Simple refined IV methods of closed-loop system identification

Peter C. Young, Hugues Garnier, Marion Gilson

Simple Refined IV Methods of Closed-Loop System Identification

Peter C. Young * **Hugues Garnier and Marion Gilson** **

* *Systems and Control Group, Centre for Research on Environmental Systems and Statistics, Lancaster University, U.K; School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney Australia and Fenner School of Environment and Society, Australian National University, Canberra. e-mail: p.young@ lancaster.ac.uk).*

** *Centre de Recherche en Automatique de Nancy, Nancy-University, CRNS, BP 239, F-54506 Vandœuvre-les-Nancy, France.*

Abstract: The paper describes a simple, two-stage instrumental variable method of closed loop identification and estimation. This can be used with both continuous and discrete-time transfer function models and the enclosed system can be unstable. The paper also shows briefly how a third stage of estimation can be added that induces statistical efficiency when the enclosed system is stable.

Keywords: identification, closed loop, instrumental variable, continuous-time, discrete-time.

1. INTRODUCTION

Provided there is a sufficiently exciting, external command input signal, the identification and estimation of a continuous-time transfer function model for a system enclosed within a closed automatic control loop has always been relatively straightforward when using *Instrumental Variable* (IV) estimation methodology. Over the past 40 years, examples range from early contributions, such as [Young, 1970], to very recent publications (e.g. Ahmed et al. [2008], Gilson et al. [2008]), although the number of published papers dealing with continuous-time systems is far less than those that consider discrete-time closed loop systems (see e.g. Van den Hof [1998] and the references therein). Also, most of the algorithms that have been suggested are sub-optimal in statistical terms: while the IV implementation ensures that the parameter estimates are consistent and asymptotically unbiased, they are not normally statistically efficient (minimum variance).

Some recent papers have considered optimal IV solutions to the problem. For instance, Gilson et al. [2008] utilize the *Refined Instrumental Variable* (RIV) algorithm for *Continuous-time models* of the 'hybrid' *Box-Jenkins* type (RIVCBJ), where the system is modeled in continuous-time and the additive noise process is a standard, discrete-time *AutoRegressive, Moving Average* (ARMA) process. This is a recent modification of the original *Simplified Refined Instrumental Variable* (SRIVC) algorithm [Young and Jakeman, 1980] that was only optimal in the case of additive white noise. Although appealing, the Gilson et al. approach is quite complex. In this paper, therefore, we consider a rather simpler approach and show how it can be extended to provide an optimal solution.

First, two rather obvious and very simple two-stage RIVCBJ-type approaches are investigated, both of which are sub-optimal. These are in the same spirit as the two-stage algorithm suggested by Van den Hof and Schrama [1993] for discrete-time systems and they have the same advantage of not requiring prior knowledge of the control system. However, the new two-stage algorithms are more sophisticated than the Van den Hof and Schrama approach because each stage exploits the appropriate open loop SRIVC/RIVCBJ algorithms: namely, the SRIVC algorithm (rather than the FIR model estimation used by Van den Hof) for estimating the control input signal, followed by the estimation of the enclosed, controlled system, based on this estimated control input, using either the SRIVC or the full RIVCBJ algorithm. Moreover, the unified nature of refined IV estimation [Young, 2008a] means that this same approach can also be applied in the discrete-time system case using the SRIV and RIVBJ algorithms for discrete-time transfer function model estimation. Conveniently, all of these algorithms are already available as computational routines the CAPTAIN Toolbox for Matlab, a fully functional demonstration version of which can be downloaded from http://www.es.lancs.ac.uk/cres/captain/.

Although the two-stage algorithms perform well and can even work well if the enclosed system is unstable, they are not statistically efficient. Conveniently, however, they point the way to the addition of a third stage that can induce optimality and so provide parameter estimates that have similar properties to those obtained from the RIVCBJ algorithm applied in the equivalent open loop situation. section 5 of the paper outlines this three-stage algorithm, which is very simple but restricted to situations where the enclosed system is stable.

2. THE GENERALIZED BOX-JENKINS MODEL

The two-stage algorithms are applied to either the hybrid continuous-time, stochastic transfer function model or the equivalent discrete-time model. Both of these models can be considered as special examples of the following *Generalized Box-Jenkins* (GBJ) model, which is a hybrid modification of the original Box-Jenkins model [Box and Jenkins, 1970]:

$$\begin{aligned} x(t_k) &= \frac{B(\rho)}{A(\rho)} u(t_k) \\ y(t_k) &= x(t_k) + \xi(t_k) \\ \xi(t_k) &= \frac{D(z^{-1})}{C(z^{-1})} e(t_k) \qquad e(t_k) = \mathcal{N}(0, \sigma^2) \end{aligned} \tag{1}$$

Here $u(t_k)$, $x(t_k)$, $y(t_k)$, $\xi(t_k)$ and $e(t_k)$ are, respectively: the control input; the noise-free output; the noisy, measured output; the additive ARMA noise at the output; and the zero mean, normally distributed white noise source to the ARMA noise model, all sampled uniformly at a sampling interval Δt. The generalized operator ρ is the backward shift operator z^{-1} in the discrete-time case and the inverse of the derivative operator s^{-1}, where $s = d/dt$, in the hybrid continuous-time case.

This model is assumed to be contained within a feedback system, such as that shown in Figure 1, which represents the specific case of a unity feedback controller with the forward path control element represented as a ratio of polynomials $P(\rho)$ and $Q(\rho)$. However, the estimation procedures described in this paper will work with any linear control system structure or design. The command input to the control system is denoted by $r(t_k)$ and this is assumed to be statistically independent of the additive noise to the system $\xi(t_k)$. The control system used in the illustrative example considered later in section 4 is based on the discrete-time *Proportional-Integral-Plus* (PIP) control system design methodology (see e.g. Young et al. [1987], Taylor et al. [2000] and the prior references therein).

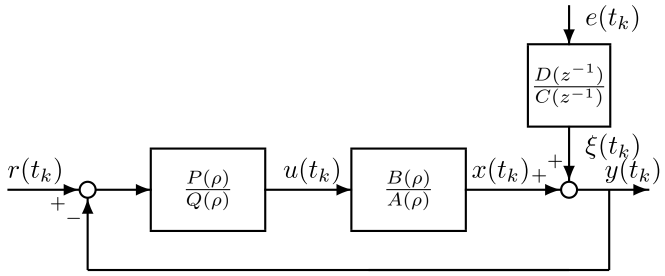

Fig. 1. Closed-loop configuration

3. TWO-STAGE CLOSED LOOP ESTIMATION

Provided measurements of $r(t_k)$, $u(t_k)$ and $y(t_k)$ are available, then two rather obvious approaches to closed loop estimation are as follows:

(1) Estimate the parameters of the TF model between $r(t_k)$ and the measured, noisy control input $u(t_k)$ using the appropriate simplified refined IV algorithm. The deterministic output of this model then provides a good estimate $\hat{u}_k$ of the 'noise-free' input to the enclosed system and so, at the second stage, the appropriate refined IV algorithms can be used again to estimate the required transfer function between $\hat{u}(t_k)$ and the noisy $y(t_k)$. This is the Two-Stage approach suggested by Young [2008a].

(2) Use the appropriate refined IV algorithm to estimate the parameters of the TF model for the whole closed loop system between $r(t_k)$ and the measured, noisy output $y(t_k)$. The deterministic output of this model then provides a good estimate $\hat{x}_k$ of the noise-free output from the system and the appropriate Refined IV algorithm can be used again, this time to estimate the transfer function between the two estimated variables $\hat{u}(t_k)$ and $\hat{x}(t_k)$. This approach is less satisfying in statistical terms than the first method because the final estimation involves two estimated noise-free variables, without direct reference to the measured output $y(t_k)$.

As pointed out previously and confirmed in the simulation examples of the next section 4, these algorithms can be used even if the enclosed system is unstable. However, it is necessary to use algorithms at the second stage of the estimation that are modified to allow for the estimation of an unstable system (the standard implementations in CAPTAIN are automatically restricted to stable systems).

4. ILLUSTRATIVE EXAMPLES: ESTIMATION WITHIN A PIP-LQ CLOSED LOOP

In this section, we consider a number of simulation examples, all of which concern the estimation of a TF model for a system contained within a digital PIP closed loop control system. The two-stage algorithms are implemented using existing routines available in the CAPTAIN Toolbox, where the the SRIVC and SRIV algorithms are options in the CAPTAIN **rivcbj** and **rivbj** routines, respectively.

In all of the examples, the TF system is based on the following second order, non-minimum phase, continuous-time model:

$$\frac{B(s)}{A(s)} = \frac{-0.5s + 1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{2}$$

where ζ and ω_n take on various values; while the discrete-time ARMA noise process always takes the form

$$\frac{D(z^{-1})}{C(z^{-1})} = \frac{1 + 0.5z^{-1}}{1 - 0.85z^{-1}} \tag{3}$$

and σ^2 in the model (1) is adjusted to provide different levels of additive noise. In the case of continuous-time estimation, this system is simulated in continuous-time within Simulink for 300 secs., using the variable step length ODE45 (Dormer and Prince) solver and enclosed within in a discrete-time, PIP control loop based on optimal *Linear-Quadratic* (LQ) design. For discrete-time model estimation, the Simulink simulation is based on the discrete-time equivalent of the continuous-time model, as obtained by conversion using the **c2d** routine in Matlab, with the zero-order-hold option at the selected sampling interval.

The PIP-LQ control system is implemented in the forward path form, with a diagonal cost function weighting matrix having weights of 100 on the integral-of-error state and unity on the other non-minimal state variables (past values of the input and output signals): see Taylor et al. [2000]. In all cases, the PIP control system is designed on the basis of the discrete-time equivalent of the continuous-time model at the selected sampling interval.

The first example evaluates the continuous-time model estimation performance with $\zeta = 0.05$ and $\omega_n^2 = 2$, and the sampling interval for both estimation and control system design is set at $\Delta t = 0.02$ seconds (15000 samples). The forward path PIP control transfer function in this case is:

$$\frac{P(z^{-1})}{Q(z^{-1})} = \frac{7.8553z^{-1} - 15.682z^{-2} + 7.8331z^{-3}}{1 - 2.4443z^{-1} + 2.0597z^{-2} - 0.61532z^{-3}} \quad (4)$$

In the second example, a discrete-time model is estimated, again with $\zeta = 0.05$ and $\omega_n^2 = 2$, but the sampling interval is set ten times larger at $\Delta t = 0.2$ seconds (1500 samples), which ensures that the eigenvalues of the discrete-time model are not too close to the unit circle in the complex z-plane. The forward path PIP control transfer function in this case is:

$$\frac{P(z^{-1})}{Q(z^{-1})} = \frac{4.2091z^{-1} - 7.971z^{-2} + 4.0917z^{-3}}{1 - 1.0985z^{-1} + 0.27071z^{-2} - 0.17222z^{-3}} \quad (5)$$

Note that no attempt is made here to control the closed loop non-minimum phase behaviour, although this is possible in PIP design using command input anticipation [Taylor et al., 1994]. Also, the system is not designed to maximize disturbance rejection, so the circulatory noise effects can be quite high.

4.1 Stable continuous-time system

In this first example, the command input signal $r(t_k)$ is a ± 1.0 PRBS signal of length 15000 samples. The *Monte Carlo Simulation* (MCS) analysis involves 100 realizations based on random selections of the white noise sequence $e(t_k), k = 1, 2, \ldots 15000$ and the white noise variance σ^2 is selected at two levels: first, 0.0025 in order to yield a 'standard' noise-to-signal ratio of 0.30 by standard deviation (i.e. std(noise)/std(signal)=0.3) on the control input signal $u(t_k)$ and 0.23 on the output signal $y(t_k)$; and then this is raised to 0.0133, in order to increase the noise-signal ratio at the input and output to 0.7 and 0.5, respectively. A typical segment of the data in the latter 'high' noise case is shown in Figure 2, where we see that the circulatory noise is quite large and highly coloured.

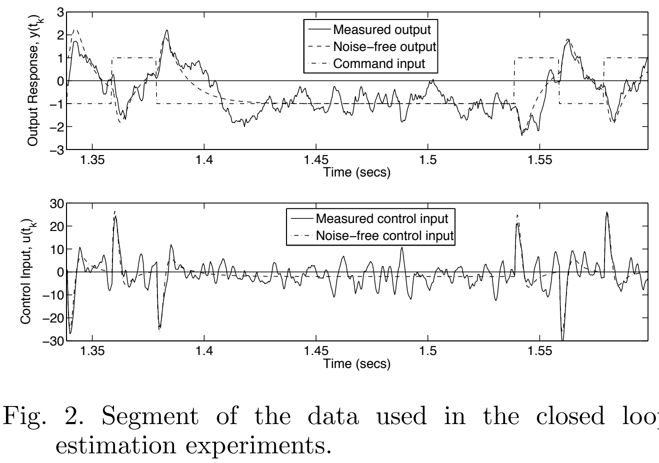

Fig. 2. Segment of the data used in the closed loop estimation experiments.

The YIC [Young, 1989] identifies a third order, [3 3 0] model between $r(t_k)$ and $u(t_k)$ with an ARMA(2,2) noise model. The SRIVC option of the **rivcbj** algorithm then estimates a model with a coefficient of determination $R_T^2 = 0.999$, based on the error between the [3 3 0] model output and the underlying *noise-free* input (i.e. the input $u(t_k)$ when there is no additive noise $\xi(t_k)$). In other words, 99.9% of the underlying noise-free control input variance is explained by the deterministic output of the estimated [3 3 0] model between $r(t_k)$ and $u(t_k)$.

Table 1. MCS Results: Continuous-Time Model (SD: standard deviation; SN: standard noise; HN: high noise)

Method	a_1	a_2	b_0	b_1
True	0.141	2.0	-0.5	1.0
Method 1 (SN)	0.140	1.999	-0.509	0.983
SD	0.0109	0.0167	0.0040	0.0099
Method 2 (SN)	0.140	1.999	-0.5090	0.984
SD	0.0108	0.0162	0.0045	0.0104
Method 1 (HN)	0.140	2.005	-0.511	0.987
SD	0.0328	0.0381	0.0088	0.0212
Method 2 (HN)	0.140	2.005	-0.510	0.988
SD	0.0319	0.0381	0.0099	0.0225

The results of the MCS analysis are shown in Table 1. These and other MCS results at even higher noise levels show that the parameter estimates are clearly consistent and asymptotically unbiased. As expected, however, the standard deviations of the TF denominator parameters are higher than those obtained using a statistically efficient three-stage algorithm (see later, section 5). Figure 3 shows the $5 \rightarrow 95$ percentile bounds (95% confidence bounds) associated with the ensemble of impulse responses obtained by Method 1 in the standard noise situation.

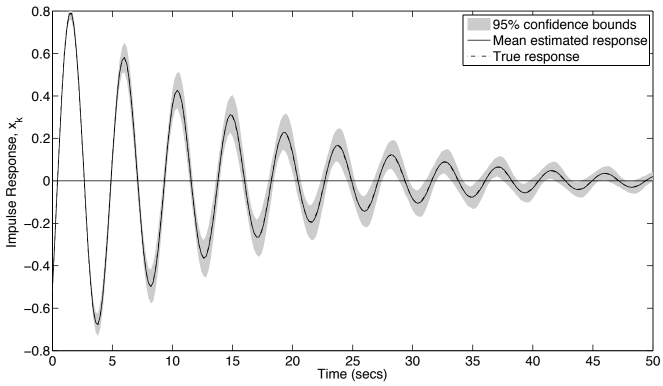

Fig. 3. MCS ensemble of continuous-time impulse responses obtained for Method 1.

4.2 Stable, discrete-time system

In this example, the command input signal $r(t_k)$ is a ± 1.0 PRBS signal of length 1500 samples and the white noise input signal $e(t_k)$ has variance of $\sigma^2 = 0.01$. The resulting noise-to-signal ratios, by standard deviation, are 0.35 at the output and 0.30 at the control input. The YIC (see previous sub-section 4.1) identifies a third order, [3 3 1] model between $r(t_k)$ and $u(t_k)$, with an ARMA(2,2) noise model. The SRIV option of the **rivbj** algorithm then estimates a model with a coefficient of determination between the [3 3 1] model output and the underlying noise-free input of $R_T^2 = 0.999$.

The results of the MCS analysis are presented in Table 2. These were obtained from 100 Monte Carlo realizations

Table 2. MCS Results: Discrete-Time Model

Method	a_1	a_2	b_0	b_1
True	-1.894	0.972	-0.0776	0.1167
Method 1	-1.894	0.972	-0.0776	0.1168
SD	0.0036	0.0033	0.0020	0.0018
pem	-1.893	0.972	-0.0774	0.1164
SD	0.0035	0.0032	0.0019	0.0018
Method 2	-1.894	0.972	-0.0765	0.1150
SD	0.0032	0.0028	0.0041	0.0046

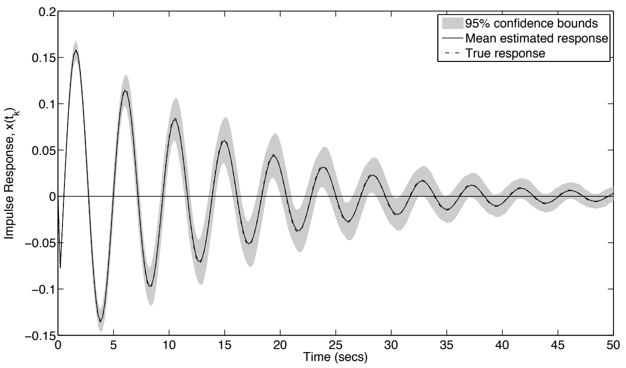

Fig. 4. MCS ensemble of discrete-time impulse responses obtained for Method 1.

based on random selections of the white noise sequence $e(t_k), k = 1, 2, \ldots 1500$. As in the continuous-time case of the previous section 4.1, these and other MCS results at higher noise levels show that the parameter estimates are clearly consistent and asymptotically unbiased. Also, while the standard deviations of the TF denominator parameters are higher than those obtained using a statistically efficient three-stage algorithm, the standard deviations match those obtained when the **pem** algorithm from the SID Toolbox in Matlab is used in place of **rivbj** at the second stage. In both cases, an ARMA(2,2) model is identified for the additive noise but, in order to simplify the presentation, the parameter estimates for these ARMA models are not shown.

Finally, the quality of the estimation is demonstrated graphically in Figure 4, which shows the ensemble of impulse responses for Method 1. Here, the 95% confidence bounds are a little higher than in Figure 3 for the continuous-time system, but this reflects the shorter sample size and the higher output noise level than in the continuous-time example.

4.3 Unstable and marginally stable system estimation

This sub-section presents three simulation examples that demonstrate the efficacy of the simple two-stage algorithms in the difficult but practically relevant case of unstable or marginally stable systems described by unstable continuous and discrete-time TF models. Although convergent estimates are obtained at high levels of noise, the variance of the parameter estimates is naturally higher than in the case of a stable system because the estimates are more sensitive when the system is unstable. This means that associated properties of the model, such as the unstable impulse response, can have very high variances.

Unstable continuous-time, oscillatory system For this example, the damping parameter ζ in the continuous-time model (2) is set to -0.05, i.e. the negative of its value in the stable system analysis. The MCS results presented in Table 3 are for a low noise situation, with the 11% noise by standard deviation at the output and 15% at the control input; while those in Table 4 are for output and control input noise levels of 22% and 30%, respectively. It is clear that, in both of these cases the two methods yield quite similar results. If the noise is increased still further,

Table 3. MCS Results, CT model, low noise

Method	a_1	a_2	b_0	b_1
True	-0.1414	2.0	-0.5	1.0
Method 1	-0.1407	1.992	-0.493	1.009
SD	0.0049	0.0106	0.0022	0.0077
Method 2	-0.1413	1.993	-0.492	1.010
SD	0.0038	0.0089	0.0017	0.0060

Table 4. MCS Results, CT model, standard noise

Method	a_1	a_2	b_0	b_1
True	-0.1414	2.0	-0.5	1.0
Method 1	-0.1415	1.989	-0.493	1.008
SD	0.0142	0.0262	0.0059	0.0199
Method 2	-0.1397	1.990	-0.493	1.007
SD	0.0109	0.0215	0.0039	0.0152

however, to give 100% noise at the output and 140% at control input, then method 2 performs considerably better than method 1, which yields unacceptable results. In this very high noise case, the ensemble averages obtained from the MCS analysis for Method 2 are as follows:

$$\begin{aligned} \hat{a}_1 &= -0.115(0.064) \quad \hat{a}_2 = 1.999(0.114) \\ \hat{b}_0 &= -0.493(0.026) \quad \hat{b}_1 = 0.999(0.084) \end{aligned} \tag{6}$$

where the standard deviations are shown in parentheses. We see that the estimates are still consistent and asymptotically unbiased but the variance is now very high, implying that the sample size in such very high noise situations would have to be substantially increased to obtain practically useful results. However, these are extremely high noise levels for a closed loop system and it is likely, in practice, that efforts would be made to reduce the noise levels both by good experiment design and tuning control system gains to enhance disturbance rejection.

Finally, Figure 5 compares the ensemble of impulse responses obtained for Method 1 in the low noise case (left panel) with those generated in the medium noise case (right panel). As one would expect, the variance of related properties, such as the impulse response, increase markedly as the noise level increases, because of their sensitivity to uncertainty when the system is unstable. Not surprisingly, therefore, the spread of impulse responses associated with the estimation results in (6) is very large indeed.

Discrete-time, unstable, oscillatory system This example is the discrete-time equivalent of the above continuous-time example and the MCS results are presented in Tables 5 and 6. Overall, the results are quite similar to those obtained in the continuous-time case and so the ensemble of impulse responses and higher noise results are not

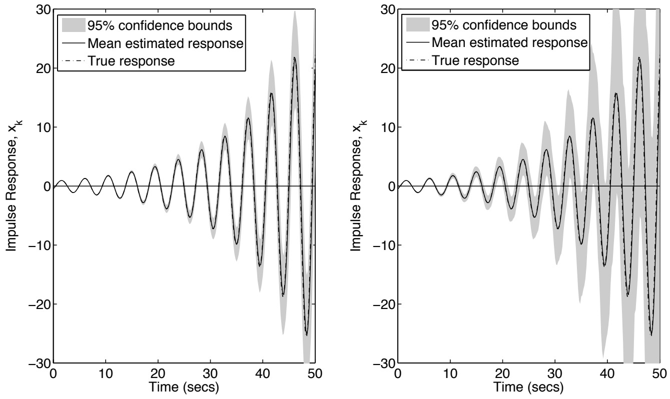

Fig. 5. MCS ensemble of continuous-time impulse responses obtained for Method 1: low noise, left panel; medium noise, right panel.

shown. Also shown in the tables, however, are the results obtained when the **pem** algorithm from the SID Toolbox in Matlab is used, rather than the **rivbj** algorithm. The results are not so good in this case, but this is probably because the algorithm is not intended for use with unstable systems.

Table 5. MCS Results, DT model, low noise

Method	a_1	a_2	b_0	b_1
True (CT)	-0.1414	2.0	-0.5	1.0
True (DT equivalent)	-1.948	1.029	-0.080	0.120
Method 1	-1.948	1.029	-0.080	0.121
SD	0.0027	0.0023	0.0039	0.0035
pem	-1.884	0.958	-0.051	0.0835
SD	0.0028	0.0021	0.0015	0.0021
method 2	-1.948	1.028	-0.0791	0.119
SD	0.0026	0.0024	0.0110	0.0105

Table 6. MCS Results, DT model, standard noise

Method	a_1	a_2	b_0	b_1
True (CT)	-0.1414	2.0	-0.5	1.0
True (DT equivalent)	-1.948	1.029	-0.080	0.120
Method 1	-1.947	1.027	-0.080	0.120
SD	0.0070	0.0074	0.0053	0.0049
pem	-1.885	0.959	-0.050	0.0828
SD	0.0075	0.0063	0.0042	0.0055
Method 2 (MCS)	-1.952	1.032	-0.083	0.123
SD	0.0082	0.0073	0.0087	0.078

At the very high noise level, the MCS analysis results are as follows, where the standard deviations are shown in parentheses:

$$\begin{aligned} \hat{a}_1 &= -1.946(0.058) \quad \hat{a}_2 = 1.027(0.067) \\ \hat{b}_0 &= -0.078(0.014) \quad \hat{b}_1 = 0.119(0.010) \end{aligned} \tag{7}$$

Continuous-time, double integrator system The continuous-time model (2) is converted to a double integrator system if $\omega_n = 0$. All other settings in the simulations are the same as in the stable system example of section 4.1 in the 'standard' noise situation. In this double integrator case, the open-loop system is, of course, violently unstable, and so this provides an acute test for any closed-loop identification algorithm.

Table 7. MCS Results: CT, Double Integrator

Method	a_1	a_2	b_0	b_1
True	0.0	0.0	-0.5	1.0
Method 1	-0.0175	-0.0059	-0.487	1.031
SD	0.0549	0.0224	0.0116	0.0539
Method 2	-0.0239	-0.0036	-0.485	1.034
SD	0.0561	0.0205	0.0111	0.0538

A typical set of of MCS results are summarized in Table 7 and we see that the estimated TF denominator parameters are insignificantly different from zero in statistical terms, with the standard deviations several times larger than the estimated values. In this example, however, there were two 'outlier' realizations, where the estimation errors were larger than those obtained in the rest of the ensemble. These two realizations were removed to compute the results shown in Table 7.

5. THREE-STAGE CLOSED LOOP ESTIMATION

Recent research [Young, 2008b] has shown how the two stage algorithm of Method 1 can be extended to include a third stage that induces statistical efficiency in the case of a stable enclosed system. However, this algorithm has the disadvantage that it cannot be used when the enclosed system is unstable.

The three-stages of the estimation algorithm are as follows, where it will be noted that the first two stages are very similar to the Method 1, outlined in the previous subsection 3, except that full Refined IV algorithms are utilized throughout:

Stage 1 Estimate the TF between the command input $r(t_k)$ and the measured, noisy control input $u(t_k)$ using the appropriate Refined IV algorithm, and generate an estimate $\hat{u}(t_k)$ of the underlying noise-free control input using this model.

Stage 2 Use the appropriate Refined IV algorithm to obtain initial, two stage estimates $\hat{A}(\rho)$ and $\hat{B}(\rho)$ of the system TF model polynomials $A(\rho)$ and $B(\rho)$, respectively, based on the estimated noise-free control input signal $\hat{u}(t_k)$ obtained in Stage 1 and the noisy measured output signal $y(t_k)$.

Stage 3 Compute the estimated noise part of the control input signal from $\hat{\eta}_{ni}(t_k) = u(t_k) - \hat{u}(t_k)$ and transfer this to the output using the system model obtained in Stage 2, i.e.

$$\hat{\eta}_{no}(t_k) = \frac{\hat{B}(\rho)}{\hat{A}(\rho)} \hat{\eta}_{ni}(t_k) \tag{8}$$

Subtract this estimated output noise signal from the measured output to obtain the following estimate of the output signal that does not include the circulatory noise component from the closed loop,

$$\hat{y}(t_k) = y(t_k) - \hat{\eta}_{no}(t_k) \tag{9}$$

This is, therefore, an estimate of the noise-free output plus only the additive noise $\xi(t_k)$. As a result, the data set $\{\hat{u}(t_k); \hat{y}(t_k)\}$ provides an estimate of the data set that would have been obtained if the system was being estimated in the equivalent open-loop situation. Finally, therefore, use the appropriate open-loop Refined IV algorithm for a second time to re-estimate the system model based on this constructed data set.

It should be noted that this three-stage procedure is straightforward to implement because it only makes use of estimation routines already available in the CAPTAIN Toolbox. As a result, new CAPTAIN routines **clrivcbj** and **clrivbj**, based on the three-stage algorithm and incorporating its simpler two-stage progenitors as options, are currently undergoing final β-testing.

Table 8. 3-Stage CT Algorithm Estimation (SR: single run; SE: standard error; SD: standard deviation; OL: equivalent open loop)

Method	a_1	a_2	b_0	b_1	c_1	d_1
True	0.141	2.0	-0.5	1.0	-0.85	0.5
SR	0.141	2.002	-0.509	0.981	-0.851	0.499
SE	0.0021	0.0028	0.0028	0.0055	0.0045	0.0075
MCS	0.140	2.000	-0.509	0.978	-0.856	0.487
SD	0.0025	0.0030	0.0030	0.0055	0.0047	0.0067
OL	0.140	1.999	-0.509	0.978	-0.852	0.501
SD	0.0022	0.0024	0.0030	0.0055	0.0047	0.0066

Table 8 shows the MCS results obtained with the **clrivcbj** routine in the case of the example described earlier in section 4.1 and these can be compared directly with the equivalent results shown in Table 1. The main difference in performance is a clear improvement in statistical efficiency, as demonstrated by the considerable reduction in the standard deviations (SD) of the TF denominator parameter estimates. In addition, the standard error (SE) estimates obtained from a typical single realization (SR) match the MCS standard deviation (SD) values; and the estimation results match the optimal results obtained in the equivalent open loop situation (OL) using the **rivcbj** algorithm, demonstrating the optimality of the **clrivcbj** estimation. Similar results are obtained when the **clrivbj** routine is applied to the discrete-time example in section 4.2.

6. CONCLUSIONS

This paper describes and evaluates an extremely simple but powerful, unified approach to the identification and estimation of a continuous *or* discrete-time transfer function model for a dynamic system enclosed with a feedback control loop, without requiring any knowledge of the control system structure or parameters. The paper concentrates on the application to continuous-time systems but discrete-time results are included to demonstrate the unified nature of the general approach. The continuous-time approach has its usual advantages (e.g. [Young and Garnier, 2006]) but it is computationally more intensive: the continuous-time algorithms are approximately ten times slower than the equivalent discrete-time algorithms.

The two-stage estimates of the TF model parameters are sub-optimal in statistical terms but the simulation results show that the TF model parameter estimates are statistically consistent and asymptotically unbiased for quite high levels of noise in the closed loop system. Moreover, this performance is maintained even if the enclosed system is unstable. In addition, recent research [Young, 2008b] has shown how an additional third stage can induce statistical optimality provided the enclosed system is stable. Early results from this three-stage algorithm are presented that demonstrate the statistical efficiency of the procedure.

All of the simulation results reported in the paper have been obtained using algorithms already available in the CAPTAIN Toolbox for Matlab[1], including the *Proportional-Integral-Plus* (PIP-LQ) control system design routines that are used in the design of the control systems.

REFERENCES

S. Ahmed, B. Huang, and S. L. Shah. Identification from step responses with transient initial conditions. *Journal of Process Control*, 18:121–130, 2008.

G. E. P. Box and G. M. Jenkins. *Time Series Analysis Forecasting and Control*. Holden-Day: San Francisco, 1970.

M. Gilson, H. Garnier, P. C. Young, and P. Van den Hof. Instrumental variable methods for closed-loop continuous-time model identification. In Hugues Garnier and Liuping Wang, editors, *Identification of Continuous-Time Models from Sampled Data*, pages 133–160. Springer-Verlag: London, 2008.

P. M. J. Van den Hof and R. J. P. Schrama. An indirect method for transfer function estimation from closed loop data. *Automatica*, 29:1523–1527, 1993.

C. J. Taylor, P. C. Young, and A. Chotai. On the relationship between GPC and PIP control. In D. W. Clarke, editor, *Advances in Model-Based Predictive Control*, pages 53–68, Oxford, 1994. Oxford University Press.

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

P. M. J. Van den Hof. Closed-loop issues in system identification. *Annual Reviews in Control*, 22:173–186, 1998.

P. C. Young. An instrumental variable method for real-time identification of a noisy process. *Automatica*, 6: 271–287, 1970.

P. C. Young. Recursive estimation, forecasting and adaptive control. In C. T. Leondes, editor, *Control and Dynamic Systems*, pages 119–166. Academic Press: San Diego, 1989.

P. C. Young. The refined instrumental variable method: unified estimation of discrete and continuous-time transfer function models. *Journal Européen des Systèmes Automatisés*, 42:149–179, 2008a.

P. C. Young. A three stage refined IV algorithm for closed loop identification. Centre for Research on Environmental Systems and Statistics, Report TR/210b, Lancaster University, 2008b.

P. C. Young and H. Garnier. Identification and estimation of continuous-time, data-based mechanistic models for environmental systems. *Environmental Modelling & Software*, 21:1055–1072, 2006.

P. C. Young and A. J. Jakeman. Refined instrumental variable methods of time-series analysis: Part III, 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. *International Journal of Control*, 46:1861–1881, 1987.

[1] see http://www.es.lancs.ac.uk/cres/captain/