threshold, after which the neuron emits an output spike and its potential is reset to an initial value

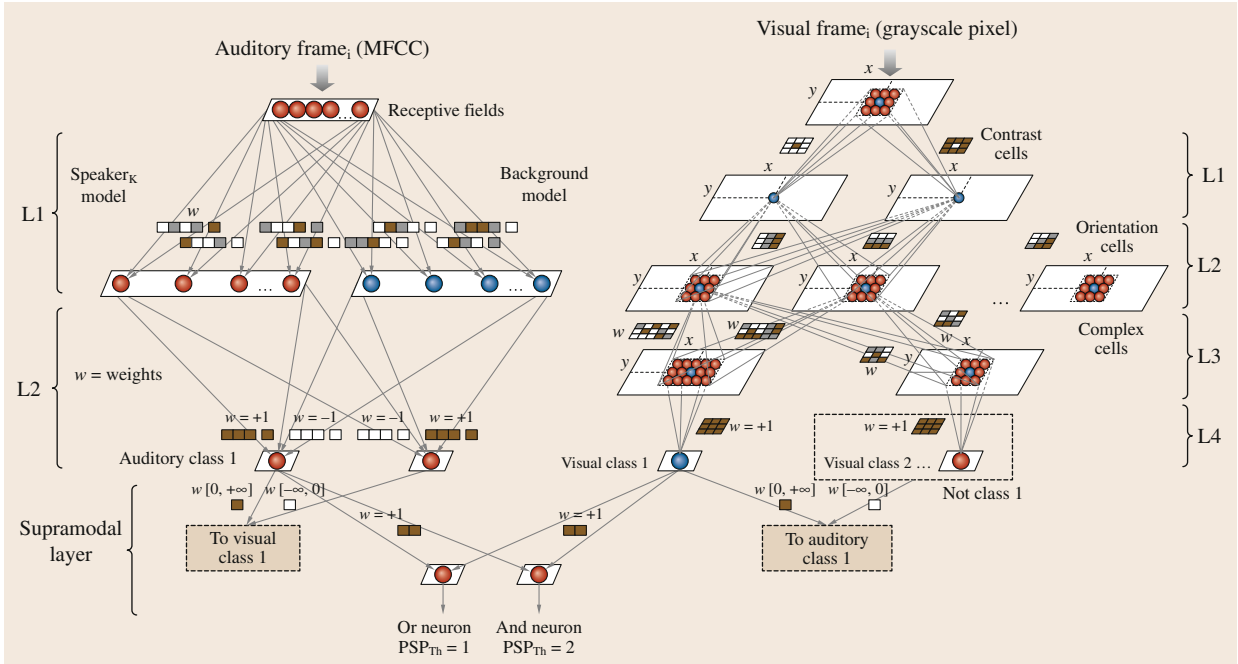


Fig. 40.7 An exemplar eSNN for multimodal auditory-visual information processing in the case study problem of speaker authentication (after [40.56])

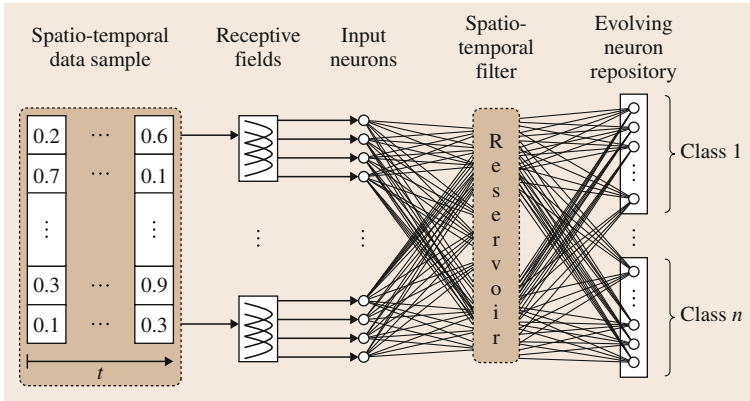


Fig. 40.8 A reservoir-based eSNN for spatio-temporal pattern classification (after [40.55])

the case study problem of speaker authentication is shown in Fig. 40.7.

Different eSNN models use different architectures. Figure 40.8 shows a reservoir-based eSNN for spatio-temporal pattern recognition where the reservoir [40.57] uses the spike-time-dependent plasticity (STDP) learning rule [40.58], and the output classifier that classifies spatio-temporal activities of the reservoir uses rank-order learning rule [40.54].

40.3.3 Extracting Fuzzy Rules from eSNN

Extracting fuzzy rules from an eSNN would make eSNN not only efficient learning models, but also knowledge-based models. A method was proposed in [40.59] and illustrated in Fig. 40.9a,b. Based on the connection weights w between the receptive field layer L1 and the class output neuron layer L2 fuzzy rules are extracted.

40.3.4 eSNN Applications

Different eSNN models and systems have been developed for different applications, such as:

- eSNN for spatio- and spectro-temporal pattern recognition – <http://ncs.ethz.ch/projects/evospike>
- Dynamic eSNN (deSNN) for moving object recognition – [40.60]
- Spike pattern association neuron (SPAN) for generation of precise time spike sequences as a response to recognized input spiking patterns – [40.61]
- Environmental and ecological modeling – [40.44]
- EEG data modeling – [40.62]
- Neuromorphic SNN hardware – [40.63, 64]
- Neurogenetic models (Sect. 40.4).

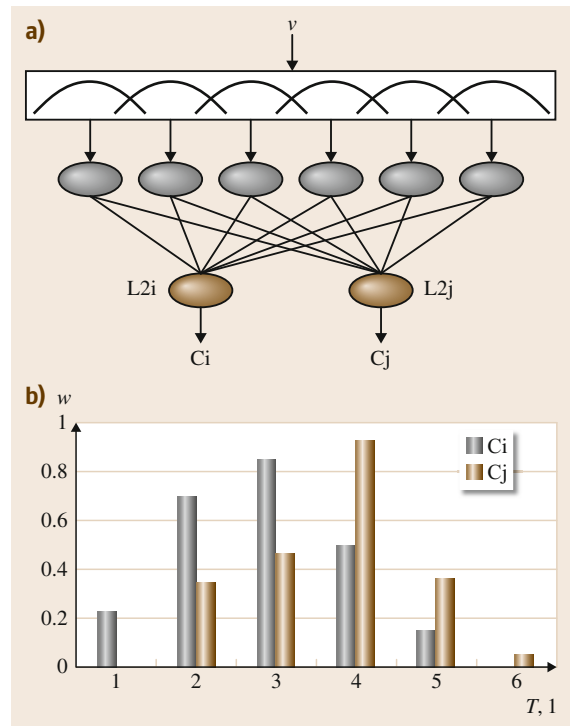


Fig. 40.9 (a) A simple structure of an eSNN for 2-class classification based on one input variable using six receptive fields to convert the input values into spike trains. (b) The connection weights of the connections to class Ci and Cj output neurons, respectively, are interpreted as fuzzy rules: IF(input variable v is SMALL) THEN class Ci; IF(v is LARGE) THEN class Cj

A review of eSNN methods, systems and their applications can be found in [40.65].

40.4 Computational Neuro-Genetic Modeling (CNGM)

40.4.1 Principles

A neuro-genetic model of a neuron was proposed in [40.41, 66]. It utilizes 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 – Fig. 40.10.

A CNGM is a dynamical model that has two dynamical sub-models:

- GRN, which models dynamical interaction between genes/proteins over time scale T1
- eSNN, which models dynamical interaction between spiking neurons at a time scale T2.

The two sub-models interact over time.

40.4.2 The NeuroCube Framework

A further development of the eSNN and the CNGM was achieved with the introduction of the NeuroCube framework [40.67]. The main idea is to support the creation of multi-modular integrated systems, where different modules, consisting of different neuronal types and genetic parameters *correspond* in a way to different parts of the brain and different functions (e.g., vision, sensory information processing, sound recognition, motor-control) and the whole system works in an integrated mode for brain signal pattern recognition. A concrete model built with the use of the NeuroCube would have a specific structure and a set of algorithms depending on the problem and the application conditions, e.g., classification of EEG, recognition of func-

tional magneto-resonance imaging (fMRI) data, brain computer interfaces, emotional cognitive robotics, and modeling Alzheimer’s disease.

A block diagram of the NeuroCube framework is shown in Fig. 40.11. It consists of the following modules:

- An input information encoding module
- A NeuroCube module
- An output module
- A gene regulatory network (GRN) module.

The main principles of the NeuroCube framework are:

1. NeuroCube is a framework to model brain data (and not a brain model or a brain map).
2. NeuroCube is a selective, approximate map of relevant to the brain data brain regions, along with relevant genetic information, into a 3-D spiking neuronal structure.
3. An initial NeuroCube structure can include known connections between different areas of the brain.
4. There are two types of data used for both training a particular NeuroCube and to recall it on new data: (a) data, measuring the activity of the brain when certain stimuli are presented, e.g., (EEG, fMRI); (b) direct stimuli data, e.g., sound, spoken language, video data, tactile data, odor data, etc.
5. A NeuroCube architecture, consisting of a NeuroCube module, (GRN)s at the lowest level, and a higher-level evaluation (classification) module.
6. Different types of neurons and learning rules can be used in different areas of the architecture.
7. Memory of the system is represented as a combination of: (a) short-term memory, represented as changes of the neuronal membranes and temporary changes of synaptic efficacy; (b) long-term memory, represented as a stable establishment of synaptic efficacy; (c) genetic memory, 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.
8. Parameters in the NeuroCube are defined by genes/proteins that form dynamic GRN models.
9. NeuroCube can potentially capture in its internal representation both spatial and temporal characteristics from multimodal brain data.
10. The structure and the functionality of a NeuroCube architecture evolve in time from incoming data.

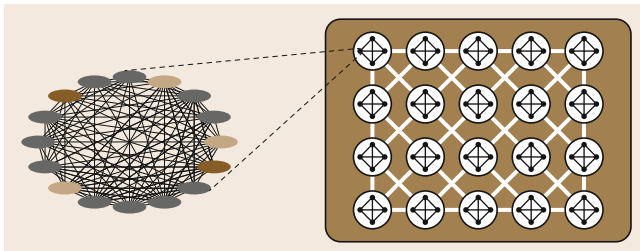


Fig. 40.10 A schematic diagram of a computational neuro-genetic modeling (CNGM) framework consisting of a gene/protein regulatory network (GRN) as part of an eSNN (after [40.41])

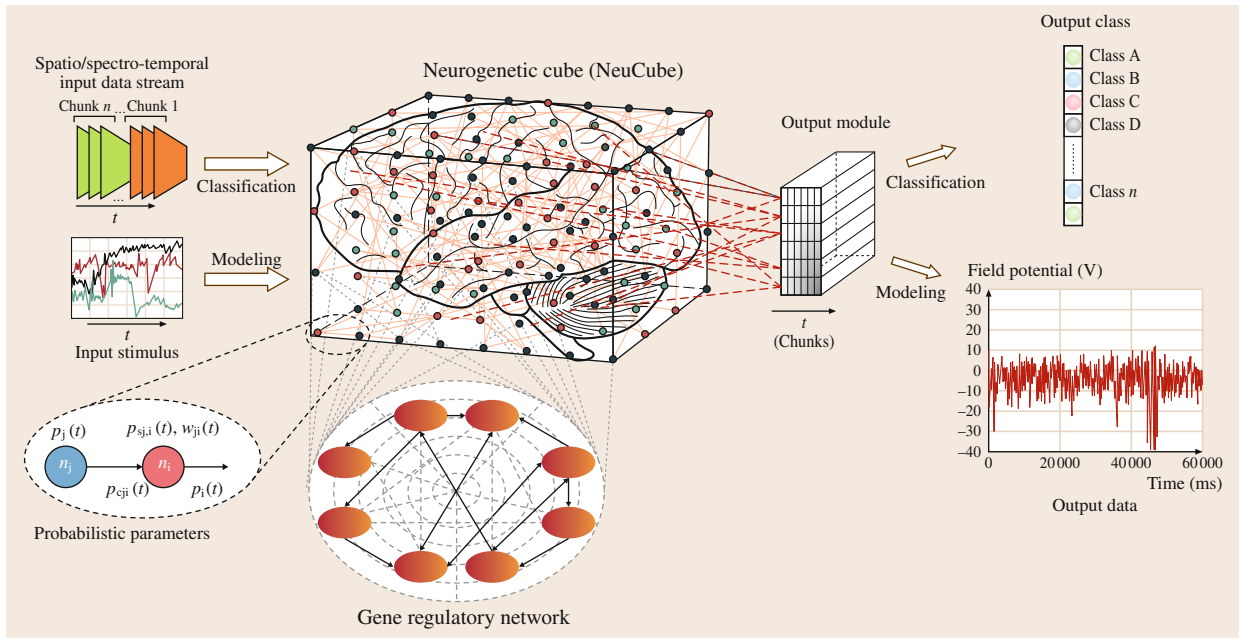


Fig. 40.11 The NeuroCube framework (after [40.67])

40.4.3 Quantum-Inspired Optimization of eSNN and CNGM

A CNGM has a large number of parameters that need to be optimized for an efficient performance. Quantum-inspired optimization methods are suitable for this purpose as they can deal with a large number of variables and will converge in much faster time than any other optimization algorithms [40.68]. Quantum-inspired eSNN (QeSNN) use the principle of superposition of states to represent and optimize features (input variables) and parameters of the eSNN including genes in a GRN [40.44]. They are optimized through a quantum-inspired genetic algorithm [40.44]

or a quantum-inspired particle swarm optimization algorithm [40.69]. Features are represented as *qubits* in a superposition of 1 (selected), with a probability α , and 0 (not selected) with a probability β . When the model has to be calculated, the quantum bits *collapse* in 1 or 0.

40.4.4 Applications of CNGM

Various applications of CNGM have been developed such as:

- Modeling brain diseases [40.41, 70]
- EEG and fMRI spatio-temporal pattern recognition [40.67].

40.5 Conclusions and Further Directions

This chapter presented a brief overview of the main principles of a class of neural networks called evolving connectionist systems (ECOS) along with their applications for computational intelligence. ECOS facilitate fast and accurate learning from data and new knowledge discovery across application areas. They

integrate principles from neural networks, fuzzy systems, evolutionary computation, and quantum computing. The future directions and applications of ECOS are foreseen as a further integration of principles from information science-, bio-informatics, and neuro-informatics [40.71].

References

- 40.1 N. Kasabov: Evolving fuzzy neural networks – Algorithms, applications and biological motivation. In: *Methodologies for the Conception, Design Application of Soft Computing*, ed. by T. Yamakawa, G. Matsumoto (World Scientific, Singapore 1998) pp. 271–274
- 40.2 N. Kasabov: *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering* (MIT, Cambridge 1996) p. 550
- 40.3 N. Kasabov, S. Shishkov: A connectionist production system with partial match and its use for approximate reasoning, *Connect. Sci.* **5**(3/4), 275–305 (1993)
- 40.4 N. Kasabov: Hybrid connectionist production system, *J. Syst. Eng.* **3**(1), 15–21 (1993)
- 40.5 L.A. Zadeh: Fuzzy sets, *Inf. Control* **8**, 338–353 (1965)
- 40.6 L.A. Zadeh: Fuzzy logic, *IEEE Computer* **21**, 83–93 (1988)
- 40.7 L.A. Zadeh: A theory of approximate reasoning. In: *Machine Intelligence*, Vol. 9, ed. by J.E. Hayes, D. Michie, L.J. Mikulich (Ellis Horwood, Chichester 1979) pp. 149–194
- 40.8 N. Kasabov: Incorporating neural networks into production systems and a practical approach towards realisation of fuzzy expert systems, *Comput. Sci. Inf.* **21**(2), 26–34 (1991)
- 40.9 N. Kasabov: Hybrid connectionist fuzzy production systems – Towards building comprehensive AI, *Intell. Autom. Soft Comput.* **1**(4), 351–360 (1995)
- 40.10 N. Kasabov: Connectionist fuzzy production systems, *Lect. Notes Artif. Intell.* **847**, 114–128 (1994)
- 40.11 N. Kasabov: Hybrid connectionist fuzzy systems for speech recognition and the use of connectionist production systems, *Lect. Notes Artif. Intell.* **1011**, 19–33 (1995)
- 40.12 T. Yamakawa, E. Uchino, T. Miki, H. Kusanagi: A neo fuzzy neuron and its application to system identification and prediction of the system behaviour, *Proc. 2nd Int. Conf. Fuzzy Log. Neural Netw.* (Iizuka, Japan 1992) pp. 477–483
- 40.13 T. Yamakawa, S. Tomoda: A fuzzy neuron and its application to pattern recognition, *Proc. 3rd IFSA Congr.*, ed. by J. Bezdek (Seattle, Washington 1989) pp. 1–9
- 40.14 T. Furuhashi, T. Hasegawa, S. Horikawa, Y. Uchikawa: An adaptive fuzzy controller using fuzzy neural networks, *Proc. 5th IFSA World Congr. Seoul* (1993) pp. 769–772
- 40.15 N. Kasabov, J.S. Kim, M. Watts, A. Gray: FuNN/2 – A fuzzy neural network architecture for adaptive learning and knowledge acquisition, *Inf. Sci.* **101**(3/4), 155–175 (1997)
- 40.16 N. Kasabov: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning, *IEEE Trans. Syst. Man Cybern. B* **31**(6), 902–918 (2001)
- 40.17 D. Deng, N. Kasabov: On-line pattern analysis by evolving self-organising maps, *Neurocomputing* **51**, 87–103 (2003)
- 40.18 N. Kasabov: *Evolving Connectionist Systems: Methods and Applications in Bioinformatics, Brain Study and Intelligent Machines*, *Perspective in Neural Computing* (Springer, Berlin, Heidelberg 2003)
- 40.19 M. Watts: A decade of Kasabov's evolving connectionist systems: A review, *IEEE Trans. Syst. Man Cybern. C* **39**(3), 253–269 (2009)
- 40.20 N. Kohonen: *Self-Organizing Maps*, 2nd edn. (Springer, Berlin, Heidelberg 1997)
- 40.21 F. Girosi: Regularization theory, radial basis functions and networks. In: *From Statistics to Neural Networks*, ed. by V. Cherkassky, J.H. Friedman, H. Wechsler (Springer, Heidelberg 1994) pp. 166–187
- 40.22 G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds, D.B. Rosen: Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analogue multidimensional maps, *IEEE Trans. Neural Netw.* **3**(5), 698–713 (1991)
- 40.23 B. Fritzke: A growing neural gas network learns topologies, *Adv. Neural Inf. Process. Syst.* **7**, 625–632 (1995)
- 40.24 J. Platt: A resource allocating network for function interpolation, *Neural Comput.* **3**, 213–225 (1991)
- 40.25 N. Kasabov, Q. Song: DENFIS: Dynamic, evolving neural-fuzzy inference systems and its application for time-series prediction, *IEEE Trans. Fuzzy Syst.* **10**, 144–154 (2002)
- 40.26 N. Kasabov: *Evolving Connectionist Systems: The Knowledge Engineering Approach* (Springer, Berlin, Heidelberg 2007)
- 40.27 J. Bezdek: A review of probabilistic, fuzzy, and neural models for pattern recognition, *J. Intell. Fuzzy Syst.* **1**, 1–25 (1993)
- 40.28 J. Bezdek (Ed.): *Analysis of Fuzzy Information* (CRC, Boca Raton 1987)
- 40.29 J. Bezdek: *Pattern Recognition with Fuzzy Objective Function Algorithms* (Plenum, New York 1981)
- 40.30 R.R. Yager, D. Filev: Generation of fuzzy rules by mountain clustering, *J. Intell. Fuzzy Syst.* **2**, 209–219 (1994)
- 40.31 Q. Song, N. Kasabov: NFI: A neuro-fuzzy inference method for transductive reasoning, *IEEE Trans. Fuzzy Syst.* **13**(6), 799–808 (2005)
- 40.32 Q. Song, N. Kasabov: TWNFI – A transductive neuro-fuzzy inference system with weighted data normalisation for personalised modelling, *Neural Netw.* **19**(10), 1591–1596 (2006)
- 40.33 N. Kasabov, Y. Hu: Integrated optimisation method for personalised modelling and case studies for medical decision support, *Int. J. Funct. Inf. Pers. Med.* **3**(3), 236–256 (2010)
- 40.34 N. Kasabov: Global, local and personalised modelling and profile discovery in bioinformatics: An

- integrated approach, *Pattern Recognit. Lett.* **28**(6), 673–685 (2007)
- 40.35 S. Ozawa, S. Pang, N. Kasabov: On-line feature selection for adaptive evolving connectionist systems, *Int. J. Innov. Comput. Inf. Control* **2**(1), 181–192 (2006)
- 40.36 S. Ozawa, S. Pang, N. Kasabov: Incremental learning of feature space and classifier for online pattern recognition, *Int. J. Knowl. Intell. Eng. Syst.* **10**, 57–65 (2006)
- 40.37 M. Watts: *Evolving Connectionist Systems: Characterisation, Simplification, Formalisation, Explanation and Optimisation*, Ph.D. Thesis (University of Otago, Dunedin 2004)
- 40.38 N.L. Mineu, A.J. da Silva, T.B. Ludermir: Evolving neural networks using differential evolution with neighborhood-based mutation and simple sub-population scheme, *Proc. Braz. Symp. Neural Netw. SBRN* (2012) pp. 190–195
- 40.39 P. Angelov: *Evolving Rule-Based Models: A Tool for Design of Flexible Adaptive Systems* (Springer, Berlin, Heidelberg 2002)
- 40.40 N. Kasabov: Adaptive modelling and discovery in bioinformatics: The evolving connectionist approach, *Int. J. Intell. Syst.* **23**, 545–555 (2008)
- 40.41 L. Benuskova, N. Kasabov: *Computational Neuro-Genetic Modelling* (Springer, Berlin, Heidelberg 2007)
- 40.42 L. Huang, Q. Song, N. Kasabov: Evolving connectionist system based role allocation for robotic soccer, *Int. J. Adv. Robot. Syst.* **5**(1), 59–62 (2008)
- 40.43 N. Kasabov: Adaptation and interaction in dynamical systems: Modelling and rule discovery through evolving connectionist systems, *Appl. Soft Comput.* **6**(3), 307–322 (2006)
- 40.44 S. Schliebs, M. Defoin-Platel, S.P. Worner, N. Kasabov: Integrated feature and parameter optimization for evolving spiking neural networks: Exploring heterogeneous probabilistic models, *Neural Netw.* **22**, 623–632 (2009)
- 40.45 N. Kasabov, E. Postma, J. van den Herik: AVIS: A connectionist-based framework for integrated auditory and visual information processing, *Inf. Sci.* **123**, 127–148 (2000)
- 40.46 S. Pang, T. Ban, Y. Kadobayashi, K. Kasabov: LDA merging and splitting with applications to multi-agent cooperative learning and system alteration, *IEEE Trans. Syst. Man Cybern. B* **42**(2), 552–564 (2012)
- 40.47 H. Widiputra, R. Pears, N. Kasabov: Multiple time-series prediction through multiple time-series relationships profiling and clustered recurring trends, *Lect. Notes Artif. Intell.* **6635**, 161–172 (2011)
- 40.48 D. Hebb: *The Organization of Behavior* (Wiley, New York 1949)
- 40.49 A.L. Hodgkin, A.F. Huxley: A quantitative description of membrane current and its application to conduction and excitation in nerve, *J. Physiol.* **117**, 500–544 (1952)
- 40.50 J. Hopfield: Pattern recognition computation using action potential timing for stimulus representation, *Nature* **376**, 33–36 (1995)
- 40.51 W. Maass: *Computing with spiking neurons*. In: *Pulsed Neural Networks*, ed. by W. Maass, C.M. Bishop (MIT, Cambridge 1998) pp. 55–81
- 40.52 W. Gerstner: Time structure of the activity of neural network models, *Phys. Rev. E* **51**, 738–758 (1995)
- 40.53 E.M. Izhikevich: Which model to use for cortical spiking neurons?, *IEEE Trans. Neural Netw.* **15**(5), 1063–1070 (2004)
- 40.54 S. Thorpe, A. Delorme, R. van Ruller: Spike-based strategies for rapid processing, *Neural Netw.* **14**(6/7), 715–725 (2001)
- 40.55 N. Kasabov: To spike or not to spike: A probabilistic spiking neuron model, *Neural Netw.* **23**(1), 16–19 (2010)
- 40.56 S. Wysoski, L. Benuskova, N. Kasabov: Evolving spiking neural networks for audiovisual information processing, *Neural Netw.* **23**(7), 819–836 (2010)
- 40.57 D. Verstraeten, B. Schrauwen, M. d’Haene, D. Stroobandt: An experimental unification of reservoir computing methods, *Neural Netw.* **20**(3), 391–403 (2007)
- 40.58 S. Song, K. Miller, L. Abbott: Competitive Hebbian learning through spike-timing-dependent synaptic plasticity, *Nat. Neurosci.* **3**, 919–926 (2000)
- 40.59 S. Soltic, N. Kasabov: Knowledge extraction from evolving spiking neural networks with rank order population coding, *Int. J. Neural Syst.* **20**(6), 437–445 (2010)
- 40.60 N. Kasabov, K. Dhoble, N. Nuntalid, G. Indiveri: Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition, *Neural Netw.* **41**, 188–201 (2013)
- 40.61 A. Moheemmed, S. Schliebs, S. Matsuda, N. Kasabov: SPAN: Spike pattern association neuron for learning spatio-temporal spike patterns, *Int. J. Neural Syst.* **22**(4), 1250012 (2012)
- 40.62 N. Nuntalid, K. Dhoble, N. Kasabov: EEG classification with BSA spike encoding algorithm and evolving probabilistic spiking neural network, *Lect. Notes Comput. Sci.* **7062**, 451–460 (2011)
- 40.63 G. Indiveri, B. Linares-Barranco, T.J. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, K. Boahen: Neuromorphic silicon neuron circuits, *Front. Neurosci.* **5**, 5 (2011)
- 40.64 G. Indiveri, E. Chicca, R.J. Douglas: Artificial cognitive systems: From VLSI networks of spiking neurons to neuromorphic cognition, *Cogn. Comput.* **1**(2), 119–127 (2009)
- 40.65 S. Schliebs, N. Kasabov: Evolving spiking neural networks – a survey, *Evol. Syst.* **4**(2), 87–98 (2013)
- 40.66 N. Kasabov, L. Benuskova, S. Wysoski: A computational neurogenetic model of a spiking neuron, *Neural Netw. IJCNN’05. Proc.* (2005) pp. 446–451

- 40.67 N. Kasabov: NeuCube EvoSpike architecture for spatio-temporal modelling and pattern recognition of brain signals, *Lect. Notes Comput. Sci.* **7477**, 225–243 (2012)
- 40.68 M. Defoin-Platel, S. Schliebs, N. Kasabov: Quantum-inspired evolutionary algorithm: A multi-model EDA, *IEEE Trans. Evol. Comput.* **13**(6), 1218–1232 (2009)
- 40.69 H. Nuzly, A. Hamed, S.M. Shamsuddin: Probabilistic evolving spiking neural network optimization using dynamic quantum inspired particle swarm optimization, *Aust. J. Intell. Inf. Process. Syst.* **11**(1), 5–15 (2010)
- 40.70 N. Kasabov, R. Schliebs, H. Kojima: Probabilistic computational neurogenetic framework: From modelling cognitive systems to Alzheimer's disease, *IEEE Trans. Auton. Ment. Dev.* **3**(4), 300–311 (2011)
- 40.71 N. Kasabov (Ed.): *Springer Handbook of Bio/Neuroinformatics* (Springer, Berlin, Heidelberg 2014)