Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-7 (2016) 127–132

Optimal PID control of double integrating processes

Chriss Grimholt Sigurd Skogestad*

Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway
**e-mail: skoge@ntnu.no*

Abstract: In this paper we investigate optimal PID control of a double integrating plus delay process and compare with the SIMC rules. What makes the double integrating process special is that derivative action is actually necessary for stabilization. In control, there is generally a trade-off between performance and robustness, so there does not exist a single optimal controller. However, for a given robustness level (here defined in terms of the M_S-value) we can find the optimal controller which minimizes the performance J (here defined as the integrated absolute error (IAE)-value for disturbances). Interestingly, the SIMC PID controller is almost identical to the optimal PID controller. This can be seen by comparing the pareto-optimal curve for J as a function of M_S, with the curve found by varying the SIMC tuning parameter τ_c.

Keywords: PID control, optimization, double integrating plus time delay, SIMC rules.

1. INTRODUCTION

In this paper we investigate optimal PID control of a double integrating plus delay process,

$$G(s) = \frac{k'' e^{-\theta s}}{s^2} \tag{1}$$

where k'' is the process gain and θ is the time delay. We will mostly consider the serial (or cascade) PID form,

$$K_{\text{PID}}(s) = \frac{k_c(\tau_i s + 1)(\tau_d s + 1)}{\tau_i s}, \tag{2}$$

where k_c, τ_i and τ_d are the controller gain, integral time and derivative time. However, we will also compare with the more general parallel (ideal) form, which can have complex zeroes. For other notation, see Figure 1.

What makes the double integrating process special, is that derivative action is actually necessary for stabilization. Because the feedback system is unstable with proportional only controllers, traditional tuning methods like Ziegler and Nichols (1942) cannot be applied.

The SIMC method for PID controller tuning (Skogestad, 2003) has already found wide industrial usage. The SIMC rules are analytically derived, and from a first or second order process we can easily find the PI and PID controller setting, respectively. Even though the rule was originally derived mainly with simplicity in mind, recent studies have found that the resulting settings are very close to optimal (Grimholt and Skogestad, 2012, 2013). For the double integrating process, the SIMC rule gives the following PID settings for the serial form in (2):

$$k_c = \frac{1}{k''}\frac{1}{4(\tau_c+\theta)^2}, \quad \tau_i = 4(\tau_c+\theta), \quad \tau_d = 4(\tau_c+\theta). \tag{3}$$

The SIMC rule has one tuning parameter τ_c which can be used to trade off between performance (favoured by small τ_c) and robustness (favoured by large τ_c). For most processes, the recommended value for "tight control" (good performance subject to acceptable robustness) is $\tau_c = \theta$, but, as we will see, a value closer to 1.5θ may be better for the double integrating process.

There are many industrial and mechanical systems that have double integrating behaviour. Furthermore, the double integrating process is a special case of second-order processes

$$G(s) = \frac{k e^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)}, \tag{4}$$

with

$$k'' = k/(\tau_1 \tau_2). \tag{5}$$

The original SIMC PID tunings for a second-order process are (Skogestad, 2003)

$$k_c = \frac{1}{k}\frac{\tau_1}{(\tau_c+\theta)}, \tag{6}$$
$$\tau_i = \min\{\tau_1,\ 4(\tau_c+\theta)\}, \quad \tau_d = \tau_2.$$

The SIMC rule in (6) does not apply to double integrating processes, but by considering the SIMC rule for double integrating process in (3), we can generalize (6) to get a single SIMC PID-rule which covers all second-order processes:

$$k_c = \frac{1}{k}\frac{\tau_1}{(\tau_c+\theta)}\frac{\tau_2}{\tau_d}, \tag{7}$$
$$\tau_i = \min\{\tau_1,\ 4(\tau_c+\theta)\}, \quad \tau_d = \min\{\tau_2,\ 4(\tau_c+\theta)\}.$$

For processes with $\tau_1 > \tau_2 > 4(\tau_c+\theta)$ these settings are identical to those for the double integrating process in (3). Thus, a second-order process with large time constants τ_1 and τ_2, may be represented as a double integrating process.

It is generally difficult to achieve good performance for a double integrating process if the time delay θ is large, especially for disturbances at the input (d_u) which result in ramp deviations at the output (y). Because a ramp increases with t^2, the achievable IAE of the output increases proportionally with θ^3 (Skogestad, 2003) for a double integrating process, resulting in poor performance for double

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.228

Table 1. Reference and optimal PID controllers for double integrating processes ($k'' = 1$ and $\theta = 1$) with $M_S = 1.59$.

	k_c	τ_i	τ_d	IAE$_{dy}$	IAE$_{du}$	J	M_S	M_T	$1/\text{GM}_l$	GM	DM
Optimal controller for output disturbance	0.02	∞	24.13	**4.15**	∞	∞	1.59	1.09	∞	3.20	2.07
Optimal controller for input disturbance	0.04	10.12	10.12	6.28	**288.56**	1.257	1.59	1.71	5.79	4.02	1.80
Optimal PID combined (minimize J)	0.04	10.74	10.79	5.80	303.68	1.225	1.59	1.61	6.72	3.76	1.78

IAE$_{dy}$ and IAE$_{du}$ are for a unit step disturbance on output (y) and input (u), respectively.

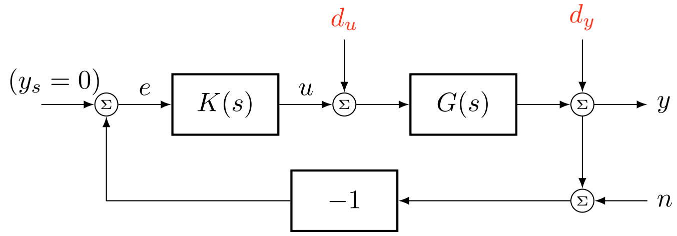

Fig. 1. Block diagram of the feedback control system. We do not consider setpoint changes (y_s) or noise (n).

integrating processes with a large delay. Thus, in practice, cascade control is used for control of double integrating processes whenever possible, which results in control of two integrating processes (for which IAE only increases with θ^2). For example, for mechanical systems, the process from u = force (acceleration) to y = position is a double integrating process, but by measuring and controlling also y_2 = velocity, we instead get two integrating processes. Rao and Bernstein (2001) study control of a double integrator with input saturation. They mostly assume full state feedback, which requires two measurements, similar to the use of cascade control. However, in this paper we consider the case where only the output y is measured and the process is double integrating.

Because many important industrial processes can be classified as double integrating, we want in this paper to investigate optimal PID control and the optimality of SIMC for this type of processes. Optimality is generally difficult to define as there are many issues to consider, including:

- Output performance
- Stability robustness
- Input usage
- Noise sensitivity

This may be considered a multiobjective optimization problem, but we consider only the main dimension of the trade-off space, namely high versus low controller gain. High controller gain favours good output performance, whereas low controller gain favours the three other objectives listed above. We can then simplify and say that there are two main objectives:

1. Performance (here measured in terms of IAE)
2. Robustness (here measured in terms of M_S-value)

Pareto optimality applies to multiobjective problems, and implies that no further improvement can be made in objective 1 without sacrificing objective 2. The idea is then to find the Pareto optimal controller, and compare with the SIMC tuning.

The paper is structured as follows. First the evaluation criteria and the optimization problem are defined. Then the optimal trade-off between performance and robustness is found and compared with SIMC. Following this, is a small comparison between serial and parallel PID controller. The paper ends with a time domain comparison between the different controllers, and a discussion.

2. EVALUATION CRITERIA

2.1 Performance

In this paper we choose to quantify performance terms of the integral absolute error (IAE),

$$\text{IAE} = \int_0^\infty \left| y(t) - y_s(t) \right| \mathrm{d}t. \tag{8}$$

To balance the servo/regulatory trade-off, we choose as the performance index a weighted average of IAE for a step input disturbance d_u and step output d_y,

$$J(p) = 0.5 \left(\frac{\text{IAE}_{dy}(p)}{\text{IAE}^\circ_{dy}} + \frac{\text{IAE}_{du}(p)}{\text{IAE}^\circ_{du}} \right) \tag{9}$$

where IAE$^\circ_{dy}$ and IAE$^\circ_{du}$ are weighting factors, and p is the controller parameters. In this paper, we select the two weighting factors as the optimal IAE values when using PID control, for input and output disturbances, separately (as recommended by Boyd and Barratt (1991)). To ensure robust reference PID controllers, they are required to have $M_S = 1.59$, and the weighting factors are IAE$^\circ_{dy} = 4.15$ and IAE$^\circ_{du} = 288.56$ (see Table 1).

As seen from Table 1, the optimal PID controller for combined input and output disturbances (J) favours input disturbances, and is almost identical to the optimal controller when only considering input disturbance (d_u). Therefore, it would be sufficient for double integrating plus delay process to only consider input disturbances. Nevertheless, to keep this analysis similar to other studies we have conducted on optimal controller tuning (Grimholt and Skogestad, 2012, 2013), we have chosen to include both input and output disturbances in the cost function.

2.2 Robustness for design: M_S

In this paper, we quantify robustness in terms of M_S, defined as

$$M_S = \max_\omega \left| S(j\omega) \right| = \left\| S(j\omega) \right\|_\infty, \tag{10}$$

where $\|\cdot\|_\infty$ is the H_∞ norm (maximum peak as a function of frequency), and the sensitivity transfer functions are defined as

$$S(s) = 1/(1+G(s)\,K(s)) \quad \text{and} \quad T(s) = 1 - S(s). \tag{11}$$

In robustness terms, M_S is the inverse of the closest distance from the loop function $L(s) = G(s)K(s)$ to the critical point, -1, in a Nyquist plot (see Figure 2). Originally, we considered using the largest peak of S *and* T, denoted M_{ST}, as the robustness criterion, but as discussed later we decided to use M_S.

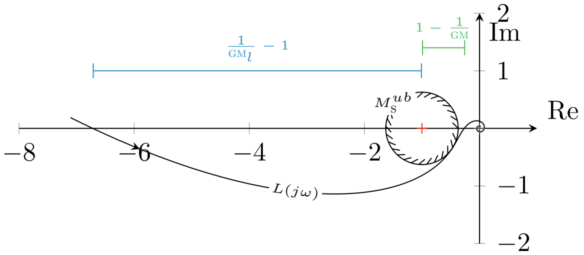

Fig. 2. Nyquist plot of the optimal PID controller for the control of double integration process (1) with $k'' = 1$ and $\theta = 1$ and robustness $M_s^{ub} = 1.59$ (Table 1). The robustness constraint M_s^{ub} and the gain margins are marked.

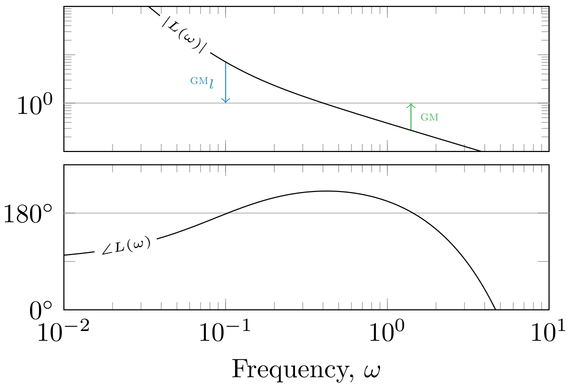

Fig. 3. Bode plot of the optimal PID controller for the control of double integration process (1) with $k'' = 1$ and $\theta = 1$ and robustness $M_s^{ub} = 1.59$ (Table 1). The gain margins are marked.

2.3 Robustness for analysis: Gain margin and Delay margin

In addition to the M_s value, we consider for robustness analysis the gain margin (GM) and the delay margin (DM), which have a clear physical meaning. The GM is defined as the factor by which we can multiply the controller gain (or more generally, the loop gain) before getting instability. Actually, as illustrated in Figure 2 and Figure 3, there are two gain margins in our case. The "normal" gain margin (GM) is the factor by which we can increase the loop gain, and the "lower" gain margin (GM_l) is the factor by which we can *decrease* the loop gain. For stability we need $\mathrm{GM} > 1$ and $\mathrm{GM}_l < 1$, but for acceptable robustness we typically want $\mathrm{GM} > 3$ and $\mathrm{GM}_l < 0.33$. In the Tables, we show $1/\mathrm{GM}_l$ (the factor by which the gain can be reduced) which typically should be larger than 3.

The delay margin is the allowed increase in delay in the feedback loop, $\Delta\theta_{\max}$, before we get instability. Note that $\theta_{\max} = \mathrm{PM}/\omega_c$ where PM [radians] is the phase margin and ω_c [rad/s] is the gain crossover frequency. In this paper, we will use the relative delay margin, defined as $\mathrm{DM} = \theta_{\max}/\theta$.

2.4 Optimal Trade-off

The optimal PID controllers are found by solving the following optimization problem,

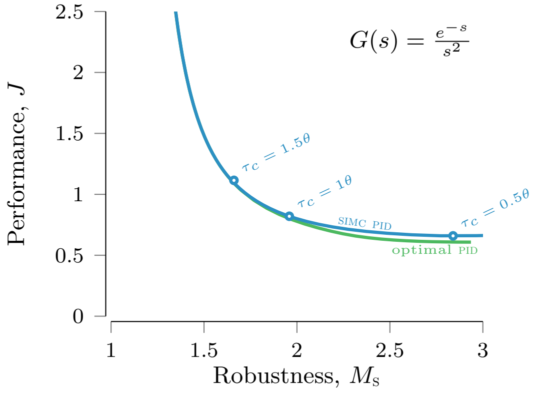

Fig. 4. Trade-off between performance and robustness for Pareto optimal PID controllers (green curve) and SIMC PID (blue curve).

$$\min_{p} \quad J(p) = 0.5\left(\frac{\mathrm{IAE}_{dy}(p)}{\mathrm{IAE}^{\circ}_{dy}} + \frac{\mathrm{IAE}_{du}(p)}{\mathrm{IAE}^{\circ}_{du}}\right) \tag{12}$$

$$\text{subject to:} \quad M_s(p) \leq M_s^{ub} \tag{13}$$

where the controller $K(s)$ is a PID controller. For more details on how to solve the optimization problem, see Grimholt and Skogestad (2015). To find the optimal trade-off between performance (J) and robustness (M_s), the optimization problem is solved repeatedly with different upper limits on the robustness (M_s^{ub}).

3. OPTIMAL PID SETTINGS AND COMPARISON WITH SIMC

The optimal and SIMC PID controllers are given in Table 2 for four values of M_s (1.4, 1.59, 1.8 and 2). The Pareto-optimal trade-off between performance (J = IAE) and robustness (M_s) is shown in Figure 4 (green curve) and compared with the SIMC PID-controller (blue curve). The trade-off curves for the SIMC controllers were generated by varying the tuning parameter τ_c from a large to a small value. The SIMC controllers corresponding to three specific choices are shown by circles:

- $\tau_c = 1.5\theta$ (smoother tuning)
- $\tau_c = \theta$ (default tight tuning)
- $\tau_c = 0.5\theta$ (more aggressive tuning)

For all robustness levels (in terms of M_s), we find that the SIMC rule is very close to the optimal. However, for the normally recommended tuning ($\tau_c = \theta$), M_s is quite high, being close to $M_s = 2$. A better value for the SIMC PID tuning constant in (3) is $\tau_c = 1.5\theta$ which gives $M_s = 1.65$.

The corresponding optimal and SIMC PID tuning parameters are shown in Figure 5 as a function of the robustness M_s. We find that the optimal PID-controller (serial form) always has $\tau_i = \tau_d$. The results also show that in the more robust region ($M_s < 1.6$), the SIMC tuning parameters are almost identical to the optimal PID controller. In the less robust region with higher performance ($M_s > 1.6$), the SIMC controller gain is slightly higher and the integral and derivative time slightly smaller than the optimal. However, as seen from Figure 4, this deviation from the optimal PID parameters has little effect on performance.

Table 2. Optimal and SIMC for double integrating processes ($k'' = 1$ and $\theta = 1$) with robustness $M_S = 1.40$, 1.59, 1.80, and 2.00.

	$k_c k''\theta^2$	τ_i/θ	τ_d/θ	IAE$_{dy}/\theta$	IAE$_{du}/k''\theta^3$	J	M_S	M_T	$1/\text{GM}_l$	GM	DM	τ_c/θ
Optimal PID	0.0209	13.67	13.67	7.25	653.1	2.006	1.40	1.56	6.69	5.14	2.62	–
SIMC	0.0209	13.84	13.84	7.16	662.7	2.011	1.40	1.55	6.86	5.08	2.62	2.46
Optimal PID	0.0354	10.74	10.79	5.80	303.7	1.225	1.59	1.61	6.72	3.76	1.78	–
SIMC	0.0356	10.60	10.60	5.88	297.7	1.225	1.59	1.63	6.52	3.80	1.78	1.65
Optimal PID	0.0505	9.37	9.38	4.96	185.7	0.919	1.80	1.66	7.02	2.99	1.32	–
SIMC	0.0512	8.84	8.84	5.24	172.7	0.931	1.80	1.74	6.23	3.10	1.31	1.21
Optimal PID	0.0625	8.64	8.76	4.46	138.2	0.777	2.00	1.72	7.39	2.55	1.06	–
SIMC	0.0651	7.84	7.84	4.91	120.5	0.801	2.00	1.86	6.01	2.70	1.05	0.96

IAE$_{dy}$ and IAE$_{du}$ are for a unit step disturbance on output (y) and input (u), respectively.

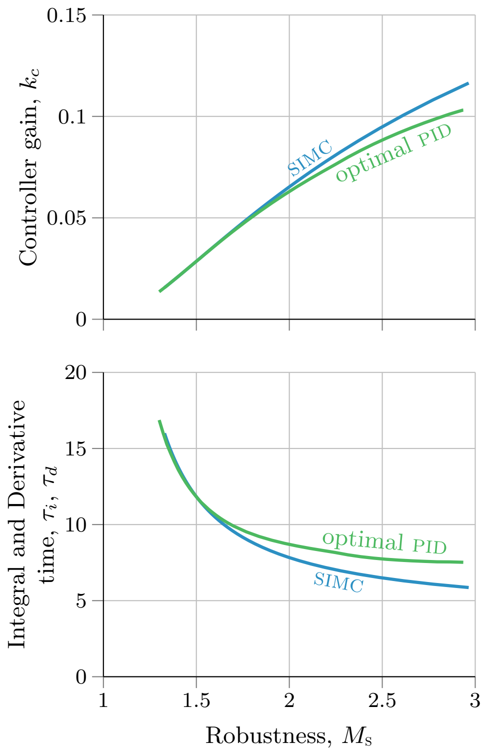

Fig. 5. Optimal and SIMC PID tuning parameters as a function of robustness for control of double integration process (1) with $k'' = 1$ and $\theta = 1$. For both controllers $\tau_i = \tau_d$.

4. PARALLEL VS. SERIAL PID CONTROLLER

The analysis in this paper is for the serial PID controller in (2). A more general PID controller is the parallel, or ideal, PID controller which allows for complex zeroes,

$$K_{\text{PID}}^{\text{parallel}}(s) = k_c'\left(1 + \frac{1}{\tau_i' s} + \tau_d' s\right). \tag{14}$$

Parallel PID controller parameters (14) can be calculated from serial PID controller parameters (2) by

$$f = 1 + \tau_d/\tau_i,\ k_c' = k_c f,\ \tau_i' = \tau_i f,\ \text{and}\ \tau_d' = \tau_d/f. \tag{15}$$

From Figure 5, we see that the optimal serial controller has equal integral and derivative times, which means that $f = 2$ and that we are just at the limit to having complex zeros. This indicates that better performance can be achieved by allowing for complex zeros by using a parallel PID controller.

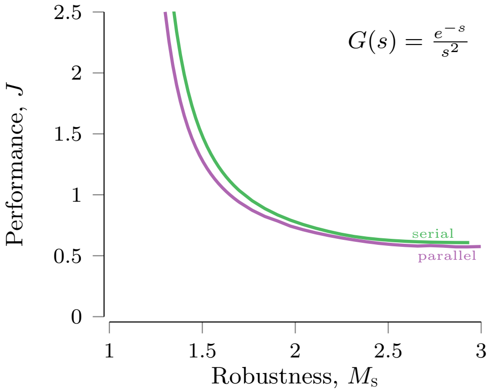

Fig. 6. Pareto optimal trade-off curves for optimal serial PID controller (the form used in this paper) and optimal parallel PID controller.

A comparison of the optimal trade-off curves for serial and parallel PID controllers is shown in Figure 6. Although the benefit of using the parallel PID form increases with increasing robustness, we see that the overall improvement is quite small. Thus, the serial implementation is sufficient for double integrating plus delay processes. A selection of optimal tunings are given in Table 3 with corresponding gain and delay margins.

5. SIMULATIONS

The responses to setpoint change and input disturbance for optimal serial, optimal parallel and SIMC PID controllers with M_S-value 1.40, 1.59, and 1.80 are shown in Figure 7. The corresponding tuning parameters are given in Tables 2 and 3.

As expected from the trade-off curves, the responses to the input disturbance are similar for the three controllers. The optimal parallel and serial PID controllers have almost the same peak deviation, but the parallel controller has better settling time. The SIMC and optimal PID controllers have almost identical responses.

6. DISCUSSION

6.1 Comparison with previous work

There is relatively little work on PID control of double integrating processes. Shamsuzzoha and Lee (2008) use IMC as a basis for designing PID controllers with λ (equivalent to

Table 3. Optimal parallel PID controllers of double integrating processes ($k'' = 1$ and $\theta = 1$) with robustness $M_S = 1.40, 1.59, 1.80$, and 2.00.

$k'_c k'' \theta^2$	τ'_i/θ	τ'_d/θ	IAE_{dy}/θ	$\text{IAE}_{du}/k''\theta^3$	J	M_S	M_T	$1/\text{GM}_l$	GM	DM
0.0416	16.39	7.17	7.88	411.6	1.663	1.40	1.64	4.22	4.95	2.63
0.0694	13.39	5.76	6.17	198.4	1.088	1.59	1.64	4.46	3.62	1.80
0.0974	11.98	5.09	5.21	126.1	0.847	1.80	1.63	4.81	2.89	1.34
0.1215	11.28	4.68	4.70	94.7	0.731	2.00	1.68	5.10	2.49	1.08

IAE_{dy} and IAE_{du} are for a unit step disturbance on output (y) and input (u), respectively.

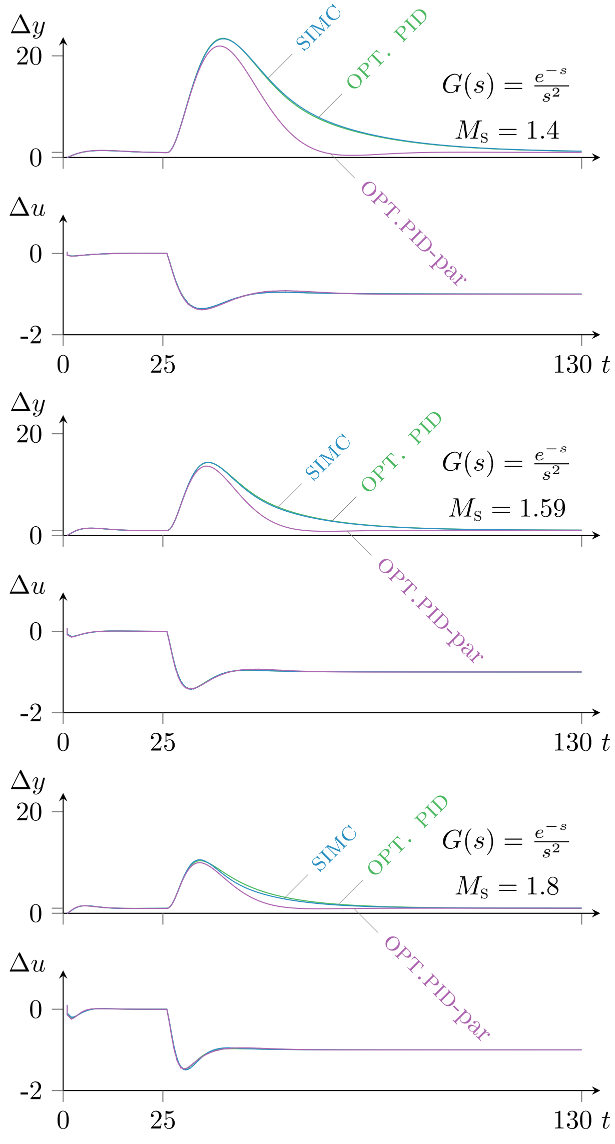

Fig. 7. Step responses to a setpoint step change ($t = 0$), and an input disturbance ($t = 25$) for the optimal PID, optimal parallel PID, and SIMC PID controllers with robustness $M_S = 1.4$ (top), $M_S = 1.59$ (middle), and $M_S = 1.8$ (bottom).

τ_c) as the tuning parameter. For the double integrating process $G(s) = e^{-0.8s}/s^2$ and $\lambda = 1.25$ they obtain a parallel PID controller with $k'_c = 0.3510$, $\tau'_i = 5.880$, and $\tau'_d = 2.343$. In the more general form the parameters are $k'_c k'' \theta^2 = 0.2246$, $\tau'_i/\theta = 7.35$, and $\tau_d/\theta = 2.93$. This controller has poor robustness with M_S= 2.87. In terms of our IAE performance measure, it gives $J = 0.69$ which is a little above the optimal $J = 0.57$ in Figure 6. Shamsuzzoha and Lee (2008) claim that their PID controller is significantly better than the SIMC PID controller, but this is incorrect. The reason for the error is that they follow Liu et al. (2004) who failed to use (15) to translate the double integrating SIMC-settings (3) from serial to parallel form.

More recently, Hassaan (2015) has considered optimal PID control of double integrating processes using the serial PID structure (which he calls the PD-PI controller). He considers a variety of performance objectives, including IAE, integrated time absolute error (ITAE) and integrated squared error (ISE). However, he includes no robustness requirement, which means that his "optimal" solution has poor robustness. It would correspond to the largest M_S-value (where J has its minimum) on our robustness-performance trade-off plots.

6.2 Criteria for robustness

In this paper, we quantify robustness in terms of M_S, defined as the peak value of $S(s)$. Actually, we originally considered using the largest value of M_S and M_T (Garpinger and Hägglund, 2008),

$$M_{ST} = \max\{M_S, M_T\}, \tag{16}$$

where

$$M_T = \max_{\omega} \left|T(j\omega)\right| = \left\|T(j\omega)\right\|_{\infty}.$$

For most stable processes, $M_S \geq M_T$, but for unstable process, including the double integrating process, it may happen that $M_T > M_S$, and this is why we originally used the largest value of M_S and M_T (Garpinger and Hägglund, 2008) as the robustness criterion.

As seen from the (Figure 4), performance (J) gets very poor when M_S approaches about 1.3, and the corresponding value for M_T is a little higher. Thus, there will be problems when trying to specify too low values for M_S or M_T. In particular, we encountered this problem with the SIMC controller when specifying low values for M_T. For example, when specifying $M_T = 1.4$ and using SIMC, we had to increase τ_c/θ to 32.5 (compared to 2.46 with $M_S = 1.4$), resulting in very poor performance with $J = 58.9$ (compared to 2.85). However, as noted, specifying low values for M_S was much less of a problem and resulted in reasonable designs. Also, when we analysed more carefully the results, we could not see that the lower value for M_T was giving any benefit in terms of improved upper and lower gain margins and delay margins. We therefore decided to base the robustness criterion on M_S only.

6.3 Generalised SIMC - for second-order processes

As noted in the introduction, second-order processes with sufficiently large time constants ($\tau_2 > 4(\tau_c + \theta)$) should

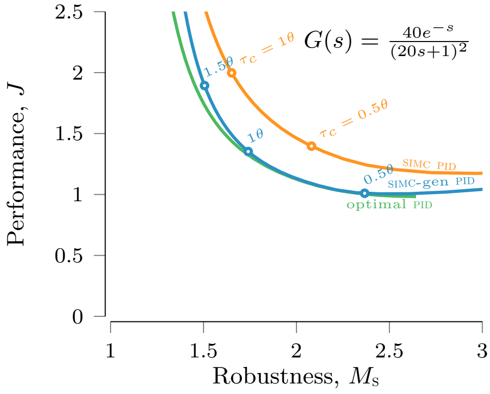

Fig. 8. Trade-off plot for Pareto optimal PID controller, SIMC PID in (6) and generalised SIMC PID controller in (7) of a almost double integrating second-order process.

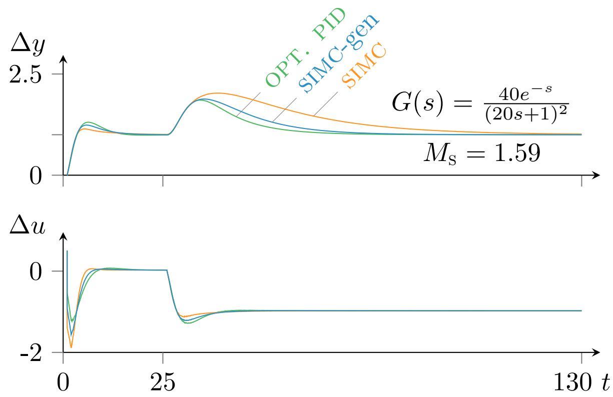

Fig. 9. Step responses to a setpoint step change ($t = 0$), and an input disturbance ($t = 25$) for the optimal PID controller, SIMC rule (6) second-order processes, and generalised SIMC (7) for second-order processes, all with robustness $M_s = 1.59$.

be approximated as double integrating processes. As an example, consider the process

$$G(s) = \frac{40e^{-s}}{(20s+1)^2}. \tag{17}$$

If we use the original SIMC rules for a second-order process in (6) then $\tau_d = 20$, and the derivative time will be larger than the integral time in most cases. For example, with $\tau_c = \theta = 1$, we get $\tau_i = 8$. However, if we use the "genarized" rule in (7), which is equivalent to representing the process as a double integrating process with $k'' = 40/20^2 = 0.1$, then we get $\tau_d = \tau_i = 8$. To confirm that this gives better performance, consider the trade-off curve in Figure 8. We see that the PID controller based on the double integrating process, that is, using the generalized SIMC settings in (7), is almost identical to the optimal PID controller, whereas the PID controller based on a second-order process, using the standard SIMC settings in (6), has significantly poorer performance for input disturbances. For output changes (and setpoint changes) the standard settings in (6) are a little better, but not significantly. This is also illustrated by the simulations in Figure 9.

7. CONCLUSION

In this paper we have derived optimal PID controller settings for a double integrating the process and compared the performance versus robustness trade-off with that obtained when varying the tuning parameter τ_c for the SIMC-controller in (3). As seen from Figure 4, the SIMC controller has almost identical performance with the optimal, in particular for more robust designs (with lower value of M_s). This means that the simple SIMC PID tuning rules given in (3) are essentially the optimal. This is quite surprising, because the double integrating SIMC rules were originally derived in a fairly *ad hoc* manner, aiming more towards simplicity than optimality.

We also find that for PID tuning, a second-order process (4) with $\tau_2 > 4(\tau_c + \theta)$ should be approximated as a double integrating process (1) with gain $k''{=}k/(\tau_1\tau_2)$. Alternatively and equivalently, we may use the "generalized" SIMC rules in (7).

REFERENCES

Boyd, S.P. and Barratt, C.H. (1991). *Linear controller design: limits of performance*. Prentice Hall Englewood Cliffs, NJ.

Garpinger, O. and Hägglund, T. (2008). A software tool for robust PID design. In *Proc. 17th IFAC World Congress, Seoul, Korea*.

Grimholt, C. and Skogestad, S. (2012). Optimal PI-control and verification of the SIMC tuning rule. In *IFAC conference on Advances in PID control (PID'12)*. The International Federation of Automatic Control.

Grimholt, C. and Skogestad, S. (2013). Optimal PID-control on first order plus time delay systems & verification of the simc rules. In *10th IFAC International Symposium on Dynamics and Control of Process Systems*.

Grimholt, C. and Skogestad, S. (2015). Optimization of fixed order controllers using exact gradients. *Unpublished, submitted to Journal of Process Control*.

Hassaan, G.A. (2015). Controller tuning for disturbance rejection associated with delayed double integrating processes: Part i: PD-PI controller. *International Journal of Computer Techniques*, 2, 110–115.

Liu, T., He, X., Gu, D., and Zhang, W.D. (2004). Analytical decoupling control design for dynamic plants with time delay and double integrators. *IEE Proc.-Control Theory Appl.*, 151(6), 745–753.

Rao, V. and Bernstein, D. (2001). Naive control of the double integrator. *IEEE Control Syst. Mag.*, 21, 86–97.

Shamsuzzoha, M. and Lee, M. (2008). PID controller design for integrating processes with time delay. *Korean Journal of Chemical Engineering*, 25, 637–645.

Skogestad, S. (2003). Simple analytic rules for model reduction and PID controller tuning. *Journal of process control*, 13(4), 291–309.

Ziegler, J.G. and Nichols, N.B. (1942). Optimum settings for automatic controllers. *Trans. ASME*, 64(11), 759–768.