Advantages of Hierarchical Organisation in Neural Networks

Yagmur Denizhan [1] **and Gursel Karacor** [2]
[1] *Department of Electrical and Electronics Engineering, Bogazici University, 80815, Bebek, Istanbul, Turkey, Phone: +902123581540, Fax: +902122872465,*
e-mail: denizhan@boun.edu.tr
[2] *Department of Computer Engineering, Turkish Airforce Academy, Yesilyurt, Istanbul, Turkey, Phone: +902126632490*
e-mail: g.karacor@hho.edu.tr

Abstract- *Artificial neural networks, which are inspired by the structure and functioning of the vertebrate-brain, are powerful modelling tools. However, the black-box representation they provide does not allow the usage of the huge accumulation of theoretical knowledge on system dynamics. Similarly, they also do not seem to provide any clue for the symbolic operations typical for the higher functioning mode of the human brain.*
In this study a "chaos control" problem is used as a test case to demonstrate the viability of extracting an analytical model from an artificial neural network. The results are used to comment on the advantages of hierarchical organisation not only in artificial but also natural neural networks.

Keywords: models, neural networks, hierarchical organisation

1. Introduction

All living beings are endowed with a model (or set of models) describing themselves and their environment. Considering the living being as a complex structure with many organisational levels in the sense of Metasystem Transition Theory [Turchin, 1999], one can even claim that any sub-system at any organisational level has a model about itself and its environment. It should be noted that here the concept of "model" is used in a more general sense meaning "representation".

At low organisational levels this representation is inseparable from physical, chemical and geometrical properties of the structure. For example, such properties of a receptor at a cell membrane can well be considered as a representation or a model of the environment because they specify the acceptable ligands, i.e. those components of the environment which are relevant and have a meaning for the cell.

Any "model" used by any organisational level of an organism is a result of a learning and adaptation procedure, which must have taken place in the course of evolution, during the developmental stages of the organism or during its lifetime. Even the genetic code can be considered as such a model of the organism and its environment. The models at lower and hence more ancient organisational levels are well established and stable, whereas those at highest levels keep being constructed and revised during the lifetime of the living being.

The model at any level is used for anticipating or predicting the potential results of actions and choosing the proper action or strategy in a given situation. At lower organisational levels of an organism this procedure occurs quite automatically and it is hard to specify the subject of choice and deliberate action due to lack of agency. However, when we consider the higher organisational levels of an organism, particularly when it belongs to a higher species, the deliberate, goal-oriented, model-based behaviour accompanied by on-going model formation becomes apparent. In higher vertebrates the higher levels of the model are supposed to be hidden in the synaptic dynamics of the brain, which in itself exhibits a hierarchical organisation.

In this paper we will try to arrive at some clues about the hierarchical organisation of these highest levels, namely of the vertebrate brain, by considering a simplified technological imitation: Artificial Neural Networks.

Within the last decades artificial neural networks (ANN's) have been developed, which are meant to imitate the learning process of the brain by modifying the synaptic weights. They can be trained to represent -at least theoretically- any data set and hence to become a model of the system that has produced the data. Although widely recognised as a powerful modelling tool, artificial neural networks have the drawback of being "opaque", i.e. they provide a non-analytical, non-transparent, black-box-like description of the systems they represent. Consequently, no analytical investigations are possible on these inaccessible internal models.

On the other hand, neural networks as a model of the functioning of the (let us say, human) brain do not seem to provide a support for what we observe when we look at it from the other end, i.e. introspectively. The neural network model, which possibly quite well describes the brain at the neuronal level, does not account for the hierarchical, abstract and analytic structures, which we celebrate as the crown of human thinking: analytic thinking.

The aim of this paper is to demonstrate the advantages of using the synaptic organisation of an artificial neural network as an intermediate tool for higher level processing rather than as a direct model. The results of this approach, which allows a combination of the immense modelling capability of ANN's with the advantages of analytical models, will then be interpreted in order to draw some philosophical conclusions about the trend towards a hierarchical organisation of the human brain.

For this purpose, the control problem of a chaotic system has been chosen as a test case. The dynamic equations of the chaotic system are assumed to be unknown, hence have to be estimated from the system data. In classical OGY control the least mean square error estimate of the local system dynamics is obtained from system data and the thus obtained local linear system model is used when applying a classical linear control technique. An alternative approach presented in [Iplikci, 1999] employs an ANN, which after being trained by the system data provides a suitable control action as its output. The study presented in this paper, however, extracts an analytical model from an ANN trained by system data and subsequently uses this analytical model when applying classical control methods. A comparison of the simulation results obtained for chaos control via ANN and chaos control using analytical equations extracted from ANN has demonstrated the superiority of the latter in terms of control precision.

Section 2 of this paper provides a brief review of the classical OGY control of chaotic systems and of a modified version, where both modelling and control tasks are achieved by an ANN. Section 3 presents the alternative approach based on using ANN's as an intermediate tool. Simulation results obtained via classical OGY control, ANN-based control and OGY control on basis of analytical system model extracted from ANN are shown and compared in section 4. Finally, in Section 5 the simulation results are discussed and some conclusions drawn from artificial neural networks are generalised to comment on the nature of hierarchical organisation and the evolutionary tendency of complexity increase in natural neural networks.

2. OGY Control and Its Artificial Neural Network Version

2.1. Classical OGY Control

Since 1990 when Ott, Grebogi and Yorke the so-called OGY control [Ott et al., 1990] has been widely accepted as an effective method of controlling chaotic systems. Due its simplicity and the realistic assumptions, under which it can operate, this method has attracted the attention of many researchers.

The OGY control is applicable to chaotic systems the dynamic equations of which are not known to the controller. The control input to the system is assumed to be through a s-dimensional control parameter vector. These control parameters are allowed to vary by a small amount about their nominal values, which are represented by a nominal control parameter vector $\mathbf{p}_{nom}$.

This approach is based on the presence of many unstable equilibrium points and periodic orbits embedded in the strange attractors chaotic systems. Furthermore, the instability of all these equilibrium behaviours is of saddle-type; i.e. system trajectories tend to approach an equilibrium behaviour at least in one direction while they try to escape from it in other directions. It is assumed that out of the large repertoire of equilibrium behaviours one can be chosen as the "desirable behaviour", which from here onward will be referred to as the "target". The ergodicity of chaotic systems on their strange attractors guarantees that the system will sooner or later pass close enough to any chosen target. The main idea of OGY control is to stabilise the otherwise unstable target via small control parameter variations while the system is in a close neighbourhood. The stabilisation is achieved by linear control techniques employing a local linear model obtained from system data by least mean square error estimation.

First sufficient amount of system data is gathered and analysed in order to choose the "target". In order to provide a unified notation for discrete-time and continuous-time systems, it is more adequate to work with a discrete-time representation of the dynamics. In case of continuous-time systems the so-called Poincaré map (describing the relationship between successive points, at which the system trajectory pierces the Poincaré hyper-surface in a specified direction) will be employed. The chosen target will be represented as an N-dimensional vector $\mathbf{z}^*$, either denoting an equilibrium point of an N-dimensional discrete-time system or an equilibrium point of the Poincaré map of an (N+1)-dimensional continuous-time system.

Next another set of system data is gathered from a close neighbourhood of the target while steadily varying the control parameters in their allowable range in a random manner. These data can be represented as $\{\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{p}_i\}$ for i = 1, 2, ..., L. Here, $\mathbf{z}_i$ denotes the n-dimensional state vector at the i^{th} iteration in case an n-dimensional discrete-time system, or the n-dimensional the vector representing the point where the phase trajectory pierces the Poincaré surface for the i^{th} time in case of an (n+1)-dimensional continuous-time system; $\mathbf{p}_i$ denotes the vector of control parameters applied at the i^{th} iteration and $\mathbf{z}_{i+1}$ the phase point (or Poincaré surface piercing) reached at the next iteration under the application of $\mathbf{p}_i$. A local linear model around $\mathbf{z}^*$ can be obtained from this data set by least mean square error estimation:

$$\mathbf{z}_{i+1} - \mathbf{z}^* = \underline{\underline{\mathbf{A}}}\,(\mathbf{z}_i - \mathbf{z}^*) + \underline{\underline{\mathbf{B}}}\,(\mathbf{p}_i - \mathbf{p}_{nom}) \tag{1}$$

The controller waits until the system enters the OGY-region, a close neighbourhood of $\mathbf{z}^*$ where the local linear model is valid. Within that region Linear Control Theory is employed to calculate $\mathbf{p}_i$ which will stabilise $\mathbf{z}^*$ as shown in eq. (2).

$$\mathbf{p}_i = \mathbf{p}_{nom} - \left(\underline{\underline{\mathbf{B}}}^T \underline{\underline{\mathbf{B}}}\right)^{-1} \underline{\underline{\mathbf{B}\mathbf{A}}}\left(\mathbf{z}_i - \mathbf{z}^*\right) \tag{2}$$

Since the control parameters are allowed to vary only within a limited range, $\mathbf{p}_i$ is left at its nominal value if eq. (2) gives a result outside this allowable range.

2.2. ANN-Based Chaos Control

In [Iplikci, 1999] Iplikci has demonstrated the possibility of employing ANNs for OGY-type chaos control, i.e. local stabilisation of the chosen target using a neural network. Here, instead of obtaining a local linear model as given in eq. (1), an ANN has been trained by the data set $\{\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{p}_i\}$ for i = 1, 2, ..., L using port 1 for the present state $\mathbf{z}_i$, port 2 for the next state $\mathbf{z}_{i+1}$, and the output port for the control parameter vector $\mathbf{p}_i$ as shown in figure 1.a. In other words, an internal model is created in the neural network, representing both the local system dynamics and its dependence on control parameter variations.

After being trained, the ANN provides as an output the necessary control parameter vector $\mathbf{p}_i$ when any present state $\mathbf{z}_i$ within the OGY-region is fed to port 1 and $\mathbf{z}^*$ (i.e. the desired next state) is fed to port 2 (figure 1.b).

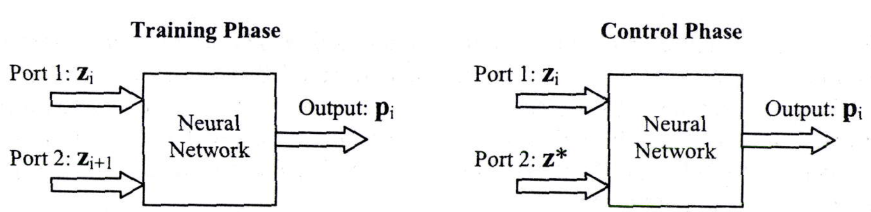

Figure 1. Training (a) and control (b) phases of the ANN used for chaos control.

For this purpose Radial Basis Function (RBF) based neural networks have been preferred due to their high approximation capability. The architecture of an RBF-based neural network with input **x** and output y is shown in figure 2. For the sake of simplicity a single scalar output is considered. However, the scheme can easily be extended to a configuration with many output nodes.

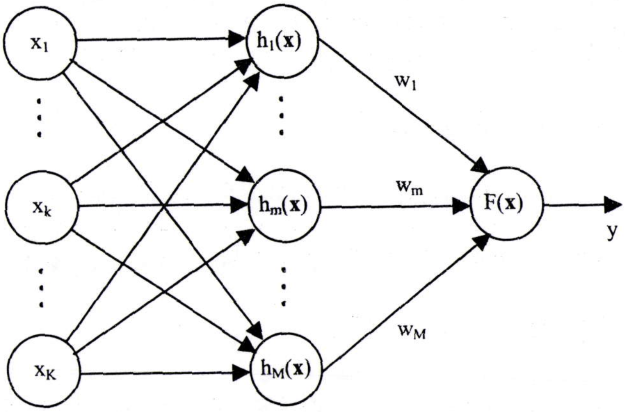

Figure 2. The architecture of an RBF-based neural network with input $\mathbf{x} = \{x_1, \ldots, x_K\}$ and output y.

The network uses a linear model,

$$F(\mathbf{x}) = \sum_{m=1}^{M} w_m h_m(\mathbf{x}) \tag{3}$$

where w_m's are the weights and the h_m's are the Radial Basis Functions. Gaussian, Cauchy and multi-quadric functions can be used as a Radial Basis Function. In this work multi-quadric basis functions (equation 4) have been preferred because they provide a slightly better approximation.

$$h(\mathbf{x}) = \frac{\sqrt{r^2 + \|\mathbf{x} - \mathbf{c}\|^2}}{r} \tag{4}$$

Here, **c** is the centre and r is the radius of the function. The training algorithm tries to cover the data space with a sufficient number of radial basis functions at appropriate loci (**c**'s) and with the appropriate radii such that the predicted sum-squared error is minimised. In that sense, Radial Basis Functions can be regarded as a smooth transition between Fuzzy Logic and Artificial Neural Networks. Further details can be obtained from Orr's web page [Orr, 1996].

3. OGY Control Using an Analytical Model Extracted from ANN

Although powerful modelling tools, artificial neural networks can be criticised for not lending themselves to analytical calculations. However, Karacor in [Karacor, 2002] has demonstrated a possibility of using a neural network as intermediate modelling tool to reach an analytical description of the latent model in it.

The study presented here is an OGY-control based on an analytical model, which is extracted from a trained ANN, rather than obtained directly from system data on basis of least mean square error criterion. The model can be extracted from a trained ANN by means of Taylor's expansion about the equilibrium point and the nominal values of the control parameter vector in the spirit of Lyapunov's linearisation method for non-linear systems.

However, if desired, the model extracted from the ANN can be taken to any higher order by considering the higher order terms in the Taylor's expansion, thus allowing a model valid for a larger neighbourhood of the equilibrium point.
Let us assume that an RBF-based ANN has been trained by the system data $\{\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{p}_i\}$ for i = 1, 2, ..., L gathered from a close neighbourhood of $\mathbf{z}^*$ as shown in figure 3.

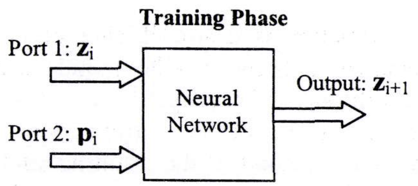

Figure 3. Training of a neural network from which an analytical system model will

An RBF-based ANN trained as given in figure 3 by data gathered from the close neighbourhood of the target $\mathbf{z}^*$ has the following representation for $(z_{i+1})_j$ the j[th] component of $\mathbf{z}_{i+1}$:

$$(z_{i+1})_j = \sum_{m=1}^{M} w_{jm} h_{jm}(\mathbf{z}_i, \mathbf{p}_i) \tag{5}$$

where $h_{jm}(\mathbf{z}_i)$ can be expressed as follows:

$$h_{jm}(\mathbf{z}_i, \mathbf{p}_i) = \frac{\sqrt{r_{jm}^2 + \|\mathbf{z}_i - \mathbf{c}_{jm}\|^2 + \|\mathbf{p}_i - \mathbf{p}_{nom}\|^2}}{r_{jm}} \tag{6}$$

Hence Taylor's expansion about $\mathbf{z}^*$ and $\mathbf{p}_{nom}$ can be expressed as given in eq. (5).

$$\begin{aligned}(z_{i+1})_j = \sum_{m=1}^{M} w_{jm} h_{jm}(\mathbf{z}^*, \mathbf{p}_{nom}) + \left[\sum_{m=1}^{M} w_{jm} \frac{\partial h_{jm}(\mathbf{z}_i, \mathbf{p}_{nom})}{\partial \mathbf{z}_i}\Big|_{\mathbf{z}_i=\mathbf{z}^*}\right](\mathbf{z}_i - \mathbf{z}^*) \\ + \left[\sum_{m=1}^{M} w_{jm} \frac{\partial h_{jm}(\mathbf{z}^*, \mathbf{p}_i)}{\partial \mathbf{z}_i}\Big|_{\mathbf{p}_i=\mathbf{p}_{nom}}\right](\mathbf{p}_i - \mathbf{p}_{nom}) + H.O.T.\end{aligned} \tag{7}$$

The first term on the right hand side of eq. (7) can be identified as the j[th] component of $\mathbf{z}^*$. Taking it to the left and generalising the equation for all j=1,..., n eq. (8) can be obtained, which results in the eq. (1) when higher order terms in $(\mathbf{z}_i - \mathbf{z}^*)$ and $(\mathbf{p}_i - \mathbf{p}_{nom})$ are neglected.

$$\mathbf{z}_{i+1} - \mathbf{z}^* = \underline{\underline{\mathbf{A}}}\,(\mathbf{z}_i - \mathbf{z}^*) + \underline{\underline{\mathbf{B}}}\,(\mathbf{p}_i - \mathbf{p}_{nom}) + \text{H.O.T.} \tag{8}$$

Comparing eqs. (7) and (8), the matrices in eq. (7) can be identified as follows:

$$\underline{\underline{\mathbf{A}}} = \begin{bmatrix} \sum_{m=1}^{M} w_{1m} \dfrac{\partial h_{1m}(\mathbf{z}_i, \mathbf{p}_{nom})}{\partial \mathbf{z}_i} \Big|_{\mathbf{z}_i = \mathbf{z}^*} \\ \bullet \\ \bullet \\ \sum_{m=1}^{M} w_{nm} \dfrac{\partial h_{nm}(\mathbf{z}_i, \mathbf{p}_{nom})}{\partial \mathbf{z}_i} \Big|_{\mathbf{z}_i = \mathbf{z}^*} \end{bmatrix} \quad \textit{and} \quad \underline{\underline{\mathbf{B}}} = \begin{bmatrix} \sum_{m=1}^{M} w_{1m} \dfrac{\partial h_{1m}(\mathbf{z}^*, \mathbf{p}_i)}{\partial \mathbf{z}_i} \Big|_{\mathbf{p}_i = \mathbf{p}_{nom}} \\ \bullet \\ \bullet \\ \sum_{m=1}^{M} w_{sm} \dfrac{\partial h_{sm}(\mathbf{z}^*, \mathbf{p}_i)}{\partial \mathbf{z}_i} \Big|_{\mathbf{p}_i = \mathbf{p}_{nom}} \end{bmatrix} \tag{9}$$

If a more precise approximation is desired higher order terms in eq. (7) can be included where the associated matrices can be found taking higher derivatives, a calculation not presented for the sake of brevity.

After obtaining the local linear model the control action can be calculated from eq. (2) as used in the original OGY method. If the resulting control parameter variation is outside the allowable range, $\mathbf{p}_i$ is simply left at its nominal value.

4. Simulation Results

The different control approaches presented in sections 2 and 3 have been applied to two discrete-time chaotic systems and one continuous-time system shown in Table 1.

Table 1. Chaotic systems used for comparing different local chaos control strategies.

Name	System equations	Coordinates of the chosen target	Nominal values and allowable variations of the control parameters
Logistic map	$x_{n+1} = px_n(1 - x_n)$	$x^* = 0.7436$	$p_{nom} = 3.9 \pm 0.1$
Henon map	$x_{n+1} = p + 0.3y_n - x_n^2$ $y_{n+1} = x_n$	$x^* = 0.8717$ $y^* = 0.8717$	$p_{nom} = 1.37 \pm 0.03$
Lorenz system	$\dot{x} = \sigma(y - x)$ $\dot{y} = (\rho x - y - xz)$ $\dot{z} = (xz - \beta z)$	Periodic orbit piercing the Poincaré surface through $y = 8.4313$ at $x^* = 14.23$ and $z^* = 39.80$	$\sigma_{nom} = 10 \pm 0.3$ $\rho_{nom} = 28 \pm 0.84$ $\beta_{nom} = 2.67 \pm 0.08$

Table 2 shows the model parameter matrices for the three chaotic systems considered in this study. In case of discrete-time maps, also the true values of the model parameters, which are analytically calculated from the system equations, are given for the sake of comparison.

Table 2. Model parameters estimated by different methods for the three chaotic

Form of the analytical model: $z_{i+1} - z^* = \underline{\underline{A}}(z_i - z^*) + \underline{\underline{B}}(p_i - p_{nom})$			
System	**Method of estimation**	$\underline{\underline{A}}$	$\underline{\underline{B}}$
Logistic Map	Analytical (true value)	−1.90008	0.190659
	Least mean square est.	-1.8862	0.1775
	Extracted from ANN	-1.8915	0.1854
Henon Map	Analytical (true value)	$\begin{bmatrix} -1.7434 & 0.3 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
	Least mean square est.	$\begin{bmatrix} -1.7622 & 0.2308 \\ 1.0287 & 0.0551 \end{bmatrix}$	$\begin{bmatrix} 0.9298 \\ -0.5683 \end{bmatrix}$
	Extracted from ANN	$\begin{bmatrix} -1.7345 & 0.3042 \\ 1.0007 & -0.0003302 \end{bmatrix}$	$\begin{bmatrix} 1.0023 \\ 0.00033553 \end{bmatrix}$
Lorenz System	Least mean square est.	$\begin{bmatrix} 0.0854 & 0.185 \\ 0.55 & 2.07 \end{bmatrix}$	$\begin{bmatrix} 0.0008 & -0.003 & 0.1497 \\ 0.0015 & 0.0091 & 0.1515 \end{bmatrix}$
	Extracted from ANN	$\begin{bmatrix} 0.0541 & 0.181 \\ 0.621 & 2.11 \end{bmatrix}$	$\begin{bmatrix} 0.0005 & -0.001 & 0.135 \\ 0.0013 & 0.0081 & 0.136 \end{bmatrix}$

In Table 3 the performances of various model and controller combinations are compared indicating the following criteria:

Radius of training region: The radius of the phase space about the target where the training data are gathered.

Maximum control region radius: The maximum radius of the neighbourhood about the target where the OGY method can stabilise the target.

Training sample size: The number of the training patterns representing the input-output relation of the system. This number defines the amount of data used to train the neural network or to obtain the estimates of matrices in the linear model used in the original OGY method. For the different control approaches, which make use of a neural network, the same amount of data is used in order to allow a fair comparison. This amount corresponds to the minimum number of training patterns needed to obtain a reasonable performance from the OGY controller employing a linear analytical system model extracted from the neural network. On the other hand, for a reasonable control performance a much higher amount of training data is needed when estimating the matrices by least mean square error criterion.

Training time: The total computation time (in seconds) spent during the data gathering and training phases. For the original OGY control method this time consists of data gathering only.

Average deviation from target: The root-mean-square deviation of the state(s) of the system from the target, after it has been stabilised. These values indicate the control quality.

Average control effort: The root-mean-square deviation of the control parameters from their nominal values during stabilisation of the target. It indicates the average control effort needed to stabilise the target.

Average convergence time: The average time (in terms of number of iterations) until the system converges to the target. For the sake of a fair comparison, the systems have been started from the same initial condition when simulating different control approaches.

Table 3 Simulation results for the three chaotic systems

System	Criteria	AM + OGY control	Classical OGY control	ANN based control	ANN + AM + OGY control
Logistic map	Radius of training region	-	0.5	0.5	0.5
	Maximum control region radius	0.48	0.45	0.45	0.48
	Training sample size	-	10,000	250	250
	Training time	-	25	63	32
	Average deviation from target	0.0000298	0.000455	0.00067	0.0000342
	Average control effort	0.000226	0.0072	0.08	0.000242
	Average convergence time	18	176	441	21
Henon map	Radius of training region	-	0.5	0.5	0.5
	Maximum control region radius	1.76	1.55	1.50	1.61
	Training sample size	-	10,000	1000	1000
	Training time	-	225	922	391
	Average deviation from target	0.0000509	0.0044	0.0054	0.000783
	Average control effort	0.0000921	0.0045	0.072	0.000932
	Average convergence time	65	528	553	486
Lorenz system	Radius of training region	-	1,5	1.5	1.5
	Maximum control region radius	-	2.32	2.20	2.42
	Training sample size	-	50,000	3500	3500
	Training time	-	4263	7213	5701
	Average deviation from target	-	0.056	0.063	0.019
	Average control effort	-	0.0422	0.0463	0.0157
	Average convergence time	-	284	292	122

The different control strategies, results of which are presented in Table 3 are as follows:

Analytic model + OGY control (AM+OGY): In this approach the local linear model is calculated from known system equations and is used by the linear controller given in eq. (2). Its results are given for the sake of comparison. For the Lorenz system this item is missing because an analytical description of the corresponding Poincaré map cannot be obtained although we know the differential equations governing the continuous-time system.

Classical OGY: In this approach a local linear model is obtained from the training data by least mean square error criterion and is used by the linear controller given in eq. (2).

ANN-based control: Here a neural network trained with system data directly provides the necessary control parameter variations.

Analytic model extracted from NN + OGY control (ANN+AM+OGY): This is the novel approach presented in this paper. A linear system model is extracted from the trained neural network by Taylor's expansion and is used by the linear controller as in the classical OGY method.

Table 3 reveals the slight disadvantages of the ANN-based control as compared to the classical OGY method in all respects except for the advantage in terms of the amount of training data needed. Furthermore, tables 2 and 3 demonstrate the superiority of the ANN+AM+OGY method over both other methods.

5. Discussion and Conclusion

The results obtained in this case study can be interpreted at various levels of generalisation:

a) This study, on basis of the specific example of chaos control, demonstrates the possibility of extracting analytical models from artificial neural networks.
   Whether the simple method of Taylor's expansion can be used for extracting an analytical model depends on the type of the problem. In the specific example of OGY-type chaos control Taylor's expansion provides a valid method because here a local model around the target-state and the nominal control parameter values is needed. In cases where such a condition is not valid, different methods may be needed for extracting analytic models from trained neural networks.
   It should also be noted that not all types of artificial neural networks lend themselves to Taylor's expansion. For example neural networks which use hard non-linearities are not suitable because their derivatives are undefined at the point of non-linearity. Also recurrent and feedback type neural networks are not suitable for Taylor's expansion.

b) The control quality depends both on the correctness of the model and the quality of the controller. In that respect, the results obtained for different model/controller combinations considered in this study are compared below:
   The first thing that strikes one's attention in table 2 is the superiority of the analytic model extracted from an ANN over the estimates obtained by least mean square error criterion (this can at least be directly observed in case of the maps because the true matrix values are given), although the former uses much less data than the latter. This is a result of the high approximation and interpolation capability of RBF-based neural networks as compared to the simple least mean square error approximation algorithm. The superiority of the control performance of ANN+AM+OGY as compared to the classical OGY control is due to the accuracy of the ANN+AM system model.

On the other hand, the control superiority (particularly apparent in terms of smaller "average deviation from target" and smaller "average control effort") of the ANN+AM+OGY method over the ANN-based control is a less trivial result. One may wonder why such a large difference in the control performance arises when in both cases neural networks are used trained with the same data. In this case the answer lies in the control function rather than the modelling. The ANN-based controller consists of a neural network with the current state and the desired next sate as its inputs and the control parameter (which will take the system to the desired state at the next iteration) as its output. The only way to use such a black box for stabilising the target is to feed the target-state as the desired state. This is, however, a rather primitive control approach that cannot make use of the theoretical knowledge, which tells that it may not be always possible to take the system to the target-state within one step. In such cases the output of the neural network will be a meaningless algorithmic artefact. As opposed to that, the OGY controller in the ANN+AM+OGY approach exploits System and Control Theory to produce meaningful control inputs, which drive the system closer to the target at each iterative step.

c) The following basic conclusions can be drawn from this analysis:

A modular structure where tasks at different hierarchical levels (in this case, modelling is a task at a lower hierarchical level and control at a higher level) are fulfilled by different parts is more efficient than a mono-block structure, which tries to fulfil all tasks at the same time. This is due to the fact that in a modular structure each part can be optimised for the particular task it is responsible for (in the case of ANN+AM+OGY, an artificial neural network for efficient data interpolation and a well-established analytic controller for stabilising the target).

Reduction of large amounts of raw data in terms of real system variables to a smaller number of symbolic parameters can provide an efficient way of data compression and data interpolation. This is what lies at the heart of neural networks, which -after being trained by tens of thousands of data patterns- are characterised by many (but definitely less than the amount of training data by orders of magnitude) weighting coefficients (plus centres and radii of radial basis function in case of RBF-based neural networks). This operation not only allows a huge data reduction but also results in a transformation from real variables to symbolic variables.

Efficient representation of knowledge gathered and refined from past experience (in this case, Dynamic System Theory and Control Theory) can only be realised in a highly symbolic manner (in this case, analytic descriptions of dynamic systems and analytic descriptions of controllers, which guarantee stability). This operation results in an even further data reduction (note the small number of parameters characterising a linear system model) and a transformation into more symbolic variables.

Tasks, where the relevant information contained in large amounts of data have to be efficiently combined with knowledge gained from past experience, require a hierarchical structure as a modelling and computation tool. In such a system lower level units have to handle and reduce large amounts data expressed in real variables into symbolic variables; whereas higher level units have to process symbolic results from lower level units.

d) Although the presented results originate from a rather simplistic engineering problem far from being representative even for the simplest task of the simplest vertebrate, the logical conclusions drawn from them inspire a generalisation to natural neural networks that make up the brains of higher vertebrates:

The modelling and control efficiency needed in face of the complex life tasks may have created a selective pressure not only towards larger brains but also towards more hierarchically organised brains. One of the main characteristics posed by the life tasks is the necessity to combine reliable past knowledge with limited amounts of up-to-date data. Past knowledge has to be represented in a highly compressed symbolic manner for the sake of efficiency. In the technical example presented in this paper this corresponds to the theoretical knowledge on system dynamics and linear control, which is not integrated into the artificial neural network but is provided by the operator.

A human being, if asked to perform a similar task, e.g. to model and control an unknown system, would also make use of some theoretical knowledge represented in a highly symbolic manner is stored in and retrieved from higher organisational levels of the synaptic network. In the terminology of the technical example presented in this study, it can be said that "the higher organisational levels of the human brain use the lower levels as intermediate computational tools". The natural neural network that makes up the human brain can be understood only in view of the complex hierarchy it exhibits. Only in view of this fact can the seeming distance between abstract and analytic thinking capacity of the human brain and its neural network structure be bridged.

Keeping in mind the complex life tasks, which require an appropriate combination of past knowledge with present data, which in return requires higher organisational levels operating in terms of more symbolic variables, one can claim that the human brain with its most symbolic mode of operation -analytic thinking- must have emerged under this selective pressure.

It should, however, be noted that the technical example considered in this study is too simplistic to account for a phenomenon that renders this selective pressure much more severe; namely the circular dynamic relationship between the model, the actions and the entities to be modelled (i.e. the self and the environment). The constructed model of the self and the environment gives rise to actions, which eventually alter both the self and the environment, and this circular relation constitutes one of the major positive feedback mechanisms that drive the increase of complexity [Karatay and Denizhan, 2000] both in the brains of higher vertebrates and in the environment.

References

Iplikci, S., *A Neural Network Based Local Control and Targeting Method for Chaotic Dynamics*, M.S. Thesis, Boğaziçi University, 1999.

Karacor, G., *Extraction of Local Analytical Model of a Chaotic System via Naural Networks*, M.S. Thesis, Boğaziçi University, 2002.
Karatay, V. and Denizhan, Y., "The Non-Decreasing Character of Complexity: A Biological Approach", International Journal of Computing Anticipatory Systems, ed. D.M. Dubois, Vol. 5, pp.169-182, 2000.
Orr, M., "Introduction to Radial Basis Functions", 1996, http://www.cns.ed.ac.uk/people/markintro/intro.html.
Ott, E., C. Grebogy, and J. Yorke, "Controlling Chaos", *Physical Review Letters,* Vol. 64, No.11, pp. 1196-1199, March 1990.
Turchin, V. "A Dialogue on Metasystem Transition", *World Futures: the Journal of General Evolution*, 1999.