Practical experience with unified discrete and continuous–time, multi–input identification for control system design

C. James Taylor * **Peter C. Young** ** **Philip Cross** *

* *Engineering Department, Lancaster University, UK (e–mail: c.taylor@lancaster.ac.uk)*
** *Lancaster Environment Centre, Lancaster University, UK; Fenner School of Environment and Society, Australian National University; (email: p.young@lancaster.ac.uk)*

Abstract: The paper is concerned with the practical aspects of a unified approach to the identification and estimation of multiple–input, single–output (MISO) transfer function models for both continuous and discrete–time systems. The estimation algorithms considered in the paper are based on the Refined Instrumental Variable (RIV) approach to identification and estimation, where the MISO model denominator polynomials are normally constrained to be equal. Unconstrained RIV estimation presents a more difficult problem and it is necessary to exploit an iterative, back–fitting routine to handle this more general situation. The paper focuses on the practical realization of this back–fitting algorithm, including its initiation from either common denominator MISO or repeated SISO estimation. The rivcdd algorithm for continuous–time model estimation, as implemented in the CAPTAIN Toolbox for Matlab^{TM}, is then used in three practical examples: first, the modelling of solute transport and dispersion in a water body; secondly, modelling for two control problems, namely a pair of connected laboratory DC motors and a nonlinear wind turbine simulation.

Keywords: identification; discrete–time; continuous–time; instrumental variables; MISO

1. INTRODUCTION

This paper considers the identification of multiple–input, single–output (MISO) systems, where the denominator polynomials of the elemental transfer functions associated with each input variable are not constrained to be identical. The underlying routine is based on the latest version of the *Refined Instrumental Variable* algorithm, with full Box–Jenkins model estimation and identification [see e.g. Young, 2011], in which both the discrete–time (RIV) and the 'hybrid' continuous–time (RIVC) algorithms allow for the specification of a discrete–time Auto–Regressive Moving Average (ARMA) noise model. These are implemented as the routines rivbj and rivcbj in the Captain Toolbox for Matlab^{TM} [Taylor et al., 2007, Young, 2011].

However, the RIV/RIVC algorithms assume a common denominator polynomial for the MISO system, with only the numerator polynomial associated with each input variable estimated separately. The present paper outlines how two new CAPTAIN routines rivdd and rivcdd have been developed to handle the more general case, with the elemental transfer functions defined by separate numerator and denominator polynomials. The iterative 'back–fitting' framework developed here follows directly from earlier research by Young and Jakeman [1980] and Garnier et al. [2007], for the discrete and continuous–time cases. In the present research, these tools are brought together into a unified approach for both discrete–time and continuous–time system identification, with a particular focus on the continuous–time algorithms and their application to multivariable control problems. Hence, the paper discusses the practical realization of the back–fitting algorithm and develops recommendations for initializing the routines at the start of this algorithm.

To highlight these issues and to evaluate the approach, the paper considers three practical examples: first, the modelling of solute transport and dispersion in a water body [Martinez and Wise, 2003, Young, 2011]; and then the modelling of connected laboratory DC motors [Bytronic Associates, 1981] and a nonlinear wind turbine simulation [Leithead and Connor, 2000]. Although not the main focus of the paper, the estimated MISO models in the latter two examples are used subsequently for the development of either discrete or continuous–time control systems, based on *Non–Minimal State Space* (NMSS) design methods [Young et al., 1987, Taylor et al., 2000].

Section 2 defines the initial model structure and briefly reviews the RIV approach for estimating the model coefficients. Section 3 develops the algorithm for estimating independent denominator polynomials and describes its practical realization. This is followed in sections 4 and 5 by the worked examples and conclusions.

2. BACKGROUND

Consider the following deterministic, continuous–time (CT) and discrete–time (DT) MISO transfer function models with common denominator polynomials:

CT Model

$$x(t) = \frac{B_1(s)}{A(s)}u_1(t-\delta_1) + \cdots + \frac{B_p(s)}{A(s)}u_p(k-\delta_p) \quad (1)$$

where s is the differential operator, i.e. $s^p x(t) = d^p x(t)/dt^p$, while $x(t)$ and $u_i(t)$ are the noise–free output and input signals, respectively, sampled uniformly at discrete times $t_1 \cdots t_N$. Although such a uniform sampling restriction is not essential in a CT environment, a non–uniform sampling option is not considered in this paper.

DT Model Here, the uniformly sampled signals, denoted by $x(k) = x(t_k)$ and $u_i(k) = u_i(t_k)$ are related by,

$$x(k) = \frac{\Phi_1(z^{-1})}{\Theta(z^{-1})}u_1(k) + \cdots + \frac{\Phi_p(z^{-1})}{\Theta(z^{-1})}u_p(k) \quad (2)$$

where z^{-1} is the backward shift operator, i.e. $z^{-r}y(k) = y(k-r)$.

For brevity, the definitions of the polynomials $A(s)$, $B_i(s)$, $\Theta(z^{-1})$ and $\Phi_i(z^{-1})$ are omitted but they take a similar form to those given in the references cited above. Specific examples of the CT polynomials are given in section 4. In the case of the CT model, the pure time delay δ_i is assumed to be an integer number related to the sampling time Δt. In the DT case, time delays are conveniently represented by setting the leading elements of the appropriate numerator polynomials to zero.

With uniformly sampled data, the measured output $y(k)$ in either the DT or CT cases, is assumed to be corrupted by an additive measurement noise $\xi(k)$,

$$y(k) = x(k) + \xi(k) \quad (3)$$

where,

$$\xi(k) = \frac{D(z^{-1})}{C(z^{-1})}e(k) \quad ; \quad e(k) = \mathcal{N}(0, \sigma^2) \quad (4)$$

is the ARMA noise part of the model, in which the Gaussian white noise input $e(k)$ has zero mean value and variance σ^2. For CT systems, $x(k)$ and $u_i(k)$, $i = 1, 2, \ldots, p$, are uniformly sampled values of $x(t)$ and $u_i(t)$, $i = 1, 2, \ldots, p$, in equation (1), where the sample time is the $k\Delta t$.

The models given by either equation (2) or (1), together with the measurement equations (3) and (4), are estimated using the rivbj and rivcbj routines of the CAPTAIN Toolbox [Taylor et al., 2007, Young, 2011], respectively. Full details of the algorithms are given in Young [2011] and the prior references therein. It will suffice here, therefore, to point out that the model is estimated using an optimal *Instrumental Variable* (IV) approach that generates asymptotically unbiased and minimum variance estimates of the system and noise model parameters. This involves an iterative (or recursive–iterative, if recursive estimates are required) 'relaxation' algorithm where equations (2) and (1) are utilised as 'auxiliary models' to generate IVs as part of the iterative optimization routine.

In order to achieve optimality both the CT and DT algorithms include optimal prefiltering of the input, output and IV signals. In the situation where $D(z^{-1}) = 1$ and $C(z^{-1}) = A(z^{-1})$, the prefilter transfer function is simply $1/A(z^{-1})$, and the estimation approach is then referred to as the *Simplified Refined Instrumental Variable* (SRIV) algorithm. This is an option of the rivbj and rivcbj routines in CAPTAIN and is important in the initiation of the rivdd and rivcdd routines.

3. UNCONSTRAINED ESTIMATION

The present section considers the identification of MISO models that are similar to equation (3), but where each input $u_i(k)$, $i = 1, 2, \ldots, p$, in equations (2) or (1) are associated with different denominator polynomials, i.e. $A_1(s) \cdots A_p(s)$ or $\Theta_1(z^{-1}) \cdots \Theta_p(z^{-1})$, respectively. This approach is often more realistic than assuming an identical denominator for each transfer function and, as discussed later, can have important practical advantages in the context of multivariable control system design. In this case, however, a modified RIV approach is required since the full MISO model cannot be manipulated into the 'pseudo–linear' form [Solo, 1980, Young, 2011] required for the standard RIV approach.

The iterative, back–fitting algorithm developed to solve this problem applies to both CT and DT models, and derives from similar algorithms for DT models in Jakeman et al. [1980] and CT models in Garnier et al. [2007]. The approach is based on RIV estimation of p Single–Input, Single–Output (SISO) models,

$$y_i(k) = x_i(k) + \xi(k) \qquad i = 1, 2, \ldots, p \quad (5)$$

where $x_i(k)$ is the noise–free output associated with the i^{th} input variable. The CT noise–free model is considered in the following rather informal manner (mixing transfer functions in s with sampled input and output variables),

$$x_i(k) = \frac{B_i(s)}{A_i(s)}u_i(k) \quad (6)$$

which represents a 'snapshot' of the continuous–time sub–model output at the k^{th} sampling time instant. For DT systems, the noise–free model is,

$$x_i(k) = \frac{\Phi_i(z^{-1})}{\Theta_i(z^{-1})}u_i(k) \quad (7)$$

Of course, the $x_i(k)$, $i = 1, 2, \ldots, p$, are not available for measurement and so the back–fitting concept assumes that it is possible to estimate the parameters of equations (7) or (6) by successively generating estimates of $\hat{x}_i(k)$, $i = 1, 2, \ldots, p$, of $x_i(k)$ and then applying the appropriate RIV or RIVC algorithm in SISO mode, using the sampled input–output data $\{u_i(k), \hat{x}_i(k)\}$, to estimate the constituent sub–models of the MISO model. Here, the $\hat{x}_i(k)$, $i = 1, 2, \ldots, p$, estimates are based on the latest updated sub–models. In particular, using equations (3), (5) and (6) for CT systems, this operation can be specified as follows:

$$\hat{y}_i(k) = y(k) - \sum_{j \neq i}^{p} \hat{x}_j(k) = y(k) - \sum_{j \neq i}^{p} \frac{\hat{B}_j(s)}{\hat{A}_j(s)}u_j(k) \quad (8)$$

where the 'hats' on the polynomials indicate the latest estimates available in the iterative algorithm. The DT sub–models are defined in a similar manner, using equations (3), (5) and (7) for DT systems.

The 'relaxation' algorithm for estimating the final MISO model proceeds with a cyclic SISO exploration of each input–output data set obtained in the above manner. The rivcdd algorithm, as implemented in CAPTAIN, can then be summarised as follows:

(1) **Stage 1: Initialization.** The first stage of the routine takes one of two (user–selected) options, where the rivcbj algorithm is utilised in SRIV mode, i.e. with $D(z^{-1}) = 1$ and $C(z^{-1}) = A(z^{-1})$ as discussed in section 2, to estimate either: (i) the MISO model (1) with common denominator polynomial; or (ii) the set of p SISO models for the relationships between $y(k)$ and each $u_i(k)$. For option (i), equation (8) is revised to use the common denominator $A(s) = A_1(s) = A_2(s) \cdots = A_p(s)$. Equation (8) is then utilised to generate $\hat{y}_i(k)$ for $i = 1 \cdots p$.

(2) **Stage 2: Back–fitting iterations.** The rivcbj algorithm is again utilised in SRIV mode, here to estimate p SISO models for the relationship between $\hat{y}_i(k)$ and $u_i(k)$ $\forall i$. Equation (8) is subsequently utilised to update $y_i(k)$ for $i = 1 \cdots p$. This step is repeated $j = 1 \cdots (J-1)$ times, where J is the total number of back–fitting iterations. Normally convergence is achieved in $J \leq 5$ iterations but this is determined by the application of a more convenient automatic convergence criterion.

(3) **Stage 3: Final estimates with noise model.** For the final J^{th} iteration, the rivcbj algorithm is utilised in full, hybrid Box–Jenkins mode to estimate SISO models for the relationship between $y_i(k)$ and $u_i(k)$ $\forall i$, together with the ARMA noise model.

The estimated polynomials of the MISO model, together with numerous estimation statistics, such as the parameter estimation error covariance matrices and statistical measures of model fit, are computed and returned. Full details are omitted for brevity, but these are equivalent to those obtained for conventional rivbj/rivcbj estimation. The rivdd algorithm utilises exactly the same procedures as those outlined above, except that the rivbj algorithm naturally replaces the rivcbj algorithm.

Remarks

(1) The rivbj and rivcbj algorithms are able to produce recursive estimates of the model parameters which are useful in model structure/order identification (see the first example in section 4).

(2) Note that even when a noise model is specified, the SRIV option is used in Stage 2. This considerably enhances the robustness and speed of the algorithm in practical applications, without affecting the final estimates.

(3) There is no guarantee of convergence for the above back–fitting algorithm, although there have been no problems in all applications considered so far. However, convergence is affected by the choice of the initiation option. This is most dependent on the relative *Steady State Gain* (SSG) of each elemental TF model and the consequent contribution that it makes to the explanation of the output $y(k)$. When SISO estimation is used to estimate the i^{th} TF in the MISO situation, the effects of the other inputs, $u_j(k)$, $j \neq i$, act as noise on the the output and so the noise/signal ratio will be large when the relative SSG is small. In such a situation, the resulting SISO estimate can be poor and this can lead to a poor initiation. An example of this is in the case of the well known 'Winding' process example [Garnier et al., 2007], as considered also in Young [2011], where the estimated SSGs are 1.2, 0.17 and 0.25. SISO initiation fails in this case, while common denominator MISO initiation works well. At this stage of the practical evaluation, therefore, we recommend that this is considered as the preferable option [1] .

(4) Although the rivcdd algorithm has worked well for practical applications, it is slower than rivdd: e.g. in the Winding process example the computation time on a 2.66 GHz Mac OS X is 9.4 secs., compared with the 5.1 secs. of rivdd.

(5) It is essential that any practical implementation of the approach includes suitable diagnostics and error checking. In this regard, the Captain rivcdd and rivdd implementation is designed to smoothly return null results and appropriate warnings/advice should any converge problems arise.

(6) The nature of the back–fitting procedure means that the covariance matrix for all the MISO model parameters (system and noise) is constrained to be block diagonal [see Garnier et al., 2007]. This constraint can be justified theoretically in the case of SISO model estimation [see Pierce, 1972, Young, 2011], where the resulting estimates are consistent and statistically efficient. But the parameter estimates do not necessarily retain these properties in the MISO situation.

(7) Following from the comments in (6), it is important to note that the RIV approach ensures that, if satisfactory convergence occurs, the estimates are always consistent, asymptotically unbiased, and should be efficient or relatively efficient: see the next comment (8). Indeed, in the case of the Winding process example, rivdd model estimation produces better results than pem [see Young, 2011] (although it is not so computationally efficient: taking 5.1 secs., compared with 3.9 secs. for pem on a 2.66 GHz Mac OS X).

(8) Finally, the rivcdd and rivdd estimates should be statistically efficient if the p input variables are statistically independent. However, more theoretical research is required in this connection, particularly since all MISO estimation methods can suffer from problems of ambiguity when the input variables are correlated (the dynamic equivalent of collinearity in linear regression). More research is also required to establish whether the ordering of the back–fitting operations can affect the estimation results.

4. PRACTICAL EXAMPLES

Although some of the above comments are empirically–based and cannot be generalised to all problems, they have been developed from numerous numerical simulation studies and control applications, three of which are discussed below. Also, other studies in the previously cited references report comprehensive Monte Carlo simulation results that establish the good statistical properties of the rivcdd estimates, as well as those of the rivdd estimates when compared with the pem estimates.

4.1 Modelling Tracer Experiment Data

This example is concerned with modelling the transport and dispersion behaviour of solutes in a wetland area,

[1] Note: this is in contrast to Garnier et al. [2007], who prefer the SISO option.

based on data obtained in a tracer experiment [Martinez and Wise, 2003]. The data are plotted in Fig. 1, which also shows the output of the rivcdd estimated model. Full details of the data used in this example are also given by Young [2011]. The best identified model has the following structure:

$$\begin{aligned} x(t) &= \frac{b_{10}}{s+a_{11}}u_1(t-26) + \frac{b_{20}}{s+a_{21}}u_2(t-40) \\ \text{or} \quad x(t) &= \frac{G_1}{1+T_1 s}u_1(t-26) + \frac{G_2}{1+T_2 s}u_2(t-40) \\ y(t_k) &= x(t_k) + \xi(t_k) \end{aligned} \tag{9}$$

where the additive noise is modelled by an AIC identified AR(8) process. The associated parameter estimates for the system model are as follows:

$$\begin{aligned} &\hat{a}_{11} = 0.09544(0.0015);\ \hat{a}_{21} = 0.02582(0.00085) \\ &\hat{b}_{10} = 0.06703(0.0010);\ \hat{b}_{20} = 0.008270(0.00025) \\ &G_1 = 0.70;\ G_2 = 0.32;\ \sigma^2_{\xi(k)} = 0.0016 \\ &T_1 = 10.5 \text{ h};\ T_2 = 38.7 \text{ h};\ \sigma^2_{e(k)} = 0.00017 \end{aligned}$$

The coefficient of determination based on the simulated output error is $R^2_T = 0.998$ and, since the noise is modelled in this case, it is possible to compute the coefficient of determination based on the residual white noise (one step–ahead prediction errors), $e(k)$, which is evaluated as $R^2 = 0.9998$. By contrast, the identified, first order, common denominator MISO model does not explain the data so well ($R^2_T = 0.996$); and the second order model provides a similar explanation of the data but is not so well identified, with the recursive estimates taking a much longer time to converge (clear signs of over–parameterization). Also, the first order, common denominator model has component TFs with equal time constants of $T_1 = T_2 = 24.7$ hours (with the same time delays of 26 and 40 hours), compared with the very different time constants obtained above, and the latter make more physical sense. Note that the total steady state gain of the model (9) is close to unity $G_1 + G_2 = 1.02$, which also makes physical sense in this case.

Fig. 2 compares the outputs of the two estimated first order transfer functions, as well as their combined effect (i.e. the deterministic output of the full model, which is the same as the model output plotted in Fig. 1), with the output tracer series. Fig. 3 shows the recursive estimation results, demonstrating the recursive estimation capability of rivcdd and indicating the good convergence, after the initial uncertainty, with well defined, low variance estimates of all the system model parameters. The recursive estimates of the noise model parameters are not shown in order to conserve space.

4.2 Connected Laboratory DC Motors

This straightforward laboratory demonstrator considers the angular velocity of a DC electric motor subjected to a time–varying disturbance load, where the load is generated by a second DC motor connected by means of a shaft and universal joints. The apparatus comprises two circuit boards, each consisting of a DC motor, a power supply, an optical sensor and a drive controller. The motors are of a 12V permanent magnet type with an integral gearbox, for which the output shaft can run at 330rpm.

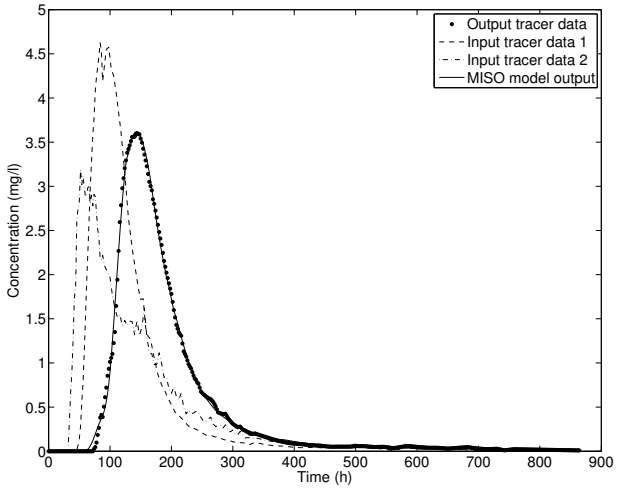

Fig. 1. Tracer example: plots of the output and the two input tracer concentration changes, together with the deterministic output of the rivcdd estimated model.

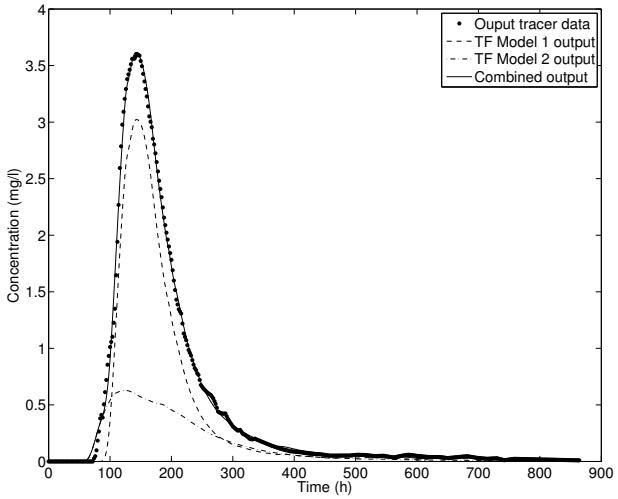

Fig. 2. Tracer example: rivcdd estimation results showing the contribution that the two additive component TF model outputs make to the total output.

The power supply is stepped down from 230V to 18V via a transformer, then full–wave rectified and regulated. A slotted disk rotates with the motor, interrupting the light beam generated by the optical sensor and producing a signal train of pulses related to the rotational velocity of the motor [Bytronic Associates, 1981].

The experiment is controlled with dSPACE hardware and software [dSPACE GmbH, 2008]. A real–time interface program generates C code from block diagrams designed using Simulink^{TM} [The MathWorks, 2009]. The speed of the motor is governed by the average of the high frequency pulse width modulated voltage applied across its terminals, related to the 'duty cycle' of the signal. Here, the duty cycle of a signal is defined as the fraction of one period, in which it has an active high state. In

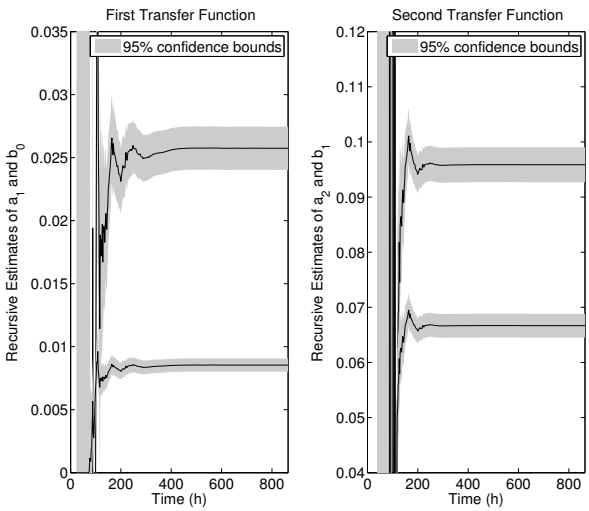

Fig. 3. Tracer example: recursive estimation results.

addition, the dSPACE controller is used to measure the time difference between successive pulses from the optical sensor, from which the angular velocity of the controlled motor is obtained.

A sampling rate of 100Hz has been chosen for data collection and control, since trial–and–error experimentation suggests this yields suitable closed–loop performance. For the experimental data illustrated by Fig. 4, the SRIV option of the rivcdd routine yields the following system model,

$$x(t) = \frac{226.9}{s + 4.398} u_1(t) - \frac{263.7}{s + 3.882} u_2(t) \qquad (10)$$

where $u_1(t)$ and $u_2(t)$ are the duty cycle ($0 \rightarrow 1$) of the controlled and disturbance motors, respectively, and $x(t)$ is rotational velocity (rad/s) of the controlled motor. The Coefficient of Determination $R_T^2 = 0.9923$. Note that differences in the time constants for the two motors emerge from the analysis, whilst the gain of the disturbance model is negative since $u_2(t)$ operates in the opposite direction to $u_1(t)$. Cross [2011] shows how such a MISO model is utilised for feed–forward control system design. In such cases, it can be useful to have independently defined transfer functions for the actuator and disturbance.

4.3 Nonlinear Wind Turbine Simulation

A nonlinear simulation of a variable speed wind turbine that includes the rotor aerodynamics, drive–train dynamics, power generation unit dynamics and wind speed disturbances has been developed by Leithead and Connor [2000]. Here, the output variables $x_1(t)$ and $x_2(t)$ represent the hub velocity and generator reaction torque respectively, whilst the control inputs $u_1(t)$ and $u_2(t)$ are the pitch angle demand and firing angle. The rivcdd analysis, using the SRIV option, yields the following linear, multivariable, MISO models, which have been used in an illustrative multivariable control exercise [Cross, 2011],

$$\begin{aligned} x_1(t) &= \frac{b_{11}}{s + a_{11}} u_1(t) + \frac{b_{12}}{s + a_{12}} u_2(t) \\ x_2(t) &= \frac{b_{21}}{s + a_{21}} u_1(t) + \frac{b_{22}s + b_{23}}{s + a_{22}} u_2(t) \end{aligned} \qquad (11)$$

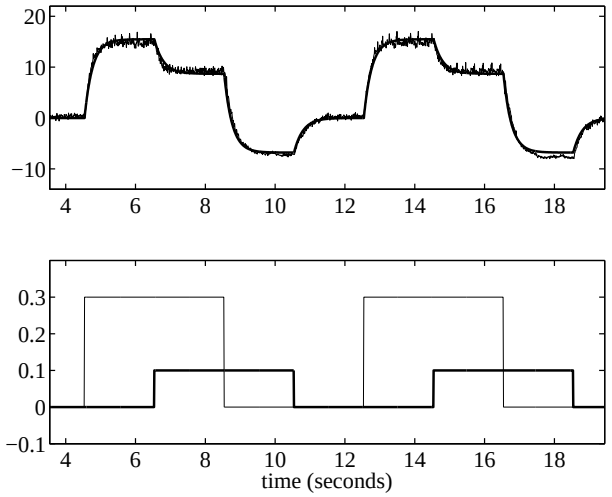

Fig. 4. Connected DC motors open–loop data. Upper subplot: rotational velocity (noisy traces) and CT back–fitted model (10). Lower subplot: scaled voltages to controlled (thin trace) and disturbance motors.

Table 1 compares parameter estimates for: (i) Common denominator (cd) MISO models for each output variable, obtained from one open–loop simulation experiment in which both input variables are active; (ii) MISO models with different denominator (dd) polynomials, obtained from the same data set using the back–fitting algorithm; and (iii) SISO models obtained from two new open–loop experiments, i.e. with one input constant. With regard to a practical device, models (i) and (ii) are the most realistic in terms of the data utilised for the analysis; however, for this simulation example, the SISO models (iii) provide a useful benchmark.

In the latter regard, Table 1 shows that the back–fitting algorithm returns parameter estimates that are, in most cases, almost the same as the identified SISO models. Note that whilst all three approaches yield similar estimates (not surprising for this deterministic, albeit nonlinear simulation example), only the back–fitting algorithm is able to isolate different SISO transfer functions for each input–output pathway, from a single realistic experiment (not shown) in which both $u_1(t)$ and $u_2(t)$ are active.

The closed–loop response of the nonlinear simulation is shown by Fig. 5, in which the off–diagonal subplots demonstrate that $x_1(t)$ and $x_2(t)$ have been successfully decoupled using the MISO model as the basis for NMSS control system design. Although details of the CT pole assignment control algorithm, with analytical multivariable decoupling, are beyond the scope of the present paper, this example demonstrates the practical utility of the identified models in (11).

It should be stressed that Fig. 5 is based on an illustrative set of operating levels and disturbance signal. With some realizations of the stochastic wind disturbance, the control performance deteriorates significantly. For the problematic cases, the disturbance takes the output variable sufficiently far from the design operating condition to cause substantial model mismatch, indicating that state–dependent, nonlinear versions of the control algorithm may be required [Taylor et al., 2009, 2011]. Nonetheless, the

linear models identified here, provide information about the underlying model structure for future investigations into state–dependent parameter models.

Table 1. CT linear models obtained for small peturbations of the nonlinear wind turbine simulation.

	MISO (cd)	MISO (dd)	SISO
a_{11}	0.0759	0.0754	0.0755
a_{12}	0.0759	0.0763	0.0763
a_{21}	0.0770	0.0760	0.0762
a_{22}	0.0770	0.0790	0.0784
b_{11}	–1.3258	–1.3168	–1.3177
b_{12}	0.1689	0.1697	0.1697
b_{21}	–0.2394	–0.2357	–0.2367
b_{22}	–0.4741	–0.4784	–0.4778
b_{23}	–0.0057	–0.0059	–0.0058

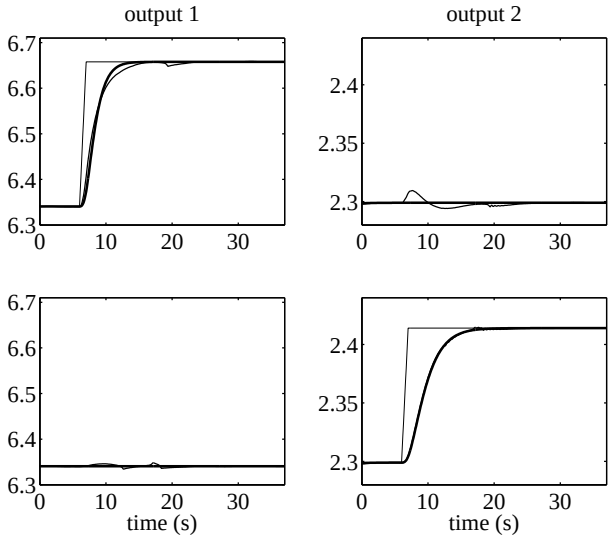

Fig. 5. Closed–loop response of nonlinear wind turbine simulation with disturbances (thin traces), and CT linear models (thick), to step changes in the command input for $x_1(t)$ (upper subplots) and $x_2(t)$ (lower).

5. CONCLUSIONS

This paper has considered unified algorithms for the estimation of multiple–input, single–output transfer function models, where the characteristic polynomials associated with each input are not constrained to be identical. Following a similar approach to Jakeman et al. [1980] and Garnier et al. [2007], iterative, back–fitting algorithms are developed that allow for the identification and estimation of both continuous and discrete–time MISO systems.

The paper has focused on the continuous–time version of the algorithm, using both simulation studies and laboratory control applications to develop robust default settings. Although convergence of the algorithm is not guaranteed, it seems to be strongly convergent and robust in the practical applications considered so far, provided it is initiated satisfactorily. In this regard, two initiation options are available in the rivcdd and rivdd routines that have been developed for the CAPTAIN Toolbox.

ACKNOWLEDGEMENTS

The CAPTAIN toolbox can be freely downloaded from `http://www.es.lancs.ac.uk/cres/captain` or via a link at the website `http://captaintoolbox.co.uk`.

REFERENCES

Bytronic Associates. *Notes and software for the TQ-Bytronic DC motor unit*, 1981.

P. Cross. *Continuous–time PIP control of marine energy converters*. PhD thesis, Lancaster University, Engineering Department (subject to viva), 2011.

dSPACE GmbH. *Experiment Guide For ControlDesk Release 6.2*, 2008.

H. Garnier, M. Gilson, P. C. Young, and E. Huselstein. An optimal IV technique for identifying continuous–time transfer function model of multiple input systems. *Control Engineering Practice*, 15:471–486, 2007.

L. J. Jakeman, L. P. Steele, and P. C. Young. Instrumental variable algorithms for multiple input systems described by multiple transfer functions. *IEEE Transactions on Systems, Man, and Cybernetics*, 10:593–602, 1980.

W. E. Leithead and B. Connor. Control of variable speed wind turbines: dynamic models. *International Journal of Control*, 73:1173–1188, 2000.

C. J. Martinez and W. R. Wise. Analysis of constructed teatment wetland hydraulics with the transient storage model OTIS. *Ecological Engineering*, 20:221–222, 2003.

D. A. Pierce. Least squares estimation in dynamic disturbance time–series models. *Biometrika*, 5:73–78, 1972.

V. Solo. Some aspects of recursive parameter estimation. *International Journal of Control*, 32:395–410, 1980.

C. J. Taylor, A. Chotai, and P. C. Young. State space control system design based on non–minimal state–variable feedback: further generalisation and unification results. *International Journal of Control*, 73:1329–1345, 2000.

C. J. Taylor, D. J. Pedregal, P. C. Young, and W. Tych. Environmental time series analysis and forecasting with the Captain Toolbox. *Environmental Modelling and Software*, 22:797–814, 2007.

C. J. Taylor, A. Chotai, and P. C. Young. Nonlinear control by input–output state variable feedback pole assignment. *International Journal of Control*, 82:1029–1044, 2009.

C. J. Taylor, A. Chotai, and K. J. Burnham. Controllable forms for stabilising pole assignment design of generalised bilinear systems. *Electronics Letters*, 47:437–439, 2011.

The MathWorks. *Matlab/Simulink*, 2009.

P. C. Young. *Recursive Estimation and Time Series Analysis: An Introduction for the Student and Practitioner*. Springer, 2011.

P. C. Young and A. J. Jakeman. Refined instrumental variable methods of recursive time-series analysis, part 3: extensions. *International Journal of Control*, 31:741–764, 1980.

P. C. Young, M. A. Behzadi, C. L. Wang, and A. Chotai. Direct digital and adaptive control by input–output, state variable feedback pole assignment. *International Journal of Control*, 46:1867–1881, 1987.