

Hybrid Algorithm with Weighted Nonlinear Glial Ratio Neural Networks for Coal Mine Rescue Operation

Mary Opokua Ansong

Institute of System Engineering, Faculty of Science,
Jiangsu University, 301 Xuefu, Zhenjiang
212013, P.R. China. &

Department of Computer Science, Faculty of Applied Science
P. O. Box 854, Kumasi Polytechnic, Kumasi-Ghana.

Yao Hong-Xing [corresponding author]

Institute of System Engineering, Faculty of Science,
Jiangsu University, 301 Xuefu, Zhenjiang
212013, P.R. China. &

College of Finance and Economics,
Jiangsu University, 301 Xuefu, Zhenjiang
212013, P.R. China.

Jun Steed Huang

Computer Science and Technology
School of Computer Science & Telecommunication
Suqian College, Jiangsu University
301 Xuefu, Zhenjiang, 212013, P.R. China. &
Institute of Electrical and Computer
Ottawa University, 2625 Queensview Drive Ottawa
ON Canada K2B 8K2, Canada

Abstract

In this paper a Glial Ratio (g-ratio) mix hybrids of 67% Sigmoid and 33% Radial functions (HSCR-BF_{gr}) based on Particle swarm optimisation with the highest survivability of all possible routing redundancies, reliability, efficiency, fault tolerant with minimum fitness error is proposed for underground rescue operation. Nonlinear weights of cosine and sine were imposed on the g-ratio hybrids. In addition we introduced a nonlinear weight with the g-ratio on the Gaussian RBF. The performance of the Hybrid with negative cosine weight (HSCR-BF_{grcos}) was the best among the various g-ratio hybrids as compared to Gaussian with the same nonlinear weight. The hybrid with negative nonlinear cosine weight yielded the best results with an optimised error of 0.011. The proposed Nonlinear Hybrid Algorithm has better capability of approximation to underlying functions with a fast learning speed, high scalability, robusticity and is competitive to the Gaussian with the same nonlinear weight.

Key words: Hybrid Neural Networks, Rescue Operation, Particle Swarm Optimization, Glial Ratio (g-ratio), wireless sensor network, Gaussian Radial Basis Function, nonlinear weight.

1. Introduction

The discipline of neural networks originates from an understanding of the human brain. The brain cells are generic term for the neurons and glial cells. The body's actions and reactions are monitored and regulated by the brain. Continually the brain receives sensory information, rapidly analyzes this data and then respond, controlling body actions and functions. The brain is known to be divided into left and right cerebral hemispheres, i.e. the neocortex which is the center of higher-order thinking, learning and memory; and the cerebellum which is responsible for the balance, posture, coordination and motor control (movement).

To coordinate motor control, there are 3×10^{10} different types of neurons with each neuron connecting to about 10^4 synapses. Neurons are nerve cells that process and transmit information through the nervous system while glial cells provide support, protection, and nutrition to the neurons (Valdez & Melin, 2008), (Halgren, 1994) (Fischl, 2004). Real-world network analysis is complex as it consists of millions of nodes connected by numerous set of edges that make it difficult to analyze and comprehend. The emergence of wireless sensor networks (WSNs) and artificial neural networks (ANN) has improved the ability to analyze such complex structures. The ANN often referred to as neural networks (NN), is a mathematical or computational model based on biological neural networks. It consists of an interconnected group of artificial neurons and process information using a connectionist approach to computation (Fernández-Navarro, Herva's-Martí'nez, Sanchez-Monedero, & Gutiérrez, (2011)) (Fernández-Navarro, Herva's-Martí'nez, A., Pen'a-Barragán, & López-Granados, 2012) and (Soh, et al., 2010).

In most cases an ANN is an adaptive system that changes its structure based on external or internal information flowing through the network during the learning phase. The topology of a neural network can be recurrent i.e. with feedback from the output or feed-forward where the data flows from the input to the output units with no feedback connections (Munoz & Ramosy, 2007). The sigmoid basis function (SBF) and radial basis function (RBF) are the most commonly used algorithms in neural training. According to biological theory, the RBF, related to brain memory function uses radial basis function as activation function (Neruda & Kudova, 2005). The output of the network is a linear combination of radial basis function of the input and neural parameters. Radial basis function networks have many uses, including function approximation, time series prediction, classification, and system control. The structure supports the academic school of connectionist and the idea was first formulated by (Broomhead & Lowe, 1988). SBF a mathematical function having an "S" shape (sigmoid curve), is related to brain reasoning and the structure favors the computational believers, often, sigmoid function refers to the special case of the logistic function.

Another example is the Gompertz curve which is used in modeling systems that saturate at large values of input, e.g. the ogee curve used in spillway of dams (Broomhead & Lowe, 1988). A wide variety of sigmoid functions have been used as activation functions of neurons, including the logistic and hyperbolic tangent weight functions. Sigmoid curves are also common in statistics such as integrals and logistic distribution, normal distribution, and Student's probability density functions. In our opinion, SBF offers nonlinear effects for large input value, RBF provide nonlinear effect at small input value. A nonlinear hybrid of both will result in more nonlinear blending across the entire region. Wireless sensor networks (WSN) gather and process data from the environments and make possible many applications such as environment monitoring, support logistics, health care and emergency response systems as well as military operations. Transmitting data wirelessly impact significant benefits to those investigating buildings, thus allowing them to deploy sensors and monitor from a remote location. Multi-hop transmission in wireless sensor networks conforms to the underground tunnel structure and provides more scalability for communication system construction in rescue situations.

A significant discovery in the field of complex networks has shown that a large number of complex networks including the internet, are scale-free and their connectivity distribution is described by the power-law of the form $\rho(k) \sim k^{-\phi}$, such that it allows for a few nodes of very high degree to exist make it difficult for random attack. A scale-free wireless network topology was therefore used (Jang, Healy, & Mirosław, 2008) (Pan, Tsai, TsengTsai, & Tseng, 2006), and (Kumar, Sukumar, & Nageswari, 2013). However large scale networks such as WSN are usually associated with the challenge of scalability (Goh & Mandic, 2007), in terms of hardware requirements such as memory, central processing unit (CPU) or time, execution, consensus problems associated with distributed algorithm and parallel programming. These problems have been addressed using techniques of localization, routing-free and range-free in sensor networks by (Li & Qin, 2013), (Li, Wang, & Li, 2013) and neural network predictive models in both sigmoid and radial basis functions (Leblecioglu & Halici, 1997) as well as distributed estimation control fields such as multiple redundant manipulators and task execution (Li, Chen, Liu, Li, & Liang, 2012), (Li, Cui, Li, Liu, & Lou, 2012). To this end, we model the incident location as a pure random event, and calculate the probability that communication chain through particular rock layers to the ground is not broken, and let neural network memorize the complicated relationship; such that when real accident happens, the neural network resident in the robot is used to predict the probability based on the rock layer it sees instantly. If the result is positive, the robot waits to receive the rescue signal; otherwise it he moves deeper to the next layer and repeats the procedure.

Since the brain's function is a combination of the left and the right cerebral hemispheres one could be justified to combine some percentages of the Radial and Sigmoid transfer functions neural networks in an attempt to come out with a routing topology that is reliable, efficient, and fault tolerant in the application of the underground rescue system using wireless sensor networks (Chen, Chuah, & Zhao, 2008), (Rajpal, Shishodia, & Sekhon, 2006) This paper proposes a nonlinear Hybrid Neural Networks using Radial and Sigmoid transfer functions in underground communication, based on particle swarm optimisation.

Section 2 explains the preliminaries to the study and generates the routing path that have the highest survival probability the neural training. Section 3 discusses the network optimization model based on the nonlinear weight on the compact radial basis function. Section 4 shows the simulation results of the various hybrids and compare the best with the Gaussian or general RBF (GRBF) and section 5 gives a summary of the findings.

2. Preliminaries

2.1.1 Sensor Deployment

Topological deployment of sensor nodes affects the performance of the routing protocol [20, 21] The ratio of communication range to sensing range as well as the distance between sensor nodes, can affect the network topology.

Let Ω be the sensor sequence for the deployment of total sensors $T=xyz= LRC$, such that

$$\Omega = \begin{cases} \text{For } t = t + 1 \\ \text{node}(t, 2) = (\| -(R+1) * (1 + (-1)^{\text{tog}J}) \| / 2 + j) & \text{for } i = 1 : L, j = 1 : R, k = 1 : C \\ \text{node}(t, 3) = (\| -(C+1) * (1 + (-1)^{\text{tog}K}) \| / 2 + k) \end{cases} \quad (1)$$

$\text{tog}J = \text{ceil}(t / C / R)$ and $\text{tog}K = \text{ceil}(t / C)$ check source and destination node respectively.

$\Omega(i, j, k) = \{1, 1, 1\}, \{1, 2, 1\}, \dots, \{i^{\text{th}}, j^{\text{th}}, k^{\text{th}}\}$ for level 1, row 1, column 1], [level 1, row 2 column 1], [...], and [i^{th} level, j^{th} row, and k^{th} column], respectively. Therefore for a $T=T \times T$, in an underground mine with dimensions of $L=3$, $R=2$ and $C=1$ for depth (level), row (length), and width (column) respectively with 'pm' a sensor apart, implies minimum of 6 sensor nodes will have to be deployed.

2.1.2 Communication

Through-The-Earth (TTE) Communication system transmits voice and data through solid earth, rock, and concrete and is suitable for challenging underground environments such as mines, tunnels, and subways. There were stationary sensor nodes monitoring carbon mono-oxide, temperature, etc. as well as mobile sensors (humans and vehicles) distributed uniformly. Both stationary and mobile sensor nodes were connected to either the Access Point (AP) and/or Access Point Heads (AP Heads) based on transmission range requirements (Chen, Chuah, & Zhao, 2008). The AP Heads serve as cluster leaders and are located in areas where the rock is relatively soft or better signal penetration. This will ensure that nodes are able to transmit the information they receive from APs and sensor nodes. The APs are connected to other APs or TTE. The TTE is dropped through a drilled hole down 300 metres apart based on the rock type. The depth and rock type determine the required number of TTEs needed. Next the DATA-mule is discharged to carry items such as food, water and equipments to the miners underground and return with underground information to rescue team.

2.1.3 Signal/Transmission Reach

Major challenges of sensor networks include battery constraints and energy efficiency to prolong the network lifetime, underground characteristics, transmission range and topology design, among others. Several routing approach for safety evacuation have been proposed by (Simplício, Barreto, Margi, & Carvalho, 2010), (Li, Li, & Yang, 2011), (Tan, Huang, Wu, & Cai, 2011), (Ren, Huang, Cheng, Zhao, & Zhang, 2013), (Ahuja, Ravindra, Orlin, Pallottino, & M.), (Liu & Luo, 2012), and (Shi & Wang, 2007). These were developed depending on specific emergency situations and management requirements. Transmitting data wirelessly impact significant benefits to those investigating buildings, and allowing them to deploy sensors and monitor from a remote location (Jang, Healy, & Mirosław, 2008), (Pan, Tsai, TsengTsai, & Tseng, 2006).

To effectively gain the needed results, researchers have come out with a number of techniques to address the problem of topology control (TC). These include localization of nodes and time; error and path-loss; transmission range and total load each node experiences; and energy conservation which is very crucial in optimizing efficiency and minimizing cost in wireless sensor networks (Feng, Xiao, & Cui, 2011) (Zarifzadeh, Nayyeri, & Yazdan, 2008) (Sausen, Spohn, & Perkusich, 2010). Minimizing transmission range of wireless sensor networks is vital to the efficient routing of the network. This is because the amount of communication energy that each sensor consumes is highly related to its transmission range (Chen, Chuah, & Zhao, 2008). The nodes signal reach $N\delta$ was defined as the integration of the change of the minimum and maximum signal reach, taking into consideration the number of cases (τ) of the rock structure β , from the range $0.7 \geq \beta \leq 0.9$, where 0.7 is the soft-rock and 0.9 is the hardest rock. β is the rock hardness, $N\delta$ is the signal reach for a node, $N\delta_{min}$ and $N\delta_{max}$ are minimum and maximum signal reach respectively. The node signal reach is calculated as

$$N\delta = N\delta_{min} + \int_{min}^{max} N\delta dr \Big|_{\beta=0.7 \leq \beta \leq 0.9} \tag{2}$$

where $N\delta_{min} = \min (L, \min (Row, Col))$ and $N\delta_{max} = \max (L, \max (Row, Col))$

For a connection to be made the absolute difference between i, j should be less than the node signal reach- $N\delta$.

The connection Matrix was given as $\varphi k(i, j) = 1, \text{if } \|i - j\| \leq N\delta$ Otherwise 0;

The relationship between rock hardness and the signal reach is a complicated nonlinear function, which is related to the skin depth of the rock with alternating currents concentrated in the outer region of a conductor (skin depth) by opposing internal magnetic fields, as follows:

$$\text{Skin depth} = \sqrt{2 / (\rho * \omega * \sigma)}$$

ρ = material conductivity, ω = frequency, σ = magnetic permeability, ω = frequency,

The signal (B-field) is attenuated by cube of distance (d), and $B = (k)d^3$

Signal Reach (distance) = 3 * skin Depth

Table 1 identifies 6 common rocks found in mines in relation to hardness or softness of each rock.

A routing path was modeled using a number of TxT size matrices namely the connection matrix (φk), routing matrix (φr), explosion matrix (φx), failed matrix (φf), hope matrix (φh), optimized matrix (φo) and the exit matrix (φe). The hardware survival rate vector (φH) and the survival rate vector of each miner (φv) were also generated, (φh) \neq (φH). A sensor node is named by its 3-D integer (x,y,z) coordinates, where $1 \leq x \leq R, 1 \leq y \leq C, 1 \leq z \leq L$ for $T=R*C*L$ being total number of nodes. If the node (a,b,c) is connected with node (d,e,f) then the element on $((a-1)*C*L+(b-1)*L+c)^{th}$ row and column is 1, otherwise 0 and routing was limited to total multiple-points connections available. In arriving at the final optimized vector for transmission, each matrix was generated τ times.

The $\varphi r \subset \varphi k \begin{cases} M_\rho \leq \alpha \\ M_\rho \text{ is even} \end{cases}$, M_ρ representing the maximum point-to- multi-point connection was imposed on it

such that M_ρ is even allowing bidirectional communication, and i, j were checking source and destination nodes respectively.

$$\rightarrow \varphi r = 1, \text{if } \|i - j\| \geq (M_\rho / 2) \text{ otherwise } 0; \text{ for } i, j = 1:T, \tag{4}$$

2.1.4 Hardware, software and Network Fault Tolerant considerations

Network security is a critical issues in wireless sensor networks as it significantly affects the efficiency of the communication and many key management schemes and fault diagnosis had been proposed to mitigate this constraint (Zarifzadeh, Nayyeri, & Yazdan, 2008), and (Riaz, et al., (2008), (Wang, Zhou, Liu, & Wu, 2012). In an event of accident (ψ) occurring, the routing path would be affected by $(1 - \psi)$ where ψ is any random value within β , that would cause explosion on φr matrix and result in $\varphi f \leftarrow \varphi x$ such that,

$$\varphi x = (1 - \psi)\varphi r; \quad \varphi f(i, j) = 1 \Big\| \text{if } \{ \varphi x(i, j) < \lambda L \} \tag{5}$$

$\varphi f(i, j) = 0$ if $\{\varphi x(i, j) \geq \lambda H\}$ else $\varphi f = \varphi x(i, j) / \lambda H$ for $(\lambda L$ or $\lambda H)$ representing the lower and higher accident impact thresholds respectively. Elements 0, 1, and 2 in φx imply the link(s) were not affected, element 3, 4, 5, 6, represent a probability for the links been able to transmit data while any figure above 7 means the link is totally dead. Region 1, indicates that links are not affected, region 2 gives the probability of link available and the last region indicates the link is completely down (Figure 1). The matrices $\varphi i, \varphi k, \varphi r, \varphi x, \varphi h$ for $\tau = n$, with dimensions of $L=3, R=2$ and $C=1$ for depth (level), row (length), and width (column) respectively were generated as follows:

$$Seq = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \\ 2 & 1 & 1 \\ 3 & 1 & 1 \\ 3 & 2 & 1 \end{pmatrix} \quad i\varphi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \varphi k = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

$$\varphi r = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \quad \varphi x = \begin{pmatrix} 0 & 1 & 2 & 2 & 0 & 0 \\ 1 & 5 & 1 & 2 & 0 & 0 \\ 0 & 8 & 1 & 5 & 9 & 1 \\ 1 & 1 & 5 & 1 & 1 & 17 \\ 1 & 1 & 4 & 11 & 0 & 2 \\ 4 & 4 & 19 & 5 & 3 & 0 \end{pmatrix} \quad \varphi f = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & .6 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & .6 & 0 & 1 \\ 1 & 1 & .6 & 1 & 1 & 0 \\ 1 & 1 & .75 & 0 & 1 & 1 \\ .75 & .75 & 0 & .6 & 1 & 1 \end{pmatrix}$$

1=transmission path explosion sizes

Element ‘0’ on φf depicts a connection while ‘1’ means availability of connection and represent the connection to the fixed sink node(s) along the edge, or the emergency connection to the mobile data mule(s). A new set of routing path (φh) and exit matrix (φe) for transmission was calculated as

$$\varphi h = \varphi f * \varphi r \text{ and } \varphi e = Ne - \psi \tag{6}$$

The mathematical objective here was to find an optimized routing matrix φo that has the maximum survivability. The exit matrix φe described the success rate from each node to the sink(s), φe assumes Ne exits are available with an error margin ψ . In most practical applications, more than one sinks are used, and sink node is either through the fiber or TTE. It is important to note that, in real rescue situations the software and hardware including radio frequency identification (RFID) may fail as a result of the effect from φx , a matrix φs was used to describe software or relational database management system (RDBMS) failure rate including bugs or attacks as

$$\varphi s = 1 - \left(\left(\frac{1}{T + random} \right) ("Geometric", fail, T, T) \right) \tag{7}$$

$$\varphi h = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & .6 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & .6 & 0 & 0 \\ 0 & 0 & .6 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & .6 & 1 & 1 \end{pmatrix} \quad \varphi o = \begin{pmatrix} .8 & .8 & .8 & 0 & 0 & 0 \\ .8 & .48 & .8 & 0 & 0 & 0 \\ 0 & 0 & .8 & .48 & 0 & 0 \\ 0 & 0 & .48 & .8 & .8 & 0 \\ 0 & 0 & 0 & 0 & .8 & .8 \\ 0 & 0 & 0 & .48 & .8 & .8 \end{pmatrix} \quad \varphi e = \begin{pmatrix} 1.44 & 1.44 & 1.44 & 0 & 0 & 0 \\ 1.44 & .864 & 1.44 & 0 & 0 & 0 \\ 0 & 0 & 1.44 & .864 & 0 & 0 \\ 0 & 0 & .864 & 1.44 & 1.44 & 0 \\ 0 & 0 & 0 & 0 & 1.44 & 1.44 \\ 0 & 0 & 0 & .864 & 1.44 & 1.44 \end{pmatrix}$$

To obtain the final survival vector (φR) it was assumed each miner will have an RFID; a vector φI was used to describe its failure rate, including risks of running out of battery and another vector φH for the hardware failure rate.

$$\varphi I = 1 - \left(\left(\frac{1}{T + r} \right) * ("Geometric", fail, 1, T) \right) \tag{8}$$

T = total number of nodes, r is a random number generated from the vector T

$$r : 0 \rightarrow \infty; T + random : T \rightarrow \infty$$

$\frac{1}{T+r} : \frac{1}{T} \rightarrow 0; 1 - \frac{1}{T+r} : 1 - \frac{1}{T} \rightarrow 1$ is the minimum, therefore for T=N nodes, we have

$$1 - \frac{1}{N} = \frac{N-1}{N} \rightarrow 1 \text{ for } \frac{N-1}{N} \rightarrow \text{node is dead, } 1 \rightarrow \text{node is alive}$$

$$\varphi H = \min (1, \varphi I[X\Psi]) ; \text{ for } [X\Psi] = \varphi e / M_p \tag{9}$$

The survival rate of each miner was (φv). To ensure that a reliable system is in place for emergencies the final survival rate vector (φR) was calculated.

$$\varphi v = \varphi s * \varphi e; \varphi R = \varphi v * \varphi s; \text{ and } R = \varphi R, \tag{10}$$

$\varphi s =$	$\begin{pmatrix} .8333 & .8889 & .9091 & .8333 & .8571 & .8333 \\ .8889 & .8333 & .8750 & .8571 & .8750 & .8333 \\ .8571 & .9167 & .8750 & .9091 & .8889 & .8333 \\ .8750 & .8571 & .8571 & .8571 & .8571 & .8333 \\ .8333 & .8333 & .8571 & .8333 & .8889 & .9000 \\ .9167 & .8333 & .9091 & .8333 & .8889 & .8571 \end{pmatrix}$	<p>The survival rate of RFID, hardware and of each miner vectors were displayed as</p> $\begin{aligned} \varphi I &= 0.8571 & 0.8750 & 0.8571 & 0.8889 & 0.8571 & 0.8333 \\ \varphi H &= 0.6236 & 0.4976 & 1 & 0.6851 & 0.9286 & 0.6086 \\ \varphi R &= 0.5409 & 0.4281 & 0.8834 & 0.5850 & 0.8134 & 0.5163 \end{aligned}$
---------------	--	--

Optimization was done numerically using Matlab simulation tool to find the optimum set of routing table through particle swarm search for rescue operation as discussed in the preliminaries using dimensions of L=3,R=2, and C=1. φR is only one case of the 6 vectors, and R is the average of all the 6, 6 represent cases of rock types.

3. The Network Optimization Model

Having found the optimum set of routing table that has the highest survival probability of communicating with, and rescuing miners, it is important to train the neurons such that the initial error will be minimized and more importantly the model must be reliable (Goh & Mandic, 2007). The topology of a neural network can be recurrent or with feedback contained in the network from the output back to the input and the feed-forward where the data flow from the input to the output units. The data processing can extend over several layers of units, but no feedback connections are present, that is, connections extending from the outputs units to inputs units in the same layer or previous layers. Many researchers have come out with neural network predictive models in both sigmoid and radial basis functions, (Munoz & Ramosy, 2007) and Castano, Fernandez-Navarro, Gutierrez and Hervas-Martinez [2012] with applications such as nonlinear transformation, (Leblecioglu & Halici, 1997) extreme learning machine and predicting accuracy in gene classification among others (Fernández-Navarro, Hervas-Martínez, Sanchez-Monedero, & Gutiérrez, (2011)). An optimized vector R was generated as the optimum set of transmission routing table that has the highest survival probability for data transmission Eq. 10.

$$PN = \sum_{i=1}^N RiSi + \sum_{i=1}^2 Si + \sum_{i=1}^2 SiSi + 1, \text{ for } 1 \leq i \leq s \text{ and } 1 \leq k \leq R \tag{11}$$

R, S1, S2 are number of neurons at input, hidden and output layers respectively. PN is the position of the nth particle.

$$Wi = SiR, W2 = SiR, \dots, Wm = SmR, \text{ and } Pi = [R(i - j) + K] = Wi(i, k)$$

There are two thresholds; ($S1 \Rightarrow B1$)

$$\text{Hidden : } B_1(i, j) = P(RS1 + S1S2) \Rightarrow S1 \text{ and Output : } B_2(i, j) = P(RS1 + S1S2) \Rightarrow S2$$

3.1 Artificial Neural networks (ANN)

Artificial Neural networks (ANN) are learning algorithm used to minimize the error between the neural network output and desired output. This is important where relationships exist between weights within the hidden and output layers, and among weights of more hidden layers. In addition other parameters including Mean Iteration, Standard variation, Standard deviation and convergent time (in sections) were evaluated.

The architecture of the learning algorithm, and the activation functions were included in neural networks. Neurons are trained to process store, recognize, and retrieve patterns or database entries to solve combinatorial optimization problems. After encoding the particles, the fitness function was then determined. The goodness of the fit was diagnosed using mean squared error (MSE) as

$$MSE = \frac{1}{ns} \sum_{j=1}^n \sum_{i=1}^s (Y_{j,i} - y_{j,i})^2 \tag{12}$$

where n is number of samples, s is the number of neurons at output layer, $Y_{j,i}$ is the ideal value of j^{th} sample at i^{th} output and $y_{j,i}$ is the actual value of j^{th} sample at i^{th} output. The sigmoid Basis function was given as

$$\left. \begin{aligned} \log sig(R), R = W.P + B \\ \log sig(W.P + B) = \frac{1}{1 + e^{-(W.P+B)}} \end{aligned} \right\} \tag{13}$$

Neuron function S(sigmoid) is logsig, W is weight matrix, P is input vector, B is threshold and the generalised Gaussian function as

$$\phi(r) = e^{-(\epsilon r)^2} \tag{14}$$

However this involves additional square operation and poses computation burden. We therefore proposed a Compact Radial Basis function based on the Gaussian Radial Basis function and Helens' definition (Zhang, Genton, & Liu, 2004) expressed as

$$\left. \begin{aligned} (\exp(-\text{abs}(R)), R = W.P + B) \\ (\exp(-\text{abs}((W.P + B))) \end{aligned} \right\} \\ \phi(out) = (\exp(-\text{abs}(R))) \tag{15}$$

W is weight matrix, is an input vector B is threshold. The focus was to improve on the radial basis function for the mine application.

From Helen's definition (Zhang, Genton, & Liu, 2004) an example of the RBF kernels can be stated as, a function

$\psi : [0, \infty) \rightarrow \mathbb{R}$ such that $k(x, x') = \psi(\|x - x'\|)$ where $x, x' \in \mathcal{X}$ and $\|x - x'\|$ denotes the Euclidean norm with

$$k(x, x') = \exp\left(\frac{-\|x - x'\|^2}{\delta^2}\right).$$

she argued that the global support for RBF radials or kernels has resulted in dense

Gram matrices that can affect large datasets and therefore constructed the following two equations

$$k_{C,v}(x, x') = \phi_{C,v}(\|x - x'\|)k(x, x') \text{ and } \phi_{C,v}(\|x - x'\|) = \left[\left(1 - \frac{\|x - x'\|}{C} \right)_+ \right]^v \text{ where } C > 0, v \geq \frac{d+1}{2}, \text{ and the dot}$$

$(\cdot)_+$ is the positive part. The function $\phi_C(\cdot)$ is a sparsifying operator, which thresholds all the entries satisfying

$(\|x - x'\|) \geq C$ to zeros in the Gram matrix. The new kernel resulting from this construction preserves positive definiteness. This means that given any pair of inputs x and x' where $x = x'$ the shrinkage (the smaller C) is imposed on the function value, $k(x, x')$; the result is that the Gram matrices K and $K_{C,v}$ can be either very similar or quite different, depending on the choice of C.

3.2 The Proposed Hybrid

1. The nonlinear weight g-ratio (HSCR-BF_{gr})

As part of the nervous system, oligodendrocytes are closely related to nerve cells, and provide supporting roles for neurons. The activity of the axons of the neurons depends crucially on myelin sheaths, which reduce ion leakage and decrease the capacitance of the cell membrane. Myelin also increases impulse speed, as saltatory propagation of action potentials occurs at the nodes in between cells.

Oligodendrocytes provide the same functionality as the insulation on a household electrical wire. Furthermore, impulse speed of myelinated axons increases linearly with the axon diameter. The insulation must be proportional to the diameter of the fiber inside. The optimal g-ratio of axon diameter divided by the total fiber diameter (which includes the myelin) is 0.55 to 0.72 (Kumar, Sukumar, & Nageswari, 2013). In our proposed model the weight of 0.67 was considered for SBF which is the g-ratio for our splenium, and the RBF occupied the rest. Other g-ratio hybrids were considered. From Eq. 16, 17 and 18 the nonlinear weight of $[\pm \cos(R)]$ or $*[\pm \sin(R)]$ were imposed on the CRBF with g-ratio (HSCR-BF_{gr}) as $\{(1-\rho)[\exp(-|R|)]\} * [\pm \cos(R)]$ before combining with the SBF

$$\text{HSCR-BF}_{gr}: Y(out) = \rho \log sig(R) + (1-\rho) [\exp(-|R|)] \quad (16)$$

$$\text{HSCR-BF}_{grcos}: Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|)) * [-\cos(R)]\} \quad (17)$$

$$\text{HSGR-BF}_{+grcos}: Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|)) * [+ \cos(R)]\} \quad (18)$$

$$\text{HSCR-BF}_{grsin}: Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|)) * [-\sin(R)]\} \quad (19)$$

$$\text{HSGR-BF}_{+grsin}: Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|)) * [+ \sin(R)]\} \quad (20)$$

$$\text{HSCR-BF}_{Egrcos}: Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|)) * [-\cos(R)]\} \quad (21)$$

where ρ values of 50% and 67% were used. The nonlinear weight of $[\pm \cos(R)]$ were imposed on the Gaussian RBF (GRBF) as well before combined with the SBF as

$$Y(out) = \rho \log sig(R) + \{(1-\rho)(\exp(-|R|^2)) * [-\cos(R)]\} \quad (22)$$

The first and second derivatives were introduced to check the nonlinearity of each hybrid based on Eq. 17-20. To examine the Memory and Time Efficiency of the Algorithm, the 2nd order polynomial was used to assess the Performance among the Parameters i.e. CPU Usage, Memory and Time Efficiency, the 5th order was used for the assessment of the individual parameters within a particular Hybrid while the 6th order was used for Performance among hybrids and given as follows:

$$\begin{cases} Y_r = \beta_3 X_1^2 + \beta_2 X_1 + \beta_0 \\ Y_r = \beta_6 X_1^5 + \beta_5 X_1^4 + \beta_4 X_1^3 + \beta_3 X_1^2 + \beta_2 X_1 + \beta_0 \\ Y_r = \beta_7 X_1^6 + \beta_6 X_1^5 + \beta_5 X_1^4 + \beta_4 X_1^3 + \beta_3 X_1^2 + \beta_2 X_1 + \beta_0 \end{cases} \quad (23)$$

where x is time (seconds), β is the co-efficient of the polynomial

4. Particle Swarm Optimization

Particle swarm optimization (PSO), an evolutionary algorithm, is a population based stochastic optimization technique. The idea was conceived by an American researcher and social psychologist James Kennedy in the 1950s. The theory is inspired by social behavior of bird flocking or fish schooling. The method falls within the category of Swarm Intelligence methods for Solving Global Optimization problems. Literature has shown that the PSO is an effective alternative to the established evolutionary algorithms (GA). It is also established that PSO is easily applicable to real world complex problems with discrete, continuous and non-linear design parameters and retains the conceptual simplicity of GA (Kennedy & Eberhart, 1995), (Eberhart, Eberhart, & Shi, 2001). Each particle within the swarm is given an initial random position and an initial speed of propagation. The position of the particle represents a solution to the problem as described in a matrix τ , where M and N represent the number of particles in the simulation and the number of dimensions of the problem respectively (Malhotra & Negi, 2013), (Gies & Rahmat-Samii, 2003). A random position representing a possible solution to the problem, with an initial associated velocity representing a function of the distance from the particle's current position to the previous position of good fitness value were given. A velocity matrix v_{el} with the same dimensions as matrix τ_x described this.

$$\tau_x = \begin{pmatrix} \tau^{11}, \tau^{12}, \dots, \tau^{1N} \\ \tau^{21}, \tau^{22}, \dots, \tau^{2N} \\ \vdots \\ \tau^{m1}, \tau^{m2}, \dots, \tau^{MN} \end{pmatrix} \quad V_{el} = \begin{pmatrix} v^{11}, v^{12}, \dots, v^{1N} \\ v^{21}, v^{22}, \dots, v^{2N} \\ \vdots \\ v^{m1}, v^{m2}, \dots, v^{MN} \end{pmatrix}$$

While moving in the search space, particles commit to memory the position of the best solution they have found. At each iteration of the algorithm, each particle moves with a speed that is a weighted sum of three components: the old speed, a speed component that drives the particles towards the location in the search space, where it previously found the best solution so far, and a speed component that drives the particle towards the location in the search space where the neighbour particles found the best solution so far (Gies & Rahmat-Samii, 2003),

(Rajpal, Shishodia, & Sekhon, 2006). The personal best position can be represented by an NxN matrix (ρ_{best}) and the global best position is an N-dimensional vector G_{best} :

$$\rho_{best} = \begin{pmatrix} \rho^{11}, \rho^{12}, \dots, \rho^{1N} \\ \rho^{21}, \rho^{22}, \dots, \rho^{2N} \\ \vdots \\ \rho^{m1}, \rho^{m2}, \dots, \rho^{MN} \end{pmatrix}$$

$$G_{best} = (g_{best}^1, g_{best}^{12}, \dots, g_{best}^N)$$

All particles move towards the personal and the global best, with τ , ρ_{best} , V_{el} and G_{best} containing all the required information by the particle swarm algorithm. These matrices are updated on each successive iteration.

$$V_{mn} = V_{mn} + \gamma_{c1} \eta_{r1} (pbest_{mn} - X_{mn}) + \gamma_{c2} \eta_{r2} (gbest_n - X_{mn})$$

$$X_{mn} = V_{mn} \tag{24}$$

γ_{c1} and γ_{c2} are constants set to 1.3 and 2 respectively and η_{r1} η_{r2} are random numbers.

4.1 Adaptive mutation according to threshold

To prevent particles from not converging or converging at local minimum, an adaptive mutation according to threshold was introduced. Particles positions were updated with new value only when the new value is greater than the previous value. 20% of particles of those obtaining lower values were made to mutate for faster convergence (Ansong, Yao, & Huang, 2013). The input layer takes the final survival vector (Eq. 10), with a number of hidden layers and an output layer. The feed-forward neural network was used. The structure of Adaptive Mutation PSO (AMPSO) with threshold was used Figure 3, (Peng & Pan, 2011) and (Pantazis & Alevizakou, 2013).

5. Results and Discussion

5.1 The final survival vector

The $R = (0.5409 \ 0.4281 \ 0.8834 \ 0.585 \ 0.8130 \ 0.5163)$ from Eq 10 is the routing path with maximum survival probability for a total of 6 nodes deployed. It describes the success rate from each node to the sink(s). In most practical applications, more than one sinks are used, and sink node is either through the fiber or TTE connection. The size of the vector depends on the dimensions of the field. The elements $R = (0.5409 \ 0.4281 \ 0.8834 \ 0.585 \ 0.8130 \ 0.5163)$ represent the probability of 54%, 43%, 88%, 59%, 81% and 52% success of each node transmitting data to and from its source or destination. It assists decision makers as to whether to send data through one or more nodes, or send each message twice. The total nodes used for the simulation was 300 with underground mine dimensions of L=10, R=6 and C=5 for depth (level), row (length), and width (column) respectively with ‘pm’ a sensor apart, pm=100, $M_\rho=4$, and $Ne=2$. The PSO training used swarm size of 20, maximum position was set to 100, max velocity =1, number of neurons = 6 and maximum number of iteration =250. The thresholds λL and λH were 3 and 6 respectively, $\tau =6$ cases-thus each matrix and vectors were run 6 times before neural training.

The top half of each figure (Figure 4) indicates the optimized error or the final error after the neural training and the bottom half reveals model survival probability. The survival probability indicates that the model survived between 90-100% where rock cases was relatively soft (≥ 0.7). The survival probability declines as the rock becomes harder and approaches 0.8. At the hardest rock of 0.9, the survival probability fell between 72- 84% for the entire hybrids. In view of this the AP heads had to be deployed at a location where the rock is relatively soft for maximum signal strength. Each computer simulation incorporated all the 6 different cases of rock hardness/softness to produced the matrices. Figure 4 (a-e) represent different scenarios of SBF and CRBF Hybrids with the $[\pm \cos(R)]$ $[\pm \sin(R)]$ nonlinear weights while (4f) represent the SBF and Gaussian Hybrids for $[+\cos(R)]$ nonlinear weight. Detailed location analysis or the scalability of the model in relation to survival probability range, robot location and different rock types are recorded in Tables 5.

5.2 Training results

Figure 4 again discusses the G-ratio of 67% splenium for SBF and 33% CRBF hybrids (HSCR-BF_{gr}) with negative and positive nonlinear weights of cosine a) HSCR-BF_{-grcos}, b) HSCR-BF_{+grcos}. From the training the HSCR-BF_{-grcos} had a steady and compact routing path which was consistent through all rock layers, with initial survival probability between 87.7%-100% in soft layers declining to 69.5%-87.4% at harder rock layers. The HSCR-BF_{+grcos} was more dispersed with 87.9% -97.8% at the soft rock and 71.9%-88.9% at the hard rock layers. the positive and negative nonlinear weights with sine c) HSCR-BF_{-grsin}, d) HSCR-BF_{+grsin} showed, HSCR-BF_{-grsin} streamed well at both the initial and middle stages from 87.9%-98.5% at soft rock layers and 66.5% -0.83.6% at hard rock layers with a steady and compact routing path. The HSCR-BF_{+grsin} was also consistent through the first four rock layers with initial survival probability between 90%-100% in soft layers declining to 71% -85.4% at harder rock layers, with lessened transmitting probability through the last two layers. This could hamper rescue mission due to battery drain or collision from traffic congestion.

However it could be used in areas where routing conditions are much better such as rescue situations on surface. The survival probability of the negative nonlinear weight Gaussian hybrids e) HSCR-BF_{-Ggrcos} is compact both at the initial (i.e. 87.3%-99.6%) and latter stages (70.98%-83.2%). The f) HSCR-BF_{-Egrcos}, was similar to that of sine with a narrowed transmission part at the last two stages of rock type. The final errors read from the fitness function of Figures 4 indicated an average optimised errors of 0.011, 0.0121, 0.01208, 0.0129, 0.216 and 0.0117 from (Figures 6-11,calculated) for HSCR-BF_{-grcos}, HSCR-BF_{-grsin}, HSCR-BF_{+grsin}, HSCR-BF_{+grcos}, HSCR-BF_{-Egrcos}, and HSCR-BF_{-Ggrcos}, respectively as against the target error of 0.01. Earlier results for the optimised error of SBF and CRBF were 0.018 and 0.011 respectively [39]. Subsequently an equal g-ratio of both SBF and CRBF with nonlinear weight of negative cosine was performed resulting in an optimised error 0.0216. At the initial positions (Figure 5a) particles were sensitive to inputs as they moved quickly in the search space towards the target using more of the SBF but as particles peaked closer to the target they became less sensitive and more CRBF were used to keep the error at minimum for accuracy. Figure 5(b-e) further indicated the regions in which most of the nonlinear occurred as follows; (b) HSCR-BF_{-grcos} at region A and B, (c) HSCR-BF_{+grcos} at C and D, (d) HSCR-BF_{-grsin} at E and F and (e) HSCR-BF_{+grsin} at G and H respectively.

5.3 Performance of Parameters of the Various Hybrids

The trend of the various parameters of each hybrid were analysed. Each hybrid was run 10 times and the average performance were recorded. The 6th order polynomial was used to assess the performance trend of all the parameters i.e. mean iteration (MI), standard variance (Std Var), standard deviation (Std Dev), and Convergent time (Conver. Time) for each hybrid (Figures 6-11) and (Table 3). The performance of HSCR-BF_{-grcos} was the best with R² of 0.9613, 0.9074, 0.8745, 0.9452 and 0.5730 for final error (F.ERR), standard deviation (Std Dev), standard variance (Std Var), mean iteration (MI), and convergent time (Conver. Time) respectively (Table 3). The HSCR-BF_{+grcos}, the HSCR-BF_{-grsin} HSCR-BF_{+grsin} followed in decreasing order of performance. It must be noted that detailed work on SBF, CRBF and GRBF, with regards to scalability, memory usage and the central processing time has been carried by (Ansong, Yao, & Huang, 2013) and a minimum error of 0.011 for CRBF as against 0.012 and 0.0168 for GRBF and SBF respectively. However the performance of HSCR-BF_{-grcos} has proving to be more enhanced as compared with the previous work despite the same optimised error. The scalability of the Algorithm, CPU Usage and Time Efficiency (Figure 12-13) was examined using 5th order polynomial and performance among the hybrids using 2nd order polynomial (Table 4).

The relationship between the various hybrids with respect to the central processing Unit (CPU), was profiled for different runs to assess its scalability (Table 5). The proposed hybrid has better usage of CPU time and memory and optimised all its parameters with higher R² of 1.

5.3 Conclusion

In summary, we made the following contributions: first we used the mix of SBF and CRBF (HSCR-BF) to present several hybrids with different G-ratio nonlinear weights of cosine and sine functions on the CRBF. We have discussed the performances of the Proposed G-ratio nonlinear weight hybrids (HSCR-BF) with their corresponding errors as 0.011, 0.013, 0.01208, 0.0128, and 0.216 for HSCR-BF_{-grcos}, HSCR-BF_{+grcos}, HSCR-BF_{-grsin}, HSCR-BF_{+grsin}, HSCR-BF_{-Egrcos} respectively as compared to 0.0117 for the Gaussian HSCR-BF_{+grcos}. The CPU time, Memory usage and assessment among the various hybrids examined yielded R² values of 1. HSCR-BF_{-grcos} among the other hybrids indicated a better performance of individual parameters. The proposed Nonlinear Hybrid Algorithm with Particle swarm optimisation has better capability of approximation to underlying functions with a fast learning speed, high scalability and robusticity and is competitive to the Gaussian with the same nonlinear weight. Further investigation into the hybrids will be made as part of our future work.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (no. 71271103) and by the Six Talents Peak Foundation of Jiangsu Province.

We would like to appreciate the immense contribution of the mining companies where the study was undertaken. We are grateful to these individuals for their immeasurable contributions to this work: Fred Attakuma, Patrick Addai, Isaac Owusu Kwankye, Thomas Kwaw Annan, Willet Agongo, Nathaniel Quansah, Francis Owusu Mensah, Clement Owusu-Asamoah, Joseph Adu-Mensah, Shadrack Aidoo, Martin Anokye, F.T. Oduro, Ernest Ekow Abano, E. Omari-Siaw and Qiang Jia.

References

- Ahuja, R. K., Ravindra, K., Orlin, J., Pallottino, S., & M., S. (n.d.). Dynamic Shortest Paths Minimizing Travel Times and Costs. *MIT Sloan Working Paper*.
- Ansong, M. O., Yao, H.-X., & Huang, J. S. (2013). Radial and Sigmoid Basis Functions Neural Networks in Wireless Sensor Routing Topology Control in Underground Mine Rescue Operation based on Particle Swarm Optimization. *International Journal of Distributed Sensor Networks-Hindawi*, DOI 10.1155/2013/376931. doi:10.1155/2013/376931
- Broomhead, D., & Lowe, D. (1988). [8] D.S. Broomhead, D. Lowe, "Multivariate functional interpolation and adaptive networks," *Complex System* 2, pp. 321-355, 1988. *Complex System*, 2, 321-355.
- Chen, Y., Chuah, C.-N., & Zhao, Q. (2008). Network configuration for optimal utilization efficiency of wireless sensor networks. *Ad Hoc Networks*, 6, 92-107.
- Delacour, J. (1994). Introduction: The Memory System Of The Brain. World Scientific.
- Doty, R. W., & Ringo, J. L. (1994). *The Memory System of The Brain (Hemispheric Distribution Of Memory Traces)*. (Vol. 5). : Pp. 636-656. doi:Doi: 10.1142/9789814354752_0016
- Eberhart, J., Eberhart, R., & Shi, Y. (2001). Particle Swarm Optimization: Developments, Applications and Resources. *IEEE*, 1, 81-86.
- Feng, X., Xiao, Z., & Cui, X. (2011). Improved RSSI for wireless Sensor Network in 3D. 7(16).
- Fernández-Navarro, F., Herva´s-Martí´nez, G. r., A., P., Pen˜a-Barrag˜an, J. M., & L´opez-Granados, F. (2012). Parameter estimation of q-Gaussian. Radial Basis Functions Neural Networks with a Hybrid Algorithm for binary classification. *Neurocomputing*, 75, 123-134.
- Fernández-Navarro, F., Herva´s-Martí´nez, C. s., Sanchez-Monedero, J., & Guti´errez, P. A. ((2011)). MELM-GRBF: A modified version of the extreme learning machine for generalized radial basis function neural networks. *Neurocomputing*, 74, 2502-2510.
- Fischl, B. V. (2004). Automatically parcellating the human cerebral cortex. *Cerebral Cortex*, 14(1), 11-22.
- Gies, D., & Rahmat-Samii, Y. (2003, August 3). Particle Swarm Optimization for Reconfigurable phase differentiated Array design. *2003 Wiley Periodicals, Inc.*, 38, pp. 168-175.

- Goh, S. L., & Mandic, D. P. (2007). An augmented CRTRL for complex-valued recurrent neural networks. *Neural Networks*, 20, 1061–1066.
- Halgren, E. (1994). *Physiological Integration Of The Declarative Memory System*. World Scientific.
- Hidayat, M. I., & Ariwahjoedi, B. (2011). Model of Neural Networks with Sigmoid and Radial basis functions for Velocity-field Reconstruction in Fluid-structure Interaction problem. *Journal of Applied Science*, 11(9), 1587-1593.
- Jang, W.-S., Healy, W. M., & Mirosław, S. J. (2008). Wireless sensor networks as part of a web-based building environmental monitoring system. *Automation in Construction*, 17, 729–736.
- Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. *IEEE, Proceedings of the IEEE International Conference on Neural Networks, 1995*, pp, 1942-1945.
- Kumar, C. K., Sukumar, R., & Nageswari, M. (2013). Sensors Lifetime Enhancement Techniques in Wireless Sensor Networks - A Critical Review. *International Journal of Computer Science and Information Technology & Security (IJCSITS)*, 3, 159-164.
- Leblecioglu, K., & Halici, U. (1997). Infinte Dimension radial Basis function neural networks for nonlinear transformations on function spaces. *Nonlinear Analysis, theory, Methods and Application*, 30(3), 1649-1654.
- Li, H., & Yang, X. G. (2012, December). Fault Diagnosis System of Hydraulic System Using Lab Windows/CVI and Virtual Instrument Technology. *International Journal of Applied Science and Technology.*, 2, 14-20.
- Li, S., & Qin, F. (2013, October). A Dynamic Neural Network Approach for Solving Nonlinear Inequalities Defined: on A Graph and Its Application to Distributed, Routing-free, Range-free Localization of WSNs. *Neurocomputing*, 117, 72-80.
- Li, S., Chen, S., Liu, B., Li, Y., & Liang, Y. (2012). Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks. *Neurocomputing*, 91, 1-10.
- Li, S., Cui, H., Li, Y., Liu, B., & Lou, Y. (2012). Decentralized Control of Collaborative Redundant Manipulators with Partial Command Coverage via Locally Connected Recurrent Neural Networks. *Neural Computing and Application*. doi:DOI 10.1007/s00521
- Li, S., Li, L., & Yang, Y. (2011). A local-world heterogeneous model of wireless sensor networks with node and link diversity. *Physica A*, 390, 1182–1191.
- Li, S., Wang, Z., & Li, Y. (2013). Using Laplacian Eigenmap as Heuristic Information to Solve Nonlinear Constraints Defined on A Graph and Its Application in Distributed Range-free Localization of Wireless Sensor Networks,. *Neural Processing Letters*, 37, pp. 411–424.
- Liu, Y., & Luo, Z. (2012). A Bi-Level Model for Planning Signalized and Uninterrupted Flow Intersections in an Evacuation Network. *Computer-Aided Civil and Infrastructure Engineering*, 27, 731–747.
- Malhotra, R., & Negi, A. (2013). Reliability modeling using Particle Swarm Optimisation. *Int.J. Syst Assur Eng Manag.*, doi:DOI 10.1007/s13198-012-0139-0
- Munoz, L. A., & Ramosy, J. J. (2007). *Similarity-based Heterogeneous Neural Networks*. Advance online publication.
- Neruda, R., & Kudova, P. (2005). Learning methods for radial basis function networks. *Future Generation Computer Learning Systems and Applications*, 21(7), 1131-1142.
- Pan, M.-S., Tsai, C.-H., TsengTsai, Y.-C., & Tseng, Y.-C. (2006). Emergency guiding and monitoring application in indoor 3D environment by Wireless Sensor Network. *Internation Journal os Sensor Netwourk*, 1, 2-10.
- Pantazis, G., & Alevizakou, E.-G. (2013). The Use of Artificial Neural Networks in Predicting Vertical Displacements of Structures. *International Journal of Applied Science and Technology*3(5), 3.
- Peng, J., & Pan, C. P. (2011). Particle swarm optimization RBF for gas emission prediction. *Journal of Safety Science and Technology* 11. 2011; 7: 77-05. *Journal of Safety Science and Technology*, 11, 77-85.
- Rajpal, P., Shishodia, K., & Sekhon, G. (2006). An artificial network for modeling reliability, availability and maintainability of a repairable system. *Reliability Engineering and System Safety*, 9, 809-819.
- Ren, G., Huang, Z., Cheng, Y., Zhao, X., & Zhang, Y. (2013). An integrated model for evacuation routing and traffic signal optimization with background demand uncertainty. *Journal of Advanced Transportation*, 47, 4–27.
- Riaz, R., Naureen, A., Akram, A., Akbar, A. H., Kim, K.-H., & Ahmed, H. F. ((2008). A unified security framework with three key management schemes for wireless sensor networks. *Computer Communications*, 31, 4269–4280.

Sausen, P. S., Spohn, M. A., & Perkusich, A. (2010). Broadcast routing in wireless sensor networks with dynamic power management and multi-coverage backbone. *Information Sciences*, 180, 653–663.

Shi, J., & Wang, S. (2007). Integrated availability model based on performance of computer networks. *Reliability Engineering and System Safety*, 92, 341–350.

Simplicio, M. A., Barreto, P. S., Margi, C. B., & Carvalho, T. C. (2010). A survey on key management mechanisms for distributed Wireless Sensor Networks. *Computer Networks*, 54, 2591–2612.

Soh, H., Lima, S., Zhanga, T., Fua, X., Kee, G., Lee, K., . . . Wong, L. (2010). Weighted complex network analysis of travel routes on the Singapore public transportation system. *Physica A*, 389, 5852–5863.

Tan, Q., Huang, G., Wu, C., & Cai, Y. (2011). IF-EM: An interval-parameter fuzzy linear programming model for environment oriented evacuation planning under uncertainty. *Journal of Advanced Transportation*, 45(4), 286–303.

Valdez, F., & Melin, P. (2008). Neural Network Optimization with a Hybrid Evolutionary Method that combines Particle Swarm and Genetic Algorithms with Fuzzy Rules. *Annual Meeting of the Fuzzy Information Processing Society, NAFIPS 2008*, (pp. 19-22). North American.

Wang, Y.-H., Zhou, Z., Liu, L., & Wu, W. (2012). Fault Tolerance and Recovery for Group Communication Services in Distributed Networks. *Journal Of Computer Science And Technology*, 27, 298 Doi 10.1007/S11390-012-1224-1.

Zarifzadeh, S., Nayyeri, A., & Yazdan, N. (2008). Efficient construction of network topology to conserve energy. *Computer Communications*, 31, 160-173.

Zhang, H. H., Genton, M., & Liu, P. (2004). Compactly supported radial basis function kernels. Retrieved from www4.stat.ncsu.edu/hzhang/research.html (2004).

Table 1 Nomenclature

Nomenclature	
HSCR-BF _g /HSGR-BF _g	- G-ratio Hybrid of SBF with CRBF/GRBF
HSCR-BF _{-grcos}	- G-ratio Hybrid of SBF and CRBF with negative nonlinear cosine weight
HSCR-BF _{+grcos}	- G-ratio Hybrid of SBF and CRBF with positive nonlinear cosine weight
HSCR-BF _{-grsin}	- G-ratio Hybrid of SBF and CRBF with negative nonlinear sine weight
HSCR-BF _{+grsin}	- G-ratio Hybrid of SBF and CRBF with positive nonlinear sine weight
HSGR-BF _{-Egrcos}	- Equal G-ratio Hybrid of SBF and CRBF with negative nonlinear cosine weight
HSGR-BF _{-Ggrcos}	- G-ratio Hybrid of SBF and GRBF with positive nonlinear cosine weight

Table 2: Common rocks found in typical mines in relation to hardness or softness

Nonlinear mapping	Mica	Coal	Granite	Feldspar	Quartz	Mineral
Softness	0.70	0.80	0.83	0.86	0.875	0.90
Hardness	2	3	5	6	7	9
Distance	750 m	470 m	390 m	315 m	278	78 m

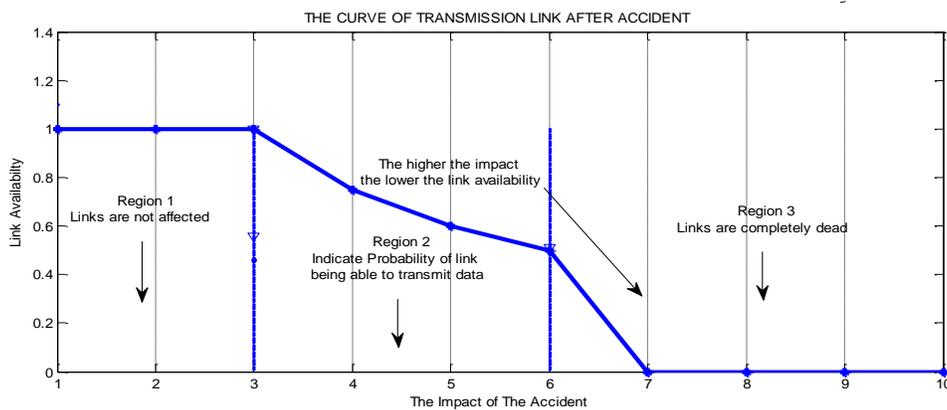


Figure 1: Impact on Transmission link after accident

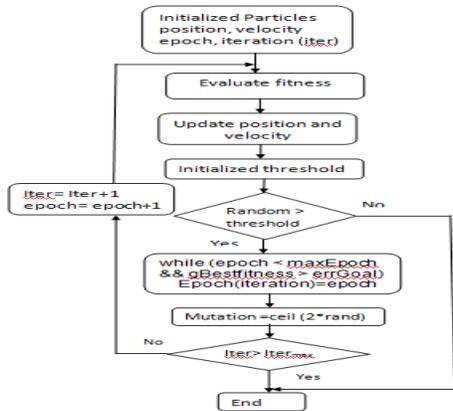


Figure 2: The structure of Adaptive PSO with threshold

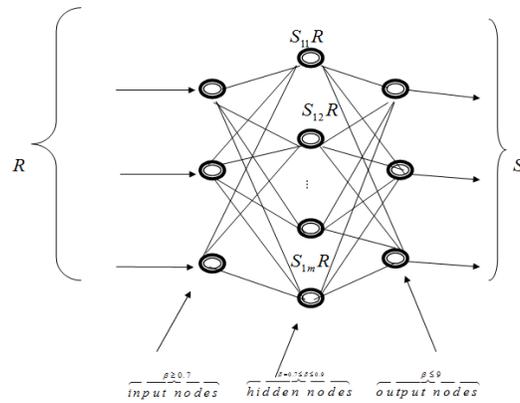


Figure 3 The structure AMPSO for CRBF, Mutation PSO (AMPSO) GRBF and SBF neural transfer functions

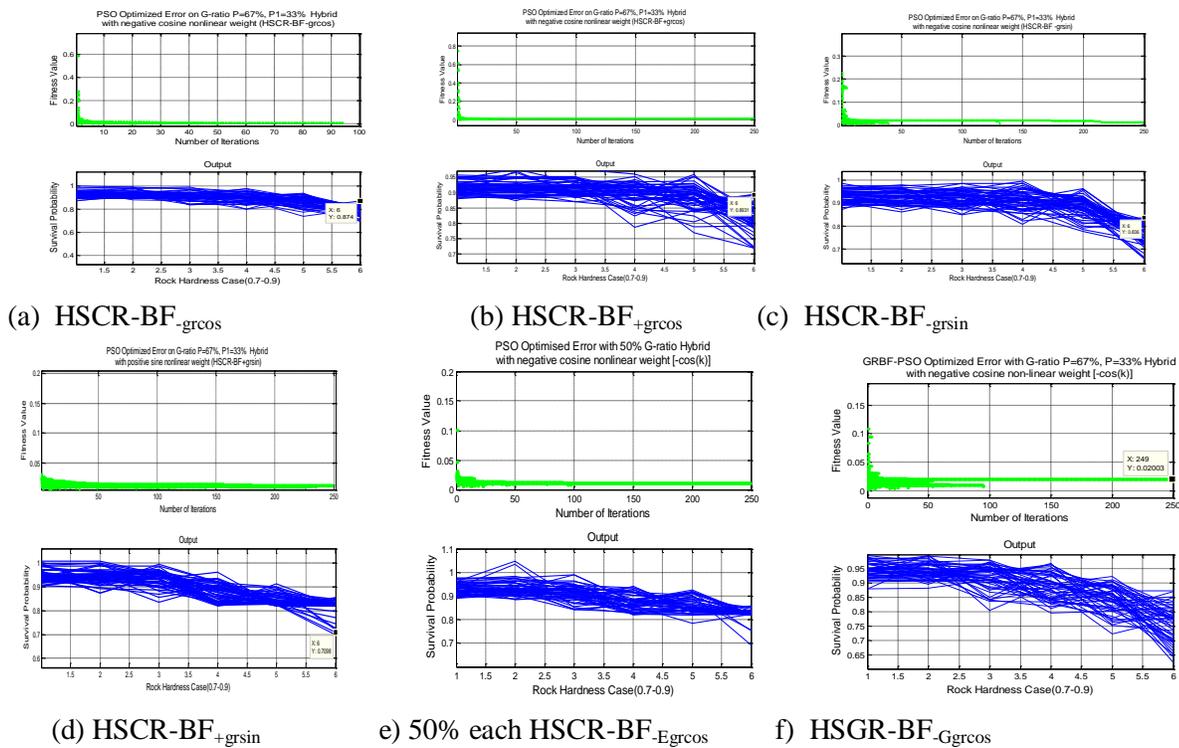


Figure 4: Hybrid of SBF and CRBF (HSCR-BF) with nonlinear weights of on the g-ratio for (a) HSCR-BF_{-grcos} , (b) HSCR-BF_{+grcos} , (c) HSCR-BF_{-grsin} , (d) HSCR-BF_{+grsin} , e) 50% each HSCR-BF_{-Egrcos} and f) HSGR-BF_{-Ggrcos}

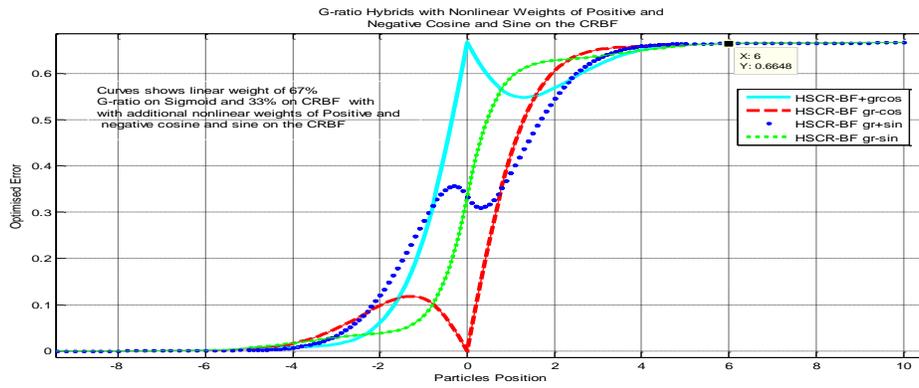
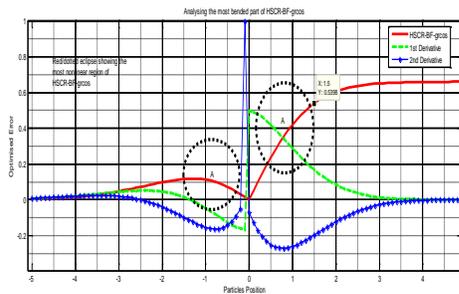
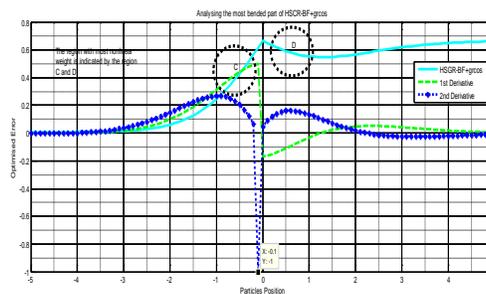


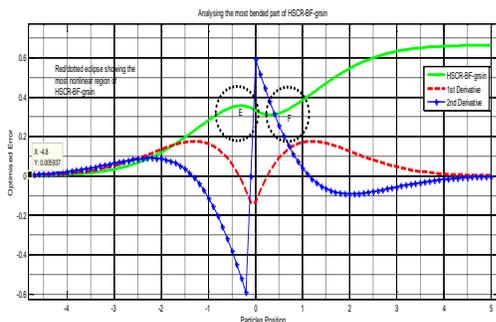
Figure 5 (a) Nonlinear weighted curves for optimized error-HSCR-BF_{-grcos}, HSCR-BF_{+grcos}, HSCR-BF_{-grsin}, HSCR-BF_{+grsin}



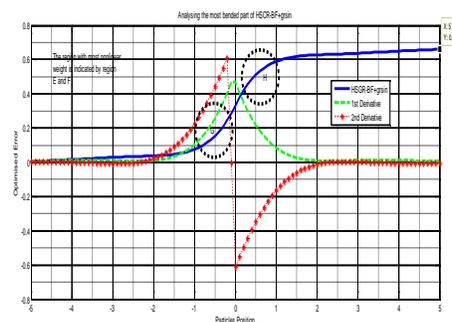
b) HSCR-BF_{-grcos}



c) HSCR-BF_{+grcos}



d) HSCR-BF_{-grsin}



e) HSCR-BF_{+grsin}

Figure 5 Nonlinear weighted curves for optimized error-HSCR-BF_{-grcos}, HSCR-BF_{+grcos}, HSCR-BF_{-grsin}, HSCR-BF_{+grsin} indicating the most nonlinear regions.

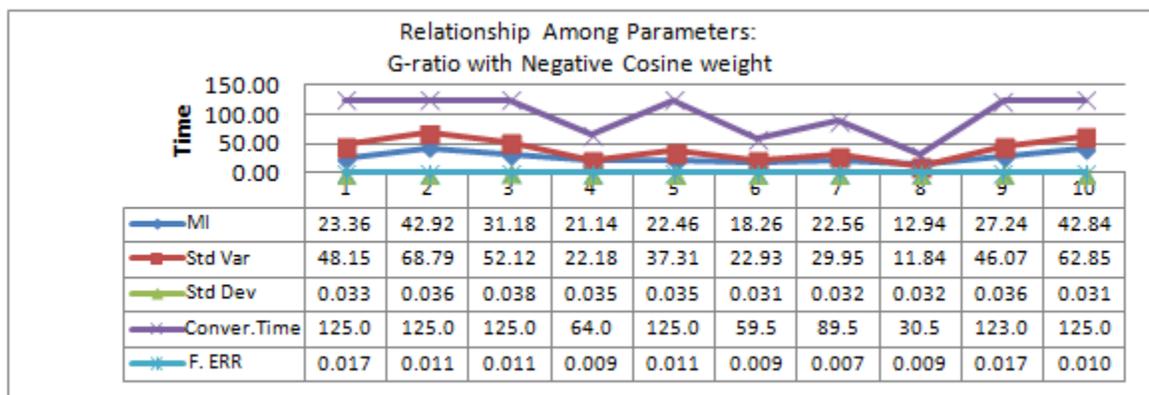


Figure 6 Relationship Among Parameters: G-ratio with Negative Cosine weight (HSCR-BF_{-grcos})

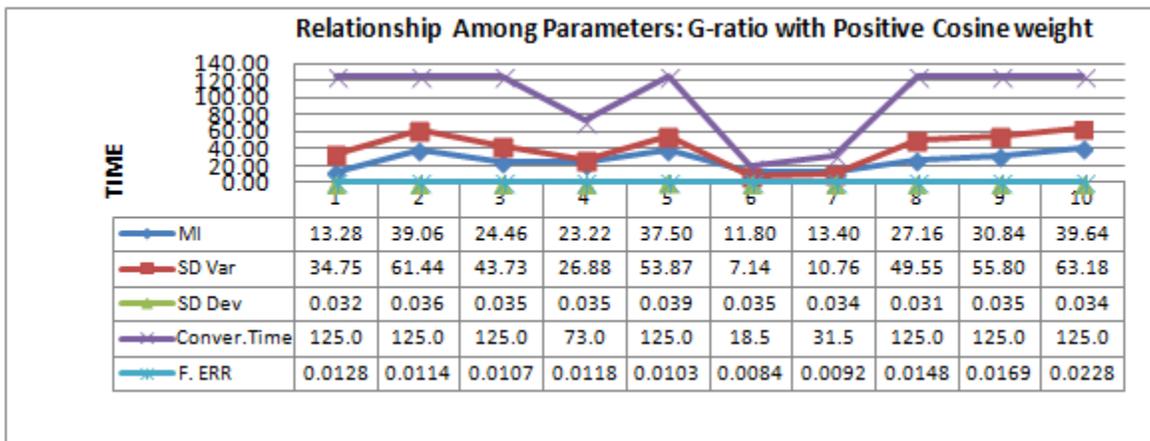


Figure 7: Relationship Among Parameters: G-ratio with Positive Cosine weight (HSCR-BF_{+grcos})

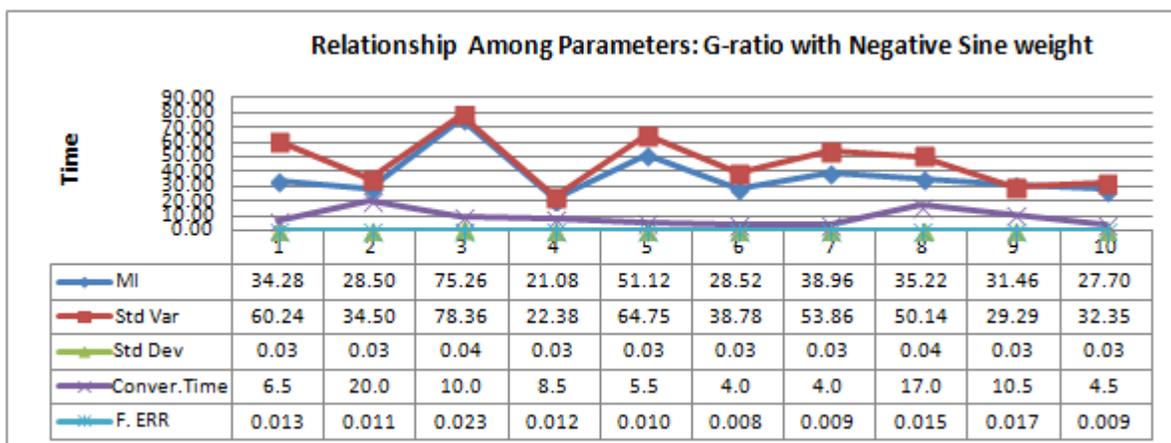


Figure 8: Relationship Among Parameters: G-ratio with Negative Sine weight (HSCR-BF_{-grsin})

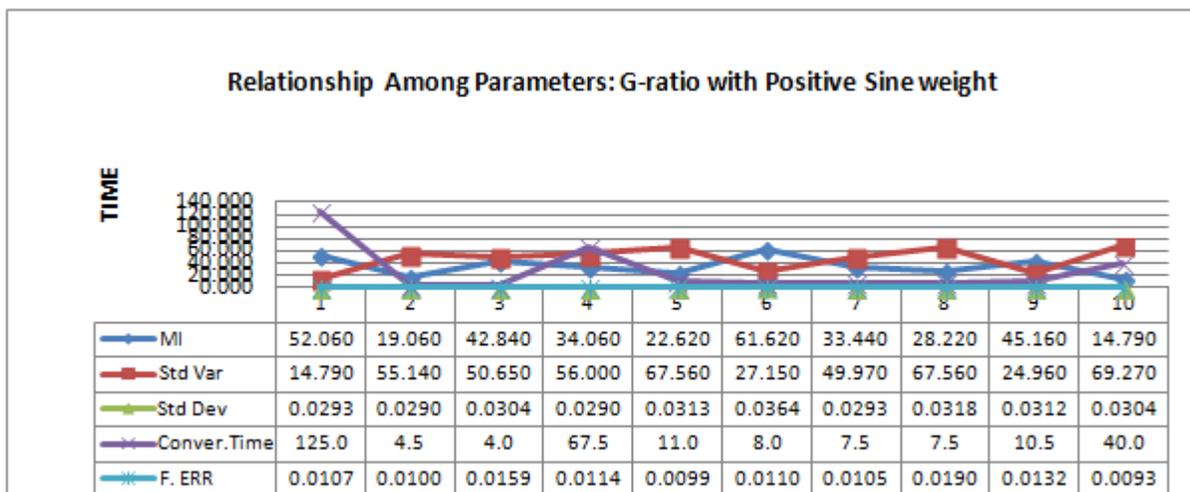


Figure 9: Relationship Among Parameters: G-ratio with Positive Sine weight (HSCR-BF_{+grsin})

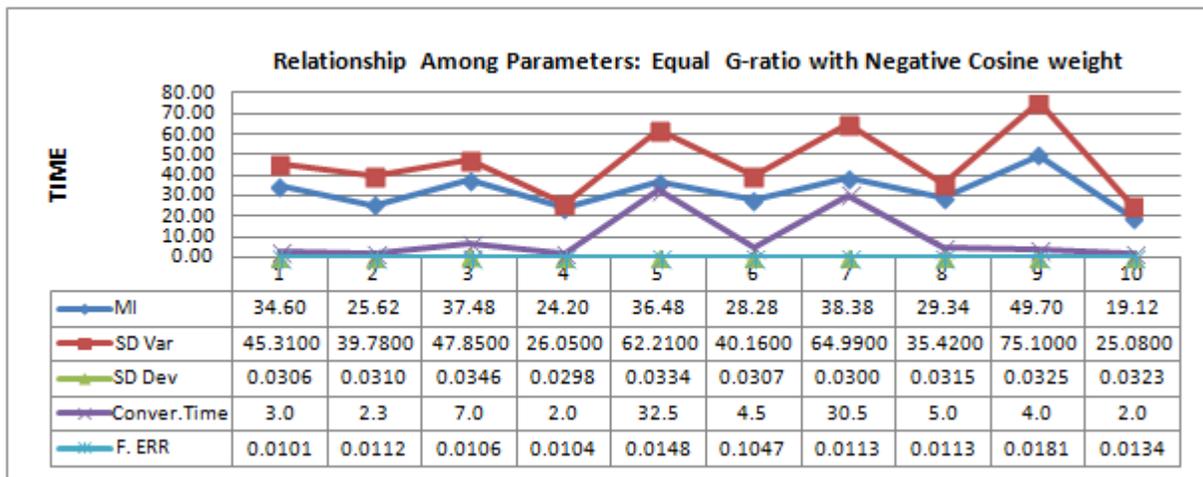


Figure 10 Relationship Among Parameters: Equal G-ratio with Negative Cosine weight (HSCR-BF_{+Egrcos})

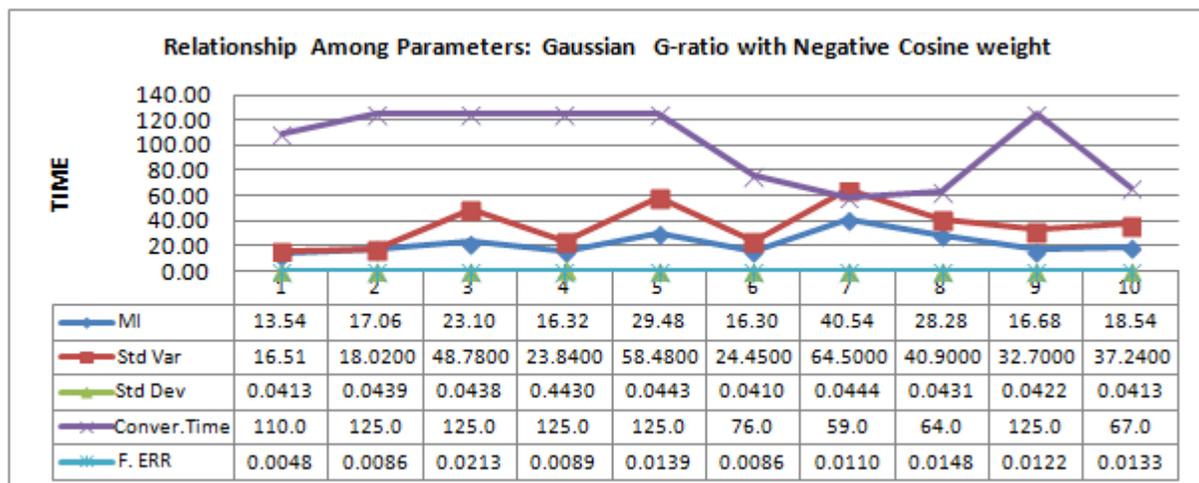


Figure 11 Relationship Among Parameters: Gaussian G-ratio with Negative Cosine weight (HSCR-BF_{-Ggrcos})

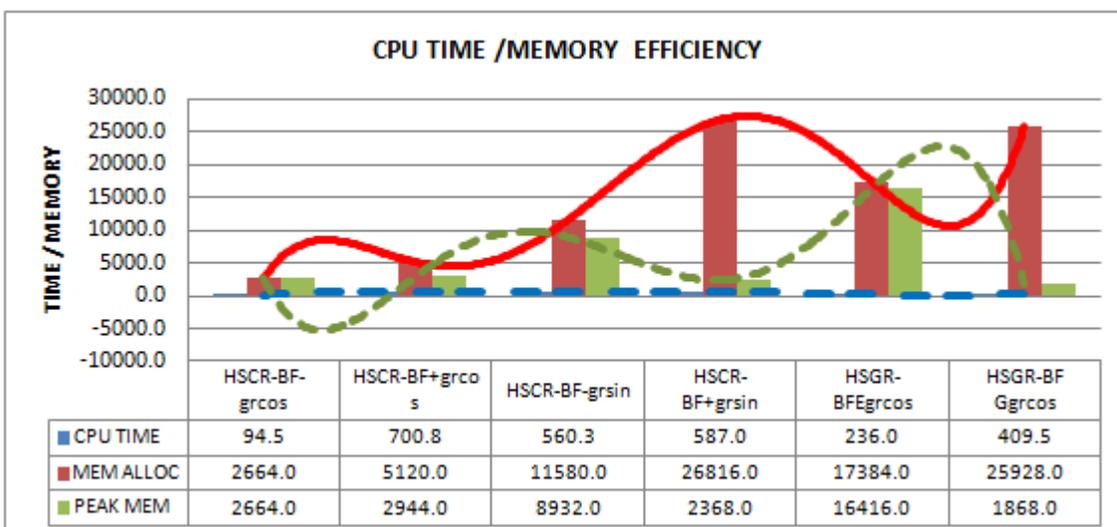


Figure 12 CPU Time and Memory Usage

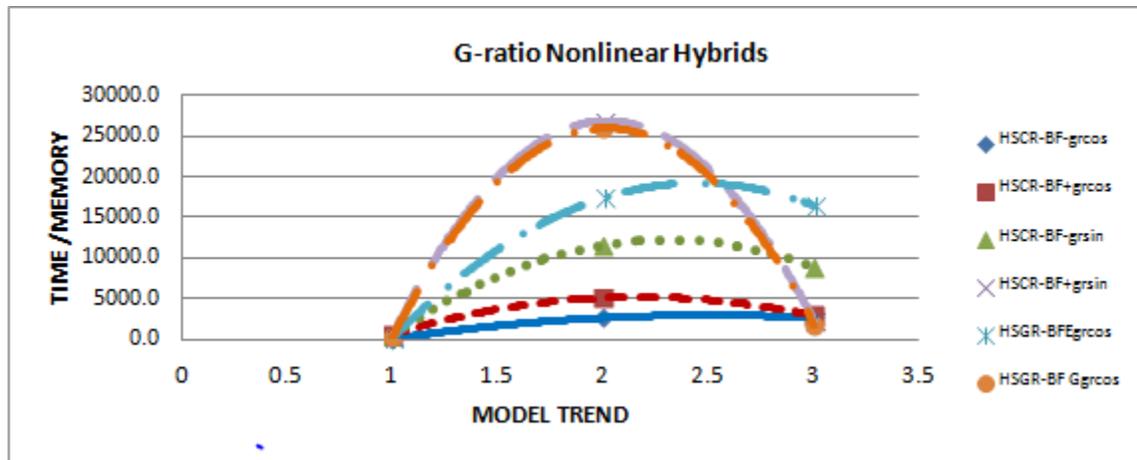


Figure 13 Usage Trend CPU and Memory in G-ratio Nonlinear Hybrids

Table 3 Future trend Among Parameters of the G-ratio Hybrids with Nonlinear Weights of Cosine and Sine/Gaussian

	B ₆	B ₅	B ₄	B ₃	B ₂	B ₁	B ₀	R ²
HSCR-BF_{-grcos}								
Final Error	-6E-06x	0.0002x	- 0.002x	0.0105x	- 0.0261x	0.0247x	0.0096	0.9613
SD Deviation	-2E-06x	6E-05x	- 0.0007x	0.0037x	0.0037x	0.0199x	0.0213	0.9074
SD Variance	-0.0319x	1.1009x	- 14.859x4	99.403x	- 340.75x	541.93x	-239.26	0.8745
Mean Iteration	-0.0136x	0.4876x	- 6.8685x	48.084x	- 173.25x	292.09x	-137.27	0.9452
Conv. Time	-0.0679x	2.2183x	- 28.083x	174.03x	- 546.16x	790.03x	-268.9	0.5730
HSCR-BF_{+grcos}								
Final Error	2E-06x	- 7E-05x	0.001x	- 0.0066x	0.0228x	- 0.0365x	0.0323	0.9655
SD Deviation	-7E06x	0.0002x	- 0.003x	0.0185x	- 0.059x	0.0898x	-0.0147	0.8594
SD Variance	-0.0226x	0.7192x	- 9.0162x	57.009x	- 190.97x	310.14x	-132.19	0.6407
Mean Iteration	-0.0187x	0.6173x	- 8.0103x	51.844x	-175.02	286.11x	-141.55	0.6405
Conv. Time	0.0015x	- 0.0923x	2.9023x	- 26.944x	101.4x	- 161.69x	210.7	0.6108
HSCR-BF_{-grsin}								
Final Error	8E-06x	- 0.0003x	0.0036x	- 0.0235x	0.0745x	- 0.1044x	0.0624	0.7439
SD Deviation	5E-06x	- 0.0002x	0.0024x	- 0.0156x	0.0503x	- 0.0745x	0.0716	0.3282
SD Variance	0.034x	- 1.1149x	14.194x	- 88.688x	281.44x	- 419.61x	272.6	0.3126
Mean Iteration	0.0246x	- 0.8196x	10.555x	- 65.793x	202.22x	- 275.56x	162.12	0.2651
Conv. Time	-0.0023x	0.0857x	- 1.3349x	10.953x	- 47.325x	94.044x	-49.55	0.7512
HSCR-BF_{+grsin}								
Final Error	8E-06x	- 0.0003x	0.0036x	- 0.0227x	0.0706x	- 0.0994x	0.0588	0.7481
SD Deviation	-4E-06x	0.0001x	- 0.0016x	0.0101x	- 0.0311x	0.044x	0.0077	0.4402
SD Variance	0.03x	- 0.9298x	11.028x4	- 61.998x	164x	- 166.01x	70.317	0.5661
Mean Iteration	-0.0045x	0.0988x	- 0.5152x	- 2.8464x	35.023x	- 103.58x	122.32	0.3865
Conv. Time	0.0516x	- 1.8653x	26.746x	- 192x	714.08x	- 1274.3x	853.57	0.9000
HSGR-BF_{-Ggrcos}								
Final Error	9E-06x	- 0.0003x	0.0037x	- 0.0225x	0.0666x	- 0.0832x	0.0403	0.5892
SD Deviation	0.0002x	- 0.0061x	0.0865x	- 0.6012x	2.0837x	- 3.2299x	1.718	0.4696
SD Variance	0.0255x	- 0.8207x	10.247x	- 62.406x	190.23x	- 258.14x	136.84	0.4424
Mean Iteration	0.0148x	- 0.4608x	5.474x	- 31.31x	88.816x	- 112.6x	63.645	0.5954
Conv. Time	-0.0578x	1.7373x	- 19.957x	111.01x	- 314.81x	431.93x	- 99.9	0.9777

Table 4 Performance among the Parameters and among the Hybrids

	B ₅	B ₄	B ₃	B ₂	B ₁	B ₀	R ²
	Performance among the Parameters						
CPU TIME	24.22x	- 424.1x	2819x	- 8865x	13079x	-6538.3	1
MEM ALLOC	952.5x	- 15880x	97684x	- 272822x	345809x	-153080	1
PEAK MEM	-1115x	18867x	- 119242x	346975x	- 454393x	211572	1

	Performance among the Hybrids			
HSCR-BF_{-grcos}	-1284.8x	6423.7x	- 5044.5	1
HSCR-BF_{-grsin}	-3297.6x	14312x	- 10314	1
HSCR-BF_{+grcos}	-6833.9x	31521x	- 24127	1
HSCR-BF_{+grsin}	-25338x	102245x	- 76319	1
HSCR-BF_{-Egrcos}	-9058x	44322x	- 35028	1
HSGR-BF_{+grcos}	-24789x	99886x	- 74687	1

Table 5: Scalability of model to survival probability range, robot location and rock type

SBF AND CRBF G-ratio Hybrid with negative nonlinear cosine weight (HSCR-BF _{-grcos})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.8769-1	0.9008-0.9911	0.8402-0.9943	0.8025-0.9776	0.7386-0.9376	0.695-0.874
(10,5,4)	0.8534-0.9793	0.8426-0.9961	0.8435-0.9342	0.7752-0.9717	0.7423-0.9402	0.6591-0.8552
(6,5,4)	0.8729-0.9849	0.8576-0.9156	0.8483-0.9756	0.7868-0.9684	0.714-0.942	0.6194-0.8043
(3,1,10)	0.7712-0.9233	0.6991-0.8349	0.4848-0.7066	0.4778-0.6731	0.4547-0.6120	0.3536-0.5146
SBF AND CRBF G-ratio Hybrid with positive nonlinear cosine weight (HSCR-BF _{+grcos})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.8792-0.9782	0.8734-0.9765	0.8702-0.9782	0.7876-0.7593	0.7688-0.9581	0.7196-0.8886
(10,5,4)	0.8945-1	0.8864-0.9974	0.8619-0.9988	0.8142-0.9649	0.8242-0.9364	0.8005-0.8922
(6,5,4)	0.901-0.9924	0.8216-0.989	0.8803-0.9666	0.8234-0.9637	0.8286-0.9049	0.6428-0.8296
(3,1,10)	0.7686-0.8818	0.6934-0.8258	0.6923-0.8644	0.5816-0.6106	0.5304-0.6477	0.4441-0.6011
SBF AND CRBF G-ratio Hybrid with negative nonlinear sine weight (HSCR-BF _{-grsin})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.8794-0.9847-	0.8706-0.9777	0.8498-0.9846	0.8068-0.9956	0.777-0.9617	0.6653-0.836
(10,5,4)	0.9079-0.9962	0.8492-0.9962	0.8278-0.9919	0.8324-0.9601	0.7583-0.9189	0.7428-0.8379
(6,5,4)	0.9241-0.9699	0.8646-1	0.913-0.975	0.8339-0.9242	0.8231-0.8585	0.8268-0.8303
(3,1,10)	0.716-0.8726	0.6555-0.7793	0.698-0.7628	0.6154-0.7529	0.4617-0.5766	0.419-0.4919
SBF AND CRBF G-ratio Hybrid with positive nonlinear sine weight (HSCR-BF _{+grsin})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.9014-0.1000	0.875-1.00	0.8349-0.9964	0.8221-0.961	0.8292-0.9121	0.7098-0.8495
(10,5,4)	0.8921-1000	0.8864-09974	0.8619-0.9988	0.8142-0.9649	0.8242-0.9364	0.8005-0.8922
(6,5,4)	0.901-0.9924	0.6305-0.9894	0.880-0.9666	0.8234-0.9637	0.8286-0.9049	0.6428-0.8296
(3,1,10)	0.6917-0.8046	0.6212-0.7269	0.6618-0.7145	0.5535-0.6961	0.6612-0.7163	0.3934-0.45
One-Half of SBF AND CRBF G-ratio Hybrid with negative nonlinear cosine weight (HSCR-BF _{-Egrcos})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.9025-0.975	0.888-0.9817	0.8516-0.9883	0.833-0.9412	0.7852-0.9262	0.8554-0.6922
(10,5,4)	0.9012-1	0.885-1	0.8289-1	0.8207-0.9352	0.7545-0.8675	0.774-0.8516
(6,5,4)	0.8813-0.9874	0.8665-0.9833	0.8716-0.9896	0.828-0.9402	0.7853-0.8912	0.7008-0.8379
(3,1,10)	0.755-0.9489	0.726-0.8184	0.6086-0.7573	0.6463-0.7646	0.5534-0.6696	0.4192-0.5389
SBF AND GRBF G-ratio Hybrid with negative nonlinear cosine weight (HSGR-BF _{-Ggrcos})						
Location	Mica	Coal	Granite	Feldspar	Quartz	Mineral
(10,6,5)	0.8904-0.9856	0.8994-0.9827	0.829-0.9947	0.8226-0.9502	0.8271-0.9245	0.7005-0.86
(10,5,4)	0.9012-1	0.8855-1	0.8289-0.9996	0.8207-0.9352	0.736-0.8625	0.774 -0.8516
(6,5,4)	0.8813-0.9874	0.8665-0.9833	0.8718-0.9896	0.828-0.9402	0.7853-0.8912	0.7008-0.8379
(3,1,10)	0.755-0.9489	0.726-0.8184	0.6086-0.7573	0.6463-0.7646	0.5534-0.6696	0.4192-0.5389