



On closed loop transient response system identification

Christer Dalen¹ David Di Ruscio²

¹Skien, Norway. E-mail: christerdalen@hotmail.com

²University College of Southeast Norway, P.O. Box 203, N-3901 Porsgrunn, Norway. E-mail: david.di.ruscio@hit.no

Abstract

Some methods for transient closed loop step response system identification presented in the literature are reviewed. Interestingly some errors in a method published in the early 80's were propagated into a recently published method. These methods are reviewed and some improved methods are suggested and presented. The methods are compared against each other on some closed loop system examples, e.g. a well pipeline-riser severe-slugging flow regime example, using Monte Carlo simulations for comparison of the methods.

Keywords: unstable systems, closed loop, step response, system identification, indirect, non-parametric, transient response, pipeline-riser

1. Introduction

Closed loop subspace System IDentification (SID) has been given a large amount of effort the last two decades resulting in e.g. the DSR_e alg., Di Ruscio (2009). A problem with closed loop subspace SID is the possibility of correlations between the controller output and the process noise, if this is present then the results may be biased. However, this is solved in DSR_e, see Di Ruscio (2009). Closed loop SID is a relatively matured field and is documented in books as e.g. Ljung (1998), Söderström and Stoica (1989) etc.

The open loop plant may be unstable or very poorly damped and has to be operated in closed loop, or it might be for economic or safety reasons (Söderström and Stoica (1989)).

In this paper our main focus is the SID method used in so-called on-line closed loop step response tuning methods: Yuwana and Seborg (1982), Jahanshahi and Skogestad (2015).

We can classify such methods as indirect, as we need to know the reference and output time series, as well

as the controller (See Fig.1). We may further call such methods nonparametric, type transient analysis, as described in Söderström and Stoica (1989), which are examples of continuous identification methods.

The only notable strength of these methods is that they are easy to use, pragmatic and easy to understand.

However, the weaknesses are rather many. Firstly, nonparametric transient step response methods are sensitive to noise and do not give accurate results (Söderström and Stoica (1989) Ch.3). Secondly, one requires a high number of measurable quantities sampled at high frequency, e.g. a settled output time series response. Third, some of these curve points are rather strenuous to identify, e.g. the first valley point. Producing larger oscillations by increasing the reference change or the controller proportional gain could help, however this may be rather unappetizing in process industries.

One contribution of this paper is to show that the identification method given in Jahanshahi and Skogestad (2015) may be improved. It should be a well-known fact that if we are given an experiment signal using a

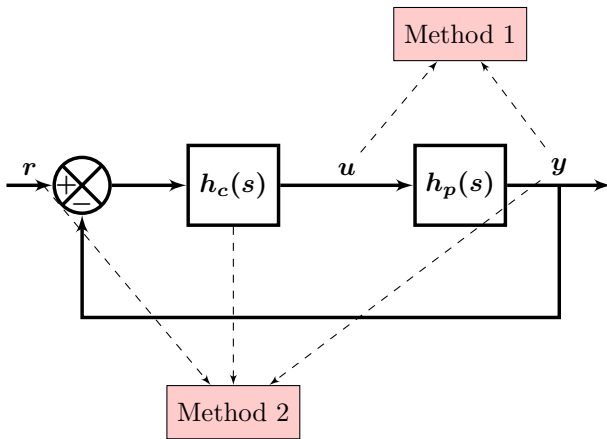


Figure 1: Block diagram of a closed loop system which is illustrating two SID methods. Method 1: Direct method using y and u . Method 2: Indirect method using r, y and h_c . Note that we also have a third approach called joint input-output method, where r, u and y are needed.

pure step signal only, then this signal is only persistent exciting of order one which is different from the second order model structure, assumed in [Jahanshahi and Skogestad \(2015\)](#). Furthermore, we may predict that by introducing process noise, this method will present itself inferior to other SID methods.

Notice however that this particular SID method, i.e. [Jahanshahi and Skogestad \(2015\)](#), is aimed for a pipeline-riser system, where the severe-slugging flow regime is modeled using a second order marginally stable/unstable open loop model. Severe-slugging flow, i.e. oscillations in the outlet flow is regarded as a problem in the offshore industry and needs to be addressed for safety and economic reasons ([Courbot \(1996\)](#)).

The contributions of this paper can be itemized as follows

- An error is discovered in [Jahanshahi and Skogestad \(2015\)](#) having its roots in [Yuwana and Seborg \(1982\)](#) for estimating the time instant of the first peak, t_p , giving implications for the ϕ estimate in [Jahanshahi and Skogestad \(2015\)](#). The modified algorithm is denoted JS DR.
- An alternative formulation to the algorithm presented in [Jahanshahi and Skogestad \(2015\)](#), is proposed. The proposed algorithm is denoted DR.
- Documentation is given for an additional error discovered in [Yuwana and Seborg \(1982\)](#) for the time constant and the time delay expressions of a first order plus time delay model.

- Counter-example on the closed loop transient response SID algorithm presented in [Jahanshahi and Skogestad \(2015\)](#). Monte Carlo simulation experiments show that the two transient response identification algorithms, JS DR and DR, considered in this paper, are inferior to the following subspace algorithms; DSR/DSR_{ry} ([Di Ruscio \(1996\)](#)) and DSR_e ([Di Ruscio \(2009\)](#)) and the Prediction Error Method (PEM).

- An m-file script developed for estimating the 6 necessary parameters from the transient closed loop step response e.g. t_p etc, is presented in App. A.

The rest of the paper is organized as follows. In Sec. 2 we formulate the improved JS DR method, as well as the proposed alternative, the DR method. In Sec. 3 we give numerical examples. In Secs. 4 and 5 discussion and concluding remarks are given, respectively.

2. On transient response SID algorithms

2.1. Improved JS DR method

Assume that the underlying system can be described by the following transfer function

$$h_p(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0}. \quad (1)$$

This particular model was used to describe the essential dynamics of a pipeline-riser system in [Jahanshahi and Skogestad \(2015\)](#).

By stabilizing with a P-controller, the closed loop transfer function becomes

$$y(s) = \frac{K(1 + \tau_z s)}{\tau^2 s^2 + 2\xi\tau s + 1} r(s), \quad (2)$$

where we introduce a reference step,

$$r(s) = \frac{r}{s}. \quad (3)$$

We restrict ourselves to $0 < \xi < 1$ and $\tau > 0$.

The problem renders itself in identifying the closed loop model parameters (ξ, τ, K, τ_z) , thereafter, by back calculation, we obtain the open loop model parameters (a_1, a_0, b_1, b_0) . These models are identified based on the data vectors,

$$Y = \begin{bmatrix} y_1^T \\ \vdots \\ y_N^T \end{bmatrix}, \quad R = \begin{bmatrix} r_1^T \\ \vdots \\ r_N^T \end{bmatrix}, \quad (4)$$

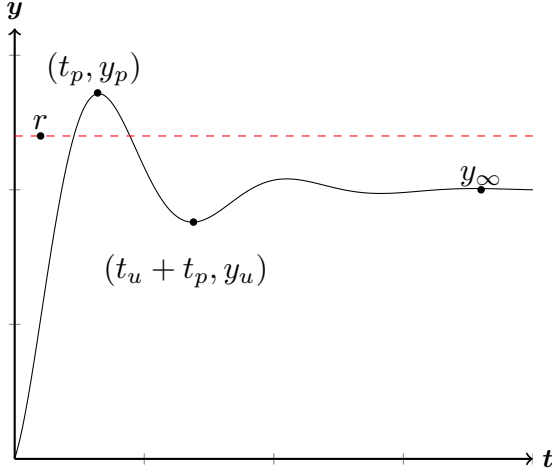


Figure 2: Closed loop step response. The x-axis t is time. The estimated parameters $(t_p, y_p, r, t_u, y_u, y_\infty)$ are used to identify (ξ, τ, K, τ_z) in Eq. (2). A MATLAB function for identifying these six parameters is found in App. A.

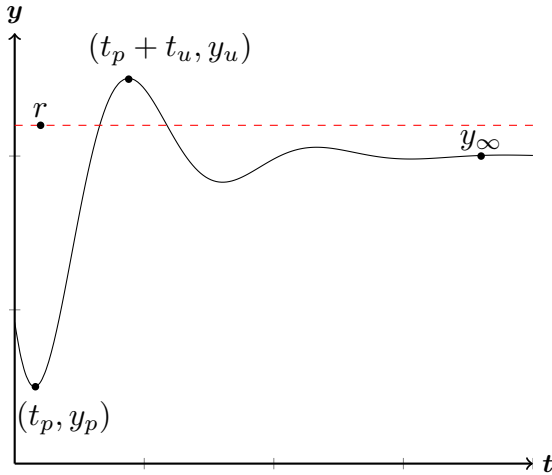


Figure 3: Closed loop step inverse response. The x-axis t is time. The estimated parameters $(t_p, y_p, r, t_u, y_u, y_\infty)$ are used to identify (ξ, τ, K, τ_z) in Eq. (2). A MATLAB function for identifying these six parameters is found in App. A.

generated from a closed loop step response as illustrated in Figs. 2 and 3.

The time transient closed loop response is

$$y(t) = Kr \left[1 - D e^{-\xi t/\tau} \sin(\omega t + \phi) \right], \quad (5)$$

where

$$D = \sqrt{\frac{1 - \frac{2\xi\tau_z}{\tau} + \left(\frac{\tau_z}{\tau}\right)^2}{1 - \xi^2}}, \quad (6)$$

$$\omega = \frac{\sqrt{1 - \xi^2}}{\tau}, \quad (7)$$

and

$$\phi = \arctan\left(\frac{\tau\sqrt{1 - \xi^2}}{\xi\tau - \tau_z}\right). \quad (8)$$

In Jahanshahi and Skogestad (2015) the time response, Eq. (A.2), is wrong, even though referring to the correct Eq. (5) or Eq. (A-1) in Yuwana and Seborg (1982).

The damping ratio ξ is estimated and deduced for a system with $\tau_z = 0$ in Söderström and Stoica (1989) Ch. 3 Eq. (3.3g), e.g.,

$$\xi = \frac{-\ln(v)}{\sqrt{\pi^2 + \ln^2(v)}}, \quad (9)$$

where the relative overshoot v is estimated as

$$v = \frac{y_\infty - y_u}{y_p - y_\infty}. \quad (10)$$

A proof of the expression of the damping ratio in Eq. (9) is given in App. F.

The steady-state gain of the closed loop system is defined as

$$K = \frac{y_\infty}{r}. \quad (11)$$

By putting the derivative of Eq. (5) w.r.t. time $\left(\frac{dy}{dt} = 0\right)$ equal to zero, thereafter solving w.r.t. $t = t_p$, we have that

$$t_p = \frac{\arctan\left(\frac{\sqrt{1 - \xi^2}}{\xi}\right) - \phi}{\omega}. \quad (12)$$

Solving Eq. (12) w.r.t. ϕ we obtain

$$\phi = \arctan\left(\frac{\sqrt{1 - \xi^2}}{\xi}\right) - t_p\omega. \quad (13)$$

For the 1st under-/overshoot we have that

$$t_p + t_u = \tau \frac{\arctan\left(\frac{\sqrt{1 - \xi^2}}{\xi}\right) + \pi - \phi}{\sqrt{1 - \xi^2}}. \quad (14)$$

Using Eqs. (12) and (14) we obtain

$$t_u = \tau \frac{\pi}{\sqrt{1-\xi^2}} = \frac{\pi}{\omega}. \quad (15)$$

Solving Eq. (15) for τ gives

$$\tau = \frac{t_u \sqrt{1-\xi^2}}{\pi}. \quad (16)$$

Evaluating Eq. (5) at $t = t_p$ gives

$$\begin{aligned} y(t_p) = y_p &= Kr \left[1 - De^{-\xi t_p/\tau} \sin(\omega t_p + \phi) \right], \\ &= Kr \left[1 - De^{-\xi t_p/\tau} \sqrt{1-\xi^2} \right], \end{aligned} \quad (17)$$

then by using $Kr = y_\infty$, and solving Eq. (17) w.r.t. D , we obtain

$$D = \frac{D_0 e^{\frac{t_p \xi}{\tau}}}{\sqrt{1-\xi^2}}, \quad (18)$$

where we have defined that

$$D_0 = \frac{y_\infty - y_p}{y_\infty}. \quad (19)$$

Where we in Eqs. (17) and (18) have used that $\sin(\omega t_p + \phi) = \sqrt{1-\xi^2}$, as shown in App. F.

Last, we solve Eq. (6) w.r.t. τ_z and get that

$$\tau_z = \tau \xi \pm \sqrt{\tau^2 \xi^2 - \tau^2 \left[(\xi^2 - 1) D^2 + 1 \right]}. \quad (20)$$

The algorithm is presented and summed up in Alg. 2.1.

Algorithm 2.1 (The JS DR algorithm)

Step 1. From the step response time series data Y and R in Eq. (4) and illustrated in Figs. 2 and 3, calculate the six parameters $(t_p, y_p, r, t_u, y_u, y_\infty)$. An algorithm for step 1 is presented in App. A.

Step 2. Calculate the 4 closed loop step response model parameters ξ from Eq. (9), τ from Eq. (16), K from Eq. (11) and τ_z from Eq. (20).

Step 3. Back calculate to the proposed open loop model structure in Eq. (1).

These adjustments propose the improved algorithm. The JS DR algorithm m-file implementation can be found in App. B.

In Jahanshahi and Skogestad (2015) Eq. (A.17) will not work for a case of inverse response, i.e. $\tau_z < 0$. This is solved by using Eq. (20) instead, however the drawback is that we need to know a priori if $\tau_z < 0$.

Interestingly, an error was found in Eq. (A-5) in Yuwana and Seborg (1982), which estimates the time instant, t_p , of the first peak. This error has propagated into wrong estimation of ϕ in Eq. (A.12), in Jahanshahi and Skogestad (2015). Notice here that Eq. (13) for ϕ should be used instead of the wrong Eq. (A.12) in Jahanshahi and Skogestad (2015). Note also that this error has no direct impact on the Yuwana-Seborg algorithm, but it impacts the estimate of τ_z in Eq. (20).

2.2. Proposed alternative algorithm (DR)

An alternative closed loop time transient response to Eq. (5) can delicately be expressed as

$$y(t) = Kr \left[1 - e^{-\frac{\xi}{\tau} t} (\cos(\omega t) + c \sin(\omega t)) \right], \quad (21)$$

where ω is the angular frequency, as defined in Eq. (7), and

$$c = \frac{\xi \tau - \tau_z}{\tau \sqrt{1-\xi^2}}. \quad (22)$$

Now, by simply solving Eq. (21) w.r.t. c at $t = t_p$, we obtain that

$$c = -\frac{\cos(t_p \omega) + \frac{e^{\frac{t_p \xi}{\tau}} (y_p - r K)}{r K}}{\sin(t_p \omega)}. \quad (23)$$

Notice that the above parameter c is related to the phase angle ϕ (Jahanshahi and Skogestad (2015)) as

$$\frac{1}{\tan(\phi)} = c, \quad (24)$$

and as in Eqs. (8) and (13).

By solving Eq. (22) w.r.t. τ_z , we have that

$$\tau_z = \xi \tau - c \tau \sqrt{1-\xi^2}, \quad (25)$$

or

$$\tau_z = \xi \tau - c \tau^2 \omega. \quad (26)$$

Eqs. (25) or (26) is more preferable than Eq. (20), partly because Eq. (20) is observed to produce complex solutions, hence, $\tau_z = \text{real}(\tau_z)$ is added in the JS DR algorithm. DR will not have the problem of having to know if $\tau_z < 0$, in prior, which gives DR an edge over JS DR.

The DR algorithm m-file implementation can be found in App. C. Note that the difference from JS DR in Alg. 2.1 is the estimating of τ_z .

Notice that an alternative formula for the time instant of the first extremum Eq. (12) is found by solving the derivative of Eq. (21) w.r.t. time. This gives

$$t_p = \frac{\frac{\pi}{2} + \arctan\left(\frac{c\xi + \tau\omega}{\xi - c\tau\omega}\right)}{\omega}. \quad (27)$$

The time instant of the first under-/overshoot is found to be

$$t_p + t_u = \frac{\frac{3\pi}{2} + \arctan\left(\frac{c\xi + \tau\omega}{\xi - c\tau\omega}\right)}{\omega}. \quad (28)$$

This gives the same as Eq. (15), which is obtained by putting Eq. (27) into Eq. (28) and solving for t_u . Solving Eq. (27) w.r.t. c we obtain

$$c = \frac{\xi + \tau\omega \tan(t_p\omega)}{\tau\omega - \xi \tan(t_p\omega)}. \quad (29)$$

We also propose an alternative DR2 algorithm, using c from Eq. (29) instead of Eq. (23). The DR2 algorithm m-file implementation can be found in App. D. Notice however, that this alg. DR2 is more sensitive to noise than both JS DR and DR.

2.3. The propagation of errors based in the Yuwana-Seborg article

Notice that in Yuwana and Seborg (1982) a first order plus time delay model, i.e.

$$h_p(s) = K \frac{e^{-d_m s}}{1 + t_m s}, \quad (30)$$

as the assumed open loop model is considered.

In Yuwana and Seborg (1982), Eqs. (9) and (10) are both wrong. These equations can also be found in Hapoglu et al. (1998).

However, the correct equations are

$$d_m = \frac{2\Delta t\sigma_2}{\pi\sigma_1}, \quad (31)$$

and

$$t_m = \frac{\Delta t\sigma_1\sigma_2}{\pi}, \quad (32)$$

where

$$\sigma_1 = \xi\sqrt{K+1} + \sqrt{\xi^2(K+1) + K - 1} \quad (33)$$

and

$$\sigma_2 = \sqrt{(1 - \xi^2)(K+1)}. \quad (34)$$

It might be on purpose that "-1" was removed in Eq. (33) for avoiding complex solutions. This error/approximation was also briefly noted in Jutan and Rodriguez (1984) and Taiwo (1993).

2.4. Further details and discussions

- The algorithms presented under this section are implemented in MATLAB as function $[b, a] = \text{alg}(Y, R, Kp, dt)$, where subscript 'alg' means JS DR or DR (see Apps. B and C for the algorithms).

- JS DR and DR are equipped with the following function, $y = \text{filtfilt}(b_f, a_f, x)$, with motivations for the incoming example. Here, 'filtfilt' is a function for zero-phase digital filtering, and b_f and a_f are from the function 'butter'. Both these functions are contained in the Signal Processing Toolbox in MATLAB. This filter is however manually tuned for the incoming example.

- An additional subspace algorithm, denoted DSR_{ry}, is presented under App. E. This algorithm is using the DSR algorithm on the closed loop data R and Y, then back calculating to the assumed open loop model.

- Note that the presented algorithms may automate various existing tuning methods, e.g. the 'Good Gain method' presented in Haugen (2012).

3. Numerical examples

Example 3.1 (Counter-example to transient SID)

Consider the continuous closed loop system described by the transfer functions, in Eq. (1) and

$$h_c(s) = K_p, \quad (35)$$

where $b_1 = -0.012$, $b_0 = -0.0041$, $a_1 = -0.0019$ and $a_0 = 0.0088$ and $K_p = -10$.

The unstable open loop model, h_p , is in fact a reduced order model based from a fourth order nonlinear model, which captures the severe-slugging behavior present in pipeline-riser processes. The open loop model parameters are chosen based on Sec. 6.1 in Jahanshahi and Skogestad (2015).

Eqs. (1) and (35) may be formulated as a discrete-time closed loop system,

$$y_k = Dx_k + w_k, \quad (36)$$

$$u_k = K_p(y_k - r_k), \quad (37)$$

$$x_{k+1} = Ax_k + Bu_k + v_k, \quad (38)$$

where the process noise v_k and measurement noise, w_k are chosen as white noise with covariance matrices $E(v_k v_k^T) = 0.5^2 I_2$ and $E(w_k w_k^T) = 0.05^2 I_1$, respectively.

Two different reference signals, r_k , are used, i.e. PRBS for the subspace algorithms/PEM and a simple step for the transient algorithms.

The closed loop system in Eqs. (36), (37) and (38) was simulated from $k = 1 \dots 1000$ with $dt = 1$ [sec] for the subspace algorithms/PEM and from $k = 1 \dots 40000$ with $dt = 0.01$ [sec] for the transient algorithms as illustrated in Fig. 4. Monte Carlo simulations are

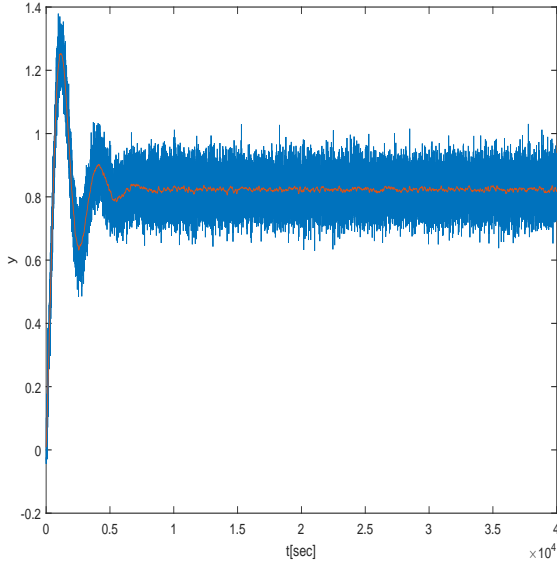


Figure 4: MATLAB zero-phase filtering $y = \text{filtfilt}(b, a, x)$.

performed using $M = 100$ experiments. The results from the simulations are the parameter estimates (b_1, b_0, a_1, a_0) in the open loop model (Eqs. (35)).

For comparing the results from the Monte Carlo simulations, we consider the following performance criterion, which is given by Eq. (B.67b) in *Söderström and Stoica (1989)*:

$$P_{alg} = \frac{N}{M-1} \sum_{i=1}^M (\hat{\theta}_i - \theta_0)(\hat{\theta}_i - \theta_0)^T, \quad (39)$$

where $\hat{\theta} = [\hat{a}, \hat{b}]$ is the estimated parameters, $\theta_0 = [a, b]$ is the true parameters, M simulations and N samples.

Results from the Monte Carlo simulations are shown in Figs. 5, 6 and Tab. 1. Our proposed algorithm, DR, is far more robust than the JSDR algorithm. Furthermore it shows that the subspace algorithms and PEM render themselves superior over the transient algorithms. PEM has the best performance while DSR_ry is the runner-up candidate just before DSR_e. Note that DR2 alg. is dropped in this example, reason being rather unstable results.

Example 3.2 (van de Vusse chemical reactor)

A chemical isothermal reactor (*van de Vusse (1962)*) is studied in this example. Consider the following closed

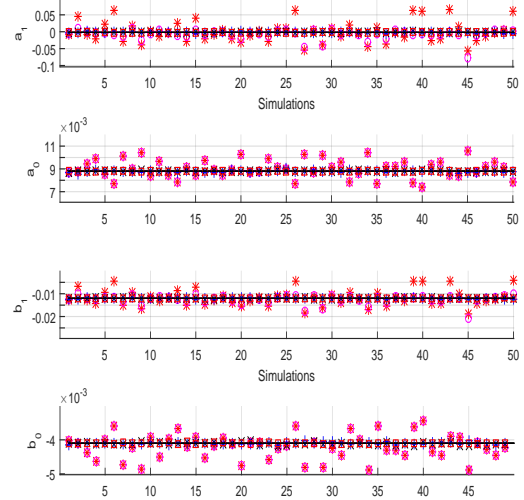


Figure 5: Parameter estimates of the open loop system, $h_p(s)$ in Eq. (35) showing JSDR (red asterisk), DR (magenta circle), PEM (red square), DSR_e (blue plus) and DSR_ry (black cross). Monte Carlo simulations are performed using $M = 100$ experiments. Only the first 50 points are plotted. The last 50 points can be seen in Fig. 6.

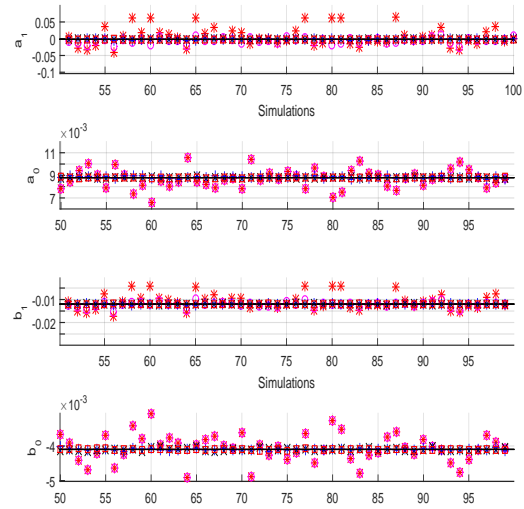


Figure 6: Parameter estimates of the open loop system, $h_p(s)$ in Eq. (1) showing JSDR (red asterisk), DR (magenta circle), PEM (red square), DSR_e (blue plus) and DSR_ry (black cross). Monte Carlo simulations are performed using $M = 100$ experiments. Only the last 50 points are plotted. The first 50 points can be seen in Fig. 5.

Table 1: Comparing the performance criteria generated from the different algorithms.

alg	P_{alg}
PEM	0.0001
DSR_e	0.0029
DSR_ry	0.0010
JSDR	39.2542
DR	6.8918

loop system

$$y = x_2, \quad (40)$$

$$u = K_p(r - y), \quad (41)$$

$$\dot{x}_1 = -k_1x_1 - k_3x_1^2 + (v - x_1)u, \quad (42)$$

$$\dot{x}_2 = k_1x_1 - k_2x_2 - x_2u, \quad (43)$$

where the reaction rate coefficients are $k_1 = 50$, $k_2 = 100$ and $k_3 = 10$. u is the feed flow rate, y is the concentration of the product out the outlet. The concentration of the byproduct into the reactor, v , is treated as an unknown constant or slowly varying disturbance with nominal value $v^s = 10$.

A sufficient large proportional gain is chosen, $K_p = 140$, i.e. large enough for stabilizing the closed loop system and giving a relatively large overshoot preferable for the transient algorithms.

Choosing a reference signal $r = 1.02$ gives the steady state $x^s = [2.1046, 0.8847]^T$. The closed loop system is then simulated on time $0 < t < 0.20$ (Fig. 7) with a sampling time equal 0.0001, given the initial state $x(t = 0) = x^s$ and a reference step for the transient algs. and a PRBS for the subspace algs.

However, it should be noted that the generated data is detrended, i.e. $Y := Y - Jy^s$, $R := R - Jr$ and $U := U - Ju^s$, where $y^s = x_2^s$, J is a vector of ones of correct length and N is the length of e.g. Y .

In Tab. 2, the DR2 algorithm estimates a model which is closest to the Jacobian estimate, based on y^s, u^s, x^s .

We compare the identified models, generated from the following algs., JSDR, DR, DR2, DSR_e and DSR_ry, over $M = 100$ different validation sets, i.e. M different PRBS reference realizations.

The performance is measured by the criterion,

$$Valg = \frac{1}{NM} \sum_{k=1}^M \left[\sum_{i=1}^N (\hat{y}_i - y_i)^2 \right]_k, \quad (44)$$

where \hat{y}_i is the true output and y_i is the simulated output from the identified models.

Results from the simulations are shown in Tab. 3. In this particular example it is shown that the JSDR alg. is the best performing of the transient algs., however the subspace algorithms gives the best models.

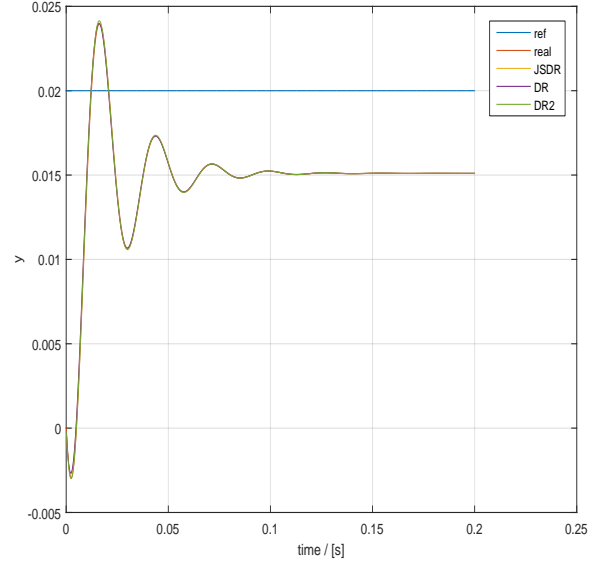


Figure 7: Comparing the closed loop step response of the van de Vusse chemical reactor model vs. the closed loop models, as in Eq. (2), generated from the transient algorithms JSDR, DR and DR2.

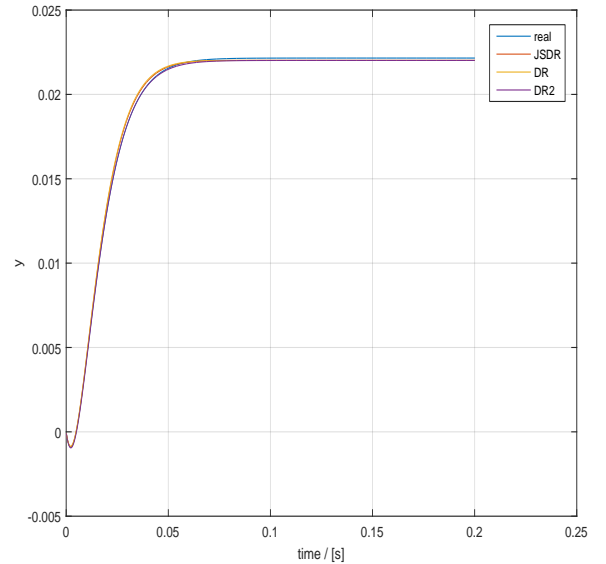


Figure 8: Comparing, using a step response, the Jacobian estimate, denoted 'real', vs. the open loop models, as in Eq. (1), identified using the transient algorithms JSDR, DR and DR2. Tab. 1 shows the numerical results.

Table 2: Comparing the different algorithms on estimating the parameters for the open loop model Eq. (1). Comparing the Jacobian estimate, donated 'real', vs the transient algorithms; JS DR, DR and DR2. Fig. 8 shows the graphical results.

	real	JS DR	DR	DR2
b_0	295.9561	293.1288	293.1288	293.1288
b_1	-0.8847	-0.8578	-0.8444	-0.9045
a_0 [e+04]	1.3358	1.3315	1.3315	1.3315
a_1 [e+04]	0.0231	0.0221	0.0219	0.0227
V [e-07]	0	2.2370	2.6703	1.3618

Table 3: Comparing the identified models, generated from the following algs., JS DR, DR, DR2, DSR_e and DSR_ry, over $M = 100$ different validation sets. The performance is measured by the criterion, V_{alg} .

V_{alg}	Value [e-08]
$V_{JS DR}$	0.7903
V_{DR}	1.6191
V_{DR2}	1.1782
V_{DSR_e}	0.0280
V_{DSR_ry}	0.0281

4. Discussion

- Notice that by defining v as in Eq. (10) and evaluating the time response, Eq. (5), it can be shown that the expression for ξ becomes as in Eq. (9). This was however also pointed out in the appendix in Yuwana and Seborg (1982).
- Notice that the back calculation step to developing the open loop model is not unique, as also described in Secs. 2.2 and 2.3.
- Notice, relative small sampling time dt have to be used for obtaining accurate estimates.
- As demonstrated in Ex. 3.1 the transient identification methods need a large number of observations compared to the classical subspace and prediction error methods. Even of this, the quality of the estimation results is poor compared to PEM/DSR as illustrated in Tab. 1.

5. Concluding Remarks

A couple of errors in Yuwana and Seborg (1982) have been documented, one of which caused implications for the algorithm presented in Jahanshahi and Skogestad (2015).

A new alternative formulation for the algorithm presented in Jahanshahi and Skogestad (2015) is proposed and compared with Monte Carlo simulations. The new algorithm DR is found to outperform the JS DR algorithm, however these transient response algorithms are rather inferior in comparison to the highly respected subspace algorithms, i.e. DSR/DSR_ry (Di Ruscio (1996)), DSR_e (Di Ruscio (2009)) and the classical prediction error method (PEM), e.g. Ljung (1998).

A. Function for estimating the 6 variables in Figure 2.

```
function [dyp, dyinf, dys, dyu, tu, tp] ...
=id_6_param2(Y,R,T)
```

```
[dyp, ip]=max(Y);
tp = T(ip);
dyinf=Y(end);
dys=max(R);
[dyu, iu]=min(Y(ip:end));
tu = T(ip+iu-1); tu=tu-tp;
```

B. The JS DR alg.

```
function [b, a, tauz, tau, xi, K2] ...
=JS DR(Y,R,T, Kc0, tz, fc)
% ON THE ARGUMENTS:
% Outputs in Y and references in R.
% Kc0 is the proportional gain used in
% the closed loop experiment.
% Sampling time is dt.
% The sign of tz need to be known in
% advance of the identification,
% tz='neg' for inverse response case,
% i.e. tz<0,
% else tz>0 is considered.
% A 1st order zero phase filter with
% cut-off frequency of fc (fc=0.01).
% Filter is not used when fc=-1.
```

```
if (fc~-1)
    [b, a]=butter(1, fc);
    Y=filtfilt(b, a, Y);
end
```

```
% Easy algorithm for finding the
% 6 variables for identification.
[dyp, dyinf, dys, dyu, tu, tp] ...
=id_6_param2(Y,R,T);
```

```
% Estimating the 5 variables xi, tau,
% K2, phi, tauz for the closed-loop
```



```

v=(dyinf-dyu)/(dyp-dyinf);
xi=-log(v)/sqrt(pi^2+(log(v))^2);
tau=tu*sqrt(1-xi^2)/pi;
K2=dyinf/dys;
E=sqrt(1-xi^2)/tau;
phi=atan(sqrt(1-xi^2)/xi)...
-tp*sqrt(1-xi^2)/tau;
D0=(dyp-dyinf)/dyinf;
D=D0/(exp(-xi*tp/tau)*sin(E*tp+phi));
if(strcmp(tz,'neg'))
    tauz=xi*tau-sqrt(xi^2*tau^2...
-tau^2*(1-D^2*(1-xi^2)));
else
    tauz=xi*tau+sqrt(xi^2*tau^2...
tau^2*(1-D^2*(1-xi^2)));
end
tauz=real(tauz);
% Back-calculating to the
% open-loop model
Kp=dyinf/(Kc0*abs(dys-dyinf));
a0=1/(tau^2*(1+Kc0*Kp));
b0=Kp*a0;
b1=K2*tauz/(Kc0*tau^2);
a1=-2*xi/tau+Kc0*b1;
b=[0 b1 b0];
a=[1 -a1 a0];
    
```

C. The DR alg.

```

function [b,a,tauz,tau,xi,K]...
=DR(Y,R,T,Kp,fc)
% ON THE ARGUMENTS:
% Outputs in Y and references in R.
% Kc0 is the proportional gain used
% in the closed loop experiment.
% Sampling time is dt.
% A 1st order zero phase filter with
% cut-off frequency of fc (fc=0.01).
% Filter is not used when fc=-1.
if(fc~-1)
    
```

```

[b,a]=butter(1,fc);
Y=filtfilt(b,a,Y);
end
[dyp,dyinf,dys,dyu,tu,tp]...
=id_6_param2(Y,R,T);
% Id v and xi
v=(dyinf-dyu)/(dyp-dyinf);
xi=-log(v)/sqrt(pi^2+log(v)^2);
tau=tu*sqrt(1-xi^2)/pi;
K=dyinf/dys;
% The omega parameter
w=sqrt(1-xi^2)/tau;
% Calculate c parameter
work=(K*dys-dyp)/(K*dys);
work=work*exp(xi*tp/tau);
c=(work-cos(w*tp))/sin(w*tp);
tauz=xi*tau-c*tau*sqrt(1-xi^2);
% Back calculate with new formulas
b0=K/(tau^2*Kp);
b1=tauz*b0;
a0=1/tau^2-Kp*b0;
a1=-2*xi/tau+Kp*b1;
a=[1,-a1,a0];
b=[0,b1,b0];
    
```

D. The DR2 alg.

```

function [b,a,tauz,tau,xi,K]...
=DR2(Y,R,T,Kc0,fc)
% ON THE ARGUMENTS:
% Outputs in Y and references in R.
% Kc0 is the proportional gain used
% in the closed loop experiment.
% Sampling time is dt.
% A 1st order zero phase filter with
% cut-off frequency of fc (fc=0.01).
% Filter is not used when fc=-1.
if(fc~-1)
    [b,a]=butter(1,fc);
    Y=filtfilt(b,a,Y);
end
% Simple algorithm for finding the 6
% variables for identification.
[dyp,dyinf,dys,dyu,tu,tp]...
    
```

```
=id_6_param2(Y,R,T);

% estimate xi tau w c K
v=(dyinf-dyu)/(dyp-dyinf);
xi=-log(v)/sqrt(pi^2+(log(v))^2);
tau=tu*sqrt(1-xi^2)/pi;
w=sqrt(1-xi^2)/tau;
c=(xi+tau*w*tan(tp*w))/(tau*w-xi*tan(tp*w));
K=dyinf/dys;
```

```
tauz=xi*tau-c*tau*sqrt(1-xi^2);
```

```
% Back-calculating to the
% open-loop model
Kp=dyinf/(Kc0*abs(dys-dyinf));
```

```
a0=1/(tau^2*(1+Kc0*Kp));
b0=Kp*a0;
b1=K*tauz/(Kc0*tau^2);
a1=-2*xi/tau+Kc0*b1;
```

```
b=[0 b1 b0];
a=[1 -a1 a0];
```

E. The DSR_{ry} method

```
function [num,den]=DSR_ry(Y,R,Kp,dt)
% [a,b,d,e]=dsr_ry(Y,R,Kp)
% ALGORITHM
% 1. Id. closed loop model from r to y,
% i.e. y=h_ry*r
% 2. Back calculate to open loop model
```

```
L=8; J=15; n=2;
[a,b,d,e]=dsr(Y,R,L,0,J,1,n);
```

```
%1) Discrete model info
[bd_dsr,ad_dsr]=ss2tf(a,b,d,e);
% Closed loop model h_ry=hc*hp/(1+hc*hp)
b0_ry=bd_dsr(3); b1_ry=bd_dsr(2);
a0_ry=ad_dsr(3); a1_ry=ad_dsr(2);
```

```
% Back calculate to open loop discrete
%aaa model
```

```
b0=b0_ry/Kp;
b1=b1_ry/Kp;

a0=a0_ry-Kp*b0;
a1=a1_ry-Kp*b1;
```

```
num_d=[0 b1 b0];
den_d=[1 a1 a0];
```

```
%2) Continuous time model info
h_pd=tf(num_d,den_d,dt);
h_pc=d2c(h_pd,'zoh');
[num,den]=tfdata(h_pc,'v');
```

F. Proof of Eq. (9)

The peak at the first extremum $t = t_p$ is given from Eq. (5), i.e.

$$y_p = y(t_p) = y_\infty - y_\infty De^{-\frac{\xi t_p}{\tau}} \sqrt{1 - \xi^2}, \quad (45)$$

where we have used that $Kr = y_\infty$ and that

$$\sin(\omega t_p + \phi) = \sin(\arctan(\frac{\sqrt{1 - \xi^2}}{\xi})) = \sqrt{1 - \xi^2}. \quad (46)$$

This is seen from Fig. 9.

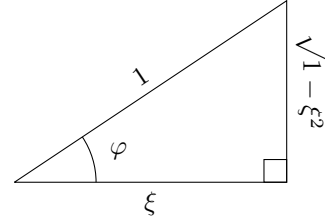


Figure 9: Right triangle.

The peak at the second extremum $t = t_p + t_u$ is given by

$$y_u = y(t_p + t_u) \quad (47)$$

$$\begin{aligned} &= y_\infty - y_\infty De^{-\frac{\xi(t_p+t_u)}{\tau}} \sin(\omega(t_p + t_u) + \phi) \\ &= y_\infty - y_\infty De^{-\frac{\xi(t_p+t_u)}{\tau}} \sin(\arctan(\frac{\sqrt{1 - \xi^2}}{\xi}) + \pi) \\ &= y_\infty + y_\infty De^{-\frac{\xi(t_p+t_u)}{\tau}} \sin(\arctan(\frac{\sqrt{1 - \xi^2}}{\xi})) \\ &= y_\infty + y_\infty De^{-\frac{\xi(t_p+t_u)}{\tau}} \sqrt{1 - \xi^2}, \end{aligned} \quad (48)$$

where we have used that $\sin(\phi + \pi) = -\sin(\phi)$ for an angle ϕ .

From this we evaluate the numerator and denominator in Eq. (10) as

$$y_\infty - y_u = -y_\infty De^{-\frac{\xi(t_p+t_u)}{\tau}} \sqrt{1 - \xi^2}, \quad (49)$$

$$y_p - y_\infty = -y_\infty De^{-\frac{\xi t_p}{\tau}} \sqrt{1 - \xi^2}. \quad (50)$$

Substituting Eqs (49) and (50) into the expression v for the relative overshoot Eq. (10) we obtain

$$v = \frac{y_\infty - y_u}{y_p - y_\infty} = \frac{e^{-\frac{\xi(t_p+t_u)}{\tau}}}{e^{-\frac{\xi t_p}{\tau}}} = e^{-\frac{\xi t_u}{\tau}}. \quad (51)$$

Hence, the relative overshoot v is independent of the term D (Eq. (6)) and hence independent of the system zero t_z .

Using Eq. (15) for t_u we obtain

$$v = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}. \quad (52)$$

From Eq. (52) we have the equation

$$\ln(v) = -\frac{\xi\pi}{\sqrt{1-\xi^2}}. \quad (53)$$

Solving Eq. (53) for ξ^2 we obtain

$$\xi^2 = \frac{\ln^2(v)}{\pi^2 + \ln^2(v)}. \quad (54)$$

Assuming $0 \leq v \leq 1$ we have to use the positive root and hence

$$\xi = \frac{-\ln(v)}{\sqrt{\pi^2 + \ln^2(v)}}, \quad (55)$$

where $0 \leq v \leq 1$ is the relative damping.

References

- Courbot, A. Prevention of Severe Slugging in the Dunbar 16' Multiphase Pipeline. *Proc. Annual Offshore Tech. Conf.*, 1996. doi:[10.4043/8196-MS](https://doi.org/10.4043/8196-MS).
- Di Ruscio, D. Combined Deterministic and Stochastic System Identification and Realization: DSR - A Subspace Approach Based on Observations. *Modeling, Identification and Control*, 1996. 17(3):193–230. doi:[10.4173/mic.1996.3.3](https://doi.org/10.4173/mic.1996.3.3).
- Di Ruscio, D. Closed and Open Loop Subspace System Identification of the Kalman Filter. *Modeling, Identification and Control*, 2009. 30(2):71–86. doi:[10.4173/mic.2009.2.3](https://doi.org/10.4173/mic.2009.2.3).
- Hapoglu, H., Balas, C. E., and Wardle, A. P. Self-tuning and conventional control of an industrial scale packed distillation column. *Tr. J. of Engineering and Environmental Sciences*, 1998. 22:1–8. URL <http://journals.tubitak.gov.tr/engineering/issues/muh-98-22-1/muh-22-1-1-95065.pdf>.
- Haugen, F. The Good Gain method for simple experimental tuning of PI controllers. *Modeling, Identification and Control*, 2012. 33(4):141–152. doi:[10.4173/mic.2012.4.3](https://doi.org/10.4173/mic.2012.4.3).
- Jahanshahi, E. and Skogestad, S. Anti-slug control solutions based on identified model. *Journal of Process Control*, 2015. 30(0):58 – 68. doi:[10.1016/j.jprocont.2014.12.007](https://doi.org/10.1016/j.jprocont.2014.12.007).
- Jutan, A. and Rodriguez, E. Extension of a New Method for On-Line Controller Tuning. *The Canadian Journal of Chemical Engineering*, 1984. 62(6):802–807. doi:[10.1002/cjce.5450620610](https://doi.org/10.1002/cjce.5450620610).
- Ljung, L. *System Identification: Theory for the User*. Pearson Education, 1998. doi:[10.1002/047134608X.W1046](https://doi.org/10.1002/047134608X.W1046).
- Söderström, T. and Stoica, P. *System Identification*. Prentice Hall, 1989.
- Taiwo, O. Comparison of four methods of on-line identification and controller tuning. *IEE Proceedings D - Control Theory and Applications*, 1993. 140(5):323–327. doi:[10.1049/ip-d.1993.0043](https://doi.org/10.1049/ip-d.1993.0043).
- van de Vusse, J. A new model for the stirred tank reactor. *Chemical Engineering Science*, 1962. 17(7):507 – 521. doi:[10.1016/0009-2509\(62\)87002-X](https://doi.org/10.1016/0009-2509(62)87002-X).
- Yuwana, M. and Seborg, D. E. A New Method for Online Controller Tuning. *AIChE Journal*, 1982. 28(3):434–440. doi:[10.1002/aic.690280311](https://doi.org/10.1002/aic.690280311).