



## **Chapter II**

# **Simultaneous Evolution of Network Architectures and Connection Weights in Artificial Neural Networks**

Ruhul A. Sarker, University of New South Wales, Australia

Hussein A. Abbass, University of New South Wales, Australia

## **Abstract**

---

*Artificial Neural Networks (ANNs) have become popular among researchers and practitioners for modeling complex real-world problems. One of the latest research areas in this field is evolving ANNs. In this chapter, we investigate the simultaneous evolution of network architectures and connection weights in ANNs. In simultaneous evolution, we use the well-known concept of multiobjective optimization and subsequently evolutionary multiobjective algorithms to evolve ANNs. The results are promising when compared with the traditional ANN algorithms. It is expected that this methodology would provide better solutions to many applications of ANNs.*

## Introduction

---

Feed-forward ANNs have found extensive acceptance in many disciplines for modeling complex real-world problems including the finance and manufacturing domains. An ANN is formed from a group of units, called neurons or processing elements, connected with arcs, called synapses or links, where each arc is associated with a weight representing the strength of the connection, and usually the nodes are organized in layers. Each neuron has an input equal to the weighted sum of the outputs of those neurons connected to it. The weighted sum of the inputs represents the activation of the neuron. The activation signal is passed through a transfer function to produce a single neuron's output. The transfer function introduces nonlinearity to the network. The behavior of a neural network is determined by the transfer functions, the learning rule by which arcs update their weights, and the architecture itself in terms of the number of connections and layers. Training is the process of adjusting the networks' weights to minimize the difference between the network output and the desired output on a suitable metric space. Once the network is trained, it can be tested by a new dataset.

As previously mentioned, the performance of a neural network for a given problem depends on the transfer function, network architecture, connection weights, inputs, and learning rule. The architecture of an ANN includes its topological structure, that is, connectivity and number of nodes in the network. The architectural design is crucial for successful application of ANNs because the architecture has a significant impact on the overall processing capabilities of the network. In most function-approximation problems, one hidden layer is sufficient to approximate continuous functions (Basheer, 2000; Hecht-Nielsen, 1990). Generally, two hidden layers may be necessary for learning functions with discontinuities (Hecht-Nielsen, 1990). The determination of the appropriate number of hidden layers and number of hidden nodes in each layer is one of the important tasks in ANN design. A network with too few hidden nodes would be incapable of differentiating between complex patterns, leading to a lower estimate of the actual trend. In contrast, if the network has too many hidden nodes it will follow the noise in the data due to overparameterization leading to poor generalization for test data (Basheer & Hajmeer, 2000). As the number of hidden nodes increases, training becomes excessively time-consuming.

The most popular approach to finding the optimal number of hidden nodes is by trial and error. Methods for network growing such as cascade correlation (Fahlman & Lebiere, 1990) and for network pruning such as optimal brain damage (LeCun, Denker, & Solla, 1990) have been used to overcome the unstructured and somehow unmethodical process for determining good network architecture. However, all these methods still suffer from their slow convergence and long training time. Nowadays, many researchers use evolutionary algorithms to find the appropriate network architecture by minimizing the output error (Kim & Han, 2000; Yao & Liu, 1998).

Weight training in ANNs is usually formulated as a minimization of an error function, such as the mean square error between target and actual outputs averaged over all examples (training data) by iteratively adjusting connection weights. Most training algorithms,

13 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/simultaneous-evolution-network-architectures-connection/5347](http://www.igi-global.com/chapter/simultaneous-evolution-network-architectures-connection/5347)

## Related Content

---

### An RRAM-Based Neuromorphic Accelerator for Speech-Based Emotion Recognition

Afroz Fatima and Abhijit Pethe (2023). *Neuromorphic Computing Systems for Industry 4.0* (pp. 63-93).

[www.irma-international.org/chapter/an-rram-based-neuromorphic-accelerator-for-speech-based-emotion-recognition/326834](http://www.irma-international.org/chapter/an-rram-based-neuromorphic-accelerator-for-speech-based-emotion-recognition/326834)

### Optimization Algorithms

(2014). *Medical Diagnosis Using Artificial Neural Networks* (pp. 182-199).

[www.irma-international.org/chapter/optimization-algorithms/111010](http://www.irma-international.org/chapter/optimization-algorithms/111010)

### Feature Selection Algorithm Using Relative Odds for Data Mining Classification

Donald Douglas Atsa'am (2020). *Big Data Analytics for Sustainable Computing* (pp. 81-106).

[www.irma-international.org/chapter/feature-selection-algorithm-using-relative-odds-for-data-mining-classification/238606](http://www.irma-international.org/chapter/feature-selection-algorithm-using-relative-odds-for-data-mining-classification/238606)

### Cooperative Control of Unknown Networked Lagrange Systems using Higher Order Neural Networks

Gang Chen and Frank L. Lewis (2013). *Artificial Higher Order Neural Networks for Modeling and Simulation* (pp. 214-236).

[www.irma-international.org/chapter/cooperative-control-unknown-networked-lagrange/71801](http://www.irma-international.org/chapter/cooperative-control-unknown-networked-lagrange/71801)

### L(h,k)-Labeling of Intersection Graphs

Sk. Amanathulla and Madhumangal Pal (2020). *Handbook of Research on Advanced Applications of Graph Theory in Modern Society* (pp. 135-170).

[www.irma-international.org/chapter/lhk-labeling-of-intersection-graphs/235536](http://www.irma-international.org/chapter/lhk-labeling-of-intersection-graphs/235536)