Journal of Soft Computing and Artificial Intelligence
Research Article 2(2): pp:69-78 (2021),
Received: June 25, 2021 Accepted: August 30, 2021
JSCAI
ISSN: 2717-8226

Training Feedforward Neural Networks to Predict the Size of the Population by Using a New Hybrid Method Hestenes-Stiefel (HS) and Dai-Yuan (DY)

[1]Hisham M. Khudhur, [2]Khalil K. Abbo, [3]Aydin M. Khudhur

[1]Mathematics Department, College of Computer Science and mathematics, University of Mosul, Mosul, Iraq
[2]Department of Mathematics, College of Basic Education, University of Telafer, Tall'Afar, Iraq
[3]Department of Studies and planning, Presidency of telafer university, University of Telafer, Tall'Afar, Iraq

Corresponding author: First A. Author (e-mail:- hisham892020@uomosul.edu.iq).

ABSTRACT We proposed a new conjugate gradient type hybrid approach in this study, which is based on merging Hestenes-Stiefel and Dai-Yuan algorithms using the spectral direction conjugate algorithm, we showed their absolute convergence. Under some assumptions and they satisfied the gradient property. The numerical results demonstrate the efficacy of the developed feedforward neural network training approach. To estimate the size of the population using the Thomas Malthus population model, and Our numerical results were very close to the model of the Tomas Malthose Model, we can use the method to predict other problems through the use of ann.

KEYWORDS: Algorithms; ANN; Conjugate Gradient; Hybrid; Population.

1. INTRODUCTION

Multilayer feedforward neural networks (MLFFNN) are parallel computational models made up of densely interconnected, adaptable processing units that have an innate proclivity for learning from experience as well as discovering new information. They have been effectively employed in various domains of artificial intelligence [1],[2], [7], and [8] are often found to be more efficient and accurate than other classification techniques [17] due to their exceptional capability of self-learning and self-adapting. Feedforward neural networks (FNN) are often operated using the following equations:

$$net_j^i = \sum_{i=1}^{N_{L-1}} w_{i,j}^{i-1} x_j^{i-1} + b_j^i, O_j^l = f\left(net_j^l\right) \tag{1}$$

Where $f\ (net_j^l)$ is the activation function , net_j^l is the sum of the weight inputs for the j-th node in the l - th layer (j=1,2,…, Nl), $w_{i,j}$ is the weights from the i-th neuron to the j-th neuron at the $l-1, l-th$ layer ,respectively, b_j^l is the bias of the $j-th$ neuron at the $l-th$ layer and x_j^l is the output of the j-th neuron which belongs to the $l-th$ layer. The goal of training a neural network is to iteratively change its weights to minimize the difference between the network's actual output and the training set's desired output [26]. Finding such a minimum is actually the same as finding an optimal minimization of the error function, which is defined as:

$$E(w) = \frac{1}{2}\sum_{j=1}^{P} \sum_{i=1}^{M} \left(O_i^{(j)} - T_i^{(j)}\right)^2 \quad (2)$$

The variables O_i and T_i are the desired and the actual output of the $i-th$ neuron, respectively. The index j denotes the particular learning pattern. The vector w is composed of all weights in the net. The most extensively used approach for training multilayer feedforward neural networks is backpropagation (BP). The weight vector w is adjusted using the steepest descent with respect to E in the typical backpropagation algorithm:

$$w_{k+1} = w_k - \alpha_k g_k, g_k = \nabla E(w_k) \quad (3)$$

Where the constant α is the learning rate belongs to the interval (0,1) and w_k is a vector representing the weights at iteration (epoch) step k. The back propagation process takes an inordinate amount of time to modify the weights between the units in the network since the steepest descent method has a slow convergence rate and the search for the global minimum frequently becomes stranded at a bad local minimum[15]. As a result, many studies have proposed ways to improve this method, with several relying on a novel adaptive learning rate [4], [2], and [1]. Others utilize other cost functions or dynamic modification of the learning parameters [28], while others use the momentum term [27], [20], [13].Many people use weight initialization procedures that are unique to them [24]. The majority of them use higher order gradient optimization algorithms to reduce the appropriately error function [16], [22], a multivariable function that is dependent on the network's weights. However, the issue of speeding up the learning process remains. Especially when using big training sets and networks. The training of neural networks can be expressed as a non-linear unconstrained optimization problem [23], [3], [14].

The following is a breakdown of how this search is structured. The conjugate gradient algorithms are briefly described in Section 2. Section 3: A new hybrid conjugate gradient algorithm has been developed. Model of the Population, Section 4. Section 5 contains numerical comparisons and experiments.

2. CG TECHNIQUE

Due to their speed and simplicity, conjugate gradient (CG) methods are among the most often and efficiently utilized approaches for large-scale optimization issues. Due to their simplicity and minimal memory needs, conjugate gradient algorithms play a key role in rapidly training neural networks. They do not require the evaluation of the Hessian matrix or the impractical storage of an approximation of it. There are various conjugate gradient algorithms in the literature that have been extensively used for neural network training in a range of applications [5], [18]. The linear combination of the negative gradient vector at the current iteration with the previous search direction is the key idea for calculating the search direction. The method for determining the search direction is as follows:

$$d_1 = -g_1; d_{k+1} = -g_{k+1} + \beta_k d_k \tag{4}$$

Conjugate gradient methods differ in their way of defining the multiplier β_k . The most famous approaches were proposed by Fletcher Reeves (FR), Polak–Ribere (PR) and Hestenes–Stifel (HS) [11], [25], [12]:

$$\beta^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \;,\; \beta^{PR} = \frac{g_{k+1}^T y_k}{g_k^T g_k} \;,\; \beta^{PR} = \frac{g_{k+1}^T y_k}{g_k^T y_k}$$

The conjugate gradient methods using β^{FR} update were shown to be globally convergent [9], [21], [19]. However the corresponding methods using β^{PR} or β^{HS} update are generally more efficient ever without satisfying the global convergence property. [6] In the convergence analysis and implementations of CG methods, one often requires the inexact lien search such as the Wolfe line search. The standard Wolfe line search requites σ_k satisfying:

$$E(W_k + \alpha_k d_k) \leq E(W_k) + \rho \alpha_k g_k^T d_k \tag{5}$$

$$g(W_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k \tag{6}$$

or strong Wolfe line search:

$$E(W_k + \alpha_k d_k) \leq E(W_k) + \rho \alpha_k g_k^T d_k \tag{7}$$

$$\left| g_{k+1} + d_k \right| \leq -\sigma g_k d_k \tag{8}$$

Where $0 < \rho < \sigma < 1$

Moreover, an important issue of CG algorithms is that when the search direction (4) fails to be descent (by Descent, we mean $g_k^T d_k < 0 \; \forall k$ directions we restart the algorithm using the negative gradient direction to grantee convergence . A more sophisticated and popular restarting is the Powell restart.

$$\left| g_{k+1}^T g_k \right| \geq 0.2 \left\| g_{k+1} \right\|^2 \tag{9}$$

Where, $\| \;\|$ denotes to the Euclidean norm. Other important issue for then CG methods is that the search directions generated from equation (4) are conjugate if the objective function is convex and line search is exact i.e:

$$d_i^T\ Gd_j = 0, \forall\ i \neq j \tag{10}$$

Where, G is the Hessian matrix for the objective function. Dai and Lioa in [10] showed that the equation (10) can be written as follows:

$$d_{k+1}^T y_k = 0 \tag{11}$$

which is called pure conjugacy condition and generalize to the

$$d_{k+1}^T y_k = t\ g_{k+1}^T s_k \quad , \quad t > 0 \quad , \quad s_k = W_{k+1} - W_k \tag{12}$$

for general objective function with inexact line search.

3. A NEW CONJUGATE ALGORITHM HYBRID ALGORITHM

We will derive New in this section. Unconstrained optimization using a hybrid conjugate gradient technique. Using the direction conjugate algorithm, the Hestenes-Stiefel and Dai-Yuan algorithms were combined [16]. The formula for determining direction is known to us

$$d_{k+1} = -g_{k+1} + \beta_{k+1} d_k \tag{13}$$

Hestenes-Stiefel algorithm

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k} \tag{14}$$

Dai-Yuan algorithm

$$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{y_k^T d_k} \tag{15}$$

Suppose that

$$\beta_{k+1}^* = \eta \beta_{k+1}^{HS} \tag{16}$$

$$\beta_{k+1}^{KH1} = \eta \beta_{k+1}^{DY} + (1-\eta)\beta_{k+1}^{HS} \tag{17}$$

$$d_{k+1} = -g_{k+1} + \eta \beta_{k+1}^{HS} d_k \tag{18}$$

$$d_{k+1} = -g_{k+1} + (\eta \beta_{k+1}^{DY} + (1-\eta)\beta_{k+1}^{HS}) d_k \tag{19}$$

Equality of equation (18) with (19) and note of the d_k equal in equations (18) and (19) we get

$$-g_{k+1}+\eta\beta_{k+1}^{HS}d_k=-g_{k+1}+(\eta\beta_{k+1}^{DY}+(1-\eta)\beta_{k+1}^{HS})d_k \tag{20}$$

Subtracting the g_{k+1} of two said from above equation we have

$$\eta\beta_{k+1}^{HS}d_k=(\eta\beta_{k+1}^{DY}+(1-\eta)\beta_{k+1}^{HS})d_k \tag{21}$$

After some algebra, we get

$$\eta=\frac{\beta_{k+1}^{HS}}{2\beta_{k+1}^{HS}-\beta_{k+1}^{DY}} \tag{22}$$

Substituting η in the equation (19)

$$\beta_{k+1}^{KH1}=\frac{\beta_{k+1}^{HS}\beta_{k+1}^{DY}}{2\beta_{k+1}^{HS}-\beta_{k+1}^{DY}}+(1-\frac{\beta_{k+1}^{HS}}{2\beta_{k+1}^{HS}-\beta_{k+1}^{DY}})\beta_{k+1}^{HS} \tag{23}$$

After some algebra of above equation we get a new formula denote by β_{k+1}^{KH1} is defined by

$$\beta_{k+1}^{KH1}=\frac{(g_{k+1}^{T}y_k)^2}{d_k^T y_k(2g_{k+1}^{T}y_k-\|g_{k+1}\|^2)} \tag{24}$$

Substituting above equation in spectral direction conjugate algorithm, which is developed [1].

There for we have

$$d_{k+1}=-(\xi+\beta_{k+1}^{KH1}\frac{d_k^T y_k}{\|g_{k+1}\|^2})g_{k+1}+\beta_{k+1}^{KH1}d_k \tag{25}$$

New Algorithm KH1

Step[1]:- Initialize w_1 and choose σ , ρ such that $0<\rho<\sigma<1$, $\xi\in[0,1]$ $E_G,\varepsilon>0$ and $K_{\max}$, set $k=1$.

Step[2]:- Calculate the error function value E_k and its gradient g_k .

Step[3]:- If $(E_k<E_G)$ or $\|g_k\|<\varepsilon$, set $w^*=w_k$ and $E^*=E_k$, return goal is meet and stop.

Step[4]:- compute the descent direction :

if $k=1$ then, $d_k=-g_k$ go to step 6

Else

$$\beta_{k+1}^{KH1} = \frac{(g_{k+1}^{T} y_k)^2}{d_k^T y_k (2 g_{k+1}^T y_k - \|g_{k+1}\|^2)}$$ and then compute:

$$d_{k+1} = -(\xi + \beta_{k+1}^{KH1} \frac{d_k^T y_k}{\|g_{k+1}\|^2}) g_{k+1} + \beta_{k+1}^{KH1} d_k .$$

Step[5]:- Compute the learning rate α_k by line search procedure, such the standard Wolfe conditions (5) and (6).

Step[6]:- update the weights:

$w_{k+1} = w_k + \alpha_k d_k$ and set $k = k + 1$.

Step[7]:- If $k > k_{\max}$ return Error goal not meet and stop else go to step (2).

4. POPULATION MODEL

The term "population" refers to all living organisms of the same type that reside in the same geographical area. The phrase "population" is used at the conference Sociology aware to describe to the human people who dwell in a country or region group. And the special science of demography is concerned with the statistical elements of human population.

The topic of modeling the population of the key issues that affected the environment Ecology science is the application of mathematical modeling for the study of movement) dynamic organisms in the growth and decay. Changes in population sizes as a result of interactions between individuals in the natural environment with members of the same gender, as well as other types of living animals, are included in the population modeling study. Knowing the population is one of the most important concerns for various countries around the world, and many conduct censuses every ten years to determine the true population numbers, and the importance of knowing the population in its close association with various aspects of life in human societies, and strong development plans and their relationship to and provision of [16].

It was true for population modeling beginning in the eighteenth century with the development of tools for modeling the change in the number of individuals to comprehend population expansion and contraction. When contemplating the fate of humanity, British scholar Thomas Malthus Thomas Robert Malthus (1834-1766) and one of the pioneers in this field, to remark that the population of human beings grows according to a geometric pattern [16]. Thomas Malthus' formula for population growth

$N(t) = N(0)e^{ct}$

t	The time
$N(t)$	Number of people
$N(0)$	Number of people in time $t = 0$

$$N(t) = N(0)e^{0.3t}; N(0) = 3.9$$

5. EXPERIMENTS AND RESULTS

The performance of the algorithm KH1 has been studied using a computer simulation. The simulations were run in MATLAB (7.6) win8 and hp laptop, and the MSBP's performance was evaluated and compared to batch versions of the above approach. and In this part of the research we write the results of the learned artificial neural network and a comparison with the results of the Thomas Malthus Modeling, and as shown in table (1) and drawing (1) as drawing (1) illustrates the comparison between the Modeling results and the artificial neural network results, and the drawing (2) shows the square error rate resulting from training the artificial neural network.

Table 1. ANN Solution vs. Thomas Malthus Modeling

Thomas Malthus Modeling Population (million people)	ANN Population (million people)
4.1194	4.1157
4.1437	4.1506
4.1681	4.1739
4.1927	4.1903
4.2673	4.265
4.3689	4.3682
4.473	4.4666
4.6065	4.6152
4.6337	4.6402
4.661	4.6709
4.7161	4.7222
4.744	4.7374
4.8569	4.8474
5.0019	5.0044
5.0314	5.0257
5.0611	5.0559
5.0909	5.0917
5.121	5.1236
5.1512	5.146
5.5279	5.5211
5.7265	5.7181
6.4037	6.407
6.7122	6.7165
6.8721	6.8684
7.0772	7.0869

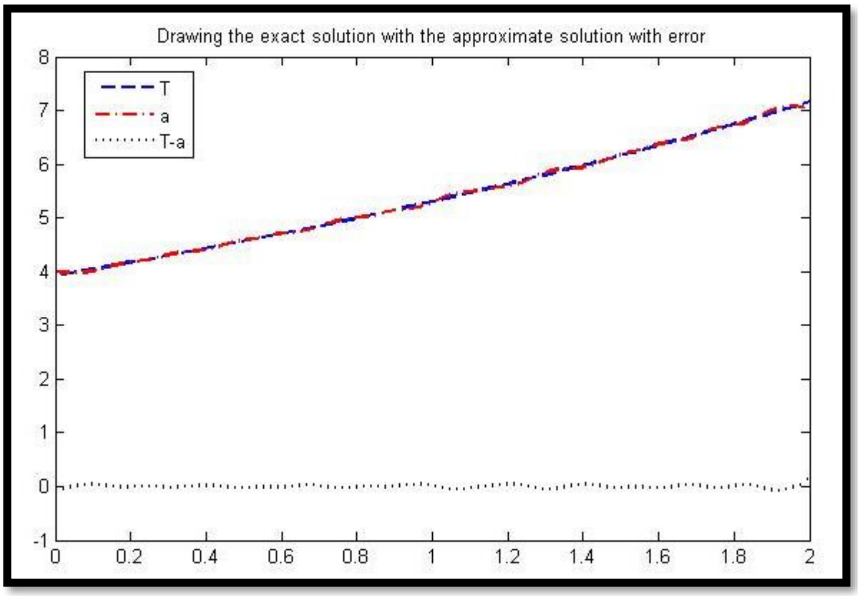

Figure.1. Comparison between the Modeling results and the artificial neural network results with error

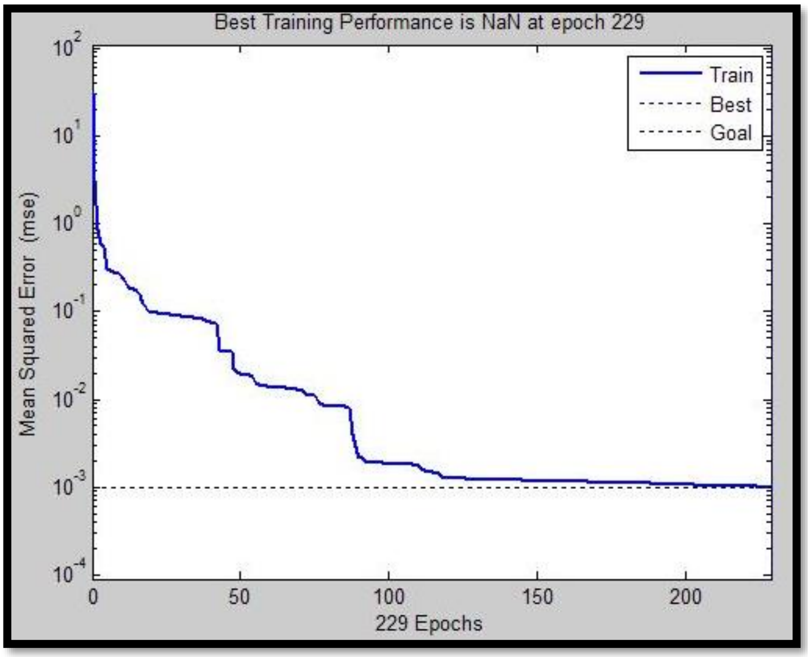

Figure.2. The Mean Squared Error (MSE)

6. CONCLUSIONS

In this work, we have proposed a new CG method for training neural networks which are based on the hybrid algorithm Hestenes-Stiefel and Dai-Yuan. The proposed method preserves the strong convergence properties and descent property. The proposed method is suitable for training large-scale neural networks. Our numerical experiments have shown that the proposed method efficient to predict the size of the population, We can use the method to predict other problems through the use of neural networks, as well as solving optimization, fuzzy optimization problems, solving fuzzy equations, and fuzzy neural networks.

REFFRNCES

[1] Abbo, K. and Hind, M. 2012. Improving the learning rate of the Backpropagation Algorithm Aitken process'. *Iraqi Journal of the statistical sciences, accepted (to appear)*. (2012).

[2] Abbo, K. and Mohammed, H. 2014. Conjugate Gradient Algorithm Based on Aitken's Process for Training Neural Networks. *AL-Rafidain Journal of Computer Sciences and Mathematics*. 11, 1 (2014). DOI:https://doi.org/10.33899/csmj.2014.163730.

[3] Abbo, K.K. and Khudhur, H.M. 2016. New A hybrid conjugate gradient Fletcher-Reeves and Polak-Ribiere algorithm for unconstrained optimization. *Tikrit Journal of Pure Science*. 21, 1 (2016), 124–129.

[4] Abbo, K.K. and Khudhur, H.M. 2016. New A hybrid Hestenes-Stiefel and Dai-Yuan conjugate gradient algorithms for unconstrained optimization. *Tikrit Journal of Pure Science*. 21, 1 (2016), 118–123.

[5] Abbo, K.K., Laylani, Y.A. and Khudhur, H.M. 2016. Proposed new Scaled conjugate gradient algorithm for Unconstrained Optimization. *International Journal of Enhanced Research in Science, Technology & Engineering*. 5, 7 (2016).

[6] ABBO, K.K., Laylani, Y.A. and Khudhur, H.M. 2018. A NEW SPECTRAL CONJUGATE GRADIENT ALGORITHM FOR UNCONSTRAINED OPTIMIZATION. *International Journal of Mathematics and Computer Applications Research (IJMCAR)*. 8, (2018), 1–9.

[7] Abdullah, Z.M., Hameed, M., Hisham, M.K. and Khaleel, M.A. 2019. Modified new conjugate gradient method for Unconstrained Optimization. *Tikrit Journal of Pure Science*. 24, 5 (2019), 86–90.

[8] Ahmed, A.S. 2018. *Optimization Methods For Learning Artificial Neural Networks*. University of Mosul.

[9] Al-Baali, M. 1985. Descent property and global convergence of the Fletcher—Reeves method with inexact line search. *IMA Journal of Numerical Analysis*. 5, 1 (1985), 121–124.

[10] Dai, Y.H. and Liao, L.Z. 2001. New conjugacy conditions and related nonlinear conjugate gradient methods. *Applied Mathematics and Optimization*. 43, 1 (2001), 87–101. DOI:https://doi.org/10.1007/s002450010019.

[11] Fletcher, R. and Reeves, C.M. 1964. Function minimization by conjugate gradients. *The Computer Journal*. 7, 2 (1964), 149–154. DOI:https://doi.org/10.1093/comjnl/7.2.149.

[12] Hestenes, M.R. and Stiefel, E. 1952. *Methods of conjugate gradients for solving linear systems*. NBS Washington, DC.

[13] Hmich, A., Badri, A. and Sahel, A. 2011. Automatic speaker identification by using the neural network. *International Conference on Multimedia Computing and Systems -Proceedings* (2011).

[14] Jabbar, H.N., Abbo, K.K. and Khudhur, H.M. 2018. Four--Term Conjugate Gradient (CG) Method Based on Pure Conjugacy Condition for Unconstrained Optimization. *kirkuk university journal for scientific studies*. 13, 2 (2018), 101–113.

[15] Khudhur, H.M. 2020. Modified Barzilai-Borwein Method for Steepest Descent Method to Solving Fuzzy Optimization Problems(FOP). *Albahir journal*. 12, 23–24 (2020), 63–72.

[16] Khudhur, H.M. 2015. *Numerical and analytical study of some descent algorithms to solve unconstrained Optimization problems*. University of Mosul.

[17] Khudhur, H.M. and Abbo, K.K. 2021. A New Conjugate Gradient Method for Learning Fuzzy Neural Networks. *Journal of Multidisciplinary Modeling and Optimization*. 3, 2 (2021), 57–69.

[18] Khudhur, H.M. and Abbo, K.K. 2021. A New Type of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Algebraic Equations. *Journal of Physics: Conference Series*. 1879, 2 (2021), 22111. DOI:https://doi.org/10.1088/1742-6596/1879/2/022111.

[19] Khudhur, H.M. and Abbo, K.K. 2021. New hybrid of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Equations. *Journal of Soft Computing and Artificial Intelligence*. 2, 1 (2021), 1–8.

[20] Lange, N., Bishop, C.M. and Ripley, B.D. 1997. Neural Networks for Pattern Recognition. *Journal of the American Statistical Association*. 92, 440 (1997). DOI:https://doi.org/10.2307/2965437.

[21] Laylani, Y.A., Abbo, K.K. and Khudhur, H.M. 2018. Training feed forward neural network with modified Fletcher-Reeves method. *Journal of Multidisciplinary Modeling and Optimization*. 1, 1 (2018), 14–22.

[22] Livieris, I.E. and Pintelas, P. 2012. An Advanced Conjugate Gradient Training Algorithm Based on a Modified Secant Equation. *ISRN Artificial Intelligence*. 2012, (2012). DOI:https://doi.org/10.5402/2012/486361.

[23] Livieris, I.E., Sotiropoulos, D.G. and Pintelas, P. 2009. On descent spectral CG algorithms for training recurrent neural networks. *PCI 2009 - 13th Panhellenic Conference on Informatics* (2009).

[24] Nguyen, D. and Widrow, B. 1990. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. *1990 IJCNN International Joint Conference on Neural Networks* (1990), 21–26.

[25] Polak, E. and Ribiere, G. 1969. Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*. 3, R1 (1969), 35–43.

[26] Rumelhart, D.E., Hinton, G.E. and Williams, R.J. 2013. Learning Internal Representations by Error Propagation. *Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence*.

[27] Svozil, D., Kvasnička, V. and Pospíchal, J. 1997. Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems* (1997).

[28] Walczak, S. and Cerpa, N. 1999. Heuristic principles for the design of artificial neural networks. *Information and Software Technology*. 41, 2 (1999). DOI:https://doi.org/10.1016/S0950-5849(98)00116-5.