A NOVEL DYNAMIC NEURAL NETWORK STRUCTURE FOR NONLINEAR SYSTEM IDENTIFICATION

Jiamei Deng [*]
Victor M. Becerra and Slawomir Nasuto [**]

[*] *Department of Cybernetics, University of Reading, Reading RG6 6AY, UK and Department of Automation, Shanghai University, Shanghai 200072, P.R.China*
[**] *Department of Cybernetics, University of Reading, Reading RG6 6AY, UK*

Abstract: Dynamic neural networks are often used for nonlinear system identification. This paper presents a novel series-parallel dynamic neural network structure which is suitable for nonlinear system identification. A theoretical proof is given showing that this type of dynamic neural network is able to approximate finite trajectories of nonlinear dynamical systems. Also, this neural network is trained to identify a practical nonlinear 3D crane system.

Keywords: recurrent neural networks, system identification, nonlinear systems

1. INTRODUCTION

The introduction of artificial neural networks methods for modelling, identification and control maybe the most innovative technical development in the past two decades in the control field (Narendra and Parthasarathy, 1990; Miller *et al.*, 1990). A recurrent neural network is a closed loop system, with feedback paths introducing dynamics into the model. They can be trained to learn the system dynamics without assuming much knowledge about the structure of the system under consideration.

Dynamic neural networks (DNNs) have important properties that make them convenient to be used together with nonlinear control approaches based on state space models and differential geometry (Garces *et al.*, 2003). The development of novel empirical model structures, such as DNNs, is a relevant challenge being addressed in this work. This paper presents a new dynamic neural network structure which is suitable for the identification of highly nonlinear systems. A theoretical proof showing how this dynamic neural network can approximate finite trajectories of general nonlinear dynamic systems is given. To illustrate the capabilities of the new structure, a network is trained to identify a real nonlinear 3D crane system. As the proposed dynamic neural network uses both the plant output and input for its training and operation, it can be classified as a series-parallel model (Narendra and Parthasarathy, 1990).

The paper is organized as follows. Section 2 discusses the universal approximation property of static multilayer perceptrons. Section 3 introduces the class of dynamic neural networks of interest in this paper. Section 4 discusses theoretical results on the approximation ability of dynamic neural networks. Section 5 presents an example. Finally, Section 6 gives concluding remarks.

2. THE UNIVERSAL APPROXIMATION PROPERTY OF STATIC MULTILAYER NETWORKS

An important result of approximation theory states that a three-layer feedforward neural network with sigmoidal activation functions in the hidden layer and linear activation functions in the output layer, has the ability to approximate any continuous mapping $f : \mathbb{R}^n \rightarrow \mathbb{R}^q$ to arbitrary precision, provided that the number of units in the hidden layer is sufficiently large.

The following theorem is a version of the fundamental approximation theorem provided by Funahashi (Funahashi, 1989). Similar results have been obtained by Cybenko (Cybenko, 1989) and others.

Theorem 1. Let K be a compact set of $\mathbb{R}^n$ and $f : K \rightarrow \mathbb{R}^q$ be a continuous mapping. Then, for arbitrary $\epsilon > 0$, there exists an integer N_h, a $q \times N_h$ matrix W_2, an $N_h \times n$ matrix W_1, and an N_h dimensional vector b such that:

$$\max_{x \in K} ||f(x) - W_2 \sigma(W_1 x + b)|| \leq \epsilon, \qquad (1)$$

where $\sigma : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$ is a sigmoid mapping whose elements are defined as follows:

$$\sigma(z) = \begin{bmatrix} \sigma(z_1) \\ \vdots \\ \sigma(z_{N_h}), \end{bmatrix} \qquad (2)$$

where $z = [z_1, \ldots, z_{N_h}]^T \in \mathbb{R}^{N_h}$.

For the proof of the above theorem, see (Funahashi, 1989).

3. DYNAMIC NEURAL NETWORKS

This paper concentrates on the properties of a class of dynamic neural networks henceforth known as Type 1 DNN.

Dynamic neural networks are made of interconnected dynamic neurons, also called units. The class of neuron of interest in this paper is described by the following differential equation:

$$\dot{x}_i = -\beta_i x_i + \sum_{j=1}^{N} \omega_{ij} \sigma(y_j) + \sum_{j=1}^{m} \gamma_{ij} u_j, \qquad (3)$$

where β_i, ω_{ij} and γ_{ij} are adjustable weights, with $1/\beta_i$ a positive time constant and x_i the activation state of the ith unit, y_j the actual system output or the hidden state of the jth unit, $\sigma : \mathbb{R} \rightarrow \mathbb{R}$ a sigmoid function and $u_1, \ldots, u_m$ the input signals.

A dynamic neural network is formed by a single layer of N units. The first n units are taken as the output of the network, leaving $N - n$ units as hidden neurons. A type 1 DNN is defined by the following vectorised expression:

$$\begin{aligned} \dot{x} &= -\beta x + \omega \sigma(y) + \gamma u \\ y_n &= C_n x \end{aligned} \quad , \qquad (4)$$

where x are coordinates on $\mathbb{R}^N$, $\beta \in \mathbb{R}^{N \times N}$ is a diagonal matrix with diagonal elements $\{\beta_1, \ldots, \beta_N\}$, $\omega \in \mathbb{R}^{N \times N}$, $\gamma \in \mathbb{R}^{N \times m}$ are weight matrices, $\sigma(x) = [\sigma(x_1), \ldots, \sigma(x_N)]^T$ is a vector sigmoid function, $u \in \mathbb{R}^m$ is the input vector, $y_n \in \mathbb{R}^n$ is the plant output vector, $y = [{y_n}^T, x_{n+1}, ..., x_N]^T$, $C_n = [I_{n \times N}, 0_{n \times (N-n)}]$.

A type 1 DNN differs from the dynamic neural network described in Chapter 4 of the book (Garces *et al.*, 2003), which in this paper is known as type 2 DNN, in the argument of the vector sigmoid function $\sigma(\cdot)$. A type 2 DNN is described by the following vectorised expression:

$$\begin{aligned} \dot{x} &= -\beta x + \omega \sigma(x) + \gamma u \\ y_n &= C_n x \end{aligned} \quad , \qquad (5)$$

Define the output state vector $x^p = [x_1^p, ..., x_n^p]^T = y_n$ as the internal state of the n output units. Define the hidden state vector $x^h = [x_1^h, ..., x_{N-n}^h]^T$ as the internal state of the $N - n$ hidden units. A type 1 DNN uses plant output and the hidden state in the argument of the vector sigmoid function $\sigma(\cdot)$, while a type 2 DNN uses the whole state vector of the network, which consists of the output states and the hidden states, in the argument of the vector sigmoid function. The difference is illustrated in Figure 1 and in Figure 2.

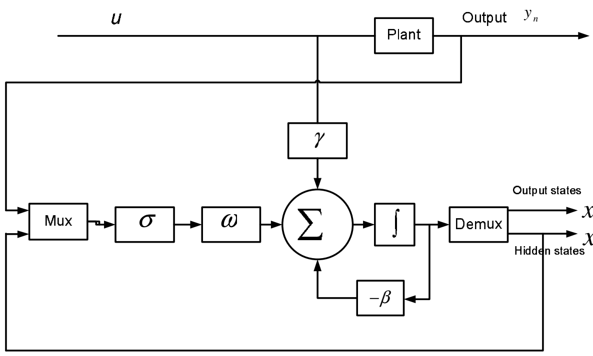

Fig. 1. Block diagram of type 1 DNN

4. APPROXIMATION ABILITY OF TYPE 1 DYNAMIC NEURAL NETWORKS

This section describes how any finite time trajectory of a given finite-dimensional non–autonomous dynamic system $\dot{x}(t) = f(x(t), u(t))$ can be approximated by a type 1 DNN. The theory uses the fundamental approximation theorem of neural networks and shows that, under certain conditions, there exists a dynamic neural network with a sufficient number of hidden units such that

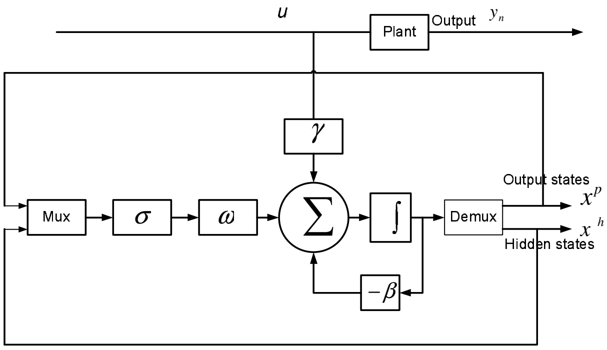

Fig. 2. Block diagram of type 2 DNN

the approximation error is bounded to a desired level. This theory is inspired by previous work on the approximation of finite trajectories of autonomous nonlinear systems (Funahashi and Nakamura, 1993; Kimura and Nakano, 1998). The book (Garces *et al.*, 2003) presents a theorem that shows that a type 2 DNN can approximate general nonlinear systems.

Corollary 1. (Garces *et al.*, 2003). Let K and U be compact subsets of $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively, and $f : K \times U \rightarrow \mathbb{R}^n$ be a continuous mapping. Then, for arbitrary $\epsilon > 0$, there exists an integer N_h, an $n \times N_h$ matrix W_2, an $N_h \times n$ matrix W_1, an $N_h \times m$ matrix γ_1, and an N_h dimensional vector b such that:

$$\max_{x \in K, u \in U} ||f(x,u) - W_2 \sigma(W_1 x + \gamma_1 u + b)|| \leq \epsilon, \quad (6)$$

where $\sigma : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$ is a sigmoid mapping whose elements are defined as follows:

$$\sigma(z) = \begin{bmatrix} \sigma(z_1) \\ \vdots \\ \sigma(z_{N_h}), \end{bmatrix} \quad (7)$$

where $z = [z_1, \ldots, z_{N_h}]^T \in \mathbb{R}^{N_h}$.

Proof. The proof follows directly from Theorem 1, by making the following substitutions: $K \leftarrow K \times U$, $q \leftarrow (n+m)$, $x \leftarrow [x^T\ u^T]^T$, $W_1 \leftarrow [W_1\ \gamma_1]$.

Theorem 2. Let D be an open subset of $\mathbb{R}^n$, and U and open subset of $\mathbb{R}^m$. Let $f : D \times U \rightarrow \mathbb{R}^n$ be a C^1-mapping, $u : [0,T] \rightarrow U$ be a C^1 function, $\tilde{K}$ be a compact subset of D. Suppose that there exists a set $K \subset \tilde{K}$ so that any solution $x(t)$ with initial value $x(0) \in K$ of the non-autonomous system

$$\dot{x}(t) = f(x(t), u(t)) \quad (8)$$

is defined on $I = [0,T]$ $(0 < T < \infty)$ for $u(t) \in U$ with $t \in I$, and is included in $\tilde{K}$ for any $t \in I$. Then, for an arbitrary $\varepsilon > 0$, there exists a non-autonomous dynamic neural network with n output units with states $x^o \in \mathbb{R}^n$ and N_h hidden units with states $x^h \in \mathbb{R}^{N_h}$, of the form:

$$\dot{z} = -\beta z + \omega \sigma(z) + \gamma \bar{u}, \quad (9)$$

where $z = [x^{oT}\ x^{h^T}]^T \in \mathbb{R}^{n+N_h}$, $\bar{u} = [u^T\ \dot{u}^T]^T \in \mathbb{R}^{2m}$, $\beta \in \mathbb{R}^{n+N_h \times n+N_h}$ is a diagonal matrix, $\omega \in \mathbb{R}^{n+N_h \times n+N_h}$ and $\gamma \in \mathbb{R}^{n+N_h \times 2m}$ are weight matrices, such that for a solution $x(t)$ satisfying Equation (8), and an appropriate initial state, the states of the output units of the network, $x^o(t)$, approximate the solution of the non-autonomous system:

$$\max_{t \in I} ||x(t) - x^o(t)|| < \varepsilon; I = [0,T] \quad (0 < T < \infty). \quad (10)$$

Proof. See the book (Garces *et al.*, 2003).

Theorem 3. Let D be an open subset of $\mathbb{R}^n$, and U and open subset of $\mathbb{R}^m$. Let $f : D \times U \rightarrow \mathbb{R}^n$ be a C^1-mapping, $u : [0,T] \rightarrow U$ be a C^1 function, $\tilde{K}$ be a compact subset of D. Suppose that there exists a set $K \subset \tilde{K}$ so that any solution $x(t)$ with initial value $x(0) \in K$ of the non-autonomous system

$$\dot{x}(t) = f(x(t), u(t)), \quad (11)$$

is defined on $I = [0,T]$ $(0 < T < \infty)$ for $u(t) \in U$ with $t \in I$, and is included in $\tilde{K}$ for any $t \in I$. Then, for an arbitrary $\varepsilon_1 > 0$, there exists a non-autonomous dynamic neural network with n output units with states $x^p \in \mathbb{R}^n$ and N_h hidden units with states $x^h \in \mathbb{R}^{N_h}$, of the form:

$$\dot{z} = -\beta z + \omega \sigma(z_1) + \gamma \bar{u}, \quad (12)$$

where $z = [x^{pT}\ x^{h^T}]^T \in \mathbb{R}^{n+N_h}$, $z_1 = [x^T\ x^{h^T}]^T \in \mathbb{R}^{n+N_h}$, $\bar{u} = [u^T\ \dot{u}^T]^T \in \mathbb{R}^{2m}$, $\beta \in \mathbb{R}^{n+N_h \times n+N_h}$ is a diagonal matrix, $\omega \in \mathbb{R}^{n+N_h \times n+N_h}$ and $\gamma \in \mathbb{R}^{n+N_h \times 2m}$ are weight matrices, such that for a solution $x(t)$ satisfying Equation (11), and an appropriate initial state, the states of the output units of the network, $x^p(t)$, approximate the solution of the non-autonomous system:

$$\max_{t \in I} ||x(t) - x^p(t)|| < \varepsilon_1; I = [0,T] \quad (0 < T < \infty). \quad (13)$$

Proof. This proof uses Lemmas 1, 2 and 3, which are given in the appendix. For given $\varepsilon_1 > 0$, choose $\varepsilon > 0$, $\varepsilon_2 > 0$ and such that $\varepsilon + \varepsilon_2 \leq \varepsilon_1$, $\varepsilon_2 \leq \frac{\eta_1 l_G}{exp(l_G T - 1)}$. Define now the mapping F : $\mathbb{R}^{n+N_h} \times \mathbb{R}^{2m} \rightarrow \mathbb{R}^{n+N_h}$ as follows:

$$F(z, \bar{\bar{u}}) = -\beta z + \omega \sigma(z) + \bar{\gamma} \bar{\bar{u}}. \quad (14)$$

Then the dynamic system defined by F is:

$$\dot{z} = -\beta z + \omega \sigma(z) + \bar{\gamma} \bar{\bar{u}}, \quad (15)$$

where $\bar{\bar{u}} = [\bar{u}\ \delta z]^T$, $\delta z = [\delta x\ 0_{N_h \times N_h}]^T$, $\delta x = [x - x^p]$, $\bar{\bar{\gamma}} = [\bar{\gamma}\ 0_{(n+N_h) \times (n+N_h)}]$. Equation (15) is equivalent to Equation (9).

Define a new mapping $\tilde{F} : \mathbb{R}^{n+N_h} \times \mathbb{R}^{2m} \rightarrow \mathbb{R}^{n+N_h}$ as follows:

$$\tilde{F}(\tilde{z}, \bar{\bar{u}}) = -\beta\tilde{z} + \omega\sigma(\tilde{z} + [0_{n\times n}\ I_{(n+N_h)\times(n+N_h)}]\bar{\bar{u}}) + \bar{\gamma}\bar{\bar{u}}. \tag{16}$$

Then the dynamic system defined by $\tilde{F}$ is:

$$\dot{\tilde{z}} = -\beta\tilde{z} + \omega\sigma(\tilde{z} + [0_{n\times n}\ I_{(n+N_h)\times(n+N_h)}]\bar{\bar{u}}) + \bar{\gamma}\bar{\bar{u}}. \tag{17}$$

Equation (17) is equivalent to Equation (12). Let l_G is the Lipschitz constant of F in z. It is not difficult to infer that $\tilde{F}$ is also Lipschitz, so that Lemma 2 is applicable to F and $\tilde{F}$.

Note that

$$||F(\tilde{z}, \bar{\bar{u}}) - \tilde{F}(\tilde{z}, \bar{\bar{u}})|| = ||\omega|| \cdot ||\sigma(z) - \sigma(z + \delta z)|| \tag{18}$$

Suppose that x_i is an element of z and that δx_i is an element of δz. Sigmoid function is a continuous and differentiable function. By using Taylor expansion to this sigmoid function:

$$||\sigma({x^o}_i) - \sigma({x^o}_i + \delta x_i)|| =$$
$$|| - \sigma'({x^o}_i)\delta x_i - \frac{1}{2}\sigma''({x_o}^i){\delta x_i}^2 - \cdots - O({\delta x_i}^n)|| \tag{19}$$

where

$$O({\delta x_i}^n) = \int_z^{z+\delta z} f^{(n+1)}(t)\frac{(z-t)^n}{n!}dt, \tag{20}$$

by using Lemma 3

$$O({\delta x_i}^n) = \sigma^{(n+1)}(\zeta)\frac{(z-\zeta)^n}{n!}\delta z, \tag{21}$$

for $\zeta \in [z,\ z+\delta z]$, therefore,

$$O({\delta x_i}^n) \le \sigma^{(n+1)}(\zeta)\frac{{\delta z}^{n+1}}{n!}, \tag{22}$$

According to Equation (22), Equation (19) becomes

$$||\sigma({x^o}_i) - \sigma({x^o}_i + \delta x_i)|| \le \delta x_i d \le \varepsilon d, \tag{23}$$

where $d = || - \sigma'({x^o}_i) - \frac{1}{2}\sigma''({x_o}^i)\delta x_i - \cdots - \sigma^{(n+1)}(\zeta)\frac{\delta z^n}{n!}||$ is bounded. In conclusion, Equation (23) can be written as:

$$||\sigma({x^o}_i) - \sigma({x^o}_i + \delta x_i)|| \le \varepsilon d, \tag{24}$$

According to Equation (24), Equation (18) can be written as:

$$||F(\tilde{z}, \bar{\bar{u}}) - \tilde{F}(\tilde{z}, \bar{\bar{u}})|| \le ||\omega|| d\varepsilon, \tag{25}$$

Equation (25) can be written as:

$$||F(\tilde{z}, \bar{\bar{u}}) - \tilde{F}(\tilde{z}, \bar{\bar{u}})|| \le \eta_1, \tag{26}$$

by using Lemma 2

$$||x^o(t) - x^p(t)|| \le \frac{\eta_1}{l_G}(exp(l_G t) - 1), \tag{27}$$

$$\max_{t\in I} ||x^o(t) - x^p(t)|| < \varepsilon_2, \tag{28}$$

$$\begin{aligned} &\max_{t\in I} ||x(t) - x^p(t)|| \le \\ &\max_{t\in I}(||x(t) - x^o(t)|| + ||x^o(t) - x^p(t)||) \le \\ &\max_{t\in I} ||x(t) - x^o(t)|| + \max_{t\in I} ||x^o(t) - x^p(t)|| \le \\ &(\varepsilon_2 + \varepsilon) \le\ \varepsilon_1. \end{aligned} \tag{29}$$

which completes the proof. ■

5. EXAMPLE

The 3D crane consists of a payload hanging on a pendulum-like lift-line wound by a motor mounted on a cart (Figure 3). The 3D crane system is multivariable, it exhibits highly nonlinear dymamics, and has oscillatory behaviour with different time scales, which makes it a challenging benchmark for nonlinear identification, particularly with recurrent model structures. The payload is lifted and lowered in the z direction. Both the rail and the cart are capable of horizontal motion in the x direction. The cart is capable of horizontal motion along the rail in the y direction. Therefore the payload attached to the end of the lift-line can move freely in 3 dimensions. The 3D crane is driven by the three DC motors and is fully interfaced to MATLAB and SIMULINK. The crane has three manipulated inputs, which are the references to PWM circuits that drive three DC motors, and five measurements obtained via optical encoders.

The schematic diagram of the 3D crane is given in Figure 4.

Fig. 3. The 3D crane system setup.

There are five measured quantities:

- x_w (not shown in Figure 4) denotes the distance of the rail with the cart from the centre of the construction frame;
- y_w (not shown in Figure 4) denotes the distance of the cart from the centre of the rail;
- R denotes the length of the lift-line;
- α denotes the angle between the y axis and the lift-line;
- β denotes the angle between the negative direction on the z axis and the projection of the lift-line onto the xz plane.

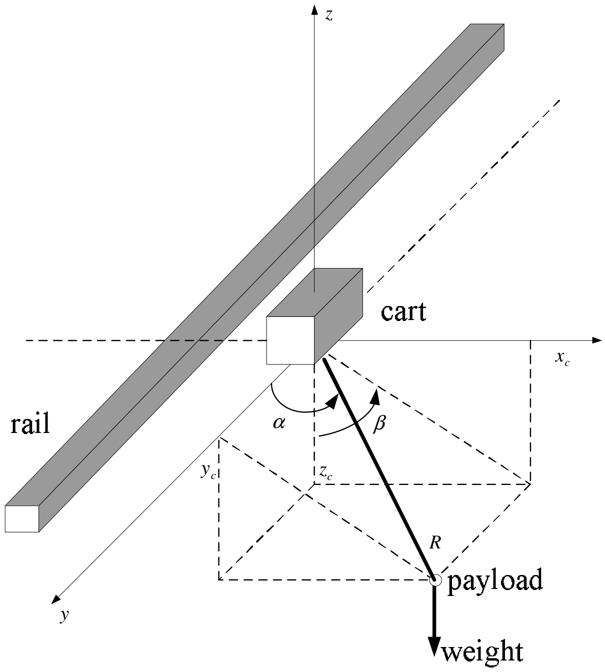

Fig. 4. 3D crane system: coordinates and forces.

The position in cartesian co-ordinates of the payload is denoted by x_c, y_c, z_c and can be found from the five measurements using kinematic equations.

Two neural networks of type 1 and 2 were used to identify three-input three-output models, which had as inputs the three reference voltages to the PWM circuits and as outputs the three coordinates of the payload position. Training was performed using a genetic algorithm with real enconding (Deng and Becerra, 2003). In both cases, a 6-state dynamic neural network structure was chosen. Figure 5 shows the training output and the model output using the type 1 neural network. Figure 6 shows the validation output and model output for the same case. Figure 7 shows the training output and the model output for type 2 dynamic neural network. Figure 8 shows the validation data and model output for the same case.

It is not difficult to see that a type 2 DNN had problems to approximate the dynamic behaviour of the system, whereas the type 1 DNN, which was easier to train, was able to approximate the system more accurately. The better approximation capability exhibited by the type 1 DNN can be attributed to the fact that this structure uses both output and input information, as it is a series-parallel model.

6. CONCLUSIONS

This paper presented a novel dynamic neural network structure and it has been proved that the network has the ability to approximate finite trajectories of non-autonomous nonlinear dynamic systems. An example has been given to demonstrate the effectivity of the proposed structure in approximating complex nonlinear dynamics, and its performance has been favourably compared,

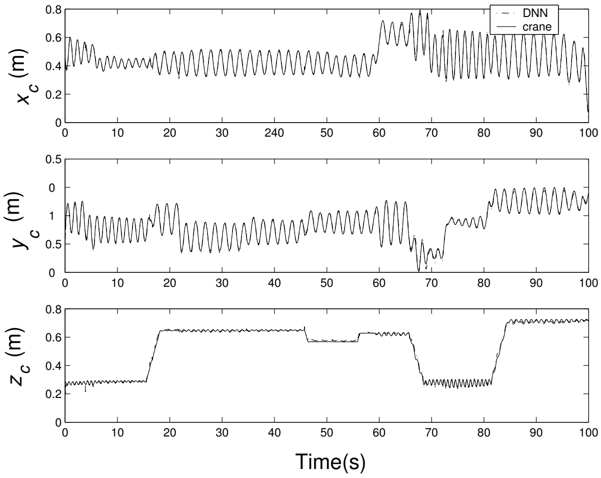

Fig. 5. Training trajectories and model outputs using the type 1 DNN

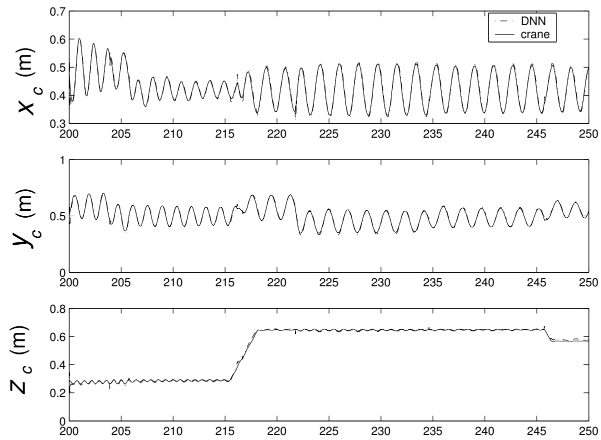

Fig. 6. Validation trajectories and model outputs using the type 1 DNN

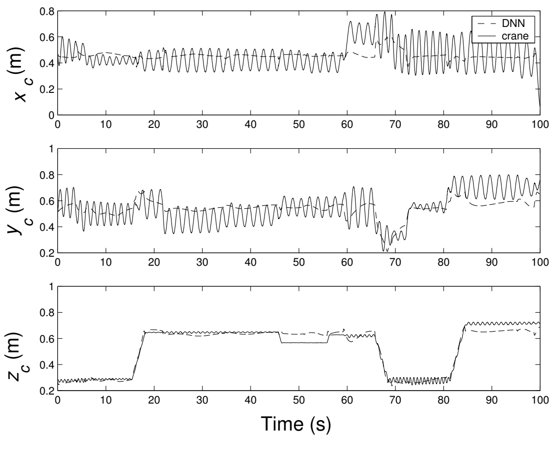

Fig. 7. Training trajectories and model outputs using the type 2 DNN

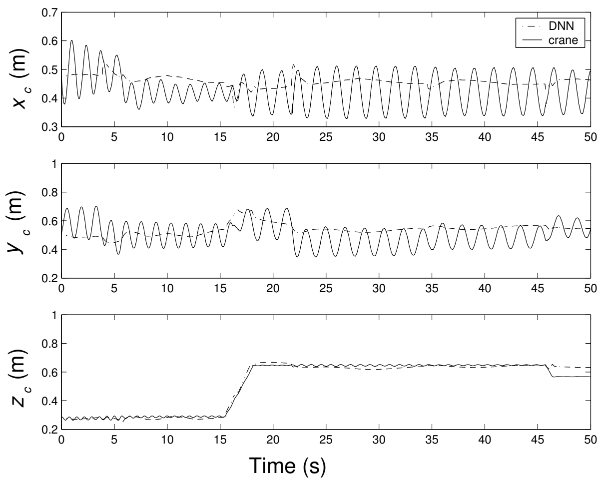

Fig. 8. Validation trajectories and model outputs using the type 2 DNN

in terms of training difficulty and approximation ability, with a previously proposed dynamic neural network structure.

7. ACKNOWLEDGEMENT

This work has been supported by the UK EPSRC under Grant Reference Number GR/R64193, and the University of Reading Chinese Studentship Programme.

APPENDIX

The following Lemmas are useful for the proof of Theorem 3.

Lemma 1. (Gronwall's inequality). Let $v : [t_0, t_f] \rightarrow \mathbb{R}$ be continuous and nonnegative. Suppose that $C \geq 0$ and $L \geq 0$ are real numbers such that

$$v(t) \leq C + \int_{t_0}^{t} Lv(\tau)d\tau \qquad (.1)$$

for all $t \in [t_0, t_f]$. Then

$$v(t) \leq C \exp(L|t - t_0|) \qquad (.2)$$

for all $t \in [t_0, t_f]$

Proof. See Chapter 8 of (Hirsch and Smale, 1974).

Lemma 2. Let F, $\tilde{F} : S \times U \rightarrow \mathbb{R}^n$ be Lipschitz continuous mappings and L be a Lipschitz constant of $F(x, u)$ in x on $S \times U$. Suppose that for all $x \in S$ and $u \in U$:

$$||F(x, u) - \tilde{F}(x, u)|| < \varepsilon \qquad (.3)$$

If $x(t)$ and $\tilde{x}(t)$, are solutions to

$$\begin{aligned} \dot{x} &= F(x, u) \\ \dot{\tilde{x}} &= \tilde{F}(\tilde{x}, u) \end{aligned} \qquad (.4)$$

respectively, on some interval $I = \{t \in \mathbb{R} | t_0 \leq t \leq t_f\}$, and $x(t_0) = \tilde{x}(t_0)$, then

$$||x(t) - \tilde{x}(t)|| \leq \frac{\varepsilon}{L} (\exp(L|t - t_0|) - 1) \qquad (.5)$$

holds for all $t \in I$.

Proof. Please see Chapter 15 of (Hirsch and Smale, 1974).

Lemma 3. Let $f(x)$ be an integrable function in the interval (a, b). A point c can be found between a and b such that

$$\int_a^b f(x)dx = f(c)(a - b) \qquad (.6)$$

Proof. See Chapter XIII of (Khinchin, 1960).

REFERENCES

Cybenko, G. (1989). Approximation by superpositions of sigmoidal functions. *Mathematics of Control, Signals and Systems* **2**, 303–314.

Deng, J. and V. M. Becerra (2003). Modelling of a 3d crane system using parallel dynamic neural networks. In: *Proceeding 2nd IEEE System, Man and Cybernetics UK and Republic of Ireland Chapter Conference: Cybernetic Intelligence, Chanllenges and Advances.* Reading, UK. pp. 148–153.

Funahashi, K. (1989). On the approximate realization of continuous-mappings by neural networks. *Neural Networks* **2**(3), 183–192.

Funahashi, K. and Y. Nakamura (1993). Approximation of dynamical-systems by continuous-time recurrent neural networks. *Neural Networks* **6**(6), 801–806.

Garces, F., V.M. Becerra, C. Kambhampati and K. Warwick (2003). *Strategies for feedback linearisation: a dynamic neural network approach.* Springer. London.

Hirsch, M. W. and S. Smale (1974). *Differential equations, dynamical systems and linear algebra.* Academic Press. San Diego, CA.

Khinchin, A.A. (1960). *A course of mathematical analysis.* Hindustan Publishing CORP.. Delhi.

Kimura, M. and R. Nakano (1998). Learning dynamical systems by recurrent neural networks from orbits. *Neural Networks* **11**(9), 1589–1599.

Miller, W. T., R.S. Sutton and P. J. Werbos (1990). *Neural networks for control.* MIT Press, Cambridge.

Narendra, K. S. and K. Parthasarathy (1990). Application and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks* **1**, 4–27.