Design of Closed-Loop Input Shaping Controllers

Ulrich Staehlin[1] staehlin@eng.buffalo.edu
Tarunraj Singh[2] tsingh@eng.buffalo.edu
Department of Mechanical and Aerospace Engineering
State University of New York at Buffalo, Buffalo, NY 14260
http://code.eng.buffalo.edu/tdf

Abstract

Input Shaping/Time Delay Prefilter are techniques for robust vibration control of slewing flexible structures. These are open-loop schemes and cannot handle uncertain initial conditions and effects of disturbances. In this paper, we develop a technique for the design of closed loop input shapers and illustrate it on a simple benchmark problem. Range of uncertainties over which the system remains stable is determined and is used as a design tool. It should be noted that the structure of the controller is similar to the Internal Model Control (IMC).

1 Introduction

The seminal paper of Singer and Seering [1] proposed a technique to desensitize the *Posicast* controller proposed by Tallman and Smith [2]. Since then, numerous papers have appeared which deal with the design of robust input shaper. Singh and Vadali [3] have demonstrated that time-delay prefilters result in the same performance as input shapers. They pose the problem as the design of a time-delay filter whose zeros cancel the underdamped poles of the plant. To achieve robustness, they demonstrate that by locating multiple zeros of the time-delay filter at the expected location of the uncertain underdamped poles of the system, the shaped input profile is desensitized to errors in system frequency and damping. Singh and Vadali [4] also propose a simple design technique where the delay-time of the time-delay filter are specified by the user. These time-delay filters will be used in this work for the design of closed loop input shapers.

Earlier closed loop shaped-input controllers add the input shaper to a closed loop system and adjust or derive the parameters of the controller according to the modified response. Kapila et al. [5] used a standard input shaper in conjunction with a full-state feedback controller which encloses the prefilter and plant. The feedback matrix has been derived using an LMI approach. Zuo and Wang [6] placed an input shaper in a closed loop consisting of a PD controller and an inverse dynamics linearization.

In this paper a structure also used in Internal Model Control (IMC) will be taken to transform the feedforward controller to a feedback controller.

The proposed technique begins with a motivation for the selection of the form of the feedback controller in Section 2. This is followed by a stability analysis in Section 3. Section 4 presents a simple approach for synthesizing the controller followed by illustrating the controller on a simple system in Section 5.

2 Design of the Controller

The motivation for the proposed controller is a simple technique described by Trächtler [7] for the design of deadbeat controllers. This approach describes a technique to design a control profile $\bar{u}(t)$ to achieve a desired performance which results in an open loop system. This control profile is then used to synthesize a feedback controller to achieve the same performance.

For our derivation we assume that the system is required to perform a rest-to-rest maneuver and the corresponding input is $w(t) = w_0 1(t)$, where $1(t)$ is the Heaviside step function. For the reference input $w(t)$, we need to determine a feedback controller which will result in $\bar{u}(t)$. With the knowledge of the system response, the tracking error $e(t)$ will be used in conjunction with $\bar{u}(t)$ to derive the feedback controller. This controller is given by

$$G_C(s) = \frac{\bar{U}(s)}{E(s)} \tag{1}$$

where $\bar{U}(s)$ and $E(s)$ are the Laplace transforms of $\bar{u}(t)$ and $e(t)$ respectively. Since we know our control profile $\bar{u}(t)$, the error is given by

$$e(t) = w(t) - y(t) = w(t) - \bar{u}(t) * h(t) \tag{2}$$

[1] Graduate Student, Mechanical & Aerospace Engineering
[2] Associate Professor, Mechanical & Aerospace Engineering

where $h(t)$ is the impulse response of the plant. This is illustrated in Figure 1 where $G_{P,A}(s)$ is the assumed transfer function for the plant and $G_{FF}(s)$ is the corresponding transfer function of the feedforward controller.

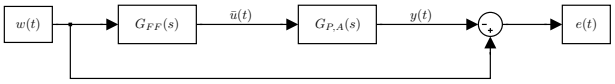

Figure 1: System showing the derivation of the error

Equation (1) can be rewritten using Equation (2) as

$$G_C(s) = \frac{\bar{U}(s)}{E(s)} = \frac{\bar{U}(s)}{W(s) - \bar{U}(s)H(s)}$$

The transfer function $G_C(s)$ represents a positive feedback loop with the feedforward controller in the forward path and a model of the plant in the feedback path. With $\bar{U}(s) = W(s)G_{FF}(s)$ and $H(s) = G_{P,A}(s)$ this leads to

$$\begin{aligned} G_C(s) &= \frac{W(s)\,G_{FF}(s)}{W(s) - W(s)\,G_{FF}(s)\,G_{P,A}(s)} \\ &= \frac{G_{FF}(s)}{1 - G_{FF}(s)\,G_{P,A}(s)}. \end{aligned} \tag{3}$$

For this derivation we have not constrained ourselves to any specific form of $\bar{u}(t)$. It could represent the output of a feedback controller, a feedforward controller, or it could be arbitrarily chosen or derived to get a desired response (like for the deadbeat controller). Since we already have time-delay prefilters designed for robust vibration control, we exploit them to generate $\bar{u}(t)$. Figure 2 shows a block diagram of the final feedback system with the controller highlighted.

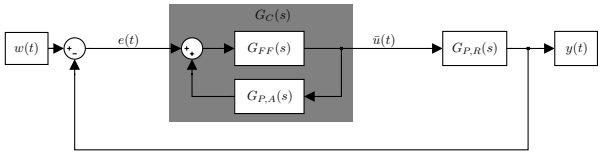

Figure 2: Feedback loop using a feedforward controller

The structure of the feedback controller shown in figure 2 is also used in Internal Model Control (IMC) (see for example [8]). The main difference is that for IMC the controller labeled $G_{FF}(s)$ is not known and IMC is used to determine it. In our case, we use this structure to go from a feedfoward to a feedback controller, with complete knowledge of all the parts of the controller.

Assuming the general form of a time-delay filter as:

$$G_{FF}(s) = \sum_{i=0}^{m} A_i\, e^{-i\,sT} \tag{4}$$

where m is the number of delays, T is the delay time and A_i are the gains for the different delays. The feedback controller reduces to

$$\begin{aligned} G_C(s) &= \frac{G_{FF}(s)}{1 - G_{FF}(s)\,G_{P,A}(s)} \\ &= \frac{\sum_{i=0}^{m} A_i\, e^{-i\,sT}}{1 - G_{P,A}(s)\sum_{i=0}^{m} A_i\, e^{-i\,sT}} \end{aligned} \tag{5}$$

which will be used for stability analysis.

One drawback of this proposed method is that one has to know the plant exactly since it is used to calculate the denominator of the transfer function of the controller (Equation (3)).

3 Stability Analysis

For the stability analysis only undamped system will be considered. Therefore the transfer functions of the assumed and real plant are

$$G_{P,A}(s) = \frac{\omega_A^2}{s^2 + \omega_A^2} \tag{6}$$

$$G_{P,R}(s) = \frac{\omega_R^2}{s^2 + \omega_R^2} \tag{7}$$

respectively. Here ω_R is the real natural frequency and ω_A is the assumed natural frequency. Giving the designer the greatest flexibility to influence the performance of the system, the transfer function of the Proportional Plus User Selected Multiple Delay (PPUSMD) controller is selected which is,

$$G_{FF}(s) = A_0 + A_1 e^{-sT} + A_2 e^{-2sT} \tag{8}$$

with

$$A_0 = \frac{1}{2 \cdot (1 - \cos(\omega_A T))} \tag{9}$$

$$A_1 = \frac{-2\cos(\omega_A T)}{2 \cdot (1 - \cos(\omega_A T))} \tag{10}$$

$$A_2 = \frac{1}{2 \cdot (1 - \cos(\omega_A T))} = A_0 \tag{11}$$

and T chosen by the designer. This leads to the controller's transfer function

$$G_C(s) = \frac{\left(A_0 + A_1 e^{-sT} + A_2 e^{-2sT}\right)\left(s^2 + \omega_A^2\right)}{\left(s^2 + \omega_A^2\right) - \omega_A^2\left(A_0 + A_1 e^{-sT} + A_2 e^{-2sT}\right)} \tag{12}$$

In order to narrow down the possible region of stability, the characteristic equation

$$1 + \frac{\left(A_0 + A_1 e^{-sT} + A_2 e^{-2sT}\right)\left(s^2 + \omega_A^2\right)}{\left(s^2 + \omega_A^2\right) - \omega_A^2\left(A_0 + A_1 e^{-sT} + A_2 e^{-2sT}\right)} \frac{\omega_R^2}{s^2 + \omega_R^2} = 0 \tag{13}$$

has to be solved for $s = j\phi$, resulting in the points where the system changes from an unstable to a stable region and vice-versa. This leads to

$$\phi = \pm k\frac{\pi}{T} \tag{14a}$$

$$\omega_R^2 = k^2\frac{\pi^2}{T^2}\cdot\frac{k^2\frac{\pi^2}{T^2}+\omega_A^2\left(A_0+A_1(-1)^k+A_2-1\right)}{k^2\frac{\pi^2}{T^2}\left(1+A_0+A_1(-1)^k+A_2\right)-\omega_A^2} \tag{14b}$$

$$k\in\mathbb{N}\quad k\neq\frac{\omega_A T}{\pi}\text{ if }\frac{\omega_A T}{\pi}\text{ is an integer}$$

or, after plugging in A_0, A_1 and A_2 and rearranging the Equation (14) results in

$$\phi = \pm k\frac{\pi}{T} \tag{15a}$$

$$\omega_R^2 = k^2\tfrac{\pi^2}{T^2}\times \frac{k^2\frac{\pi^2}{T^2}(\cos(\omega_A T)-1)+\omega_A^2\cos(\omega_A T)\left((-1)^k-1\right)}{k^2\frac{\pi^2}{T^2}(\cos(\omega_A T)+(-1)^k\cos(\omega_A T)-2)+\omega_A^2(1-\cos(\omega_A T))} \tag{15b}$$

$$k\in\mathbb{N}\quad k\neq\frac{\omega_A T}{\pi}$$

Another solution of equation (13) is

$$\phi = \pm\,\omega_A \tag{16}$$

$$\omega_R = \omega_A. \tag{17}$$

For ω_R corresponding to $k=\frac{\omega_A T}{\pi}$, both the numerator and the denominator of the characteristic equation (13) are 0. Taking the limit shows, that the characteristic equation is satisfied for this point, resulting in $\omega_R = \omega_A$, which is already a solution (see equation (17)).

Figure 3 shows the possible points of stability changes corresponding to $k = 1\ldots6$ for $T = 0\ldots6$ seconds and $\omega_A = 3\frac{rad}{sec}$. In addition, the delay time for the so called PPD (single delay) and PPMD (multiple delay) controllers (which correspond to prefilters defined by Equation (8) where the delay times are $T=\frac{\pi}{\omega_A}$) is marked to give some orientation (see [4] for more details on these controllers).

We will look at the slope of the root locus at the crossing points with the imaginary axis (see equations (15) and (17)) to find the regions of stability. The equation for the root locus we use here is derived by rewriting equation (13)

$$F_{RL,\omega_R^2}(s) = s^2\left(s^2+\omega_A^2(1-A_0-A_1e^{-sT}-A_2e^{-2sT})\right)+\omega_R^2\left(\omega_A^2+s^2(1+A_0+A_1e^{-sT}+A_2e^{-2sT})\right)=0 \tag{18}$$

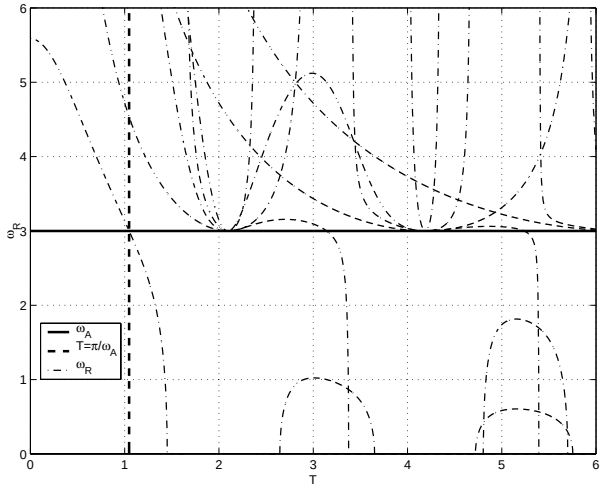

Figure 3: Points of possible changes of stability for $\omega_A = 3\frac{rad}{sec}$. Only the important curves are plotted.

To be able to look at the real and imaginary part separately we will substitute s, the complex variable in equation (18), by $s=\sigma+j\phi$. This leads to the two slopes

$$\frac{d\sigma}{d\omega_R} = \Re\left\{\frac{ds}{d\omega_R}\right\} \tag{19}$$

$$\frac{d\phi}{d\omega_R} = \Im\left\{\frac{ds}{d\omega_R}\right\} \tag{20}$$

with

$$\frac{ds}{d\omega_R} = -\frac{\frac{\partial F_{RL,\omega_R^2}}{\partial\omega_R}}{\frac{\partial F_{RL,\omega_R^2}}{\partial s}} \tag{21}$$

First we will look at the crossing at $\omega_R=\omega_A$, also meaning $\phi=\pm\omega_A$. Calculating the slope (using L'Hospital's rule) leads to

$$\left.\frac{d\sigma}{d\omega_R}\right|_{\substack{\sigma=0\\ \phi=\omega_A\\ \omega_R=\omega_A}} = \frac{1}{4}\left(\cos(\omega_A T)-1\right) \tag{22}$$

As can easy be seen, this slope is always less than 0, except for $T=\frac{2n\pi}{\omega_A}$, $n\in\mathbb{N}$. This implies, at this crossing the system is always going from an unstable to a possibly stable region, except at the specified T. For $\omega_R=\omega_A$, the closed loop system is asymptotically stable, meaning there are no unstable poles at this point, since this corresponds to perfect knowledge of the system model and the structure of the controller results in a classic time-delay feedforward controller which is asymptotically stable. Since there are no other crossings of the imaginary axis for $\omega_R=\omega_A$, there can not be any new unstable poles for ω_R slightly larger than ω_A. Therefore the crossing at this point goes from an unstable to a stable region.

Substituting T in equation (15) by $\frac{2n\pi}{\omega_A}$, the only configurations where equation (22) is not less than zero, leads to $\omega_R^2 = \omega_A^2$. That means, that the only crossing of the imaginary axis is at $\omega_R = \omega_A$. For $\omega_R = 0$, there are 3 poles at $s = 0$. From root locus theory we know, that this means that 3 branches of the root locus start at $s = 0$ and the angle between them is 120°. One of these branches is going towards $s = -\infty$, the others are going towards the unstable area. Therefore the system has to be unstable for $0 \leq \omega_R < \omega_A$, since there are no crossings of the imaginary axis before. The stability of the system for $\omega_R \geq \omega_A$ will be studied later.

First we consider $k = 2n$ with $n \in \mathbb{N}$, which corresponds to k being even. The slope simplifies for even k to

$$\left.\frac{d\sigma}{d\omega_R}\right|_{\sigma=0} = 2\frac{\left(8\pi^2n^2 - \omega_A^2T^2\right)^2}{4\pi^2n^2 - \omega_A^2T^2} \times \frac{\Re\left\{\sqrt{\frac{1}{8\pi^2n^2 - \omega_A^2T^2}}\left(4\pi^2n^2 - \omega_A^2T^2 + 8j\pi n\right)\right\}}{\left(4\pi^2n^2 - \omega_A^2T^2\right)^2 + 64n^2\pi^2} \tag{23}$$

This leads to three cases

$$\left.\frac{d\sigma}{d\omega_R}\right|_{\sigma=0} = 2\frac{\left(8\pi^2n^2 - \omega_A^2T^2\right)^2}{4\pi^2n^2 - \omega_A^2T^2} \times \frac{\sqrt{\frac{1}{\left|8\pi^2n^2 - \omega_A^2T^2\right|}}}{\left(4\pi^2n^2 - \omega_A^2T^2\right)^2 + 64n^2\pi^2} \times \tag{24}$$

$$\begin{cases} 4\pi^2n^2 - \omega_A^2T^2 & T < \sqrt{8}\frac{n\pi}{\omega_A} \\ 0 & T = \sqrt{8}\frac{n\pi}{\omega_A} \\ -8\pi n & T > \sqrt{8}\frac{n\pi}{\omega_A} \end{cases}$$

For $T \neq \sqrt{8}\frac{n\pi}{\omega_A}$, the slope is always > 0. That means, increasing ω_R leads to increasing σ. Since $\sigma = 0$ at the crossing points, the system is unstable for ω_R greater than the corresponding $\omega_R(k)$ (equation (15)). For $T = \sqrt{8}\frac{n\pi}{\omega_A}$, the slope is zero. Looking at a plot of the possible changes of stability (like figure 3) we can see, that the only region we are interested in is the one for $\frac{(k-1)\pi}{\omega_A} < T < \frac{k\pi}{\omega_A}$. For a T not in this region an ω_R for a different k defines the bounds of the stability region. Therefore, we don't have to care about the case where the slope is 0, since this corresponds to

$$T = \frac{\sqrt{2}k\pi}{\omega_A} \tag{25}$$

which is outside the bounds we are currently considering. Hence, a different k defines the stability boundary.

For $k = 2n + 1$ with $n \in \mathbb{N}$, the odd case, the slope is given by

$$\left.\frac{d\sigma}{d\omega_R}\right|_{\sigma=0} = \Re\left\{2\sqrt{S_{\text{slope}}}\frac{N_{\text{slope}}}{D_{\text{slope}}}\right\} \tag{26a}$$

with

$$S_{\text{slope}} = \frac{2\omega_A^2T^2\cos(\omega_AT)}{\omega_A^2T^2(\cos(\omega_AT) - 1) + 2\pi^2(2n+1)^2} - \frac{\pi^2(2n+1)^2(\cos(\omega_AT) - 1)}{\omega_A^2T^2(\cos(\omega_AT) - 1) + 2\pi^2(2n+1)^2} \tag{26b}$$

$$N_{\text{slope}} = \left((\cos(\omega_AT) - 1)\omega_A^2T^2 + 2(2n+1)^2\pi^2\right)^2 \tag{26c}$$

$$D_{\text{slope}} = \pi(2n+1)\left(1 - \cos^2(\omega_AT)\right) \times \left(\pi^2(2n+1)^2 - \omega_A^2T^2\right)^2 + 4j(\cos(\omega_AT) - 1) \times \left(\pi^2(2n+1)^2 - \omega_A^2T^2\right) \times \left((2n+1)^2\pi^2 + \omega_A^2T^2\cos(\omega_AT)\right) \tag{26d}$$

It can be seen, that $S_{\text{slope}} > 0$ for $T < \frac{(2n+1)\pi}{\omega_A}$, N_{slope} is always > 0 and $\Re\{D_{\text{slope}}\}$ is also always ≥ 0. As stated before, the only region we are interested in is $\frac{(k-1)\pi}{\omega_A} < T < \frac{k\pi}{\omega_A}$. For a T not in this region an ω_R for a different k defines the bounds of the stability region. In this region of interest, we see, that the slope is always positive, meaning the system is again going unstable for an ω_R greater than the one at the crossing point.

Now we can also say, that for $T = \frac{2n\pi}{\omega_A}$ the system is only stable for $\omega_R = \omega_A$, since all the crossings are at that point and we just showed that the system turns unstable for any point above one of these crossings.

Summing that up we can say, the the system is stable for

$$\omega_A \leq \omega_R \leq \min(\omega_R(k) > \omega_A) \tag{27}$$

Figure 4 shows this region of stability for $\omega_A = 3\frac{rad}{sec}$.

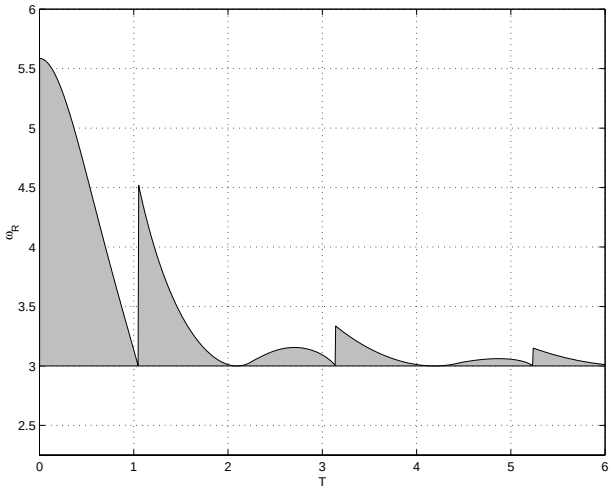

Figure 4: Region of stability for $\omega_A = 3\frac{rad}{sec}$

4 Selecting ω_A for fixed ω_R and T

From the stability analysis we know, that the lower stability bound is always $\omega_A = \omega_R$. The upper bound is given by equation (15), with $\frac{(k-1)\pi}{\omega_A} \leq T \leq \frac{k\pi}{\omega_A}$. Rearranging these equations we are able to find

$$\begin{aligned} \frac{\omega_R T}{\pi} \leq & k \leq \frac{\omega_R T}{\pi} + 1 \\ & k \in \mathbb{N} \end{aligned} \tag{28}$$

Using this k, ω_A can be freely chosen in the bounds

$$\frac{(k-1)\pi}{T} \leq \omega_A \leq \frac{k\pi}{T} \tag{29}$$

$$\omega_R \geq \omega_A \tag{30}$$

After choosing ω_A, the upper stability bound ω_R should be calculated using equation (15) to find out if there is enough stability margin. If not, ω_A should be changed and the bound checked until the requirements are met. Figure 5 shows a flowchart of the entire tuning process. Plotting the upper bound of ω_R versus ω_A helps in choosing a good ω_A. Figure 6 shows such a plot for $T = 0.5$ with $\omega_R = 3\frac{rad}{sec}$ marked to give some guidance.

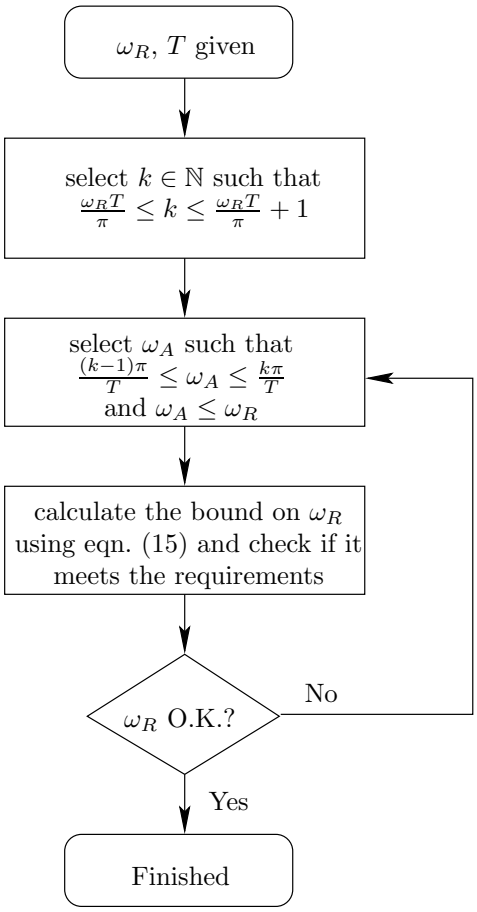

Figure 5: Parameter Tuning Process

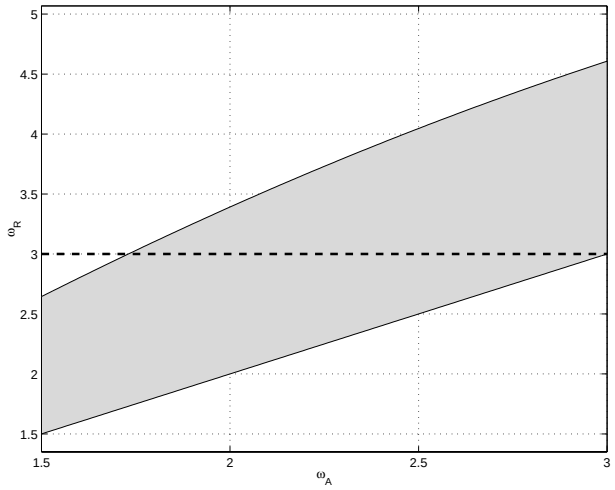

Figure 6: Stability region for $T = 0.5sec$. $\omega_R = 3\frac{rad}{sec}$ is marked for guidance

5 Simulations

The system for the simulations consists of the derived controller and a second order plant with no damping (see equation (7)) in a feedback setting. The time scales of the following plots are different from each other to show the interesting parts in more detail. At time $t = 0$ there was always a step from 0 to 1 applied to all of the systems.

For the simulation ω_A, the assumed natural frequency is chosen to be $3\frac{rad}{sec}$. Figure 7 shows the step response of a system with $T = \frac{\pi}{3\omega_A}$ for matching assumed and real natural frequency. It can clearly be seen, that the steady state value is reached without overshoot in twice the delay time. This is also the response you would get using the feedforward version of the controller with the same delay time and the same real plant.

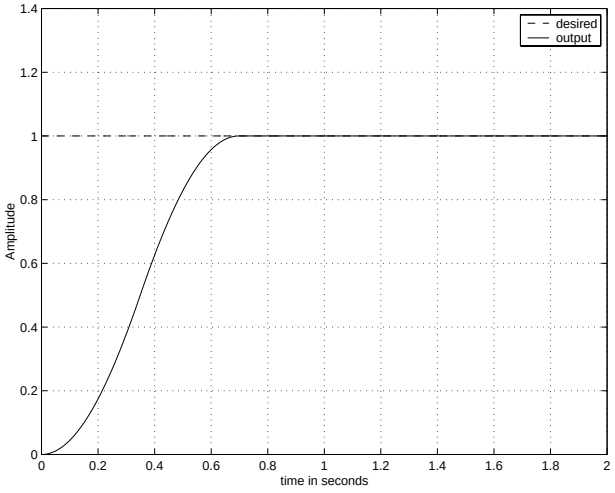

Figure 7: Simulation for $\omega_R = \omega_A = 3\frac{rad}{sec}$ with $T = \frac{\pi}{3\omega_A}$

Figure 8 shows the step response using the same controller, but this time the real natural frequency is $\omega_R = 4\frac{rad}{sec}$, which means 33% error. Even for this configuration the system reaches the steady state value in a reasonably short time and the overshoot stays within 20%.

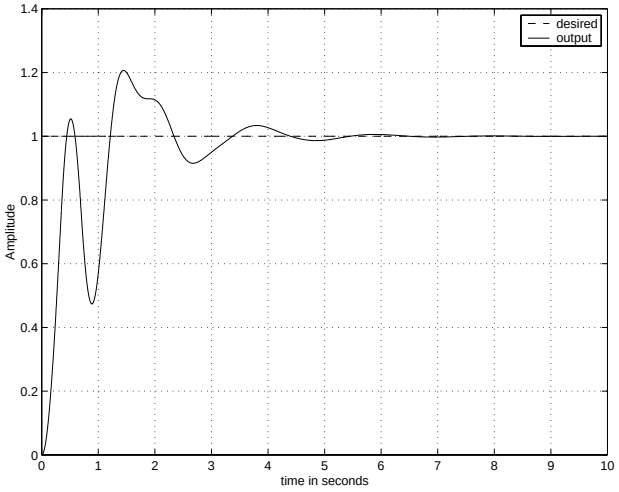

Figure 8: Simulation for $\omega_A = 3\frac{rad}{sec}$ and $\omega_R = 4\frac{rad}{sec}$ with $T = \frac{\pi}{3\omega_A}$

For validation purposes only, figure 9 shows, that the system goes unstable for the real natural frequency ω_R being less than the assumed natural frequency ω_A. The upper bounds for ω_R as illustrated in figure 4 were corroborated by simulations.

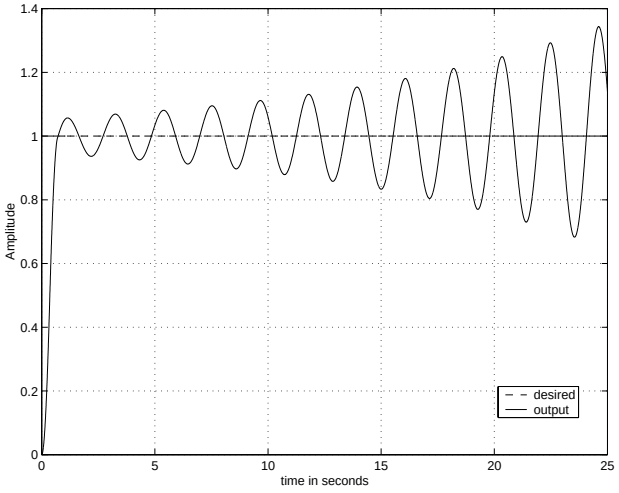

Figure 9: Simulation for $\omega_A = 3\frac{rad}{sec}$ and $\omega_R = 2.9\frac{rad}{sec}$ with $T = \frac{\pi}{3\omega_A}$

6 Conclusion

Transforming the Time Delay feedforward controllers using a structure already known from IMC led to the Time Delay Feedback Controller. Due to the easy and well known structure the design is fast and simple. The stability region was shown for a two delay controller with user selectable time delay, the most general case. Simulations showed the behavior of the controller in the case of under- and overestimation of the natural frequency of the plant.

References

[1] N.C. Singer and W.P. Seering. Preshaping Command Inputs to Reduce System Vibrations. *ASME Journal of Dynamic Systems, Measurement, and Control*, 112:76–82, 1990.

[2] G.H. Tallman and G.H. Smith. Analog study of Dead-Beat Posicast Control. *IRE Transcactions on Automatic Control*, AC-3:14–21, 1958.

[3] T. Singh and S.R. Vadali. Robust Time-Delay Control. *ASME Journal of Dynamic Systems, Measurement, and Control*, 115:303–306, June 1993.

[4] T. Singh and S. R. Vadali. Robust Time-Delay Control of Multimode Systems. *International Journal of Control*, 62(6), pp. 1319-1339.

[5] Vikram Kapila, Anthony Tzes, and Qiguo Yan. Closed-Loop Input Shaping for Flexible Structures Using Time-Ddelay Control. *Journal of Dynamic Systems, Measurements and Control*, 122:454–460, September 2000.

[6] Kai Zuo and David Wang. Closed Loop Shaped-Input Control of a Class of Manipulators with a Single Flexible Link. In *Proceedings of the IEEE International Conference of Robotics and Automation, Nice, France*, volume 1, pages 782–787, May 1992.

[7] Dr.-Ing. Ansgar Trächtler. *Digitale Regelungen.* Robert Bosch GmbH Stuttgart. Lecture Notes, `http://www-mrt.mach.uni-karlsruhe.de/Lehre/DR.html`.

[8] M. Morari and E. Zafiriou. *Robust Process Control.* Prentice Hall, 1989.