symmetry

MDPI

Article

Extending the Model-Based Controller Design to Higher-Order Plant Models and Measurement Noise

Mikulas Huba [1,*] and Damir Vrancic [2]

[1] Institute of Automotive Mechatronics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, 812 19 Bratislava, Slovakia
[2] Department of Computer Automation and Control, J. Stefan Institute, SI-1000 Ljubljana, Slovenia; damir.vrancic@ijs.si
* Correspondence: mikulas.huba@stuba.sk; Tel.: +421-905-524-357

Abstract: The article extends a model-based controller design to higher-order systems, focusing on the speed and shapes of the closed loop responses, including the noise attenuation. It shows that, to obtain simple but reliable results, it is necessary to pay attention to the initial process identification and modelling and also to modify the target closed-loop transfer functions, which must remain causal. To attenuate high initial control signal peaks, appropriate pre-filters are introduced. In order to work with as few parameters as possible, all higher-order transfer functions (process models, target closed loops, pre-filters and noise-attenuation filters) are selected in the form of binomial filters with multiple time constants. Consequently, the so-called "half-rule", used to reduce too complex process transfer functions, has been modified accordingly. Because derived controllers can lead to different transient dynamics depending on the context of use, the article recalls the need to introduce dynamic classes of control to clarify the mission of individual types of controllers. Consequently, also the performance evaluation using the total variation (TV) criterion had to be refined. Indeed, in its original version, TV is not suitable to distinguish between reasonable and excessive control effort due to improper tuning and noise. The modified TVs allow evaluating higher order systems with multiple changes in direction of their control signal increase without contributing to the excessive control increments. The advantages of the proposed modifications, compared to the traditional approaches, are made clear through simulation examples.

Keywords: model based control; plant modeling; dead-time approximation; delay equivalences; filtration; noise attenuation; derivative action; total variation; monotonicity

Citation: Huba, M.; Vrancic, D. Extending the Model-Based Controller Design to Higher-Order Plant Models and Measurement Noise. *Symmetry* **2021**, *13*, 798. https://doi.org/10.3390/sym13050798

Academic Editor: Dmitry V. Dolgy

Received: 5 January 2021
Accepted: 29 April 2021
Published: 4 May 2021

1. Introduction

The development of embedded computers and programmable devices has led to an amazing expansion of applications for automatic control. At the same time, the expansion of embedded solutions also has some drawbacks. Typically, the design aspects in embedded systems are usually much more important than the optimal design of control systems. The reason for this may be the large number of different control approaches that have been developed in the last century. Selecting the most suitable one seems to be a difficult and, above all, time-consuming task. The time required for control loop optimization was quite limited even a few decades ago [1]. Now the situation is even worse under the constant pressure of management deadlines. The problems are also caused by the ever-increasing performance requirements and limits of the improved control devices, which also affect linear PID design to varying degrees. The explosion of various new methods, among them also those of fractional controller design [2,3]) cannot be overlooked, but new solutions can also be documented by the recent revision of SIMC (SImple Control) tuning rules. In the original work [4], for the most frequently used first-order time delayed (FOTD) models [5], the PI controller was predominantly proposed. Recently, IAE optimization-based PID controller modifications have been presented in [6] for this task to increase the

achievable performance limits. Moreover, in the context of works dealing with fractional-order PID controllers that eventually lead to implementation by higher-order controllers (HO) [2], or works proposing directly HO derivatives (see, e.g., [7–11] and the references therein), a trend toward using HO controllers is evident. The nice features of the model-based approach were its constructiveness, simplicity and clarity. It coped well with the requirements [4,12] that the controller design should be:

1. well motivated,
2. preferably model-based,
3. analytically derived,
4. simple and easy to remember,
5. work well for a variety of processes,
6. provide fast tracking speed and good disturbance rejection,
7. provide stability and robustness with lower variance of process inputs, and
8. reduce sensitivity to measurement noise.

However, forced by the ever-increasing requirements for improved performance and robustness of transients, the analysis of the original SIMC design performed in this paper also reveals some ad hoc simplifications. These could be useful for controlling PI, but limit possible generalizations of the approach. The SIMC author has already attempted to improve the method by introducing the iSIMC approach, which, however, somewhat diminishes the advantages of the original analytical design by adding numerical optimization [6].

In addition to the generalization of the model-based SIMC design for higher order systems approximated by the transfer function ${}^{j}S(s)$ with a j-tuple time time constant T_j, the original SIMC approach is modified, extended or supplemented in seven other aspects. These include the early stages of controlled process identification, selection of target transfer functions, reduction of more complex process transfer functions to a suitable transfer function ${}^{j}S(s)$, pre-filter design, performance measures used to evaluate the design and its visualization, and design optimization itself.

The choice of the desired first-order closed-loop transfer function for higher-order processes resulted in a noncausal PID controller. Such simplifications complicated the design of the controller, as the realization of appropriate filters needed to implement the controller was far from trivial. By choosing a causal target transfer function, this problem can be avoided: its specification also includes the design of the necessary filters and thus the design of a PID controller is directly comparable to PI.

The mentioned design imbalance between PI and PID controllers is further reduced by a new formulation of the "half-rule", using an analytical model reduction to simplify more complex transfer functions. A measure of control performance called total variation (TV) has been proposed for the design [4]. Two decades ago, the introduction of TV to evaluate total controller performance was a revolutionary step that demonstrated the need to consider the shape of controller output. However, the limitations of using TV are based on the process orders. Namely, the higher order systems undergo several changes in the direction of the control signal in optimal response. However, such optimal control increases the value of TV and thus deceptively impairs its suitability. Another performance measure to evaluate the speed of transients is the integral of the absolute control error (IAE). Unfortunately, its optimal value usually results in overshoots of the controlled signal. However, these can be unacceptable for many (e.g., mechatronic) applications. To reduce the overshoot, additional design constraints, for example, the sensitivity peaks, were needed. As will be shown later, the aforementioned drawbacks of using both TV and IAE measures can be eliminated by using IAE in combination with modified TV measures based on deviations from the ideal input and output shapes of the system step responses.

Another disadvantage of the original SIMC design is that, with respect to the setpoint step responses, it leads to exaggerated kicks of the control signals to the setpoint changes, even for the simplest solutions. These can lead to the entire control solution being infeasible

in practice, especially for lag- dominant systems. Without a suitable design of a pre-filter (reference filter), the control dynamics for setpoint changes can become unacceptable.

After weighing all the problematic aspects, we finally conclude (supported by some recently published works [10,11,13]) that an increase in control performance, robustness and noise suppression can only be achieved by using higher order controllers.

With the aim of addressing the aforementioned trends, the paper is organized as follows. Section 2 builds an internal structure for SIMC design, extends it with pre-filters, and shows that the design is no longer limited to PI and PID controllers, but can be extended to HO controllers even for FOTD plant models. The section also gives some guidance on designing noise filters, modifying the half-rule method, and selection of the desired closed-loop transfer function.

The performance of the modified controller is measured by a refined total variation measure, which is summarized in Section 3. These further allow the modification of the loop optimization to focus on minimizing unnecessary signal increments at both the input and output of the system. The simulation examples given in Section 4, which illustrate the main features and advantages of the newly proposed modifications, are then discussed in Section 5. The final summary and possible further developments of the method are given in the conclusions.

2. Exploring the SIMC Method for Different Plant and Dead-Time Approximations

Figure 1 shows the controller $R(s)$ and the process $F(s)$. Before revising the results of the model-based controller design from [4], we first consider the pre-filter $P(s) = 1$. Also, the limitations of the control signal are not yet considered explicitly, but they must be respected at least implicitly. The corresponding closed-loop transfer function between the setpoint $W(s)$ and the process output $Y(s)$ is then

$$F_{cl}(s) = \frac{Y(s)}{W(s)} = P(s)\frac{R(s)F(s)}{1 + R(s)F(s)} = \frac{R(s)F(s)}{1 + R(s)F(s)}. \quad (1)$$

For the given desired closed-loop transfer function $F_{cl}(s)$, the controller can be calculated from (1) as follows:

$$R(s) = \frac{U(s)}{E(s)} = \frac{F_{cl}(s)}{1 - F_{cl}(s)}\frac{1}{F(s)}. \quad (2)$$

Numerous features of such a design have been studied and popularized by [4] in the form of simple analytical rules. The aim of our work is to modify some of its ad hoc features and generalize the above approach so that it can be effectively and reliably applied to higher order processes. In doing so, several steps of the original design need to be revised, including the use of higher-order process models if necessary. Since the modified controller design depends entirely on the order of the process model, the applied model order will be indicated by a superscript before the transfer function, for example, $F(s) = {}^{j}F(s)$ (e.g., ${}^{2}F(s)$ when using the second-order process model).

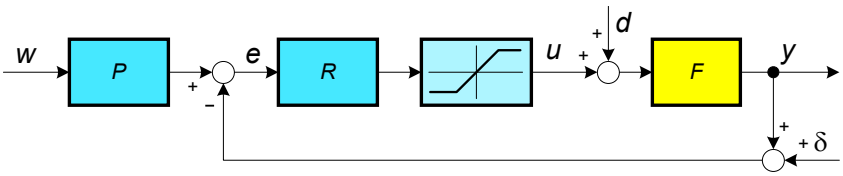

Figure 1. The controller R and the process F in the closed-loop configuration with a possible control signal limitation; P is the pre-filter, *d*-disturbance, δ-measurement noise.

2.1. Controllers Based on FOTD Models

As mentioned earlier, when using the stable first-order time-delay plant (FOTD) models, all associated transfer functions and controllers are denoted by a superscript "1".

For example, $F(s) = {}^1F(s)$ expresses the transfer function of the process between its input $U(s)$ and its output $Y(s)$ as:

$$ {}^1F(s) = \frac{Y(s)}{U(s)} = \frac{Ke^{-T_d s}}{1 + T_1 s}, \tag{3}$$

where T_d is the time delay, K is the plant gain and T_1 is the process time constant. Since the process has a time delay, the desired closed loop transfer function $F_{cl}(s) = Y(s)/W(s)$ between the reference setpoint $W(s)$ and the plant output $Y(s)$ must also include the time delay:

$$ {}^1F_{cl}(s) = \frac{1}{1 + T_{c1} s} e^{-T_d s} \tag{4}$$

where T_{c1} is the desired closed loop time constant.

Once the desired closed loop transfer function (4) is defined, the controller $R(s)$ parameters can be calculated from (2). However, the exponential term (the time delay) appears in the denominator of (2). To solve the equation, the simplest solution is to approximate the exponential term by the first-order Taylor series:

$$ e^{-T_d s} \approx 1 - T_d s. \tag{5}$$

The above approach yields the first-order controller (PI):

$$ {}^1R^1(s) = \frac{U(s)}{E(s)} = \frac{1 + T_1 s}{K(T_{c1} + T_d)s} = K_c(1 + \frac{1}{T_i s}), \tag{6}$$

where

$$ T_i = T_1 > 0;\ K_c = \frac{T_1}{K(T_{c1} + T_d)} \tag{7}$$

and where the left index "1" in ${}^1R^1$ denotes the order of the process model used to compute the controller parameters, and the right index "1" denotes the controller (in this case PI controller) order. Note that this nomenclature will be used throughout the paper from now on.

As indicated in (7), the integrative time constant and the process model time constant should be positive. This implies that the method can only be applied to stable processes. This restriction also applies to HO controllers.

Note that the recommended values for the closed-loop time constant are [4]:

$$ T_{c1} \geq T_d. \tag{8}$$

However, as will be shown later, the above restriction is only a guideline. For the simplest controllers, T_{c1} should be chosen even larger, while it can be reduced for HO controllers. To distinguish the process model parameters (${}^1F_m(s)$) from the actual process parameters (${}^1F(s)$), the subscript "m" is appended to the process parameters in the following text:

$$ {}^1F_m(s) = \frac{K_m e^{-T_{dm} s}}{1 + T_{1m} s}. \tag{9}$$

Besides developing the exponential term in the denominator of (2) with the first-order Taylor series (5), we can also use the Padé approximation [14]:

$$ e^{-T_d s} \approx \frac{1 - sT_d/2}{1 + sT_d/2}. \tag{10}$$

In this case, we obtain the following second-order (PID) controller:

$$ {}^1R^2(s) = K_c\left(1 + \frac{1}{T_i s}\right)\frac{1 + T_D s}{1 + T_{f1} s}, \tag{11}$$

where

$$T_i = T_1;\ T_D = T_d/2;$$
$$K_c = \frac{T_1}{K(T_{c1}+T_d)};\ T_{f1} = \frac{T_{c1}T_d}{2(T_{c1}+T_d)}. \tag{12}$$

The controller (11) is briefly discussed in [4], where the conclusion was that "it probably does not justify the increased complexity of the controller and the increased sensitivity to measurement noise". However, in a recent paper [6], the use of the derivative term is again proposed to improve the closed-loop performance.

Note that (similar to traditional analog controllers) the filter time constant T_{f1} is not arbitrary but uniquely given. Moreover, with $T_{c1} = 0.75T_d$ the parameters T_i, T_D and K_c (12) correspond to the recommended parameters of one of the oldest methods of PID controller tuning [15].

The use of higher order Taylor approximations does not lead to stable controllers, while this is not true for Padé approximations. Indeed, a stable higher order controller is obtained by evolving the exponential term in the denominator of (2) to the second order Padé approximations [14]:

$$e^{-T_d s} \approx \frac{1 - T_d/2s + T_d^2 s^2/12}{1 + T_d/2s + T_d^2 s^2/12} \tag{13}$$

From this approach, the third-order ${}^1R^3$-proportional-integrative-derivative-accelerative (${}^1PIDA$) controller is obtained:

$${}^1R^3(s) = K_c\left(1 + \frac{1}{T_i s}\right)\frac{1 + T_{D1}s + T_{D2}s^2}{1 + T_{f1}s + T_{f2}s^2}. \tag{14}$$

where

$$T_i = T_1;\ T_{D1} = T_d/2;\ T_{D2} = T_d^2/12;$$
$$K_c = \frac{T_1}{K(T_{c1}+T_d)};\ T_{f1} = \frac{0.5T_{c1}T_d}{T_{c1}+T_d};\ T_{f2} = \frac{0.0833T_{c1}T_d^2}{T_{c1}+T_d}. \tag{15}$$

At this point, the reader may wonder why we should use HO PID controllers when sometimes the PID controllers already have excessive noise gain? As shown in [7–11] and as will be discussed later, the HO controllers can significantly improve control performance even in noisy environments without introducing excessive noise at the controller output. The higher order Padé approximations of the time delay lead to a better description of the controlled system and hence improved control performance. To simplify the further derivations for higher order process models, we will first introduce the normalized dimensionless parameters. By introducing the new time scale defined by the following complex variable:

$$p = T_d s, \tag{16}$$

all other time variables are then related to the time delay T_d (note that $T_d > 0$):

$${}^1F(p) = \frac{Ke^{-p}}{1 + \tau_1 p};\ {}^1F_{cl}(p) = \frac{1}{1 + \tau_{c1}s}e^{-p};$$
$$\tau_{c1} = \frac{T_{c1}}{T_d};\ \tau_i = \frac{T_i}{T_d};\ {}^1\kappa = K_c K;$$
$$\tau_{D1} = \frac{T_{D1}}{T_d};\ \tau_{D2} = \frac{T_{D2}}{T_d^2};\ \tau_{f1} = \frac{T_{f1}}{T_d};\ \tau_{f2} = \frac{T_{f2}}{T_d^2}. \tag{17}$$

Note that in (17), in addition to the normalized process and controller times denoted by the variables τ, the controller gain K_c is also normalized by the parameter ${}^1\kappa$. The controller parameters can then be expressed in a more compact form. For example, the solution (15) can now be expressed as:

$$\begin{array}{l} {}^1\kappa = \tau_1/[(\tau_{c1}+1)];\ \tau_{D1} = 1/2;\ \tau_{D2} = 1/12; \\ \tau_{f1} = 0.5/(1+1/\tau_{c1});\ \tau_{f2} = \tau_{f1}/6. \end{array} \tag{18}$$

In addition to the FOTD process model, the HO models ${}^jF(s), j > 1$ can also be used, as derived in the following subsections.

2.2. Controllers Based on SOTD Models

Remark 1 (The first major change in the SIMC design)**.** *The requirement of the first-order closed-loop transfer function for the second-order plant models in [4] leads to an ideal (improper) controller that cannot be realized in practice. This step was probably motivated by the design of PID controllers, prevalent at the time of the mentioned work [4], which did not take into account the calculation of the controller filter. However, such a design violated the requirements of causality, which it complied with only when choosing the appropriate delay of the desired closed-loop transfer function. This also makes the comparison of different controllers under the influence of noise difficult or impossible. Therefore, the mentioned design did not lead to a more efficient design of PID controllers, but resulted in comments such as that the derivative term is difficult to tune [16], that the design is not suitable for noisy and time-delayed processes [1], and that the PI control is preferred because of its simplicity [6].*

In view of the problems mentioned in Remark 1, for the second-order time-delayed (SOTD) plant model with a double time constant T_2

$${}^2F(s) = \frac{Ke^{-T_d s}}{(1+T_2 s)^2}, \tag{19}$$

the desired closed-loop transfer function obeying controller causality should be

$${}^2F_{cl}(s) = \frac{1}{(1+T_{c2}s)^2}e^{-T_d s}. \tag{20}$$

Then the first-order Taylor approximation (5) to (2) yields the second-order ${}^2R^2$ controller

$$\begin{array}{l} {}^2R^2(s) = K_c\dfrac{1+T_2 s}{T_2 s}\dfrac{1+T_2 s}{1+T_{f1}s}e^{-T_d s}; \\ K_c = \dfrac{T_2}{K(2\,T_{c2}+T_d)};\ T_{f1} = \dfrac{T_{c2}^2}{2\,T_{c2}+T_d} \end{array} \tag{21}$$

Note that ${}^2R^2$ is also a PID controller, but it must be distinguished from the ^{1}PID controller (11) from the previous section, by index ^{2}PID and by optimally tuned parameters. The difference between the two types of PID controllers mentioned above is crucial, which was pointed out some time ago [17]. In contrast to the SIMC PID controller, the change in the calculated K_c is obvious. In contrast to the original design, the filter constant T_{f1} is not chosen (e.g., as $T_D/100$), but results directly from the required transfer function ${}^2F_{cl}(s)$. Writing with dimensionless parameters

$$\tau_2 = T_2/T_d,;\ \tau_{c2} = T_{c2}/T_d,;\ {}^2\kappa = K_c K;\ \tau_{D1} = T_D/T_d, \tag{22}$$

We get

$${}^2\kappa = \tau_2/[(2\tau_{c2}+1)];\ \tau_i = \tau_2;\ \tau_{D1} = \tau_2;\ \tau_{f1} = \tau_{c2}^2/(2\tau_{c2}+1). \tag{23}$$

From the 1st order Padé approximation (10) we obtain the third-order ${}^2R^3$ controller

$$
\begin{aligned}
&{}^2R^3(s) = K_c\left(1+\frac{1}{T_i s}\right)\frac{1+T_{D1}s+T_{D2}s^2}{1+T_{f1}s+T_{f2}s^2};\\
&T_i = T_2;\ T_{D1} = T_2 + T_d/2;\ T_{D2} = T_2 T_d/2\\
&K_c = \frac{T_2}{K(2\,T_{c2}+T_d)};\\
&T_{f1} = \frac{T_{c2}(T_{c2}+T_d)}{2\,T_{c2}+T_d};\ T_{f2} = \frac{0.5T_{c2}^2 T_d}{2\,T_{c2}+T_d},
\end{aligned}
\tag{24}
$$

with the dimensionless parameters

$$
\begin{aligned}
&{}^2\kappa = \tau_2/[(2\tau_{c2}+1)]; \tau_i = \tau_2; \tau_{D1} = \tau_2 + 1/2; \tau_{D2} = \tau_2/2\\
&\tau_{f1} = \tau_{c2}(\tau_{c2}+1)/(2\tau_{c2}+1);\ \tau_{f2} = 0.5\tau_{c2}^2/(2\tau_{c2}+1)
\end{aligned}
\tag{25}
$$

Again, we must not forget to distinguish this ^{2}PIDA controller from ^{1}PIDA (14) from the previous section. Similarly, we can derive the controller using the 2nd order Padé approximation. To simplify and unify the nomenclature used in the following text, we should replace the traditional abbreviations (PI, PID, or PIDA) with the shorter term ${}^j\mathrm{R}^m(s), m \geq j$ according to the following definition.

Definition 1 (${}^j\mathrm{R}^m(s), m \geq j$ controller)**.** *The designated ${}^jR^m(s)$ is the model-based controller* (2) *with $F(s) = {}^jF(s)$ where j denotes the order of the plant model. The symbol m denotes the controller order, which depends on the chosen dead-time approximations* (5), (10), *or* (13).

2.3. Controller Based on TOTD Models

The third-order time delayed (TOTD) plant model with a triple time constant T_3

$$
{}^3F(s) = \frac{Ke^{-T_d s}}{(1+T_3 s)^3}
\tag{26}
$$

with the required closed loop transfer function

$$
{}^3F_{cl}(s) = \frac{1}{(1+T_{c3}s)^3}e^{-T_d s};\ {}^3F_{cl}(p) = \frac{1}{(1+\tau_{c3}s)^3}e^{-p}
\tag{27}
$$

and the first-order Taylor approximation (5) yield a solution of (2) in the form of the third-order ${}^3R^3$ controller (^{3}PIDA controller)

$$
\begin{aligned}
&{}^3R^3(s) = K_c\frac{1+T_i s}{T_i s}\frac{1+T_{D1}s+T_{D2}s^2}{1+T_{f1}s+T_{f2}s^2};\\
&T_i = T_3;\ T_{D1} = 2T_3;\ T_{D2} = T_3^2;\\
&K_c = \frac{T_3}{K(3\,T_{c3}+T_d)};\\
&T_{f1} = \frac{3T_{c3}^2}{3\,T_{c3}+T_d};\ T_{f2} = \frac{T_{c3}^3}{3\,T_{c3}+T_d}.
\end{aligned}
\tag{28}
$$

In the dimensionless parameters

$$
\begin{aligned}
&{}^3\kappa = K_c K = \tau_3/[(3\tau_{c3}+1)];\\
&\tau_i = \tau_3;\ \tau_{D1} = 2\tau_3;\ \tau_{D2} = \tau_3^2;\\
&\tau_{f1} = 3\tau_{c3}^2/(3\tau_{c3}+1);\ \tau_{f2} = \tau_{c3}^3/(3\tau_{c3}+1).
\end{aligned}
\tag{29}
$$

As can be seen, the model-based design leads to up to 3 different PIDA controllers, each with a different task and optimal settings. Similarly, we could obtain ${}^3R^4(s)$ with the first and ${}^3R^5(s)$ with the second-order Padé approximations.

2.4. Controller Based on QOTD Models

The fourth-order time delayed model with a quadruple time constant T_4 and dead-time will be defined as

$$ {}^4F(s) = \frac{Ke^{-T_d s}}{(1+T_4 s)^4} \tag{30} $$

The desired closed-loop transfer function will be chosen as:

$$ {}^4F_{cl}(s) = \frac{1}{(1+T_{c4}s)^4} e^{-T_d s}. \tag{31} $$

Then, considering (2), (5) gives the following fourth-order controller ${}^4R^4$ with the 3rd-order derivative action

$$ \begin{aligned} &{}^4R^4(s) = K_c \frac{1+T_i s}{T_i s} \frac{1+T_{D1}s+T_{D2}s^2+T_{D3}s^3}{1+T_{f1}s+T_{f2}s^2+T_{f3}s^3};\ T_i = T_4;\ T_{D1} = 3T_4;\ T_{D2} = 3T_4^2;\ T_{D3} = T_4^3; \\ &K_c = \frac{T_4}{K(4\,T_{c4}+T_d)}; \\ &T_{f1} = \frac{6T_{c4}^2}{4\,T_{c4}+T_d};\ T_{f2} = \frac{4T_{c4}^3}{4\,T_{c4}+T_d};\ T_{f3} = \frac{T_{c4}^4}{4\,T_{c4}+T_d}. \end{aligned} \tag{32} $$

Such controllers are used, for example, in mechatronics and consider feedback of position, velocity, acceleration and jerk. We could therefore refer to them as ^{4}PIDAJ.

To control the fourth-order system considered with $T_d = 0$ and denoted as E4 in [4]

$$ {}^4F(s) = \frac{1}{(1+s)^4} \tag{33} $$

Its model order was first reduced by the "half-rule" method. Since, in [4], also the desired closed loop (4) does not satisfy the causality conditions, we will use this example to compare the traditional and modified model-based approaches. In contrast to the ideal PID proposed in [4], the newly proposed solution (32) clearly separates the filter required in (2) (with the time constants T_{f1}, T_{f2}, T_{f3}) from a possible additional filter that attenuates the measurement noise. Thus, the proper ${}^4R^4(s)$ transfer function simplifies the evaluation and comparison of noise attenuation filters.

2.5. Why Just the Multiple Plant Time Constants?

Definition 2 (*j*th-order time delayed (jOTD) plant models ${}^jF(s)$ and the corresponding *m*th-order ${}^j\mathrm{R}^m(s)$ controllers, $j = 1, 2, ...; m \geq j$). *For the sake of simplicity, all the above plant models* ${}^jF(s)$

$$ {}^jF(s) = \frac{Y(s)}{U(s)} = \frac{e^{-T_d s}}{(1+T_j s)^j};\ j \in [1,4] \tag{34} $$

used for derivation of ${}^jR^m(s)$ *controllers consider just a single j-tuple time constant* T_j.

The multiple time constants (see e.g., [18]) have long been used to decrease the number of identified parameters and thus to avoid ill-conditioned identification relationships and simplify their solution. Obviously, by considering not equal stable time constants in ${}^jF(s)$, $j = 1, 2, ...$, the family of ${}^j\mathrm{R}^m(s)$ controllers could be significantly extended and be more accurate representation of the actual process. However, the controller design should be robust enough to neglect small differences between the time constants of the model, or to reduce the model transfer functions appropriately in the case of higher differences. Therefore, we are going to modify accordingly the original "half-rule", proposed in [4], used for model reduction.

Additionally, for not equal model time constants, we would also have to select which time constants T_j will be used for the calculation of the controller integral time parameter T_i and which for the calculation of the derivative terms time constants. As has been shown

in [19], in constrained control, the properties of the loop with simple anti-windup according to [1] can vary significantly by the mentioned choice of time constants.

2.6. Low-Pass Noise Attenuation Filters

Due to the proportional term, the high frequency noise is not attenuated even in the simplest PI controller. To decrease the noise level, some of works (without giving any arguments) are suggesting using the 2nd-order Butterworth filters [20], while other recommend using the simplest binomial filters at the controller input, or output [21]

$$Q_n(s) = 1/\left(T_f s + 1\right)^n. \tag{35}$$

With regard to the minimum number of parameters, we will also prefer this second option in this work. The filter will be included in the controller settings using a half-rule and its modification.

2.7. Original Half Rule Method

To satisfy PI and PID controller design, ref. [4] proposed a two-step procedure:

Step 1. Obtain a FOTD or SOTD process model. The effective delay and time constants in this model may be obtained using the half-rule. Thereby, the half-rule was formulated for a mix of different time constants. When simplifying process transfer function including several delays

- the largest neglected (denominator) time constant (lag) has been distributed evenly to the effective delay and the smallest retained time constant,
- the effective delay has summarized (besides of above contribution) the original plant delay and different shorter loop delays.

Step 2. Derive model-based controller settings. The PI controller parameters can be obtained from the FOTD model, whereas the PID controller parameters are calculated from the SOTD model. However, we will further examine how the original HR method supports the design of HO controllers.

Thereby, by the chosen target model transfer function, the strategy of the reduction process is not clearly defined. On the one hand side, by adding half of the neglected time constant to the smallest retained time constant, it seems to decrease differences between the retained (dominant) time constants. However, on the other side, in the reduction process starting with a double dominant plant time constant and an n-fold shorter (filter) time constant with the aim to get a second-order model, Skogestad's approach would modified just one of the dominant time constants by one half of the neglected filter time constant. Its second half, together with the remaining $n-1$ time constants, would be included into the dead-time. We consider such an increase in the number of mutually different time constants of the process to be unnecessary.

Since from context of the paper [4] one could understand that, preferably, FOTD models and PI controller should be used and for many years, the dominant use of PI controllers only confirmed success of such an approach, we could ignore the previous remark.

2.8. Modified Half Rule for Multiple Time Constants

Note that recently, under denotation "improved SIMC (iSIMC)", a modified IAE-optimization-based rules [6,22] have been published. They also provide the PID controller (i.e., ^{1}PID) parameters for FOTD processes. Publication of iSIMC may also be considered as one of the attempts to generalize SIMC rules to HO controllers. However, with respect to FOTD models used, this modification provided no incentive for HR changes. Such motivation arises only when we deal with controllers from branches corresponding to the model of higher orders. (Note, thereby, that, with respect to the chosen branch of controllers, for example, PIDA controller may correspond to ^{1}PIDA, ^{2}PIDA or ^{3}PIDA controller and the reduction process should take these possibilities into account.)

As mentioned above, the effort to minimize the number of loop parameters when working with HO controllers leads us to use simplified system models and noise-attenuation filters with multiple time constants. All these impulses are leading us to a modification of the half-rule (MHR) as follow:

Step 1. Depending on the process type, obtain stable FOTD, SOTD, TOTD, or QOTD (possibly also higher-order) process models. By using the modified half-rule approach (as explained below), the effective time constants and time-delay in the chosen model may be calculated so as to simultaneously design the appropriate controller filter without a limitation on the model order.

Step 2. For the controller branch defined by the considered stable process model, derive the model-based controller settings. The first branch of ^{1}PI, ^{1}PID, or ^{1}PIDA-settings result from a FOTD model. The second branch of controllers (^{2}PID, or ^{2}PIDA) results from the SOTD process model. Similarly, the third branch of controllers (starting with PIDA corresponding to ^{3}PIDA and possibly continuing to higher-order ^{3}R^m controllers) results from TOTD plant models.

Although more accurate solutions can be proposed for the approximation of more complex processes (e.g., by the method of moments according to [23]), with regard to simplicity and tradition the low-pass binomial filters will be included into the plant delay by a modified half-rule (MHR). When starting with j nearly equal dominant time constants and wishing to keep this number, a symmetric distribution of one half of sum of the neglected smaller (possibly n-fold filter time constants) to all the retained j dominant time constants will be preferred. The 2nd half of this sum will be added to dead-time.

Definition 3 (Modified Half Rule (MHR) for jOTD models)**.** *When working with a combination of jOTD system* (34) *with ntuple filter time constant* T_f (35)*, the plant model parameters will be modified to keep a constant open loop average residence time (sum of all delays and time constants) [1]. Furthermore, MHR will symmetrically modify the j-tuple dominant time constant* T_j *which will be calculated from the identified value* T_{jm} *according to*

$$\begin{array}{l} jT_j = jT_{jm} + T_{f0}/2;\ T_d = T_m + T_{f0}/2 \\ T_{f0} = nT_f << T_{1m} \end{array} \tag{36}$$

Similar to the original HR method, MHR maintains the average residence time (ART) of the system (sum of time constants and delays). However, in the case of an n-fold time constant T_f of a filter, not only one dominant time constant is corrected, but by equal parts all the dominant time constants. The correction is using half of the filter ART $T_{f0} = nT_f$, not only half of one of its time constants T_f. As a result, the equivalent dead-time will be shorter than with HR, and the corresponding transients can be faster.

Thereby, such a simplification will be expected to yield expected results just for some limited T_{f0}.

Remark 2 (Conditions for a continuous-time domain application)**.** *The continuous-time-domain design methodology can also be applied to today's mostly digitally implemented controllers, provided that the sampling period* T_s *used is negligible compared to the smallest filter time constant, that is,*

$$T_s << T_f \tag{37}$$

By keeping this requirement, for given hardware limits on T_s *and some chosen* T_{f0} *yielding* $T_f \approx T_{f0}/n$*, the filter order n in* (36) *will always be limited.*

3. Refined Performance Measures

The aim to increase performance of SIMC control unavoidably requires refinement of the performance measures introduced in [4]. In the new setup of the optimal control design [20], one has to deal with a trade-off between speed of control error attenuation (measured usually in terms of integral of absolute error)

$$IAE = \int_0^\infty |e(t)|dt \; ; \;\; e = w - y \; ; \;\; w = setpoint, \tag{38}$$

Measurement noise injection (influencing primarily the "excessive control effort", denoted also as controller activity, or input usage, but including possibly also the "output wobbling" [19]) and the robustness.

To avoid undesired effects due to inadequate loop robustness, [6] used in the IAE optimization the sensitivity constraints, as, for example, defined by

$$max\{M_s, M_t\} = 1.59 \tag{39}$$

Thereby, M_s and M_t represent the peaks of sensitivity functions

$$\begin{array}{l} M_s = max\left\{\left|\frac{1}{1+L(j\omega)}\right|\right\} \; ; \; M_t = max\left\{\left|\frac{L(j\omega}{1+L(j\omega)}\right|\right\} \\ \omega \geq 0; \; L(s) = R(s)F(s) \end{array} \tag{40}$$

which correspond to PI control of a stable plant with $T_{c1} = T_d, T_i = T_1$.

Despite the fact that the use of M_s and M_t represents one of the pillars of the traditional robust design of PID control, it does not always lead to adequate conclusions [11]. Therefore, we replace the sensitivity constraints in this article with restrictions on the shape of transients at the input and output of the plant.

3.1. Ideal Shapes of Step Responses at the Plant Input and Output

In the era of relay minimum time control of *n*th-order systems, which brought the first systematic research of optimal control, the requirement to terminate the process in *n* (rectangular) pulses (control intervals) was used dominantly. It was first mentioned in work by Feldbaum [24]. From this point of view, later formulated modification resulting from maximum/minimum principle [25] brought differences just for the case of systems with complex poles for a large distance between the initial and final states. In the case of jOTD systems with real poles, the conclusions of Feldbaum's theorem could be formulated in a simplified form and without a proof as:

Theorem 1 (Feldbaum Theorem)**.** *For linear systems with j real poles and full relative degree $r = j$, after a step change of the reference setpoint or the disturbance, the number of control signal intervals required to achieve the neighbourhood of the desired state, with piece-wise alternating control signal limits, is equal to j.*

In engineering practice, we approach the minimum-time control only exceptionally. Greater emphasis is placed on smooth continuous changes of the control signal (manipulated variable) and well-damped steady states. In the minimum-time control developed especially for military applications (as missile control, fire control, etc.) the steady-states are frequently not considered. Therefore, due to different priorities, in the literature focused on PID controllers, the Feldbaum's theorem has been practically forgotten and has rarely been mentioned [26,27]. In addition, when controlling stable systems, the effectively observed number of pulses may be less than the order of the system under consideration. In this respect, it should be noted that with sufficiently smooth and slow processes, it is possible to fill the formulations of the following Lemma:

Lemma 1. *In each Bounded-Input-Bounded-Output (BIBO) stable system, after a step change of the reference setpoint or disturbance, a neighbourhood of the desired state may be achieved with monotonic setpoint step responses of the controller output (plant input) $u(t)$.*

Proof. With regard to a simpler explanation, consider a discrete-time control of a BIBO stable system implemented with some sampling period T_s. For such systems, it is always possible to find (sufficiently small) increments of input Δu such that the corresponding over-regulation (under-regulation) does not exceed an acceptable fraction of the required

deviation from monotonicity. After the transient is over, it is possible to add another input increment and wait for the corresponding transient to fade. Thus, by choosing sufficiently slow input increase, it is always possible to achieve (nearly) monotonic plant output response with a monotonic input and to extend such a control also to the continuous-time control with $T_s \to 0$. □

Although, with some exceptions such as above mentioned [26,27], which continued to use Feldbaum's theorem on optimal control also for smoother control responses, this feature of relay time-optimal controllers dealing with the number of optimal control pulses disappeared from the PID control over time. However, the ever-increasing demands placed on the performance of PID control lead to the fact that when evaluating its dynamics of transients, it will again be necessary to pay attention to control responses obtaining possibly a higher number of pulses. However, due to the requirement of smoothness of processes, these pulses may not be obvious at first glance and thus new performance measures may be required [28]. We will show this on the example of the system $1/(1+s)^4$ (33) (process E4 from [4]), from which we require monotonic closed loop setpoint step response with the time constant shortened from $T_4 = 1$ to $T_{c4} = 0.18$.

The unit setpoint step responses in Figure 2 illustrate that in order to achieve a smooth monotonic increase of y from initial to final steady state, the course of the first derivative $y^{(1)}$ must have one extreme at time $T_{1m} = 0.54$ and start and end with zero values. However, for the same purpose, the course $y^{(2)}$ must have a maximum at some point $T_{21} \in (0, T_{1m})$ and a minimum at $T_{22} \in (T_{1m}, \infty)$. As higher derivatives proceed, the number of extremes increases, with maxima alternating with minima and the time of the first maximum gradually decreasing. Finally, in the course of the highest derivative $y^{(4)}$ the first monotonic interval between $t = 0^-$ and $t = 0^+$ shrinks to zero. Furthermore, we are no longer able to optically distinguish extremes with higher indices due to high amplitude of initial peak.

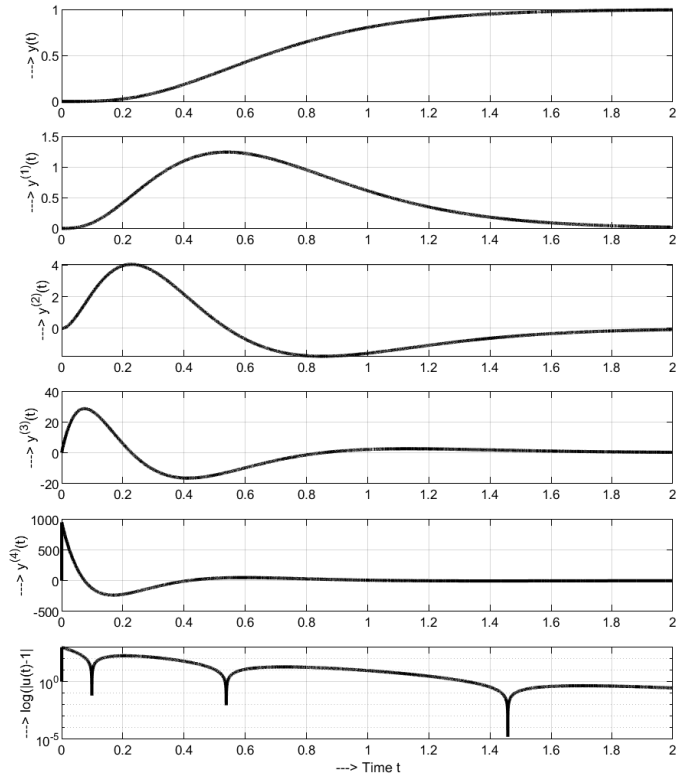

Figure 2. Unit setpoint step responses of system (33) for $T_{c4} = 0.18$ with a monotonic increase of the output $y(t)$ and four pulses (5 monotonic intervals) of the input $u(t)$, the first interval of $u(t)$ increase between $t = 0^-$ and $t = 0^+$ shorten to zero.

The same applies to the course of $u(t)$, which, by alternating increases and decreases, affects the sign of the highest derivative. Therefore, in order to illustrate the increase and decrease of $u(t)$ and the corresponding pulses of control with respect to the steady value $u_\infty = 1$, the course of $log(u(t) - 1)$ can be used. The peaks of this response illustrate the change in the sign of $u(t) - 1$ and document that an alternative of the Feldbaum's theorem can also be formulated for a smooth control signal $u(t)$. However, the decreasing amplitudes of extremes with higher indices, together with the Lemma 1, suggest that nearly optimal responses can also be achieved by simpler control with lower number of pulses.

We encounter similar piece-wise monotonic responses achieved with the same controller when evaluating input disturbance step responses in Figure 3. However, the ideal output response begins with a 1P shape, and the ideal system input $u(t)$ is monotonic in this case.

To simplify the relevant formulations, we will introduce the concept of m-pulse responses in the following.

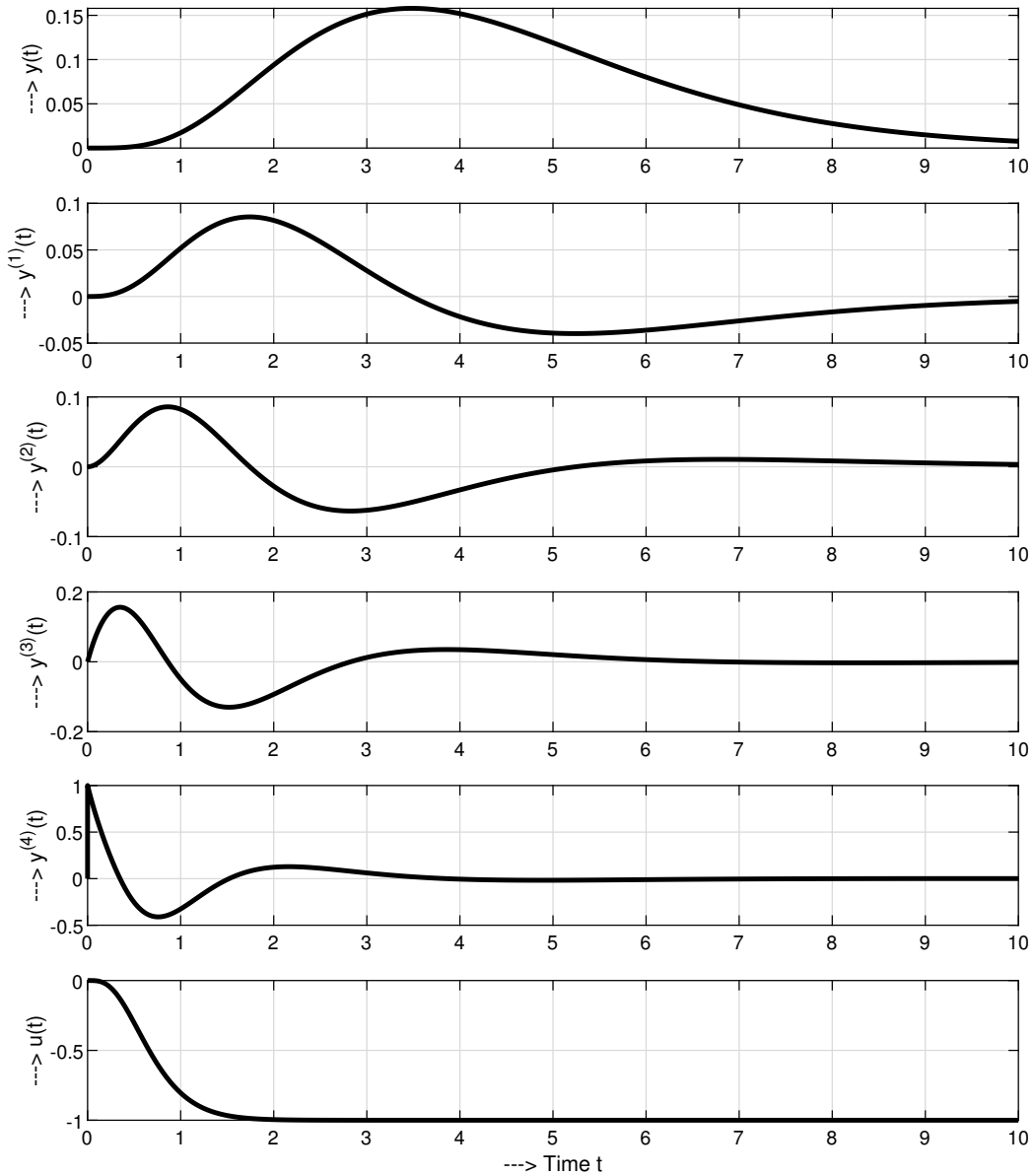

Figure 3. Unit input disturbance step responses of system (33) for $T_{c4} = 0.18$ with 1P output $y(t)$ and monotonic input $u(t)$.

Definition 4 (*mP* function $u(t)$)**.** *Let $u(t)$ be a function corresponding to the closed loop control signal of a stable jth-order linear time-delayed system that is:*

- *continuous for $t \in (0, T), T \to \infty$,*
- *with possible discontinuity at $t = 0^+$ and*
- *with initial value $u_0 = u(0^-)$ and final steady-state value $u_T = u(T)$.*

Let in a step response $u(t)$ can be found for $0 < t < T$ m extremes ($0 \leq m \leq j$) lying alternately over and below (or vice versa) the level u_T (as shown in Figure 4) and fulfilling (with the denotation $u_i = u(t_i)$; $i = 1, 2, ..., m$ for $0 < t_1 < ... < t_m$) conditions

$$(u_i - u_T)\ (u_{i+1} - u_T) < 0;\ i = 1, 2, ..., m - 1 \tag{41}$$

Then the function $u(t)$, which is monotonic on each of the $m + 1$ intervals not containing one of the above extreme points $u_i, i = 1, 2, ..., m$, is called as m-Pulse (mP) function.

In case of discontinuity at $t = 0$, the first extreme point can also be moved to the origin $t = 0^+$ (i.e., $u_1 = u(0^+)$, thus shrinking the first monotonic interval between u_0 and u_1 with $t \in (0^-, 0^+)$ to zero.

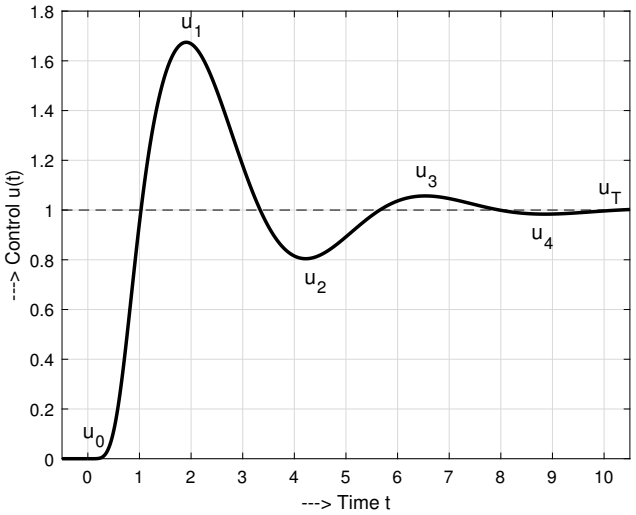

Figure 4. 4P control signal response with extreme values u_1, u_2, u_3 and u_4 outlining 5 monotonic intervals.

According to this terminology, monotonic transients can also be referred to as 0P and periodic responses as ∞P functions.

As we will show in the solved examples, the above described model-based control can tend to high initial peaks of the control signal. Such controllers may not be feasible in practice. However, as will be shown later, the initial peaks can be alleviated by a modified controller design.

Definition 5 (Dynamical classes of control)**.** *Dynamical class of control DCN [17] is used to denote control with all setpoint step responses given by NP input-0P output pairs $u(t), y(t)$. In other words, it denotes setpoint step responses with the plant input $u(t), t \in (0, \infty)$ consisting of $N + 1$ alternately monotonically increasing and decreasing (or vice versa) segments, which are associated with the monotonic plant output $y(t), t \in (0, \infty)$.*

In the case of the disturbance step responses, it denotes NP input-1P output pairs $u(t), y(t)$ with the plant input $u(t), t \in (0, \infty)$ consisting of $N + 1$ monotonic segments, which are associated with the plant output $y(t), t \in (0, \infty)$ consisting of two monotonic response segments.

Thus, setpoint step responses of DC0 includes transients with a monotonic course of the control signal corresponding according to Lemma 1 to a monotonic response at the output. The disturbance step responses of DC0 consist of two monotonic intervals at the output associated with a monotonic change of the control signal.

Similarly, the setpoint step responses of DC2 include control signal $u(t)$ transient with 3 monotonic intervals and a monotonic process output response $y(t)$. The disturbance step responses of DC5 consist of two monotonic intervals of the output signal and 6 monotonic intervals of the control signal.

As illustrated by input disturbance step responses in Figure 3, the same controller as considered in Figure 2 may give setpoint and disturbance step responses from fully different dynamical classes.

Definition 6 (Symmetric controller design)**.** *A symmetric controller design assumes setpoint and disturbance step responses from the same dynamic class.*

The question of the symmetry of the dynamics of setpoint and disturbance responses is of interest to us mainly in terms of their applicability, when at the same controller setting, diametrically different requirements regarding the amplitudes of the control signal may be placed on the actuators. However, it should also be remembered that because the considered model-based approach is based only on the requirements formulated for setpoint responses (1), we can modify the disturbance responses only indirectly.

3.2. Shape Related Performance Measures for Useful/Excessive Output Increments

IAE values used for evaluating speed of the transientsmay be applied both in analytical derivations and in experimental evaluation of the controller design. Since the setpoint step responses can also be improved by an appropriate feedforward, the analysis will preferably focus on the input (load) disturbance step responses given fully by the feedback controller. The achieved responses strongly depend on possible uncertainty of the considered plant model, with an uncertainty impact similar to external disturbances [29]. This gives additional motivation to deal with the disturbance responses, as an indispensable part of the robustness analysis.

Together with the requirement to have the output responses as fast as possible, it is also necessary to consider shapes of actual output and input signals. Thus, although a minimum of (38) usually corresponds to a slight output overshooting, in numerous applications, an ideal setpoint step response has to be monotonic. This corresponds to a situation, when the sum of absolute values of all output increments $\sum_i(|y_{i+1} - y_i|)$ equals to the net output change $|y_\infty - y_0|$ specified by the initial and the final values y_0 and y_∞. Therefore, to characterize the output deviations from monotonicity, the excessive output variation (the total sum of absolute increments [4] reduced by the useful output change) can be used

$$TV_0(y) = \int_0^\infty \left(\left| \frac{dy}{dt} \right| - sign(y_\infty - y_0) \frac{dy}{dt} \right) dt \approx \sum_i(|y_{i+1} - y_i|) - |y_\infty - y_0| \tag{42}$$

Such "excessive" increments yield the best view on the "smoothness" of the output response: the ideally smooth monotonic output change corresponds to $TV_0(y) = 0$, else $TV_0(y) > 0$. By limiting the deviations from the monotonicity, we limit also the magnitude of the maximum overshooting, as well as the permanent oscillations of the system.

Similarly as considered in [30] for FOTD plants, an ideal input disturbance step response of the considered stable jOTD plants has always the shape of an one-pulse (1P) curve at the plant output (see $y(t)$ in Figure 3). It means that after eliminating imbalance due to a disturbance step change by a corresponding manipulated variable change, the output stops to diverge and then it monotonically returns to the desired reference value. Two monotonic intervals of such a 1P disturbance rejection process are thereby separated by an extreme point $y_m \notin (y_0, y_\infty)$ and the monotonicity evaluation according to (42) has to be applied twice. Output deviations from an ideal 1P behavior summarize the deviations from monotonicity on these two intervals

$$TV_1(y) = \sum_i |y_{i+1} - y_i| - |2y_m - y_\infty - y_0| \tag{43}$$

In case of several extreme points outside of the strip $y \in (y_0, y_\infty)$, the maximal deviation has to be chosen as y_m.

3.3. Shape Related Performance Measures for Useful/Excessive Input Increments

A much more complex situation occurs, when evaluating the optimality of the input variables of the considered stable jOTD plants.

As shown in [28,30], in case of the single integrator the controller output (plant input) corresponding to the setpoint and input disturbance steps have to ideally consist of two monotonic intervals forming an 1P shape (as $y^{(1)}(t)$ in Figure 2). Then, similarly as above, deviations of the plant controller output $u(t)$ from an ideal 1P step response should be constrained in terms of $TV_1(u)$ measures. However, for control of the stable FOTD plants it may be meaningful (see Lemma 1) to consider also input with lower number of control pulses, that is, with a monotonic shape and to consider its evaluation using $TV_0(u)$ measure. The decision regarding the choice of $TV_0(u)$ or $TV_1(u)$ is entirely up to the designer and his subjective evaluation of the specific features of the application.

When it comes to controlling the position of the moving mass represented in mechatronics by integrative second-order models, a 2P (two-pulse) input (similar to $y^{(2)}(t)$ in Figure 2) is already needed to achieve a monotonic change in output. It will be dominated with two extremes, one for the acceleration and one for the braking phases. These extremes are separating the total input response into 3 possible monotone sections [31]). Unlike a chain of integrators, when controlling stable 2nd -order systems, the monotonic course of the output can be achieved by 2P, 1P, but also by 0P input.

Similarly, to control a chain of j integrators, an jP type input composed of $j+1$ monotonic sections is needed. However, the situation is complicated by the fact that this time we are not working with integrators, but stable systems, the resulting behavior of which may be more varied and represented by mP functions specified in Definition 4.

Remark 3 (Shape requirements on input of stable jOTD systems)**.** *When controlling stable jOTD systems, the input may approach the jP signal (with $j+1$ monotonic intervals) only at high, nearly minimum-time control requirements, which correspond to $T_d \to 0$, a negligible measurement noise and a negligible plant uncertainty. Else, the number of monotonic intervals will be lower, in the limit case just one (with no extreme point). As a result, accurate evaluation of deviations from the ideal input shapes represents an open problem, when controlling stable jOTD systems, since the number of pulses and monotonic segments varies and may be different for the setpoint and disturbance step responses. With a higher level of noise and uncertainty, deviations from monotonicity using $TV_0(u)$, or deviations from the 1P signal using $TV_1(u)$ will usually suffice. However, this does not exclude even more complex situations with a higher number of pulses or monotone intervals of the control signal, which makes the use of a TV performance measure in the form introduced in [4] questionable. This evaluates more frequent changes to the control signal as inappropriate and unwanted.*

For example, for the transient in Figure 4 with values

$$u_0 = 0;\ u_1 = 1.68;\ u_2 = 0.80;\ u_3 = 1.05;\ u_4 = 0.98;\ u_T = 1; \tag{44}$$

The traditional "input usage" evaluation yields traditional and modified total variation values

$$TV = 2.9;\ TV_0 = 1.9;\ TV_1 = 0.54;\ TV_2 = 0.14;\ TV_3 = 0.04;\ TV_4 = 0. \tag{45}$$

The deviation from monotonicity would be calculated as $TV_0 = TV - |u_T - u_0| = 1.9$. Of course, transients with monotonic input (required for some special applications) may be too slow in a general case. Therefore, frequently a more active control is required. The deviation from 1P response (required typically in PI control) is already much less, just $TV_1 = TV - |2u_1 - u_T - u_0| = 0.54$. The deviation from 2P calculated as $TV_2 =$

$TV - |2u_1 - 2u_2 + u_T - u_0| = 0.14$ is yet lower. Similarly, $TV_3 = TV - |2u_1 - 2u_2 + 2u_3 - u_T - u_0| = 0.04$ representing deviations from 3P shape. Finally, we may conclude that for the 4th order system, which was under control, the analysed response with $TV_4 = TV - |2u_1 - 2u_2 + 2u_3 - 2u_4 + u_T - u_0| = 0.0$ represents a fully effective 4P control, without an excessive effort. An optimization process based on TV value would never propose such a solution, even if it is very fast and without excessive control increments. Again, we recall that the decision regarding the choice of a preferred shape related performance measure is entirely up to the designer and his subjective evaluation of the specific features of the application.

Of course, the above calculation refers to ideal waveforms without the influence of measurement noise or imperfect controller settings due to uncertainties, imperfect identification, or time changes of the controlled system. In order to be able to address the influence of individual design factors on the achieved control performance, the evaluation of the circuit will probably need to be divided into several phases.

3.4. First Evaluation Step—Idealized Situation with No Noise

For testing impact of the controller tuning it is recommended to start with an idealized situation without measurement noise and no uncertainties in the plant dynamics. Then, if the control error does not change its sign (which is fulfilled for $TV_0(y_s) = 0$ and $TV_1(y_d) = 0$), the IAE values (38) may be calculated as the integral of error (IE). Then, from application of Laplace transform of $e(t)$ denoted as $E(s)$ follows

$$IAE = IE = \int_0^\infty e(t)dt = \lim_{s\to 0} E(s) = E(0); \quad e(t) = w(t) - y(t) \tag{46}$$

For the unit setpoint and input disturbance steps, for all above nominal controllers follows from

$$E_s(s) = 1/[(1 + {}^jF(s)\,{}^jR^j(s))s];\ E_d(s) = {}^jF(s)/[(1 + {}^jF(s)\,{}^jR^j(s))s]; \tag{47}$$

values

$$IAE_s = \frac{T_i}{K_c K};\ IAE_d = \frac{T_i}{K_c} \tag{48}$$

In ideal situations, with zero shape related deviations at the output, for controllers based on models ${}^1F(s) - {}^4F(s)$ we get

$$\begin{aligned} &{}^1IAE_s = T_{c1} + T_d;\ {}^1IAE_d = K(T_{c1} + T_d) \\ &{}^2IAE_s = 2T_{c2} + T_d;\ {}^2IAE_d = K(2T_{c2} + T_d) \\ &\quad\vdots \\ &{}^jIAE_s = jT_{cj} + T_d;\ {}^jIAE_d = K(jT_{cj} + T_d) \end{aligned} \tag{49}$$

Thereby, the figure $jT_{cj} + T_d$ corresponding to the target transfer functions with j-tuple time constants T_j denotes the average residence time (ART) of the closed loop system [1]. This is once more to stress that above formulas hold just in the nominal case. In practical applications, they will also depend on the accuracy of the system model used.

3.5. Optimization Problem

Let us continue with summarizing basic facts:

1. Traditional optimization based on quadratic cost functions (LQ control design) does not distinguish useful and excessive signal increments which significantly limits effectiveness of its application.
2. Similarly, the use of TV to evaluate control efforts does not distinguish between useful and redundant increments of control signal. This can cause a problem especially when controlling higher order systems and requiring several active impulses of control.

3. Separation of the excessive and useful increments (both at the input and output) enables to focus fully on an effective minimization of the superfluous changes.
4. In application to evaluation of the setpoint step responses of the plant output y_s, the modified performance measure $TV_0(y_s)$ (42) has a clear mathematical and physical interpretation as a deviation from monotonicity.
5. In the new setup of the optimal control design [20], one has to deal with a trade-off between speed of control error attenuation (IAE), measurement noise injection resulting into "excessive control effort" ("controller activity/input usage" [6], or the "output wobbling") and "robustness".
6. Optimal controller and filter tuning is expected to depend on the noise parameters. Thus, without considering filtration properties, a "generally" optimal PID tuning becomes questionable.

For the loop optimization, different cost functions and different optimization constraints may be defined. A "holistic" loop optimization requiring for the plant model ${}^jF(s)$ fast and smooth transients, that is, considering both the plant input and output, may be looking for a minimal value of the cost function

$$ {}^jJ_k(u) = IAE^k\ TV_j(u) \tag{50}$$

By the parameter k it is possible to weight contributions of IAE (speed of control) into the resulting product. The problem remains how to consistently compare the shape of transients proposed using models of different orders j.

A simpler situation occurs during the evaluation with the task to minimize the output wobbling. For the setpoint and disturbance step responses the cost functions may be defined as

$$J_k(y_s) = IAE_s^k\ TV_0(y_s);\ J_k(y_d) = IAE_d^k\ TV_1(y_d) \tag{51}$$

Applications of above measures to dominant first-order plants ($j = 1$) may be found in [10,11].

3.6. Speed-Effort and Speed-Wobbling Characteristics

Impact of chosen tuning parameters on the trade-off between the speed of control and the shape related deviations at the input and output, may be illustrated by several types of characteristics.

In this paper, they will be based on

- the shape related deviations at the input or output, that is, the measures expressing, how far are the measured transients from their ideally required shapes (variable ξ) and
- IAE measure characterizing the speed of the control error attenuation (variable η)

Dependence of these two basic measures of the closed loop performance, when either

$$\begin{array}{l} \xi = TV_1(u),\ \eta = IAE^k, \text{ or} \\ \xi = TV_0(y_s)\ \eta = IAE_s^k \text{ or } \xi = TV_1(y_d),\ \eta = IAE_d^k \end{array} \tag{52}$$

Define two types of the loop characteristics. Into the PID controller design they have been introduced in [32]. Here, they will be denoted as the *speed-effort* (SE) and *speed-wobbling* (SW) characteristics.

3.7. IAE-Optimization-Based Tuning of Noisy FOTD Plants

Whereas we may agree with great part of conclusions of [6], one of the basic problems of the IAE-optimization-based "improved" SIMC rules [6] seems to be that, once wishing to be rigorous, the optimization has to be repeated for each new set of parameters of FOTD plant (3) and filter (35). Therefore, it does not allow simple analytical filter consideration. The only possibility is to simplify the plant description (e.g., by application of MHR) up to an integral model. However, such a consideration of the noise attenuation filters (added

by the half-rule to the plant dead-time) changes the actual sensitivity values, which again makes use of the optimization-based approaches questionable.

4. Modified Controllers with Reduced Initial Control Signal Peak

One of the shortcomings of the traditional model-based approach is that it does not pay higher attention to the achieved shapes of the control signal responses. Reference [4] only mentions the possibility to avoid derivative kick on setpoint change by following industry practice and differentiating only the output of the system. However, such a solution is not as efficient in terms of adapting the dynamics of the closed-loop responses when compared to pre-filters, or setpoint weighting used in two-degree-of-freedom (2DOF) PID control. We will show that an effective solution to this problem is also related to the actual distribution of dynamic elements of a feedback control system.

As we have already shown above, the model-based design presented in the introductory sections does not provide the same setpoint and disturbance rejection dynamics. During the setpoint step changes, the excessive initial control signal kicks may not be feasible in practice. In further derivations we will show, how to adjust the control signals by using the pre-filters P added to the circuit in Figure 1. We will use an auxiliary system in Figure 5 (firstly with $F_P(s) = 1$), which has a clearly defined optimal shapes of the control signal $u(t)$ and of a hypothetical output y_0 for a setpoint step. This auxiliary system still includes the same jOTD model as considered previously. However, the setpoint response related to its output $y_m = y_j$ will be different than it should correspond to the required target closed-loop transfer function

$$ {}^{j}F_{cl}(s) = \frac{Y(s)}{W(s)} = \frac{e^{-T_d s}}{(1 + T_{cj}s)^j}. \tag{53} $$

Finally, we show the impact of modifications of the pre-filter P (e.g., its omission equivalent to $P = 1$) on the equivalent pre-filter F_P and the shape of the transients in the auxiliary circuit.

Lemma 2 (Equivalence of model-based controllers according to Figures 1 and 5)**.** *For stable systems that can be approximated by jOTD model* (34) *with a j-tuple time constant T_j, an auxiliary control structure with the setpoint and disturbance step responses of DC0 may be specified according to Figure 5 with*

$$ {}^{j}F_p(s) = 1;\ {}^{j}F_y(s) = \frac{U_{af}(s)}{Y_m(s)} = \frac{1}{K}\left(\frac{1 + T_j s}{1 + T_f s}\right)^j;\ {}^{j}F_u(s) = \frac{U_f(s)}{U(s)} = \frac{e^{-T_d s}}{(1 + T_f s)^j}. \tag{54} $$

This can be transformed to the structure according to Figure 1 with the controllers $R(s) = {}^{j}R^{j}(s)$

$$ {}^{j}R^{j}(s) = \frac{(1 + T_j s)^j}{K[(1 + T_{cj}s)^j - 1 + T_d s]} \tag{55} $$

and the the pre-filter $P(s) = {}^{j}P(s)$

$$ {}^{j}P(s) = \left(\frac{1 + T_{cj}s}{1 + T_j s}\right)^j. \tag{56} $$

Thereby, controllers ${}^{j}R^{j}(s)$ coincide with the solutions defined for $j \in [1, 4]$ by the Equations (6), (7), (21), (28) *and* (32).

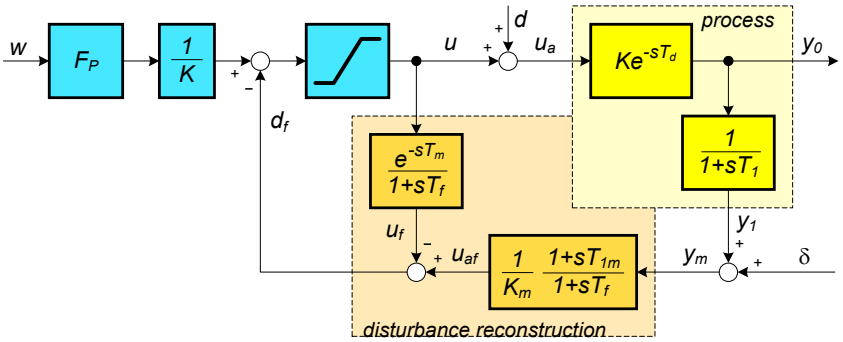

Figure 5. Static feedforward with input disturbance reconstruction and compensation and a hypothetical process dynamics decomposition into the feedforward and feedback path drawn for a FOTD system; δ-measurement noise.

Proof. Given the simplicity and wide use of the PI controller, we start the proof with of $j = 1$, ${}^1F_p(s) = 1$ (see Figure 5). The constructive approach gives the simplest controller of DC0 for the FOTD plant model in the feedback of the auxiliary system. Thereby, we will not assume that the measured output signal y_m represents directly the controlled output y. In this hypothetical situation considering a memoryless system with a (possibly dominant) transport delay in the direct control branch, that is, with $y = y_0$. All explicitly considered (stable) time constants of the jOTD model will be located in the feedback and interpreted as sensor dynamics or noise attenuation filters. Although such a situation seem to be rare in practical applications, in considering both the distribution of the dynamical terms in the loop, as well as the shapes of particular loop signals, it provides a fixed point from which we can start when analyzing shapes of the signals in Figure 1 even at higher values of j. This controller scheme drawn for $j = 1$ can be considered as a generalization of the historically first dead-time compensator from Reswick [33].

With a sufficiently long filter time constant T_f preventing exaggerated responses to possible disturbances, after a step change of the reference setpoint signal w, we get both a step change of the controller output $u(t)$ and the plant output y, as well as a monotonic (exponential) change of the measured output y_m. Since the step change represents a limit case of monotonic changes, in terms of both considered outputs y_0 and y_1, the circuit belongs to DC0 and will remain in it also for the plant time constant T_1 moved to the direct branch (i.e., for $y = y_1$).

In a nominal case, with the dead-time estimate $T_m = T_d$ and the model time constant $T_{1m} = T_1$, we can denote the observer transfer function based on filtered inversion of the process dynamics as

$$ {}^1F_y(s) = \frac{U_{af}(s)}{Y_m(s)} = \frac{1 + T_1 s}{K(1 + T_f s)}. \tag{57}$$

Estimate of the process dead-time and the used observer filter are included in

$$ {}^1F_u(s) = \frac{U_f(s)}{U(s)} = \frac{e^{-T_d s}}{1 + T_f s}. \tag{58}$$

For $y = y_0$, this control structure provides the closed loop transfer functions

$$ {}^1F_{wy0}(s) = \frac{Y_0(s)}{W(s)} = e^{-T_d s};\ {}^1F_d(s) = \frac{Y_0(s)}{D(s)} = Ke^{-T_d s}\left(1 - \frac{e^{-T_d s}}{1 + T_f s}\right). \tag{59}$$

They guarantee both the setpoint and the disturbance responses from DC0. Thereby, tha auxiliary loop has clearly defined shapes of all internal signals corresponding to step inputs: the setpoint steps lead to step changes of $u(t)$ and $y_0(t)$ and to an "open-loop" response of $y_1(t)$. The local loop with a positive feedback via ${}^1F_u(s)$ may be replaced (when substituting for T_d (5)) by

$$ {}^1S_u(s) = \frac{1}{1 - {}^1F_u(s)} = \frac{1 + T_f s}{(1 + T_f s) - e^{-T_d s}} \approx \frac{1 + T_f s}{(T_f + T_d)s}. \tag{60} $$

Moving ${}^1F_y$ from the feedback to the direct branch, merging with ${}^1S_u$ and the inverse gain $1/K$, is then leading with $T_f = T_{c1}$ to the controller (6) and (7). The difference, however, is that this time we also got a pre-filter

$$ {}^1P(s) = \frac{1 + T_f s}{1 + T_1 s} = \frac{1 + T_{c1} s}{1 + T_1 s}. \tag{61} $$

From the design, which is to compensate for the shift of ${}^1F_y$ from the feedback to ${}^1R^1$ located in the direct control path. In addition, from (59) we know that for a sufficiently large value of $T_f = T_{c1}$, the disturbance reconstruction will not spoil the step character of the setpoint step responses of the variables u and y_0, whereby $y_m = y_1$ remains monotonic. Both the setpoint and the disturbance responses retain the monotonic character even for the time constants T_1 located in the direct control branch (i.e., with $y = y_1$).

Next, we derive the design of the controller for the SOTD system (19) and show again that in addition to the controller (21) itself, the design of a suitable pre-filter must also be considered. From the derivation we can then easily come to generalizations for higher order systems.

For a SOTD nominal process (19) in the feedback loop of an auxiliary system with ${}^2F_p(s) = 1$

$$ {}^2F_y(s) = \frac{U_{af}(s)}{Y_m(s)} = \frac{(1 + T_2 s)^2}{K(1 + T_f s)^2} \tag{62} $$

$$ {}^2F_u(s) = \frac{U_f(s)}{U(s)} = \frac{e^{-T_d s}}{(1 + T_f s)^2} \tag{63} $$

and for $y = y_0$, this control structure provides the closed loop transfer functions

$$ {}^2F_{wy0}(s) = \frac{Y_0(s)}{W(s)} = e^{-T_d s};\ {}^2F_d(s) = \frac{Y_0(s)}{D(s)} = K e^{-T_d s}\left(1 - \frac{e^{-T_d s}}{(1 + T_f s)^2}\right). \tag{64} $$

Guaranteeing both the setpoint and the disturbance responses from DC0. Ideally, after a setpoint step, $u(t)$ and $y_0(t)$ also show a step change and $y_2(t)$ corresponds to an "open-loop" monotonic response. By replacing the local loop with a positive feedback via ${}^2F_u(s)$ we get

$$ {}^2S_u(s) = \frac{1}{1 - {}^2F_u(s)} = \frac{(1 + T_f s)^2}{(1 + T_f s)^2 - e^{-T_d s}}. \tag{65} $$

Moving ${}^2F_y$ from the feedback to the direct branch, merging with ${}^2S_u$ and the inverse gain $1/K$, substituting for T_d (5) and $T_f = T_{c2}$, is then leading to the controller (21). Again, the difference is that this time we also got a pre-filter

$$ {}^2P(s) = \left(\frac{1 + T_f s}{1 + T_2 s}\right)^2 = \left(\frac{1 + T_{c2} s}{1 + T_2 s}\right)^2. \tag{66} $$

In addition, from (64) we know that for a sufficiently large value of $T_f = T_{c2}$, the disturbance reconstruction will not spoil the step character of the setpoint step responses of the variables u and y_h, whereby y_m remains monotonic. Both the setpoint and disturbance responses retain the monotonic character even for both time constants T_2 located in the direct control branch (i.e., with $y = y_m$). □

In Lemma 2, the model-based controller design from previous sections derived for jOTD models (34) using the target transfer function (53) has been compared with an auxiliary system with clearly defined shapes of all internal variables. As a result, the

control structure of DC0 according to Figure 5 has been shown to be equivalent to the structure according to Figure 1 with the controller $R(s) = {}^{j}R^{j}(s)$ and the the pre-filter $P(s) = {}^{j}P(s)$ (56). However, the equivalence assumed a step in the control signal after the setpoint steps, which is far from suitable for all applications. The possibilities of further modifications of setpoint responses are described by the following theorem.

Theorem 2 (Pre-filter design for model-based control)**.** *When omitting pre-filter in Figure 1 (i.e., working with $P(s) = 1$, as in SIMC design [4]), although the output of the j-tuple time time constant T_j fulfills behavior prescribed by the target transfer function* (53)*, the control signal may not be feasible for smaller values of T_{cj} . When applying an equivalent change defined by $F_p = P^{-1}$ to the auxiliary system, the target transfer function* (53) *used for design of ${}^{j}R^{j}(s)$ in the structure according to Figure 1 will be matched by the measured signal $y_m = y_j$ in the structure according to Figure 5.*

To get smooth setpoint step responses without initial kicks of the control signal, the pre-filter $P(s)$ has to be simplified to a strictly proper transfer function. Thereby, by using the same controllers $R(s)$ with lower order pre-filters $P(s)$ than given by (56)*, it is possible to speed up the setpoint step responses of the structure according to Figure 1, which then already correspond to a higher dynamic class DCN, $0 < N \leq j$.*

Proof. Omitting the pre-filter $P(s) = {}^{1}P(s)$ from the structure in Figure 1 with $R(s) = {}^{1}R^{1}(s)$ corresponds to a modified pre-filter $P(s) = {}^{1}P(s)({}^{1}P(s))^{-1} = 1$. Therefore, it is equivalent to adding the pre-filter ${}^{1}F_p(s) = ({}^{1}P(s))^{-1} = (1 + T_1 s)/(1 + T_{c1} s)$ to the structure from Figure 5. Then, $U(s)/W(s) = (1 + T_1 s)/[K(1 + T_{c1} s)]$, $Y_0(s)/W(s) = (1 + T_1 s)e^{-T_d s}/(1 + T_{c1} s)$ and the transfer function $Y_1(s)/W(s) = e^{-T_d s}/(1 + T_{c1} s)$ matches exactly the target transfer function (4). However, the course $u(t)$ will no longer have the shape of a step and, for relatively short T_{c1} values, high peaks with the amplitude $U_{max} = \lim_{t \to 0} u(t) = \lim_{s \to \infty} sU(s) = T_1/(KT_{c1})$ may occur in it after unit setpoint steps.

Therefore, where appropriate, the initial kicks of the setpoint step responses $u(t)$ of both considered structures may be completely eliminated by using the pre-filters

$$ {}^{1}P(s) = \frac{1}{1 + T_1 s}; \; {}^{1}F_p(s) = \frac{1}{1 + T_{c1} s}; \tag{67} $$

Definition 7 (Controller ^{0}PI)**.** *With respect to the dynamical class DC0 of the achieved step responses, controller* (6) *and* (7) *extended by the pre-filter* (67) *could also be denoted as ${}^{0}PI$.*

^{0}PI yields smoother monotonic $u(t)$ course from DC0, that is, without overshooting, which may be important in design of control respecting given constraints, or in design of systems with hysteresis. The setpoint step response at the output of FOTD model $y_1(t)$ are, however, slower than required by (4). They will not be faster than the open-loop FOTD responses. Two dynamical classes of PI control have been firstly identified in [34].

Experimentally, the monotonic step responses of ^{0}PI can yet be accelerated by decreasing the pre-filter time constants in (67), which could lead to transients going on with an acceptable overshooting of $u(t)$.

Similarly, for SOTD models, the step character of the setpoint step responses $u(t)$ of the structure according to Figure 5 can be replaced with a smoother (but slightly slower) continuous course of $u(t)$ by simplifying the pre-filters according to

$$ {}^{2}P(s) = \frac{1}{(1 + T_2 s)^2}; \; {}^{2}F_p(s) = \frac{1}{(1 + T_{c2} s)^2}. \tag{68} $$

Definition 8 (^{0}PID controllers for lag and dead-time dominant plants)**.** *PID Controller* (21) *extended according to Figures 1 and 5 by the pre-filters $P(s)$ and $F_p(s)$* (68)*, which at the plant input and output yield setpoint, or monotonic step responses of DC0, may be denoted as ${}^{0}PID$ controllers.*

However, ^{0}PID controllers with monotonic step responses from DC0 can also be designed on the basis of the ^{1}PID controller (12) supplemented by the pre-filters (67).

While the use of a solution based on SOTD models can be expected to be advantageous for controlling lag-dominant processes, a controller based on FOTD models may appear to be more advantageous for controlling dead-time dominant processes.

As a compromise between (66) and (68) we can choose $P(s)$ in the form (68), but of a lower order, thus speeding up the setpoint responses. These can then belong to a higher dynamic class.

We proceed similarly for other values of j (e.g., in designing ^{0}PIDA controllers), with generalizing multiplicity of the numerator and denominator of the pre-filters (66), or (68) to

$$^jP(s) = \frac{1}{(1+T_js)^j};\ ^jF_p(s) = \frac{1}{(1+T_{cj}s)^j}. \tag{69}$$

Although circuits with relatively large $T_f = T_{cj}$ values are the most resistant to uncertainties and noise and do not attack the control signal constraints, it may be interesting to use smaller values. As T_f decreases, the speed of setting the signal $y = y_m$ to the desired reference value w increases. Situations with monotonic responses $y = y_m$ achieved under $u(t)$ with $0 \le N \le j$ extremes then correspond to control from a higher dynamic class DCN. This can, however, require to consider control signal constraints and to use appropriate anti-windup schemes [35–38], or more advanced design methods [39,40]. □

4.1. Integrative Controllers for the Simplest Pure Dead-Time Plant Models

To complement the family of integrative controllers, we will include even simpler process models with pure delay. The control structure in Figure 5 can also be advantageously modified for situations where the time constants $T_{1m} << T_d$ can be neglected ($j = 0$) when leading to a predictive integrative (I) controller. In such a situation, with a first order reconstruction filter

$$\begin{aligned} &^0F_{cl}(s) = \frac{Y(s)}{W(s)} = e^{-T_ds};\ ^0F_d(s) = \frac{Y(s)}{D(s)} = Ke^{-T_ds}\left(1 - \frac{e^{-T_ds}}{1+T_fs}\right) \\ &^0P(s) = 1 + T_fs;\ ^0F_p(s) = 1 + T_fs. \end{aligned} \tag{70}$$

By omitting the not feasible pre-filters ^0P and ^0F_p in the auxiliary circuit according to Figure 5 and in Figure 1, we will replace the step-wise responses of $u(t)$ and $y(t)$ by smoother exponential transients, which correspond to:

$$^0F_{cl}(s) = \frac{Y(s)}{W(s)} = \frac{e^{-T_ds}}{1+T_fs};\ ^0P(s) = 1;\ ^0F_p(s) = 1. \tag{71}$$

It also means that the tuning parameter is $T_{c0} = T_f$.

For the dead-time approximation (5), the approach gives equivalent loops with I-controller

$$\begin{aligned} &^0R^1(s) = \frac{U(s)}{E(s)} = \frac{K_i}{s};\ K_i = \frac{1}{K(T_f+T_d)} \\ &^0F_{cl}(s) = \frac{Y(s)}{W(s)} = \frac{e^{-T_ds}}{(T_f+T_d)s + e^{-T_ds}};\ ^0F_d(s) = \frac{Y(s)}{D(s)} = K(T_f+T_d)\frac{se^{-T_ds}}{(T_f+T_d)s + e^{-T_ds}}. \end{aligned} \tag{72}$$

If we continue in this way using the Padé approximation (10), we get another integrative controller that might be denoted as a filtered ^{0}PI controller, or ^{0}I-PD controller given by the equations

$$
\begin{aligned}
&{}^0R^2(s) = \frac{U(s)}{E(s)} = K_c\frac{1+T_is}{T_is}\frac{1}{1+T_{f1}s} = \frac{K_i}{s}\frac{1+T_Ds}{1+T_{f1}s};\\
&K_c = \frac{T_i}{K(T_f+T_d)};\ T_i = \frac{T_d}{2}; K_i = \frac{1}{K(T_f+T_d)};\ T_D = \frac{T_d}{2};\ T_{f1} = \frac{T_fT_d}{2(T_f+T_d))}.
\end{aligned}
\tag{73}
$$

It may be used with pre-filters (71).

The 2nd-order Padé approximation (13) yields similarly solution that may be denoted as ^{0}I-PDD^2 controller

$$
\begin{aligned}
&{}^0R^3(s) = \frac{U(s)}{E(s)} = \frac{K_i}{s}\frac{1+T_{D1}s+T_{D2}s^2}{1+T_{f1}s+T_{f2}s}\\
&K_i = \frac{1}{K(T_f+T_d)};\ T_{D1} = \frac{T_d}{2};\ T_{D2} = \frac{T_d^2}{12};\ T_{f1} = \frac{T_fT_d}{2(T_f+T_d))};\ T_{f2} = \frac{T_fT_d^2}{12(T_f+T_d))}.
\end{aligned}
\tag{74}
$$

For the pre-filter design applies the same as for the I-controller.

Of course, the solutions based on the elimination of dead-time from the controller structure had its justification in the time of analog controllers. Today, it might be more advantageous to implement this control based on a default scheme with disturbance reconstruction and compensation in Figure 5. Especially in the simplest considered situation, such a "predictive" integrative (I) controller can be significantly more efficient than an I controller. However, it can be effectively and robustly tuned just by the performance portrait method [4,41,42].

4.2. Pre-Filter Design for the SIMC PID Controller

When applying the above procedure to the SIMC PID for the plant $Ke^{-T_ds}/[(1+T_1s)(1+T_2s)]$ according to [4], it must be taken into account that this design leads to an ideal controller (with non-causal target transfer function). Thus, as in Figure 5, the first-order filter is applied, while in the feedback loop, the two time constants T_1 and T_2 are compensated, that is,

$$
F_y(s) = \frac{(1+T_1s)(1+T_2s)}{K(1+T_fs)};\ F_u(s) = \frac{e^{-T_ds}}{(1+T_fs)}.
\tag{75}
$$

By replacing the inner loop with

$$
S_u(s) = \frac{1}{1-F_u(s)} = \frac{1+T_fs}{1+T_fs-e^{-T_ds}},
\tag{76}
$$

moving F_y to a straight branch, merging with S_u and $1/K$, substituting for T_d (5) and considering $T_f = T_c$, both the pre-filter $P(s)$ and the ideal SIMC PID controller $R(s)$ can be calculated as:

$$
\begin{aligned}
&P(s) = \frac{(1+T_fs)}{(1+T_1s)(1+T_2s)} = \frac{(1+T_cs)}{(1+T_1s)(1+T_2s)};\\
&R(s) = K_c\frac{(1+T_is)(1+T_Ds)}{T_is};\ K_c = \frac{T_1}{K(T_c+T_d)};\ T_i = T_1;\ T_D = T_2.
\end{aligned}
\tag{77}
$$

Practical implementation of the controller requires using the first-order low-pass filter with the time constant $T_f = \alpha T_D;\ \alpha \in [0.01, 0.1]$ [4].

Smoother monotonic responses $u(t)$ can again be achieved by selecting the pre-filter according to

$$
P(s) = \frac{1}{(1+T_1s)(1+T_2s)};\ F_p(s) = \frac{1}{(1+T_cs)}.
\tag{78}
$$

The speed of setpoint step responses may be further accelerated with considering just the first-order pre-filter $P(s)$ (67).

5. Illustrative Examples

The differences between the traditional SIMC method with HR and the proposed design modifications using MHR, will be illustrated on several examples. For the sake of simplicity, where appropriate, the upper index "1" used to denote ^{1}PI, ^{1}PID, or ^{1}PIDA controller will be omitted.

5.1. Example 1: SIMC and Newly Proposed Control of FOTD System with the 2nd Order Low-Pass Filter

In this example, the 2nd order binomial filter (35) inspired by [27] (with $n = 2$) has been added to SIMC PI and PID controllers (SPI and SPID) and to all three first-branch controllers (^{1}PI, ^{1}PID and ^{1}PIDA) based on FOTD models.

According to HR, the SPI and SPID controllers are tuned as follows

$$T_1 = T_{1m} + 0.5T_f;\ T_d = T_m + 1.5T_f \tag{79}$$

For SPID, the derivative action time constant [6] and the series controller filter time constant for noisy processes [4] have been specified according to

$$T_D = T_d/3;\ T_{f1} = 0.1T_D. \tag{80}$$

According to MHR (36), used in ^{1}PI, ^{1}PID and ^{1}PIDA tuning, the identified plant time constant $T_{1m} = 1$ and the identified plant model dead-time $T_m = 1$ have to be modified according to

$$T_1 = T_{1m} + T_f;\ T_d = T_m + T_f. \tag{81}$$

Particular loop parameters have been specified as follows:

$$\begin{array}{l} \text{PI: } T_{c1} = 0.9;\ T_f = 0.05 \\ \text{PID: } T_{c1} = 0.7;\ T_f = 0.1 \\ \text{PIDA: } T_{c1} = 0.5;\ T_f = 0.15. \end{array} \tag{82}$$

The required closed loop time constant T_{c1} under the PI controller has been chosen slightly below the recommended (8) with the aim to keep the speed of transients close to PID and PIDA control. Since the PI controller does not include "aggressive" derivative term, the filter time constant T_f may be decreased. PID control, in general, allows faster transients, which is reflected by smaller T_{c1}. However, due to an increased noise level, T_f has been intuitively increased as well. Both mentioned modifications have also been applied on PIDA controller.

Performance measures IAE_d, $TV_1(u_d)$, and $TV_1(y_d)$ on unit input step disturbance, for all three controllers (82) are given in Figure 6. For all controllers, IAE_d values are nearly equal. In terms of $TV_1(u_d)$, the lowest excessive control effort, when there is no noise ($\delta = 0$), is achieved with PIDA controller, while the highest effort is obtained with PID controller.

Concerning the process output signal, the PID and PIDA controllers yield nearly ideal 1P responses with $TV_1(y_d) \approx 0$. The highest excessive output changes $TV_1(y_d)$ are obtained with the PI control. With existing measurement noise $|\delta| \leq 0.2$, the output changes are even larger than the ones obtained with the PID and PIDA controllers. For all three controllers, the excessive control effort, due to the measurement noise, significantly exceeds the one without the noise ($\delta = 0$). Surprisingly, PIDA controller, with the fastest transients and seemingly "aggressive" 2nd order derivative action, shows lower excessive control effort than the simplest and slowest PI control. This is well documented by the combined cost functions (50) and (51) in Figure 7. The normalized values, based on the PI controller, show that the PID controller with $k = 1$ has slightly higher value of $J_k(u_d)$. However, the PIDA controller's cost function is lower. The superiority of PID and PIDA control is even more evident when emphasizing the speed of transients ($k = 6$). Obviously, the optimization of the chosen controller and filter tuning is far from being trivial and requires to develop

a systemic approach. Given the unlimited range of different requirements of practice (represented by the parameter k), the search for a globally optimal method of controller tuning can therefore be considered erroneous, even when restricted just to simple PID control [43].

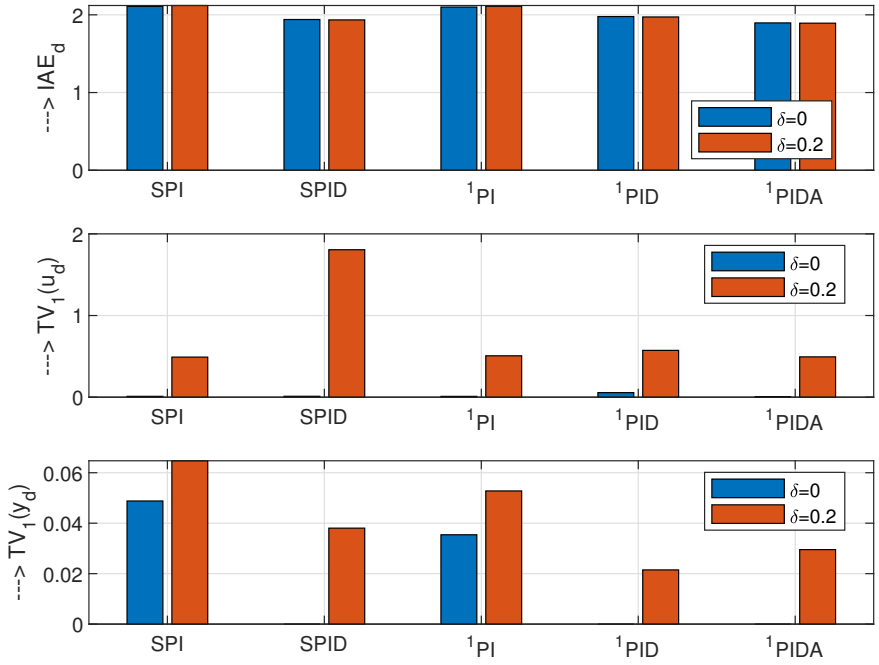

Figure 6. Performance of the loops with FOTD plant, SIMC PI ad PID controllers (SPI and SPID) tuned for (79), ^{1}PI (6) and ^{1}PID (12) and ^{1}PIDA (15) controllers tuned for model (81). All controllers are using the 2nd order filter (35) with parameters (82). The performance is calculated for no external noise ($\delta = 0$) and for noise amplitudes $|\delta| \leq 0.2$; $K = 1$; $T_{1m} = 1$; $T_m = 1$; $T_s = 0.001$; $t_{sim} = 12$.

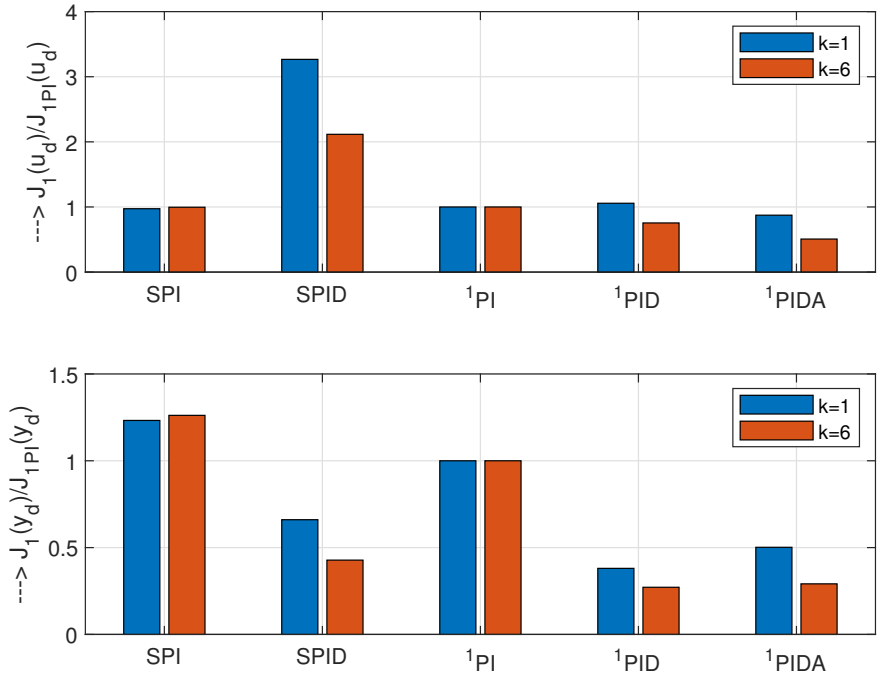

Figure 7. Holistic cost functions (50) and (51) corresponding to performance measures in Figure 6 related to ^{1}PI control for external noise with amplitudes $|\delta| \leq 0.2$ for $k = 1$ and $k = 6$.

From the noise attenuation point of view, in comparison with SPI and SPID controller, the ^{1}PI and ^{1}PID controllers show improved noise attenuation, which is demonstrated (for ^{1}PID) by lower shape-related deviations in Figure 6, or (for both ^{1}PI and ^{1}PID) by the lower values of the cost functions in Figure 7. Transients under noisy process signals are shown in Figure 8. As indicated by increased $TV_1(y_d)$, under ^{1}PI control, a relatively short T_{c1} value leads to a moderate output undershooting. Although the amplitude of the noise δ superimposed according to Figure 1 on the output variable exceeds 25% of the maximal useful signal, on the control variable $u(t)$ and even less on the output $y(t)$, its effect is relatively small.

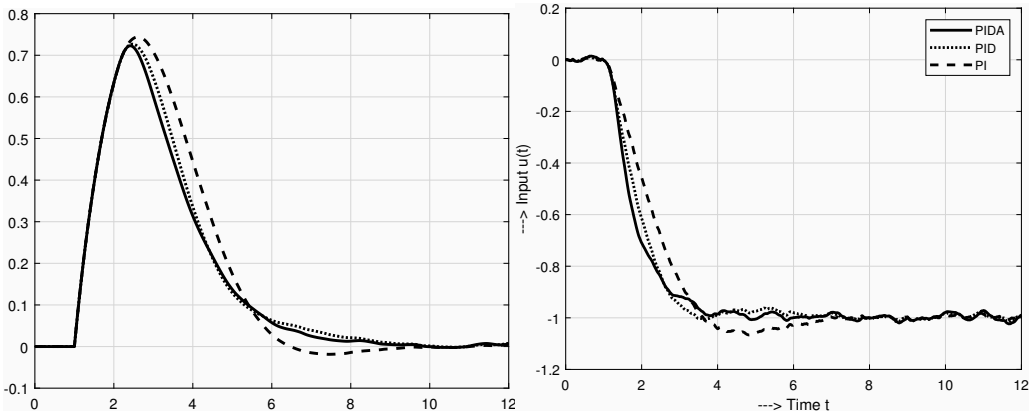

Figure 8. Disturbance step responses of the loop with FOTD plant and ^{1}PI (6), ^{1}PID (12) and ^{1}PIDA (15) controllers, with 2nd order filter (35) with parameters (81) and (82); $K = 1$; $T_m = 1$; $T_{1m} = 1$; $|\delta| \leq 0.2$; $T_s = 0.001$; $t_{sim} = 12$.

The process output responses, under ^{1}PID and ^{1}PIDA controllers, are nearly ideal 1P responses with $TV_1(y), TV_1(u) \approx 0$. Therefore, in the case with no noise we may expect IAE values (49) to be close to the values achieved by simulation. Thus, under ^{1}PID control with $T_f = 0.1$, choosing the equivalent model parameters (81) $T_d = T_1 = 1.1$, we get for $T_{c1} = 0.7$ $^1IAE = 1.98$ (49), which is nearly the same as $^1IAE = 1.978$ from simulation. Similarly, for ^{1}PIDA control with $T_f = 0.15$ resulting in $T_d = T_1 = 1.15$ (81), we get for $T_{c1} = 0.5$ $^1IAE = 1.897$ (49), which is again nearly the same as $^1IAE = 1.896$ from simulation. These calculations may be considered as an experimental verification of the applicability of MHR from Definition 3. They also illustrate motivation of [6] to return to the ^{1}PID control, which was rejected in the initial work [4].

5.2. Example 2:SE/SW Based Analysis of Controller + Filter Tuning—No Noise

The aim of this example is to explain the choice of the parameter T_{c1} in (82) by exploring SE and SW characteristics of the particular controllers with the chosen filters (35).

To get almost constant filter delay for different filter orders n, the filter time constants T_f will be derived by MHR from the filter average residence time $T_{f0} = nT_f$ according to

$$T_f = T_{f0}/n;\ n \in [1,4]. \tag{83}$$

Filtered ^{1}PI control with three different values of T_{f0} yields SE and SW characteristics shown in Figure 9. By increasing T_{c1} the corresponding IAE values increase. Thereby, the curves corresponding to different filter degrees n mostly overlap, which supports simplified filter description (83). To get nearly 1P responses at the plant output for $T_{f0} = 0.1$, when $TV_1(y) \to 0$, one needs to choose $T_{c1} \geq 1.3$. For $T_{f0} = 0.2$ it may already be achieved with $T_{c1} \geq 1.2$.

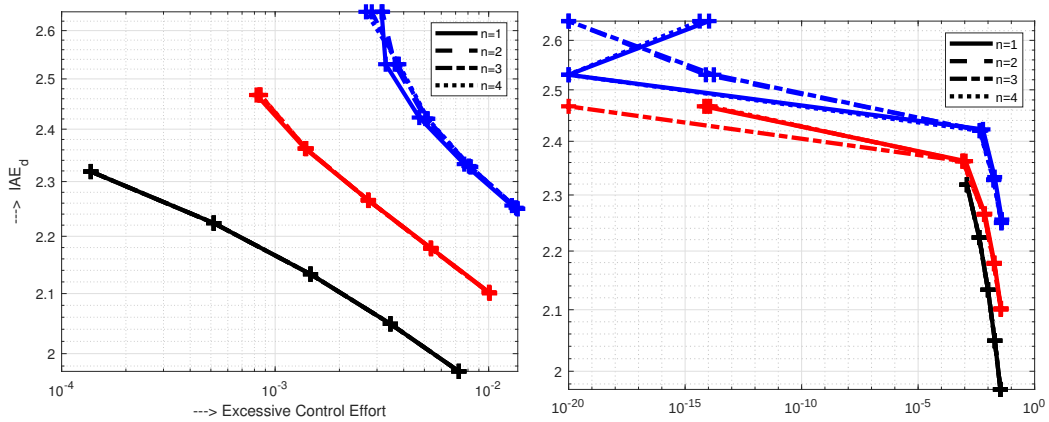

Figure 9. SE and SW characteristics of the loop with FOTD plant, $Q_n(s)$ (35) with $n \in [1,4]$ and ^{1}PI (6) controller with parameters $T_{c1} = \{0.9, 1.0, 1.1, 1.2, 1.3\}$ and $T_{f0} = 0.01$ (black), $T_{f0} = 0.1$ (red) and $T_{f0} = 0.2$ (blue) $K = 1$; $T_m = 1$; $T_{1m} = 1$; $\delta = 0$; $T_s = 0.001$; $t_{sim} = 20$

To get nearly 1P transients at the input (not covered by these figures), for $T_{f0} = 0.1$ and 0.2 one needs to choose $T_{c1} = 1.5$ and 1.4, respectively. Thus, for PI controller, and desired zero shape related deviations, even when including T_{f0} into the equivalent dead-time, the recommendation (8) is too weak.

Remark 4 (Lag and delay dominant processes)**.** *It can be stated that the above conclusions regarding the choice of $T_c = T_{c1}$, to guarantee the smallest possible deviation of responses from their ideal shapes, are in a good agreement with the recommendations of "smoother tuning" in SIMC [6] for PI controller, giving $T_c = 1.5T_d$. Of course, the considered process with $T_{1m} = T_m = 1$, does not even represent all stable 1st order processes. However, regarding the choice of T_f, the obtained results can thus be easily interpreted using the ratio T_f/T_m or T_f/T_{1m}. Furthermore, some other recommendations are anticipated for lag-dominant processes with $T_m < T_{1m}$ and still other for delay-dominant with $T_m > T_{1m}$. In this article, however, we will not discuss in detail all possible cases, but we plan to offer the reader an interactive web application, where the particular cases can be easily verified.*

^{1}PID control (12) inspected for $T_{c1} = \{0.3, 0.4, 0.5, 0.6, 0.7\}$ yields nearly 1P output transients for $T_{c1} \geq 0.5$ (Figure 10). At the input this happens just for $T_{c1} \geq 0.8$ ($T_{f0} = 0.01$) or $T_{c1} \geq 0.9$ ($T_{f0} \geq 0.1$). The value $T_{c1} = 0.7$ in (82) guarantees ideal shapes just at the output.

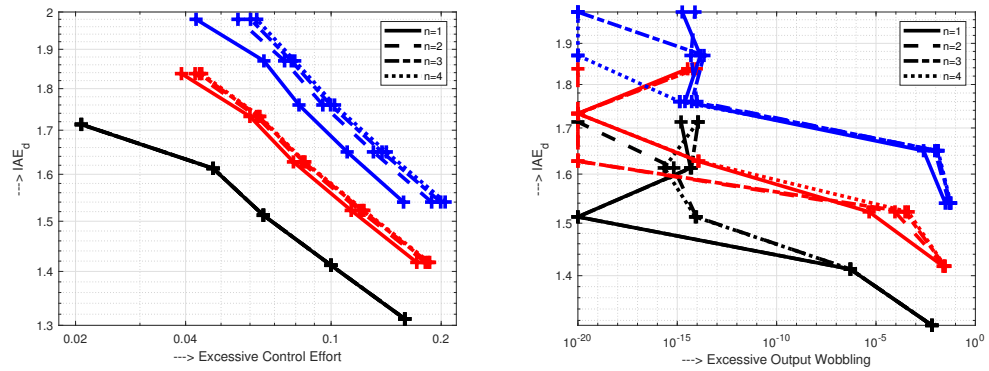

Figure 10. SE and SW characteristics of the loop with FOTD plant, $Q_n(s)$ (35) with $n \in [1,4]$ and ^{1}PID (12) controller with parameters $T_{c1} = \{0.3, 0.4, 0.5, 0.6, 0.7\}$ and $T_{f0} = 0.01$ (black), $T_{f0} = 0.1$ (red) and $T_{f0} = 0.2$ (blue), $K = 1$; $T_m = 1$; $T_{1m} = 1$; $\delta = 0$; $T_s = 0.001$; $t_{sim} = 20$.

Improvements achieved for the same range of T_{c1} by augmenting ^{1}PID to ^{1}PIDA control filtered by (35) (Figure 11) are already not so remarkable as when extending ^{1}PI to ^{1}PID. SE and WE characteristics show that at the output, with $T_{f0} \geq 0.1$ the transients become 1P already for $T_{c1} \geq 0.4$ and at the input for $T_{c1} \geq 0.7$. Thereby, they allow to achieve nearly ideal 1P transients at the input and output with lower IAE values. Again, the value $T_{c1} = 0.5$ in (82) guarantees ideal 1P shapes just at the output.

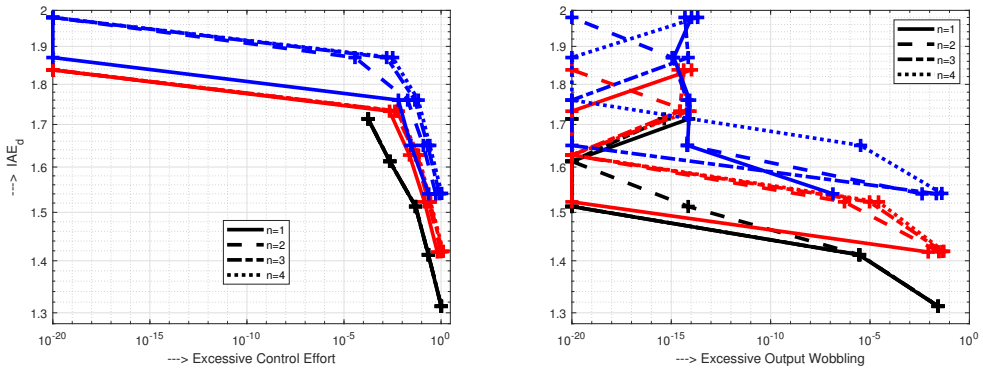

Figure 11. SE and SW characteristics of the loop with FOTD plant, $Q_n(s)$ (35) with $n \in [1,4]$ and ^{1}PIDA (15) controller with parameters $T_{c1} = \{0.30.4, 0.5, 0.6, 0.7\}$ and $T_{f0} = 0.01$ (black), $T_{f0} = 0.1$ (red) and $T_{f0} = 0.2$ (blue), $K = 1$; $T_m = 1$; $T_{1m} = 1$; $\delta = 0$; $T_s = 0.001$; $t_{sim} = 20$.

5.3. Example 3: SE and SW Characteristics—External Noise

The following example will illustrate the results given in [6], where it was shown that "smooth" (i.e., nearly 1P) input and output transients are achieved by slightly different values of T_{c1} than recommended in (8) or in (82) in Example 1. The tested controllers, filters and the optimal parameters T_c are given below:

$$\begin{aligned} &\text{PI: } T_{c1} = 1.2 \\ &\text{PID: } T_{c1} = 1.0 \\ &\text{PIDA: } T_{c1} = 0.8 \\ &T_{f0} = \{0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4\} \\ &T_f = T_{f0}/n;\ n \in [1,4] \\ &T_1 = T_{1m} + T_{f0}/2;\ T_d = T_m + T_{f0}/2;\ |\delta| \leq 0.2 \end{aligned} \tag{84}$$

The results of the experiment (see SE and SW characteristics in Figure 12) show that by increasing the controllers order, the IAE values decrease without significant increase of the controller effort. Thus the results evidently refute general belief that controllers with a derivative action are not suitable for noisy systems. On the contrary, it is clearly shown that the closed-loop transients may be accelerated, with simultaneously decreasing the controller output noise, just by increasing the controller derivative order. In other words, when properly combining filtration, which yields smoother, but slightly slower transients, with derivative action (prediction) allowing faster, but noise amplifying dynamics, you may get faster responses without impairing their closed-loop shapes. Once upgrading PI to PID (as already done in [6]), we can continue with upgrading to PIDA, or to higher-order controllers (as used in the actual implementation of fractional order controllers, where the filters commonly exceed the order of 10)? During the era of pneumatic and other analog controllers, the PI control is used due to its simplicity [6]. However, as documented by increasing interest in fractional order PID, implemented by the HO controllers [2], the controller order does not play a role anymore in today's software-implemented controllers. Taking into account that all the controllers are calculated from the same process model (in our case FOTD), it does not complicate process identification. Grimholt and Skogestad [6] are aware that with respect to the PI-tuning with $T_c = T_d$, the iSIMC PID-tuning with $T_c = T_d/2$ improves IAE performance by about 30%, while keeping about the same

robustness level ($M_s \approx 1.7$). Such PID controller is even better in almost all aspects than a well-tuned Smith Predictor. However, the HO controllers may also require solving problems associated with control constraints [19]. Moreover, frequent discard of the derivative term [1,16] may only be explained by improperly solved filtration problems. Here, it is important to mention that the differences between $n = 1$ and $n > 1$ in Figure 12 would be even more pronounced on extended x-. This means that the excessive control effort may be even lower by using at least 2nd order filters.

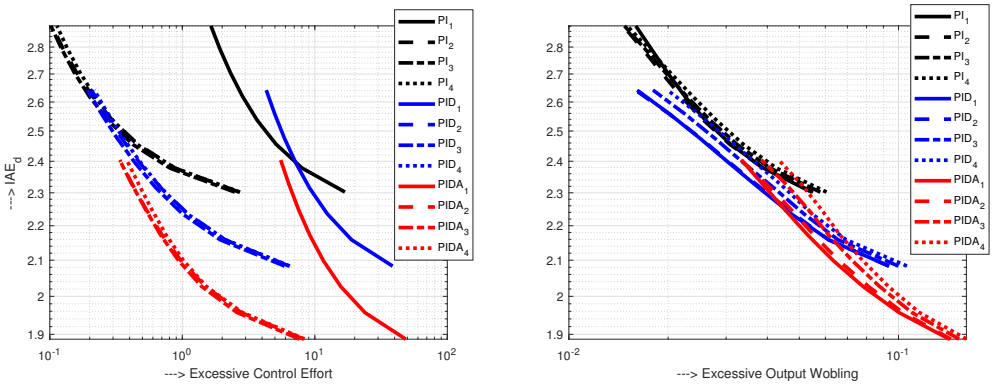

Figure 12. SE and SW characteristics of the loop with FOTD plant, $Q_n(s)$ (35) with $n \in [1,4]$ and with ^{1}PI (6), ^{1}PID (12) and ^{1}PIDA (15) controller with parameters (84), $K = 1$; $T_m = 1$; $T_{1m} = 1$;$|\delta| \leq 0.2$; $T_s = 0.001$; $t_{sim} = 20$.

5.4. Example 4: SIMC and Newly Proposed Control of Fourth-Order System, No Noise

In [4], SIMC PI and PID control have also been applied to the fourth-order system (30) with $K = 1, T_d = 0$ and $T_4 = 1$ (33). By means of HR, this plant has either been approximated by FOTD model [4]

$$^1F_{HR}(s) = \frac{e^{-2.5s}}{1+1.5s} \tag{85}$$

or by SOTD model

$$^2F_{HR}(s) = \frac{e^{-1.5s}}{(1+1.5s)(1+s)} \tag{86}$$

For $T_c = T_d = 2.5$, the FOTD model yields a SIMC PI (SPI) controller with

$$T_i = 1.5;\ K_c = T_1/[K(T_c + T_d)] = 1.5/5 = 0.3 \tag{87}$$

For $T_c = T_d = 1.5$, the SOTD model yields the SIMC PID controller (SPID) with parameters

$$T_i = 1.5;\ T_D = 1;\ K_c = T_1/[K(T_c + T_d)] = 1.5/3 = 0.5;\ T_{f1} = T_D/100 = 0.01 \tag{88}$$

On the other side, application of MHR yields the following models

$$^1F_{MHR}(s) = \frac{e^{-1.5s}}{1+2.5s} \tag{89}$$

$$^2F_{MHR}(s) = \frac{e^{-s}}{(1+1.5s)(1+1.5s)} \tag{90}$$

$$^3F_{MHR}(s) = \frac{e^{-0.5s}}{(1+1.1667s)(1+1.1667s)(1+1.1667s)} \tag{91}$$

The above models have been used in design of ^{1}PI and ^{1}PID controllers, ^{2}PID and ^{2}PIDA controllers and ^{3}PIDA controller. In each case, we have chosen $T_c = T_d$.

From the QOTD model (33) follows $T_d = 0$. However, zero T_d value, and low values of T_c, can lead to high controller gains causing oscillations. This can be avoided by taking into account the neglected time constants, which will eventually be translated into the effective dead-time value. These enable to replace the ideal $T_d = 0$ with some positive number, chosen for example, as $T_d = 0.2T_4$. In our case, with $T_4 = 1$, the plant approximation used for the controller design is

$$^4F(s) = \frac{e^{-0.2s}}{(1+s)^4}. \tag{92}$$

Remark 5 (Reliability of model (33) in controller design)**.** *Although the addition of a transport delay to the model* (33) *may seem like a non-system solution, in reality, adhoc simplifications can be expected already in the process of obtaining* (33)*. The assumption that the process has only a four-tuple dominant time constant and no other minor delays may correspond to reality only in very unlikely circumstances. Exact identification of shorter delays runs into numerical problems—in trying to identify exactly large dominant time constants, the determination of small delays in accuracy ceases. Nevertheless, if we could determine them, in the end, according to HR and MHR, we will still include them in T_d. The problem can easily be demonstrated for example, when using Strejc identification [18]. There, the zero value of the transport delay is output only for specially measured process values, which we probably do not get on repeated identification. To obtain a model without a transport delay, the measured values must be rounded appropriately. Models based on such data manipulation may yield quite high data fitting in the identification evaluation. They can also give good results in controller design using reduced-order process models, when shorter time constants neglected are negligible compared to the dead-time resulting from model order reduction. However, they are not appropriate for a reliable controller design using models* (30) *with $T_d = 0$.*

Figure 13 shows that for $j = 1$ and $j = 2$, an increase of the derivative action order (e.g., from ^{1}PI to ^{1}PID, or from ^{2}PID to ^{2}PIDA) has virtually no effect on the process responses. However, the higher accuracy of the delay approximation, by using higher-order transfer functions and thus increasing the controller order, does not have a significant effect on the closed loop response in lag-dominant processes. The advantage of the newly proposed design is the possibility to significantly improve the control performance with simultaneously achieving nearly ideal shapes of process output transients.

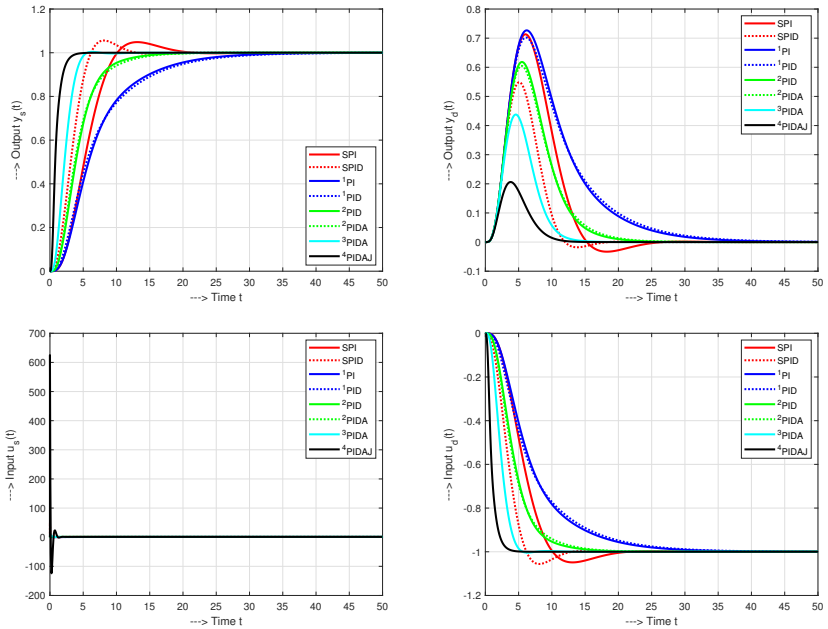

Figure 13. Setpoint and input disturbance unit step responses of the system (33) by SPI, SPID, ^{1}PI, ^{1}PID, ^{2}PID, ^{2}PIDA, ^{3}PIDA and ^{4}PIDAJ controllers, no noise.

For all used controllers, the course of the control action at the setpoint change exhibits large signal swing. The initial control kick can be attenuated by using the derivative filter derived in Section 4. Another way to avoid derivative kick on setpoint change [4], is to follow industry practice by differentiating only the output of the system (or by using a setpoint weighting parameters). However, such a solution is not as efficient in terms of adapting the dynamics of the closed-loop responses when compared to pre-filters.

5.5. Example 5: Comparing SIMC PI and PID Controllers with the Newly Proposed Solutions Applied to Fourth-Order System

A comparison of the SIMC PI controller (SPI) and the newly designed ^{1}PI controller in the above example implemented for $T_c = T_d$ shows faster SPI setpoint step response and slower, but strictly monotonic setpoint step response of ^{1}PI control. However, such a comparison for a single tuning parameter value does not capture the resulting performance of the whole family of controllers. Figure 14 shows such responses corresponding to the plant approximations (85), (89) and the pre-filter (67) with the following T_c values

$$T_c = \{0.75, 1, 1.5, 2, 2.5, 3, 3.5, 4\}. \tag{93}$$

They illustrate that the nearly ideal shapes of the plant input and output signals, corresponding to the FOTD process model, are achieved at higher T_c values than recommended in (8).

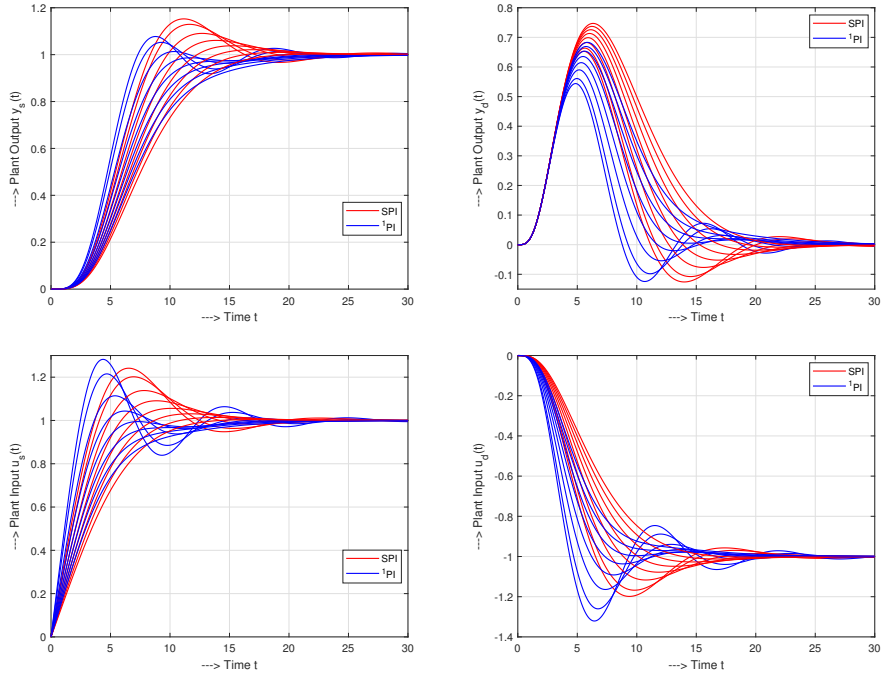

Figure 14. Setpoint (left) and input disturbance unit step responses (right) of the system (33) with SPI (red), and ^{1}PI controllers (blue) for T_c (93), pre-filter (67) and the plant approximations (85) and (89), no noise.

Thereby, the dynamics of the considered controllers depends on T_c in different ways. The SE and SW characteristics in Figure 15 show that for the same values of shape-related deviations the IAE values of SPI controller are generally higher than for ^{1}PI controller. For both controllers, increasing the T_c decrease the excessive control effort and wobbling. However, the overshooting of the SPI controller, which indeed gives less IAE than for monotonic responses, is manifested by higher values of output wobbling and it does not fall to zero even for very high values of T_c. From the results of comparison, it is clear that MHR gives better results than the HR when reducing higher order transfer functions for tuning the PI controllers.

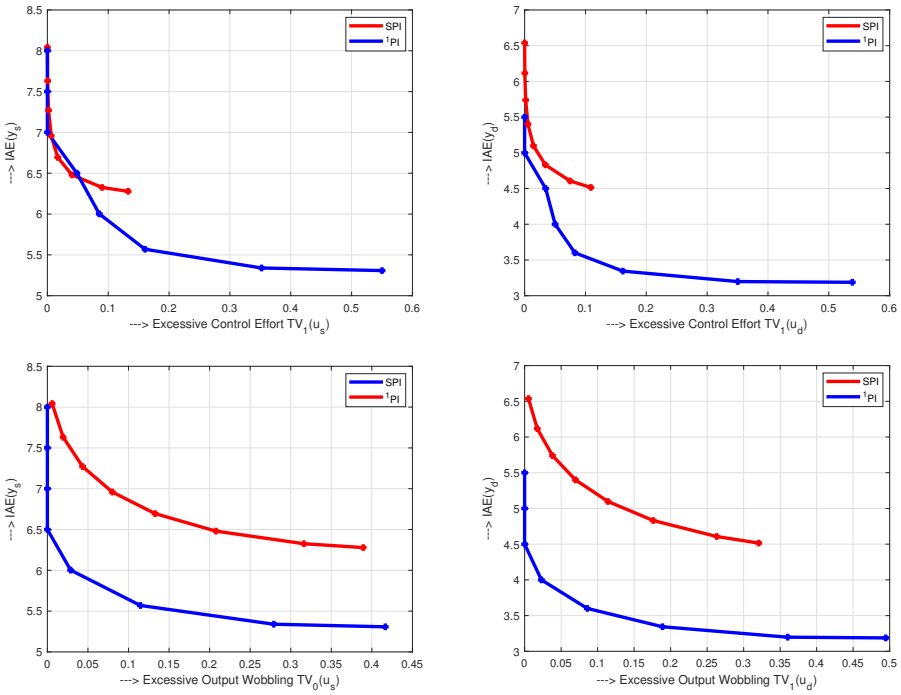

Figure 15. SE and SW characteristics corresponding to transients with with SPI controller (red) in Figure 14 correspond mostly to higher IAE values than the transients with ^{1}PI controller (blue), no noise.

Let us now discuss the details of the setpoint step responses of the PID controllers from Figure 13. The process (33) is represented by the second order models (86), or (90). Neglecting the initial peak, the process input is practically zero during the simulation time (Figure 16). Such kind of control cannot lead to robust closed-loop control. As T_c decreases, this initial peak narrows and increases. Although the ^{2}PID controller is much better in this respect than SPID, the initial peaks increase by decreasing T_c. The problem can be solved by using pre-filters (68), or (78) derived in Sections 4 and 4.2. In Figure 17, the transients of both compared controllers (SPID and ^{2}PID) are shown for

$$T_c = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}. \tag{94}$$

Obviously, the closed-loop responses are significantly faster when compared to the PI controllers in Figure 14. Thanks to the pre-filters used, their closed loop responses are smooth and without distinct peaks.

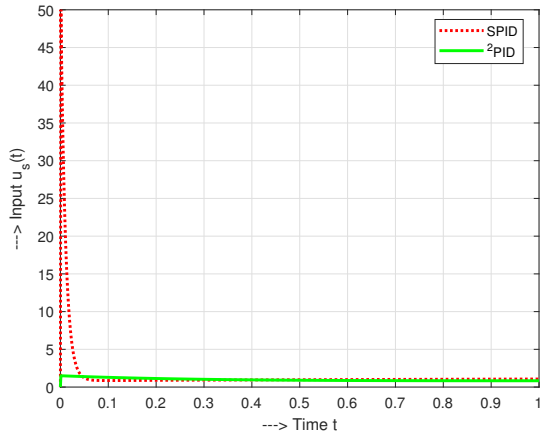

Figure 16. Detail of the setpoint unit step responses of the system (33) from Figure 13 corresponding to SPID and ^{2}PID controllers, no noise.

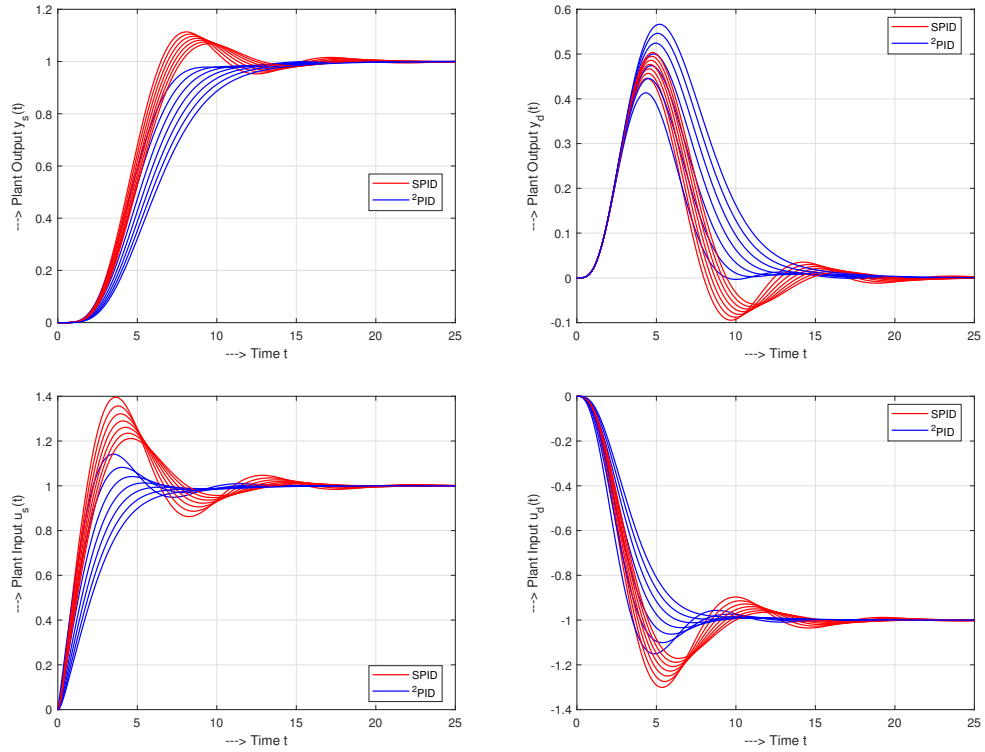

Figure 17. Setpoint and input disturbance unit step responses of the system (33) with SPID (red) and ^{2}PID controllers (blue), T_c (94), pre-filters (78) and (68) and the plant approximations (85) and (89), no noise.

The SE and SW characteristics in Figure 18 show again that, by increasing T_c, the IAE value changes much less for SPID controller and they are always above the values for ^{2}PID controller for the same shape-related deviations. The ^{2}PID controller results in zero shape-related deviations in almost the entire range of considered T_c values. Again, it can be concluded that although processes with overshooting give less IAE than monotonic responses, the MHR gives, in combination with ^{2}PID control, lower IAE values than the HR, when reducing the HO transfer function (33) order for the SPID controller design.

Remark 6 (Ambiguities of design and evaluation regarding the choice of pre-filter and ideal shape of $u(t)$)**.** *The SE characteristics in Figure 18 evaluate the deviations from the ideal shape of the input signal $u(t)$ using $TV_2(u)$ criterion. The criterion was chosen to unify the selection of controller and the evaluation of its performance. Similarly, when evaluating SPI and ^{1}PI controllers in Figure 15, we calculated the deviations by using $TV_1(u)$. However, the requirement for consistency of the controller design may lead to conclusion that we still control the fourth-order system (30), so we should use $TV_4(u)$. Another possible proposal could require that we adapt the performance measures to the most complex controller used.*

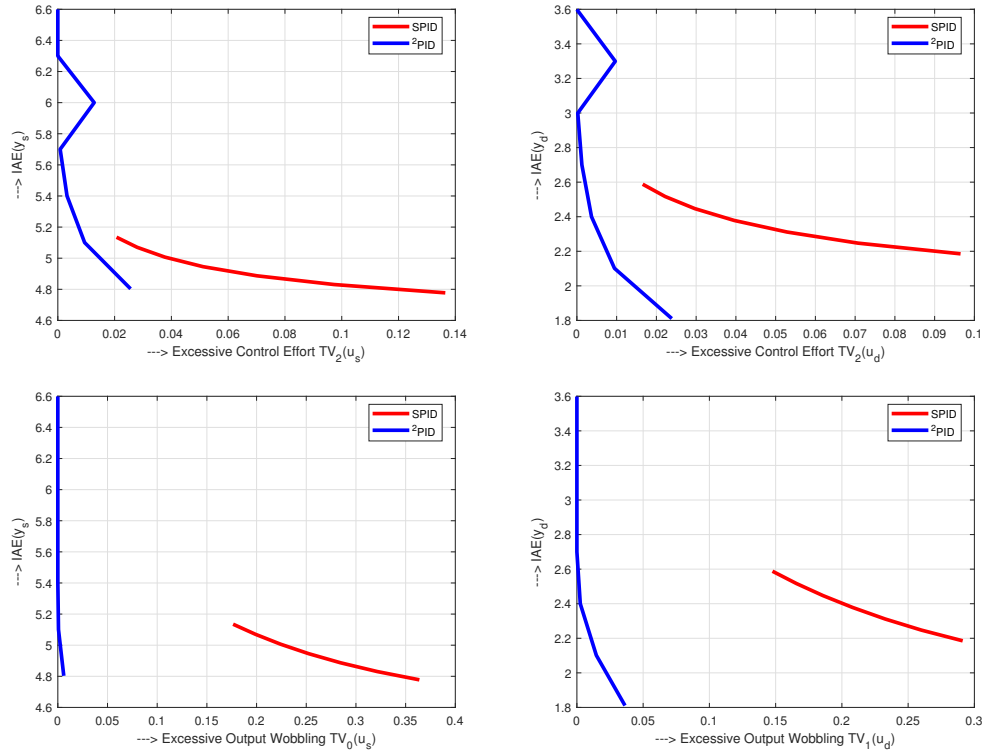

Figure 18. SE and SW characteristics corresponding to transients in Figure 17 with SPID (red) and ^{2}PID controllers (blue), no noise.

To illustrate broad spectrum of possible design alternations, the next comparison on the same example considers SPID controller (88) with pre-filter (78) and ^{3}PIDA controller (28) based on the model (91) with a reduced-order pre-filter

$$^3P^2(s) = \frac{1}{(1+T_3 s)^2}. \tag{95}$$

Both control structures have been compared for the following tuning parameters:

$$T_c = \{0.25, 0.4, 0.55, 0.75, 1.0\}. \tag{96}$$

To sufficiently suppress the measurement noise, a second-order filter $Q_2(s)$ has been used for SPID and a fourth-order filter $Q_4(s)$ for ^{3}PIDA control, both with time constants $T_f = 0.15$, at the sampling period $T_s = 0.001$.

The closed-loop responses on unit step setpoint and disturbance unit step responses changes in Figure 19 show significantly faster dynamics of ^{3}PIDA control compared to SPID control. The conclusions are also supported by the SE and SW characteristics in Figure 20. Given that SPID is based on the 2nd order model, we might expected ideal output shapes to be achieved with 2P input (i.e., transients from DC2) and use in the corresponding evaluation of the excessive control effort the $TV_2(u)$ measure. Using the same arguments for ^{3}PIDA controller, the performance evaluation should work with $TV_3(u)$. To illustrate the emerging differences, we will compare both performance measures.

The SE characteristics in Figure 20 demonstrate the difference between both measures. They show that, the process input, in terms of $TV_3(u)$, when using the SPID controller, also provides ideal 3P responses for some larger values of the considered T_c setting (96). However, the achievable range of IAE values is significantly narrower and the output signal deviations from the ideal responses are larger than with ^{3}PIDA. Note that while with ^{3}PIDA control, T_c mainly affects the closed-loop speed without much effect on input and output shape deviations. On the other hand, with SPID, the correlation between the T_c

and the output shape is quite significant. At the same time, transients with ^{3}PIDA control are both faster and smoother.

When using $TV_2(u)$, the considered responses $u(t)$ would give larger deviations than yielded by $TV_3(u)$. From the comparison of numerical values in (45), it is then clear that the controller optimization based on TV prevents achieving the optimal performance, especially for the higher-order systems.

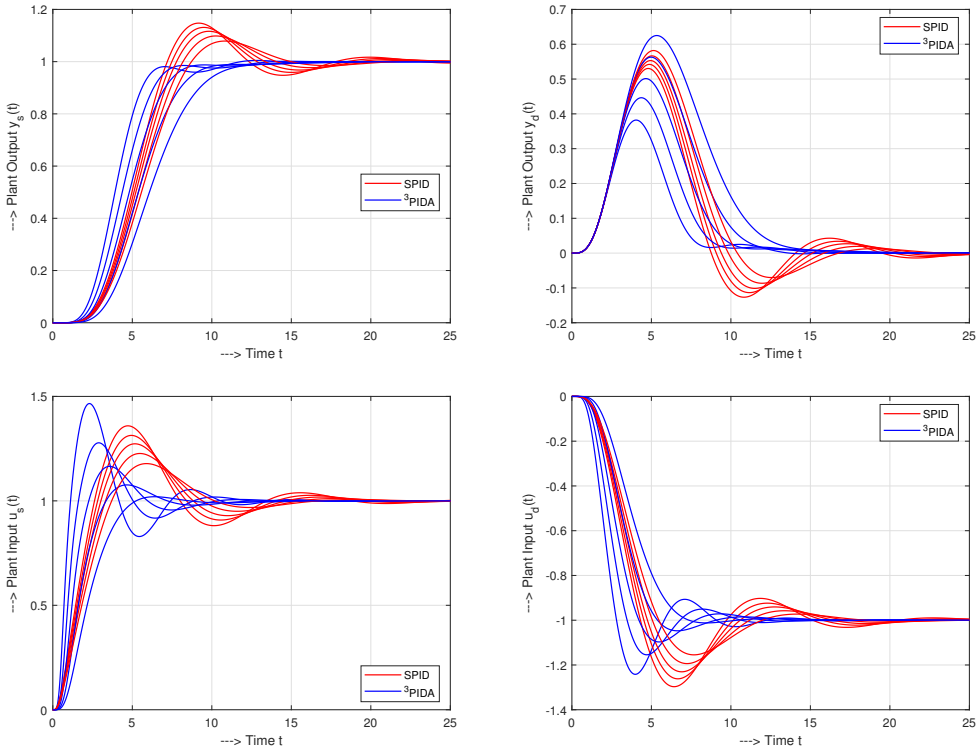

Figure 19. Setpoint and input disturbance unit step responses of the system (33) with SPID+$Q_2(s)$ (red) and ^{3}PIDA+$Q_4(s)$ controllers (blue), T_c (96), $T_f = 0.15$, pre-filters (78) and $^3P(s) = 1/(1+T_3 s)^2$ and the plant approximations (85) and (91).

A similar conclusion is achieved when comparing the structure of SPID +$Q_2(s)$ with the ^{4}PIDAJ controller proposed in Example 4 for the plant (92) and supplemented by the noise attenuation filter $Q_4(s)$ with $T_f = 0.15$ and the reduced-order prefilter

$$^4p^3(s) = \frac{1}{(1+T_4 s)^3}. \tag{97}$$

In order to be able to speed up the loop responses under ^{4}PIDAJ control, with respect to the ^{3}PIDA controller, another T_c value was added to the tuning parameter set (96):

$$T_c = \{0.15, 0.25, 0.4, 0.55, 0.75, 1.0\}. \tag{98}$$

Analysis of the family of time responses in Figure 21 yields the SE and SW characteristics in Figure 22.

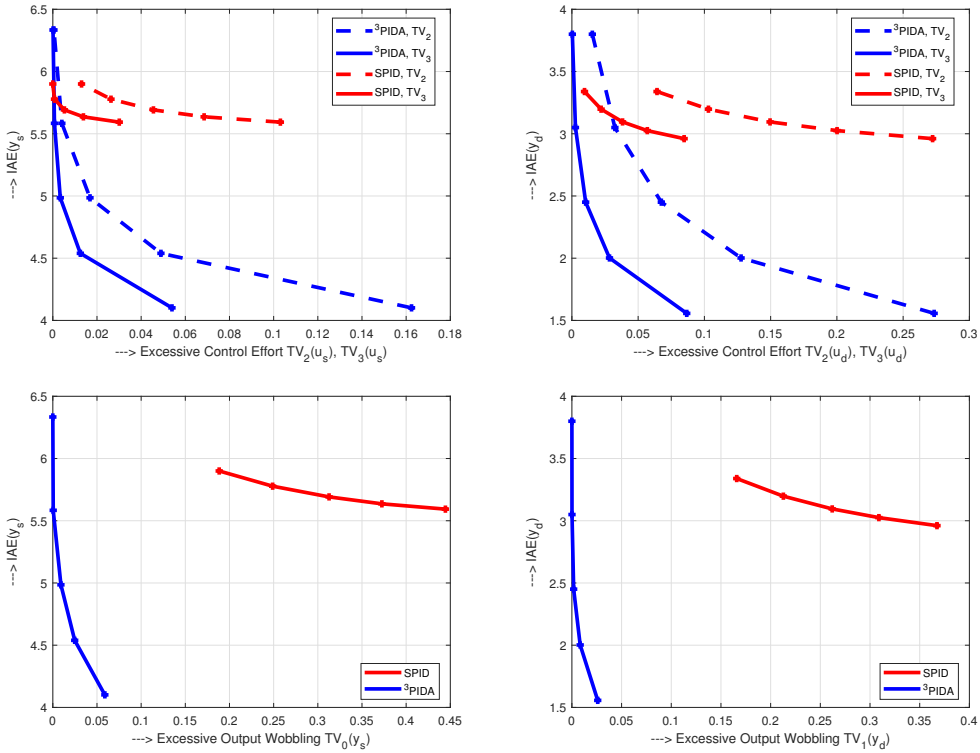

Figure 20. SE and SW characteristics of SPID and ^{3}PIDA controllers corresponding to transients in Figure 19.

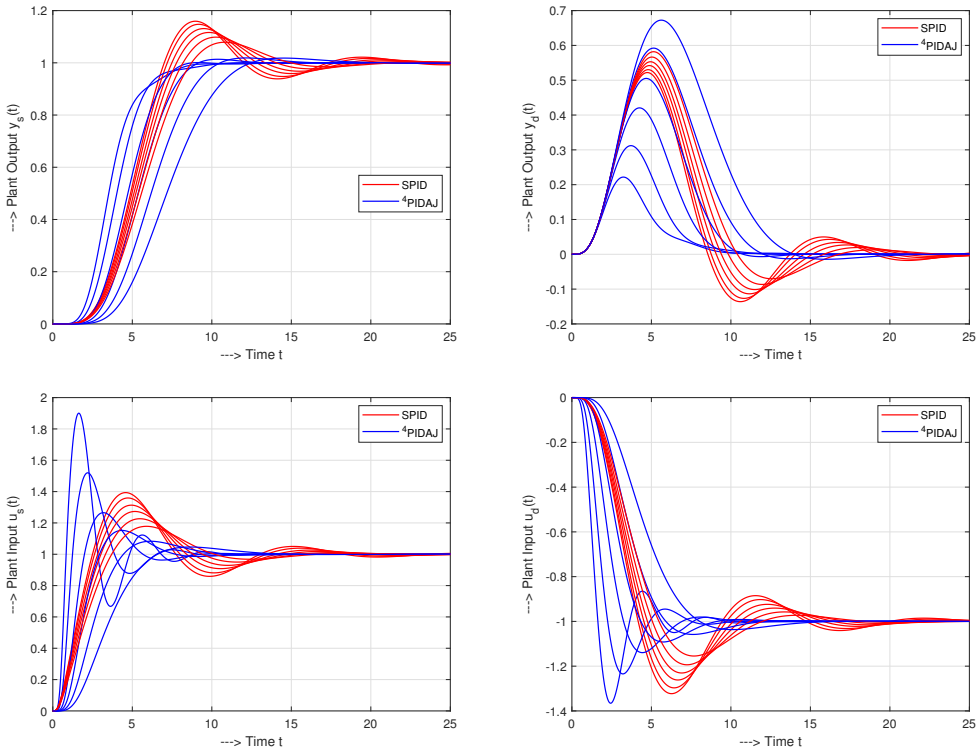

Figure 21. Setpoint and input disturbance unit step responses of the system (33) with SPID+$Q_2(s)$ (red) and ^{4}PIDAJ+$Q_4(s)$ controllers (blue), T_c (98), $T_f = 0.15$, pre-filters (78), $^4P^3(s) = 1/(1+T_4 s)^3$ (97) and the plant approximations (85) and (92).

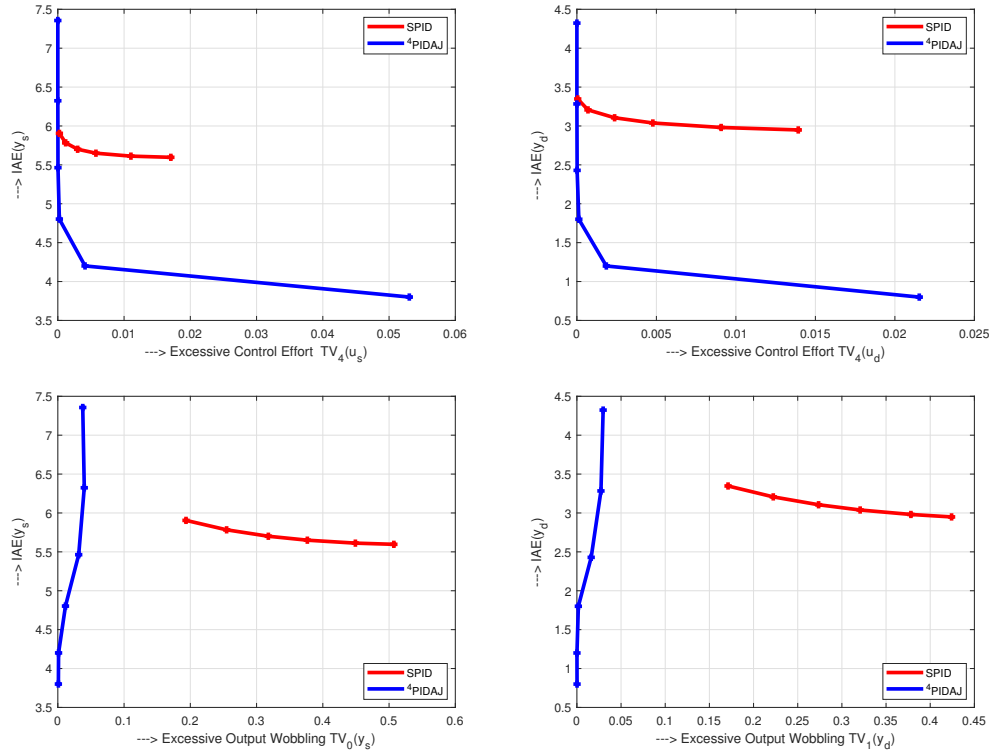

Figure 22. SE and SW characteristics of SPID and ^{4}PIDAJ controllers corresponding to transients in Figure 21.

The comparison with the characteristics in Figure 20 shows a significant decrease of $TV_4(u)$ values. The performance improvements for the ^{4}PIDAJ controller, in terms of $TV_4(u)$, are particularly visible in disturbance responses.

5.6. Example 6: SIMC and Newly Proposed Control of QOTD System with 3rd-Order Noise Attenuation Filters

The dynamics of the plant considered in the previous example will be evaluated with an external measurement noise with an amplitude $|\delta| \leq 0.1$ generated in Matlab/Simulink by a Uniform Random Number block. To attenuate the noise impact, all considered controllers will be extended by the binomial filters (35) with the time constant $T_f = 0.1$. Based on the plant approximations and taking into account the filters $Q_n(s)$ using HR and MHR we get

$$\begin{aligned}
&SPI + Q_1(s) : T_1 = 1.5;\ T_d = 2.5 + T_f;\ T_c = 2T_d;\ P(s) = \frac{1}{1+T_1 s}\\
&SPID + Q_2(s) : T_1 = 1.5;\ T_2 = 1;\ T_d = 1.5 + 2T_f;\ T_c = T_d;\ P(s) = \frac{1}{1+T_1 s};\ T_{f1} = \frac{T_2}{100}\\
&{}^1PI + Q_1(s) : T_1 = 2.5 + \frac{T_f}{2};\ T_d = 1.5 + \frac{T_f}{2};\ T_{c1} = 2T_d;\ {}^1P(s) = \frac{1}{1+T_1 s}\\
&{}^2PID + Q_2(s) : T_2 = 1.5 + \frac{T_f}{2};\ T_d = 1 + T_f;\ T_{c2} = T_d;\ {}^2P(s) = \frac{1}{(1+T_2 s)^2}\\
&{}^3PIDA + Q_4(s) : T_3 = 1.1667 + \frac{2T_f}{3};\ T_d = 0.5 + 2T_f;\ T_{c3} = T_d;\ {}^3P(s) = \frac{1}{(1+T_3 s)^3}\\
&{}^4PIDAJ + Q_6(s) : T_4 = 1 + \frac{3T_f}{4};\ T_d = 0.2T_4 + 3T_f;\ T_{c4} = T_d;\ {}^4P^2(s) = \frac{1}{(1+T_4 s)^2}.
\end{aligned} \tag{99}$$

Similar to the previous example, in order to obtain nearly ideal shapes of transients, SPI and ^{1}PI controllers are tuned with $T_c = T_{c1} = 2T_d$. All other controllers are tuned with the default value of $T_c = T_{ci} = T_d, i = 2, 3, 4$ with pre-filters (69) except for SPID and ^{4}PIDAJ, which use reduced-order pre-filters $^4P^2(s)$ (99).

Since ^{1}PID and ^{2}PIDA controllers in Example 4 did not confirm any significant improvement over the ^{1}PI and ^{2}PID controllers, we are no longer considering them in this comparison.

The obtained transients in Figure 23 show that even under a relatively high measurement noise and higher-order derivative actions by some controllers, it is possible to significantly accelerate the responses while maintaining a relatively low control efforts. Here, it is worth noting that the lowest values of the excessive controller effort are not produced by the simplest controllers, despite the slow transients. However, when we look at the control signal responses in Figure 23 (associated with the nearly ideal responses of the output), we see that they are nearly monotonic. One pulse is visible in them only in case of setpoint response under ^{4}PIDAJ with reduced-order pre-filter $^4P^2(s)$. Indications of 1P responses are also seen when using ^{2}PID and ^{3}PIDA controllers. Therefore, with respect to the Definition 5, these are the responses from DC0. In order for several pulses of the action variable to stand out more significantly (as discussed in Example 5), the level of measurement noise should be reduced.

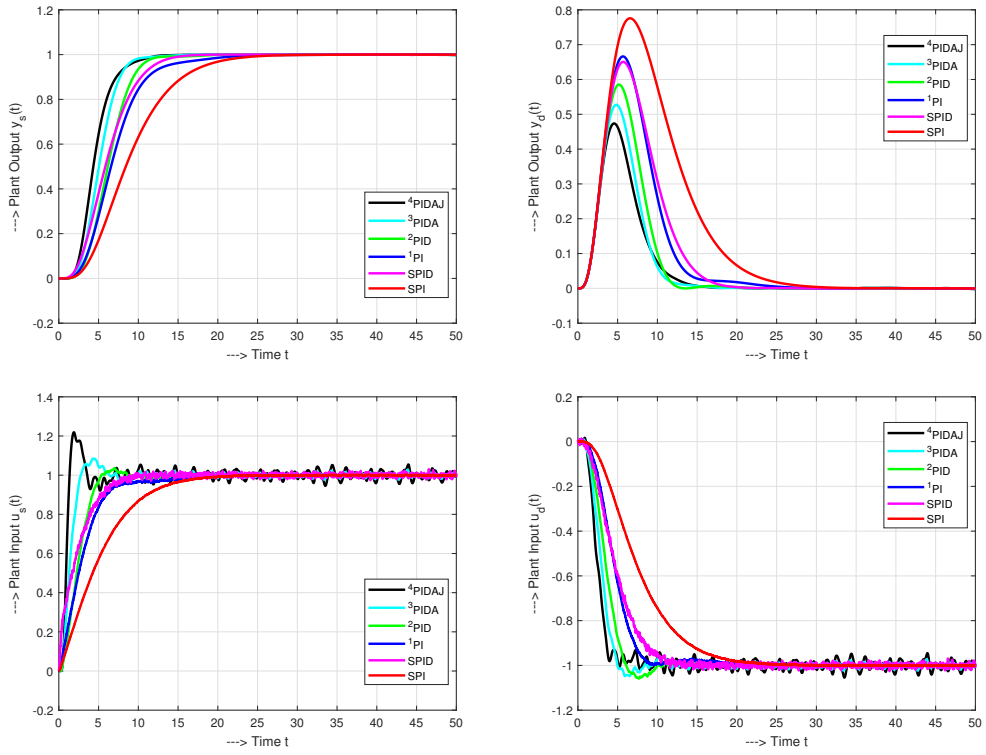

Figure 23. Setpoint and input disturbance unit step responses of the system (33) for SPI, SPID, ^{1}PI, ^{2}PID, ^{3}PIDA and ^{4}PIDAJ controllers combined with the noise attenuation filters and tuning parameters for a measurement noise with an amplitude $|\delta| \leq 0.1$ (i.e., up to 10% of the setpoint step) generated in Matlab/Simulink by a Uniform Random Number block.

Although the IAE values in Figure 24 (see also Tables 1 and 2) decrease when using higher-order controllers, they do not show entire behavior of the responses, which should take into account both the speed of transients and excessive controller effort or output oscillation. From this point of view, the corresponding cost functions (50) and (51) show interesting results.

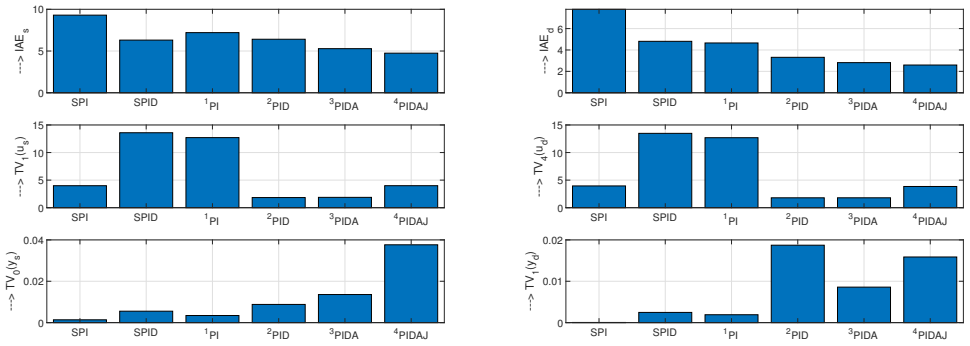

Figure 24. Performance measures corresponding to transients in Figure 23.

Namely, for $k = 1$, when placing equal emphasis on the speed of transients and shape deviations, the optimal values of $J_1(u_s)$ and $J_1(u_d)$ are achieved with ^{3}PIDA controller. By the same controller we get very good values also in terms of output wobbling $J_1(y)$. However, the best values of $J_1(y)$ are obtained by the slowest SPI controller. On the other hand, when reflecting with $k = 5$ increased demands on the speed of transients, the ^{4}PIDAJ controller, resulting in the fastest transients for both setpoint and disturbance rejection responses, yields very good values also in terms of combined cost functions $J_5(u)$ and $J_5(y)$.

Table 1. Evaluation of the setpoint responses from Example 6.

Controller	SPI	SPID	^{1}PI	^{2}PID	^{3}PIDA	^{4}PIDAJ
IAE_s	9.3005	6.3096	7.2040	6.4126	5.2836	**4.7447**
$TV_4(u_s)$	3.9871	13.5908	12.7076	1.8372	**1.8740**	3.9900
$TV_0(y_s)10^3$	**1.3649**	5.5289	3.4414	8.7998	13.5847	22.1266
$J_1(u_s)$	37.0818	85.7518	91.5451	11.7814	**9.9014**	18.9313
$J_1(y_s)10^2$	**1.2695**	3.4885	2.4792	5.6429	7.1777	10.4984
$J_5(u_s)10^{-5}$	2.7745	1.3591	2.4656	0.1992	**0.0772**	0.0959
$J_5(y_s)$	94.9803	55.2887	66.7736	95.4184	55.9382	**53.2052**

Table 2. Evaluation of the disturbance responses from Example 6.

Controller	SPI	SPID	^{1}PI	^{2}PID	^{3}PIDA	^{4}PIDAJ
IAE_d	7.8032	4.8093	4.6555	3.3136	2.8161	**2.5935**
$TV_4(u_d)$	3.9458	13.4804	12.6906	**1.7935**	1.7950	3.8461
$TV_1(y_d)10^2$	**0.0428**	2.4693	1.9083	18.7217	8.5841	15.8702
$J_1(u_d)$	30.7900	64.8315	59.0811	5.9429	**5.0548**	9.9748
$J_1(y_d)10^2$	**0.0334**	1.1876	0.8884	6.2037	2.4174	4.1159
$J_5(u_d)10^{-5}$	1.1416	0.3468	0.2775	0.0072	**0.0032**	0.0045
$J_5(y_d)$	**1.2392**	6.3532	4.1732	7.4794	1.5203	1.8621

6. Discussion: Everything Should Be Made as Simple as Possible, but Not Simpler

The discussion of the achieved results can be briefly summarized in a well-known statement attributed to Albert Einstein. The newly presented analytical design of HO-PID controllers and its comparison with SIMC and optimization-based iSIMC and other similar works leads us namely to the following comments—the published paper, "Probably the best simple PID tuning rules in the world" [43], was in fact mainly focused on "The best simple PI tuning rules". Reference [4] was written in the same direction by suggesting adhoc choice in numerous points, as the choice of the target transfer function, half-rule for the reduction of complex plant models, or performance evaluation based on the IAE and TV. Later on, the SIMC author with the co-author analyzed the possibilities and limitations of the original method [44]. Recently, they have tried to extend the design to the PID

controllers [6,22], where, they encountered the limitations of the initial postulates and tried to overcome them by using optimization methods. When trying to replace analytical approaches by the seemingly simpler numerical optimization approaches, we should note that:

- obtaining a perfect process model is frequently associated with trial-and-error approaches enabling to achieve the highest possible match between theoretical and experimentally obtained results (underpinned by appropriate identification results).
- These expectations are usually interpreted in terms of multicriteria cost functions, instead of a single general-purpose cost function and a single all-encompassing optimization.
- Diverse requirements led to the birth of fuzzy control based on the use of linguistically formulated conditions of optimality [45]. However, similar objectives can be easily achieved with simple analytical and modular approaches offering more direct relation to the tuning parameters, especially when they are designed to optimally cover the specific requirements.
- When looking for the optimal solution for a wider class of problems, the price to be paid is a wide range of existing and newly emerging controllers and methods for evaluating them. A simple list of existing solutions (as offered by [5]), with their ever-growing number, may not lead to a simplification and clarity of the situation. From this point of view, it seems more efficient to classify existing solutions into dynamic classes of control [17,39].
- From this perspective, the clear structure, openness, flexibility of adaptation and compatibility with the concept of dynamic classes can be considered as the main advantages of the newly proposed modifications to the original SIMC method.

It was found out that several of the proposed simplifications of the original SIMC were counterproductive or required certain modification. At the same time, as shown on the example of the system (33) taken from [4], there was identified the danger of exaggeration. Of course, simplifications are not avoided by the modified method. In order to suffice with the minimum possible number of parameters, it works with multiple time constants of the model already in the identification of the system, the choice of the target behavior, pre-filters and the choice of noise filters. By increasing the number of considered parameters, it would perhaps be possible to slightly improve the closed loop responses. However, the question is whether, in the face of all the existing uncertainties in real systems, it is worth it.

In the HO controller design and evaluation stages, it was essential to take into account the more complex shapes of transients. In addition to paying attention to the filter choice and limiting the main tuning parameter, this lead to design of pre-filters or setpoint weighting solutions. Our research led to final realization that, although based on SIMC, the core of its original concepts can be used on more complex controllers with higher order derivative actions.

7. Conclusions

This paper can be considered as an extension of the SIMC method to HO controllers with user-defined filter order. Although the paper focused on the stable process models, the SIMC method provided a versatile environment to satisfy challenging transient response requirements with some simple modifications. Noise attenuation requirements were met by choosing the filter order and time constant.

The key factor for the successful implementation of the HO controllers was appropriate signal filtering, since the higher-order derivatives without filters can be the main obstacle to practical implementation of the controller. The results of the illustrative examples showed that the proposed approach was suitable for implementation at different levels of process measurement noise.

The newly developed model-based approach for HO stable plants therefore provided excellent control results, which allow its implementation even in very demanding applications.

Author Contributions: Writing-original draft preparation, M.H. and D.V. Simulations, M.H. Editing, D.V. and M.H. Project administration, M.H. Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the grants APVV SK-IL-RD-18-0008 Platoon Modelling and Control for mixed autonomous and conventional vehicles: a laboratory experimental analysis, VEGA 1/0745/19 Control and modelling of mechatronic systems in emobility and P2-0001 financed by the Slovenian Research Agency.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Supported by Slovenská e-akadémia, n. o.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

1P	One-Pulse, response with 2 monotonic segments (1 extreme point)
2P	Two-Pulse, response with 3 monotonic segments (2 extreme points)
ART	Average Residence Time
BIBO	Bounded-Input-Bounded-Output
FOTD	First-Order Time Delayed
HO	Higher Order
HR	Half-Rule
I	Integrative
IMC	Internal Model Control
jOTD	j-Order Time Delayed
MHR	Modified Half-Rule
mP	m-Pulse, response with $m+1$ monotonic segments (m extreme points)
PI	Proportional-Integrative
PID	Proportional-Integrative-Derivative
PIDA	Proportional-Integrative-Derivative-Accelerative
PIDAJ	Proportional-Integrative-Derivative-Acceleration-Jerk
${}^{j}\mathrm{R}^{m}$	mth-order controller for jth-order stable plant, $m \geq j$
${}^{j}\mathrm{R}^{m}_{n}$	mth-order controller for jth-order stable plant combined with nth-order filter $Q_n(s)$, $m \geq j, n \geq 0$
SE	Speed - Effort
SIMC	Simple Control/Skogestad IMC
SPI	SIMC PI controller
SPID	SIMC PID controller
SW	Speed - Wobbling
SOTD	Second-Order Time Delayed
QOTD	Fourth-Order Time Delayed (with quadruple time constant)
TOTD	Third-Order Time Delayed

References

1. Åström, K.J.; Hägglund, T. *Advanced PID Control*; ISA, Research Triangle Park: Durham, NC, USA, 2006.
2. Tepljakov, A.; Alagoz, B.B.; Yeroglu, C.; Gonzalez, E.; HosseinNia, S.H.; Petlenkov, E. FOPID Controllers and Their Industrial Applications: A Survey of Recent Results. *IFAC-PapersOnLine* **2018**, *51*, 25–30. [CrossRef]
3. Efe, M.O. Fractional Order Systems in Industrial Automation: A Survey. *IEEE Trans. Ind. Inform.* **2011**, *7*, 582–591. [CrossRef]
4. Skogestad, S. Simple analytic rules for model reduction and PID controller tuning. *J. Process Control* **2003**, *13*, 291–309. [CrossRef]
5. O'Dwyer, A. *Handbook of PI and PID Controller Tuning Rules*, 3rd ed.; Imperial College Press: London, UK, 2009.
6. Grimholt, C.; Skogestad, S. Optimal PI and PID control of first-order plus delay processes and evaluation of the original and improved SIMC rules. *J. Process Control* **2018**, *70*, 36–46. [CrossRef]
7. Huba, M.; Vrančič, D.; Bisták, P. PID$^{m}_{n}$ Control for IPDT Plants. Part 1: Disturbance Response. In Proceedings of the 26th Mediterranean Conference on Control and Automation (MED), Zadar, Croatia, 19–22 June 2018.

8. Huba, M.; Vrančič, D. Comparing filtered PI, PID and PIDD^2 control for the FOTD plants. In Proceedings of the 3rd IFAC Conference on Advances in Proportional-Integral-Derivative Control, Ghent, Belgium, 9–11 May 2018.
9. Huba, M.; Vrančič, D. Introduction to the Discrete Time PID$_n^m$ Control for the IPDT Plant. In Proceedings of the 15th IFAC International Conference on Programmable Devices and Embedded Systems, Ostrava, Czech Republic, 23–25 May 2018.
10. Huba, M. Model-based higher-order PID control design. In Proceedings of the 21st IFAC World Congress, Berlin, Germany, 12–17 July 2020.
11. Huba, M.; Vrančić, D.; Bisták, P. PID Control with Higher Order Derivative Degrees for IPDT Plant Models. *IEEE Access* **2021**, *9*, 2478–2495. [CrossRef]
12. Skogestad, S. Tuning for Smooth PID Control with Acceptable Disturbance Rejection. *Ind. Eng. Chem. Res.* **2006**, *45*, 7817–7822. [CrossRef]
13. Huba, M. Exploring PID tuning strategies considering noise impact in the IPDT plant control. In Proceedings of the 16th IFAC Int. Conference on Programmable Devices and Embedded Systems, High Tatras, Slovakia, 29–31 October 2019.
14. Wei, Y.; Hu, Y.; Dai, Y.; Wang, Y. A Generalized Padé Approximation of Time Delay Operator. *Int. J. Control. Autom. Syst.* **2015**, *14*, 181–187. [CrossRef]
15. Chien, K.L.; Hrones, J.; Reswick, J. On the automatic control of generalized passive systems. *Trans. ASME* **1952**, *74*, 175–185.
16. Visioli, A. *Practical PID Control*; Springer: London, UK, 2006.
17. Huba, M.; Šimunek, M. Modular Approach to Teaching PID Control. *IEEE Trans. Ind. Electr.* **2007**, *54*, *6*, 3112–3121. [CrossRef]
18. Strejc, V. Näherungsverfahren für aperiodische Übergangscharakteristiken. *Regelungstechnik* **1959**, *7*, 124–128.
19. Huba, M. Constrained filtered PID Controller for IPDT plants. In Proceedings of the 27th Mediterranean Conference on Control and Automation (MED), Akko, Israel, 1–4 July 2019.
20. Segovia, V.R.; Hägglund, T.; Åström, K. Measurement noise filtering for PID controllers. *J. Process Control* **2014**, *24*, 299–313. [CrossRef]
21. Rivera, D.E.; Morari, M.; Skogestad, S. Internal model control. 4. PID controller design. *Ind Eng. Chem. Res.* **1986**, *25,1*, 252–265. [CrossRef]
22. Grimholt, C.; Skogestad, S. Optimal PID control of double integrating processes. *IFAC-PapersOnLine* **2016**, *49*, 127–132.
23. Åström, K.J.; Hägglund, T. *PID Controllers: Theory, Design, and Tuning*, 2nd ed.; Instrument Society of America: Research Triangle Park, NC, USA, 1995.
24. Feldbaum, A. *Optimal Control Systems*; Academic Press: Cambridge, MA, USA, 1965.
25. Pontrjagin, L.; Boltjanskij, V.; Gamkrelidze, R.; Miščenko, J. *The Mathematical Theory of Optimal Processes*; Interscience: New York, NY, USA, 1962.
26. Föllinger, O. *Regelungstechnik. 8. Auflage*; Hüthig Buch Verlag: Heidelberg, Germany, 1994.
27. Glattfelder, A.; Schaufelberger, W. *Control Systems with Input and Output Constraints*; Springer: Berlin, Germany, 2003.
28. Huba, M. Designing Robust Controller Tuning for Dead Time Systems. In *Int. Conf. System Structure and Control*; IFAC: Ancona, Italy, 2010.
29. Chen, W.H.; Yang, J.; Guo, L.; Li, S. Disturbance-Observer-Based Control and Related Methods – An Overview. *IEEE Trans. Ind. Electron.* **2016**, *63*, 1083–1095. [CrossRef]
30. Huba, M. Performance measures, performance limits and optimal PI control for the IPDT plant. *J. Process. Control.* **2013**, *23*, 500–515. [CrossRef]
31. Huba, M. Open flexible PD-controller design for different filtering properties. In Proceedings of the 39th Annual Conference of the IEEE Industrial Electronics Society (IECON), Vienna, Austria, 10–13 November 2013.
32. Huba, M. Filter choice for an effective measurement noise attenuation in PI and PID controllers. In Proceedings of the ICM2015, Nagoya, Japan, 6–8 March 2015.
33. Reswick, J.B. Disturbance-Response Feedback—A new control concept. *Trans. ASME* **1956**, *1*, 153–162.
34. Huba, M.; Kulha, P.; Skachová, Z. Two Dynamical Classes of PI-Controllers for the 1st Order Loops. In Proceedings of the Preprints 2nd IFAC Workshop "NTDCS" New Trends in Design of Control Systems, Smolenice, Slovak Republic, 7–10 September 1997; pp. 293–298.
35. Hanus, R.; Peng, Y. Conditioning technique for controllers with time delays. *IEEE Trans. Aututomatic Control* **1992**, *37*, 689–692. [CrossRef]
36. Kothare, M.; Campo, P.J.; Morari, M.; Nett, C.N. A Unified Framework for the Study of Anti-windup Designs. *Automatica* **1994**, *30*, 1869–1883. [CrossRef]
37. Peng, Y.; Vrančič, D.; Hanus, R. Anti-Windup, Bumpless and Conditioned Trnasfer Techniques for PID Controllers. *IEEE Control Syst.* **1996**, *16*, 48–57.
38. Peng, Y.; Vrančič, D.; Hanus, R. A Review of Anti-Windup, Bumpless and Conditioned Transfer. In Proceedings of the 13th IFAC World Congress, San Francisco, CA, USA, 30 June–5 July 1996; pp. 79–84.
39. Huba, M.; Bistak, P. Dynamic classes in the PID control. In Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251), San Diego, CA, USA, 2–4 June 1999; Volume 6, pp. 3868–3872.
40. Huba, M. Constrained pole assignment control. In *Current Trends in Nonlinear Systems and Control*; Menini, L., Zaccarian, L., Abdallah, C.T., Eds.; Birkhäuser: Boston, MA, USA, 2006; pp. 163–183.

41. Huba, M. Computer Design of Robust I-controller. In Proceedings of the IFAC World Congress, Milan, Italy, 28 August–2 September 2011; pp. 7468–7473.
42. Huba, M. Robust Tuning of the Simplest Dead Time Compensators. In Proceedings of the 2014 International Conference and Exposition on Electrical and Power Engineering (EPE), Iasi, Romania, 16–18 October 2014.
43. Skogestad, S. Probably the best simple PID tuning rules in the world. In Proceedings of the AIChE Annual Meeting, Reno, NV, USA, 4–9 November 2001.
44. Grimholt, C.; Skogestad, S. Optimal PI-Control and Verifcation of the SIMC Tuning Rule. *IFAC Proc. Vol.* **2012**, *45*, 11–22. [CrossRef]
45. Carvajal, J.; Chen, G.; Ogmen, H. Fuzzy PID controller: Design, performance evaluation, and stability analysis. *Inf. Sci.* **2000**, *123*, 249–270. [CrossRef]