

## PROCESS DYNAMIC MODELLING USING CONTINUOUS TIME LOCAL MODEL NETWORKS

Séamus Mc Loone and George Irwin

*Advanced Control Engineering Research Group,  
Dept. of Electrical and Electronic Engineering,  
The Queen's University of Belfast,  
Belfast, UK, BT9 5AH*

*seamus.mcloone@qub.ac.uk, g.irwin@qub.ac.uk*

**Abstract:** The capabilities of the continuous-time Local Model (LM) network for representing a non-linear dynamic process are studied by simulation of a coupled tank system. The paper shows how normalisation of the network weighting functions limits the modelling accuracy and proposes a solution via the addition of constant bias terms to each of the individual local models. *Copyright © 1998 IFAC*

**Keywords:** Multiple model, continuous-time modelling, non-linear modelling, Local Model Networks

### 1. INTRODUCTION

There has been considerable research interest in neural networks for the identification and control of non-linear dynamic systems in recent years, (Hunt, Irwin and Warwick, 1995) and (Hunt et al., 1992). Although there has been significant progress, including industrial applications in the fields of aerospace (Morita, 1993), robotics (Pham et al., 1994), power generation (Brown et al., 1995) and chemical (Lightbody et al., 1994), there remain a number of significant disadvantages to be addressed. These include a lack of transparency in the identified models which are inherently 'black-box' in nature, the difficulty in incorporating 'a-priori' plant knowledge which is usually available in practice and the weakness of theoretical support which is particularly evident when neural models are employed for control purposes.

An alternative modelling strategy, which addresses many of these limitations, is the Local Model (LM) network (Murray-Smith and Johansen, 1997), first studied by (Johansen and Foss, 1992, 1993). A LM network comprises a set of local models and

associated validity functions. The non-linear representation can be either discrete- or continuous-time, depending on the nature of the local model employed. Although the majority of the work to date has concentrated on the former, (Hunt, Johansen et al., 1996) and (Johansen and Foss, 1995), there has also been some progress on continuous-time LM networks, (Gawthrop, 1995).

In this paper the capabilities of the continuous-time Local Model (LM) network for representing a non-linear dynamic process is studied by simulation of a coupled tank system. The paper shows how normalisation of the network weighting functions limits the modelling accuracy and proposes a solution via the addition of constant bias terms to each of the individual local models.

### 2. LOCAL MODEL NETWORKS

Local Model (LM) networks were first described by (Johansen and Foss, 1992, 1993). A local model network (see Fig. 1) is a set of models, each valid for a specific regime in the operating space, weighted by

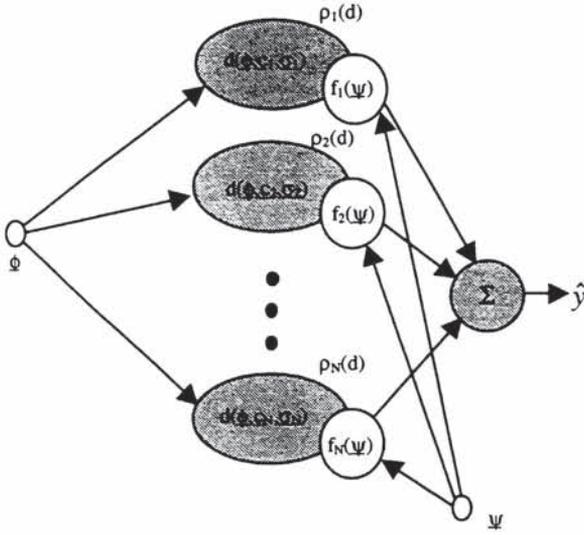


Fig. 1: General Architecture of a LM Network

some activation function. The same inputs,  $\underline{\psi}$ , are fed to all the models and the outputs are weighted according to some scheduling variable or variables,  $\underline{\phi}$ . The underlying local models can be either linear or nonlinear. The LM network output is given by:

$$\hat{y} = \sum_{i=1}^N \rho_i(d(\underline{\phi}, \underline{c}_i, \underline{\sigma}_i)) f_i(\underline{\psi}) \quad (1)$$

where  $\rho_i(d(\underline{\phi}, \underline{c}_i, \underline{\sigma}_i))$  is the basis function (in this case, a Gaussian function) of the  $i^{\text{th}}$  model,  $\underline{\phi}$  is a vector of scheduling variables,  $N$  is the number of models in the network and  $f_i(\underline{\psi})$  is the  $i^{\text{th}}$  local model output.

The LM network can be viewed as a generalisation of the RBF neural network. In the latter case, the weights associated with each basis function are constant parameters. In the LM network, these coefficients have been generalised to include more powerful functions of the inputs,  $\underline{\psi}$ . This means that a smaller number of local models can cover larger operating regimes of the input domain as illustrated in Fig. 2.

Obviously, there is a trade-off between the number and size of the operating regimes on the one hand, and the complexities of the local models on the other (Murray-Smith and Johansen, 1997). For instance, at one extreme one can have only one large operating regime that covers the full range of operation and, therefore, the local model must typically be complex since it is actually the global model. In general, a decomposition into a few *large* operating regimes will require more complex local models than a decomposition into numerous *small* operating regimes, see Fig. 2 (b) and (c). On the other extreme, one can partition the input domain into a large number of operating regimes, so that the function to be approximated can be represented by constant values locally, see Fig. 2 (a). The latter function

approximation is the principle underlying RBF networks.

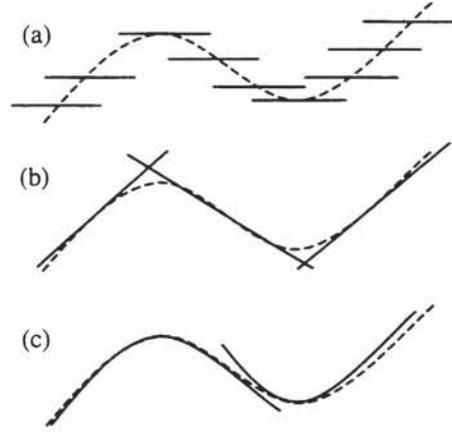


Fig. 2: Function Approximation by Local Models: (a) local constant functions, (b) local linear functions, (c) local quadratic functions.

### 2.1 Weighting Functions

In the local model network, the weighting of the local models is calculated using weighting or *activation* functions. The most commonly used weighting functions are the Gaussian basis functions, given by:

$$\rho_i = d(\underline{\phi}, \underline{c}_i, \underline{\sigma}_i) = \exp\left[-\frac{\|\underline{\phi} - \underline{c}_i\|^2}{\underline{\sigma}_i}\right] \text{ for } i = 1, 2, \dots, N \quad (2)$$

where  $\underline{c}_i$  defines the Gaussian centre,  $\underline{\sigma}_i$  defines the Gaussian width and  $\rho_i$  is the  $i^{\text{th}}$  activation function output. The scheduling variables,  $\underline{\phi}$ , can be a function of a system state, an input, and/or some other system parameter. The number of centres and widths defining the Gaussian function depends on the number of scheduling variables used in the local model network. For one scheduling variable the Gaussian activation function is simply a 2-D bell-shaped curve while for two scheduling variables it is a 3-D dome-shaped surface.

It is common practice to use normalised Gaussian basis functions, such that every point in the input space is covered by the basis functions to the same degree, i.e. the basis functions form a *partition of unity* across the input space. The normalised Gaussian function output is given by:

$$\bar{\rho}_i = \frac{\rho_i}{\sum_{j=1}^N \rho_j} \quad (3)$$

### 3. THE COUPLED TANK SYSTEM

The simulated application is a standard one for control studies and is a highly complex, nonlinear plant. Fig. 3 shows a simplified diagram of the coupled tank system.

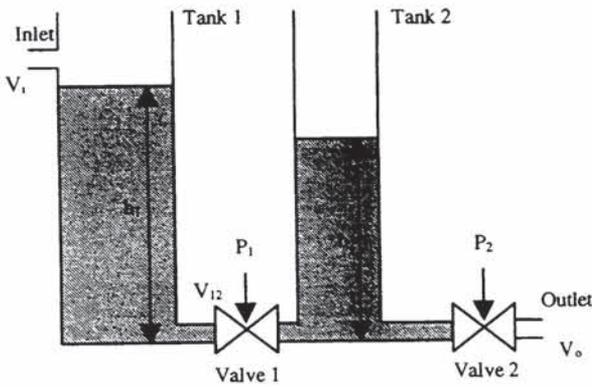


Fig. 3 Coupled Tank System

$V_1$ ,  $V_0$  and  $V_{12}$  represent the input flow rate, the output flow rate and the flow rate between the tanks respectively. The height of water in tank 1 is given by  $h_1$ , while the height of the water in tank 2 is  $h_2$ . If  $A_1$  is the cross-sectional area of tank 1 and  $A_2$  the cross-sectional area of tank 2 then the set of equations representing the coupled tank system is as follows:

$$\begin{aligned} \dot{h}_1 &= (V_1 - V_{12}) / A_1 \\ \dot{h}_2 &= (V_{12} - V_0) / A_2 \\ V_{12} &= K_1(P_1) \sqrt{h_1 - h_2} \\ V_0 &= K_2(P_2) \sqrt{h_2} \end{aligned} \quad (4)$$

where  $K_1$  and  $K_2$  are functions of the valve positions,  $P_1$  and  $P_2$  respectively.

Fig. 4 shows a graph of the flow characteristics for the valves in the coupled tank system, (Heckenthaler and Engell, 1994). This graph, along with the square roots in equations (4), illustrates the nonlinearity of this system.

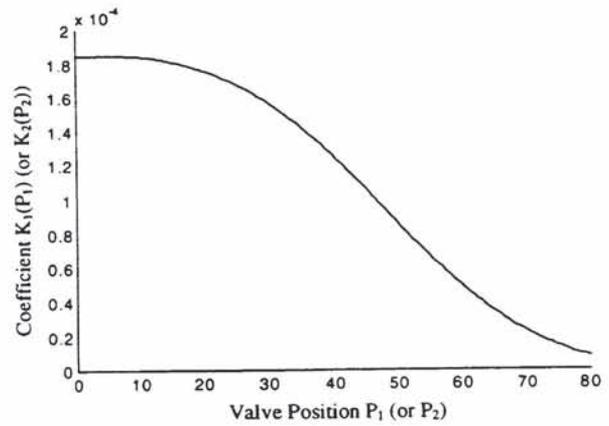


Fig. 4 Valve Flow Characteristics

### 4. LM NETWORK MODELLING OF THE TANK SYSTEM

It was decided to model the response of  $h_1$  to changes in  $P_1$  and  $P_2$ , as  $h_2$  was not affected by  $P_1$  in the steady-state. The steady-state output of  $h_1$  is shown in Fig. 5. Nine local second order linear state space models were developed at various operating points throughout the operating range. Normalised Gaussian functions were used to interpolate between the models with the centres and widths optimised using the BFGS (Broyden, Fletcher, Goldfarb and Shanno) Quasi-Newton algorithm, (Gill, Murray and Wright, 1981).

#### 4.1 Initial Modelling Results

Fig. 6 shows the steady-state output of the LM network, while the error between the actual steady-state output of the plant and that of the LM network is shown in Fig. 7. It can be seen that the LM network output is reasonably inaccurate. This is due to the fact that normalisation, defined in (3), limits the modelling accuracy of the LM network.

Normalisation causes the Gaussian functions to sum to unity throughout the operating region. In doing so, the steady-state output of the LM network is forced to lie within the space bounded by the local models chosen. Fig.8 illustrates this using a simple example, whereby two linear models are used to represent the steady-state curve shown. The steady-state output of the LM network in this particular case is forced to lie within the shaded area. It can be clearly seen that, despite any improvement in the weighting functions or change in the positioning of the linear models, the LM network will not accurately represent the nonlinear plant in the region between the operating points and, indeed, in the regions outside the operating points also.

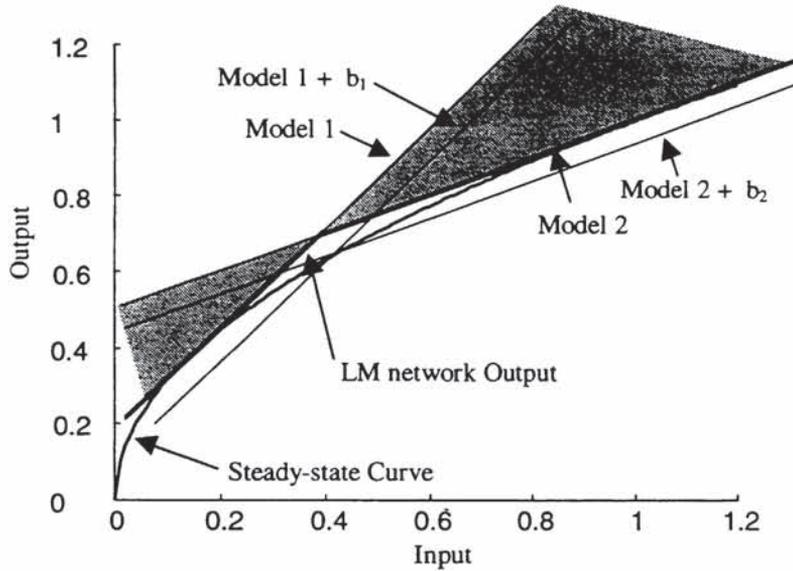


Fig. 8 Effect of Adding Constant Bias to each Model

Introducing more models will reduce the actual steady-state error but there will always exist some residual error that cannot be eliminated using this approach.

#### 4.2 Bias Term Modelling

By adding constant bias terms,  $b_1$  and  $b_2$ , to the two models in Fig. 8, they can be shifted upwards or downwards accordingly. An accurate representation may be achieved by optimising these bias terms. It is worth noting that, in order to achieve a completely accurate representation, the point of intersection between the two locally valid models needs to lie on the actual steady-state curve of the nonlinear plant.

This concept was applied to the modelling of the coupled tank system, i.e. a constant bias was added to each of the nine models. These bias terms, along with the centres and widths of the weighting functions, were optimised to give a minimal modelling error. The steady-state output is shown in Fig. 9 with the error displayed in Fig. 10. These results clearly show a remarkable improvement in the modelling accuracy of the LM network.

## 5 RESULTS

A series of step changes in  $P_1$ , with  $P_2$  held constant, was used to test the dynamic response of the LM network, see Fig. 11. The LM network with biased local models produces a better response than the original LM network.

In the latter case, 'jumps' in the response are apparent. These 'jumps' occur when the network is traversing between local models and the reason for this again lies with normalisation. The LM network is optimised so as to give a minimum error between the LM network output and the plant output. From Fig. 8, it is plain to see that the minimum error is achieved when the output of the LM network is as indicated. The transition between the local models is, in this case, not smooth and hence the 'jump' in the step response.

The addition of the bias terms results in a smooth and continuous steady-state LM network output and, hence, no 'jumps' in the relevant output in Fig. 11.

## 6 CONCLUDING DISCUSSION

The LM network is simply a set of local models, based at various operating points throughout the operating range, and a set of basis functions that interpolates between these models. In this paper local linear models were used along with normalised Gaussian activation functions to model a Coupled Tank System.

Normalisation of the Gaussian functions has the advantage of ensuring that every point in the operating region is covered by the basis function to the same degree. However, it can limit the modelling accuracy of the LM network and can sometimes result in undesirable dynamical responses.

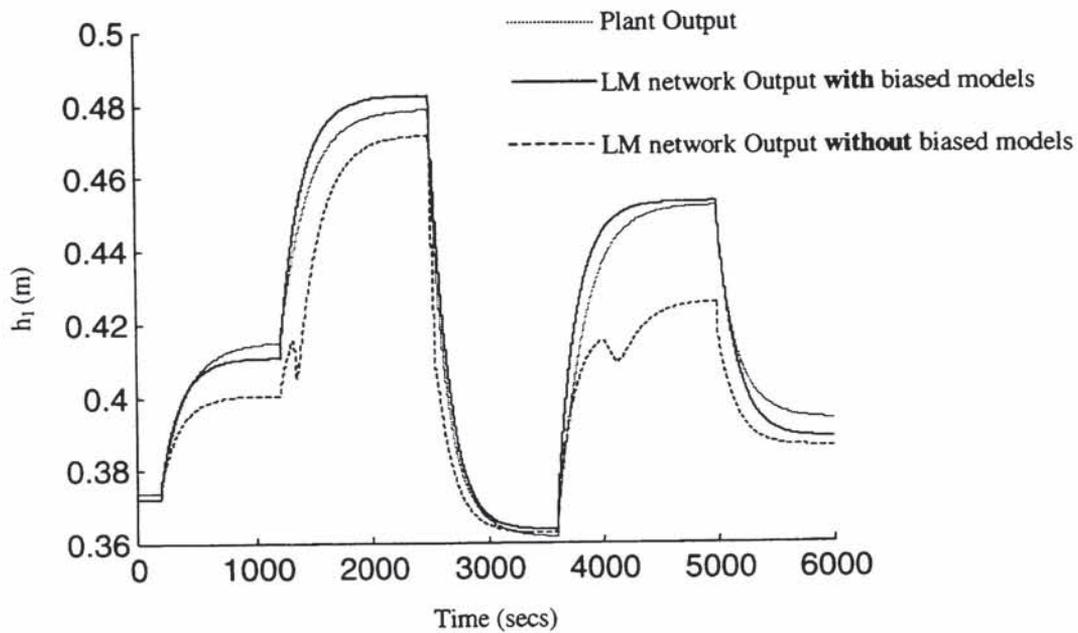


Fig. 11 Dynamical Responses of Various Systems

The steady-state output of the local models needs to be changed in order to obtain a LM network with a smooth continuous steady-state response. This was achieved by adding a constant bias to each of the local linear models. With the incorporation of these bias terms, it was possible to develop a local model network that accurately modelled the example system. In doing so, the disadvantages due to normalisation were overcome, while the associated advantages were maintained.

In this paper the input operating range was decomposed manually and suitable operating points chosen accordingly. Future work includes investigating operating regime decomposition and locating the optimal operating points for a nonlinear plant.

#### 7 REFERENCES

- Brown, M. D., Irwin, G. W., Hogg, B. W. and Swidenbank, K. E. (1995). Neural network modelling of a 200 MW boiler system, *IEE Proc. Control Theory & Applications*, **142**, No. 6, 529-536.
- Gawthrop, P. J. (1995). Continuous-time local state local model networks. *Proc. IEEE Conf. Systems, Man and Cybernetics, Vancouver, Canada*, 852-857.
- Gill, P. E., Murray, W. and Wright, M. H. (1981). *Practical Optimisation*, Academic Press, London.
- Heckenthaler, T. and Engell, S., (1994). Approximately time-optimal fuzzy control of a two-tank system. *IEEE Control Systems*, 24-30.
- Hunt, K. J., Sbarbaro, D., Zbikowski, R. and Gawthrop, P. J. (1992). Neural networks for control systems – A survey, *Automatica*, **28**, No.6, 1083-1112.
- Hunt, K.J., Irwin, G. W. and Warwick, K. (1995). *Neural Network Engineering in Dynamic Control Systems*, Springer.
- Hunt, K. J., Kalkkuhl, J. C., Fritz, H. and Johansen, T. A. (1996). Constructive empirical modelling of longitudinal vehicle dynamics using local model networks. *Control Engineering Practice*, **4**, 167-178.
- Johansen, T. A. and Foss, B. A. (1992) A NARMAX model representation for adaptive control based on local models. *Modelling, Identification and Control* **13**, 25-39.
- Johansen, T. A. and Foss, B. A. (1993). Constructing NARMAX models using ARMAX models. *Int. J. Control* **58**, 1125-1153.
- Johansen, T. A. and Foss, B. A. (1995). Empirical modelling of a heat transfer process using local models and interpolation. *Proc. American Conference, Seattle, WA*, 3654-3658.
- Lightbody, G., Irwin, G. W., Taylor, A., Kelly, K. and McCormick, J. (1994). Neural network modelling of a polymerisation reactor, *Proc. IEE Int. Conf. Control '94*, **1**, 237-242.
- Morita, S. (1993). Optimisation control for combustion parameters of petrol engines using neural networks – in the case of on-line control, *Int. J. of Vehicle Design*, **14**, No. 5/5, 552-563.
- Murray-Smith, R. and Johansen, T. A. (1997). *Multiple Model Approaches to Modelling and Control*, Taylor & Francis, Chapter 1, 3-72.
- Pham, D. T. and Oh, S. J. (1994). Adaptive control of a robot using neural network controllers and filters, *IEEE Trans. Neural Networks*, **5**, No.2, 198-212.

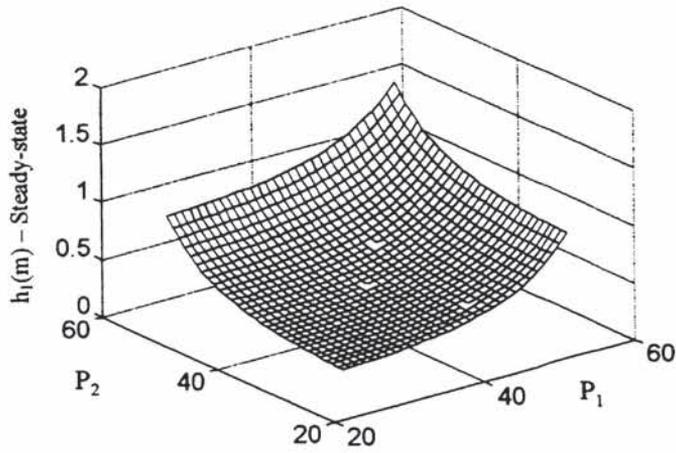


Fig. 5 Variation in steady-state height  $h_1$  with valve positions  $P_1$  and  $P_2$

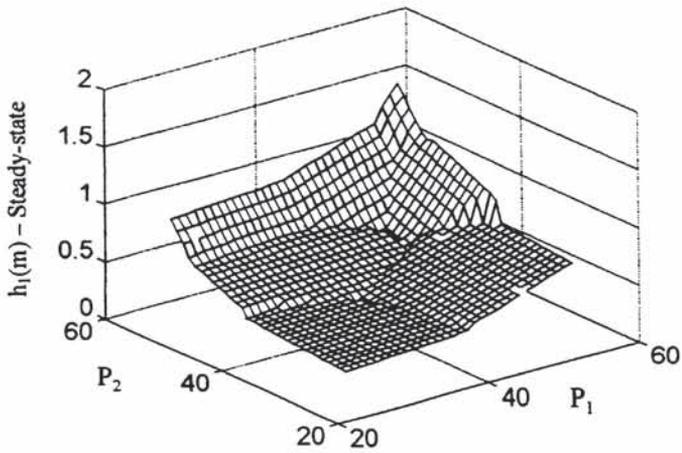


Fig. 6 LM network Output

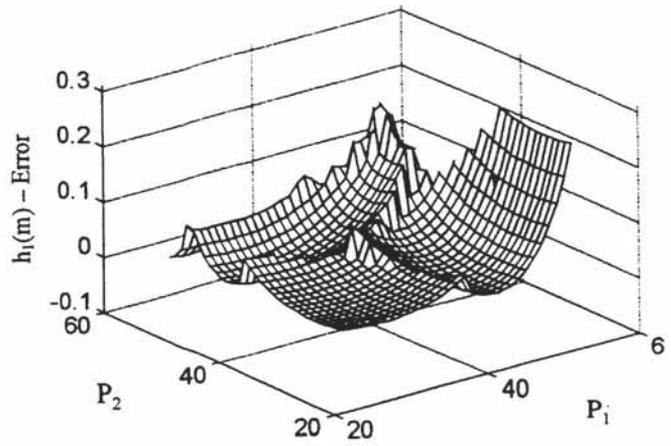


Fig. 7 Modelling Error in LM network Output

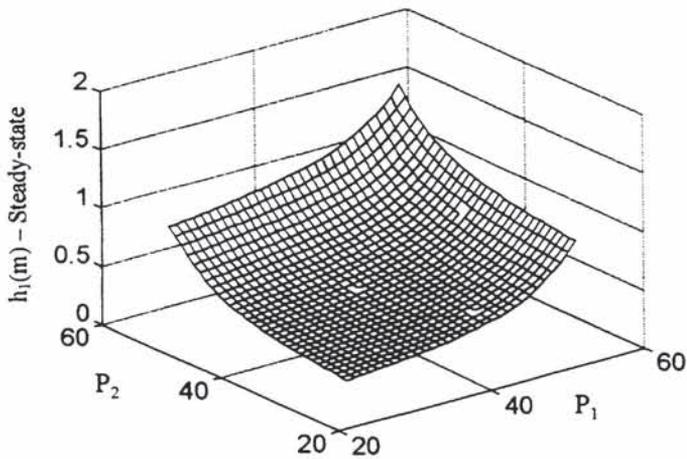


Fig. 9 Output of LM network with biased local models

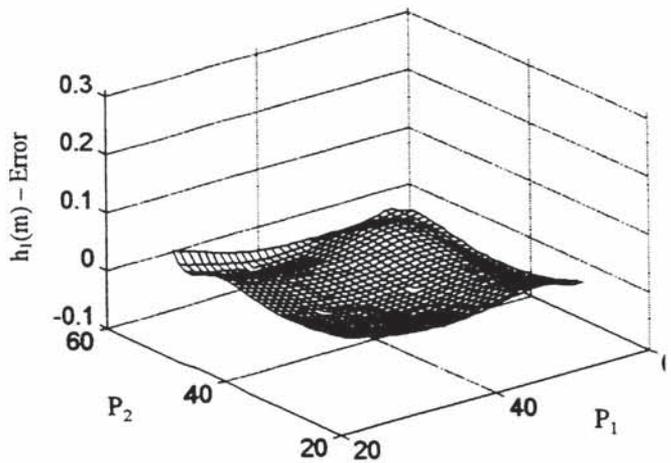


Fig. 10 Modelling Error in output of LM network with biased local models