

# Linear approximation model network and its formation via evolutionary computation

YUN LI<sup>1</sup> and KAY CHEN TAN<sup>2</sup>

<sup>1</sup>Centre for Systems & Control, and Department of Electronics & Electrical Engineering, University of Glasgow, Glasgow G12 8LT, UK

<sup>2</sup>Department of Electrical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260  
e-mail: Y.Li@elec.gla.ac.uk; eletankc@nus.edu.sg

**Abstract.** To overcome the deficiency of ‘local model network’ (LMN) techniques, an alternative ‘linear approximation model’ (LAM) network approach is proposed. Such a network models a nonlinear or practical system with multiple linear models fitted along operating trajectories, where individual models are simply networked through output or parameter interpolation. The linear models are valid for the entire operating trajectory and hence overcome the local validity of LMN models, which impose the predetermination of a scheduling variable that predicts characteristic changes of the nonlinear system. LAMs can be evolved from sampled step response data directly, eliminating the need for local linearisation upon a pre-model using derivatives of the nonlinear system. The structural difference between a LAM network and an LMN is that the overall model of the latter is a parameter-varying system and hence nonlinear, while the former remains linear time-invariant (LTI). Hence, existing LTI and transfer function theory applies to a LAM network, which is therefore easy to use for control system design. Validation results show that the proposed method offers a simple, transparent and accurate multivariable modelling technique for nonlinear systems.

**Keywords.** Modelling; system identification; linear approximation model networks; evolutionary computation; local model networks.

## 1. Introduction

A dynamic engineering system is usually nonlinear and complex in practice. The system dynamics may vary significantly with changes of operating conditions. Hence, the use of a single nominal linear model under one operating condition may be unreliable and inadequate to represent a practical system. The use of a nonlinear model may tackle the problem, but nonlinear modelling is more difficult and often requires simplifying assumptions to be made (Kemna & Mellichamp 1995; Leontaritis & Billings 1985; Vandemolengraft *et al* 1994). Even when a nonlinear model is tractably obtained and successfully applied to system simulations, it appears to be more difficult to use for control system design.

This has prompted research into modelling by a ‘local model network’ (LMN) of multiple locally linear models (Gawthrop 1995; Johansen & Foss 1992). An LMN is based on conventional linearisation using Taylor series expansions around multiple equilibrium or operating points. This requires a preliminary nonlinear model and is almost impossible to obtain from step response data, although such data are usually readily available in practice. Further, using locally perturbed linearisation loses tractability between two operating levels and each local model is valid only locally (Gray *et al* 1996; Tan 1997). The overall model is linear parameter-varying (LPV) and is hence nonlinear. Existing linear time-invariant (LTI) theory is thus hard to use, as it cannot be applied generally to LPV systems of order greater than one (Kailath 1980).

As an alternative to this, a novel ‘linear approximation model’ (LAM) network technique is proposed in this paper for modelling multivariable practical systems. Such a network consists of multiple linearising models obtainable directly from data by fitting nonlinear trajectories between two operating levels. The LAM network approach is detailed in the following section. Section 3 discusses its formation using the globally optimal evolutionary search technique. Some examples and validation results are shown in §4 and conclusions are drawn in §5.

## 2. Linear approximation model network

### 2.1 Related local model networks

Local model networks have recently been developed to approximate practical systems. An LMN consists of multiple locally linearised models that are locally valid (Gawthrop 1995; Johansen & Foss 1992). These local models are combined together by weighting through a basis function to yield an overall model that approximates the nonlinear system globally. Each local model is obtained by perturbing and linearising the system at a different equilibrium point on the operating trajectory.

In constructing an LMN for a generic nonlinear dynamical system,

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)), & x(0) &= x^0, \\ y(t) &= g(x(t), u(t)), \end{aligned} \quad (1)$$

consider linearisation around one of the equilibrium points,  $x(t) = x_i^e$  and  $u(t) = u_i^e$ . Denote the locally valid small variations of the state and input variables by  $\Delta x_i(t) = x_i(t) - x_i^e$  and  $\Delta u_i(t) = u_i(t) - u_i^e$  respectively. A local linear model is then obtained by a first-order Taylor series expansion (Franklin *et al* 1991):

$$\begin{aligned} \dot{x}_i(t) - f(x_i^e, u_i^e) &\approx \Delta \dot{x}_i(t) = A_i \Delta x_i(t) + B_i \Delta u_i(t), \\ y_i(t) - g(x_i^e, u_i^e) &\approx \Delta y_i(t) = C_i \Delta x_i(t) + D_i \Delta u_i(t), \end{aligned} \quad (2a)$$

or

$$\begin{aligned} \dot{x}_i(t) &\approx A_i x_i(t) + B_i u_i(t) + d_{xi}^e, \\ y_i(t) &\approx C_i x_i(t) + D_i u_i(t) + d_{yi}^e, \end{aligned} \quad (2b)$$

where

$$\begin{aligned} d_{xi}^e &= f(x_i^e, u_i^e) - A_i x_i^e - B_i u_i^e, \\ d_{yi}^e &= g(x_i^e, u_i^e) - C_i x_i^e - D_i u_i^e. \end{aligned} \quad (2c)$$

The  $(j, k)$  elements of the state-space matrices  $A_i, B_i, C_i, D_i$  are given by

$$\begin{aligned} A_{i,jk} &= \partial f_j / \partial x_k, \\ B_{i,jk} &= \partial f_j / \partial u_k, \\ C_{i,jk} &= \partial g_j / \partial x_k, \\ D_{i,jk} &= \partial g_j / \partial u_k. \end{aligned} \tag{3}$$

Here all partial derivatives are evaluated at the  $x(t) = x_i^e$  and  $u(t) = u_i^e$ . Clearly, when an LMN is built from a nonlinear model of the system, the LMN approach is only valid if  $f(x, u)$  and  $g(x, u)$  are differentiable.

By weighting all local models through a set of basis functions, an LMN results, which forms an overall model that globally approximates the nonlinear system (Gawthrop 1995; Gray *et al* 1996; Johansen & Foss 1992). The overall model network output is then given by

$$y(t) = \sum_i \rho_i(\varphi(t))y_i(t), \tag{4}$$

where  $\rho_i$  is a basis weighting function of some scheduling variable or variables,  $\varphi(t)$ , which may be the state, input or output variables.

It can be seen that this networking mechanism results in global interference whilst individual models are locally valid. Because of the local validity, the scheduling variables need to be pre-selected before constructing an LMN such that they can predict changes of the nonlinear system dynamics (Gray *et al* 1996; Tan 1997). Since the overall resultant network is LPV, existing LTI theory is difficult to apply (Kailath 1980).

## 2.2 Linear approximation model

In control engineering practice, there has been an increasing effort to use step response data quickly to establish a simple linear model that approximates the system under study with high quality (Tan & Li 1997; Wang & Cluett 1994). Such data are often readily available after a step command has been applied to set an operation level. Different from conventional linearisation, which is valid only in the vicinity of the operating point, identifying an approximate model needs to take into account the nonlinearity excited along the entire operating trajectory. In this paper, a simple modelling technique that meets these requirements is developed, termed the 'linear approximation model networks'.

Denote the state and output variables of the model approximating along the operating trajectory by  $\hat{x}_i(t)$  and  $\hat{y}_i(t)$ , respectively. To obtain a LAM means mathematically to approximate (1) by

$$\begin{aligned} \dot{\hat{x}}_i(t) &= A_i \hat{x}_i(t) + B_i u(t); \quad x(0) = x_i^e \text{ or } x(0) = x^0, \\ y(t) &\approx \hat{y}_i(t) = C_i \hat{x}_i(t) + D_i u(t). \end{aligned} \tag{5}$$

This gives a straightforward approximation in the entire trajectory ranging from the initial condition to the set-point, whilst an LMN of (2) is applicable only around the initial condition. Note that, alternative to (5), a transfer function or transfer function matrix may be used to represent a LAM (Tan 1997).

To obtain (5), fitting errors, as opposed to system derivatives, are needed. This eases the difficulties encountered in conventional linearisation and implies that a linear model can be obtained directly from physically sampled data without the need for an initial nonlinear

model. To achieve high modelling quality, an evolutionary computation method will be used to search for globally optimal solutions, which is described in § 3. For this, the cost function in fitting is not limited to total quadratic errors, and any linear metric such as the  $L_1, L_2$  and  $L_\infty$  norms may be used. The viability of this modelling technique will be demonstrated in § 4, together with the validation.

### 2.3 Linear approximation model network

What mainly makes a LAM more useful is its ability in globally modelling a practical system by forming a network with other LAMs obtained at different operating points. Globally, a network should offer a better model for a nonlinear system than a single LAM could. There can be two ways of adding a new LAM to form a network:

- (i) To use a relative step response that brings one set-point (as a new initial condition,  $x_i^e$ ) to another; or
- (ii) To use an absolute step response that always brings the original at-rest point (as a uniform initial condition,  $x_0$ ) to the other set-point.

The concept of the first method is somewhat similar to the LMN technique in adding models with varying initial conditions. In the second method, individual LAMs can offer a wider applicability and there is no need to switch between initial conditions when different LAMs are activated. Either method is straightforward to use and may be selected according to the available data sets. Compared with an LMN, a LAM network offers the advantage of obtaining each linear model directly from step response and hence also the advantage of a wide range of validity of individual models that can act stand-alone in a certain degree.

In modelling the system at an unseen operating level between two known ones, activating and combining the effects of the two, and only two, bounding LAMs should suffice, because of the relatively wide range of validity of individual models. A global (such as sigmoid and Gaussian) type of basis weighting function that combines more than two models as used in LMNs is unnecessary here. Instead, the simplest, linear local interpolation between the two neighbouring models as illustrated in figure 1 can be used as the weighting schedule. This scheme will be validated in § 4 with an example.

Similar to (4), take the output  $y(t)$  as an example for the network to amalgamate. Then for any operating level  $U$  between the two known ones  $U_1$  and  $U_2$ , the output can be simulated using the simple interpolation:

$$\hat{y}(t) = W_1 \hat{y}_1(t) + W_2 \hat{y}_2(t) = \frac{U_2 - U}{U_2 - U_1} \hat{y}_1(t) + \frac{U - U_1}{U_2 - U_1} \hat{y}_2(t), \quad (6)$$

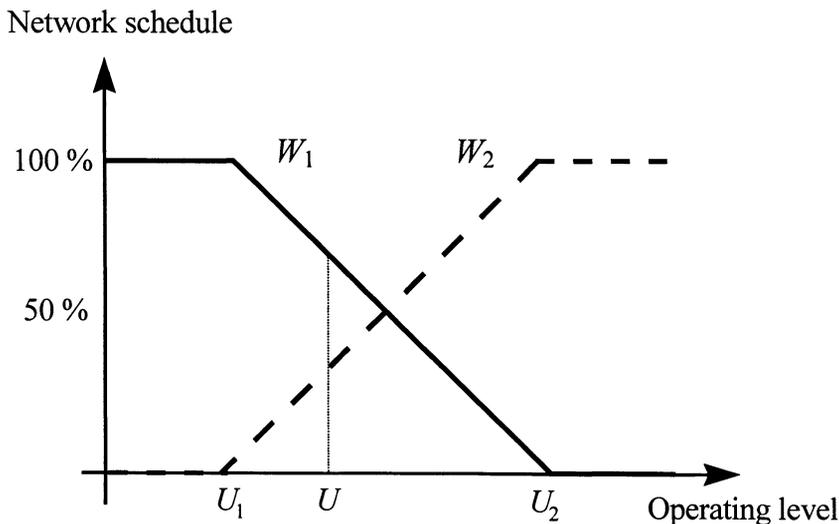
where  $\hat{y}_i(t)$  are obtained using (5).

Different from an LMN of (4) that usually activates more than two local models, a LAM network is not scheduled by a system variable and hence the amalgamated model remains linear. The order of the overall model here is the sum of two LAMs only, i.e.,

$$\text{Order}(M) = 2 \times \text{Order}(M_i), \quad (7)$$

where  $M$  denotes the overall network model and  $M_i$  a bounding LAM.

Alternative to the output, coefficient matrices in (5) can also be used as the model attributes for amalgamation because of the linearity. This means that the overall model



**Figure 1.** Static linear local interpolation suffices for joining up a LAM network.

is now

$$\begin{aligned} \hat{x}(t) &= \bar{A}\hat{x}(t) + \bar{B}u(t), \\ \hat{y}(t) &= \bar{C}\hat{x}(t) + \bar{D}u(t), \end{aligned} \tag{8}$$

which will be of the same order as one LAM. Here  $\bar{(\cdot)}$  denotes the operation of a weighted average similar to that in (6). For example,

$$\bar{A} = \frac{U_2 - U}{U_2 - U_1}A_1 + \frac{U - U_1}{U_2 - U_1}A_2. \tag{9}$$

It can be seen that with the use of more widely valid linear models, no time-dependent predictive scheduling variables are necessary. Hence, existing stability and other theories apply. This is in contrast to LMNs, whose stability must be tested by constructing a Lyapunov function (Gawthrop 1995; Johansen & Foss 1992).

### 3. LAM and network formation via evolutionary search

The merit of a model and the modelling accuracy is usually measured by an error metric  $J : \mathbf{R}^n \rightarrow \mathbf{R}^+$ ,

$$J(M) = \|w(t)e(t)\| = \|w(t)[y(t) - \hat{y}(t)]\|, \tag{10}$$

where  $n$  is the number of parameters that need to be determined in the model and  $w(t)$  is a time weighting function mainly used in time-series or adaptive modelling to ‘forget’ older data. Traditionally, numerical differentiation and conventional gradient-guidance based methods, such as the least mean squares method, are used to minimise a quadratic  $J$  for linearisation and obtaining an LMN (Beck & Arnold 1977; Johansen & Foss 1992). Such conventional methods are not recommended in the formation of a practical LAM network because:

- (i) they are only applicable when the error surface is smooth, which is rarely true in practice;
- (ii) the cost function can hardly be flexible, other than the well-behaved squared error;
- (iii) they are local optimisation methods; and
- (iv) they cannot be used with step response data, as the steady-state may lead the identification to ‘sleep’ and to invert a non-full rank matrix of many identical data.

These problems can, however, be overcome easily by evolutionary computation methods based on a *a posteriori* trial-and-error and on the Darwinian–Wallace principle of ‘survival-of-the-fittest’. Without the need for a differentiable or well-behaved performance index, an evolutionary algorithm (EA) can ‘intelligently’ explore a noisy and poorly understood space at multiple points by a population of candidate solutions leading to several globally optimised models. Often, for the convenience of argument in evolutionary computation, the ‘error’ surface  $J$  is replaced by a ‘fitness landscape’  $f : \mathbf{R}^n \rightarrow (0, 1]$ ,

$$f(M) = \frac{1}{1 + J(M)} = \frac{1}{1 + \|w(t)[y(t) - \hat{y}(t)]\|}. \tag{11}$$

An evolutionary search and optimisation process can be depicted as in figure 2. Before this process begins, an initial ‘generation’ of multiple coded ‘chromosomes’ representing candidate models is formed using existing candidates or random ones. All candidates are then evaluated and, similar to the roulette wheel example, fitter ones are selected with a high probability to reproduce repeatedly. To evolve, some search co-ordinates and information are exchanged via ‘crossover’ and are altered via ‘mutation’. This newly formed generation of candidate solutions is again subject to selection and reproduction. The evolutionary cycle then repeats, until no meaningful improvements are achieved. By slightly trading off precision in a nondeterministic manner, an EA improves tractability and robustness in global optimisation. For details on how an EA works see, for example, Goldberg (1989) and Michalewicz (1994).

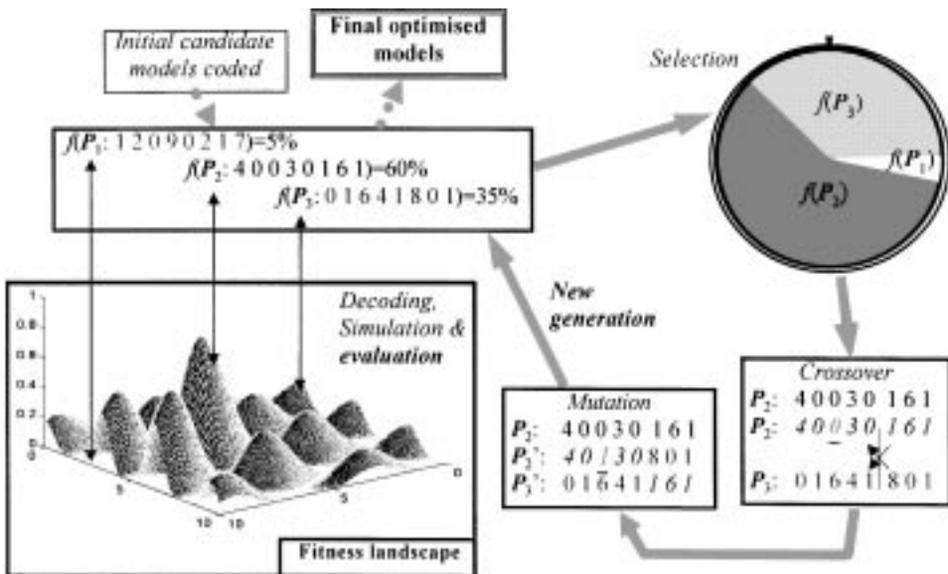


Figure 2. Global search for an optimal LAM via evolutionary computation.

In this paper, a parameter in (5) is coded using decimals with range encoding so as to search for an appropriate range of the parameter (Tan 1997). The tournament selection mechanism is adopted, as this is the most efficient (Michalewicz 1994). To overcome the well-known weakness of EAs in finding an exact optimum, fitter individuals in every generation are gradually fine-tuned through an *a posteriori* hill-climbing learning process (Tan 1997). This hybrid approach leads to a finer local exploration, a higher accuracy and a faster convergence. Such a learning enhanced EA has been applied to several system modelling and identification problems successfully (Tan *et al* 1996; Li *et al* 1997; Tan & Li 1997).

#### 4. Nonlinear coupled modelling and linearisation problems

A coupled twin-tank nonlinear system that models liquid-level regulation widely found in chemical and dairy plants is studied in this paper. Based on Bernoulli's mass-balance and flow equations, the system structure is described by

$$\begin{aligned} \begin{bmatrix} \dot{h}_1 \\ \dot{h}_2 \end{bmatrix} &= \begin{bmatrix} -\text{sgn}(h_1 - h_2) \frac{c_1 a_1}{a} (2g|h_1 - h_2|)^{1/2} \\ \text{sgn}(h_1 - h_2) \frac{c_1 a_1}{a} (2g|h_1 - h_2|)^{1/2} - \frac{c_2 a_2}{a} (2g(h_2 - H_0))^{1/2} \end{bmatrix} \\ &+ \begin{bmatrix} \frac{Q_1}{a} & 0 \\ 0 & \frac{Q_2}{a} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \end{aligned} \quad (12)$$

where the system outputs are liquid levels of tank 1,  $y_1 = h_1$ , and tank 2,  $y_2 = h_2$ , under constraint  $h_1, h_2 \geq H_0 = 0.03$  m, an equivalent height of the coupling and discharging pipes. An equivalent orifice area of the coupling pipe is  $a_1 = 0.0000396$  m<sup>2</sup> and that of the discharging pipe is  $a_2 = 0.0000386$  m<sup>2</sup>, whose equivalent discharge constants are  $c_1 = 0.53$  and  $c_2 = 0.63$ , respectively. The cross-sectional areas of both tanks are  $a = 0.01$  m<sup>2</sup> and the per-volt pump rates are  $Q_1 = Q_2 = 0.000007$  m<sup>3</sup> s<sup>-1</sup> V<sup>-1</sup>. Here,  $g = 9.81$  m s<sup>-2</sup> is the gravitational constant.

Without loss of generality, a 2-norm based fitness function

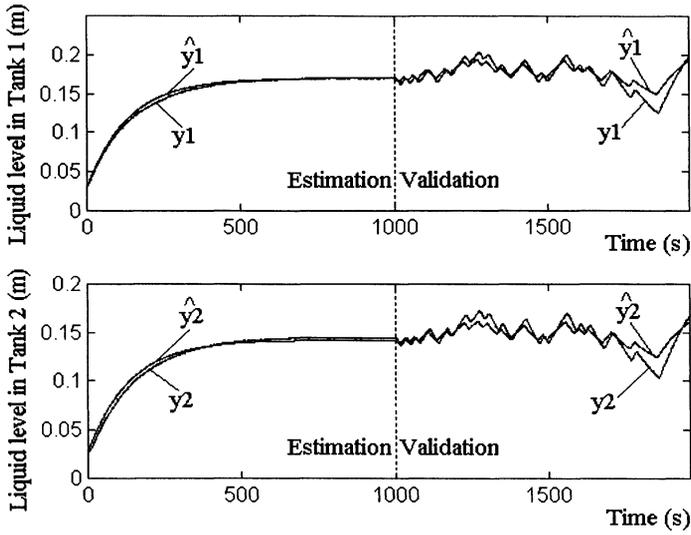
$$f(M) = \frac{1}{1 + \|y_1 - \hat{y}_1\|_2 + \|y_2 - \hat{y}_2\|_2}, \quad (13)$$

is used here to index the modelling performance. However, the 2-norm is adopted but may be replaced by any typical quadratic error-based metric, such as the widely used mean square error.

##### 4.1 Evolving a LAM from step-response data

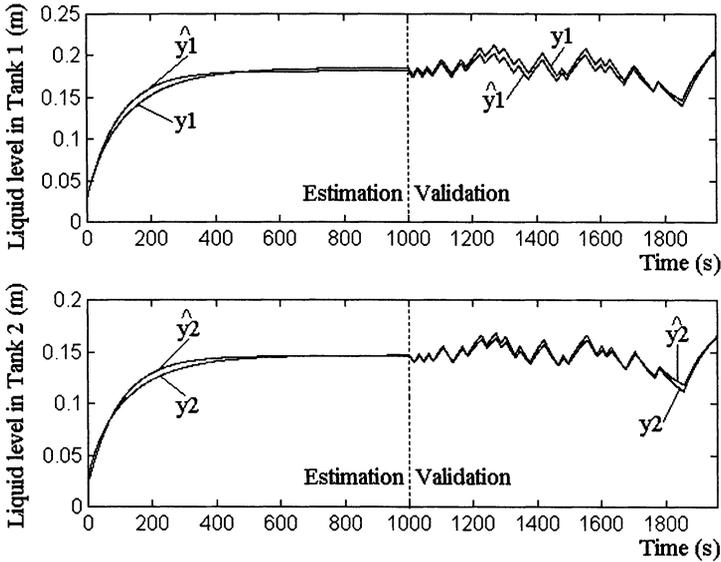
**4.1a Evolving a LAM from physically sampled data:** To evolve a LAM directly from the physical system, step responses to a 2.5 V input applied to both tanks have been sampled at one-second intervals for 1000 s. The data are used to obtain the coefficient matrices of (5), minimising (13) under the initial condition  $h_1(0) = h_2(0) = H_0 = 0.03$  m. The coefficients obtained using the hybrid EA are:

$$A = \begin{bmatrix} -0.0683 & 0.0738 \\ 0.0708 & -0.0931 \end{bmatrix}, \quad B = \begin{bmatrix} 0.63 & 0 \\ 0 & 0.63 \end{bmatrix}.$$



**Figure 3.** Data based LAM ( $\hat{y}_1, \hat{y}_2$ ) validated against seen and unseen sampled data ( $y_1, y_2$ ).

The original data and modelled results are compared in figure 3 (the first 1000 s). To further validate the LAM against unseen data and test its performance for linearisation at the set operating point, the steady state has been perturbed by small PRBS signals (the last 1000 s). It can be seen that the LAM accurately fits the nonlinear physical system in the entire operating envelope and also for local linearisation.



**Figure 4.** Model based LAM ( $\hat{y}_1, \hat{y}_2$ ) validated against the original nonlinear model ( $y_1, y_2$ ).

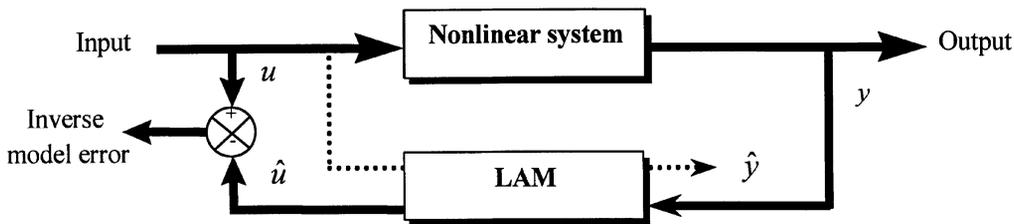


Figure 5. Inverse model validation and construction of a LAM.

4.1b *Evolving a LAM from a nonlinear model:* Applying the hybrid EA method to step (or any other types of) responses using the nonlinear model of (12) at a 2.5 V operating level under the same initial condition, a set of LAM coefficient matrices are obtained as

$$A = \begin{bmatrix} -0.0751 & 0.085 \\ 0.0914 & -0.1246 \end{bmatrix}, \quad B = \begin{bmatrix} 0.77 & 0 \\ 0 & 0.77 \end{bmatrix}.$$

Applying the same PRBS signals to the LAM and nonlinear models, the validation results are shown in figure 4. It can be seen that the LAM fits the nonlinear model accurately.

#### 4.2 Inverse model validation and construction of a LAM

4.2a *Inverse model validation of a LAM:* To guard against any possible singular behaviour of a LAM as a result of reducing a nonlinear system to a linear one, the inverse model simulation and validation technique (Thomson & Bradley 1988) can be applied here. This approach is shown in figure 5, where the output data are used to inverse-excite a candidate LAM. Then the inverse model error is used as a validation index.

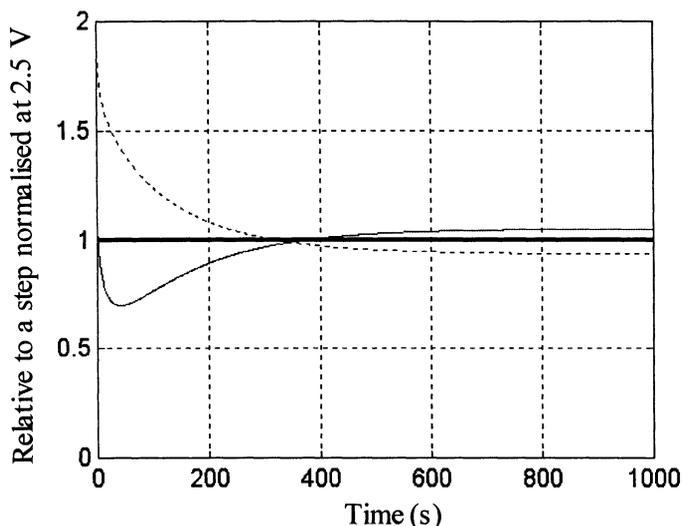
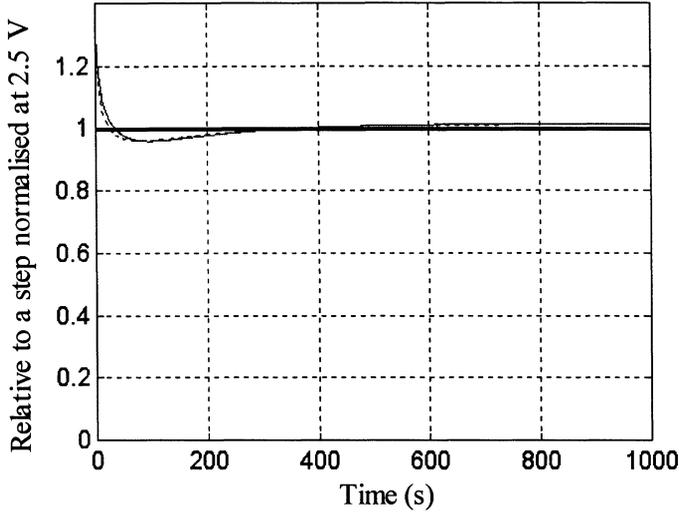


Figure 6. Input comparisons in inverse model validation:  $\hat{v}_1(t)$  {—},  $\hat{v}_2(t)$  {--},  $u(t)$  {—·—}.



**Figure 7.** Input responses in inverse-fitting a LAM:  $\hat{v}_1$ , {—},  $\hat{v}_2(t)$  {— · —},  $u(t)$  {—}.

This approach has been used to validate the LAM that is obtained from the nonlinear model. Figure 6 shows the validation results with respect to  $v_1$  and  $v_2$  normalised at 2.5 V, revealing modelling errors from an additional angle.

*4.2b Constructing a LAM via inverse modelling:* Here, the inverse model simulation technique is applied to the construction of a LAM. The performance of candidate models are also measured by (13), but with the outputs replaced by the inputs. For the same operating level at 2.5 V, optimal coefficient matrices are evolved using inverse simulation of the nonlinear model:

$$A = \begin{bmatrix} -0.01 & 0.0022 \\ 0.015 & -0.03 \end{bmatrix}, \quad B = \begin{bmatrix} 0.85 & 0 \\ 0 & 0.85 \end{bmatrix}.$$

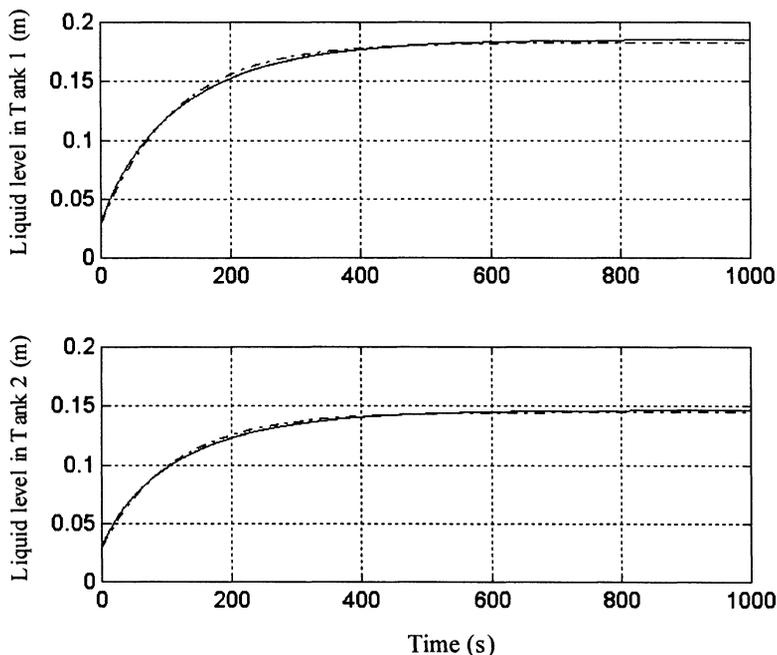
The resulting input responses are shown in figure 7. This confirms that an accurate LAM may also be obtained conveniently by inverse fitting the nonlinear system.

To validate against any possible singular behaviour of an inverse-fit LAM, forward simulations can be applied. Here, the resultant outputs are shown in figure 8, suggesting commendable modelling performance.

### 4.3 Evolving a LAM network

Without loss of generality, a simple two-LAM network is constructed here to approximate this nonlinear system. Applying the hybrid EA method to the nonlinear system of (12) at a 2 V operating level under the same initial condition leads to a set of LAM coefficient matrices:

$$A = \begin{bmatrix} -0.065 & 0.069 \\ 0.028 & -0.047 \end{bmatrix}, \quad B = \begin{bmatrix} 81.58 & 0 \\ 0 & 81.58 \end{bmatrix}.$$



**Figure 8.** Outputs of the inverse-fit LAM {—} and the original nonlinear model {- - -}.

In forming the network, another LAM at a 3 V operating level is also evolved, yielding coefficient matrices:

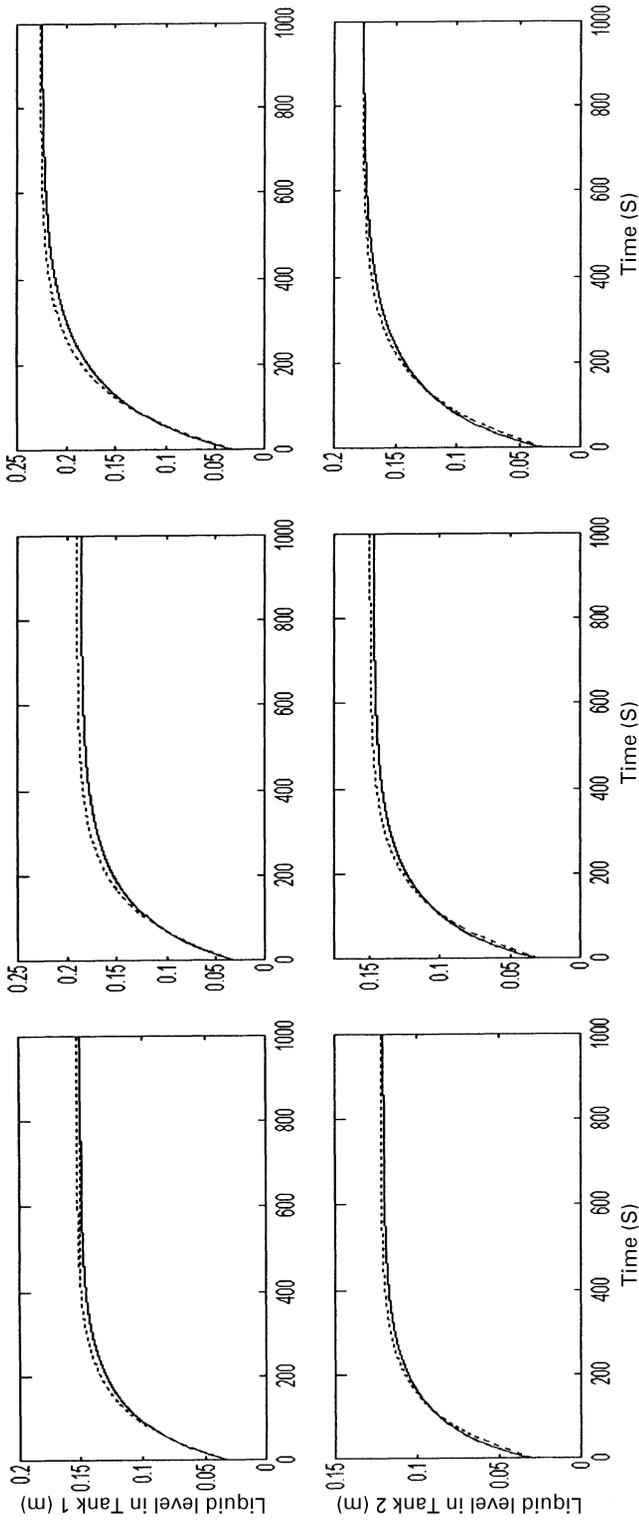
$$A = \begin{bmatrix} -0.013 & 0.008 \\ 0.049 & -0.072 \end{bmatrix}, \quad B = \begin{bmatrix} 82.18 & 0 \\ 0 & 82.18 \end{bmatrix}.$$

Based on these LAMs and (6), a network is formed and can be used to model the ‘unseen’ operating levels at, for example, 2.2, 2.5 and 2.8 V. Figure 9 shows the outputs of the network and those of the original nonlinear model. It can be seen that the LAM network offers a close approximation of the nonlinear system for a wide operating range. The interpolated outputs at 2.5 V excitation offer comparable results of the LAM evolved directly from the nonlinear system at this level (figure 4).

If interpolation is applied to the parameters, and not the outputs, of the LAMs using (9), the network results in a slightly inferior fitting. This is because the ‘network’ is now reduced to a ‘second-order’ linear model, as opposed to the ‘fourth-order’ one inherent in (6). This form of networking is therefore not always recommended.

## 5. Conclusions

Based on the globally optimal evolutionary search technique, a ‘linear approximation model network’ approach has been developed for nonlinear or practical system modelling. Such a network models a nonlinear system with multiple linearising models fitted along the operating trajectory, where individual models are simply networked through output or parameter interpolation. Validation results show that the proposed method offers a simple, transparent and accurate modelling technique for multivariable nonlinear systems.



**Figure 9.** Validation of the LAM network that models at the unseen 2.2, 2.5 and 2.8 V levels.

Advantages of this technique over the existing LMN technique include the following.

- (i) Models can be obtained directly from sampled response data of a physical system, eliminating the need for an intermediate nonlinear model or its derivatives.
- (ii) Evolving the models needs no derivative error information, which cannot be obtained reliably with noisy output data usually encountered in a practical environment, and permits flexible and multiple performance indices to be used.
- (iii) Models can be tractably evolved from step excitations and are representative to the entire operating trajectory.
- (iv) The models are easy to use without needing to pre-determine time-varying predictive scheduling variables or to superimpose upon operating points as required by LMNs.
- (v) The overall model is linear and of an order no higher than the sum of two individual models and hence does not require construction of Lyapunov functions to test its stability.
- (vi) The overall linearity can allow existing LTI theory to apply and hence can ease the task of control system design.
- (vii) The approach offers a flexibility of forming either a relatively low order model via parametric interpolation or a higher order model via output interpolation.

Future work involves the investigation of on-line model adaptation, owing to the network simplicity, and the development of a local controller network corresponding to a LAM network. Research results on these issues will be reported in due course.

## References

- Beck J V, Arnold K J 1977 *Parameter estimation in engineering and science* (New York: John Wiley & Sons)
- Franklin G F, Powell J D, Emami-Naeini A 1991 *Feedback control of dynamic systems* (Reading, MA: Addison-Wesley)
- Gawthrop P J 1995 Continuous-time local state local model networks. *Proc. 1995 IEEE Conf. Systems, Man and Cybernetics* (Piscataway, NJ: IEEE) pp 852–857
- Goldberg D E 1989 *Genetic algorithms in search, optimisation and machine learning* (Reading, MA: Addison-Wesley)
- Gray G J, Murray-Smith D J, Li Y, Sharman K C 1996 Nonlinear system modelling using output error estimation of a local model network. *Proc. 1996 Summer Computer Simulation Conference* (San Diego, CA: Soc. Comput. Simulation Int.) pp 460–465
- Johansen T A, Foss B A 1992 A NARMAX model representation for adaptive control based on local models. *Modelling, Identification and Control* 13: 25–39
- Kailath T 1980 *Linear systems* (New York: Prentice-Hall)
- Kemna H, Mellichamp D A 1995 Identification of combined physical and empirical-models using nonlinear *a-priori* knowledge. *Contr. Eng. Pract.* 3: 375–382
- Leontaritis I J, Billings S A 1985 Input-output parametric models for non-linear systems. Part I: Deterministic non-linear systems. *Int. J. Control* 41: 303–328
- Li Y, Tan K C, Gong M R 1997 Global structure evolution and local parameter learning for control system model reductions. *Evolutionary algorithms in engineering applications* (eds) D Dasgupta, Z Michalewicz (New York: Springer Verlag) pp 345–360
- Michalewicz Z 1994 *Genetic algorithms + data structure = evolutionary programs* 2nd edn (Berlin: Springer-Verlag)
- Tan K C 1997 *Evolutionary methods for modelling and control of linear and nonlinear systems*. PhD thesis, Dept. of Electron. and Electr. Eng., University of Glasgow, Glasgow

- Tan K C, Y Li 1997 Evolutionary system identification in the time-domain. *Proc. Inst. Mech. Eng.-J. Syst. Contr. Eng.* 211: 319–323
- Tan K C, Gong M R, Li Y 1996 Evolutionary linearisation in the frequency domain. *Electron. Lett.* 32: 74–76
- Thomson D G, Bradley R 1988 Validation of helicopter mathematical models by comparison of data from Nap-of-the-Earth flight tests and inverse simulation. *Proc. 14th European Rotorcraft Forum*, Milan, paper No. 87
- Vandemolengraaf M J G, Veldpaus F E, Kok J J 1994 An optimal estimation method for nonlinear mechanical systems. *J. Dyn. Syst. Meas. Contr., Trans. ASME* 116: 805–810
- Wang L, Cluett W R 1994 System identification based on closed-loop step response data. *IEE Proc. – Control Theory Appl.* 141: 107–110