TALLINN UNIVERSITY OF TECHNOLOGY

Faculty of Information Technology

Department of Computer Control

Juhani Tali

Implementation of Fractional Order PID Controller Tuning Methods Based on Control System Frequency Response

Bachelor's Thesis

Supervisor(s): Dr. Aleksei Tepljakov

Tallinn 2016

Declaration: I hereby declare that this Bachelor's thesis, my original investigation and achievement, submitted for the Bachelor's degree at Tallinn University of Technology, has not been submitted for any degree or examination.

Deklareerin, et käesolev bakalaureusetöö, mis on minu iseseisva töö tulemus, on esitatud Tallinna Tehnikaülikooli bakalaureusekraadi taotlemiseks ja selle alusel ei ole varem taotletud akadeemilist kraadi.

Juhani Tali

Date:	....
Signature:	....

Contents

Abstract

Implementation of fractional order PID controller tuning methods based on control system frequency response.

Functions with fractional order (FO) diffintegrals do not have general analytical solutions in time domain and precise numerical methods are expensive. Control system tuning methods are iterative and especially in case of FO the function value computations can be time consuming. Time domain based numerical methods for calculating step and impulse response can have a static error while the frequency domain method is claimed to be exact. This thesis analyzes step response in frequency domain using Fourier series of a low frequency square wave (FSM) method for fractional order PID controller tuning. The thesis has found that the method is promising, but sensitive to optimized function and frequency range choice and in case of unstable functions does not always give precise results. Some methods are created to improve the precision.

Kokkuvõte

Murrulist järku PID regulaatori häälestamismeetodite realiseerimine automaatjuhtimissüsteemi sageduskarakteristiku põhjal.

Murrulist järku diffintegraalidega funktsioonidel ei ole üldistatud analüütilist lahendit ning väärtuste arvutamiseks kasutatakse numbrilisi meetodeid, mis on arvutuslikult mahukad. Automaatjuhtimissüsteemide häälestamine on iteratiivne, mis eriti murrulist järku funktisoonide väärtuste arvutamise korral on ajaliselt mahukas. Hüppekaja ja impulsskaja arvutusmeetodid võivad aja domeenil põhinevate numbrilise analüüsi korral sisaldada staatilist viga. Väidetavalt on sagedusvallas samad tulemused täpsed. Antud bakalaureuse töö analüüsib murrulist järku PID kontrolleri häälestamise meetodit, kus kasutatakse sagedusvallas madalsagedusliku nelinurksignaali Fourieri rea meetodil (FSM) arvutatud hüppekaja. Bakalaureuse töös leitakse, et nimetatud meetod on paljulubav, kuid tundlik nii optimeeritava funktsiooni ja sagedusriba valiku suhtes ja mittestabiilsete funktsioonide puhul ei anna alati täpseid tulemusi. Töö käigus tuuakse mõned meetodid mis parandavad täpsust stabiilsete funktsioonide korral.

List of Figures

Chapter 1

Introduction

1.1 Goals of the work

Fractional-order (FO) calculus is an extention to conventional calculus that can be used for modeling some extraordinary dynamic systems, from magnetic levitation control systems to theoretical physics. In practice FO function values are calculated with numerical aproximation. Control system controller tuning for FO systems requires a lot of computations and is time consuming [1]. Fractional-order proportional-integral-derivative (FOPID) controller optimization takes on 2010 year desktop computer about 10 minutes and it can't be done in feasible time on current ARM computers as they are aproximately 20 times slower. This work is to find a better way to optimize FO controllers by using step and impulse response with frequency response method. FO controller optimization requires that methods described in this thesis are sufficiently precise and within a reasonable computational complexity. The goal for thesis is to validate the numerical method suitability for FOPID tuning.

1.2 Objectives and Contributions

Fourier series of a low frequency square wave (FSM) method with time delay [2] was found to be too complex for the task and a more simple method is chosen without time delay [3]. FSM method is claimed to be exact [3]. FOMCON is an existing Matlab toolbox for FO control systems [1] created in TTU. The algorithms for step FSM, impulse FSM and impulse Inverse

Fourier transform (IFTM) methods are implemented in Matlab with $\theta(n)$ complexity as a possible extension to FOMCON toolbox. The result is validated on three systems with fractional order transfer function against FOMCON results. It was found that the FSM method needs dynamic frequency range as one static range cannot be used within reasonable computational complexity. A simple method in this thesis named *W3dB* is implemented in Matlab based on [3]. Then a simple control system optimization is implemented that results in new functions that can not be calculated with the frequency results. The results are analyzed and a couple of methods for impoved step FSM method frequency range are implemented, the last methods being computationally expensive. Controller tuning method with FOPID controller output limits is implemented so that it reuses the frequency model calculations and is Matlab parallel processing compatible. The results are not conclusive and suggestions on additional research are given.

1.3 Thesis Outline

In Chapter 2 the reader is presented with mathematical background.

In Chapter 3 the FSM method is validated with simulations and algorithm improvements are made.

In Chapter 4 the FSM method is used to tune FOPID controllers for closed closed feedback systems.

Chapter 2

Theoretical Part

2.1 Fractional calculus

Fractional order calculus is an generalization to differential and integrate operators. The integro-differential operator is defined as [4]

$$
{}_aD_t^\alpha = \begin{cases} \dfrac{\mathrm{d}^\alpha}{dt^\alpha} & R(\alpha) > 0, \\ 1 & R(\alpha) = 0, \\ \int_a^t (dt)^{-\alpha} & R(\alpha) < 0, \end{cases} \tag{2.1}
$$

where $\alpha \in R$, but α could be a complex number. The fractional-order differentiation is exactly the same with integer-order one, when $\alpha \in \mathcal{N}$.

The well-established fractional-order definitions [5] include the Cauchy definition, the Grünwald-Letnikov definition, the Riemann-Liouville definition and the Caputo definition. These definitions contain some controvery. This thesis and FOMCON toolbox is using Grünwald-Letnikov definition, Caputos definition is given as an example.

Definition (Grünvald-Letnikov)

$$
{}_aD_t^\alpha f(t) = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{[(t-a)/h]} (-1)^j \binom{\alpha}{j} f(t - jh),
$$

Definition (Caputo)

$$
{}_0D_t^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)} \int_0^t \frac{f^{(m)}f(\tau)}{(t-\tau)^{\alpha-m+1}} d\tau,
$$

where $m-1 < \alpha < m, m \in N.$

The fractional-order differentiation has the following properties [5]:

1. The fractional-order differentiation ${}_0D_t^\alpha f(t)$, with respect to t of an analytic function $f(t)$, is also analytical.

2. The fractional-order differentiation es exactly the same with integer-order one, when $\alpha \in \mathcal{N}$. Also ${}_0D_t^0 f(t) = f(t)$.

3. The fractional-order differentiation is linear,

$$
{}_0D_t^\alpha [af(t) + bg(t)] = a \cdot_0 D_t^\alpha f(t) + b \cdot_0 D_t^\alpha g(t).
$$

4. The fractional-order differentiation is commutative and also

$$
{}_0D_t^\alpha [{}_0D_t^\beta f(t)] = {}_0D_t^\beta [{}_0D_t^\alpha f(t)] = {}_0D_t^{\alpha+\beta} f(t).
$$

5. The Laplace transform is defined as

$$
\mathcal{L}[{}_0D_t^\alpha f(t)] = s^\alpha \mathcal{L}[f(t)] - \sum_{k=1}^{n-1} s^k [{}_0D_t^{\alpha-k-1} f(t)]_{t=0}
$$

   and if the derivatives of the function $f(t)$ are all equal to 0 at $t = 0$, then

$$
\mathcal{L}[{}_0D_t^\alpha f(t)] = s^\alpha \mathcal{L}[f(t)].
$$

The properties can be used to substitute fractional order integration by integration and fractional-order differentiation, this is used by [2] to avoid steady-state error by substituting $\frac{1}{s^{0.8}} = \frac{s^{0.2}}{s}$.

2.2 FO transfer function

A fractional-order control system with input $u(t)$ and output $y(t)$ can be described in form [3,6]

$$a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \ldots + a_0 D^{\alpha_0} y(t) = b_m D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \ldots + b_0 D^{\beta_0} u(t).$$

or by fractional order transfer function (FOTF) in the form

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \ldots + b_0 s^0}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \ldots + a_0 s^{\alpha_0}}.$$

2.3 Step and impulse response

Heaviside step function $H(x)$ is defined as [7]

$$H(x) = \begin{cases} 0, & x < 0 \\ (\frac{1}{2}, & x = 0) \\ 1, & x > 0 \end{cases}$$

Impulse response is described by delta function δ, it can be viewed [7] as a derivative of Heaviside step function $\frac{d}{dx}[H(x)] = \delta(d)$.

From a practical viewpoint, Heaviside step function and impulse response are used as an input to test the behavior of unknown systems, be it an electronic component or industrial system.

2.4 Fourier series and Fourier transform

Fourier series of a fuction $f(x)$is given by

$$f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n cos(nx) + \sum_{n=1}^{\infty} b_n sin(nx),$$

where

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)dx,$$

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)cos(nx)dx,$$

$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)sin(nx)dx,$$

and $n = 1, 2, \ldots$.

The Fourier transform is a generalization of the complex Fourier series, [8]

$$F(k) = F_x[f(x)](k) = \int_{-\infty}^{\infty} f(x)e^{-2\pi ikx}dk$$

and the inverse Fourier transform is

$$f(x) = F_k^{-1}[F(k)](x) = \int_{-\infty}^{\infty} F(k)e^{2\pi ikx}dk.$$

2.5 Fourier series of a low frequency square wave (FSM)

Square wave with period T when $t < T/2$ can be viewed as an approximation to step function $H(t)$. The formula for the square wave of -1 to 1 with frequency $w_s = 2\pi/T$ is $2[H(\frac{x}{T/2}) - H(\frac{x}{T/2} - 1)] - 1$ and the Fourier series is [9]

$$x(t) = \frac{4}{\pi}\sum_{k=1(2)}^{\infty}\frac{1}{k}\sin(k\omega_s t),$$

where T is the period of the square wave. If the output passes through transfer function $G(s)$ then the output [3] is

$$\begin{aligned} y_{step}(t) &= \frac{4}{\pi}\sum_{k=1(2)}^{\infty}(\frac{1}{k}Re[G(jk\omega_s)]\sin(k\omega_s t) + \frac{1}{k}Im[G(jk\omega_s)]\cos(k\omega_s t)) \\ &\approx \frac{4}{\pi}\sum_{k=1(2)}^{\infty}\frac{1}{k}Re[G(jk\omega_s)]\sin(k\omega_s t). \end{aligned} \tag{2.2}$$

Since $\frac{d}{dx}[H(x)] = \delta(d)$ we can write

$$
\begin{aligned}
y_{impulse}(t) = \frac{d}{dx} y_{step}(t) \\
= \frac{4}{\pi} \sum_{k=1(2)}^{\infty} (\omega_s \mathrm{Re}[G(jk\omega_s)] \cos(k\omega_s t) - \omega_s \mathrm{Im}[G(jk\omega_s)] \sin(k\omega_s t)) \\
\approx \frac{4}{\pi} \sum_{k=1(2)}^{\infty} \omega_s \mathrm{Re}[G(jk\omega_s)] \cos(k\omega_s t). \quad (2.3)
\end{aligned}
$$

It should be noted that $H(t)$ grows from 0 to 1, but $x_{step}(t)$ is part of a wave from -1 to 1. The practical part is comparing the results of these functions to numerical method impemented in FOMCON toolbox.

2.6 Inverse Fourier transform (IFTM)

Transfer function for impulse response g(t) exist only for $t > 0$, but the range of Fourier transform is from $-\infty$ to ∞. It is possible to extend the function to $t < 0$ by making the extended function $g(t)$ by defining even function $g(t) = g(-t)$ or by defining odd function $g(t) = -g(-t)$. The resulting function [3]in even case is

$$
g_{impulse}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} 2\mathrm{Re}G(j\omega) e^{j\omega t} d\omega = \frac{2}{\pi} \int_{0}^{\infty} \mathrm{Re}G(j\omega) cos(\omega t) d\omega
$$

and in odd case is

$$
g_{impulse}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} 2j\mathrm{Im}G(j\omega) e^{j\omega t} d\omega = -\frac{2}{\pi} \int_{0}^{\infty} \mathrm{Im}G(j\omega) sin(\omega t) d\omega.
$$

Chapter 3

FSM step response method

We are using the following available FO transfer function examples from FOMCON toolbox

$$G_1(s) = \frac{1}{14994s^{1.31} + 6009.5s^{0.97} + 1.69}, \tag{3.1}$$

$$G_2(s) = \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1}, \tag{3.2}$$

$$G_3(s) = \frac{-2s^{0.63} + 4}{2s^{3.501} + 3.8s^{2.42} + 2.6s^{1.798} + 2.5s^{1.31} + 1.5}. \tag{3.3}$$

The goal is to optimize three FOPID controllers for controlling systems described by the transfer functions $G_1(s)$, $G_2(s)$, $G_3(s)$. All three transfer functions have difficult frequency responses. Functions $G_4(s)$, $G_5(s)$ and $G_6(s)$ were found during controller optimization.

$$G_4(s) = \frac{0.8s^{2.7} + 0.5s^{1.4} + s^{0.5}}{0.8s^{2.7} + 0.5s^{1.4} + s + 2s^{0.5} + 1}, \tag{3.4}$$

$$G_5(s) = \frac{0.8s^{2.2682} + 0.5s^{0.9682} + s^{0.0682}}{0.8s^{2.2682} + 1.223e - 06s^{1.0672} + 0.5s^{0.9682} + s^{0.0682} + 1.1466 \cdot 10^{-8}}, \tag{3.5}$$

$$G_6(s) = \frac{0.062073s^{1.6843} + 0.076995s^{0.99112} + 10.702}{0.8s^{3.1911} + 0.5s^{1.8911} + 0.062073s^{1.6843} + 1.077s^{0.99112} + 10.702}. \tag{3.6}$$

Function $G_1(s)$ has a very low significant frequency as shown in Figure 3.1a.

Function $G_2(s)$ has a peak at close to 1 rad/s and with maximum amplitude of 13dB as shown in Figure 3.1b.

Function $G_3(s)$ is complex as shown in Figure 3.1c.

Function $G_4(s)$ has very low response at 0 frequency, has a narrow peak and continues at 0dB or value 1 towards infinity as illustrated in Figure 3.1d.

Function $G_5(s)$ is similar to Heaviside step function $H(t)$ and the frequency response is shown in Figure 3.5.

System with transfer function $G_6(s)$ is unstable, it is found during FOPID optimization.

3.1 Step FSM method with static frequency range

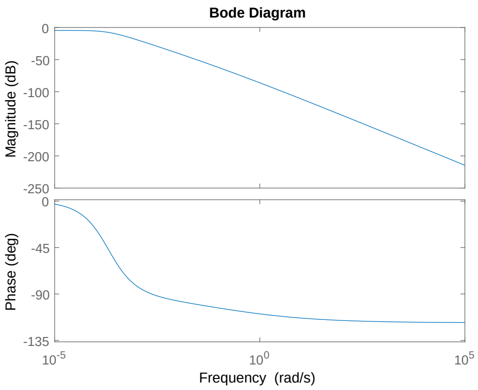
3.1.1 Time step size with FOMCON method

To validate step FSM method (2.2) the results must be compared against some other method because FO calculus has no analytical solution. FOMCON step function does less aproximations than FSM method and is better suitable as our correct function. The simulation results for system described with a transfer function of G_2 (3.2) is in Figure 3.2. In the simulation the step FSM method is using cutoff frequency of 2.789 rad/s and the frequency range is divided to 10001 parts. During simulation the difference between 200 time steps FSM and 4000 time steps FSM was minimal. FOMCON step had more dependency on step size, for 1000 time steps the results with FSM were comparable and with 4000 steps the results were practically equal.

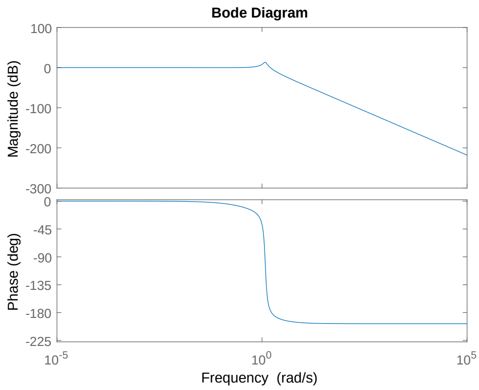
The result is that for some functions FSM method can give excellent results and for results validation FOMCON step method must be used with at least 4000 time steps. FSM method was slower, it took 0.77s to compute compared with FOMCON 0.31s. Calculating 200 steps with FSM was 0.08s. Controller tuning can use less points and for few time steps the FSM method can be more efficient.

3.1.2 Static frequency range

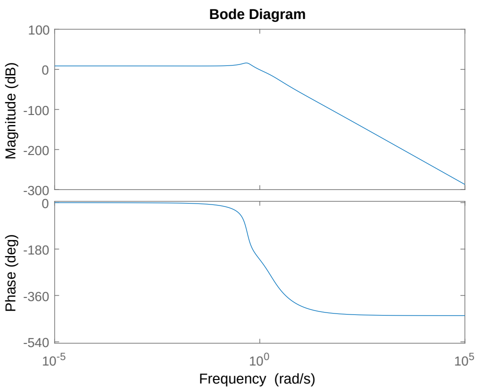
Systems described with transfer functions $G_1(s)$ (3.1) and $G_3(s)$ (3.3) have different significant frequency ranges as can be seen in Figure 3.1a and Figure 3.1c. An experiment was made with two different and good fitting frequency ranges, for $G_1(s)$ with range from 0 to 0.0004 and $G_3(s)$ with frequency range from 0 to 7.4. The results are visualized in Figure 3.3 and Figure



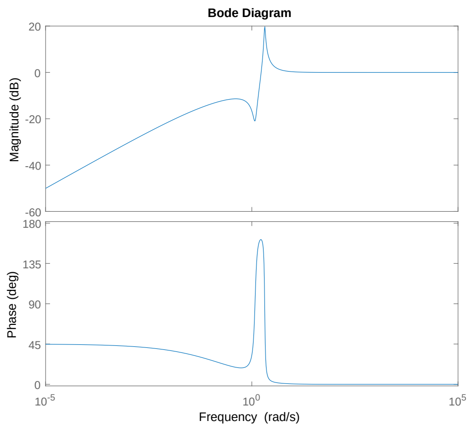
(a) Bode diagram of a system described by the transfer function $G_1(s)$.



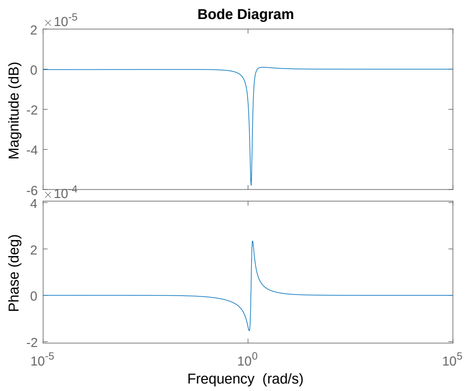
(b) Bode diagram of a system described by the transfer function $G_2(s)$.



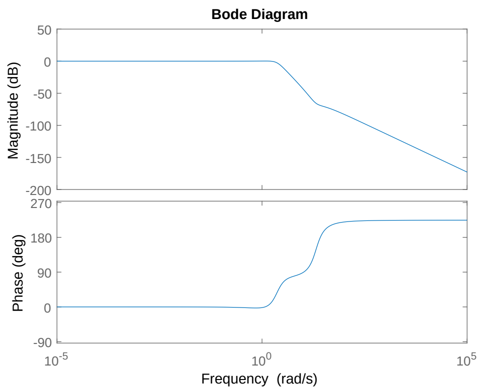
(c) Bode diagram of a system described by the transfer function $G_3(s)$.



(d) Bode diagram of a system described by the transfer function $G_4(s)$.



(e) Bode diagram of a system described by the transfer function $G_5(s)$.



(f) Bode diagram of a system described by the transfer function $G_6(s)$.

Figure 3.1: Bode diagrams

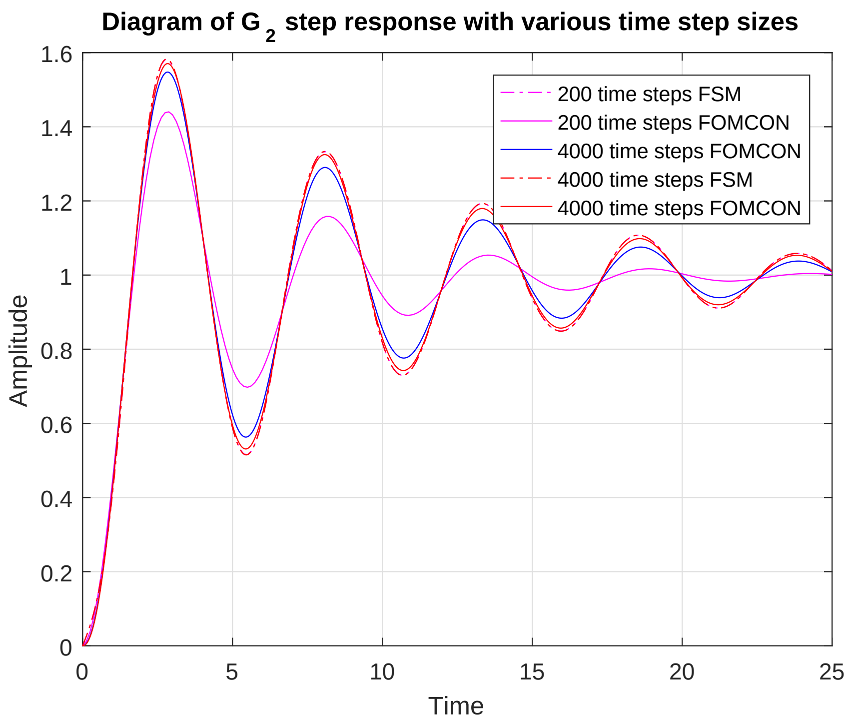


Figure 3.2: Time diagram of a system described by transfer function G_2 with FOMCON and FSM methods.

show that one frequency range is in practice not usable for all FO systems. The wrong frequency range for the system resulted in both cases a result that is similar to constant function zero.

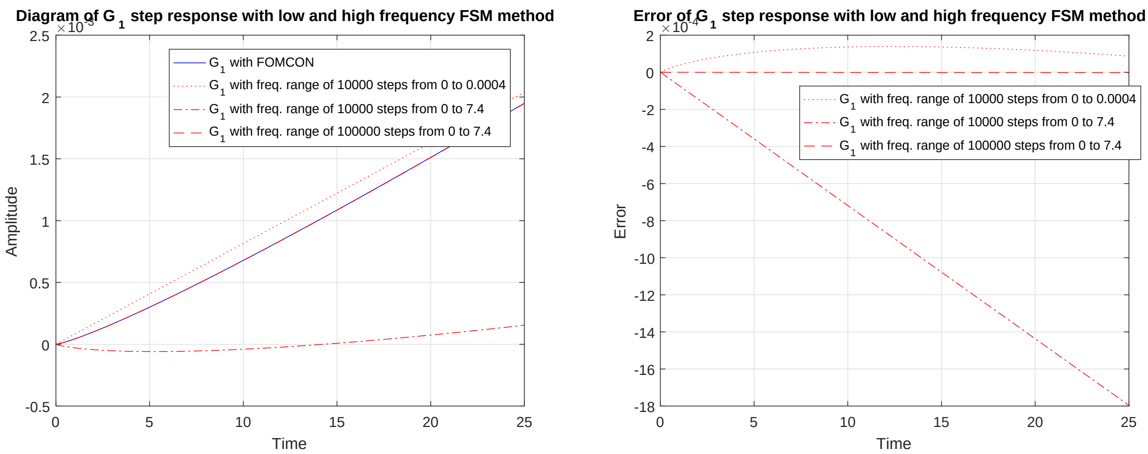


Figure 3.3: Time diagram of a system described by transfer function $G_1(s)$ with different frequency ranges.

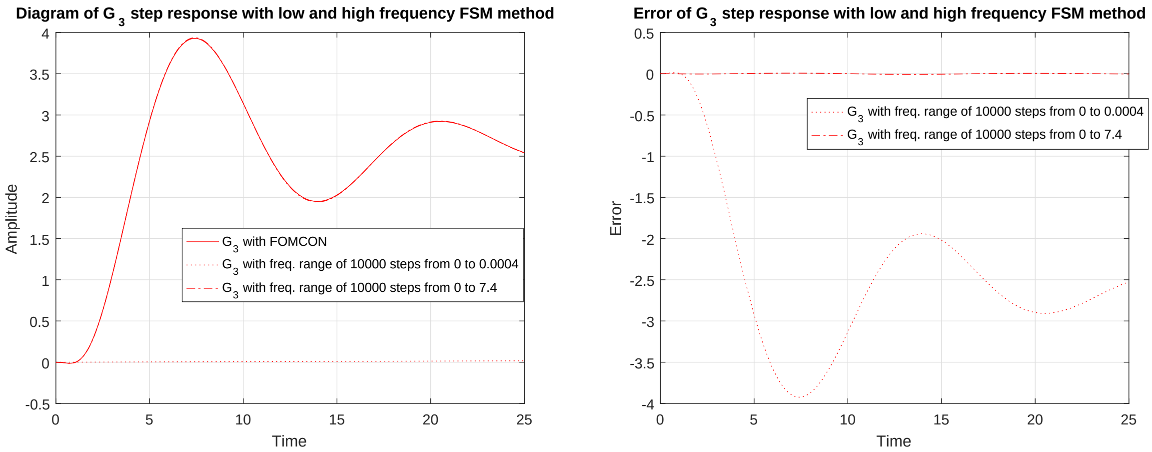


Figure 3.4: Time diagram of a system described by transfer function $G_3(s)$ with different frequency ranges.

3.2 Static frequency range for impulse FSM and impulse IFTM methods

3.3 Dynamic frequency range for step FSM

3.3.1 W3dB method

The 3dB method is described in [3] with the method $w_s = 0.01 w_{3dB}$, where w_{3dB} is the first 3 dB point on G(jw) below G(0), which is assumed to be finite. The frequency range is $[0 : w_s : 100 * w_{3dB}]$ and it contains 10001 points. W3dB method gives good results for simple functions but fails on $G_4(s)$ (3.4) because the value of $G(j*0) = 0$.

3.3.2 Unbalanced search

Unbalanced frequency search method searches for last point x in logarithmic scale from $10^{-5}$to $10^5$where the difference between $G(j*10^5)$ and $G(j*x)$ is larger than $10^{-2} * \max(G())$. The constant 10^5 is the maximum frequency in search range and 10^{-2} as precision is $1/\sqrt{(10001)}$ where 10001 is average step count. This method gives a good frequency range for $G_4(s)$ (3.4) but cannot be used to find frequency range for $G_5(s)$ (3.5). From theoretical aspect it does not take into account that the step FSM method 2.2 summand contains $1/k$ and the value for high frequency is reduced.

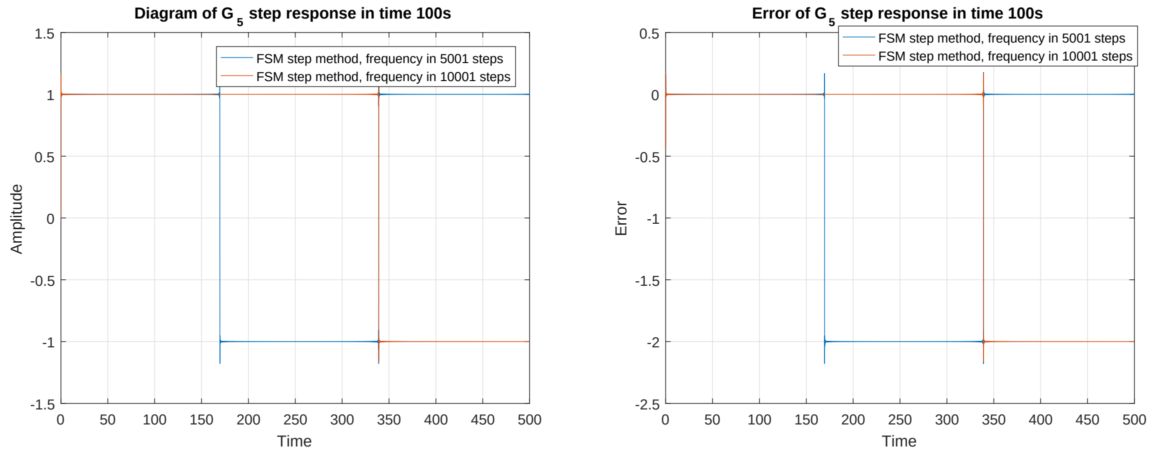


Figure 3.5: Time diagram of a system described by transfer function $G_5(s)$ for long time period.

3.3.3 Balanced search

Unbalanced search method is improved by using the exact summand from 2.2 by replacing $G()$ with $1/k \cdot G() \cdot \sin(k \cdot t)$ where time t=1. Here the weight $1/k$ reduces the high frequency summands and $sin(k \cdot t)$ reduces the low frequency summands. This method can be used to find usable frequency range for system described with transfer function $G_5(s)$ (3.5).

3.4 FSM step method implementation considerations

3.4.1 Effect of time on frequency range

The FSM method (2.2) is based on square wave with frequency $w_s = 2\pi/T$ and described by Heaviside step functions as $2[H(\frac{x}{T/2}) - H(\frac{x}{T/2} - 1)] - 1$. To test the time and frequency dependency function $G_5(s)$ evaluated with FSM step method, this shows the foundation of FSM method in Figure 3.5. FSM method gives comparable result to Heaviside step function in time range where the lowest frequency $w_s < \pi/T$ where T is maximum time, this is first half the square wave period.

3.4.2 Function oscillations

Function oscillations can create problems for setting limits on FOPID optimizations and oscillating functions need more time steps for optimization weight calculations. In experiment $G_5(s)$ oscillations compared to FOMCON method were up to 20% as illustrated in Figure 3.6. FSM

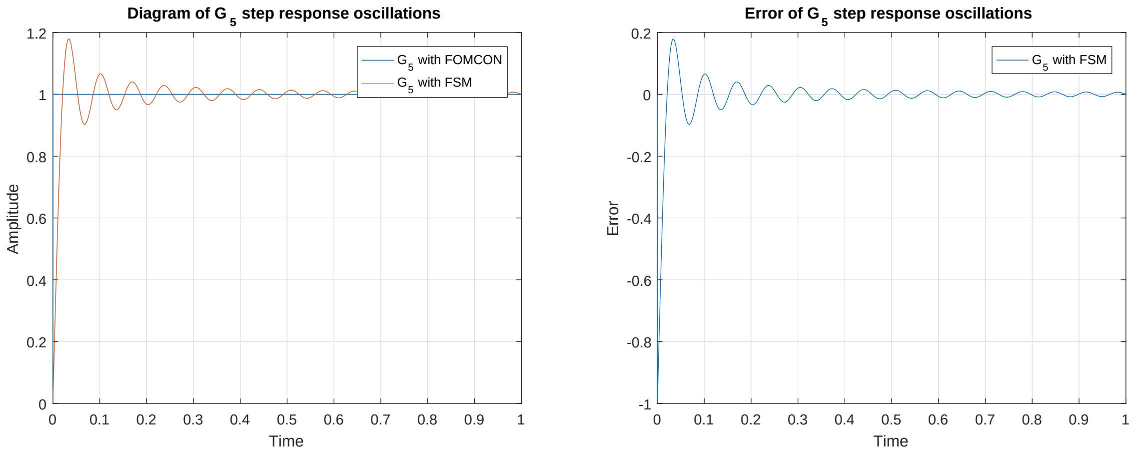


Figure 3.6: Time diagram of a system described by transfer function $G_5(s)$.

method for FOPID optimization is not the best fit if the overshoot must be controlled exactly and the results must be verified with other methods.

3.4.3 Unstable transfer function

Function $G_6(s)$ (3.6) describes a system `feedback(FOPID*G2,1)` and is by FOMCON method `isstable()` unstable, that looks like a very good FOPID for controlling object $G_2(s)$ with FSM step method as shown in Figure: 3.7. The systems step response gains expected result 1 fast and is precise, this is how FSM method based FOPID optimization sees the result. The system does not have visible changes if the frequency range and step amount are significantly increased, the used frequency range $[0 : 3.9409/30001 : 3.9409]$ looks like a good fit. The system response is significantly different when viewed with FOMCON implementation, as illustrated in Figure: 3.8. The system is for the first 15s close to value 4, then jumps to $0.5 \cdot 10^{10}$ and falls to $-3 \cdot 10^{10}$. The real world FOPID controller will work on time model and hence the FOMCON model must be considered the practical and correct result. Theoretical methods for correct answer are not available. Unstable systems are sensitive to initial conditions or as in this case, to calculating methods.

3.5 Non-linear frequency range

The optimal frequency range is important for algorithm efficiency. Step FSM method 2.2 contains periodic function weights with a period of $2 \cdot \pi \cdot t$. The algorithm precision might be

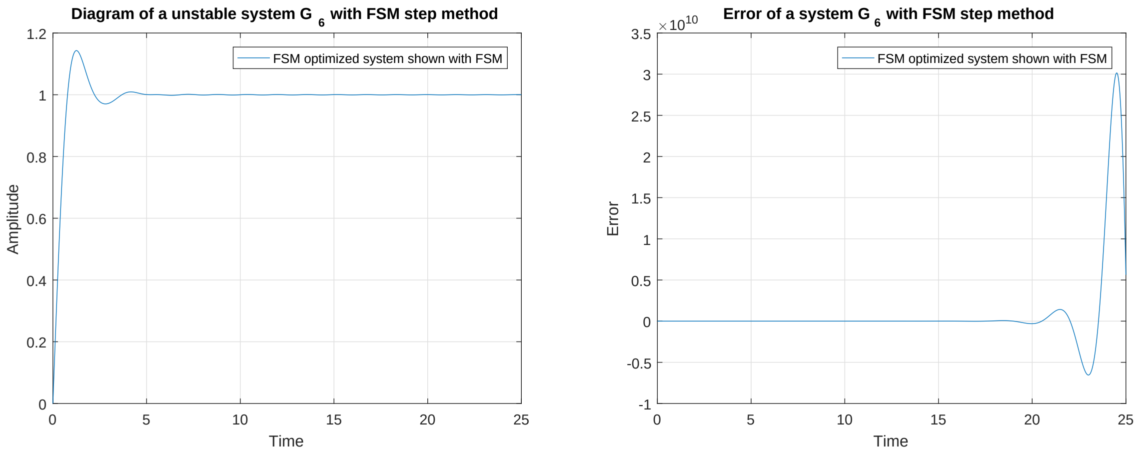


Figure 3.7: Time diagram of a system described by transfer function $G_6(s)$ with FSM step method.

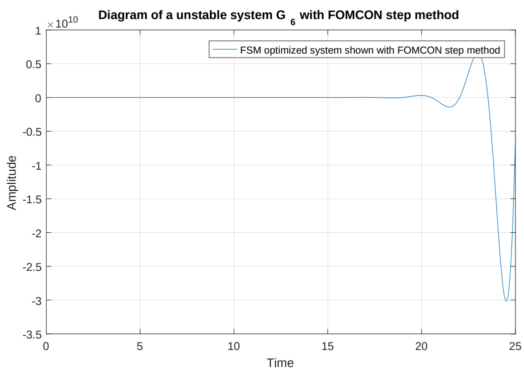


Figure 3.8: Time diagram of a system described by transfer function $G_6(s)$ with FOMCON step method.

improved by considering the summand weight as a periodic function of frequency and time so all the frequencies should be chosen as pairs $(x, x + \pi/t)$. Also for higher frequencies the weight value close to $\sin()$ period of $\pi \cdot t$ is small and frequency range should prefer values close to $n \cdot 1/2 \cdot \pi \cdot t$ where $n \in \mathbb{N}$. For times close to 0 the weight of very low frequency is small and the Heaviside function aproximation oscillations might be reduced by shifting the frequency range towards high frequency while simultaneously keeping low frequencies for time range. The computed frequency values must be reused to avoid algorithmic complexity $o(n^2)$. Non-linear frequency range is not implemented in this thesis.

Chapter 4

FOPID tuning

This chapter is searching for optimal FOPID controller for common closed feedback system

$$C_{cl}(s) = C(s)G(s)/(1 + C(s)G(s)),$$

where a plant modeled with transfer function $G(s)$ is preceded by controller

$$C(s) = K_p + K_i/s^\lambda + K_d \cdot s^\mu.$$

FOPID optimization searches for parameters K_p, K_i, K_d, λ, μ by finding the values with minimal system weight and where the conditions are true. Weight can be calculated with multiple methods, this experiment is using Integral Time-weighted Absolute Error (ITAE). Parameters are searched with iterative Matlab function fmincon(), the default algorithm is interior-point. Constants $[K_p, K_i, \lambda, K_d, \mu]$ are limited by minimum values $[0, 0, 0.01, 0, 0.01]$ and maximum values $[100, 100, 1, 100, 1]$, the limits are chosen equal to FOMCON toolbox default values. The tuning is preliminary, without gain and phase margins.

4.0.1 System with plant $G_1(s)$

System with plant G_1 is slow to respond to changes. The optimization process took 145 seconds and 14 iterations. Found FOPID parametes $[K_p, K_i, \lambda, K_d, \mu]$ are $[6.1195 \cdot 10^{-07}, 100, 1, 8.3657 \cdot 10^{-07}, 0.1197]$. The result is illustrated in Figure 4.1.

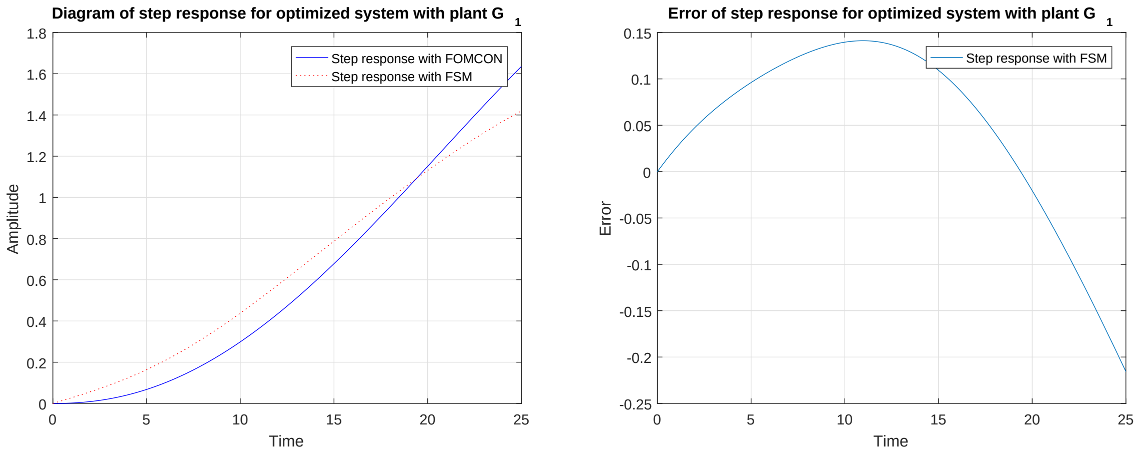


Figure 4.1: Time diagram of optimized system for plant G_1.

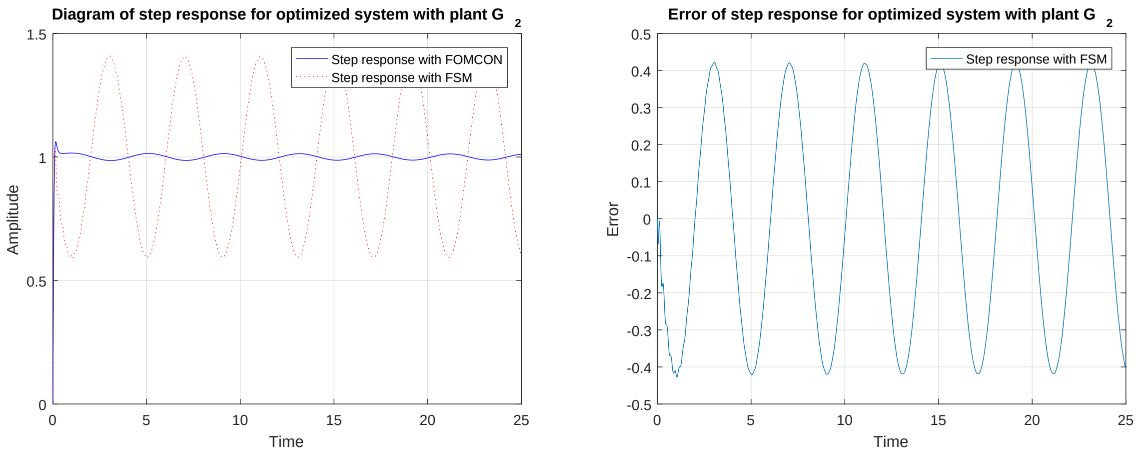


Figure 4.2: Time diagram of optimized system for plant G_2.

4.0.2 System with plant $G_2(s)$

System with plant $G_2(s)$ optimization resulted in FOPID controller parameters $[K_p, K_i, \lambda, K_d, \mu]$ values $[0.2039, 98.1659, 1, 40.3846, 0.9929]$. Optimized system step response is shown in Figure 4.2. With FSM method optimized system looks to have strong oscillations because the automatically found frequency is not sufficiently high. The reason for suboptimal frequency range is a small 5.5dB peak in frequency response. Oscillations created a problem for optimization and solution was acquired after maximum number of 50 steps and 855 seconds. For this system parallel optimization was tested and on computer with two cores the result was obtained in 823 seconds. This system optimization did not converge at 1000 time steps and the current result was acquired using 10000 time steps.

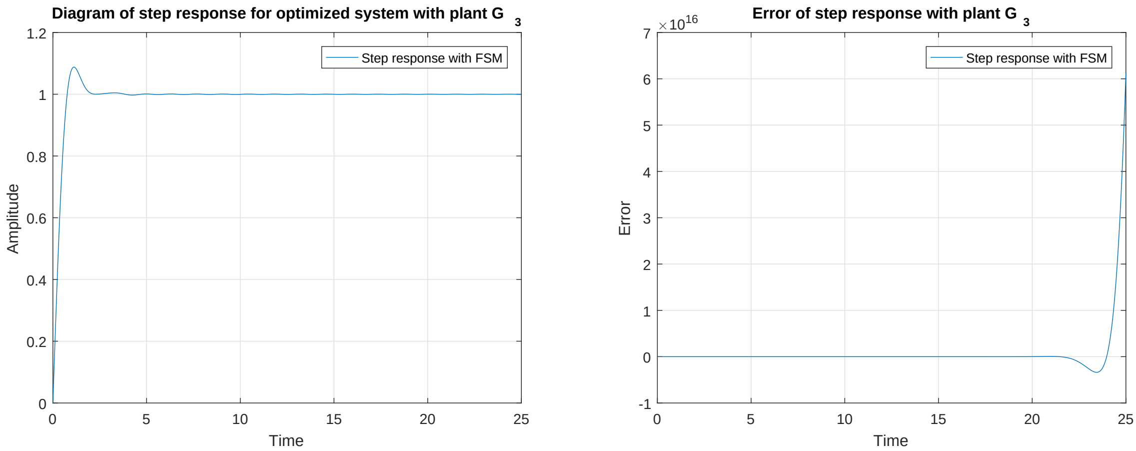


Figure 4.3: Step response diagram with FSM step method for optimized system for plant G_3.

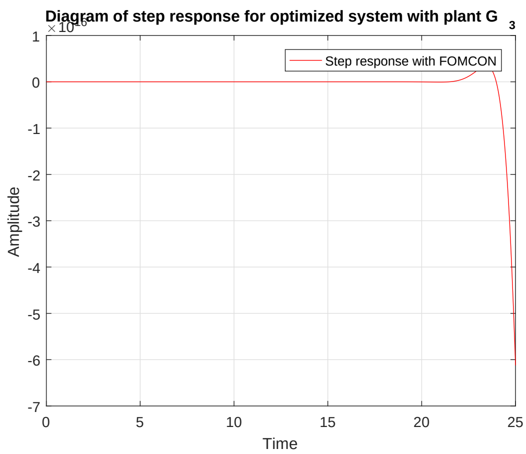


Figure 4.4: Step response diagram with FOMCON step method for optimized system for plant G_3.

4.0.3 System with plant $G_3(s)$

System with plant $G_3(s)$ optimization resulted in FOPID controller parameters $[K_p, K_i, \lambda, K_d, \mu]$ values $[2.5249, 53.9884, 0.9983, 0.7257, 0.7487]$. Optimization was finished in 13 steps and 312 seconds. The resulting system has two roots in unstable region as illustrated in Figure 4.5 and the system is unstable. With FSM method the system looks excellent, ITAE weight for 10000 time points is 0.56 as illustrated in Figure 4.3. With FOMCON method the system is not optimized and unstable with minimum at $-6 * 10^{16}$ as illustrated in Figure 4.4. The optimized FOPID controller is in practice not usable.

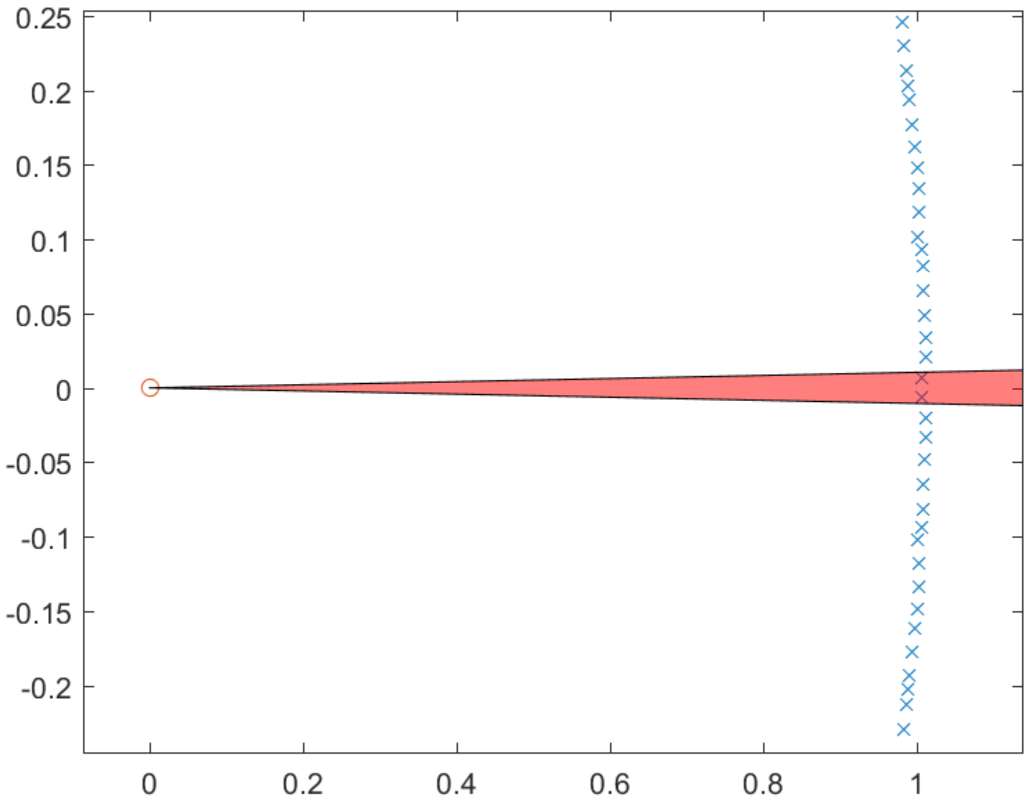


Figure 4.5: Stability of optimized system for plant $G_3(s)$.

Conclusions

This thesis mentions multiple methods and FSM step method is investigated for FOPID controller optimization. The FSM step method is theoretically correct and for some problems is significantly more efficient, but in practice due to irregularities described in this thesis, FSM step method does not always produce reliable results as shown for $G_6(s)$ and for optimized control system for plant described with transfer function $G_3(s)$. The FSM step method has oscillations and precise frequency range must be found. Some methods for finding precise frequency range are produced in this thesis. The conclusion is that FSM step method is sensitive to calculation parameters and functions. FOPID optimization can search through a wide range of functions, the problematic functions $G_4(s)$, $G_5(s)$ and $G_6(s)$ were found during search. The correct result should be considered equal to FOMCON implemented method because the real world, physical controller will control the plant in discrete time and with that method. Methods described in this thesis are promising, but currently do not allow for reliable FOPID tuning. Further investigation is needed for improving the frequency response methods further. Two areas can be considered, nonlinear frequency range and finding the exact cause of not satisfactory results with unstable functions.

Bibliography

[1] A. Tepljakov, E.Petlenkov, and J.Belikov, "A flexible matlab tool for optimal fractional-order pid controller design subject to specifications," in *Control Conference (CCC), 2012 31st Chinese*, July 2012, pp. 4698–4703.

[2] N. Tan, D. P. Atherton, and A. Yüce, "Computing step and impulse responses of closed loop fractional order time delay control systems using frequency response data," *International Journal of Dynamics and Control*, pp. 1–10, 2016.

[3] D. P. Atherton, N. Tan, and A. Yüce, "Methods for computing the time response of fractional-order systems," *IET Control Theory Applications*, vol. 9, no. 6, pp. 817–830, 2015.

[4] Y. Chen, I. Petras, and D. Xue, "Fractional order control - a tutorial," in *2009 American Control Conference*, June 2009, pp. 1397–1411.

[5] D. Xue, Y. Chen, and D. P. Atherton, *Linear feedback control: analysis and design with MATLAB*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2007.

[6] A. Tepljakov, "Fractional-order modeling and control of dynamic systems," Ph.D. dissertation, Tallinn University of Technology, 2015.

[7] R. Bracewell, *The Fourier Transform and Its Applications*, 3rd ed. McGraw Hill, 2000.

[8] E. W. Weisstein. Fourier transform. http://mathworld.wolfram.com/FourierTransform.html 05.04.2016.

[9] E. W. Weisstein. Fourier series–square wave. http://mathworld.wolfram.com/FourierSeriesSquareWave.html 05.04.2016.