Modeling, Identification and Control, Vol. 39, No. 4, 2018, pp. 273–291, ISSN 1890–1328

A Novel Process-Reaction Curve Method for Tuning PID Controllers

Christer Dalen[1] David Di Ruscio[2]

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

[2] *University of South-Eastern Norway, P.O. Box 203, N-3901 Porsgrunn, Norway. E-mail:* David.Di.Ruscio@usn.no

Abstract

A novel process-reaction curve method for tuning PID controllers for (possible) higher order processes/-models is presented. The proposed method is similar to the Ziegler-Nichols process reaction curve method, viz. only the maximum slope and lag need to be identified from an open loop step response. The relative time delay error (relative delay margin), δ is the tuning parameter. The proposed method is verified through extensive numerical simulations and is found close to optimal in many of the motivated process examples. In order to handle the wide set of process models, two model reduction modes are presented.

Keywords: PID control, model approximation, relative time delay error, robustness, performance, optimal, process-reaction curve, process control

1. Introduction

The main focus of this paper and previous work in Dalen and Di Ruscio (2017, 2018) is to approximate step responses from (possible) higher order models/systems with Double Integrating Plus Time Delay (DIPTD) models

$$H_p(s) = K\frac{e^{-\tau s}}{s^2}, \tag{1}$$

such that the (ideal/parallel) PID controllers,

$$H_c(s) = K_p(1 + \frac{1}{T_i s} + T_d s), \tag{2}$$

may be tuned to archive some kind of optimality, e.g. minimising the Integrated Absolute Error (IAE) index. In Eq. (1) K is the gain acceleration and τ is the time delay. In Eq. (2) K_p is the proportional gain, T_i is the integral time constant and T_d is the derivative time constant.

Two of the first and most used PID controller tuning methods are presented in the work of Ziegler (1941); Ziegler and Nichols (1942, 1943), viz. the Ziegler-Nichols (ZN) Process-Reaction Curve (PRC) method which is based on an open loop response, and an ultimate gain method which is based on a closed loop response. We note the ZN PRC PID controller settings as follows: $K_p = \frac{1.2}{R_1 L}$, $T_i = 2L$ and $T_d = \frac{L}{2}$, where, R_1 is the unit reaction rate (maximum slope) and L is known in this paper as Ziegler's lag (see Sec. 4 and Figure 2 for details). Note the statement that identifying process dynamics with only two parameters is insufficient; see Åström and Hägglund (2004). Note, in general, that the ZN PID controllers demonstrate poor robustness, see e.g. Åström and Hägglund (2004). One advantage with this PRC method is that the user does not need to wait for the process to reach steady state, as is usually needed for methods based on e.g. first order plus time delay model approximations.

It may be argued that the contributions of new two-parameter(R_1, L)-PRC methods converged in short time after the ZN method was published. However, a new PRC method was recently published in Dalen and Di Ruscio (2018), denoted δ-PRC, which may be seen

doi:10.4173/mic.2018.4.4

as an extension of the recent δ-PID controller tuning method in Di Ruscio and Dalen (2017), i.e. a possible model reduction step was added such that only an open loop step response (reaction curve) of the model/system was needed. Note that in contrast to ZN, the δ-PRC method offers a tuning parameter for robustness, i.e. the user may prescribe a relative time delay error (relative delay margin), $\delta = \frac{d\tau_{\max}}{\tau}$, where, $d\tau_{\max} > 0$, is the maximum time delay error (delay margin). Note that the prescribed relative time delay errors in the time constant examples of Dalen and Di Ruscio (2018) were seen to be lower and reasonably near the exact maximum time delay errors. Furthermore, the δ-PRC method was found to be sufficiently near the optimal PID controllers for a wide set of motivated time constant models/systems. By optimal we mean Pareto-Optimal (PO), i.e. minimising a Pareto performance objective, originally defined in the paper of Skogestad and Grimholt (2012), and further used in their work Grimholt and Skogestad (2013, 2016a).

The presented method (including the ZN method) may be described as heuristic. By heuristic we mean (hopefully) minimising an objective based on extensive simulations or practical implementations. Heuristic methods may in some circles cause some disfavour, as the method is not built on exact science. The value of the method is determined by the extent to which it is able to fit actual cases. In this work, the objective is the Pareto performance.

Note that the proposed model reduction technique is, in general, much more easy to apply than the half-rule technique proposed in Skogestad/Simple Internal Model Control (SIMC) tuning in the work Skogestad (2001, 2003, 2004), and also the modified half-rule in the Korea/Kyungpook national university-SIMC (K-SIMC) tuning rules presented in Lee et al. (2014). In this paper we will include possible underdamped models. Note that such models are not compatible with SIMC. However, attempts have been documented in the internal report Manum (2005).

The contributions in this paper may be itemised as follows.

- The δ-PRC method proposed in Dalen and Di Ruscio (2018) is further developed and proven on motivated process model examples.
- Model reduction modes are introduced.
- The δ-PRC method is compared to the model-based tuning methods, SIMC and K-SIMC on motivated time constant models.
- The δ-PRC method is compared to the heuristic optimisation tuning method in Dalen and Di Ruscio (2018) on motivated process models containing complex poles. The ZN PRC PID controller tuning method is also included in this comparison.
- A possible ζ-PRC tuning variant is demonstrated. In this variant the main tuning parameter is present in the model reduction step, i.e. the gain acceleration is proportionally varied.

All numerical calculations and plotting facilities are provided by using the MATLAB software, MATLAB (2016). The rest of this paper is organised as follows. In Sec. 2 the preliminary definitions are given. In Sec. 3 the PO PID controller is presented. The δ-PRC method including the model reduction modes are presented in Sec. 4. The numerical results are presented for a wide range of examples in Sec. 5. Lastly, the discussion and concluding remarks are given in Sec. 6.

2. Preliminary Definitions

Definition 2.1 (System)
The underlying systems/models are assumed to be described by the following transfer function form,

$$H_p(s) = \frac{k(T_z s + 1)}{\prod_{j=1}^{n}(T_j s + 1)(\tau_0 s^2 + 2\tau_0 \xi s + 1)}, \tag{3}$$

where, $n > 1$, the gain, $k \neq 0$, time constant T_z, time constants, $T_1 \geq T_2 \geq \ldots \geq T_j \geq 0$, the "speed" of response, $0 \leq \tau_0$, and the relative damping, ξ in the range, $0 < \xi < 1$. In Eq. (3) we assume that T_j and τ_0 are not both zero at the same time. Assuming only deterministic systems/models.

In the case of a pure single time constant process, we obtain an on-off controller, which is not a part of the topic of this paper, hence the reason for setting $n > 1$ if $\tau_0 = 0$.

One motivation for approximating models as in Eq. (3) with DIPTD models (Eq. (1)) is that for a short time interval the approximation is fairly good, which is illustrated in Figure 2. However the main motivation is that it gives close to PO PID controllers which is documented in this and previous work in Dalen and Di Ruscio (2018). In this paper we are only interested in tuning PID controllers (for models as Eq. (3)) based on DIPTD models (Di Ruscio and Dalen (2017)).

Note that proper system identification methods are recommended when including noise, e.g. Ljung (1999), DSR in Di Ruscio (1996) and DSR_e in Di Ruscio (2008, 2009).

Consider the standard feedback system with disturbances as illustrated in Figure 1. In order to compare the different controllers against each other we will consider indices such as defined in Åström and Hägglund

(1995), Seborg et al. (1989) and Skogestad (2003). We define these in the following.

Definition 2.2 (Performance)
For measuring performance in a feedback system, the IAE is defined as

$$IAE = \int_0^\infty |\, e(t) \,|\, dt, \tag{4}$$

where, $e = r - y$, is the control deviation error and r is the reference.
Furthermore, the following is defined:

- *IAE_{vu} evaluates the performance in the case of a step input disturbance ($H_v(s) = H_p(s)$), $v = 1$, with the reference, $r = 0$.*
- *IAE_{vy} evaluates the performance in the case of a step output disturbance ($H_v(s) = 1$), $v = 1$, with the reference, $r = 0$.*

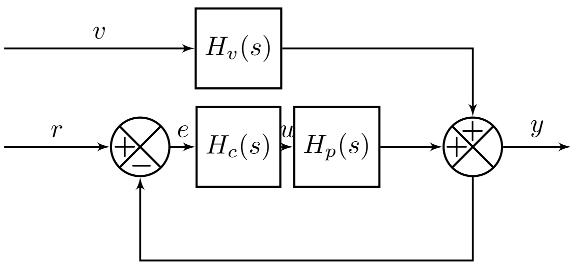

Figure 1: Control feedback system. The plant model is described by the process model $H_p(s)$ (Eq. (3)), PID controller, $H_c(s)$, (Eq. (2)) and disturbance model, $H_v(s)$, where, step disturbance, v, at the input when $H_v(s) = H_p(s)$ and at the output when $H_v(s) = 1$.

Robustness (i.e. allowing for inaccuracies in the acceleration gain and time delay in the DIPTD model in Eq. (1)) may be quantified in various ways, and in this work we define it according to Garpinger and Hägglund (2008).

Definition 2.3 (Robustness)
Robustness is defined by the sensitivity peak,

$$M_s = \max_{0 \leq \omega < \infty} |S(j\omega)| = ||S(j\omega)||_\infty, \tag{5}$$

where, $S(j\omega) = \frac{1}{1 + H_p(j\omega)H_c(j\omega)}$, and, $||\cdot||_\infty$, is the $\mathcal{H}_\infty$-norm.

For robust controllers we consider the interval $1.4 \leq M_s \leq 2.0$ (Åström and Hägglund (2006)).

To evaluate the amount of input usage we include the following measure.

Definition 2.4 (Input Usage)
Input usage is defined as Total input Value (TV)

$$TV = \int_0^\infty |\, \Delta u_k \,|\, dt, \tag{6}$$

where, $\Delta u_k = u_k - u_{k-1}$, is the control rate of change.

3. Pareto-Optimal PID Controller

For quantifying multiple performances, i.e. indices IAE_{vu}, and, IAE_{vy}, we define the following Pareto performance objective,

$$J(p) = s_r \frac{IAE_{vy}(p)}{IAE_{vy}^o} + (1 - s_r) \frac{IAE_{vu}(p)}{IAE_{vu}^o}, \tag{7}$$

where s_r is the servo-regulator parameter chosen in the range $0 \leq s_r \leq 1$ (originally introduced in Di Ruscio (2012)) for trade-off weighting between the output disturbance (servo) weighting $s_r = 1$ and input disturbance (regulator) weighting $s_r = 0$. In this work, and as in earlier papers, we will set $s_r = 0.5$ (Skogestad and Grimholt (2012)). The controller arguments are structured as $p = [K_p, T_i, T_d]^T$. $IAE_{vy}^o = \min_p IAE_{vy}(p, M_s)$ and $IAE_{vu}^o = \min_p IAE_{vu}(p, M_s)$, are the optimal output and input disturbance indices, i.o, where $M_s = 1.59$. See Table 1 for details of the reference controllers for Example 1 (E1). Note that for robust reference PID controllers we generally want $M_s = 1.59$ which corresponds to a SIMC tuned PI controller for the process model $H_p(s) = \frac{1}{s+1}e^{-s}$ (Grimholt and Skogestad (2013)).

Table 1: E1. The table shows the optimal input and output disturbance controllers for prescribed robustness, $M_s = 1.59$. * means that this value is not important and is not given.

K_p	T_i	T_d	IAE$_{vy}$	IAE$_{vu}$	M_s
12.74	1.189	0.202	0.0995	*	1.59
13.37	0.151	0.168	*	0.0229	1.59

The following main performance objective is defined in a mean square error sense,

$$V_M(x) = \frac{1}{M} \sum_{i=1}^{M} (J_x(i) - J_{PO}(i))^2, \tag{8}$$

where x is a tuning method and $M = \mathbf{length}(J)$.

4. δ-PRC Controller Tuning

The δ-PRC PID controller tuning method is defined as Algorithm 2.1 in Dalen and Di Ruscio (2018), where steps 1-2 are substituted with the following DIPTD model (Eq. (1)) approximation rules for gain acceleration and time delay,

$$K = \zeta \frac{R_1}{L}, \tag{9}$$

$$\tau = \eta L, \tag{10}$$

where $R = \max / \min_t \frac{dy}{dt}$ (i.e. min if $y(t_{final}) < 0$), is the reaction rate (maximum slope), $R_1 = \frac{R}{\Delta u}$, is the unit reaction rate, $\Delta u = 1$ (default), is the input step change, $L = t_1 - \frac{y_1}{R_1}$, is defined as Ziegler's lag, $t_1 = \arg R_1$, and, $y_1 = y(t_1)$. See Figure 2 for an illustration of the model reduction technique.

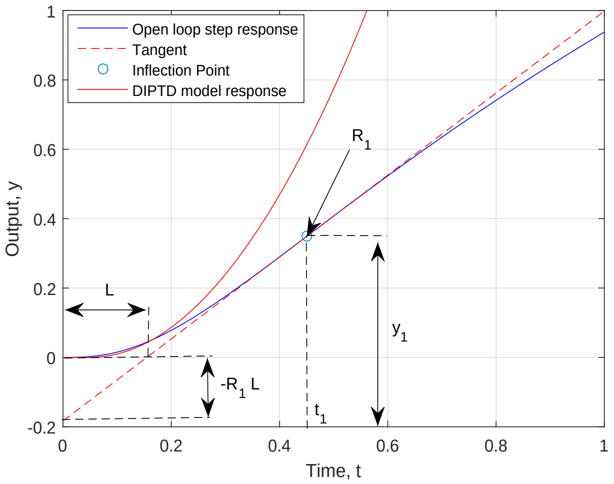

Figure 2: E1. The figure illustrates the model reduction technique. Shows the open loop step response of the higher order process model given in column 2 in Table 2. Shows the step response of the DIPTD model, $H_p(s) = K\frac{e^{-\tau s}}{s^2}$, approximation where the gain acceleration $K = \frac{R_1}{L}$ and time delay $\tau = \frac{L}{2\pi}$ (Mode 1). R_1 is the unit reaction rate and L is Ziegler's lag and (t_1, y_1) is the inflection point.

Based on extensive simulations in this and previous work (Dalen and Di Ruscio (2018)), and since we usually encounter relatively low order models, we recommend choosing ζ in the range $0 < \zeta \leq 10$. Furthermore, the main performance objective V_M (Eq. (8)) is observed to be relatively insensitive to small changes around $\eta = \frac{1}{2\pi}$ (holds at least for pure multiple pole models), hence we propose to keep this constant as in Dalen and Di Ruscio (2018). We take a shot at covering a broad set of possible models/systems and, at the same time, make the method practical for the user. We present a couple of model reduction modes in the following.

$$\left\{ \begin{array}{lcll} \text{Mode 1} & : & \zeta = 1, & \eta = \frac{1}{2\pi} \\ \text{Mode 2} & : & \zeta = 6, & \eta = \frac{1}{2\pi} \end{array} \right\} \tag{11}$$

Mode 1 corresponds to the method given in Dalen and Di Ruscio (2018) where a possible proof was given therein. Note that mode 1 works well for most processes (satisfying Eq. (3)), however, arguably, not for processes where the time constants are equal or approximately equal with order $n > 3$. For such processes we would suggest mode 2. It would be useful to have some information on the process before assigning a mode, but this is not necessary. Given that we only have two modes, the user may perform a trial-and-error approach. However, one might find the appropriate mode by fixing the main tuning parameter, δ, (e.g. $\delta = 2.12$) and observing the closed loop response, input step response, or by calculating the M_s (Eq. (5)) directly, subject to changing between modes.

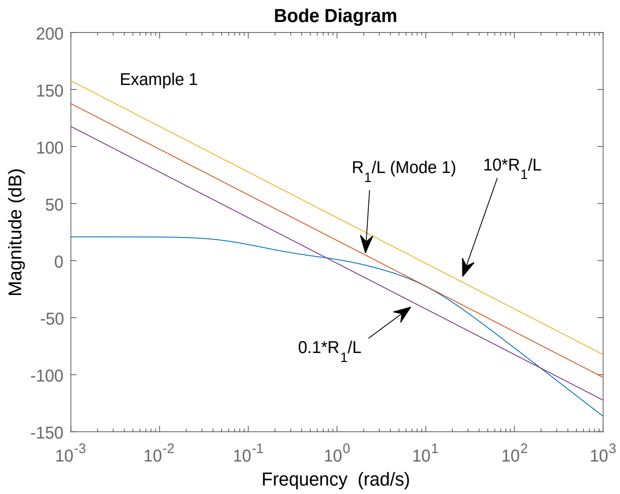

Figure 3: E1. The figure illustrates the model reduction technique. Shows the magnitude responses for the higher order process model in column 2 in Table 2 and the DIPTD model, $H_p(s) = K\frac{e^{-\tau s}}{s^2}$, approximation where the gain acceleration $K = \zeta\frac{R_1}{L}$ varies, i.e. $\zeta \in \{0.1, 1, 10\}$.

Some comments regarding optimal method product settings, $\bar{c}$, and integral-derivative ratio, γ, are given for step 3 in Algorithm 2.1 (Dalen and Di Ruscio (2018)), i.e. δ-PID controller tuning. Consider the

DIPTD model (Eq. (1)) where the gain acceleration $K = 1$ and the time delay $\tau = 1$. In the incoming we define Alg. 3.1 as Alg. 2.1 and Eq. (27) in Di Ruscio and Dalen (2017).

The first setting is obtained by solving the following optimisation problem

$$\begin{aligned}\begin{bmatrix}\bar{c}\\ \gamma\end{bmatrix} &= \arg\min_{\bar{c},\gamma} V_M(\text{Alg. } 3.1^o, \text{Alg. } 3.1\,(\bar{c},\gamma)) \\ &= \begin{bmatrix}2.24\\ 2.24\end{bmatrix}, \end{aligned} \tag{12}$$

where Alg. 3.1 $(\bar{c}, \gamma, \delta_i)$ and Alg. 3.1^o (δ_i) are pre-calculated as follows

$$J^i_{\text{Alg. }3.1} = \min_{\bar{c},\gamma} J_{\text{Alg. }3.1}\,(\bar{c},\gamma,\delta_i)\ \forall\ 1.1 \leq \delta_i \leq 3.4. \tag{13}$$

Notice that $\gamma = 2.25$ was found to be optimal in Kristiansson and Lennartson (2006).

The second setting which is used in this work and was originally proposed in Dalen and Di Ruscio (2017), is found by

$$\begin{aligned}\begin{bmatrix}\bar{c}\\ \gamma\end{bmatrix} &= \arg\min_{\bar{c},\gamma} V_M(\text{PO}, \text{Alg. } 3.1\,(\bar{c},\gamma)) \\ &= \begin{bmatrix}2.12\\ 2.12\end{bmatrix}, \end{aligned} \tag{14}$$

where Alg. 3.1 $(\bar{c}, \gamma, \delta\,(M_s^i))$ and PO (M_s^i) are pre-calculated as follows

$$J^i_{PO} = \min_p J(p, M_s^i)\ \forall\ 1.3 \leq M_s^i \leq 2.0. \tag{15}$$

Notice that the conventional ratio, $\gamma = 4$, is larger than the couple presented above, see e.g. Ziegler and Nichols (1942), Astrom and Hagglund (1984), Mantz and J. Tacconi (1989) and Skogestad (2003).

The algorithm for the δ-PRC method is presented as follows.

Algorithm 4.1 (δ-PRC PID Controller Tuning)

1. *Find Ziegler's lag, L, and the unit reaction rate, R_1 based on the open loop step response (reaction curve) of the (possible) higher order model/system.*
2. *Choose one of the two model reduction modes (proposed in Eq. (11)) based on trial-and-error. Find the gain acceleration, K, and time delay, τ, in the DIPTD model, using Eqs. (9) and (10).*
3. *Obtain the PID controller parameters K_p, T_d and T_i by using δ-PID controller tuning, viz. Alg. 2.1 and Eq. (27) in Di Ruscio and Dalen (2017), i.o.*

The above method is implemented in a MATLAB m-file function shown in App. A.

5. Numerical Results

A set of PO PID controllers is obtained for each process model example (Es1-12) using the exact gradient optimisation method in Grimholt and Skogestad (2016b).

Note that the PID controller tuning methods of SIMC and K-SIMC are based on second order plus time delay models. Furthermore the SIMC and K-SIMC tuned PID controllers are on cascade form, hence they need to be converted to the ideal/parallel form.

For Es1-6, the δ-PRC tuning method is compared to SIMC and K-SIMC in terms of trade-off curves shown in Figure 4, where the corresponding V_M measures are given in Table 3. The corresponding time-domain output and input step responses and input usage, for a prescribed robustness, $M_s = 1.59$, are illustrated in Figures 7 and 8, i.o.

For Es7-12 (i.e. complex pole examples), the δ-PRC tuning method is compared to the Opt-PRC method (Dalen and Di Ruscio (2018)) in terms of trade-off curves shown in Figure 5, where the corresponding V_M measures are given in Table 4. Notice that the ZN PRC PID controller tuning is included as a point in the trade-off plots. We present Figure 6 which is 'zoomed out' version of Figure 5. The corresponding time-domain output and input step responses and input usage, for a prescribed robustness, $M_s = 1.59$, are illustrated in Figures 9 and 10, i.o. See Table 5 for the choice of model reduction modes in the δ-PRC method for Es1-12. The prescribed PID controller parameters, including the Pareto performances J and the margins, are shown in App. B in Tables 6 and 7.

Note that other simulation examples demonstrating the performance of δ-PRC on pure time constant processes are documented in Dalen and Di Ruscio (2018).

Table 2: Shows the test batch, i.e. the motivated process model examples (Es1-12) used in the numerical simulations.

E	Process model, $H_p(s)$
1	Eq. (7) in Åström and Hägglund (2000) (similar to Eq. (13) in Skogestad (2003)) $\frac{11(2.727s+1)}{(20s+1)(s+1)(0.1s+1)^2}$
2	Eq. (2) in Åström and Hägglund (2000) where $\alpha = 0.9$ $\frac{1}{(s+1)(0.9s+1)(0.81s+1)(0.729s+1)}$
3	Eq. (2) in Åström and Hägglund (2000) where $\alpha = 0.3$ $\frac{1}{(s+1)(0.3s+1)(0.09s+1)(0.027s+1)}$
4	Eq. (4) in Åström and Hägglund (2000) Skogestad (2003) $\frac{1}{(s+1)^4}$
5	Daraz et al. (2017) Superheated Steam Temperature $\frac{0.7732}{(19s+1)^5}$
6	Eq. (30) in S. J. Sadati and Ghaderi (2012) Oxygentator (neglecting time delay) $\frac{2.963e+5}{(66.67s+1)^3}$
7	Åström et al. (1998) E30 in Shamsuzzoha (2013) $\frac{1}{(s+1)((0.333s)^2+0.667(0.333s)+1)}$
8	S. Sai Tarun (2014) Single Area Power System $\frac{2.3529}{(0.07524s+1)((0.3537s)^2+0.9171(0.3537s)+1)}$
9	Eq. (3) in Salloum et al. (2014) ElectroMechanical Actuators $\frac{1}{(0.0071s+1)((0.0084s)^2+1.662(0.0084s)+1)}$
10	Eq. (12) in Wang et al. (2017) Hydraulic Support Electro-Hydraulic System $\frac{2.6649(-0.02784s+1)}{(0.9464s)^2+1.197(0.9464s)+1}$
11	Eq. (10) in Farouk et al. (2012) (neglecting time delay) Marine Diesel Engine $\frac{1}{(2.403s+1)(0.237s+1)((0.028s)^2+1.414(0.028s)+1)}$
12	Eq. (1) in Abbasi et al. (2017) Unmanned Free Swimming Submersible Vehicle $\frac{-2.6158(2.299s+1)}{(0.8131s+1)(0.5s+1)((7.692s)^2+1.738(7.692s)+1)}$

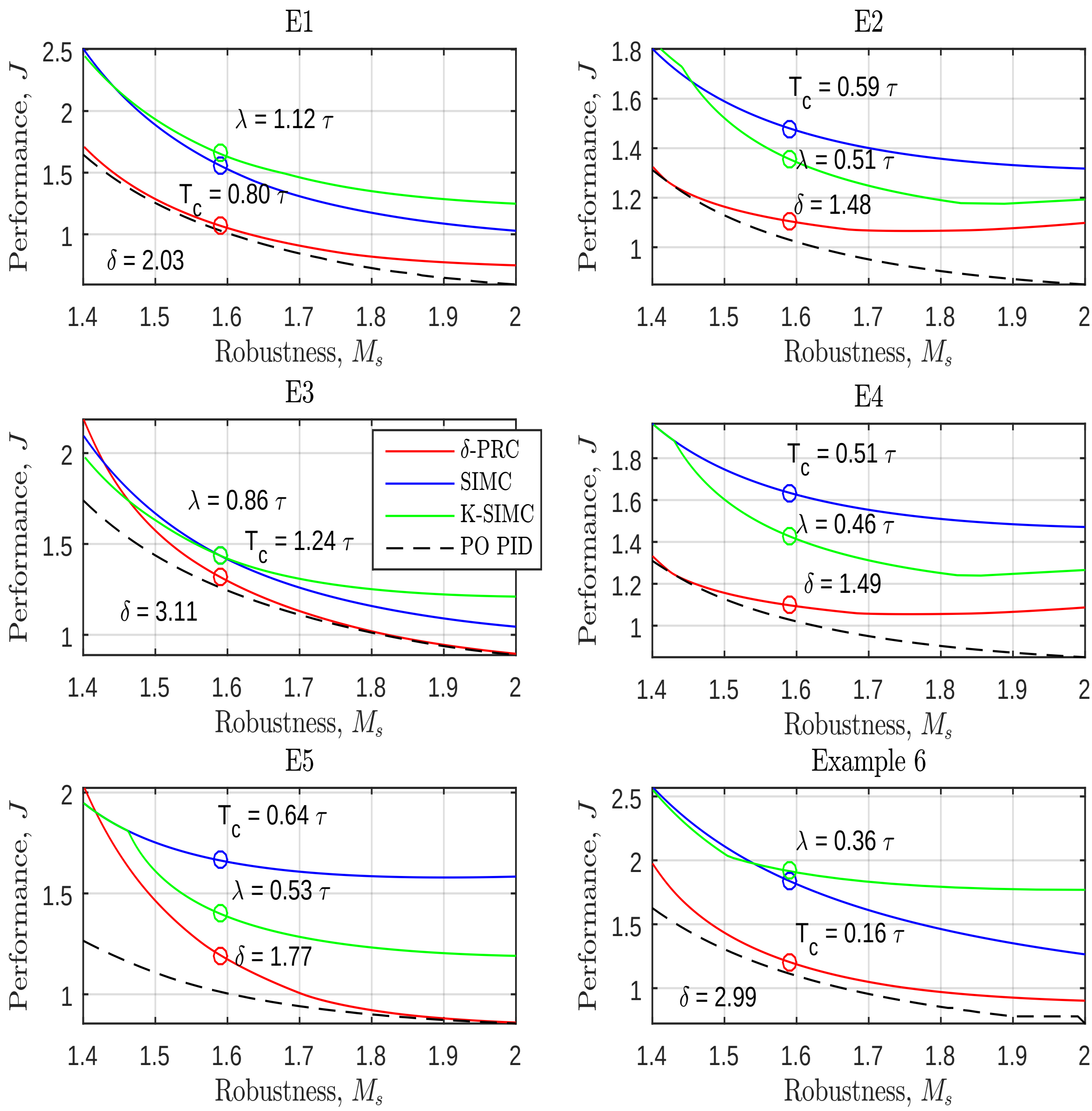

Figure 4: Es1-6. The figure shows the Pareto performance J (Eq. (7)) and M_s (Eq. (5)) trade-off curves for the methods δ-PRC (δ), SIMC (T_c), K-SIMC (λ) and PO PID (M_s), where δ is the prescribed relative time delay error, T_c and λ are the prescribed set point response time constants. The circles illustrates controllers with prescribed robustness $M_s = 1.59$.

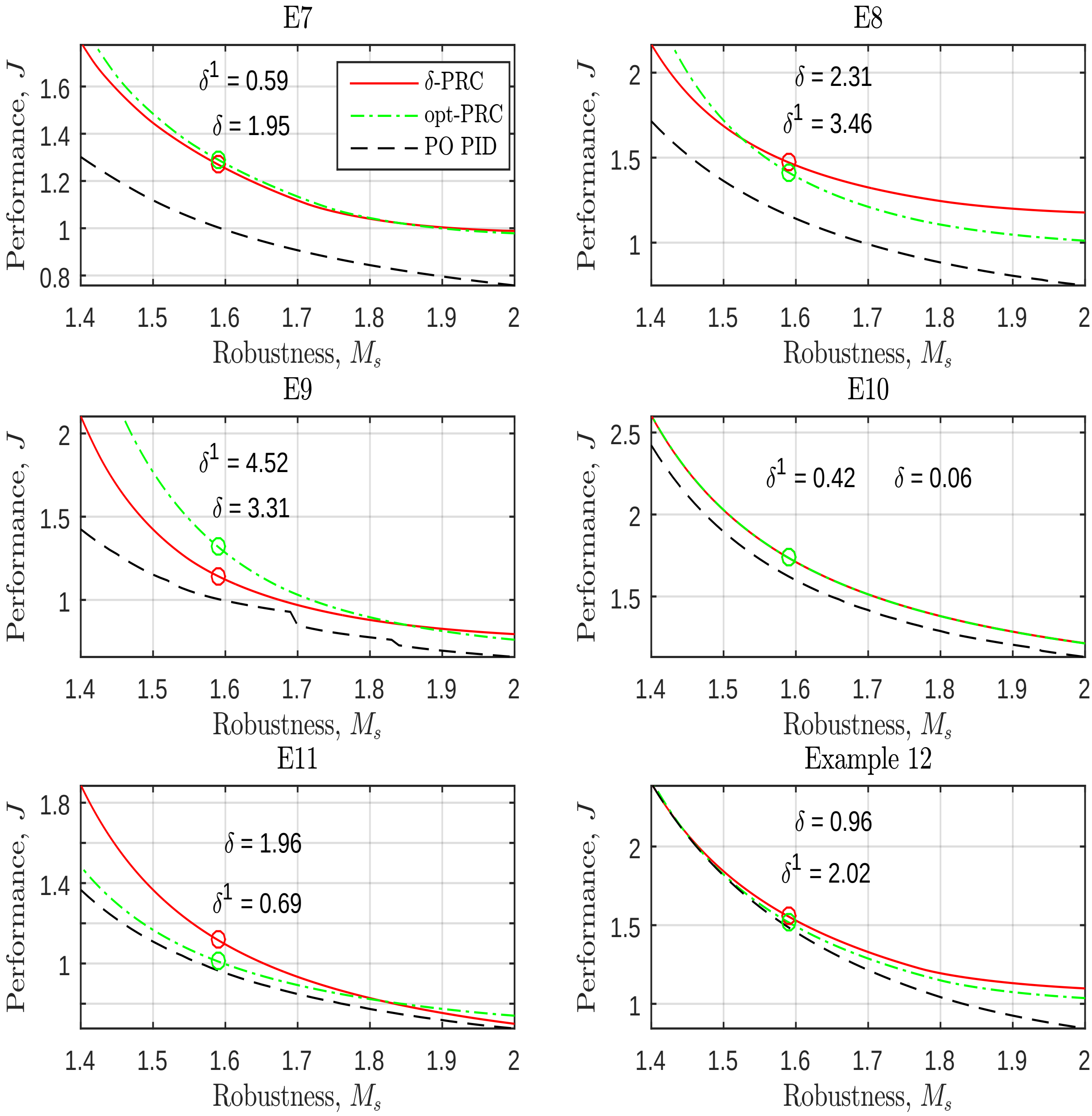

Figure 5: Es7-12. The figure shows the Pareto performance J (Eq. (7)) and robustness M_s (Eq. (5)) trade-off curves for the methods δ-PRC (δ), Opt-PRC (δ^1) and PO PID (M_s), where δ and δ^1 are the prescribed relative time delay errors. The circles illustrates controllers with prescribed robustness $M_s = 1.59$.

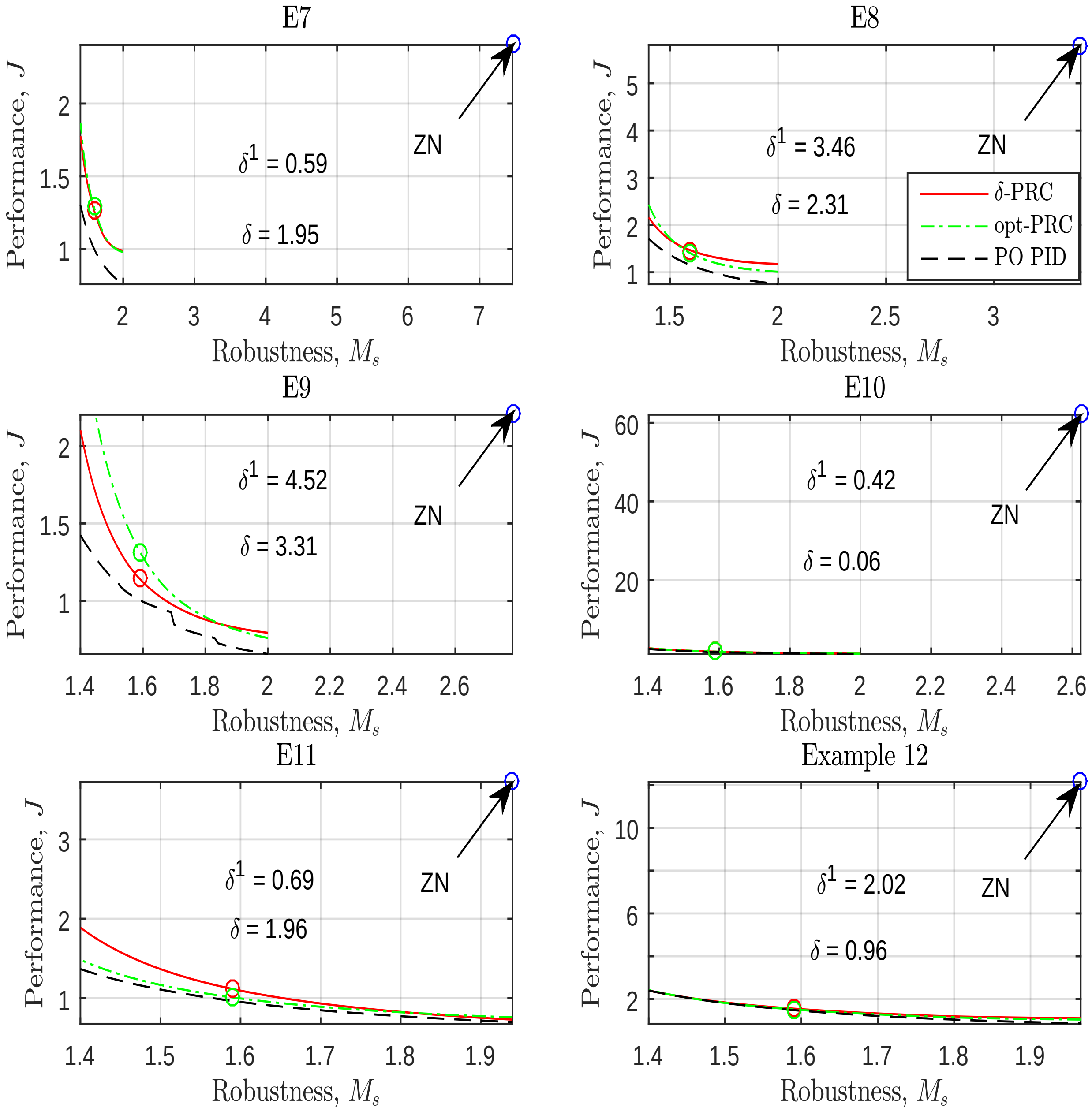

Figure 6: Es7-12. The figure shows the Pareto performance J (Eq. (7)) and robustness M_s (Eq. (5)) trade-off curves for the methods δ-PRC (δ), Opt-PRC (δ^1) and PO PID (M_s). This figure illustrates the ZN PRC PID tuning as a single point per example. This figure is a 'zoomed out' version of Figure 5.

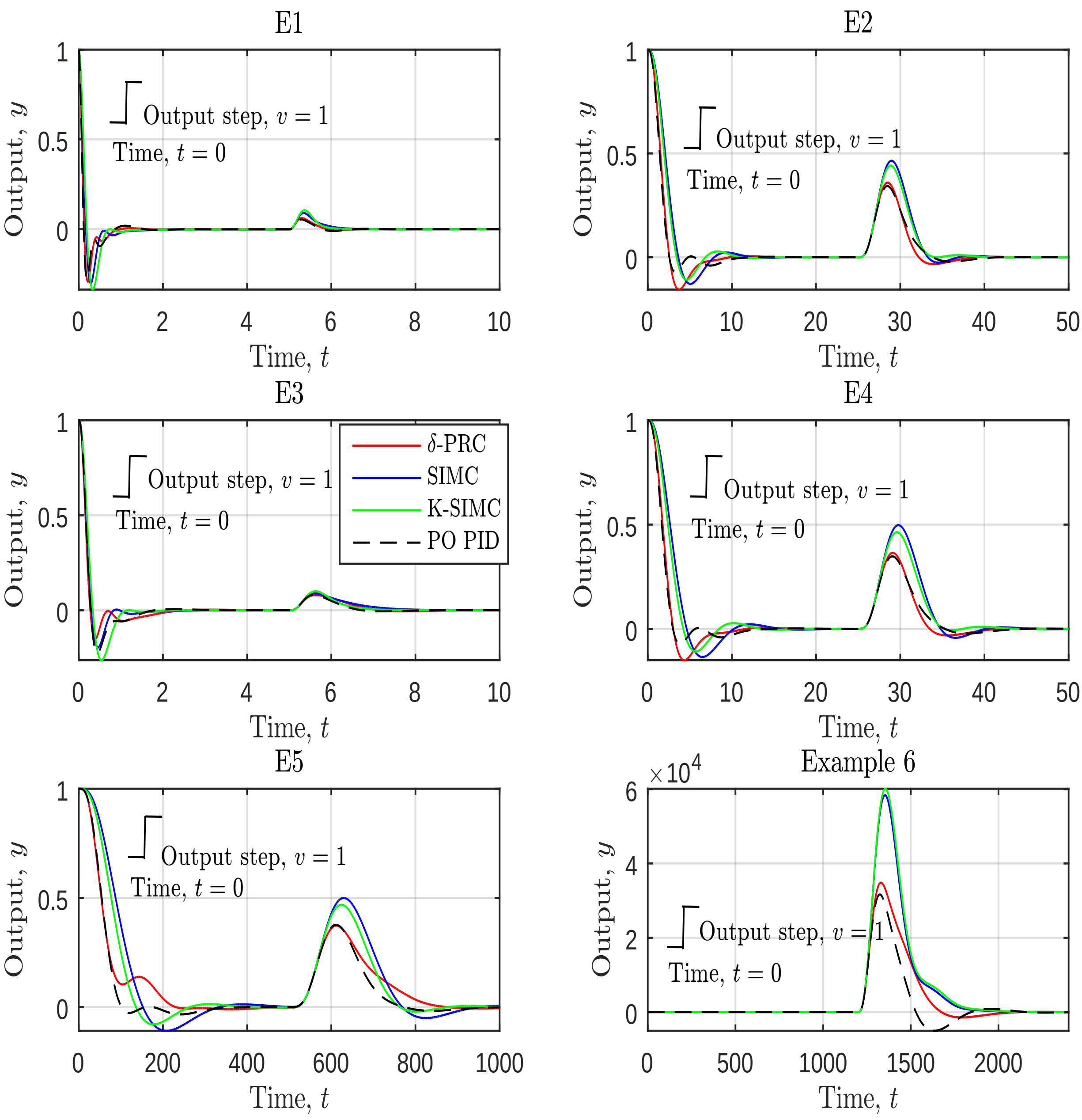

Figure 7: Es1-6. The figure illustrates the output and input step time-domain responses for a prescribed robustness, $M_s = 1.59$, for the following methods: δ-PRC, SIMC, K-SIMC and PO PID.

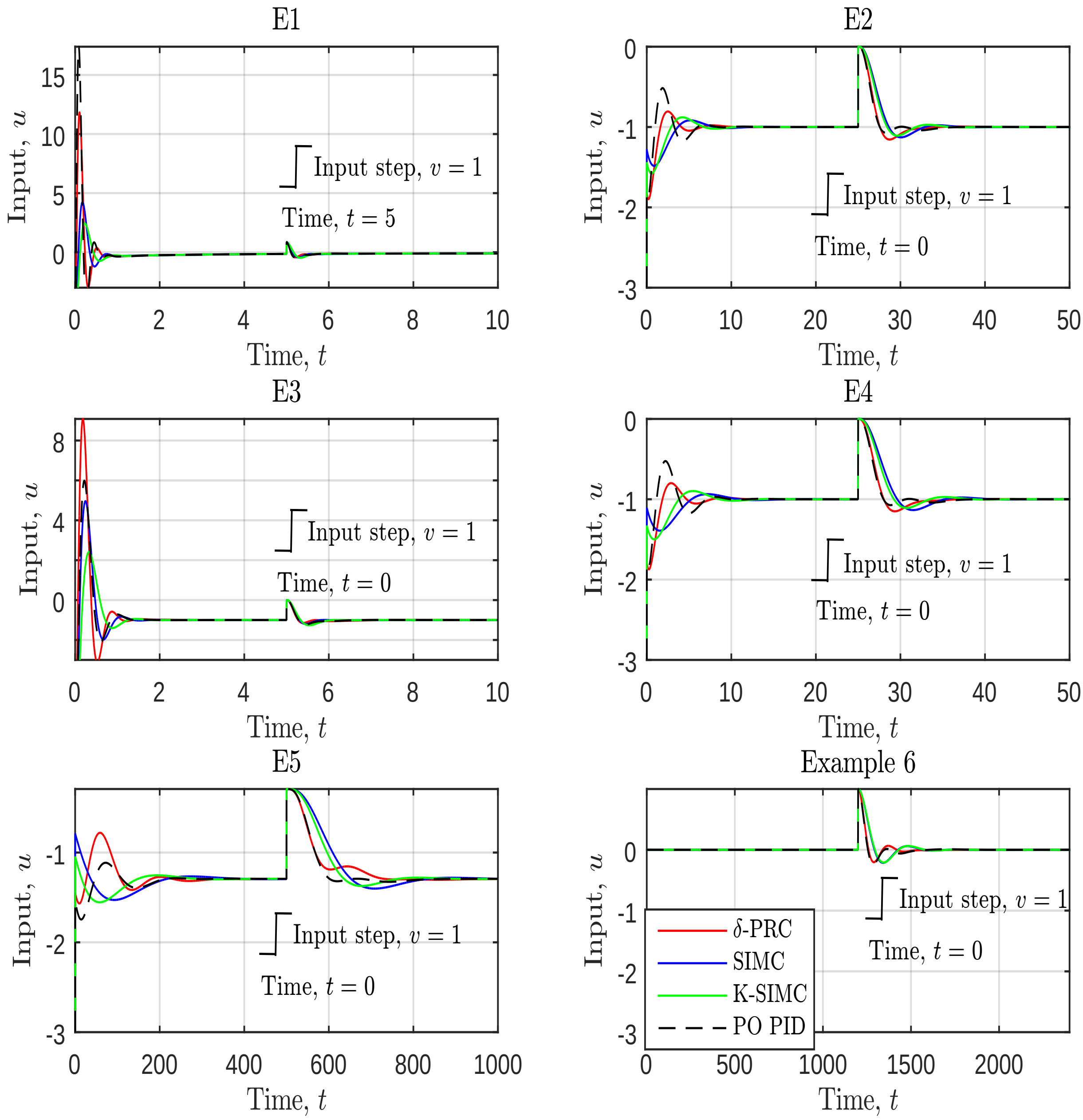

Figure 8: Es1-6. The figure illustrates the input usage for a prescribed robustness, $M_s = 1.59$, for the following methods: δ-PRC, SIMC, K-SIMC and PO PID.

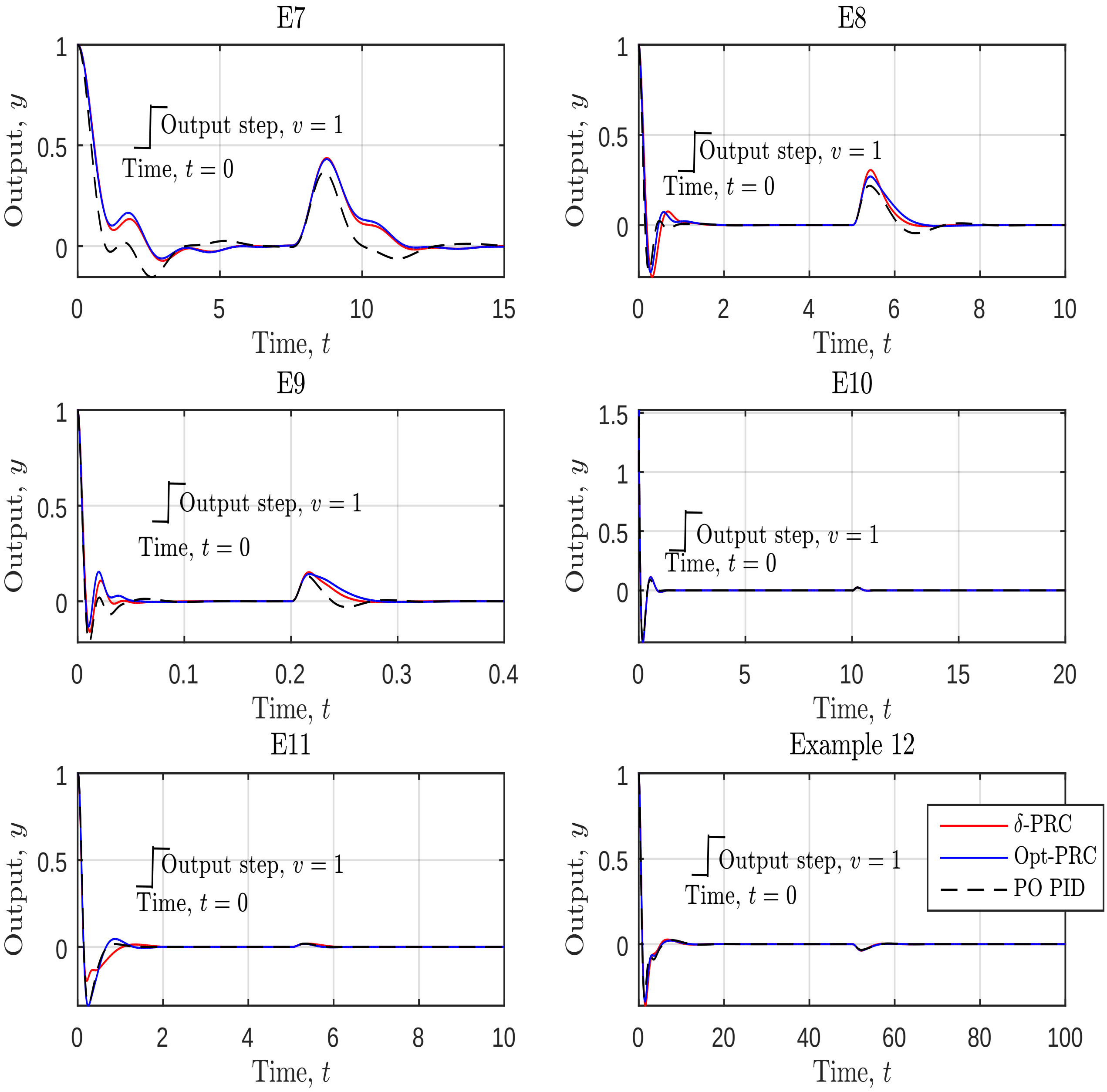

Figure 9: Es7-12. The figure illustrates the output and input step time-domain responses for a prescribed robustness, $M_s = 1.59$, for the following methods: δ-PRC, SIMC, K-SIMC and PO PID.

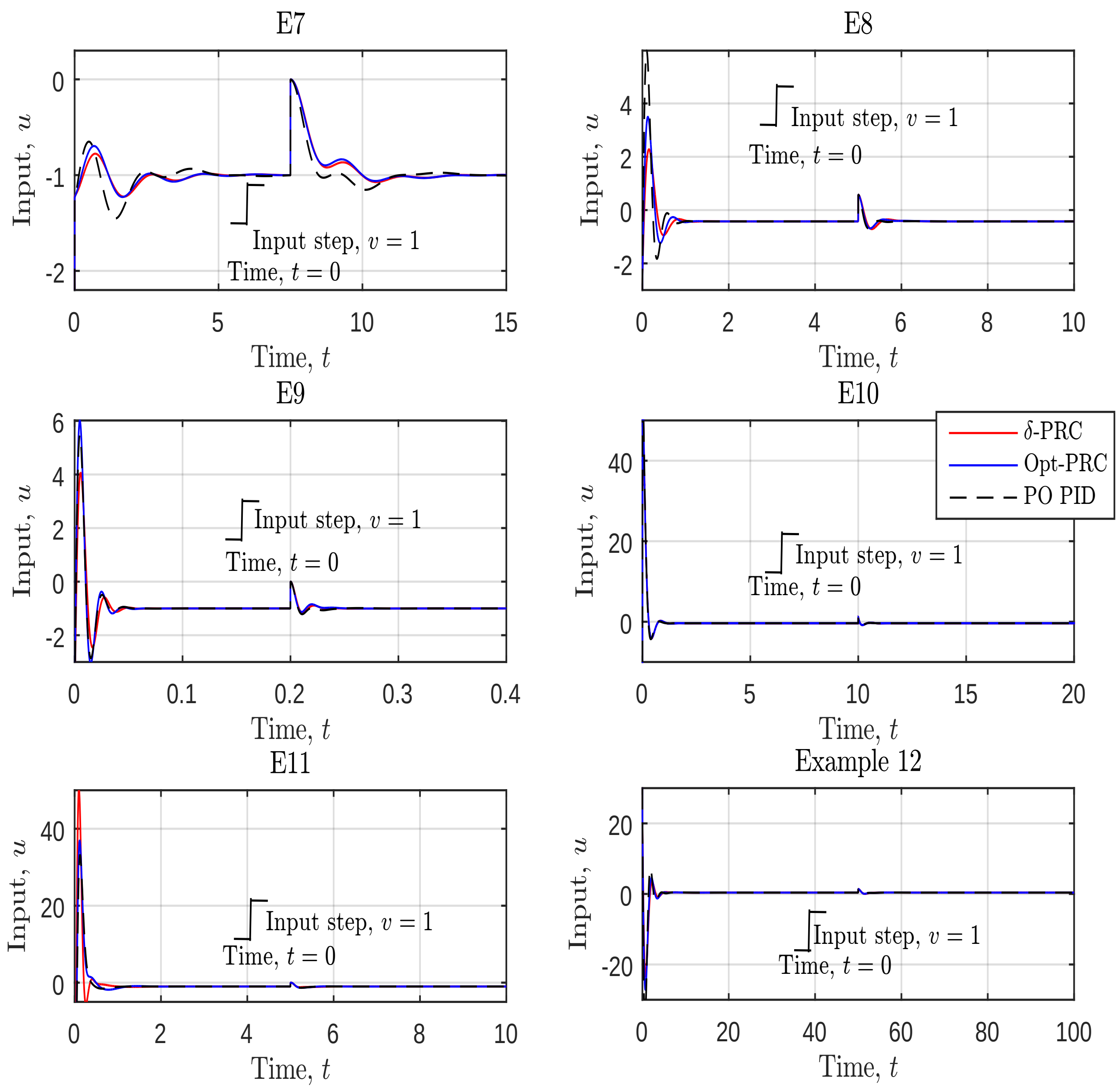

Figure 10: Es7-12. The figure illustrates the input usage for a prescribed robustness, $M_s = 1.59$, for the following methods: δ-PRC, SIMC, K-SIMC and PO PID.

Table 3: Es1-6. The table shows the main performance objective V_M (Eq. (8)) measures for the δ-PRC, K-SIMC and SIMC tuned PID controllers, i.e. corresponding to the trade-off curves in Figure 4.

E \ V_M	δ-PRC	SIMC	K-SIMC
1	0.005	0.354	0.449
2	0.012	0.212	0.157
3	0.029	0.048	0.047
4	0.010	0.382	0.237
5	0.123	0.443	0.247
6	0.025	0.566	0.709

Table 4: Es7-12. The table shows the main performance objective V_M (Eq. (8)) measures for the δ-PRC and Opt-PRC tuned PID controllers, i.e. corresponding to the trade-off curves in Figure 5.

E \ V_M	δ-PRC	Opt-PRC
7	0.091	0.115
8	0.126	0.134
9	0.086	0.391
10	0.015	0.016
11	0.061	0.004
12	0.012	0.005

6. Discussion and Concluding Remarks

The discussion and concluding remarks are itemised as follows.

- The δ-PRC method in Dalen and Di Ruscio (2018) is further developed and proven. Model reduction modes have been presented. These are demonstrated through Es1-12.
- It is seen in Es1-6 that the proposed method has an edge over the other model-based methods SIMC and K-SIMC, viz. δ-PRC is seen to be at the minimum $\frac{V_{\text{K-SIMC}}}{V_{\delta-\text{PRC}}} = 1.6$ times better than the runner-up method (K-SIMC), and at the maximum, $\frac{V_{\text{SIMC}}}{V_{\delta-\text{PRC}}} = 70.1$ times better than the runner-up method (SIMC). See Table 3.
- For Es7-12 the δ-PRC wins 4 out of 6 examples (wrt. Table 4), however Opt-PRC (Dalen and Di Ruscio (2018)) is 15 times better on E11.
- These simple heuristic modes give PID controller tuning rules which are close to optimal (PO), i.e. approximately minimising the Pareto performance objective (Eq. (7)) in many cases.
- Notice that the results in Sec. 5 are based on the (possible) higher order models in Table 2. The DIPTD model (Eq. (1)) approximation is only used for PID controller design.
- The ZN PRC PID controller tuning method is not robust, as demonstrated in Figure 6. It also illustrates the lack of performance. The worst cases show a Pareto performance $J > 60$ in E10 and a robustness $M_s > 7$ on E7.
- Some surprisingly optimal results were documented in App. C, where a tuning method based on varying the gain velocity, $K = \zeta \frac{R_1}{L}$, i.e. the tuning parameter is ζ. Note that the setting $\delta = \bar{c}$ (i.e. an ad hoc choice) equal a constant is advisable.

Table 5: The table shows the chosen model reduction modes (Eq. (11)) in the δ-PRC method for the examples Es1-12.

E	Mode
1	1
2	2
3	1
4	2
5	2
6	1
7	2
8	1
9	1
10	1
11	1
12	1

A. δ-PRC method MATLAB m-file

```
function [Kp,Ti,Td]=delta_prc_pid_tun ....
(T,Y,delta,imod,du)
% PURPOSE. Tuning an ideal PID controller
% hc(s)=Kp(1+1/(Ti*s)+Td*s)
% based on input step response data.
% [Kp,Ti,Td]=delta_prc_pid_tun ...
% (T,Y,delta,imod,du)
%
% On Input
% T, Y  - Step response data
%          T time vector
%          Y output vector
%
% delta  - Tuning parameter,
```

```
%           relative time delay error
%           delta=2.12 (default)
%
% imod - Model reduction mode
%          Choose imod=1 (default),
%          or imod=2
%
% du   - Input step change
%          Unit du=1 (default)
%
% On output
% Kp   - Proportional constant
% Ti   - Integral time constant
% Td   - Derivative time constant

if nargin == 4; du=1; end
if nargin == 3; du=1; imod=1; end
if nargin == 2; du=1; imod=1; delta=2.12;
end

eta=1/(2*pi);

if imod==2
 zeta=6;
else
 zeta=1;
end

h=T(2)-T(1); A=diff(Y)/h;

if Y(end)<0
  [R,i]=min(A);
else
  [R,i]=max(A);
end

R1=R/du;
t1=T(i);

y1=Y(i); L=t1-y1/R1;

K=zeta*R1/L; tau=eta*L;

cb=2.12; gamm=2.12;
[Kp,Td]=pd_tun_maxdelay(K,tau,delta,cb,1);
Ti=gamm*Td;

% end delta_prc_pid_tun.m
```

B. Margins

Table 6: Es1-6. The table contains the PID controller parameters for fixed robustness $M_s = 1.59$ for the methods; δ-PRC, SIMC, K-SIMC and PO PID. Also include the Pareto performance J and the input usage TV. The following margins are included: Gain Margin (GM), Phase Margin (PM) and Delay Margin (DM).

E	Method	K_p	T_i	T_d	IAE_{vy}	IAE_{vu}	J	TV	GM	PM	DM	M_s
1	δ-PRC	12.62	0.31	0.15	0.15	0.03	1.07	1919	∞	46.33	0.06	1.59
1	SIMC	10.36	0.51	0.11	0.18	0.05	1.53	1124	∞	44.48	0.08	1.59
1	K-SIMC	8.83	0.43	0.10	0.21	0.05	1.64	883	∞	42.86	0.09	1.59
1	PO PID	12.68	0.23	0.18	0.15	0.03	1.03	2346	∞	47.10	0.05	1.59
2	δ-PRC	1.81	2.02	0.95	1.93	1.36	1.10	1734	4.50	54.53	1.39	1.59
2	SIMC	1.28	2.31	0.57	2.47	1.92	1.48	727	5.04	55.31	1.85	1.59
2	K-SIMC	1.44	2.55	0.61	2.24	1.78	1.35	878	4.85	57.52	1.79	1.59
2	PO PID	1.85	2.25	1.14	1.64	1.40	1.03	2115	4.33	65.46	1.62	1.59
3	δ-PRC	8.59	0.69	0.32	0.24	0.09	1.32	2819	6.01	55.00	0.14	1.59
3	SIMC	9.43	0.99	0.23	0.24	0.11	1.44	2150	6.72	50.44	0.15	1.59
3	K-SIMC	8.88	0.77	0.19	0.30	0.09	1.43	1677	7.66	46.26	0.16	1.59
3	PO PID	9.36	0.58	0.26	0.28	0.07	1.26	2426	6.37	51.40	0.14	1.59
4	δ-PRC	1.78	2.38	1.12	2.25	1.62	1.10	204	4.49	55.34	1.66	1.59
4	SIMC	1.10	2.50	0.60	3.18	2.55	1.64	69.15	5.11	54.77	2.34	1.59
4	K-SIMC	1.32	2.91	0.66	2.77	2.22	1.43	89.96	4.83	57.59	2.21	1.59
4	PO PID	1.82	2.62	1.33	1.94	1.65	1.03	247	4.32	65.49	1.92	1.59
5	δ-PRC	1.45	74.13	34.96	69.18	52.55	1.19	511	3.27	79.98	94.67	1.59
5	SIMC	0.79	47.50	11.40	100.35	70.04	1.66	93	4.06	56.90	81.24	1.59
5	K-SIMC	1.04	58.67	12.85	84.81	59.04	1.40	137	3.77	60.11	76.46	1.59
5	PO PID	1.54	61.36	28.67	59.59	43.95	1.01	444	3.32	64.87	65.01	1.59
6	δ-PRC	0.00	143.69	67.78	50.26	7657619	1.22	2.61	∞	47.63	25.58	1.59
6	SIMC	0.00	166.68	40.00	75.93	11629006	1.85	2.58	∞	46.63	40.01	1.59
6	K-SIMC	0.00	171.13	39.03	77.48	12279424	1.93	2.58	∞	46.81	41.07	1.59
6	PO PID	0.00	90.41	76.08	51.99	6322148	1.13	2.65	∞	47.79	24.68	1.59

Table 7: Es7-12. The table contains the PID controller parameters for fixed robustness $M_s = 1.59$ for the methods; δ-PRC, Opt-PRC and PO PID. Also include the Pareto performance J and the input usage TV. The following margins are included: Gain Margin (GM), Phase Margin (PM) and Delay Margin (DM).

E	Method	K_p	T_i	T_d	IAE_{vy}	IAE_{vu}	J	TV	GM	PM	DM	M_s
7	δ-PRC	1.22	0.79	0.37	0.83	0.71	1.27	456	∞	81.19	1.14	1.59
7	Opt-PRC	1.23	0.84	0.39	0.85	0.73	1.30	489	∞	84.67	1.22	1.59
7	PO PID	1.27	0.45	0.56	0.69	0.53	0.99	720	∞	67.73	0.78	1.59
8	δ-PRC	2.57	0.45	0.21	0.20	0.18	1.48	560	∞	42.66	0.08	1.59
8	Opt-PRC	2.74	0.51	0.24	0.17	0.20	1.42	680	∞	43.95	0.07	1.59
8	PO PID	2.65	0.28	0.33	0.14	0.17	1.17	906	∞	45.40	0.07	1.59
9	δ-PRC	4.50	0.02	0.01	0.01	0.00	1.15	426	∞	48.20	0.00	1.59
9	Opt-PRC	4.39	0.02	0.01	0.01	0.01	1.33	502	∞	48.22	0.00	1.59
9	PO PID	4.43	0.01	0.01	0.01	0.00	1.00	507	∞	48.37	0.00	1.59
10	δ-PRC	36.75	0.26	0.12	0.18	0.01	1.74	4670	2.69	39.60	0.05	1.59
10	Opt-PRC	36.64	0.26	0.12	0.18	0.01	1.74	4669	2.69	39.69	0.05	1.59
10	PO PID	38.54	0.32	0.12	0.16	0.01	1.63	4644	2.70	39.08	0.04	1.59
11	δ-PRC	38.89	0.38	0.18	0.17	0.01	1.12	7166	3.98	56.61	0.08	1.59
11	Opt-PRC	46.54	0.28	0.13	0.20	0.01	1.01	6365	4.12	46.68	0.08	1.59
11	PO PID	47.50	0.38	0.12	0.18	0.01	0.97	5909	4.27	45.53	0.08	1.59
12	δ-PRC	-22.02	2.27	1.07	1.16	0.12	1.55	2441	∞	44.62	0.42	1.59
12	Opt-PRC	-22.20	2.42	1.14	1.10	0.13	1.53	2629	∞	45.37	0.40	1.59
12	PO PID	-22.11	2.23	1.33	1.05	0.13	1.49	3068	∞	46.22	0.37	1.59

C. ζ-PRC PID Controller Tuning

Note that in some examples we may possibly use ζ as the tuning parameter, i.e. for the incoming example we set $\delta = \bar{c} = \gamma = 2.12$.

Consider the following process model studied in Seborg et al. (2004) and Åström et al. (1998),

$$H_p(s) = \frac{1}{(s+1)(0.2s+1)(0.04s+1)(0.008s+1)}. \quad (16)$$

The main performance objective is $V_M = 4.2250\text{e-}4$. Interestingly, in this case, the ζ-tuning is 17 times better than mode 1 and 148 times better than SIMC in Dalen and Di Ruscio (2018). The trade-off curves are illustrated in Figure 11.

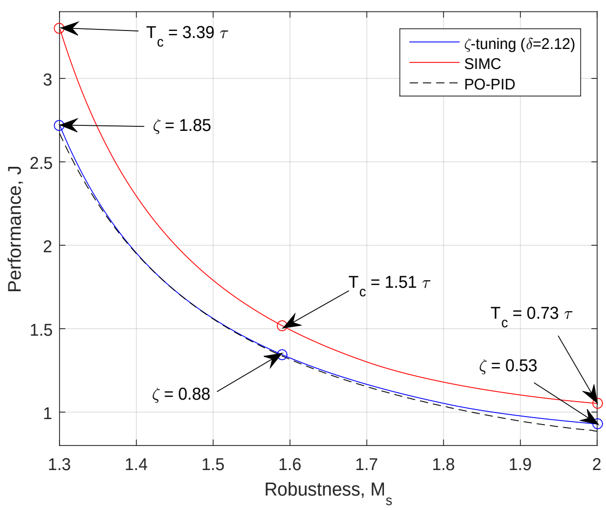

Figure 11: The figure illustrates the ζ-PRC tuning. Pareto performance J (Eq. (7)) vs. robustness M_s trade-off curves.

References

Abbasi, I., Ali, S., Ovinis, M., and Naeem, W. U-model based controller design for an unmanned free swimming submersible (UFSS) vehicle under hydrodynamic disturbances. 2017. 46:742–748.

Åström, K. and Hägglund, T. *PID Controllers: Theory, Design, and Tuning*. Instrument Society of America, 1995.

Åström, K. and Hägglund, T. Benchmark Systems for PID Control. *IFAC Proceedings Volumes*, 2000. 33(4):165 – 166. doi:10.1016/S1474-6670(17)38238-1. IFAC Workshop on Digital Control: Past, Present and Future of PID Control, Terrassa, Spain, 5-7 April 2000.

Åström, K. and Hägglund, T. Revisiting the zieglernichols step response method for pid control. *Journal of Process Control*, 2004. 14(6):635 – 650. doi:10.1016/j.jprocont.2004.01.002.

Åström, K. and Hägglund, T. *Advanced PID Control*. ISA-The Instrumentation, Systems, and Automation Society, 2006.

Astrom, K. J. and Hagglund, T. A frequency domain method for automatic tuning of simple feedback loops. In *The 23rd IEEE Conference on Decision and Control*. pages 299–304, 1984. doi:10.1109/CDC.1984.272361.

Åström, K. J., Panagopoulos, H., and Hägglund, T. Design of PI Controllers Based on Non-convex Optimization. *Automatica*, 1998. 34(5):585–601. doi:10.1016/S0005-1098(98)00011-9.

Dalen, C. and Di Ruscio, D. PD/PID controller tuning based on model approximations: Model reduction of some unstable and higher order nonlinear models. *Modeling, Identification and Control*, 2017. 38(4):185–197. doi:10.4173/mic.2017.4.3.

Dalen, C. and Di Ruscio, D. A Semi-Heuristic Process-Reaction Curve PID Controller Tuning Method. *Modeling, Identification and Control*, 2018. 39(1):37–43. doi:10.4173/mic.2018.1.4.

Daraz, A., Malik, S. A., Saleem, T., and Bhati, S. A. Ziegler Nichols Based Integral Proportional Controller for Superheated Steam Temperature Control System. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 2017. 11(5):570 – 574.

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.

Di Ruscio, D. Subspace system identification of the Kalman filter: open and closed loop systems. In *In Proc. Intl. Multi-Conf. on Engineering and Technological Innovation*. 2008.

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.

Di Ruscio, D. PI Controller Tuning Based on Integrating Plus Time Delay Models: Performance Optimal tuning. 2012. In Proceedings of the IASTED Control and Applications Conference. Crete Greece June 18-21.

Di Ruscio, D. and Dalen, C. Tuning PD and PID Controllers for Double Integrating Plus Time Delay Systems. *Modeling, Identification and Control*, 2017. 38(2):95–110. doi:10.4173/mic.2017.2.4.

Farouk, N., Sheng, L., and Said, L. Speed Control System on Marine Diesel Engine Based on a Self-Tuning Fuzzy PID Controller. 2012. 4(6):686–690.

Garpinger, O. and Hägglund, T. A Software Tool for Robust PID Design. *IFAC Proceedings Volumes*, 2008. 41(2):6416 – 6421. doi:10.3182/20080706-5-KR-1001.01082. 17th IFAC World Congress.

Grimholt, C. and Skogestad, S. PID-Control on First Order Plus Time Delay Systems & Verification of the SIMC Rules. *IFAC Proceedings Volumes*, 2013. 46(32):265–270. doi:10.3182/20131218-3-IN-2045.00122.

Grimholt, C. and Skogestad, S. Optimal PID control of double integrating processes. *IFAC-PapersOnLine*, 2016a. 49(7):127–132. doi:10.1016/j.ifacol.2016.07.228. Preprint, 11th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems.

Grimholt, C. and Skogestad, S. Optimization of Fixed-Order Controllers Using Exact Gradients. 2016b. URL `http://folk.ntnu.no/skoge/publications/2016/grimholt-jpc-pid-exact-gradient/main.pdf`. Unpublished.

Kristiansson, B. and Lennartson, B. Evaluation and simple tuning of PID controllers with high-frequency robustness. *Journal of Process Control*, 2006. 16(2):91 – 102. doi:10.1016/j.jprocont.2005.05.006.

Lee, J., Cho, W., and Edgar, T. F. Simple Analytic PID Controller Tuning Rules Revisited. *Industrial & Engineering Chemistry Research*, 2014. 53(13):5038–5047. doi:10.1021/ie4009919.

Ljung, L. *System Identification (2nd ed.): Theory for the User*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.

Mantz, R. J. and J. Tacconi, E. Complementary rules to Ziegler and Nichols' rules for a regulating and tracking controller. 1989. 49:1465–1471.

Manum, H. Extensions of Skogestads SIMC tuning rules to oscillatory and unstable processes, 2005. Internal report.

MATLAB. *Version 9.1.0.441655 (R2016b)*. The MathWorks Inc., Natick, Massachusetts, USA, 2016. Control System Toolbox, Version 9.3. Optimization Toolbox, Version 6.2.

S. J. Sadati, A. R. N. and Ghaderi, R. Fractional-Order Control of a Nonlinear Time-Delay System: Case Study in Oxygen Regulation in the Heart-Lung Machine. *Journal of Control Science and Engineering*, 2012.

S. Sai Tarun, G. I. K., P. Ramana. Controller Design and Load Frequency Control for Single Area Power System with Model Order Reduction Technique. *Int. Journal of Engineering Research and Applications*, 2014. 4(11):96–101. doi:10.3182/20131218-3-IN-2045.00122.

Salloum, R., Moaveni, B., and Arvan, M. R. Robust PID Controller Design for a Real Electromechanical Actuator. *Acta Polytechnica Hungarica*, 2014. 11(5):125–144.

Seborg, D., Edgar, T., and Mellichamp, D. *Process Dynamics and Control*. Number v. 1 in Chemical Engineering Series. Wiley, 1989.

Seborg, D., Edgar, T., and Mellichamp, D. *Process dynamics and control*. Wiley series in chemical engineering. Wiley, 2004.

Shamsuzzoha, M. Closed-Loop PI/PID Controller Tuning for Stable and Integrating Process with Time Delay. *Industrial & Engineering Chemistry Research*, 2013. 52(36):12973–12992. doi:10.1021/ie401808m.

Skogestad, S. Probably the best simple pid tuning rules in the world. In *AIChE Annual Meeting, Reno, Nevada, Nov.* 2001.

Skogestad, S. Simple analytic rules for model reduction and pid controller tuning. *Journal of Process Control*, 2003. 13(4):291 – 309. doi:10.1016/S0959-1524(02)00062-8.

Skogestad, S. Simple analytic rules for model reduction and PID controller tuning. *Modeling, Identification and Control*, 2004. 25(2):85–120. doi:10.4173/mic.2004.2.2.

Skogestad, S. and Grimholt, C. *The SIMC Method for Smooth PID Controller Tuning*, pages 147–175. Springer London, London, 2012. doi:10.1007/978-1-4471-2425-2_5.

Wang, R., Tan, C., Xu, J., Wang, Z., Jin, J., and Man, Y. Pressure Control for a Hydraulic Cylinder Based on a Self-Tuning PID Controller Optimized by a Hybrid Optimization Algorithm. *Algorithms*, 2017. 10(1):19. doi:10.3390/a10010019.

Ziegler, J. "On-the-job" adjustments of air operated recorder-controllers. *Instruments*, 1941. 16:394–7.

Ziegler, J. and Nichols, N. B. Optimum Settings for Automatic Controllers. *Trans. American Society of Mechanical Engineers*, 1942. 64:759–768.

Ziegler, J. and Nichols, N. B. Process lags in automatic control circuits. *Trans. American Society of Mechanical Engineers*, 1943. 65:433–444.