Comparative Analysis of Discrete Derivative Implementations in PID Controllers

PETR PIVOŇKA, MICHAL SCHMIDT
Brno University Of Technology
Department of Control and Instrumentation
Kolejní 2906/4
CZECH REPUBLIC

Abstract: In technical practice it shows that the key problem in a PID controller realization is in its derivative part. With an overwhelming majority of compact controllers a continuous transfer function is described in their documentation, even though the implementation is discrete. A naïve replacement of the derivative by first-order difference is unsatisfactory. In order for a PID controller to be practically usable, its derivative action must be filtered. The article discusses multiple ways how the derivative can be implemented in discrete variants of the PID controller. Their respective advantages and pitfalls are compared. One of them, which exhibits responses most similar to the continuous version and is usable in the widest range of sampling periods, is then recommended.

Key–Words: PID, derivative, discrete implementation, noise amplification

1 Introduction

Step response of an idealised PID controller's derivative part is the Dirac delta function. For physical implementability an extra time constant is usually introduced, which functions as a filter. A nice side-effect of the derivative action's filtering is suppresion of noise that would be strongly amplified by the derivative otherwise. In discrete PSD controllers there is no problem with physical implementability, because a perfect difference $(1-z^{-1})$ does exist. Nevertheless the filtration is used here too, to suppress noise and to avoid extreme actions which the difference would create with short sampling periods.

A continuous PID controller with filtered derivative action has a transfer function (from [1]):

$$F_{\mathrm{R}}(s) = K\left(1 + \frac{1}{T_{\mathrm{I}}s} + \frac{T_{\mathrm{D}}s}{\frac{T_{\mathrm{D}}}{N}s + 1}\right). \quad (1)$$

where:
K .... proportional gain,
T_{I} .... integral time constant,
T_{D} ... derivative time constant,
N .... derivative gain limit; usually $N \in < 3; 20 >$.

From now on we will concern ourselves only with the third summand which represents the filtered derivative. As the derivative action's transfer function we will consider the expression

$$F_{\mathrm{D}}(s) = \frac{T_{\mathrm{D}}s}{\frac{T_{\mathrm{D}}}{N}s + 1}. \quad (2)$$

The corresponding step response $h_{\mathrm{D}}(t)$ is a falling exponential impulse:

$$h_{\mathrm{D}}(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}F_{\mathrm{D}}(s)\right\} = \mathcal{L}^{-1}\left\{\frac{N}{s + \frac{N}{T_{\mathrm{D}}}}\right\} = \\ = Ne^{-\frac{N}{T_{\mathrm{D}}}t}; \quad t > 0 \quad (3)$$

The total area of this impulse is

$$\int_0^\infty h_{\mathrm{D}}(t)\mathrm{d}t = N\int_0^\infty e^{-\frac{N}{T_{\mathrm{D}}}t}\mathrm{d}t = \\ = N\left[-\frac{T_{\mathrm{D}}}{N}e^{-\frac{N}{T_{\mathrm{D}}}t}\right]_0^\infty = T_{\mathrm{D}}. \quad (4)$$

The Dirac delta from a non-filtered derivative has the same area. Clearly the filtering of the derivative preserves its total reaction to a step change and it just smears the response into a longer time horizon.

2 Discrete implementations of the derivative part

There is more than one way how to get from the continuous implementation to a discrete one. The discrete system never behaves exactly the same as the continuous one and there are several approaches to discretise continuous systems. Each approach has its pros and cons. We will now discuss some of them.

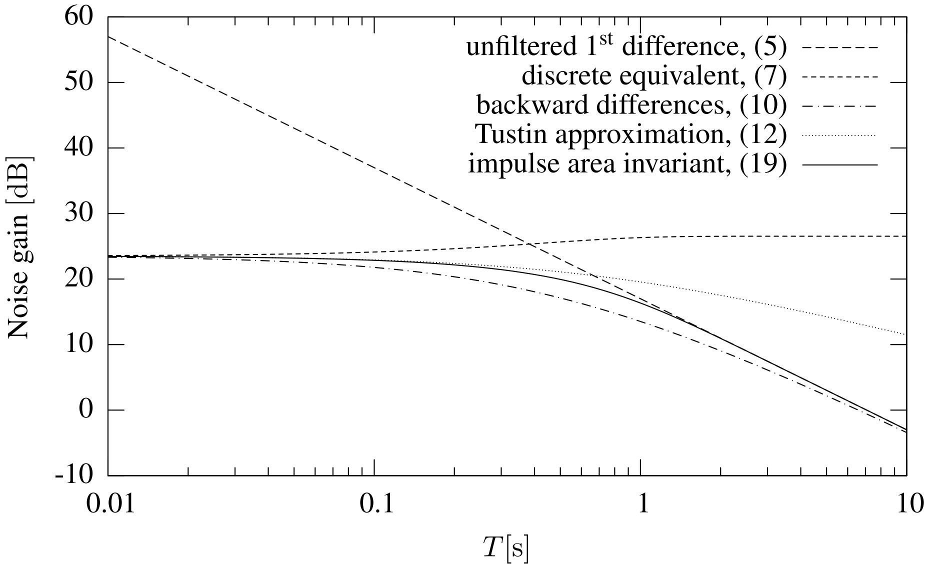

Figure 1: Noise amplification with different discrete derivative action implementations

2.1 Unfiltered first-order difference

The text-book implementation of a PSD controller has the derivative action in the form

$$F_{\mathrm{D}}(z) = \frac{T_{\mathrm{D}}}{T}(1 - z^{-1}). \tag{5}$$

Its obvious deficiency is that it produces extreme action values with short sampling periods. Therefore it is not usable in practice.

2.2 The discrete equivalent of a continuous system

In [2] a discrete variant of the filtered derivative action is proposed. It is a discrete equivalent of the continuous variant in that its step response exactly equals the response of the continous system $F_{\mathrm{D}}(s)$ in the sampling moments. The conversion is based on the idea that a zero-order hold is connected to the continuous system. The equivalent Z-transfer is then

$$\begin{aligned} F(z) &= \mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{1-e^{-sT}}{s}F(s)\right\}\right\} = \\ &= (1-z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{1}{s}F(s)\right\}\right\}. \end{aligned} \tag{6}$$

By applying (6) on (2) we get

$$\begin{aligned} F_{\mathrm{D}}(z) &= (1-z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{T_{\mathrm{D}}}{\frac{T_{\mathrm{D}}}{N}s+1}\right\}\right\} = \\ &= (1-z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{N}{s+\frac{N}{T_{\mathrm{D}}}}\right\}\right\} = \\ &= (1-z^{-1})\frac{Nz}{z-e^{-\frac{N}{T_{\mathrm{D}}}T}} = N\frac{1-z^{-1}}{1-e^{-\frac{NT}{T_{\mathrm{D}}}}z^{-1}}. \end{aligned} \tag{7}$$

2.3 Replacing the integrator with a summator

An alternative approach (also mentioned in [2]) comes from a modification of the continuous modelling schematics into a discrete one by replacing the integrator with its discrete analogy – the summator. The sampling interval has to be taken into account. The resulting model can be described by the transfer function

$$F_{\mathrm{D}}(z) = \frac{NT_{\mathrm{D}}(1-z^{-1})}{T_{\mathrm{D}} + (NT - T_{\mathrm{D}})z^{-1}}. \tag{8}$$

Good care must be taken when using this approach as the single pole of the transfer function is stable only if $\frac{NT}{T_{\mathrm{D}}} < 2$.

2.4 Backward differences

Different discrete transfer functions can be derived by substituting similarly behaving discrete operators for

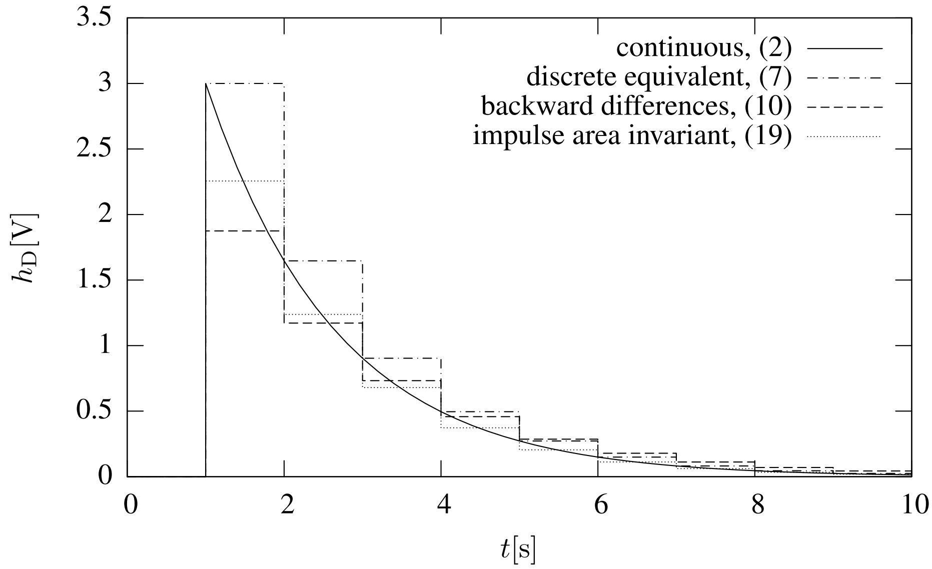

Figure 2: Comparison of derivatives' step responses

the Laplace operator s. Substitution by backward differences

$$s = \frac{1 - z^{-1}}{T} \tag{9}$$

gives

$$F_D(z) = \frac{T_D N}{T_D + NT} \frac{1 - z^{-1}}{1 - \frac{T_D}{T_D + NT} z^{-1}}. \tag{10}$$

It is prominently featured in [1] and it is also recommended by [3].

2.5 Tustin's approximation

Tustin's approximation

$$s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}} \tag{11}$$

leads to

$$F_D(z) = \frac{2NT_D}{2T_D + NT} \frac{1 - z^{-1}}{1 - \frac{2T_D - NT}{2T_D + NT} z^{-1}}. \tag{12}$$

This one too is presented in [1], which mentions its unpleasant property that its pole approaches the point -1 for larger values of T.

2.6 Derivative impulse area invariant

The proposed form of the discrete derivative action stems from the fact that the transfer function of a discrete system can be obtained if we have described its requested step response $h(k)$:

$$F(z) = (1 - z^{-1}) \mathcal{Z} \{h(k)\} \tag{13}$$

It is sufficient to specify the requirements on the step response's shape. We'll have two requirements which arise from the properties of the filtered continuous derivative's response:

- The response shall have a maximum in step $k = 0$ after which it shall fall exponentially – just like (3) does.
- The total area drawn by the response's graph shall equal T_D – exactly like it is in (4).

The first requirement is described easily:

$$h_D(k) = h_D(0) a^k; \quad a \in (0; 1) \tag{14}$$

The step response is a geometric progression with the quotient a. The response of the continuous variant is falling with the time constant $\frac{T_D}{N}$. For the discrete sequence the equivalent quotient is

$$a = e^{-\frac{NT}{T_D}}. \tag{15}$$

The response's area is composed of many little rectangular areas. The second request is written like this:

$$T_D = \sum_{k=0}^{\infty} T h_D(k) = T \sum_{k=0}^{\infty} h_D(0) \left(e^{-\frac{NT}{T_D}} \right)^k =$$

$$= h_D(0) T \sum_{k=0}^{\infty} \left(e^{-\frac{NT}{T_D}} \right)^k \tag{16}$$

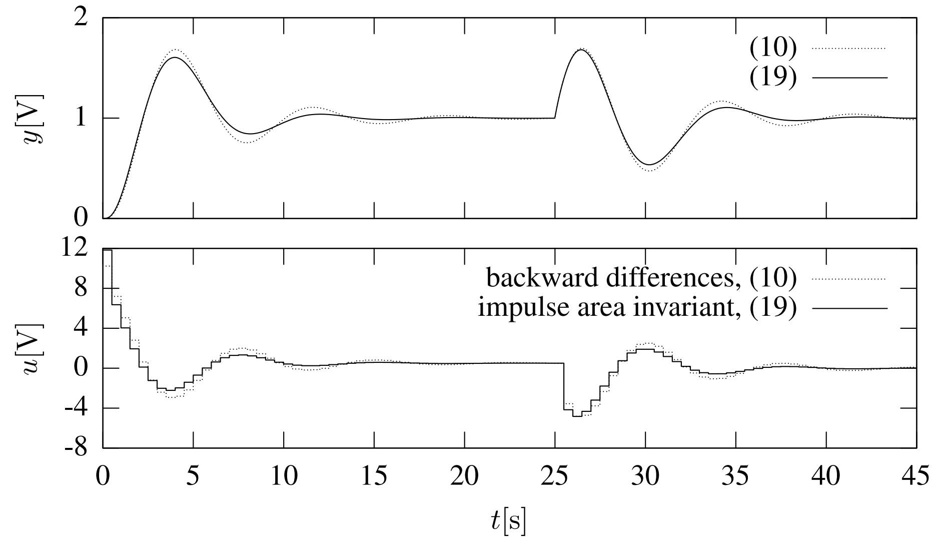

Figure 3: Discrete PID control of the system $F(s) = \frac{2}{(10s+1)(s+1)^2}$

Clearly it is an infinite geometric series, therefore:

$$T_\mathrm{D} = h_\mathrm{D}(0)T\frac{1}{1-e^{-\frac{NT}{T_\mathrm{D}}}} \qquad (17)$$

From here we'll separate $h_\mathrm{D}(0)$ and substitute into (14). The final form of the step response is:

$$h_\mathrm{D}(k) = \frac{T_\mathrm{D}}{T}\left(1-e^{-\frac{NT}{T_\mathrm{D}}}\right)\left(e^{-\frac{NT}{T_\mathrm{D}}}\right)^k \qquad (18)$$

By application of (13) we get the sought transfer function of the discrete derivative action:

$$F_\mathrm{D}(z) = \frac{T_\mathrm{D}}{T}\left(1-e^{-\frac{NT}{T_\mathrm{D}}}\right)\frac{1-z^{-1}}{1-e^{-\frac{NT}{T_\mathrm{D}}}z^{-1}} \qquad (19)$$

3 Discussion

Obviously there is an abundance of possible discrete derivative implementations. An engineer facing the task of programming a PID control algorithm would have to choose one. It is the purpose of this section to ease his decision by eliminating some of them from his consideration. It will be shown that not all are universally applicable, especially when a wide range of operating sampling periods is desired. An implementation will be recommended as the most universal and predictable one.

Given the requirement of operation with any sampling period, the 'integrator-by-summator replacement' (8) can be discarded right away, because it is unstable for long sampling periods. All other $F_\mathrm{D}(z)$ variants are always stable.

Stability is not a sufficient condition though. Some variants exhibit stable but oscillating responses. This is the case with Tustin approximation. For this reason it cannot be generally recommended.

Another failure mode is more subtle. It is demonstrated in Figure 1. It compares how each variant amplifies noise which will always be present on sensors in real systems. The 'discrete equivalent' is a notable disappointment here. Not only it amplifies noise more than other filtering variants, it even fares worse than unfiltered difference on long sampling periods. Its step response (Figure 2) shows the reason why. The area is larger than that of the continuous response. There remain only two generally usable variants: (10) and (19).

From a quick glance at Figure 1 it might seem that the backward differences replacement is better than impulse area invariant method, because it amplifies noise less. However, it can be easily shown that the transfer functions of these two methods are in fact equivalent. They only differ in their respective interpretation of the gain limit N. In the midrange of sampling periods, the backward differences variant's value of N has a significantly diminished weight. Therefore it performs stronger filtering than would correspond to the original continuous representation. The filter may have a detrimental effect on the controller's response (see Figure 3 where the same parameters were used for both controllers). To prove this is the only difference, notice that both transfer func-

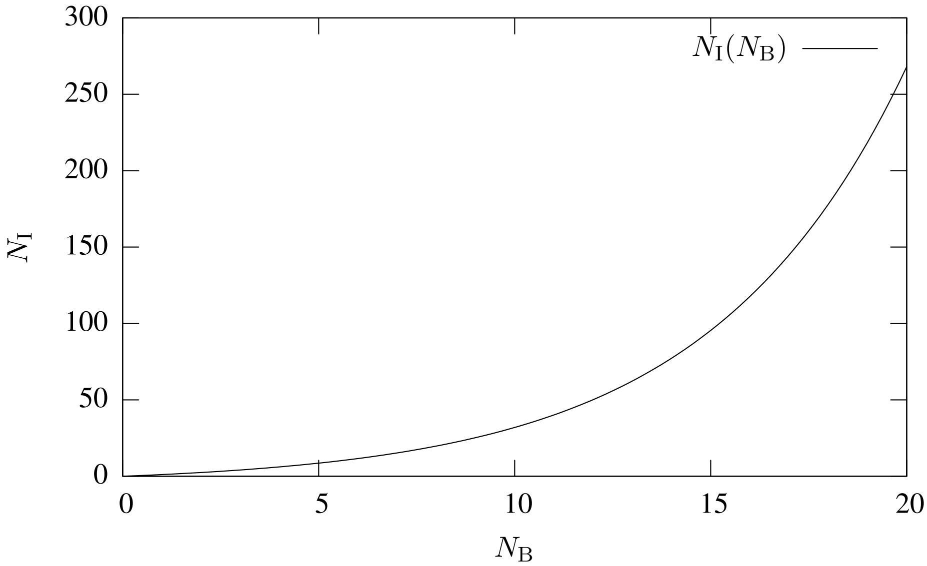

Figure 4: Gain limit conversion for $T_D = 1$ s and $T = 0,2$ s

tions are of the same form:

$$F_D(z) = A_i \frac{1 - z^{-1}}{1 - a_i z^{-1}}; \quad i \in \{B, I\} \tag{20}$$

where the coefficients are

$$A_B = \frac{T_D N_B}{T_D + N_B T} \tag{21}$$

$$a_B = \frac{T_D}{T_D + N_B T} \tag{22}$$

for backward differences and

$$A_I = \frac{T_D}{T}\left(1 - e^{-\frac{N_I T}{T_D}}\right) \tag{23}$$

$$a_I = e^{-\frac{N_I T}{T_D}} \tag{24}$$

for impulse area invariant. N_I and N_B are the respective gain limits. They represent the same thing in both implementations. By a suitable choice of their values we can obtain a perfect match of both implementations. Namely, it is possible to find an exact relation between N_I and N_B so that this holds:

$$A_M = A_B \quad \wedge \quad a_M = a_B \tag{25}$$

The equations do not contradict. Both result in (demonstrated in Figure 4):

$$N_B = \frac{T_D}{T e^{-\frac{N_I T}{T_D}}} - \frac{T_D}{T} \tag{26}$$

4 Conclusion

Several variants of the derivative component of discrete PID controllers were examined and compared. In the end only two of them proved sufficiently universal. Due to its best similarity with the continuous implementation and more adequate filtering, the 'impulse area invariant' can be declared the winner. It is noteworthy that exactly the same transfer function is hinted in [1] in a table of coefficients under the name 'ramp equivalence'. However, its authors did not dwell upon the differences between the methods. Such a comparison together with an original deduction of the method are the main contributions of this article.

Acknowledgement: The paper has been prepared as a part of the solution of Czech Science Foundation GAČR project No. 102/06/1132 Soft Computing in Control and by the Czech Ministry of Education in the frame of MSM 0021630503 Research Intention MICROSYN New Trends in Microelectronic Systems and Nanotechnologies.

References:

[1] Åström, K., Wittenmark, B.: *Computer-controlled systems (3rd ed.)*, Prentice-Hall, NJ, USA, 1997, ISBN 0-13-314899-8.

[2] Pivoňka, P.: *Digital Control Technology* (in Czech, electronic text), VUT, Brno, CZ, 2003.

[3] Bobál, V., Böhm, J., Prokop, R., Fessl, J.: *Praktické aspekty samočinně se nastavujících regulátorů: algoritmy a implementace*, VUTIUM, Brno, CZ, 1999, ISBN 80-214-1299-2.