

Estimation of model error for nonlinear system identification

V. Parameswaran
J.R. Rao

Indexing terms: Algorithms, Model error, Nonlinear system identification

Abstract: Algorithms are presented for estimation of deterministic model error in the assumed models of nonlinear discrete and continuous time systems. The explicit model error time histories are parameterised using least squares method. The parameterised models relative to the true model explain the deterministic deficiency in the chosen models, in the sense of minimum model error. The algorithms have appealing features of extended Kalman filter. The numerical simulation results are obtained by implementing the algorithms in PC MATLAB.

1 Introduction

System identification and parameter estimation technology have reached a certain level of maturity, and there are many proven tools which can be routinely applied to model dynamic systems [1, 2]. The problem of determination of suitable structure and order of a system from its input/output data has several workable solutions, especially for linear systems [3]. Structure determination for nonlinear systems is relatively more complex. There are many real life situations for which accurate identification of nonlinear terms in the model of a dynamic system is required. One such example is related to the identification of aerodynamic model, based on flight data from high angle of attack (HAOA) manoeuvres [4].

In the present work, an approach to determine the discrepancy in the postulated model, by the so-called minimum model error philosophy is used [5]. The method of invariant embedding is used to solve the resultant two-point boundary value (TPBV) problem, and to obtain very efficient and recursive algorithms to estimate the model error. The method of invariant embedding [1, 6] provides sequential estimates of the states of the nonlinear system. The estimates of deterministic model error, based on the measured data, are also obtained recursively in the present formulation. The time histories of these model discrepancy are then fitted using the least squares method to determine the coefficients of the model error description. Similarity and contradistinction of the estimators obtained by the present approach with that of extended Kalman filter are highlighted.

© IEE, 1994

Paper 1500D (C8), first received 22nd February and in revised form 27th July 1994

The authors are with the Flight Mechanic, and Control Division, National Aerospace Laboratories, Bangalore-560017, India

2 Model error estimation algorithms

2.1 Discrete-time algorithm

Let the true nonlinear system be described as

$$X(j+1) = g(X(j), j) \quad (1)$$

$$Y(j) = h(X(j), j) \quad (2)$$

Here g is the true representation of the dynamic system with its states as X . Y is the vector of observables obtained at discrete time intervals of $t_0 < t_j < t_N$. The state and measurement vectors have appropriate dimensions. Eqns. 1 and 2 are recast to explicitly express the deterministic model error as follows [5]:

$$x(j+1) = f(x(j), j) + d(j) \quad (3)$$

$$y(j) = h(x(j), j) + v(j) \quad (4)$$

Here f denotes the nominal model, the vector $v(j)$ denotes the random noise sequence with zero mean and covariance matrix Q_1^{-1} , and vector d is the model error which is to be estimated in the sense of minimum model error criterion:

$$J = \sum_{j=0}^N \{ [y(j) - h(x(j), j)]' Q_1 [y(j) - h(x(j), j)] + d'(j) Q_2 d(j) \} \quad (5)$$

Minimisation of J obtains the condition of x (the state of the model chosen to fit the data) tending very close to X (the state of the true model or system). Thus the model discrepancy can be taken as the estimation of model error, the time histories of which can then be appropriately fitted in the sense of least squares to parameterise the model discrepancy. The minimisation of J , by applying the Euler-Lagrange conditions [1, 6] leads to the following TPBV (Appendix 6) problem:

$$x(j+1) = f(x(j), j) + 1/2 Q_2^{-1} \lambda(j) \quad (6)$$

$$\lambda(j-1) = (\partial f(x(j), j) / \partial x)' \lambda(j) + 2(\partial h(x(j), j) / \partial x)' Q_1 [y(j) - h(x(j), j)] \quad (7)$$

From eqns. 3 and 6 we get the expression for model discrepancy

$$d(j) = 1/2 Q_2^{-1} \lambda(j) \quad (8)$$

The authors are grateful to Dr. S. Srinathkumar, Head, FMCD, NAL for his encouragement during the course of this work and to the reviewers for their very useful comments and suggestions.

The costate eqn. 7 is required to be solved to get the model correction terms 'd'. The technique of invariant embedding is applied to solve the above TPBV problem [6]. The resultant set of estimator equations is given as:

$$\hat{x}(j+1) = f(\hat{x}(j), j) + 2S(j+1)H(j+1) \times Q_1[y(j+1) - h(\hat{x}(j+1), j+1)] \quad (9)$$

$$S(j+1) = [I + 2P(j+1)H(j+1)Q_1H(j+1)] \times P(j+1) \quad (10)$$

$$P(j+1) = f_{\hat{x}}(\hat{x}(j), j)S(j)f_{\hat{x}}(\hat{x}(j), j)' + 1/2Q_2^{-1} \quad (11)$$

with $H(j) = \partial h(x(j), j)/\partial x(j)$ at $x(j) = \hat{x}(j)$

$$\hat{d}(j) = 2S(j)H(j)'Q_1[y(j) - h(\hat{x}(j), j)] \quad (12)$$

Eqns. 9-12 describe the invariant embedding based model error estimation (IEBMEE) algorithm in discrete time domain. Tuning of the IEBMEE algorithm is done by using appropriate values of matrices Q, and Q_2 [5, 6]. The algorithm presented above has structure similar to extended Kalman filtering algorithm. Since the system is deterministic, we interpret the inverse of matrix S (and P) as 'information providing matrix (IPM)'. Thus $\{P(j+1)\}^{-1}$ is a priori IPM and $\{S(j+1)\}^{-1}$ is the a posteriori IPM, the latter taking into account the effect of measurements in improving the estimates. The weighting matrix Q_1 affects the propagation of S which, in turn, affects the estimator gain matrix $2SH'Q_1$. Thus both of the weighting matrices have significant roles to play in the convergence of the estimates to the true states. The matrix Q, governs the propagation of effect of errors in prediction of measurements, and matrix Q_2 governs the propagation of effect of errors in deterministic part of the model. This aspect has similarity with Kalman filtering algorithm: Q_2^{-1} plays the role of covariance matrix of stochastic process noise. Thus the present algorithm based on the method of invariant embedding has some intuitively appealing features of extended Kalman filter and in contradistinction to it, obtains the estimation of deterministic model error in a sequential manner.

2.2 Continuous-time algorithm

The true nonlinear system is described as follows:

$$\dot{X}(t) = g(X(t), t) \quad (13)$$

$$Y(t) = HX(t) \quad (14)$$

Here g is the true representation of the dynamic system. The observables Y are obtained for interval $t_0 < t < T$. Following the development of the discrete-time case we get:

$$\dot{x}(t) = f(x(t), t) + d(t) \quad (15)$$

$$y(t) = H(t)x(t) + v(t) \quad (16)$$

Here f denotes the nominal model, v is additive measurement noise and the vector 'd' is the model discrepancy. The cost function is

$$J = \int_0^T \{ [y(t) - H(t)x(t)]' Q_1(t) [y(t) - H(t)x(t)] + d'(t) Q_2 d(t) \} dt \quad (17)$$

Minimisation of J [6] results in the TPBV problem:

$$\dot{x}(t) = f(x(t), t) + 1/2Q_2^{-1}(t)\lambda(t) \quad (18)$$

$$\dot{\lambda}(t) = (-\hat{c}f(x(t), t)/\hat{c}x)' \lambda(t) - 2H(t)'Q_1(t)[y(t) - H(t)x(t)] \quad (19)$$

$$d(t) = 1/2Q_2^{-1}(t)\lambda(t) \quad (20)$$

Application of invariant embedding to the TPBV problem results in

$$\dot{\hat{x}}(t) = f(\hat{x}(t), t) + 2S(t)H(t)'Q_1(t)[y(t) - H(t)\hat{x}(t)] \quad (21)$$

$$\dot{S}(t) = S(t)f_{\hat{x}}(\hat{x}(t), t)' + f_{\hat{x}}(\hat{x}(t), t)S(t) - 2S(t)H(t)'Q_1(t)H(t)S(t) + 1/2Q_2^{-1}(t) \quad (22)$$

$$d(t) = 2S(t)H(t)'Q_1(t)[y(t) - H(t)\hat{x}(t)] \quad (23)$$

Eqns. 21-23 describe the IEBMEE algorithm in continuous time. The inverse of matrix S(t) can be interpreted as 'information providing matrix (IPM)'. Similar observations as made for discrete time case also apply here with appropriate interpretations. To solve the nonlinear matrix differential equation for S, the transformation $u = Sr$, in eqn. 22 is used to obtain a set of linear differential equations [7]:

$$\dot{S}z = SF'z + FSz - 2SH'Q_1HSz + 1/2Q_2^{-1}z$$

or

$$Sr + 2SH'Q_1HSz - SF'z = Fu + 1/2Q_2^{-1}z \quad (24)$$

Here, $F = \partial f/\partial x$; and since $u = Sz$, we get $\dot{u} = \dot{S}z + S\dot{z}$ and $S\dot{z} = \dot{u} - \dot{S}z$. Using $S\dot{z}$ in eqn. 24 and defining \dot{z} as in eqn. 25 we get:

$$\dot{z} = -F'z + 2H'Q_1Hu \quad (25)$$

$$\dot{u} = 1/2Q_2^{-1}z + Fu \quad (26)$$

Eqns. 25 and 26 are solved by using transition matrix method.

3 Numerical Simulation results

The simulation and estimation algorithms have been implemented using available and newly formulated (.m) functions in PC MATLAB. In the MATLAB implementation of the above equations the corresponding partial differentiation of the functions f and h is carried out by using finite difference method. This approach renders the computation of various quantities very straightforward for the above sequential equations. Thus any change of nonlinear model can be easily incorporated while iteratively scanning various nonlinear functions. The latter aspect offers great deal of flexibility in trying out various nonlinear functions/models for real data analysis.

3.1 Discrete-time nonlinear system

The true nonlinear model used has the following form:

$$X_1(j+1) = 0.5X_1(j) + 0.623X_2(j) - 0.05X_1^3(j) + 2.5 \cos(0.3j) \quad (27)$$

$$X_2(j+1) = 0.1X_2(j) + 0.1 \cos(0.3j) \quad (28)$$

The model discrepancy is affected by eliminating one or more terms from eqn. 27. The deficient model is then used in the IEBMEE algorithm as f' . In the first exercise of estimating the model error by the new algorithm, the cubic nonlinear term is ignored from the true model. The time history match of the true and estimated states (x_1, x_2) is shown in Fig. 1. The comparison of the true and estimated model error time histories display excellent match. The corresponding model error time history is fitted with a general model of the form:

$$d(j) = a_1X_1(j) + a_2X_2(j) + a_3X_1^3(j) + a_4X_1^2(j) \quad (29)$$

It must be noted that square term is not present in the true model but it is intended to be estimated to see the performance of the algorithm. The estimates of the coefficients

conditions are far away from the true (unknown) conditions, more initial data points would be required to be eliminated from the least squares process. This is not a limitation of the method.

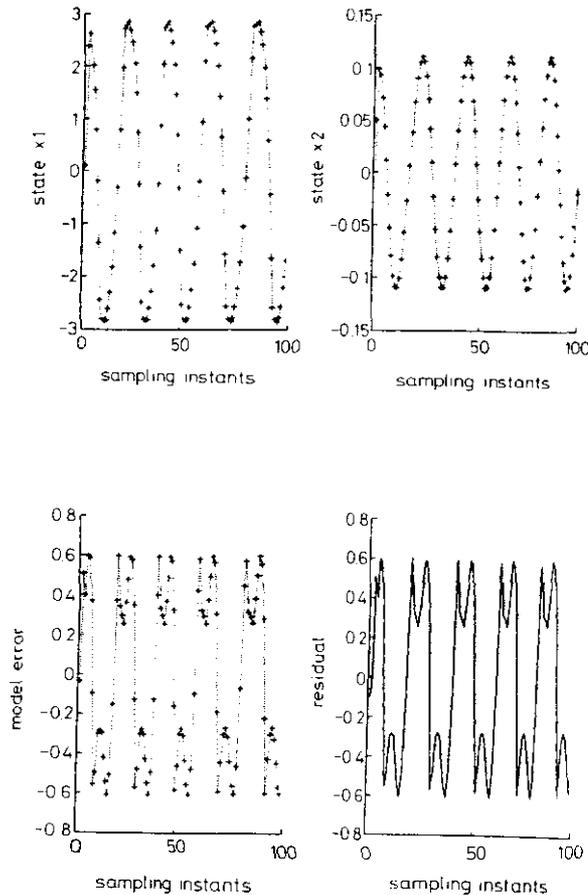


Fig. 1 Time histories of state, model error and residuals (X_1, X_2^3 terms deleted-DT case II)
 --- true
 + est.

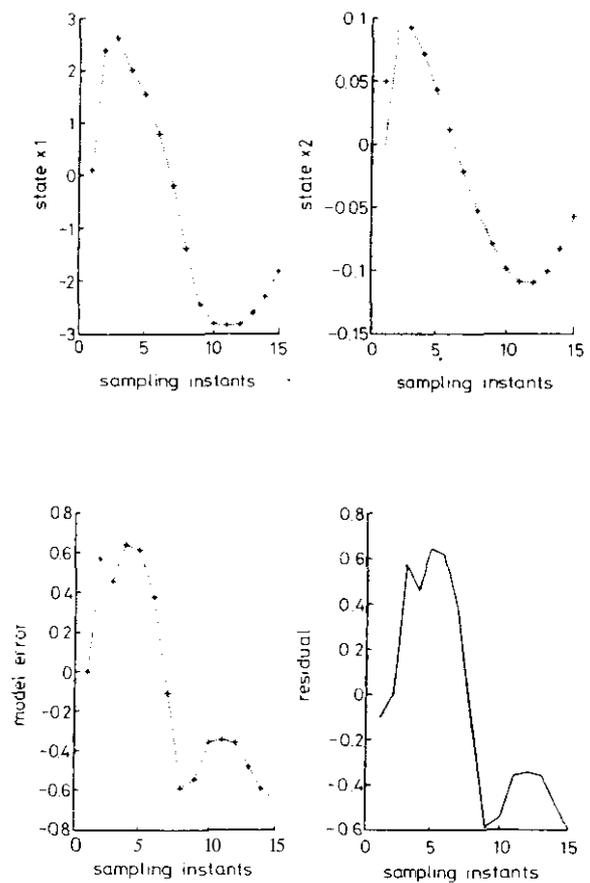


Fig. 2 Time histories of state, model error and residuals (X_1, X_2, X_1^3 terms deleted-DT case IV)
 --- true
 + est.

icients of eqn. 29 are obtained by least squares method and are shown in Table 1. The number of data points used for state and parameter estimation are about 100 for

Table 1: Nonlinear identification results: discrete-time

Parameter	a_1 (X_1)	a_2 (X_2)	a_3 (X_1^3)	a_4 (X_2^3)	Terms removed
TV	(0.5)	(0.623)	(-0.05)	(0)	—
I	(0.5)	(0.623)	-0.0499	↑	X_1^3
II	0.4996	(0.623)	-0.05	$\approx 10^{-4}$	X_1, X_1^3
III	0.4998	0.6161	-0.05	↓	X_1, X_2, X_1^3
IV*	0.5000	0.6118	-0.05	↓	X_1, X_2, X_1^3

Note: (.) , N = true values. * Number of data points used = 6

cases 1-11. The table also gives complete results of the system identification using the IERMEE-LS algorithm when additional terms are deleted from the true model. Even when only six data points are used for estimation of model error and least squares fit, case IV, the results are quite accurate (Table 1, Fig. 2). The effect of initial conditions of the states on the estimation accuracy was also studied. Even if a few initial data points of (estimated) states and the estimated model error are not used in the least squares parameter estimation process (when the initial guesses of the states are quite inaccurate), the coefficients are estimated with good accuracy. If the initial

3.2 Continuous-time nonlinear system

The true nonlinear model used has the following form [5]:

$$\dot{X}_1(t) = (2.5 \cos(t) - 0.32X_1(t) - X_2(t) - 0.05X_2^3(t))/2.56 \quad (30)$$

$$\dot{X}_2(t) = X_1(t) \quad (31)$$

Fig. 3 shows the results of estimation when two terms (x_2 and x_2^3) are removed from the true model. The corresponding model error time history is fitted with a general model of the form:

$$d_1(t_j) = a_1X_1(t_j) + a_2X_2(t_j) + a_3X_2^3(t_j) + a_4X_2^2(t_j) \quad j = 1, \dots, N \quad (32)$$

The parameter estimation results are given in Table 2. The number of data points used for estimation are about 300 for cases I-III. The results of estimation using the IEBMEE-LS algorithm when other terms are deleted from the true model are given in Table 2. From the above two exercises it is clear that the IEB-MEE-LS algorithms correctly estimate the model error even for the cases where the models are made deficient by removing several terms and a few data points are used. The indication of the term which is not present is also very good the estimate of a_4 being of the order of 10^{-4} . If the initial values of

the states are not very accurately known, one can iterate over the same equations, first solving them in forward direction and then, with the improved final states, in the

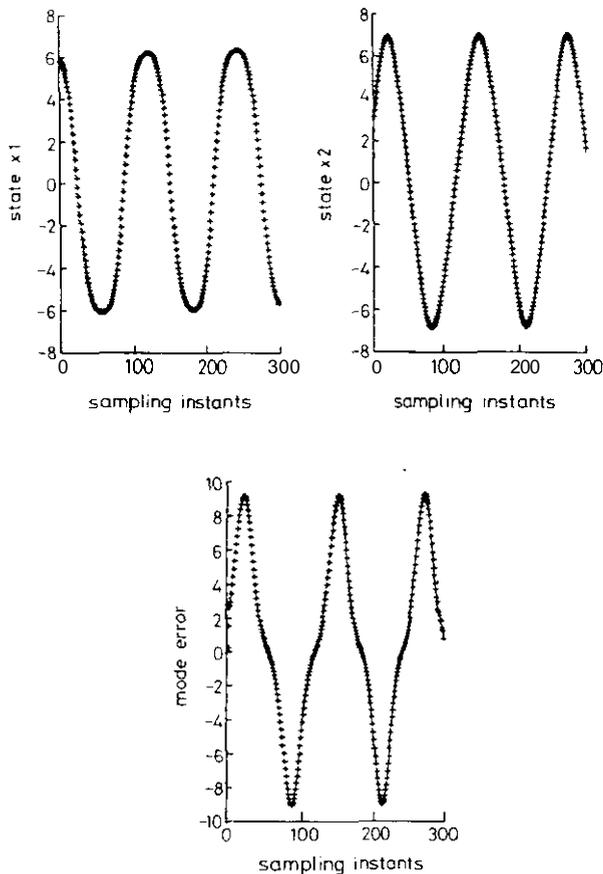


Fig. 3 Time histories of state and model error (X_2, X_2^3 terms deleted-CT case II)
 --- true
 + est.

Table 2: Nonlinear identification results: Continuous-time

Parameter	$-a_1$ (X_1)	$-a_2$ (X_2)	$-a_3$ (X_2^3)	a_4 (X_2^2)	Terms removed
TV	(0.125)	(0.3906)	(0.0195)	(0)	—
I	(0.125)	(0.3906)	0.0193	↑	X_2^3
II	(0.125)	0.3931	0.0191	$\approx 10^{-4}$	X_2, X_2^3
III	0.1242	(0.3906)	(0.0195)	↓	X_1
IV*	(0.125)	(0.3906)	0.0194	↓	X_2^3

* Less number of data used.

backward direction. This aspect retains the flexibility of sequential solution (estimates are updated at each sampling step) within an iterative process for further refinement of the estimates. However, for the case studies presented, such forward-backward iterations were not required.

3.3 Aerodynamic parameter estimation

It is important to identify and estimate the contribution of nonlinear effects in an aerodynamic model of an aircraft from flight data obtained during HAOA dynamic manoeuvre. Light transport aircraft data were simulated for further validation of the algorithm presented in Section 2.2 using the true model:

$$\dot{w} = (C_{z_0} + C_{z_\alpha} \alpha + C_{z_q} q \bar{c} / U_0 + C_{z_{\delta e}} \delta e) \bar{q} S / m + qu + g \cos \theta \quad (33)$$

$$\dot{q} = (C_{m_0} + C_{m_\alpha} \alpha + C_{m_{\alpha^2}} \alpha^2 + C_{m_q} q \bar{c} / U_0 + C_{m_{\delta e}} \delta e) \times \bar{q} S \bar{c} / I_y \quad (34)$$

$$\dot{u} = (C_{x_0} + C_{x_\alpha} \alpha + C_{x_{\alpha^2}} \alpha^2) \bar{q} S / m - qw - g \sin \theta \quad (35)$$

$$\dot{\theta} = q \quad (36)$$

Here w is the vertical velocity, q is the pitch rate, u is the forward component of aircraft velocity vector and θ is the pitch angle of the aircraft. The C s represent various aerodynamic derivatives which are to be estimated. The true model contains nonlinear terms in q and u equations. The deficient model is formulated by deleting the two nonlinear terms from the state eqns. 34 and 35. The simulated data were generated by using a doublet input, δe , as the elevator control surface deflection. The sampling time was 30 ms. Fig. 4 shows the time histories of the true and

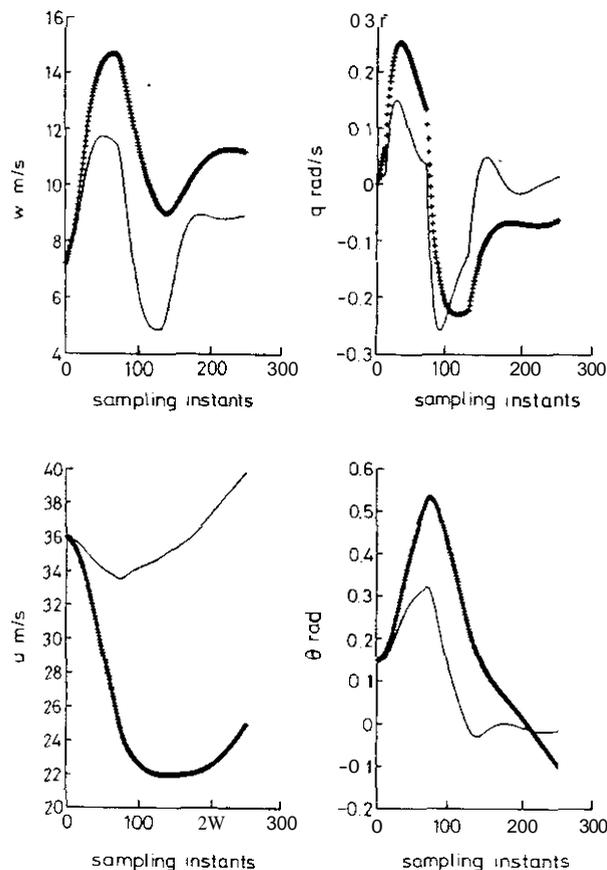


Fig. 4 Time histories of true and deficient states

($C_{x_{\alpha^2}}, C_{m_{\alpha^2}}$ terms deleted-aircraft example)

--- true
 + dci. state

deficient states. Using the continuous-time IEBMEE algorithm, the states are estimated recursively. Fig. 5 shows the true and estimated states. To determine the functional dependence of the model error on the system states, $d(t)$ must be converted into explicit function of $x(t)$, given by $K(x(t), \theta)$. Once K is found, it is added to the deficient model f' to produce the true model of the system ($g = f' + K_j$). To determine the correct form of K , for each element of d , the time history of a candidate function K is formed using the estimated states and is correlated with the model error estimates. The linear correlation coefficient ρ for the pair $(d_i(t), k_i(t))$ is then computed [8]. When $|\rho(d, k)| = 1$, $d(t)$ is linearly related to the candidate form. This form k_i is parameterised to yield the final form K . The correlation results of the deficient model are given in Table 3 and the results of

parameter estimation in Table 4. A suitable digital filter was used within the recursive structure of the IEBMEE algorithm to filter out excessive noise before the states.

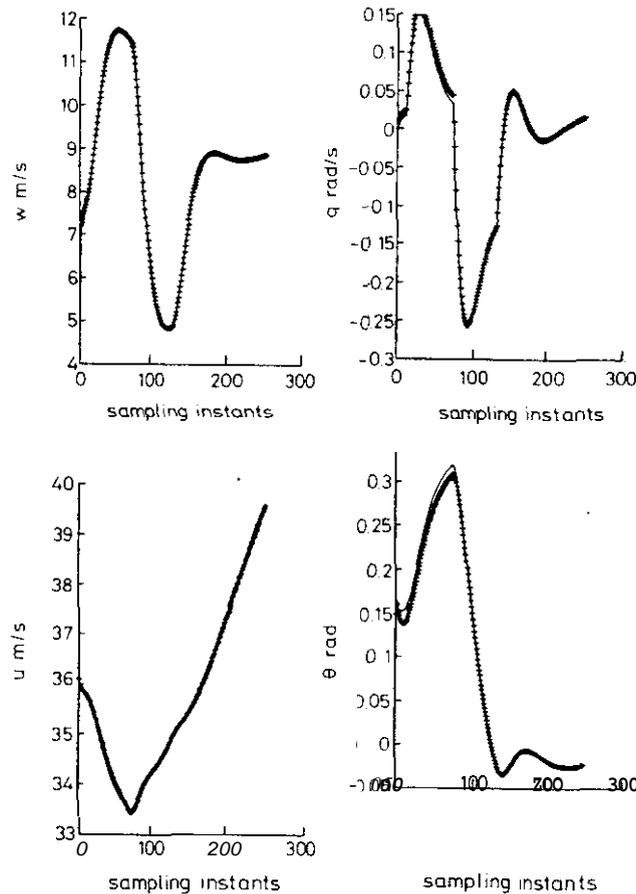


Fig. 5 Time histories of true and estimated states

($C_{x,2}$; $C_{m,2}$ terms deleted-aircraft example)

— true
+ est.

Table 3: Correlation results: aircraft case study

(1) For parameter $C_{m,2}$ ρ		(2) For parameter $C_{x,2}$ ρ	
$C_{m,2}$	09995	$C_{x,2}$	09995
$C_{m,3}$	09854	$C_{x,3}$	09853
$C_{m,4}$	09599	$C_{x,4}$	09596
$(C_{m,2} + C_{m,3})$	09982	$(C_{x,2} + C_{x,3})$	09982
$(C_{m,2} + C_{m,4})$	09988	$(C_{x,2} + C_{x,4})$	09987
$(C_{m,3} + C_{m,4})$	09800	$(C_{x,3} + C_{x,4})$	09798
$(C_{m,2} + C_{m,3} + C_{m,4})$	09970	$(C_{x,2} + C_{x,3} + C_{x,4})$	09969

Table 4: Results for aircraft case study

	$C_{x,2}$	$C_{m,2}$
TV	(3609)	(=1715)
no noise	'3605	-1656'
SNR 20	3649	-1625
SNR 10	3562	-1633

SNR = signal to noise ratio.

and the model error time histories were used in the recursive least squares parameter estimation. The entire process of: estimation of states and model error by IEBMEE algorithm, filtering of the noise, and parameter estimation by least squares is performed recursively.

4 Conclusions

Recursive algorithms based on the technique of invariant embedding are presented for estimation of deterministic

model errors in nonlinear systems. These algorithms have features similar to the extended Kalman filter. When interpreted in terms of 'information providing matrix' and in contradistinction to it, they also provide accurate estimates of the model discrepancy in the form of time histories. These time histories, when parameterised in the least squares sense, yield the coefficients of the models by which the true models were deficient. The usefulness and very high accuracy provided by these novel schemes are established by the results presented.

5 References

- 1 EYKOFF, P.: 'System identification-parameter and state estimation' (Wiley, New York, 1974)
- 2 SINHA, N.K., and KUTSZA, B.: 'Modeling and identification of multivariable dynamic systems' (Van Nostrand, New York, 1953)
- 3 JATEGAONKAR, R.V., RAOL, J.R., and BALAKRISHNA, S.: 'Determination of model order for dynamical systems', *IEEE Trans.*, 1992, **SMC-12**, (1), pp. 56-62
- 4 KLEIN, V., BATTERSON, J.G., and MURPHY, P.C.: 'Determination of airplane model structure from flight data by using modified stepwise regression'. NASA TP 1916, 1981
- 5 MOOK, D.J., and JUNKINS, J.L.: 'Minimum model error estimation for poorly modeled dynamic systems'. AIAA 25th aerospace sciences meeting (AIAA-87-0173), Reno, Nevada, 1987
- 6 DESAI, R.C., and LALWANI, C.: 'Identification techniques' (McGraw-Hill Publishing Company Ltd., New Delhi, 1973)
- 7 GELB, A. (Ed.): 'Applied optimal estimation' (MIT Press, USA, 1974)
- 8 MEYER, T.J., and MOOK, D.J.: 'Robust identification of nonlinear aerodynamic model structure'. AIAA-92-4503-CP, 1992

6 Appendix

6.1 TPBV problem and method of invariant embedding

The minimisation of 'J' using the Euler-Lagrange conditions results in a nonlinear TPBV problem. The boundary conditions are split [I, 6]. For state variable x , it is specified at the beginning $\{x(0) = x_0 - P_0 \lambda(0)\}$ and for the costate variable λ , it is specified at the end of the same interval $\{\lambda(T) = 0\}$. Eqns. 18 and 19 are written in the compact form as:

$$\dot{x}(t) = \Phi(x(t), \lambda(t), t) \quad (37)$$

$$\dot{\lambda}(t) = \Psi(x(t), \lambda(t), t) \quad (38)$$

The method of invariant embedding can be used to solve the above TPBV problem to obtain the recursive estimators for state as well as the model error. The specific problem as above is embedded into the more general case: $\lambda(T) = c$. The quantities c and T are now assumed to be variable: a change in c and T will influence the terminal value of λ . The following relations are established:

$$\lambda(T) = c \rightarrow x(T) = r(c, T) \quad (39)$$

$$\lambda(T + \Delta T) = c + \Delta c \rightarrow x(T + \Delta T) = r(c, T) + \Delta x \quad (40)$$

or

$$x(T + \Delta T) = r(c + \Delta c, T + \Delta T) \quad (41)$$

Subsequently, using eqns. 18, 19 and the above relations, the so-called invariant embedding equation is obtained as:

$$\Phi(r, c, T) = \{\partial r(c, T) / \partial c\} \Psi(r, c, T) + \partial r(c, T) / \partial T \quad (42)$$

An approximate solution in the neighbourhood of $c = 0$ is chosen: $r(c, T) = \hat{x}(T) - P(T)c$. The further development leads to the estimation algorithm for state and

model error (with T considered as a running variable) as given in the text with

$$x(0) = x_0 - P_0 \lambda(0) \quad \text{and} \quad \hat{x}(0) = x_0; \quad P(0) = P_0$$

Thus, since the terminal conditions are held invariant, the solution of the special case, the TPBV problem under question, is obtained from the solution of the more general problem.