

Handoff Management Enhancement using Soft Computing Techniques

Yang Jie

School of Information Technology
Wuhan University of Technology

Dheyaa Jasim Kadhim

Electrical Engineering Department
University of Baghdad

ABSTRACT

In 4G cellular system, due to the mobility, proper handoff or handover management is critical to ensure seamless migration from one cell to another cell. Besides of maintaining the connection quality during a handover process, any advanced handover algorithms which can minimize required resources for the connection are critical. When connections need to migrate between heterogeneous networks for performance and high-availability reasons, seamless vertical handoff (between different access technologies) is a necessary first step. In this paper, we try to simulate vertical handoff with a dynamic decision according to the Received Signal Strength (RSS); and we build a self-learning, self-optimization and self-organization to make the best decision by using Artificial Neural Network (ANN) and Fuzzy Junction Computation (FJC).

General Terms

WLAN, Cellular Network, Mobility Management and Vertical Handoff Process.

Keywords

Vertical Handoff, Received Signal Strength (RSS), Artificial Neural Network (ANN) and Fuzzy junction computation (FJC).

1. Introduction

Connection handoff is no longer limited to migration between two subnets in Wireless Local Area Network (WLAN), or between two cells in a cellular network, generally known as "horizontal handoff". In addition to roaming and horizontal handoff within homogeneous subnets supporting service continuity and Quality of Service (QoS) requires seamless Vertical Handoffs (VHO) between heterogeneous wireless access networks. There are many handover schemes that proposed for future mobile telecommunication [1][2], but the research about the handover under multi-services system is still few. As a result, an interesting problem began to grow on how to decide the "best" network to use at a "best" time.

There are two important factors to decide best network and at best time static factors and dynamic factor s. Static factors such as the bandwidth of each network, usage charges of each network, power consumption of each network interface and battery level of mobile device and dynamic factors such as information on current network conditions such as received signal strength (RSS) and current user conditions, such as a mobile host's moving speed can help in improving whole system performance.

Various algorithms for vertical handoff have been proposed in the recent literature. They are seamless handoff scheme for 4G mobile systems based on IP and OFDM [1]. This algorithm minimizes the handoff delay in providing a seamless connection. The seamless handoff scheme obtains

the physical channel for handoff in a connection-free manner with pre-synchronization and pre-forwarding IP contexts. Simulation results in [3] show that DPQ (Dynamic Priority Queue) differs from the former schemes in that the traffic priority in the queue is fixed. In new DPQ handover scheme the traffic priority is dynamically decided by their dwelling time in the queue. The DPQ scheme, which is adapted to different QoS requirements, can significantly improve the system performance, and reduce the drop rate and the block rate. In [4], the scheme addresses this need by introducing a vertical handoff algorithm for integrated UMTS-WLAN which uses signal strength as well as predefined threshold value to predict where and when the handover should occur, thus optimizing user throughput and network performance.

In [5] the handover procedure in femto-cell network is investigated. The procedure is based on 3GPP LTE specification. Three handover scenarios: hand-in, hand-out and inter-FAP are considered and analyzed. In order to achieve the optimize procedure, the handover decision policy based on mobility prediction is introduced and proposed. The reactive and proactive handover strategy is also proposed to mitigate the frequent and unnecessary handover. The result shows that reactive handover is the potential mechanism to mitigate the unnecessary handover. An Autonomic Handover Manager (AHM) based on the autonomic computing concept to decide the best network interface to handover in 4G networks [6]. AHM decides the appropriate policy for the specific service or application without the user's intervention using the context information from the mobile terminal, the network and the user. They present the context information and the context evaluation function to decide handover based on the user preferences. They then describe the scenario to validate its feasibility using multimedia conferencing service on the mobile terminal.

In this paper, a proposal dynamic decision model is simulated to get the right vertical handoff decision by computing the best network and at best time among the networks based on dynamic factor Received Signal Strength (RSS) of the network. We used Artificial Neural Network (ANN) and fuzzy junction computation to make the right decision.

2. Artificial Neural Networks (ANN)

ANNs are modeled on biological processes for information processing, including specifically the nervous system and its basic unit, the neuron. The biological neuron is shown in Figure 1.

Dendrites bring signals from other neurons into the cell body or soma, possibly multiplying each incoming signal by a transfer weighting coefficient. In the soma, cell capacitance integrates the signals which collect in the axon hillock. Once the combined signal exceeds a certain cell threshold, a signal, the action potential, is transmitted through the axon. Cell nonlinearities make the composite action potential a nonlinear

function of the combination of arriving signals. The axon connects through the synapses with the dendrites of subsequent neurons. The synapses operate through the discharge of neurotransmitter chemicals across intercellular gaps, and can be either excitatory (tending to fire the next neuron) or inhibitory (tending to prevent the firing of the next neuron).

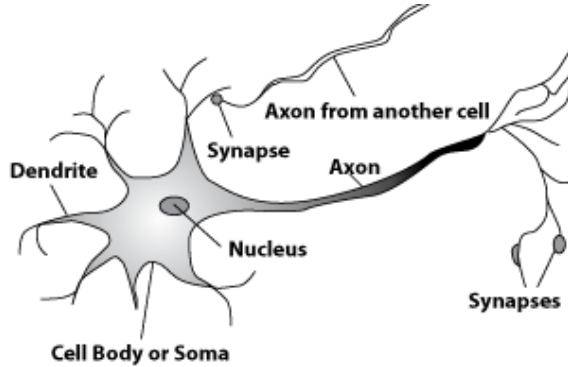


Figure1: The biological neuron

Figure 2 shows the mathematical model of the neural network, which shows the dendrite weights v_j , the firing threshold v_0 (also called the bias), the summation of weighted incoming signals, and the nonlinear function $\sigma(*)$. The cell inputs are the n signals at the time instant $x_1(k), x_2(k), x_3(k), \dots, x_n(k)$ and the output is the scalar $y(k)$, which can be expressed as;

$$y(k) = \sigma\left(\sum_{j=1}^n v_j x_j(k) + v_0\right) \quad (1)$$

Positive weights v_j correspond to excitatory synapses and negative weights to inhibitory synapses. This network was called the perceptron by Rosenblatt in 1959[7].

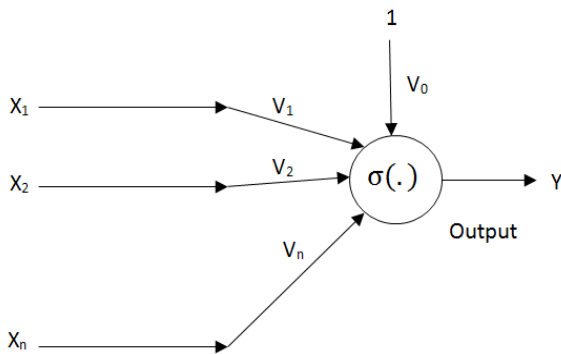


Figure 2: Mathematical of Neuron

Training mode can be implemented through supervised or unsupervised learning. In supervised learning the ANN will have an offline training phase in which neural outputs are compared against a training set. Alterations are made to the synaptic weights to limit the error in classification between the output y_k and the training set d_k . When the ANN correctly classifies the input pattern, the ANN operates in trained mode. Unsupervised learning has no external training patterns. In this mode the ANN self organizes data presented to the network and detects recurrent properties[8]. Figure 3 shows the supervised learning that will be used in this proposed work.

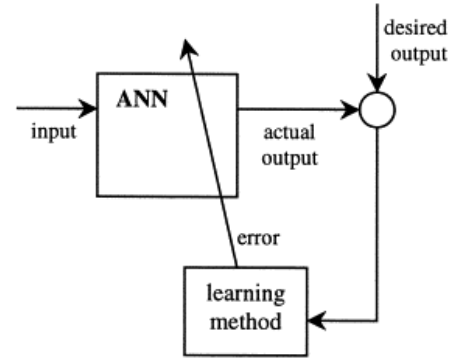


Figure 3: Schematic representation of a supervised ANN

3. Fuzzy Logic System

Fuzzy logic is useful in representing human knowledge in a specific domain of application and in reasoning with that knowledge to make useful inferences or actions. The conventional binary (bivalent) logic is crisp and allows for only two states. This logic cannot handle fuzzy descriptors, examples of which are “fast” which is a fuzzy quantifier and “weak” which is a fuzzy predicate. They are generally qualitative, descriptive, and subjective and may contain some overlapping degree of a neighboring quantity, for example, some degree of “slowness” in the case of the fuzzy quantity “fast.” Fuzzy logic allows for a realistic extension of binary, crisp logic to qualitative, subjective, and approximate situations, which often exist in problems of intelligent machines where techniques of artificial intelligence are appropriate. In fuzzy logic, the knowledge base is represented by if-then rules of fuzzy descriptors. An example of a fuzzy rule would be, “If the speed is slow and the target is far, then moderately increase the power,” which contains the fuzzy descriptors slow, far, and moderate. A fuzzy descriptor may be represented by a membership function. This function gives a membership grade between 0 and 1 for each possible value of the fuzzy descriptor it represents soft computing and intelligent.

In this proposed work, we use fuzzy junction function which performs the AND function by taking the minimum value as a result; we use it to choose the network with minimum learning time (i.e. the fastest learned network)[7].

4. Simulation and Results

The proposed model for our approach can be shown as a flowchart in Figure (4) which has been scheduling the following procedure steps:

1. Neural network for self-learning.
2. Fuzzy Junction Computation (FJC).

The neural network helps the network to be self-learning, self-organized and makes a dynamic decision. In this proposed approach a virtual table of offered networks was assumed, and this table contains six networks each with its Received Signal Strength (RSS) which was taken from actual table.

The input parameter of the neural network is:

- Input RSS of the six networks.
- Momentum term to speed the learning operation.
- Learning rate.
- Maximum allowable error.

- Optimum target for the RSS.

4.1 Self-Learning

The neural network that proposed in this paper consists of three layers, input layer contains six nodes each one for one input RSS, one hidden layer with four nodes; we choose four nodes because we find by training that it gives acceptable accuracy in minimum time and output layer with six nodes each one gives one training target RSS.

The neural network will learn the target RSS to make it closest to the available RSS because we can't change the RSS of the incoming signal but we can choose the best one among them, thus if the none of the incoming RSS is close to the target RSS then the target is changed to make it close to the input RSS and this operation is made by the back propagation learning algorithm in neural network.

As mentioned above, neural network is used for training the network signals according to an optimum RSS signal; in this proposed approach the optimum RSS is measured according to the Motorola GSM equations.

The proposed Table 1 contained six untrained networks with their RSS and their accepted error. In Table 1, we assumed that there are six available networks for the user to make a handoff for his/her call; also we assumed that the call was marked and the user is waiting for the handoff in order to communicate with the required parity in the call.

In the neural network the values of the training parameter are as follows:

- Training rate(η)=0.5
- Momentum term(α)=0.8
- Allowable error (ϵ)= 0.001
- Optimum RSS is determined from Motorola equation which is:

$$G = \frac{4\pi r^2 W}{P_{in}} \quad (2)$$

Where, r is the minimum distance from the antenna to the caller, W is the power per square centimeter and this power is taken from Motorola table shown in Table2, P_{in} is the antenna input power and G is the antenna gain. In our proposed approach, the G is performed as the optimum gain needed to make the handoff. Table 3 shows the training networks with their RSS, training error, and the delay time.

We take an example from Motorola GSM data sheet where it was as follows: (For a base station with the following characteristics: Transmit frequency is 1930MHz, Base station cabinet output power is $P = +39.0\text{dBm}$ (8 watts), Antenna feeder cable loss is $CL = 2.0\text{dB}$, Antenna input power is $P_{in} = P - CL = +39.0 - 2.0 = +37.0\text{dB}$ (5watts), Antenna gain is $G = 16.4\text{dBi}$ (43.65), and the distance that satisfied the above requirement was 1.16 m; this distance perform the far of the caller from the antenna).

Thus the optimum RSS is 8.6981 after we make the calculations.

Figure5 shows the optimum RSS with the untraining RSS of the six networks given in Table1 and Figure 6 shows the optimum RSS with the training RSS of the six networks also shown in Table3. Figure7 shows MATLAB results while it executes the learning program.

Figure8 shows the networks which are inside the range of the choose to make the handoff, the networks below the straight line which perform the threshold do not be choose for the

handoff, rather than the networks which have a RSS above the threshold are chose to hold the handoff according to their learning time.

4.2 Fuzzy Junction Computation

In this proposed approach, the fuzzy junction is used to determine the fastest network (i.e. which network is offered the handoff in shortest time).

The junction operation in the fuzzy system will determine the best time of learning by comparing each incoming RSS with the learning target RSS, and the best time is determined by taking the minimum time of learning because the first incoming RSS entered within the learning target RSS region will be the best RSS for the handoff operation. Figure8 shows the result of the fuzzy junction operation, and we can see that the network with the green curve is the fastest one because it has the shortest time which is network 6.

5. Conclusions

Back propagation learning is proposed in this paper, a feed forward neural network, which uses three layers in order to perform the self-learning to choose the best network for the handoff. Back propagation uses a supervised learning mechanism which captures predictable network behavior while also considering dynamic performance characteristics. The learning algorithm parameters are performed as a given networks using performance metrics from a commercial heterogeneous network. Results presented illustrate up to 90% performance improvement. For a supervised ANN, the optimization of synaptic weights is a tradeoff between learning time and maximum allowable error. With the progressive of the epochs the supervisor can choose the best network with the best RSS for handoff.

6. References

- [1] N.K.H. Choi and H. Yoon, "Seamless Handoff Scheme for 4G Mobile Systems based on IP and OFDM", IEEE Vehicular Technology Conference 2004.
- [2] X. Li and R. Salleh, "Handoff Techniques for 4G Wireless Mobile Internet", Information Technology Journal, Vol. 6 No. 5, PP 745-750, 2007.
- [3] W. Wang, "Dynamic Priority Queue Handover Scheme for Multiservice", IEEE Vehicular Technology Conference 2008.
- [4] S.M.H. Mansur, "Optimal Initiation based Vertical Handover for UMTS-802.11 WLAN Overlay networks", White Paper, Huawei Technologies 2009.
- [5] A. Ulvan, "Handover Scenario and Procedure in LTE-based Femtocell Networks", The 4th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, 2010.
- [6] J.M. Kang, "Towards Autonomic Handover Decision Management in 4G Networks", Lecture Notes in Computer Science, Vol. 4267, PP 145-157, 2006.
- [7] F.L. Lewis, "Neural Network Control of Nonlinear Discrete Time System", Article Book, CRC Press, 1998.
- [8] E. Fallon, "TRAWL – A Traffic Route Adapted Weighted Learning Algorithm", Lecture Notes in Computer Science, Vol. 6649, PP 1-14, 2011.

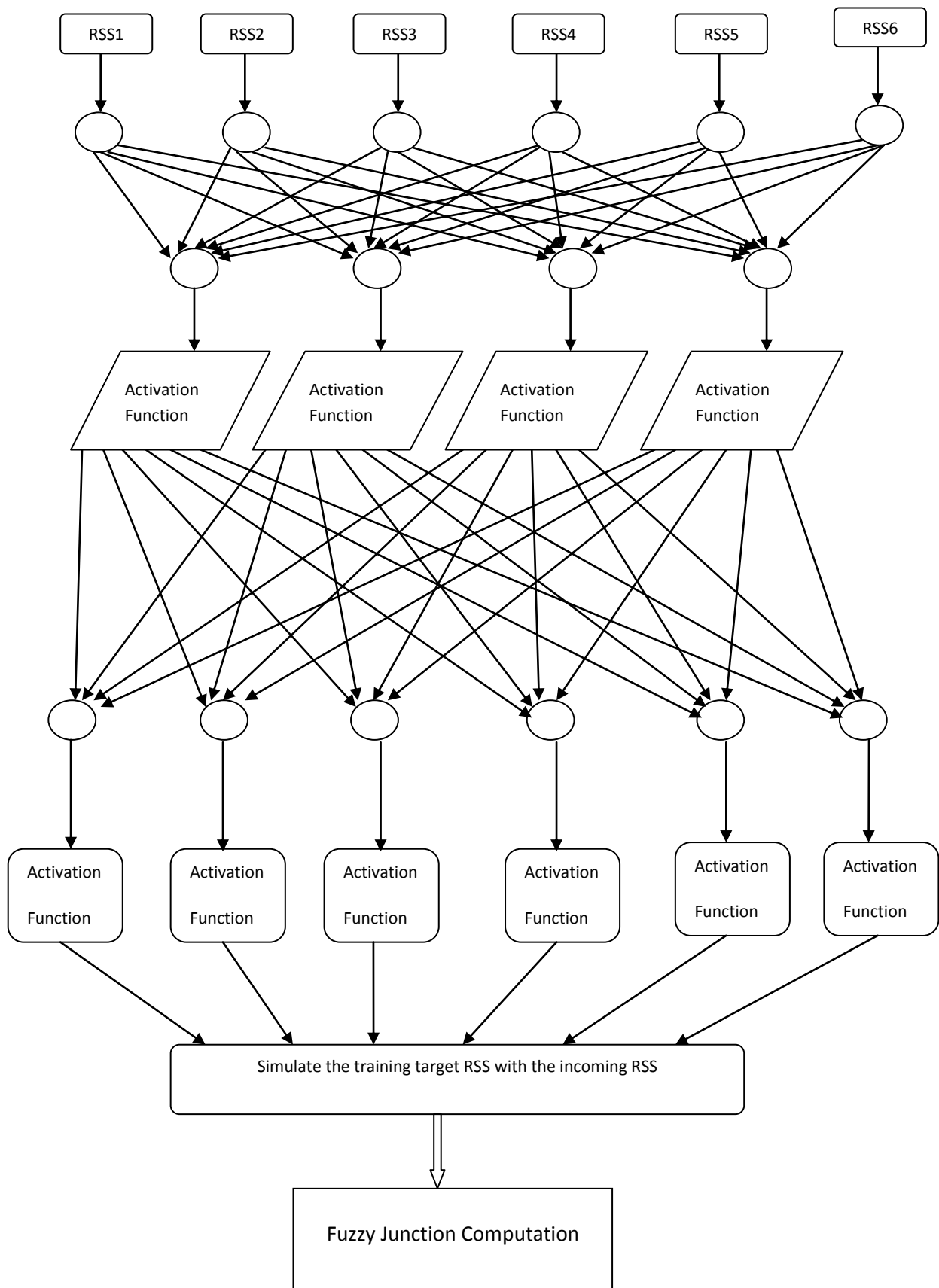


Figure 4: Proposed Approach Flowchart

Table 1: Untrained Networks

Network name	RSS of the i/p signal	Accepted error
Network1	25%	$1*10^{-7}$
Network2	37%	$1*10^{-7}$
Network3	42%	$1*10^{-7}$
Network4	57%	$1*10^{-7}$
Network5	632%	$1*10^{-7}$
Network6	87%	$1*10^{-7}$

Table 2: Motorola Frequency Ceiling

	Uncontrolled Environment		Controlled Environment	
	1930MHz	1970MHz	1930MHz	1970MHz
Ceiling	1.287mW/cm ²	1.313mW/cm ²	6.344mW/cm ²	6.567mW/cm ²

Table 3: Trained networks

Network name	RSS of the i/p signal	Accepted error	Delay/Epoch
Network1	869%	74.8125	0/500
Network2	869%	0.00180406	25/500
Network3	869%	2.47476e-006	30/500
Network4	869%	4.69083e-009	35/500
Network5	869%	4.5083e-009	46/500
Network6	869%	4.09083e-009	67/500

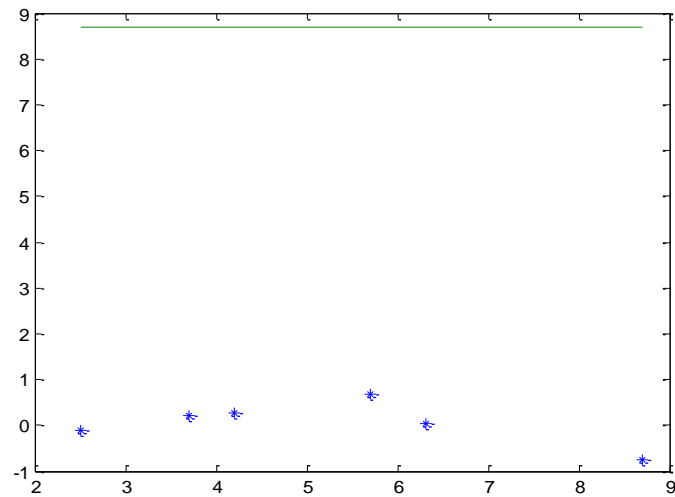


Figure 5: Inputs RSS with Motorola optimum RSS

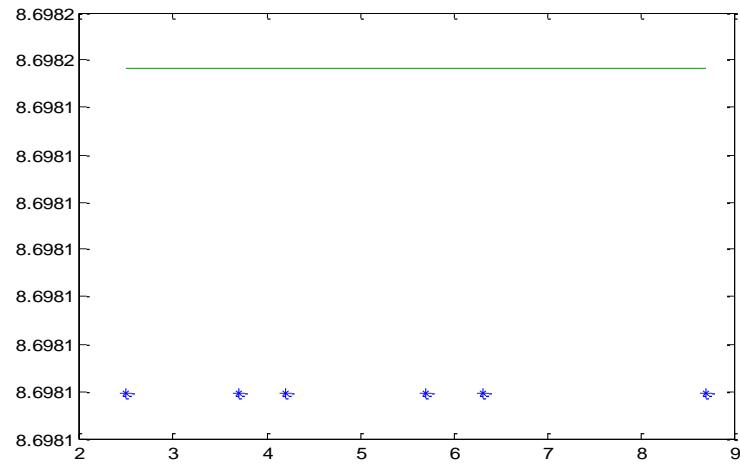


Figure 6: Trained Input RSS with Motorola Optimum RSS

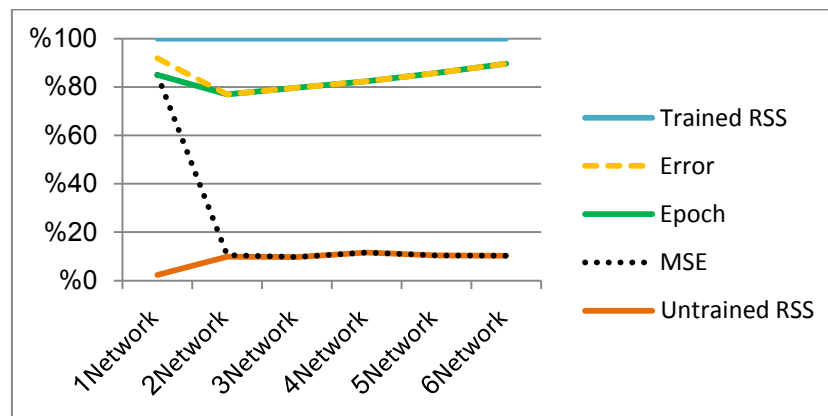


Figure 7: Matlab Training Results

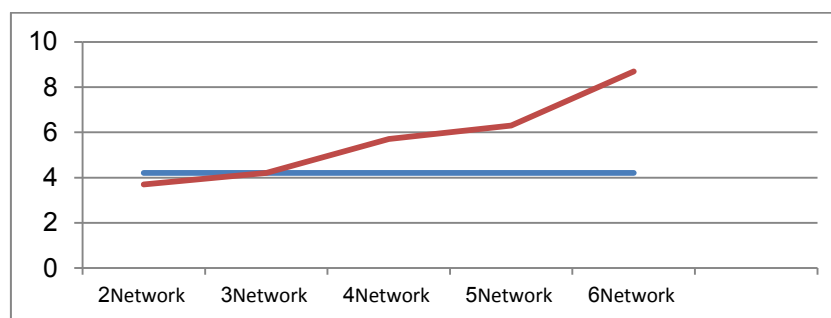


Figure 8: Threshold with Incoming Networks RSS