IEICE TRANS. FUNDAMENTALS, VOL. E79–A, NO. 10 OCTOBER 1996

PAPER *Special Section on Nonlinear Theory and its Applications*

Quaternionic Multilayer Perceptrons for Chaotic Time Series Prediction

Paolo ARENA†, Riccardo CAPONETTO††, Luigi FORTUNA†, Giovanni MUSCATO†, *and* Maria Gabriella XIBILIA†*, *Nonmembers*

SUMMARY In the paper a new type of Multilayer Perceptron, developed in Quaternion Algebra, is adopted to realize short-time prediction of chaotic time series. The new introduced neural structure, based on MLP and developed in the hypercomplex quaternion algebra (HMLP) allows accurate results with a decreased network complexity with respect to the real MLP. The short term prediction of various chaotic circuits and systems has been performed, with particular emphasys to the Chua's circuit, the Saito's circuit with hyperchaotic behaviour and the Lorenz system. The accuracy of the prediction is evaluated through a correlation index between the actual and predicted terms of the time series. A comparison of the performance obtained with both the real MLP and the hypercomplex one is also reported.

key words: *neural networks, quaternion algebra, time series prediction, chaotic systems*

1. Introduction

In the last years neural networks, and in particular Multi-Layer-Perceptrons (MLP) [1], have been used to perform non linear function approximation as well as to predict non-linear time series [2]. In particular, as regards chaotic time series, only short-term prediction can be obtained and a great accuracy is required when the prediction time increases. In fact, due to the chaotic nature of the system generating the time series, the prediction accuracy quickly degrades with the time steps. In this field, neural networks allow better results with respect to other strategies [2], due to their ability to extract the non linear maps generating the data. In time series prediction problems, MLPs are trained to perform a τ step ahead estimation of the current time-series term. The I/O relation of the network is therefore:

$$\bar{x}(t+\tau) = f(\bar{x}(t)) \tag{1}$$

where $\bar{x}(t)$ is the state at time t. Two different strategies can be used for the training. In the first strategy (direct method) τ is fixed and the network is trained to predict the target $\bar{x}(t+\tau)$ when the input sample is the measured value $\bar{x}(t)$. The second strategy (iterative method) consists in training the network to perform one step ahead prediction using the past predicted samples as inputs $(\tau-1)$ times. At the time $(t+\tau)$ the measured value $\bar{x}(t+\tau)$ is considered as input. It has been shown [2] that the latter method allows more accurate predictions and will be thereofore used in this paper. However, when the number of I/O variables grows, a drawback takes place, due to the increasing number of connections in the MLP, i.e. of free parameters in the learning algorithm. This fact degrades the network performance, enhancing the possibily of being trapped in local minima and requiring a great number of I/O samples during learning.

To alleviate this problem a new neural structure, developed in the hypercomplex quaternion algebra (HMLP) [3], [4], is proposed in this paper. As it will be discussed in the following, such a structure allows to obtain the same approximation capabilities as the real MLP with a lower number of free parameters, thus improving the convergence of the learning algorithm.
In the next section the HMLP is defined, describing both the feed-forward phase and the learning algorithm. The approximation capabilities of such a structure are also discussed. In Sect. 3 HMLPs are applied to realize the short term prediction of the Chua's circuit, the Saito's circuit and the Lorenz system. The results obtained are compared with those ones derived with the classical MLP and the obtained improvements are outlined. The fundamental definitions on quaternion algebra are reported in the appendix.

2. The Quaternion MLP

Quaternion algebra (H) has been invented by W.R. Hamilton in 1843 in order to extend in the 3-D space the properties of complex numbers. A *quaternion* can in fact be defined as a complex number with three imaginary parts, [3], [6]–[8]. In this section a new MLP structure, defined in Quaternion Algebra, is introduced and its approximation capabilities are stated. A suitable learning algorithm for such a structure is also reported. Comparisons between quaternionic MLPs and real MLPs in a large number of applications [9]–[12] show that the former structures allow problems to be

Manuscript received February 6, 1996.
Manuscript revised April 30, 1996.

†The authors are with the Dipartimento Elettrico, Elettronico e Sistemistico, Faculty of Engineering University of Catania, Viale A. Doria, 6, 95125, Catania, Italy.
††The author is with the Soft Computing and Multimedia Architecture Group at the SGS Thomson Microelectronic, Stradale Primosole 50, 95121, Catania, Italy.

*The work has been partially supported by: Progetto MURST ex 40%: Metodi neurali e non lineari per la progettazione di circuiti elettronici.

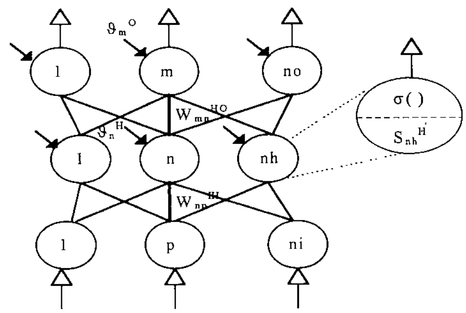


Fig. 1 Basic structure of the HMLP.

solved with the same approximation capabilities as real MLPs, but with a significant reduction (about 40%) in the number of free parameters in the learning algorithm. The simpler HMLP topologies allow therefore a faster learning phase and a lower probability to get stuck in local minima, due to the reduced number of real parameters involved in the error minimization performed by the Back-Propagation algorithm.

Let us define an HMLP (Hypercomplex Multilayer Perceptron) as a Multilayer Perceptron in which input and output values, weights and biases are quaternions, and the activation functions in the hidden layer neurons are quaternion-valued sigmoidal functions. In the following, HMLPs with only one hidden layer will be considered, since it has been proven [12] that this structure is a universal interpolator in H. However, the learning algorithm has been also developed for a multi-hidden layer structure [9]. In Fig. 1 an HMLP with only one hidden layer is shown in order to clarify the notation used in the following.

Let no, nh, ni, be the number of output, hidden and input units respectively, and:
$\mathbf{w}^{HO}_{mn} = w^{HO}_{0mn} + \mathbf{i}w^{HO}_{1mn} + \mathbf{j}w^{HO}_{2mn} + \mathbf{k}w^{HO}_{3mn}$:
the weight connecting the n-th hidden unit to the m-th output one;
$\mathbf{w}^{IH}_{np} = w^{IH}_{0np} + \mathbf{i}w^{IH}_{1np} + \mathbf{j}w^{IH}_{2np} + \mathbf{k}w^{IH}_{3np}$:
the weight connecting the p-th input unit to the n-th hidden one;
$\mathbf{\Theta}^{H}_{n} = \theta^{H}_{0n} + \mathbf{i}\theta^{H}_{1n} + \mathbf{j}\theta^{H}_{2n} + \mathbf{k}\theta^{H}_{3n}$:
the n-th hidden unit bias;
$\mathbf{\Theta}^{O}_{m} = \theta^{O}_{0m} + \mathbf{i}\theta^{O}_{1m} + \mathbf{j}\theta^{O}_{2m} + \mathbf{k}\theta^{O}_{3m}$:
the m-th output unit bias
$m = 1, \dots, no \quad h = 1, \dots, nh \quad p = 1, \dots, ni$. The superscript H will denote the quantities referring to the hidden layer, while the superscript O will denote the quantities referring to the output layer.

The feed-forward phase is obtained with the following relations (quaternionic operators are defined in the Appendix):

hidden unit activation value:

$$\mathbf{S}^{H}_{n} = \sum_{p=1}^{ni} \mathbf{w}^{IH}_{np} \otimes \mathbf{I}_p + \mathbf{\Theta}^{H}_{n} \tag{2}$$

$\mathbf{I}_p$ being the $p-th$ quaternionic input;

hidden unit output value:

$$\mathbf{X}^{H}_{n} = \boldsymbol{\sigma}(\mathbf{S}^{H}_{n}) \tag{3}$$

output unit activation value:

$$\mathbf{S}^{O}_{m} = \sum_{n=1}^{nh} \mathbf{w}^{HO}_{mn} \otimes \mathbf{X}^{H}_{n} + \mathbf{\Theta}^{O}_{m} \tag{4}$$

m-th output unit value:

$$\mathbf{X}^{O}_{m} = \mathbf{Y}_m = \boldsymbol{\sigma}(\mathbf{S}^{O}_{m}) \tag{5}$$

As regards the choice of the activation function, the following function has been selected:

$$\boldsymbol{\sigma}(\mathbf{q}) = \sigma(q_0) + \sigma(q_1)\mathbf{i} + \sigma(q_2)\mathbf{j} + \sigma(q_3)\mathbf{k} \tag{6}$$

where:

$$\sigma(q_i) = \frac{1}{1+\exp(-q_i)} \qquad i = 0\dots 3$$

is the usual sigmoidal real valued activation function. The activation function derivative is therefore $\dot{\boldsymbol{\sigma}}(\cdot) = \dot{\sigma}(\cdot) + \dot{\sigma}(\cdot)\mathbf{i} + \dot{\sigma}(\cdot)\mathbf{j} + \dot{\sigma}(\cdot)\mathbf{k}$ where $\dot{\sigma}(\cdot) = \sigma(\cdot)[1-\sigma(\cdot)]$.

The error function is defined as:

$$E = \frac{1}{2}\sum_{m=1}^{no} (\mathbf{t}_m - \mathbf{Y}_m)^2 \tag{7}$$

$\mathbf{t}_m$ being the quaternionic desired value for the $m-th$ HMLP output unit.

Following the classical steps needed to derive the Back-Propagation algorithm [1], the rules for weights (biases) updating are derived as a function of the opposite gradient value of the error function with respect to the weights (biases) themselves, by suitably using the chain rule. The learning algorithm obtained [9], [12] is reported in the following. For the weights connecting the output to the hidden layer the updating formula is:

$$\begin{aligned}\Delta\mathbf{w}^{HO}_{mn} &= \epsilon(-\nabla E_{\mathbf{w}^{HO}_{mn}}) \\ &= \epsilon[(\mathbf{t}_m - \mathbf{Y}_m) \odot \dot{\boldsymbol{\sigma}}(\mathbf{S}^{O}_{m})] \otimes \mathbf{X}^{H*}_{n} \\ &= \epsilon\boldsymbol{\delta}^{O}_{m} \otimes \mathbf{X}^{H*}_{n}\end{aligned} \tag{8}$$

where $\odot$ denotes the component-by-component product and $\epsilon \in \Re^{+}$ is the learning rate. For the weights connecting the hidden layer to the input one it results:

$$\Delta\mathbf{w}^{IH}_{np} = \epsilon(-\nabla E_{\mathbf{w}^{IH}_{np}}) = \epsilon\boldsymbol{\delta}^{H}_{n} \otimes \mathbf{I}^{*}_{p} \tag{9}$$

where:

$$\boldsymbol{\delta}^{H}_{n} = [\sum_{m=1}^{no} \mathbf{w}^{*HO}_{mn} \otimes \boldsymbol{\delta}^{O}_{m}] \odot \dot{\boldsymbol{\sigma}}(\mathbf{S}^{H}_{n}) \tag{10}$$

As regards the biases, the updating is made as follows:

$$\Delta\boldsymbol{\theta}^{O}_{m} = \epsilon\boldsymbol{\delta}^{O}_{m} \tag{11}$$

$$\Delta\boldsymbol{\theta}_n^H = \epsilon\boldsymbol{\delta}_n^H \tag{12}$$

Let us remark that the problems approached with an HMLP with p input units and m outputs can be also approached with a real MLP with $4p$ inputs and $4m$ outputs. Considering the fact that a quatenion valued sigmoidal function is equivalent to four real valued sigmoids, and that a quaternion is equivalent to four real numbers, the two equivalent real MLP and HMLP can be compared as regards the number of real parameters employed in the structure (space complexity). A comparison is performed with the same number of sigmoids; it means n sigmoids for the NMLP and $4n$ sigmoids for the real MLP. The number k_{MLP} and k_{HMLP} of parameters employed is given respectively as follows:

$$k_{HMLP} = 4(p \cdot n + n \cdot m) \tag{13}$$

$$k_{MLP} = (4p) \cdot (4n) + (4n) \cdot (4m) = 4 \cdot k_{HMLP} \tag{14}$$

As it can be observed, the number of real parameters in the HMLP is much lower than in the real MLP. This fact greatly simplifies the convergence of the learning algorithm. The great number of experiments performed has shown that in order to solve a given problem with the same approximation accuracy, a real MLP and an HMLP rarely employ exactly the same number of real sigmoids, but however the latter structure allows to save a quantity of real parameters between the 40% and the 60%: the larger the nerwork, the greater the parameters saving.

A generalization of the density theorem derived for the real MLP[5] has been proven[12] for continuous functions $f : X \rightarrow H$, where X is a compact subset of H^n. As for the real MLP[5] such a result states that HMLPs with the activation function of the type (6) are universal interpolators of continuous quaternion valued functions[4], [12].

The statement of the theorem is the following:

Theorem 2.1: Let $X \subset H^n$ be a compact subset and let $g : X \rightarrow H$ be a continuous function. Then $\forall\epsilon > 0$ there exist some coefficents $\alpha_1, \ldots \alpha_n \in \Re$, some vectors $\bar{\mathbf{y}}_1, \ldots, \bar{\mathbf{y}}_N \in H^n$ and some quaternions $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N \in H$ such that:

$$\sup_{\bar{\mathbf{x}} \in X} |g(\bar{\mathbf{x}}) - \sum_{i=1}^{N} \alpha_i \boldsymbol{\sigma}(\bar{\mathbf{y}}_i^T \cdot \bar{\mathbf{x}} + \boldsymbol{\theta}_i|) < \epsilon$$

In other words, the real vector space:

$$S = \{\sum_{i=1}^{N} \alpha_i \boldsymbol{\sigma}(\bar{\mathbf{y}}_i^T \cdot \bar{\mathbf{x}} + \boldsymbol{\theta}_i)\}$$

with N a natural number, $\alpha_i \in \Re$, $\bar{\mathbf{y}}_i \in H^n$, $\boldsymbol{\theta}_i \in H\}$ is dense in $C^0(X, H)$, the space of continuous functions $X \rightarrow H$ with the norm

$$\| g \| = \sup_{\bar{\mathbf{x}} \in X} |g(\bar{\mathbf{x}})|$$

From the formulation of the density theorem it emerges that the weights from the hidden layer to the output one may be real parameters, instead of quaternions, in order to achieve the chosen approximation degree. The relation describing the feed-forward phase as well as the learning phase can therefore be accordingly simplified. Due to the identity $H = \Re^4$ the HMLP can be used to approximate continuous real valued functions of the type $f : \Re^{4n} \rightarrow \Re^4$, maintaining the advantages introduced with the HMLP in saving a number of real parameters in the structure.

3. Prediction of Chaotic Time Series

The neural structure proposed has been applied to perform the short-term prediction of the time series generated from some canonical chaotic systems. In particular the Chua's circuit[14], [18] a Saito's circuit[19] and the Lorenz system[17] have been considered. To evaluate the prediction capability of the network, the following "correlation index"$\rho_s(\tau)$, currently employed to test the neural network performance in time series prediction[16], is considered in order to compare the predicted time series and the actual one, for a set of pattern not used during the learning phase:

$$\rho_s(\tau) = \frac{\sum_{t=1}^{M}(O_s(t) - O_{sm})(O'_s(t) - O'_{sm})}{D} \tag{15}$$

where D is given as follows:

$$\sqrt{\sum_{t=1}^{M}(O_s(t) - O_{sm})^2}\sqrt{\sum_{t=1}^{M}(O'_s(t) - O'_{sm})^2}$$

being τ the prediction step, M the number of testing samples, $O_s(t)$ the s-th component of the output at time $t + \tau$, $O_{sm}(t)$ the mean value of $O_s(t)$ on the whole set of testing samples, and $O'_s(t)$ and $O'_{sm}(t)$ the corresponding values of the actual time series. Values of the correlation index close to 1 for a particular prediction step τ, mean that the τ-step ahead zero-mean predicted time series is close to the measured one on the whole set of samples available for the testing phase.

The described correlation index will be evaluated for a set of neural topologies, both real and hypercomplex, and for different values of τ, in order to select the network showing the best performance.

3.1 The Chua's Circuit

Chua's circuit is known as the simplest autonomous circuit showing a large set of chaotic behaviours[14]. Its state equations are reported in the following:

$$\dot{x} = \alpha(y - h(x)) \tag{16a}$$

$$\dot{y} = x - y + z \tag{16b}$$

$$\dot{z} = \beta y - \gamma z \tag{16c}$$

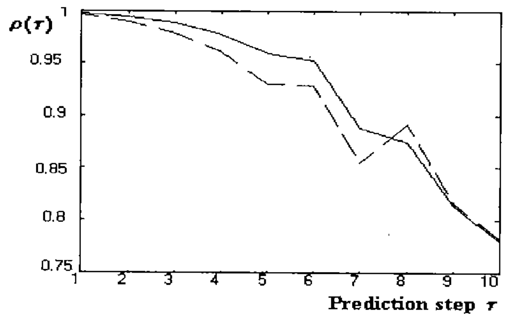


Fig. 2 Comparison between the correlation trends of the first state variables computed with the real MLP ('– –') 3-12-3 and the HMLP ('–') 1-3-1 when τ is increased

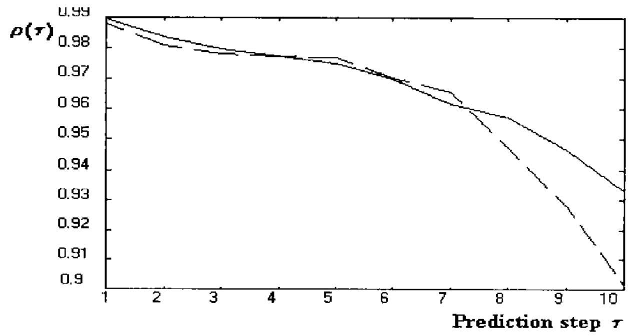


Fig. 3 Comparison between the correlation trends of the first state variables computed with the real MLP ('– –') 4-8-4 and the HMLP ('-') 1-2-1 when τ is increased.

where:

$$h(x) = m_1 x + 0.5(m_0 - m_1)(| x+1 | - | x-1 |)$$

The attractor known as 'double-scroll' [14] is obtained with the parameters

$$(\alpha, \beta, \gamma, m_0, m_1) = (9, 14.286, 0, -1/7, 2/7).$$

The corresponding time series has been obtained simulating the system with $\Delta T = 0.02$ sec and initial condition $(0.1, 0.1, 0.1)$. A set of 250 terms of the simulated time series have been used to train the neural structures. As regards the HMLP, only one input and one output neurons are needed, whose values are the quaternions $\mathbf{q}(t) = 0 + \mathbf{i}x(t) + \mathbf{j}y(t) + \mathbf{k}z(t)$ and $\mathbf{q}(t+1) = 0 + \mathbf{i}x(t+1) + \mathbf{j}y(t+1) + \mathbf{k}z(t+1)$ respectively. It means that in this case Vector Quaternions (i.e. quaternions with the real part equal to zero) are handled; the I/0 space dimension is in fact three. The corresponding real MLP has 3 input and 3 output neurons. Several topologies have been trained for a fixed number of learning cycles to compare their performance. The best results, in terms of correlation values for 250 I/O samples not used during the training, have been obtained with 12 hidden units for the real MLP and with 3 hidden units for the HMLP. In Fig. 2 a comparison between the correlation values of the first state variable as a function of the prediction time steps for both the MLP and the HMLP is reported. As it can be observed, the HMLP leads to better correlation values with respect to the real MLP. Moreover the 1-3-1 HMLP has 40 real parameters (including the biases) while the 3-12-3 MLP requires 87 parameters. The improvement introduced by the HMLP is evident. Therefore, although the approximation capabilities of the real MLP and the HMLP have been demonstrated to be the same, in several cases, like this one, the results obtained with the HMLP are quite better with respect to the other structure. Such a fact could be due to the lower number of real parameters involved into the optimization algorithm which leads to a simpler and faster learning phase.

3.2 The Saito's Circuit

The Saito's circuit, introduced in [19], is characterized by 4 state variables and five parameters which determine transition from torus doubling route to area and volume expanding chaos. The circuit dynamics is governed by the following equations:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ -\alpha_1 & \alpha_1\beta_1 \end{bmatrix} \begin{bmatrix} x_1 - \eta p_1 h(z) \\ y_1 - \eta \frac{p_1}{\beta_1} h(z) \end{bmatrix} \quad (17)$$

$$\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ -\alpha_2 & \alpha_2\beta_2 \end{bmatrix} \begin{bmatrix} x_2 - \eta p_2 h(z) \\ y_2 - \eta \frac{p_2}{\beta_2} h(z) \end{bmatrix} \quad (18)$$

where:

$$h(z) = \begin{cases} 1 & \text{per } z \geq -1 \\ -1 & \text{per } z < 1 \end{cases}$$

and $z = x_1 + x_2$, $p_1 = \dfrac{\beta_1}{1-\beta_1}$, $p_2 = \dfrac{\beta_2}{1-\beta_2}$.

The system is characterized by two positive Lyapunov exponents which make the circuit behaviour hyperchaotic. The corresponding time series has been obtained simulating the system with

$$(\alpha_1, \beta_1, \alpha_2, \beta_2, \eta) = (7.5, 0.16, 15, 0.097, 1.3),$$

starting from the initial condition

$$(x_1, y_1, x_2, y_2) = (1, 0, 1, 0)$$

and with $\Delta T = 0.002$ sec.

As in the previous application, several neural topologies have been trained with 250 I/O samples. In Fig. 3 a comparison of the correlation trends obtained with the HMLP 1-2-1 and the real MLP 4-8-4 is reported for the first state variable. As it can be observed, also in this case, the HMLP shows better performance at large prediction steps. The correlation index for the other state variables shows a similar trend. The number of free parameters are 28 and 76 respectively.

3.3 The Lorenz System

The set of equations known as Lorenz system, [17] are

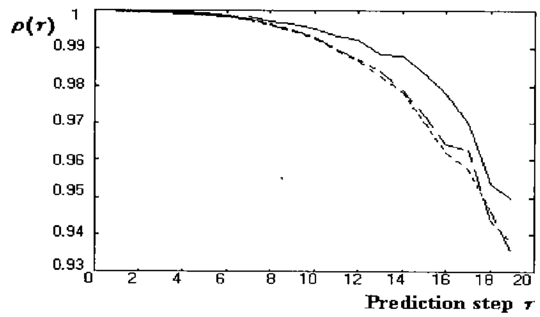


Fig. 4 Comparison between the correlation trends of the first state variables computed with the real MLP ('– –') 3-11-3, the real MLP (...) 3-12-3 and the HMLP ('-') 1-3-1 when τ is increased.

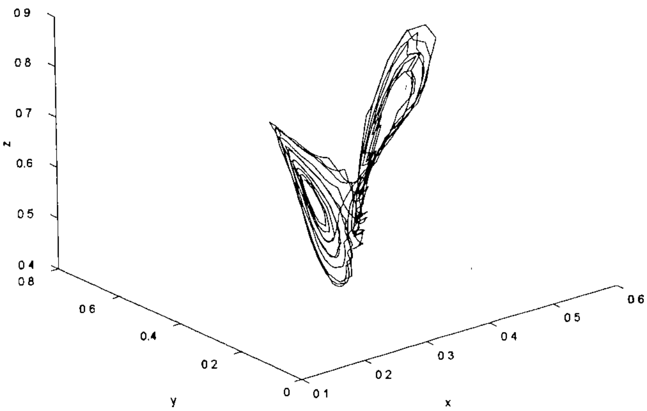


Fig. 5 The 5 step predicted Lorenz attractor realized with the 1-3-1 HMLP.

reported in the following:

$$\dot{x} = \sigma(y - x) \tag{19a}$$

$$\dot{y} = -xz + rx - y \tag{19b}$$

$$\dot{z} = xy - bz \tag{19c}$$

The nominal value of the parameters are $\sigma = 10$, $r = \frac{8}{3}$ and $b = 28$. In the reported study, a set of real and hypercomplex topologies have been trained with 200 terms of the time series. The best results have been obtained with the real 3-11-3 MLP, (comparable with the real MLP 3-12-3) and the 1-3-1 HMLP. A comparison between their performance, in terms of correlation index for the first state variable is shown in Fig. 4 where the improvement due to the use of the HMLP can be appreciated. However, also in this application the number of parameters employed by the HMLP is lower, with respect to the real MLP. As an example, in Fig. 5 the Lorenz attractor realized with the 5-step predicted output is reported. As it is emphasized the obtained performance are satisfactory. The same attractor derived by using the 5 step ahead prediction performed by the 3-11-3 real MLP is clearly a degradated version of the one reported in Fig. 5, due to the lower correlation index obtained with the real MLP.

4. Conclusions

In this paper a new type of neural architecture (HMLP), developed in the Quaternion Algebra, has been introduced to realize the short-term prediction of chaotic time series. The theoretical reported results guarantee that such a structure can approximate any quaternion valued continuous function with an arbitrary degree of accuracy. Taking into account that this results also holds for real multivariable functions, HMLPs can be employed to model MIMO dynamical systems and multi-variables time series.

Several numerical examples are reported in the paper showing the suitability of the strategy introduced, which allows a satisfactory performance, in terms of the correlation index, employing a structure with a lower number of real parameters with respect to the classical real MLP. It is to be also emphasized how the use of HMLPs allows to obtain higher correlation index values at high prediction steps with respect to the real MLP. A study of the capabilities of HMLPs linked to the reduced number of real parameters involved is currently in progress in order to theoretically justify the performance of HMLPs with respect to the real MLPs.

References

[1] D.E. Rumelhart and McClelland, "Parallel Distributed Processing: Exploration in the microstructure of cognition," MIT Press, 1986.
[2] A. Lapedes and R. Farber, "Nonlinear signal processing using neural networks: Prediction and system modelling," LA-UR-87-2662, 1987.
[3] W.R. Hamilton, "Lectures on Quaternions," Hodges and Smith, Dublin, 1853.
[4] P. Arena, L. Fortuna, G. Muscato, and M.G. Xibilia, "Multilayer perceptrons to approximate quaternion valued functions," Neural Networks, in Press.
[5] G. Cybenko, "Approximation by superposition of a sigmoidal function," Mathematics of control signals and systems, vol.2, 1989.
[6] W.R. Hamilton, "Elements of Quaternions," 3rd edition, Chelsea, New York, 1969.
[7] C.A. Deavours, "The quaternionic calculus," Am. Math. Monthly 80, pp.995–1008, 1973.
[8] K. Guerlebeck and W. Sproessig, "Quaternionic analysis and elliptic boundary value problems," Int. Series of Numerical Mathematics, vol.89, Birkhauser.
[9] P. Arena, L. Fortuna, and M.G. Xibilia, "Neural networks in multidimensional domains," Journal of System Eng., U.K., vol.6, pp.1–11, 1996.
[10] P. Arena, L. Fortuna, and M.G. Xibilia, "Neural networks for quaternion valued functions approximation," IEEE Int. Conf. on Circuits and Systems, London, 1994.
[11] P. Arena, S. Baglio, L. Fortuna, and M.G. Xibilia, "Chaotic time-series prediction via quaternionic MLPs," 1995 IEEE Int. Conf. on System Man and Cybernetics, Vancouver, Canada, Oct. 1995.
[12] M.G. Xibilia, "Learning algorithms in hybercomplex algebra and prediction of chaotic behaviours in circuits and dynamical systems (in italian)," Ph.D. Thesis, DEES, University of Catania, Italy, 1995.

[13] P. Arena, L. Fortuna, L. Occhipinti, and M.G. Xibilia, "Neural networks for quaternion valued function approximation," Proc. IEEE Int. Symp. on Circuit and Systems, vol.6, London.
[14] T. Parker and L.O. Chua, "Chaos: A tutorial for engineers," Proc. IEEE, vol.75, no.8, Aug. 1987.
[15] A. Lapedes and R. Farber, "How neural nets work," IEEE Denver Conf. on Neural Nets, preprints, 1987.
[16] H. English and Y. Hiemstra, "The correlation as cost function in neural networks," Proc. IEEE WCCI, Orlando, pp.1370–1372, 1994.
[17] E.N. Lorenz, "Deterministic nonperiodic flow," Journal Atmospheric Science, vol.20, no.130, 1963.
[18] L.O. Chua, "Chua's circuit: Ten years later," 1993 International Symposium on Nonlinear Theory and its Applications, Hawaii, Dec. 1993.
[19] K. Mitsubori and T. Saito, "Torus doubling and hyperchaos in a five dimensional hysteresis circuit," Proc. 1994 IEEE Int. Symp. on Circuit and Systems, vol.6, London.

Appendix: Quaternion Algebra

A Quaternion $\mathbf{q}$ is a generalized complex number composed of four real parameters (q_0, q_1, q_2, q_3) and of three imaginary units $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ which represent the basis unit vectors of an orthogonal reference frame:

$$\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

The imaginary parts $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ in Quaternion Algebra play the same role as the imaginary unit $i = \sqrt{-1}$ in Complex Algebra. Quaternion Algebra, usually called H (the first letter of the the name of its inventor), is an associative division algebra which includes Real and Complex algebra. A quaternion can be considered as the direct sum of a real number q and of a vector $\vec{q}$; it can therefore also be denoted as:

$$\mathbf{q} = q_0 + \vec{q}$$

The fundamental operators are reported in the following:

- conjugate:
  $q^* = q_o - \vec{q} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}$
- modulus:
  $|\mathbf{q}| = \sqrt{\mathbf{q}\mathbf{q}^*}$
- sum: For any two quaternions:
  $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$
  and:
  $\mathbf{p} = p_0 + p_1\mathbf{i} + p_2\mathbf{j} + p_3\mathbf{k}$
  $\mathbf{q}+\mathbf{p} = (q_0+p_0)+(q_1+p_1)\mathbf{i}+(q_2+p_2)\mathbf{j}+(q_3+p_3)\mathbf{k}$
- product:
  $(q_0 + \vec{q}) \otimes (p_0 + \vec{p}) = q_0p_0 - \vec{p}\cdot\vec{p} + q_0\vec{p} + p_0\vec{q} + \vec{q}\times\vec{p}$
  where '$\cdot$' and '$\times$' represent the scalar and vector product respectively, as commonly defined in Vector Algebra.

The product of two quaternions is not commutative, but the associative and distributive properties still hold. Some useful definitions are reported in the following:

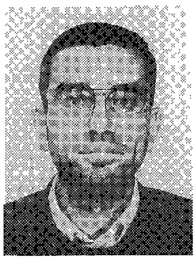
Definition 1: A function $f : H \rightarrow H = f_0(q) + \mathbf{i}f_1(q) + \mathbf{j}f_2(q) + \mathbf{k}f_3(q)$ is continuous if each of its components is a continuous real valued function; it is differentiable if each of its components is differentiable.

Definition 2: Given the operator:

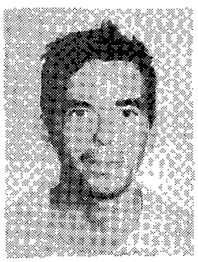
$$\nabla = \frac{d}{dq_0} + \frac{d}{dq_1}\mathbf{i} + \frac{d}{dq_2}\mathbf{j} + \frac{d}{dq_3}\mathbf{k}$$

the gradient of the function f is defined as $\nabla \otimes f$ or as $f \otimes \nabla$.

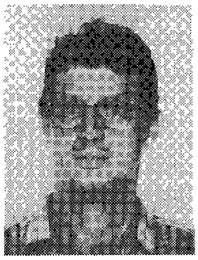
Paolo Arena was born in Catania (Italy) in 1966. He received the electronic engineering degree from the University of Catania in 1990, and the Ph.D. in electrical engineering in 1994. He is currently assistant professor of automatic control. His research interests include dynamic neural networks, complex and multidimensional neural networks and dynamical system modelling.



Riccardo Caponetto was born in Catania in 1966. He received the electronic engineering degree from the University of Catania in 1991, and the Ph.D. in electrical engineering in 1995. Since 1994 he has been working as a senior researcher at the S.G.S. Thomson Microelectronics. His interests include Soft-Computing and Robotics



Luigi Fortuna was born in Siracusa (Italy) in 1953. He received the electrical degree from the University of Catania in 1977. He is presently Full Professor of System Theory at the University of Catania. His scientific interests currently include model order reduction, identification, control theory, neural networks and nonlinear systems.



Giovanni Muscato was born in Catania (Italy) in 1965. He received the electrical degree from the University of Catania in 1988. He is presently Assistant Professor of Automatic Control at the University of Catania. His scientific interests currently include model order reduction, identification, robust control, neural networks and nonlinear systems.

Maria Gabriella Xibilia was born in Catania (Italy) in 1966. She received the electronic engineering degree from the University of Catania in 1991, and the Ph.D. in electrical engineering in 1995. She currently holds a post-doctoral position in Automatic Control at the University of Catania. Her research interests include neural networks and robotics.