COMPUTER AIDED SYNTHESIS AND DESIGN OF PID CONTROLLERS

A Thesis

by

SANDIPAN MITRA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2007

Major Subject: Electrical Engineering

COMPUTER AIDED SYNTHESIS AND DESIGN OF PID CONTROLLERS

A Thesis

by

SANDIPAN MITRA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	S.P. Bhattacharyya
Committee Members,	Aniruddha Datta
	Darbha Swaroop
	Jim Ji
Head of Department,	Costas N. Georghiades

December 2007

Major Subject: Electrical Engineering

ABSTRACT

Computer Aided Synthesis and Design of PID Controllers. (December 2007)

Sandipan Mitra, B.E., National Institute of Technology Karnataka, India

Chair of Advisory Committee: Dr. S.P. Bhattacharyya

This thesis aims to cover some aspects of synthesis and design of Proportional-Integral-Derivative (PID) controllers. The topics include computer aided design of discrete time controllers, data-based design of discrete PID controllers and data-robust design of PID controllers. These topics are of paramount in control systems literature where a lot of stress is laid upon identification of plant and robust design.

The computer aided design of discrete time controllers introduces a Graphical User Interface (GUI) based software. The controllers are: Proportional (P), Proportional-Derivative (PD),Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers. Different performance based design methods with these controllers have been introduced. The user can either explore the performance by interactively choosing controllers one by one from the entire set and visualizing its performance or specify some performance constraints and obtaining the resulting set.

In data-based design, the thesis presents a way of designing PID controllers based on input-output data. Thus, the intermediate step of identification of model from data is removed, saving considerable effort. Moreover, the data required is step response data which is easier to obtain in case of discrete time system than frequency response data. Further, a GUI developed for interactive design is also described.

In data-robust design, the problem of uncertainty in data is explored. The design method developed finds the stabilizing set which can robustly stabilize the plant with uncertainty. It has been put forward as an application to interval linear programming.

The main results of this research include a new way of designing discrete time

PID controllers directly from the data. The simulations further confirm the results. Robust design of PID controllers with data uncertainty has also been established. Additionally, as a part of this research, a GUI based software has been developed which is expected to be very beneficial to the designers in manufacturing, aerospace and petrochemical industries.

PID controllers are widely used in the industry. Any progress in this field is well acknowledged both in the industry and the academia alike. This thesis attempts a small step further in this direction.

To My brother, My inspiration - forever!!

ACKNOWLEDGMENTS

I would like to express my sincere thanks to my advisor, Dr. S.P. Bhattacharyya, for his constant guidance, support, patience and inspiration. Working under his able guidance has made me conversant with the latest trends in the field of control systems and also gave me the opportunity to meet the renowned people in this area. This helped me to gather valuable input for my research.

I would also like to thank Dr. Aniruddha Datta, Dr. Jim Ji, Dr. Darbha Swaroop and Dr. Bryan Rasmussen. Their comments, suggestions and co-operation played a huge part in the successful completion of this work.

I wish to thank Indu Ramamurthi and Bharat Narsimhan for their immense help and interest in my work. I would also wish to thank Subramaniam Ramamoorthy of National Instruments for his crucial and timely feedbacks and his efforts to commercialize the software.

Last but not the least, I would like to thank my family and friends for their life-long co-operation, constant motivation and timely support.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

In today's world, the importance of control systems cannot be undermined. It is not only integral part of the various external machines we use in our day to day life but there exists a large number of control systems exist inside ourselves too. The basic function of a control system is to make certain physical variables of a system behave in a prescribed manner like track some given input despite the presence of uncertainties and disturbances.

The current work concentrates on some aspects of control system design like computer aided design of discrete time low order controllers, data based design of discrete time controllers and data based design of controllers against uncertainty in measurements.

A. Background

By introduction of state-feedback and quadratic optimization theories by Kalman [1], though the controllers designed could meet many performances simultaneously, the controllers essentially become high order controllers. But in industry, more than 90% of the controllers used are essentially low order controllers [2]. This fact has led to a renewed interest [3] in lower order controllers.

Thus recently a lot of stress has been laid upon design of low order controllers. Hara, Shiokata, and Iwasaki [4] developed the generalization of KYP lemma designed to be valid over the prescribed frequency ranges was developed to deal with fixed order controller synthesis. Henrion et. al [5] proposed a relaxation approach to the

---

The journal model is *IEEE Transactions on Automatic Control.*

design of fixed order controllers. Haddad, Hwang, and Bernstein [6] have discussed design of fixed order controllers in the discrete-time case. Iwasaki and Skelton [7] have studied design of H-infinity controllers of fixed order. Dorato [8] has put forward the use of quantifier elimination (QE) techniques dealing with the fixed order controller design problem. Gryazina and Polyak [9] have revisited Neimarks D-Decomposition technique [10],[11] to design fixed order controllers. There have been a number of papers addressing the fixed order controller design problem using LMI techniques [12].

Proportional-Integral-Derivative (PID) controllers constitute the major portion of the low order controller market. This is mainly because of its simple structure and its functionality which offers treatment of both transient and steady-state responses.Traditionally PID controller design has been a tuning based approach (Ziegler Nichols tuning rules [13]) and the methods by which optimum controller parameters are chosen are ad hoc in nature. Other methods of controller design generally involve handling a single objective function. This function may not include all the possible specifications and the whole calculation has to be repeated again if the controller obtained after solving the objective function is not suitable for some reason .

Recently Bhattacharyya et al. [14] have developed a novel way of calculating the entire state of stabilizing controllers. This was also extended to the discrete time systems [15]. Very recently, finding the entire stabilizing region for continuous time system is extended to model free case [16] where the design was based on frequency response of the plant and no model of the plant is needed. The essential advantage of finding the entire stabilizing set lies in the fact that with this, now various performance constrains can be imposed on this basic set and subsets achieving the desired performance objectives can be achieved.

B. Objective

With the given background above, the objective behind the current work can be better understood. It consists of three major parts.

- Graphical User Interface (GUI) for discerte time control systems.
- Data based design of discrete time PID Controllers.
- Data-robust design of PID Controllers.

GUI for discrete time control systems: Graphical user interface has been developed for Proportional, Proportional-Derivative (PD), Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers in MATLAB. When the plant parameters are given, these GUIs give the entire stabilizing values for the respective controller structure. Additionally, when a particular set of controller parameters are chosen, various performance specifications like gain margin, phase margin, overshoot, rise time etc. for the closed loop system is displayed. When a few such performance constraints are given, the GUI also displays the subset of controller parameters obtaining the desired objective. Some additional features of simultaneous stabilization of a set of plants is also developed.

Data based design of discrete time PID controllers: This part consists of model free design of discrete time PID controllers. When the model of the system is not available but instead the step response data of the system is available, it is still possible to design the PID controller. For this case also a GUI has been developed which can obtain desired performance on the entire stabilizing set obtained. This work has also been published in [17].

Data-robust design of PID controllers: There may be some uncertainty in the measurement of data. This may affect the design of PID Controllers. In this part, a way of obtaining the stabilizing set of PID controllers which will robustly stabilize the plant in spite of the uncertainty in measurement has been discussed.

C. Organization of the Thesis

In the following chapters various topics are covered. In Chapter II, the theory of discrete time controller design is briefly discussed followed by a section on Computer Aided Design of lower order controllers. In Chapter III, the MATLAB based GUI for various controllers (Proportional (P), Proportional-Derivative (PD),Proportional-Integral(PI) and Proportional-Integral-Derivative (PID)) along with their algorithms and various examples is illustrated. Chapter IV consists of data-based design of discrete time PID Controller along with GUI based examples. Data-robust design of PID controllers constitutes Chapter V along with a descriptive example. In the final concluding Chapter VI, the research work is concluded and future work is discussed.

CHAPTER II

MATHEMATICAL BACKGROUND

In this chapter, the mathematical background necessary for better understanding of the concepts used in the following chapters has been discussed. In particular, two main topics are illustrated. The first section which is about root counting, phase unwrapping and stability for discrete time systems, is based on the work by Keel and Bhattacharyya [18]. This has been breifly summarized because it is fundamental for development of algorithms for computing stabilizing sets for various low order controllers. The other topic is about Computer-Aided design of controllers. It gives a mathematical description of Computer-Aided design to find set of controllers achieving various design constraints.

A. Root Counting, Phase Unwrapping and Stability for Discrete Time Systems

1. Tchebychev Representation of Polynomials and Rational Functions

As per [18], let a real polynomial be described as $P(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0$ where $a_i, i = 0, 1, \cdots, n$ are real numbers. The unit circle image of the real polynomial $P(z)$ is given by:

$$\left\{ P(z) \ : \ z = e^{j\theta}, \quad 0 \le \theta \le 2\pi \right\}. \tag{2.1}$$

Since a_i are real, $P\left(e^{j\theta}\right)$ and $P\left(e^{-j\theta}\right)$ become conjugate complex numbers, and so it sufficient to evaluate the image of upper half of the unit circle:

$$\left\{ P(z) \ : \ z = e^{j\theta}, \quad 0 \le \theta \le \pi \right\}. \tag{2.2}$$

Now it is known that

$$z^k \Big|_{z=e^{j\theta}} = \cos k\theta + j \sin k\theta, \tag{2.3}$$

such that $P\left(e^{j\theta}\right)$ can be written as

$$\begin{aligned} P\left(e^{j\theta}\right) &= \underbrace{(a_n \cos n\theta + \cdots + a_1 \cos\theta + a_0)}_{\bar{R}(\theta)} + j \underbrace{(a_n \sin n\theta + \cdots + a_1 \sin\theta)}_{\bar{I}(\theta)} \\ &= \bar{R}(\theta) + j\bar{I}(\theta). \end{aligned} \tag{2.4}$$

Using Tchebychev representation ([19], p. 71) $u = -\cos\theta$ such that variable z can be represented as

$$z = e^{j\theta} = \cos\theta + j\sin\theta = -u + j\sqrt{1-u^2} \tag{2.5}$$

As θ runs from $0 \rightarrow \pi$, u runs from -1 to $+1$. Defining

$$\begin{aligned} \cos(k\theta) : &= c_k(u) \\ \frac{\sin(k\theta)}{\sin(\theta)} &= \frac{sin(k\theta)}{\sqrt{(1-u^2)}} := s_k(u) \end{aligned} \tag{2.6}$$

where $c_k(u)$ and $s_k(u)$ are real polynomials in u. These polynomials can be recursively calculated using the following equations.

$$\begin{aligned} s_k(u) &= -\frac{1}{k}\frac{dc_k(u)}{du}, \\ c_{k+1} &= -uc_k(u) - (1-u^2)s_k(u) \\ k &= 1, 2, \cdots \end{aligned} \tag{2.7}$$

From Equation (2.7) $c_k(u)$ and $s_k(u)$ for all k can be obtained. The first few of these are shown in the Table I Thus $P(z)$ can now be represented as

$$P(z) = P\left(e^{j\theta}\right)\Big|_{u=-\cos\theta} = R(u) + j\sqrt{1-u^2}T(u) =: P_c(u). \tag{2.8}$$

where $P_c(u)$ is the Tchebycev representation of $P(z)$ and

$$R(u) = a_n c_n(u) + a_{n-1}c_{n-1}(u) + \cdots + a_1 c_1(u) + a_0 \tag{2.9}$$

Table I. Tchebychev Representation of $c_k(u)$ and $s_k(u)$ for $k = 1$ to 5

k	$c_k(u)$	$s_k(u)$
1	$-u$	1
2	$2u^2 - 1$	$-2u$
3	$-4u^3 + 3u$	$4u^2 - 1$
4	$8u^4 - 8u^2 + 1$	$-8u^3 + 4u$
5	$-16u^5 + 20u^3 - 5u$	$16u^4 - 12u^2 + 1$

$$T(u) = a_n s_n(u) + a_{n-1}s_{n-1}(u) + \cdots + a_1 s_1(u). \tag{2.10}$$

As z traverses the upper half of the unit circle, the complex plane image of $P(z)$ can be obtained by evaluating $P_c(u)$ as u runs from -1 to $+1$. It is also assumed that $P(z)$ has no roots on the unit circle. If there are unit circle roots, they can be displaced out of the circle by replacing z with $z/(1+\epsilon)$.

Now consider a rational function $Q(z)$ which is the ratio of two real polynomials $P_1(z)$ and $P_2(z)$ with no roots on unit circle. Let

$$P_i(z)|_{z=-u+j\sqrt{1-u^2}} = R_i(u) + j\sqrt{1-u^2}T_i(u), \qquad \text{for} \quad i = 1, 2. \tag{2.11}$$

Then the image of $Q(z)$ on unit circle can be computed and its Tchebychev representation $Q_c(u)$ can be written as

$$\begin{aligned} Q(z)|_{z=-u+j\sqrt{1-u^2}} &= \left.\frac{P_1(z)}{P_2(z)}\right|_{z=-u+j\sqrt{1-u^2}} \\ &= \left.\frac{P_1(z)P_2\left(z^{-1}\right)}{P_2(z)P_2\left(z^{-1}\right)}\right|_{z=-u+j\sqrt{1-u^2}} \end{aligned} \tag{2.12}$$

$$= \frac{\left(R_1(u) + j\sqrt{1-u^2}T_1(u)\right)\left(R_2(u) - j\sqrt{1-u^2}T_2(u)\right)}{\left(R_2(u) + j\sqrt{1-u^2}T_2(u)\right)\left(R_2(u) - j\sqrt{1-u^2}T_2(u)\right)}$$

$$= \frac{\overbrace{\left(R_1(u)R_2(u) + \left(1-u^2\right)T_1(u)T_2(u)\right)}^{R(u)} + j\sqrt{1-u^2}\,\overbrace{\left(T_1(u)R_2(u) - R_1(u)T_2(u)\right)}^{T(u)}}{R_2^2(u) + \left(1-u^2\right)T_2^2(u)}$$

$$= : Q_c(u). \tag{2.13}$$

2. Interlacing Condition for Schur Stability

A polynomial is Schur stable if all the zeros of the polynomial are inside the unit circle. Below is a theorem which relates the Schur stability with the interlacing of the zeros of the real and imaginary parts of the Tchebychev representation of the polynomial $P(z)$. Let $P(z)$ be a polynomial of degree n. As earlier,

$$\begin{aligned} P\left(e^{j\theta}\right) &= \bar{R}(\theta) + j\bar{T}(\theta), \qquad \text{where } \ u = -\cos\theta \\ &= R(u) + j\sqrt{1-u^2}T(u) \end{aligned} \tag{2.14}$$

where $R(u)$ and $T(u)$ are real polynomials of degree n and $n-1$, respectively. Now we state the theorem as given in [18]

Theorem II.1 $P(z)$ is Schur stable if and only if

(a) $R(u)$ has n real distinct zeros r_i, $i = 1, 2, \cdots, n$ in $(-1, 1)$.

(b) $T(u)$ has $n-1$ real distinct zeros t_j, $j = 1, 2, \cdots, n-1$ in $(-1, 1)$.

(c) The zeros r_i and t_j interlace:

$$-1 < r_1 < t_1 < r_2 < t_2 < \cdots < t_{n-1} < r_n < +1. \tag{2.15}$$

Proof II.1 Let

$$t_j = -\cos\alpha_j, \qquad \alpha_j \in (0, \pi), \qquad j = 1, 2, \cdots, n-1 \tag{2.16}$$

or

$$\begin{aligned}\alpha_j &= \cos^{-1}(-t_j), \qquad j = 1, 2, \cdots, n-1\\ \alpha_0 &= 0,\\ \alpha_n &= \pi\end{aligned}$$

and let

$$\beta_i = \cos^{-1}(-r_i), \qquad i = 1, 2, \cdots, n, \qquad \beta_i \in (0, \pi) \tag{2.17}$$

Then $(\alpha_0, \alpha_1, \cdots, \alpha_n)$ are the $n+1$ zeros of $\bar{I}(\theta) = 0$ and $(\beta_1, \beta_2, \cdots, \beta_{n-1})$ are the n zeros of $\bar{R}(\theta) = 0$. The condition (c) means that α_i and β_j satisfy:

$$0 = \alpha_0 < \beta_1 < \alpha_1 < \beta_2 < \cdots < \beta_{n-1} < \alpha_n = \pi. \tag{2.18}$$

The condition in Equation (2.18) means that the plot of $P\left(e^{j\theta}\right)$ for $\theta \in [0, \pi]$ turns counterclockwise through exactly $2n$ quadrants. Therefore,

$$\Delta_0^{\pi}[\phi_P(\theta)] = 2n \cdot \frac{\pi}{2} = n\pi \tag{2.19}$$

and this condition in Equation (2.19) is equivalent to $P(z)$ having n zeros inside the unit circle.

To illustrate, consider the following example.

Example II.1 Consider a polynomial

$$P(z) = z^6 + 0.5z^5 + 0.3z^4 + 0.2z^3 + 0.3z^2 + 0.5z + 0.7$$

From this, $R(u)$ and $T(u)$ can be evaluated as

$$\begin{aligned}R(u) &= 32u^6 - 8u^5 - 45.6u^4 + 9u^3 + 16.2u^2 - 2.25u - 0.3\\ T(u) &= -32u^5 + 8u^4 + 29.6u^3 - 5u^2 - 5.4u + 0.75.\end{aligned}$$

$R(u)$ is of degree 6 and $T(u)$ is of degree 5.So condition a) and b) of theorem are met. Further roots of $R(u)$ are $[-0.8936,\ -0.7331,\ -0.0849,\ 0.2252,\ 0.7942,\ 0.9422]$ and roots of $T(u)$ are $[-0.7758,\ -0.5000,\ 0.1358,\ 0.5000,\ 0.8900]$ such that $-1 < -0.8936 < -0.7758 < -0.7331 < -0.5 < -0.0849 < 0.1358 < 0.2252 < 0.5 < 0.7942 < 0.89 < 0.9422 < 1$. This satisfies c) of the theorem that the roots interlace. Hence the given polynomial is stable. This can be cross checked by finding the absolute values of the roots of the polynomial directly. They are

$$[0.9624,\ 0.9624,\ 0.9431,\ 0.9431,\ 0.9218,\ 0.9218]$$

which are less than 1.

3. Root Counting Formulas in Terms of Tchebychev Representations

a. Phase Unwrapping and Root Distribution

To understand the root distribution of a polynomial as the phase changes, a few notations and lemmas are stated. Let the phase of $P(z)$ evaluated at $z = e^{j\theta}$ is denoted as $\phi_P(\theta) := \mathrm{Arg}\left[\mathrm{P}\left(\mathrm{e}^{\mathrm{j}\theta}\right)\right]$.Also, $\Delta_{\theta_1}^{\theta_2}[\phi_P(\theta)]$ denotes the net change also called *unwrapped phase* of $P(e^{j\theta})$ as θ increases from θ_1 to θ_2. In terms of Tchebychev representations, $\phi_{P_c}(u) := \mathrm{Arg}[P_c(u)]$ denote the phase of $P_c(u)$ and $\Delta_{u_1}^{u_2}[\phi_{P_c}(u)]$ denote the net change in or unwrapped phase of $P_c(u)$ as u increases from u_1 to u_2.

Lemma II.1 Let the real polynomial $P(z)$ have i roots in the interior of the unit circle, and no roots on the unit circle. Then

$$\Delta_0^{\pi}[\phi_P(\theta)] = \pi i = \Delta_{-1}^{+1}[\phi_{P_c}(u)] \tag{2.20}$$

Using similar notations for rational function $Q(z)$ Lemma II.1 can be extended to another lemma

Lemma II.2 Let $Q(z) = \frac{P_1(z)}{P_2(z)}$ where the real polynomials $P_1(z)$ and $P_2(z)$ have i_1 and i_2 roots, respectively in the interior of the unit circle and no roots on the unit circle. Then

$$\Delta_0^{\pi}[\phi_Q(\theta)] = \pi\,(i_1 - i_2) = \Delta_{-1}^{+1}[\phi_{Q_c}(u)] \tag{2.21}$$

Proofs of the above lemmas are given in [18]

b. Root Counting and Phase Unwrapping

The concept of interlacing conditions for Schur stability and phase unwrapping can be combined to state a theorem as given in [18]. Let

$$\mathrm{Sgn}[x] = \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$$

The theorem is as follows:

Theorem II.2 Let $P(z)$ be a real polynomial with no roots on the unit circle and let

$$P_c(u) = R(u) + j\sqrt{1-u^2}T(u)$$

be its Tchebyshev representation. Let $t_1, \cdots, t_k$ denote the real distinct zeros of $T(u)$ of odd multiplicity, for $u \in (-1, 1)$, ordered as follows:

$$-1 < t_1 < t_2 < \cdots < t_k < +1$$

and suppose that $T(u)$ has p zeros at $u = -1$. Let $T^{(p)}(-1)$ denote the p^{th} derivative of $T(u)$ evaluated at $u = -1$. Then the number of roots i of $P(z)$ in the interior of the unit circle is given by

$$i = \frac{1}{2}\mathrm{Sgn}\left[T^{(p)}(-1)\right]\left(\mathrm{Sgn}\left[R(-1)\right] + 2\sum_{j=1}^{k}(-1)^{j}\mathrm{Sgn}\left[R\left(t_j\right)\right] + (-1)^{k+1}\mathrm{Sgn}\left[R(+1)\right]\right). \tag{2.22}$$

Proof II.2

$$P(e^{j\theta}) = \bar{R}(\theta) + j\bar{I}(\theta)$$

and define $\theta_i,\ \ i = 1, 2, \cdots, k$ through

$$t_i = -\cos\theta_i, \qquad \text{for}\ \ \theta_i \in [0, \pi].$$

Let $\theta_0 := 0$, $t_0 := -1$ and $\theta_{k+1} := \pi$, and note that the θ_i, $i = 0, 1, 2, \cdots, k+1$ are zeros of $\bar{I}(\theta)$. The proof depends on the following elementary and easily verified facts which are first stated below. The first of these is just the restatement of Lemma II.1:

(a) $\quad \Delta_0^{\pi}[\phi(\theta)] = \pi i$

(b) $\quad \Delta_0^{\pi}[\phi(\theta)] = \Delta_0^{\theta_1}[\phi(\theta)] + \Delta_{\theta_1}^{\theta_2}[\phi(\theta)] + \cdots + \Delta_{\theta_k}^{\pi}[\phi(\theta)]$

(c) $\quad \Delta_{\theta_i}^{\theta_{i+1}}[\phi(\theta)] = \frac{\pi}{2}\mathrm{Sgn}\left[\bar{I}\left(\theta_i^{+}\right)\right]\left(\mathrm{Sgn}\left[\bar{R}\left(\theta_i\right)\right) - \mathrm{Sgn}\left[\bar{R}\left(\theta_{i+1}\right)\right]\right), i = 0, 1, 2, \cdots, k$

(d) $\quad \mathrm{Sgn}\left[\bar{I}\left(\theta_i^{+}\right)\right] = -\mathrm{Sgn}\left[\bar{I}\left(\theta_{i+1}^{+}\right)\right], \quad i = 0, 1, 2, \cdots, k$

(e) $\quad \mathrm{Sgn}\left[\bar{I}(0^{+})\right] = \mathrm{Sgn}\left[T^{(p)}(-1)\right]$

(f) $\quad \mathrm{Sgn}\left[\bar{R}\left(\theta_i\right)\right] = \mathrm{Sgn}\left[R\left(t_i\right)\right], \quad i = 0, 1, 2, \cdots, k.$

Using (a) - (f),we have

$$\begin{aligned}
\pi i &= \Delta_0^{\pi}[\phi(\theta)] = \Delta_0^{\theta_1}[\phi(\theta)] + \cdots + \Delta_{\theta_k}^{\pi}[\phi(\theta)], \qquad \text{by (a) and (b)} \\
&= \frac{\pi}{2}\left(\mathrm{Sgn}\left[\bar{I}(0^{+})\right]\left(\mathrm{Sgn}\left[\bar{R}(0)\right] - \mathrm{Sgn}\left[\bar{R}\left(\theta_1\right)\right]\right) + \cdots\right. \\
&\qquad \left.\cdots + \mathrm{Sgn}\left[\bar{I}\left(\theta_k^{+}\right)\right]\left(\mathrm{Sgn}\left[\bar{R}\left(\theta_k\right)\right] - \mathrm{Sgn}\left[\bar{R}(\pi)\right]\right)\right), \qquad \text{by (c)} \\
&= \frac{\pi}{2}\left(\mathrm{Sgn}\left[\bar{I}\left(0^{+}\right)\right]\left(\left(\mathrm{Sgn}\left[\bar{R}(0)\right] - \mathrm{Sgn}\left[\bar{R}\left(\theta_1\right)\right]\right) - \left(\mathrm{Sgn}\left[\bar{R}\left(\theta_1\right)\right] - \mathrm{Sgn}\left[\bar{R}\left(\theta_2\right)\right]\right)\right.\right.
\end{aligned}$$

$$
\begin{aligned}
& +\cdots+(-1)^k\left(\operatorname{Sgn}\left[\bar{R}\left(\theta_k\right)\right]-\operatorname{Sgn}\left[\bar{R}(\pi)\right]\right)\Big)\Big), \qquad \text{by (d)} \\
= \;& \frac{\pi}{2}\operatorname{Sgn}[T^{(p)}(-1)]\left(\operatorname{Sgn}\left[\bar{R}(0)\right]-2\operatorname{Sgn}\left[\bar{R}\left(\theta_1\right)\right]+2\operatorname{Sgn}\left[\bar{R}\left(\theta_2\right)\right]+\right. \\
& \left.\cdots+(-1)^k\operatorname{Sgn}\left[\bar{R}\left(\theta_k\right)\right]+(-1)^{k+1}\operatorname{Sgn}\left[\bar{R}(\pi)\right]\right), \qquad \text{by (e)} \\
= \;& \frac{\pi}{2}\operatorname{Sgn}[T^{(p)}(-1)]\left(\operatorname{Sgn}[R(-1)]-2\operatorname{Sgn}\left[R\left(t_1\right)\right]+2\operatorname{sign}\left[R\left(t_2\right)\right]+\right. \qquad (2.23) \\
& \left.\cdots+(-1)^k 2\operatorname{Sgn}\left[R\left(t_k\right)\right]+(-1)^{k+1}\operatorname{Sgn}[R(+1)]\right), \qquad \text{by (f)}
\end{aligned}
$$

from which the result follows.

This theorem can be better understood with the following example.

Example II.2 Consider the polynomial

$$P(z)=z^6+4.7z^5+8.76z^4+8.483z^3+4.8163z^2+1.5528z+0.2164$$

From this, $R(u)$ and $T(u)$ can be evaluated as

$$
\begin{aligned}
R(u) &= 32u^6-75.2u^5+22.08u^4+60.068u^3-42.4474u^2+0.3962u+3.1601 \\
T(u) &= -32u^5+75.2u^4-38.08u^3-22.4680u^2+19.4074u-2.2302.
\end{aligned}
$$

The real roots of odd multiplicity of $T(u)$ and lying in $(-1,1)$ are:

$$-0.6176, \qquad 0.1427, \qquad 0.6886$$

By applying the formula of Theorem II.2, we have:

$$
\begin{aligned}
i &= \frac{1}{2}\operatorname{Sgn}\left[T^p(-1)\right]\cdot(\operatorname{Sgn}[R(-1)]-2\operatorname{Sgn}[R(-0.6176)] \\
&\quad +2\operatorname{Sgn}[R(0.1427)]-2\operatorname{Sgn}[R(0.6886)]+\operatorname{Sgn}[R(+1)]) \\
&= \frac{1}{2}(+1)\left[+1-2\cdot(-1)+2\cdot(+1)-2\cdot(-1)+1\right]=4.
\end{aligned}
$$

Therefore, it can be that the polynomial $P(z)$ has 4 roots in the interior of the unit circle. This is verified by determining the roots of $P(z)$ and these are

$$-1.5000 \pm j0.3000, \qquad -0.3500 \pm j0.4976, \qquad -0.5131, \qquad -0.4870\ .$$

The above results are very important for developing linear inequalities for calculating stabilizing sets for controllers as can be seen in following chapters.

B. Computer-Aided Design of Controllers

To continue with mathematical background required for the thesis, it is important to understand the approach undertaken for performance evaluation of the controllers. The method discussed below can hold true for any design problem. However in this thesis, it is used in the context of PID Controllers.

In general any design problem can be formulated as follows:
Let there be a set of n adjustable variables in a system which are labeled as $x_1,\ x_2, \cdots,\ x_n$ and let

$$\bar{x} = [x_1, x_2, \cdots, x_n] \in R^n$$

Consider m performance criteria $f_1,\ f_2, \cdots,\ f_m$ are functions of these adjustable variables such that

$$\begin{aligned} f_1(x_1, x_2, \cdots, x_n) &\in R \\ f_2(x_1, x_2, \cdots, x_n) &\in R \\ &\vdots \\ f_1(x_1, x_2, \cdots, x_n) &\in R \end{aligned} \tag{2.24}$$

Now consider that there are performance requirements like $f_1 > F_1$ or $f_1 < F_1$ where F_1 is some desired value. This in turn gives a solution set $S_1 \in R^n$ which is a set of all values of x which satisfies the above criteria. In general

$$S_i = \{\bar{x} | f_i(\bar{x}) \geq F_i\} \,\text{where} i = 1, 2, \cdots, k. \tag{2.25}$$

In case of a real world control design problem there will be a system which will have multiple such specifications and the objective is to find a set of these adjustable variables which make the closed loop system stable and also simultaneously satisfy all performance requirements. In mathematical terms, let

$$S_0 := \{\bar{x} | \text{Closed loop system is stable}\} \tag{2.26}$$

and S_i as described in (2.25), Then our objective is to find a set S such that

$$S = \cap_{i=1}^{k} S_0 \cap S_i \tag{2.27}$$

Generally the functions f_i are highly non-linear and it is not possible to solve them analytically. But these can be solved point by point using computer-aided design.

Consider the special case of a PID controller in which there are adjustable variables k_p, k_i and k_d. The closed loop system has some performance criteria like gain margin,phase margin, overshoot, rise-time etc. For this case, firstly the entire set S_0 of k_p, k_i and k_d which stabilizes the given plant is determined. The design objective is to find the set S given the performance specifications gain-margin $> F_{GM}$, phase-margin $> F_{PM}$, over-shoot $< F_{OS}$ etc. Then with the aid of computer the set$S = S_{GM} \cap S_{PM} \cap S_{OS} \cdots$ is obtained which lies within S_0. This set meets all the design objectives.

The above concept is used for performance evaluation of controllers in the following chapters.

CHAPTER III

GUI BASED DESIGN OF DISCRETE TIME CONTROLLERS

In this chapter, the design of Proportional (P), Proportional-Derivative (PD), Proportional -Integral (PI) and Proportional-Integral-Derivative (PID) controllers is studied. In the first section, the theory and algorithm is stated for each of the above types of controllers. One example to illustrate each of the above cases is also provided. In the next section, a description about the GUI and its functioning is given along with brief description of the main functions used. In the last section, a variety of examples illustrating different aspects of GUI based design are described.

A. Theory and Algorithm

1. Proportional Controllers

The theory is based on the mathematical background provided in the previous chapter and in [18]. Consider a discrete time plant represented by its transfer function

$$P(z) = \frac{N(z)}{D(z)} \tag{3.1}$$

where $N(z)$,$D(z)$ are real polynomials. Let degree$[D(z)] = n$ and degree$[N(z)] \leq n$. The plant is stabilized by a controller $C(z)$ as shown in Fig. 1. For a proportional controller,

$$C(z) = K \tag{3.2}$$

The closed loop characteristic equation of this system, $\delta(z)$ is given by

$$\delta(z, K) = D(z) + KN(z) \tag{3.3}$$

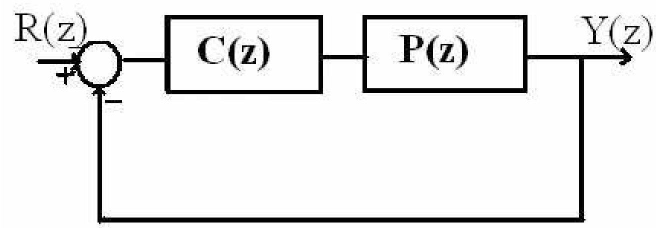

Fig. 1. Block diagram of a discrete time closed loop system

The closed loop system is stable if and only if $\delta(z)$ is Schur stable. Therefore, the problem is to find the values of K for which $\delta(z)$ is Schur stable. For this, Tchebychev representation of $D(z)$ and $N(z)$ is obtained as

$$\begin{aligned} D\left(e^{j\theta}\right) &= R_D(u) + j\sqrt{1-u^2}T_D(u) \\ N\left(e^{j\theta}\right) &= R_N(u) + j\sqrt{1-u^2}T_N(u) \end{aligned} \tag{3.4}$$

Also, $N(z^{-1})$ can be represented as

$$N\left(e^{-j\theta}\right) = R_D(u) - j\sqrt{1-u^2}T_D(u) \tag{3.5}$$

and

$$N\left(z^{-1}\right) = \frac{N_r(z)}{z^l} \tag{3.6}$$

where $N_r(z)$ is the reverse polynomial and l is the degree of $N(z)$. Next a rational function $\nu(z)$ is constructed so that

$$\nu(z) \quad = \quad \delta(z,K)N\left(z^{-1}\right)$$

$$= \quad D(z)N\left(z^{-1}\right) + KN(z)N\left(z^{-1}\right). \tag{3.7}$$

Finding the Tchebychev representation of the above equation,

$$\begin{aligned}
\nu(z) &= \left.\frac{\delta(z,K)N_r(z)}{z^l}\right|_{z=e^{j\theta}} \\
&= \left(R_D(u) + j\sqrt{1-u^2}T_D(u)\right)\left(R_N(u) - j\sqrt{1-u^2}T_N(u)\right) \\
&\qquad +K\left[R_N^2(u) + \left(1-u^2\right)T_N^2(u)\right] \\
&= \underbrace{R_D(u)R_N(u) + \left(1-u^2\right)T_D(u)T_N(u) + K\left[R_N^2(u) + \left(1-u^2\right)T_N^2(u)\right]}_{R(u,K)} \\
&\qquad +j\sqrt{1-u^2}\underbrace{\left[T_D(u)R_N(u) - R_D(u)T_N(u)\right]}_{T(u)} \\
&= P_1(u) + KP_3(u) + j\sqrt{1-u^2}P_2(u) \\
&= R(u,K) + j\sqrt{1-u^2}T(u).
\end{aligned} \tag{3.8}$$

where

$$\begin{aligned}
P_1(u) &= R_D(u)R_N(u) + \left(1-u^2\right)T_D(u)T_N(u) \\
P_2(u) &= R_N(u)T_D(u) - R_D(u)T_N(u) \\
P_3(u) &= R_N^2(u) + \left(1-u^2\right)T_N^2(u)
\end{aligned} \tag{3.9}$$

It is observed that the real part of the polynomial depends on K while the imaginary part is independent of K. Let $t_i, i = 1, 2, \cdots,$ denote the real zeros of odd multiplicity of the polynomial $T(u)$, for u in $(-1, 1)$. To t_i, $t_0 = -1$ and $t_{k+1} = 1$ is appended such that t_i varying from t_0 to t_{k+1}. Also let

$$\text{Sgn}\left[R\left(t_j, K\right)\right] = i_j, \qquad j = 0, 1, \cdots, k+1 \tag{3.10}$$

wheret each i_j can be either $+1, -1$ or 0. A particular choice of $[i_0, i_1, \cdots, i_{k+1}]$ is called a *string*.

Further let i_δ, i_{N_r} are the number of zeros of $\delta(z,K)$ and $N_r(z)$ inside unit circle. It is assumed that $N(z)$ has no unit circle zeros and hence neither does $N_r(z)$. Now applying of the formula gin Theorem II.2 gives:

$$\begin{aligned} i_\delta \quad + \quad i_{N_r} - l = \frac{1}{2}\mathrm{Sgn}\left[T^{(p)}(-1)\right] \\ \left(\mathrm{Sgn}\left[R(K,-1)\right] + 2\sum_{j=1}^{k}(-1)^j \mathrm{Sgn}\left[R\left(K,t_j\right)\right] + (-1)^{k+1}\mathrm{Sgn}\left[R(K,+1)\right]\right) \end{aligned} \tag{3.11}$$

For closed loop stability $i_\delta = n$. i_{N_r} and l are also known as i_N is known. Thus the equation yields the sets of strings corresponding to stability known as the set of feasible strings. Each feasible string gives a set of linear inequalities in K. The entire set of stabilizing gains is obtained by solving such sets of linear inequalities.

Thus, the algorithm for stabilizing Constant gain controller is as follows:

Step 1: Express $N(z)$ and $D(z)$ in terms of their Tchebychev representation as given in equation (3.4).

Step 2: Obtain $P_1(u)$,$P_2(u)$ and $P_3(u)$ as given in equation (3.9).

Step 3: Obtain the required signature of ν as

$$\sigma(\nu) = i_\delta + i_{N_r} - l \tag{3.12}$$

Step 4: Find the real distinct finite zeros of odd multiplicities of $T(u)$ between $(-1,+1)$ and arrange them as $-1 < t_1 < t_2 < \cdots < t_k < +1$ where k is the number of roots.

Step 5: Construct the sequence of numbers $i_0, i_1, \cdots, i_k, i_{k+1}$ having values 1 or -1 such that it covers all possible combinations. This set is defined as A_k such that $A_k := \{i_0, i_1, \cdots, i_k, i_{k+1}\}$.

Step 6: Determine the set of admissible strings $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$ in A_k such that the equation

$$i_\delta + i_{N_r} - l = \frac{1}{2}\mathrm{Sgn}\left[T^{(p)}(-1)\right]\left(i_0 + 2\sum_{j=1}^{k}(-1)^j i_k + (-1)^{k+1} i_{k+1}\right) \tag{3.13}$$

Step 7: If there is no admissible string, go to **Step 10**.

Step 8: For an admissible string $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$, determine the set of values of K which will simultaneously satisfy the inequalities

$$[P_1(u) + KP_3(u)]i_t > 0 \tag{3.14}$$

for all $t = 0, 1, \cdots, k+1$.

Step 9: Repeat **Step 8** for all admissible strings $I_1, I_2, \cdots, I_v$ to obtain the corresponding K ranges as $S_1, S_2, \cdots, S_v$.The entire stabilizing set of K is given by

$$S = \cup_{k=1}^{v} S_k \tag{3.15}$$

Step 10: Terminate the algorithm.

The above process is illustrated through the following example.

Example III.1 Consider the following plant

$$P(z) = \frac{N(z)}{D(z)} = \frac{z - 0.3}{z^3 + 0.6z^2 + 0.5z + 0.25} \tag{3.16}$$

As per equation (3.4)

$$\begin{aligned} R_n(u) &= -u - 0.3 \\ T_n(u) &= 1 \\ R_d(u) &= -4u^3 + 1.2u^2 + 2.5u - 0.35 \\ T_d(u) &= 4u^2 - 1.2u - 0.5 \end{aligned} \tag{3.17}$$

From this, the polynomials $P_1(u)$,$P_2(u)$ and $P_3(u)$ can be constructed as

$$\begin{aligned} P_1(u) &= -1.2u^3 + 2.36u^2 - 0.1u - 0.6050 \\ P_2(u) &= 1.2u^2 - 2.36u + 0.2 \\ P_3(u) &= -0.6u + 1.09 \end{aligned} \tag{3.18}$$

The signature σ can be evaluated using equation (3.12) as

$$\sigma(\nu) = 3 + 0 - 1 = 2 \tag{3.19}$$

Also, $Sgn[T(-1) = 1$. Therefor the only possible string satisfying equation (3.13) is $[1, -1, 1]$. The real root of odd multiplicity lying between $(-1, 1)$ is 0.0888. Arranging them in increasing order, $-1 < 0.0888 < 1$. With this data, using equation (3.14), the following inequalities are generated.

$$\begin{aligned} 1.6900K &> -3.0550 \\ 1.0367K &< 0.5961 \\ 0.4900K &> -0.4550 \end{aligned} \tag{3.20}$$

The range of K satisfying the above inequalities is $-0.9286 < K < 0.5750$. This is the range of K that stabilizes the given plant.

2. PD Controllers

As before, the theory is based on the previous chapter and [18]. Consider the same plant $P(z)$ as described in equation (3.1) and Fig. 1. Consider a typical PD controller of the form

$$C(z) = K_P + \frac{K_D}{T} \cdot \frac{z-1}{z} = \frac{(K_P T + K_D)\, z - K_D}{Tz} = \frac{\left(K_P + \frac{K_D}{T}\right)\left(z - \frac{\frac{K_D}{T}}{K_P + \frac{K_D}{T}}\right)}{z} \tag{3.21}$$

which can be re-written in the form

$$C(z) = \frac{K_1\,(z - K_2)}{z} \tag{3.22}$$

where

$$K_P = K_1 - K_1 K_2 \qquad \text{and} \qquad K_D = K_1 K_2 T \tag{3.23}$$

The characteristic equation $\delta(z)$ with the above controller structure will become

$$\delta(z) = zD(z) + K_1\,(z - K_2)\,N(z) \tag{3.24}$$

As in previous section, a rational function $\nu(z)$ is constructed

$$\begin{aligned} \nu(z) &= \delta(z) N\left(z^{-1}\right) \\ &= zD(z)N\left(z^{-1}\right) + K_1\,(z - K_2)\,N(z)N\left(z^{-1}\right) \end{aligned} \tag{3.25}$$

In Tchebychev representation,

$$\begin{aligned} \nu(z) &= \left.\delta(z) N\left(z^{-1}\right)\right|_{z=e^{j\theta}, u=-\cos\theta} \\ &= R\,(u, K_1, K_2) + j\sqrt{1-u^2}\,T\,(u, K_1) \end{aligned} \tag{3.26}$$

where

$$R\,(u, K_1, K_2) = -uP_1(u) - \left(1 - u^2\right) P_2(u) - K_1\,(u + K_2)\,P_3(u) \tag{3.27}$$

$$T\,(u, K_1) = K_1 P_3(u) + P_1(u) - uP_2(u). \tag{3.28}$$

and P_1,P_2,P_3 are as described in equation (3.9). It is observed that the real part has K_1 and K_2 while the imaginary part has only K_1. If K_1 is fixed, the roots of $T(u, K_1)$ can be obtained. With this, linear inequalities in K_2 are obtained using theorem II.2.

The algorithm for computing the stabilizing set of PD controllers is as follows:

Step 1: Express $N(z)$ and $D(z)$ in terms of their Tchebychev representation as given in equation (3.4).

Step 2: Obtain $P_1(u)$,$P_2(u)$ and $P_3(u)$ as given in equation (3.9).

Step 3: Obtain the required signature of ν as

$$\sigma(\nu) = i_\delta + i_{N_r} - l \tag{3.29}$$

Step 4: In order to satisfy equation (3.29), $T(u, K_1)$ should have atleast

$$\frac{2(i_\delta + i_{N_r} - l) - 2}{2} \tag{3.30}$$

real distinct finite zeros of odd multiplicities of $T(u, K_1)$ between $(-1, +1)$. From this, the allowable ranges of P_i,$i = 1, 2, \cdots, d$ of K_1 are determined. These resulting ranges of K_1 are the only vlues of K_1 for which stabilizing values of K_2 may exist.

Step 5: If there exists no value of K_1 satisfying **Step 4**, then output NO SOLUTION and go to **Step 16**.

Step 6: Initialize $j = 1$ and $P = P_j$.

Step 7: Pick a range $[K_{low}, K_{upp}]$ in P and initialize $K_1 = K_{low}$.

Step 8: Pick up the number of grid points N and set

$$\text{step} = \frac{K_{upp} - K_{low}}{N + 1} \tag{3.31}$$

Step 9: Increase K_1 as $K_1 = K_1 + \text{step}$. If $K_1 > K_{upp}$ go to **Step 16**.

Step 10: For a fixed value of K_1, find the real distinct finite zeros of odd multiplicities of $T(u)$ between $(-1, +1)$ and arrange them as $-1 < t_1 < t_2 < \cdots < t_k < +1$ where

k is the number of roots.

Step 11: Construct the sequence of numbers $i_0, i_1, \cdots, i_k, i_{k+1}$ having values 1 or -1 such that it covers all possible combinations. This set is defined as A_k such that $A_k := \{i_0, i_1, \cdots, i_k, i_{k+1}\}$.

Step 12: Determine the set of admissible strings $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$ in A_k such that the equation

$$i_\delta + i_{N_r} - l = \frac{1}{2}\mathrm{Sgn}\left[T^{(p)}(-1)\right]\left(i_0 + 2\sum_{j=1}^{k}(-1)^j i_k + (-1)^{k+1} i_{k+1}\right) \tag{3.32}$$

If there is no admissible string, go to **Step 16**.

Step 13: For an admissible string $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$, determine the set of values of K_2 which will simultaneously satisfy the inequalities

$$[-uP_1(u) - \left(1 - u^2\right) P_2(u) - K_1\left(u + K_2\right) P_3(u)]i_t > 0 \tag{3.33}$$

for all $t = 0, 1, \cdots, k+1$.

Step 14: Repeat **Step 13** for all admissible strings $I_1, I_2, \cdots, I_v$ to obtain the corresponding K_2 ranges as $S_1, S_2, \cdots, S_v$.The entire stabilizing set of K_2 is given by

$$S = \cup_{k=1}^{v} S_k \tag{3.34}$$

Step 15: Go to **Step 9**.

Step 16: Set $j = j + 1$ and $P = P_j$. If $j \leq d$, go to **Step 7**, else Terminate the algorithm.

Example III.2 Consider the same plant as in example III.1. The polynomials $R_n(u)$, $R_d(u)$, $T_n(u)$, $T_d(u)$ and $P_1(u)$, $P_2(u)$, $P_3(u)$ are same as previous example. The signature is given by The signature σ can be evaluated using equation (3.12) as

$$\sigma(\nu) = 4 + 0 - 1 = 3 \tag{3.35}$$

This implies that the minimum number of real distinct roots of odd multiplicites of $T(u)$ between -1 and 1 should be 2. The values of K_1 for which this condition is satisfied is $[-2.8, 0.5]$. Choosing $K_1 = -1.0$, $Sgn[T(-1,-1) = 1$. Therefore the only possible string satisfying equation (3.32) is $[1, -1, 1, -1]$. The real root of odd multiplicity lying between $(-1, 1)$ are -0.5543 and 0.6983. Arranging them in increasing order, $-1 < -0.5543 < 0.6983 < 1$. With this data, using equation (3.33), the following inequalities are generated.

$$\begin{aligned} 1.6868K_2 &> -1.3682 \\ 1.4199K_2 &< 1.8767 \\ 0.6698K_2 &> -0.8628 \\ 0.4891K_2 &< -0.0341 \end{aligned} \tag{3.36}$$

The range of K_2 satisfying the above inequalities is $-0.8111 < K_2 < -0.0607$. This is the range of K_2 that stabilizes the given plant for $K_1 = -1$. The entire stabilizing set of $K_1 - K_2$ is shown in Fig. 2.

3. PI Controllers

The same plant $P(z)$ as described in equation (3.1) and Fig. 1 is considered for this case also. However, for a PI controller, $C(z)$ is of the form

$$C(z) = K_P + K_I T \cdot \frac{z}{z-1} = \frac{(K_P + K_I T)\, z - K_P}{z-1}$$

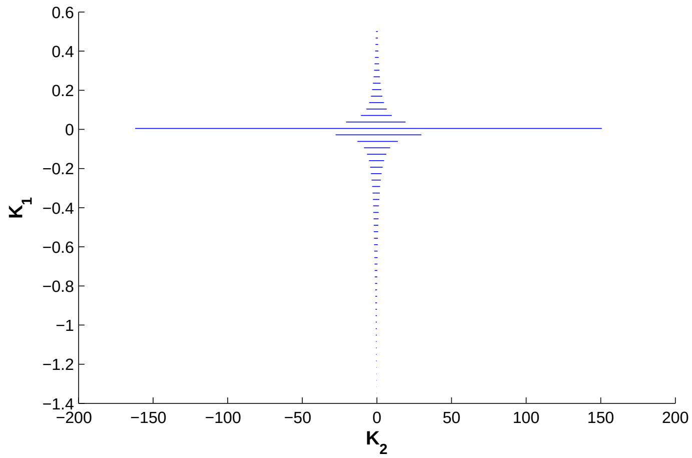

Fig. 2. Stabilizing set of PD controller for example III.2

$$= \frac{(K_P + K_I T)\left(z - \frac{K_P}{K_I T + K_P}\right)}{z-1} \tag{3.37}$$

re-writing in a different form,

$$C(z) = \frac{K_1 (z - K_2)}{z-1} \tag{3.38}$$

where

$$K_P = K_1 K_2 \qquad \text{and} \qquad K_I = \frac{K_1 - K_1 K_2}{T}. \tag{3.39}$$

The characteristic polynomial $\delta(z)$ is given by

$$\delta(z) = (z-1)D(z) + K_1 (z - K_2) N(z). \tag{3.40}$$

To achieve parameter separation, $\nu(z)$ is defined as

$$\begin{aligned} \nu(z) &= \delta(z) N\left(z^{-1}\right) \\ &= (z-1)D(z)N\left(z^{-1}\right) + K_1 (z - K_2) N(z) N\left(z^{-1}\right) \end{aligned} \tag{3.41}$$

In Tchebychev representation,

$$\begin{aligned} \nu(z) &= \delta(z)N\left(z^{-1}\right)\Big|_{z=e^{j\theta},u=-\cos\theta} \\ &= R\left(u,K_1,K_2\right)+j\sqrt{1-u^2}T\left(u,K_1\right) \qquad (3.42) \end{aligned}$$

where

$$\begin{aligned} R\left(u,K_1,K_2\right) &= -(u+1)P_1(u)-\left(1-u^2\right)P_2(u)-K_1\left(u+K_2\right)P_3(u) \qquad (3.43) \\ T\left(u,K_1\right) &= K_1P_3(u)+P_1(u)-(u+1)P_2(u). \qquad (3.44) \end{aligned}$$

and P_1,P_2,P_3 are as described in equation (3.9). It is observed that the real part has K_1 and K_2 while the imaginary part has only K_1. If K_1 is fixed, the roots of $T(u,K_1)$ can be obtained. With this, linear inequalities in K_2 are obtained using Theorem II.2.

The algorithm for PI Controller is almost the same as of a PD Controller and hence is omitted.

Example III.3 Consider the same plant as in example III.1. The polynomials $R_n(u)$, $R_d(u)$, $T_n(u)$, $T_d(u)$ and $P_1(u)$, $P_2(u)$, $P_3(u)$ are same as previous example.The signature σ can be evaluated using equation (3.12) as

$$\sigma(\nu)=4+0-1=3 \qquad (3.45)$$

This implies that the minimum number of real distinct roots of odd multiplicites of $T(u)$ between -1 and 1 should be 2. The values of K_1 for which this condition is satisfied is $[-1.8,0.8]$. Choosing $K_1=-0.1$, $Sgn[T(-1,-0.1)=1$. Therefore the only possible string satisfying equation (3.32) is $[1,-1,1,-1]$. The real root of odd multiplicity lying between $(-1,1)$ are -0.4514 and 0.4468. Arranging them in increasing order, $-1<-0.4514<0.4468<1$. With this data, using equation (3.33),

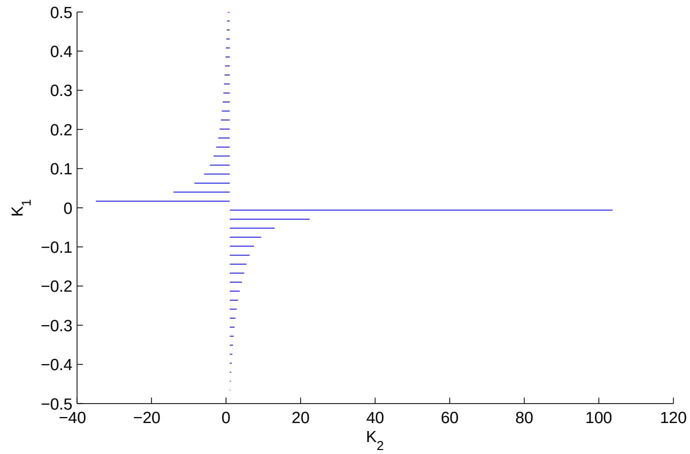

Fig. 3. Stabilizing set of PI controller for example III.3

the following inequalities are generated.

$$\begin{aligned} 0.0473K_2 &> 0.0473 \\ 0.0777K_2 &< 1.1361 \\ 0.1326K_2 &> -1.4515 \\ 0.1632K_2 &< 1.5268 \end{aligned} \tag{3.46}$$

The range of K_2 satisfying the above inequalities is $1.0 < K_2 < 9.3259$. This is the range of K_2 that stabilizes the given plant for $K_1 = -0.1$. The entire stabilizing set of $K_1 - K_2$ is shown in Fig. 3.

4. PID Controllers

For PID controller, $C(z)$ is given by

$$C(z) = K_P + K_I T \frac{z}{z-1} + \frac{K_D}{T}\frac{z-1}{z}$$

$$= \frac{K_2 z^2 + K_1 z + K_0}{z(z-1)} \tag{3.47}$$

where

$$\begin{aligned} K_P &= -K_1 - 2K_0, \\ K_I &= \frac{K_0 + K_1 + K_2}{T} \\ K_D &= K_0 T. \end{aligned} \tag{3.48}$$

The technique described in [15] to find the stabilizing set is as follows. Let the plant be as described in equation (3.1).Therefore, the characteristic equation is

$$\delta(z) := z(z-1)D(z) + \left(K_2 z^2 + K_1 z + K_0\right) N(z). \tag{3.49}$$

Multiplying with $z^{-1}N(z^{-1})$,

$$\begin{aligned} z^{-1}\delta(z)N(z^{-1}) = (z-1)D(z)N(z^{-1}) \\ + \left(K_2 z + K_1 + K_0 z^{-1}\right) N(z)N(z{-}1). \end{aligned} \tag{3.50}$$

Now using Tchebychev representation

$$\begin{aligned} & z^{-1}\delta(z)N(z^{-1}) \\ = \; & -(u+1)P_1(u) - (1-u^2)P_2(u) \\ & \qquad - \left[(2K_2 - K_3)\,u - k_1\right] P_3(u) \\ & + \jmath\sqrt{(1-u^2)}\left[-(u+1)P_2(u) + P_1(u) + K_3 P_3(u)\right] \\ = \; & R(u, K_1, K_2, K_3) + \jmath\sqrt{(1-u^2)}T(u, K_3) \end{aligned} \tag{3.51}$$

where $P_1(u)$,$P_2(u)$ and $P_3(u)$ is as described in equation (3.9)

Note that the parameters K_1, K_2 and K_3 instead of K_P, K_I and K_D without any loss of flexibility as these 2 sets are related to each other by simple coordinate transformation

$$\begin{bmatrix} K_P \\ K_I \\ K_D \end{bmatrix} = \begin{bmatrix} -2 & -1 & 0 \\ \frac{1}{T} & \frac{1}{T} & \frac{1}{T} \\ T & 0 & 0 \end{bmatrix} \begin{bmatrix} K_0 \\ K_1 \\ K_2 \end{bmatrix}$$
$$= \begin{bmatrix} -2 & -1 & 0 \\ \frac{1}{T} & \frac{1}{T} & \frac{1}{T} \\ T & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \tag{3.52}$$

It is noted that the imaginary part $T(u)$ is dependent only on K_3. Now using the Theorem II.2, the following inequalities are obtained which have K_1 and K_2 as unknowns when we fix K_3.

$$\begin{aligned} [R(u)|_{u=-1}]\, i_0 &> 0 \\ \left[R(u)|_{u=-t_j}\right] i_j &> 0 \\ [R(u)|_{u=-}]\, i_{k+1} &> 0 \end{aligned} \tag{3.53}$$

where $j = 1, \cdots, k$.

These inequalities gives feasible regions in K_1-K_2 space for fixed K_3. As K_3 is varied, the entire stabilizing set is obtained.

The algorithm for computing stabilizing set of PID controllers is as follows:

Step 1: Express $N(z)$ and $D(z)$ in terms of their Tchebychev representation as given in equation (3.4).

Step 2: Obtain $P_1(u)$,$P_2(u)$ and $P_3(u)$ as given in equation (3.9).

Step 3: Obtain the required signature of ν as

$$\sigma(\nu) = i_\delta + i_{N_r} - (l + 1) \tag{3.54}$$

Step 4: In order to satisfy equation (3.54), $T(u, K_3)$ should have atleast

$$\frac{2(i_\delta + i_{N_r} - (l+1)) - 2}{2} \tag{3.55}$$

real distinct finite zeros of odd multiplicities of $T(u, K_3)$ between $(-1, +1)$. From this, the allowable ranges of P_i,$i = 1, 2, \cdots, d$ of K_3 are determined. These resulting ranges of K_3 are the only values of K_3 for which stabilizing values in $K_1 - K_2$ may exist.

Step 5: If there exists no value of K_3 satisfying **Step 4**, then output NO SOLUTION and go to **Step 16**.

Step 6: Initialize $j = 1$ and $P = P_j$.

Step 7: Pick a range $[K_{low}, K_{upp}]$ in P and initialize $K_1 = K_{low}$.

Step 8: Pick up the number of grid points N and set

$$\text{step} = \frac{K_{upp} - K_{low}}{N+1} \tag{3.56}$$

Step 9: Increase K_3 as $K_3 = K_3 + \text{step}$. If $K_3 > K_{upp}$ go to **Step 16**.

Step 10: For a fixed value of K_3, find the real distinct finite zeros of odd multiplicities of $T(u)$ between $(-1, +1)$ and arrange them as $-1 < t_1 < t_2 < \cdots < t_k < +1$ where k is the number of roots.

Step 11: Construct the sequence of numbers $i_0, i_1, \cdots, i_k, i_{k+1}$ having values 1 or -1 such that it covers all possible combinations. This set is defined as A_{k3} such that $A_{k3} := \{i_0, i_1, \cdots, i_k, i_{k+1}\}$.

Step 12: Determine the set of admissible strings $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$ in A_{k3} such that the equation

$$i_\delta + i_{N_r} - (l+1) = \frac{1}{2}\text{Sgn}\left[T^{(p)}(-1)\right]\left(i_0 + 2\sum_{j=1}^{k}(-1)^j i_k + (-1)^{k+1} i_{k+1}\right) \tag{3.57}$$

If there is no admissible string, go to **Step 16**.

Step 13: For an admissible string $I = \{i_0, i_1, \cdots, i_k, i_{k+1}\}$, determine the set of values of $K_1 - K_2$ which will simultaneously satisfy the inequalities

$$[-(u+1)P_1(u) - \left(1-u^2\right) P_2(u) - [(2K_2 - K_3)u - K_1]P_3(u)]i_t > 0 \tag{3.58}$$

for all $t = 0, 1, \cdots, k+1$.

Step 14: Repeat **Step 13** for all admissible strings $I_1, I_2, \cdots, I_v$ to obtain the corresponding $K_1 - K_2$ sets as $S_1, S_2, \cdots, S_v$.The entire stabilizing set of $K_1 - K_2$ is given by

$$S = \cup_{k=1}^{v} S_k \tag{3.59}$$

Step 15: Go to **Step 9**.

Step 16: Set $j = j + 1$ and $P = P_j$. If $j \leq d$, go to **Step 7**, else Terminate the algorithm.

To illustrate this procedure, consider the following example.

Example III.4 Consider the same plant as in example III.1. The polynomials $R_n(u)$, $R_d(u)$, $T_n(u)$, $T_d(u)$ and $P_1(u)$, $P_2(u)$, $P_3(u)$ are same as previous example. The signature is given by The signature σ can be evaluated using equation (3.54) as

$$\sigma(\nu) = 5 + 0 - 1 = 3 \tag{3.60}$$

This implies that the minimum number of real distinct roots of odd multiplicites of $T(u)$ between -1 and 1 should be 2. The values of K_3 for which this condition is satisfied is $[-1.8, 0.8]$. Choosing $K_3 = -1$, $Sgn[T(-1,-1) = 1$. Therefore the only possible string satisfying equation (3.57) is $[1, -1, 1, -1]$. The real root of odd multiplicity lying between $(-1, 1)$ are -0.8653 and 0.4959. Arranging them in increasing order, $-1 < -0.8653 < 0.4959 < 1$. With this data, using equation (3.58), the

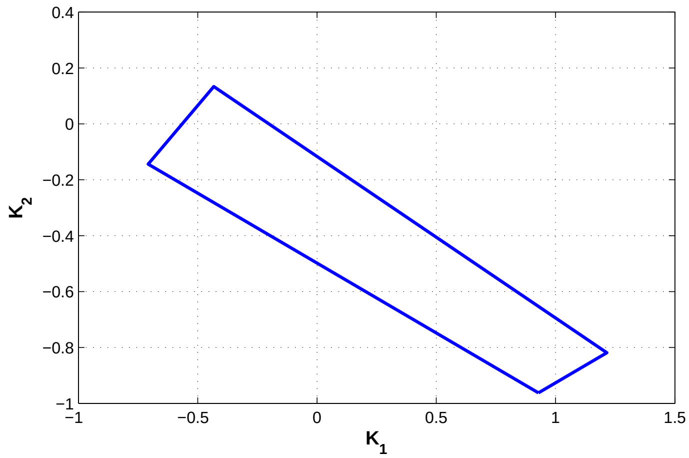

Fig. 4. Stabilizing set of PID controller at $K_3 = -1$ for example III.4

following inequalities are generated.

$$
\begin{aligned}
1.6900K_1 + 3.3800K_2 &> -1.6836 \\
1.6092K_1 + 2.7849K_2 &< -0.3252 \\
0.7925K_1 - 0.7860K_2 &> -0.4476 \\
0.4900K_1 - 0.9800K_2 &< 1.3981
\end{aligned}
\tag{3.61}
$$

The range of $K_1 - K_2$ satisfying the above inequalities is shown in Fig. 4 This is the range of K_2 that stabilizes the given plant for $K_1 = -0.1$. The entire stabilizing set of $K_1 - K_2 - K_3$ is shown in Fig. 5

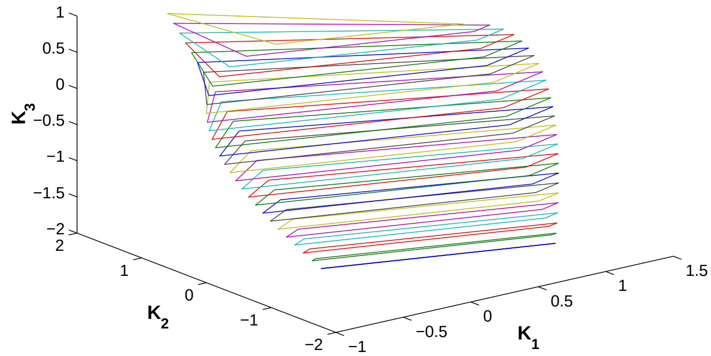

Fig. 5. 3D stabilizing set of PID controller for example III.4

B. GUI for Discrete Time Controller Design

1. The Interface

GUIs have been developed for P, PI, PD and PID controllers in MATLAB. The interface for all types of controllers is almost similar. The two variations of GUIs are as shown in Fig. 6 and Fig. 7. The inputs for the programs are the numerator, the denominator and the sampling time of the discrete time plant. The numerator and denominator are entered as an array with the leading coefficient denoting the coefficient of the highest degree of the polynomial. The sampling time is entered in seconds.

On pressing Start Button, the GUI gives the range of values of one of the parameters which is fixed which is called the free parameter. In case of PID Controller, this is K_3 while in case of PI or PD controller it is K_1. On selecting a particular value of the free parameter, the set of K_2 or $K_1 - K_2$ values is obtained depending on the type of controller.

The performance is evaluated in two ways:

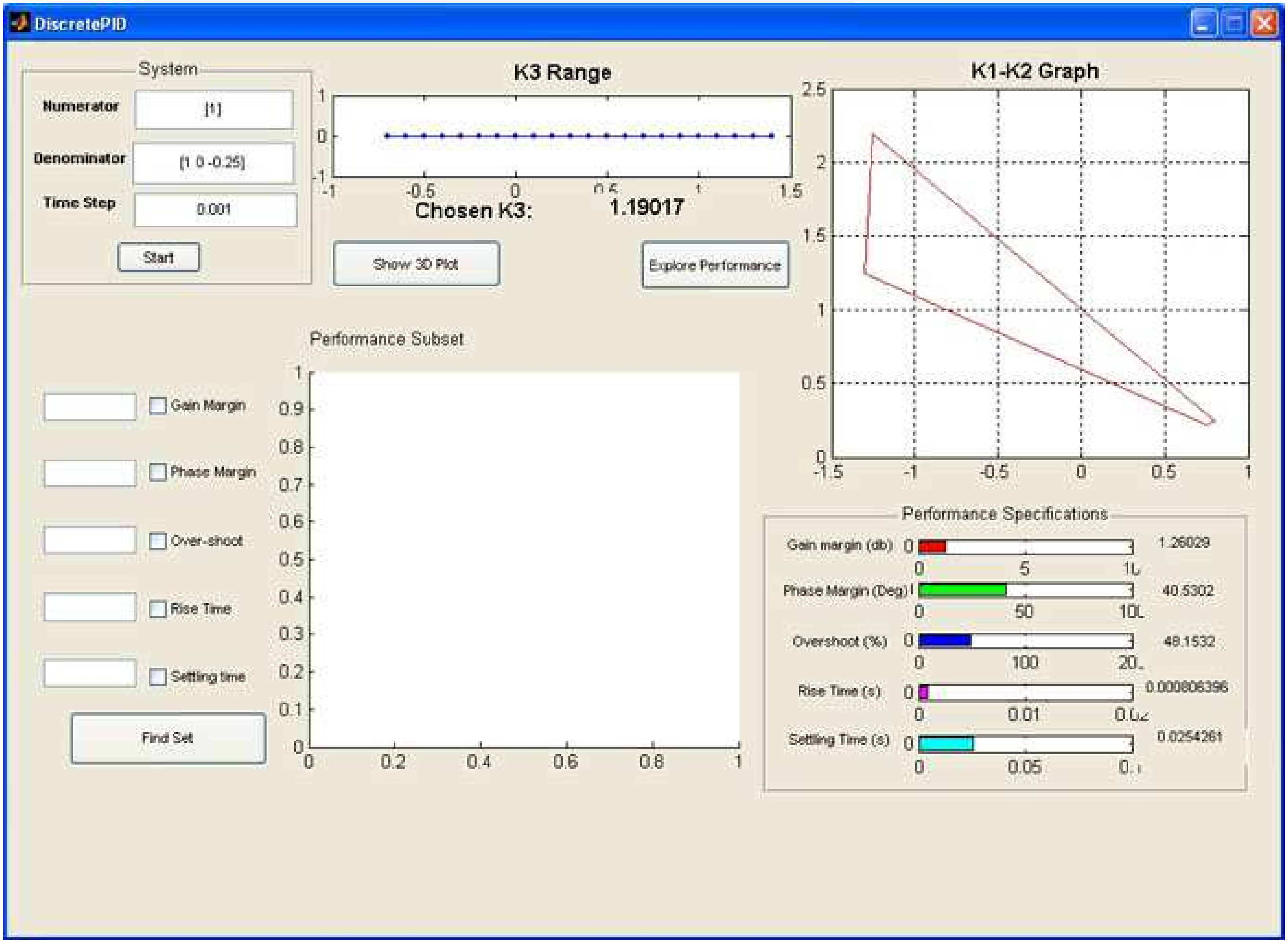

Fig. 6. GUI for PID controller design

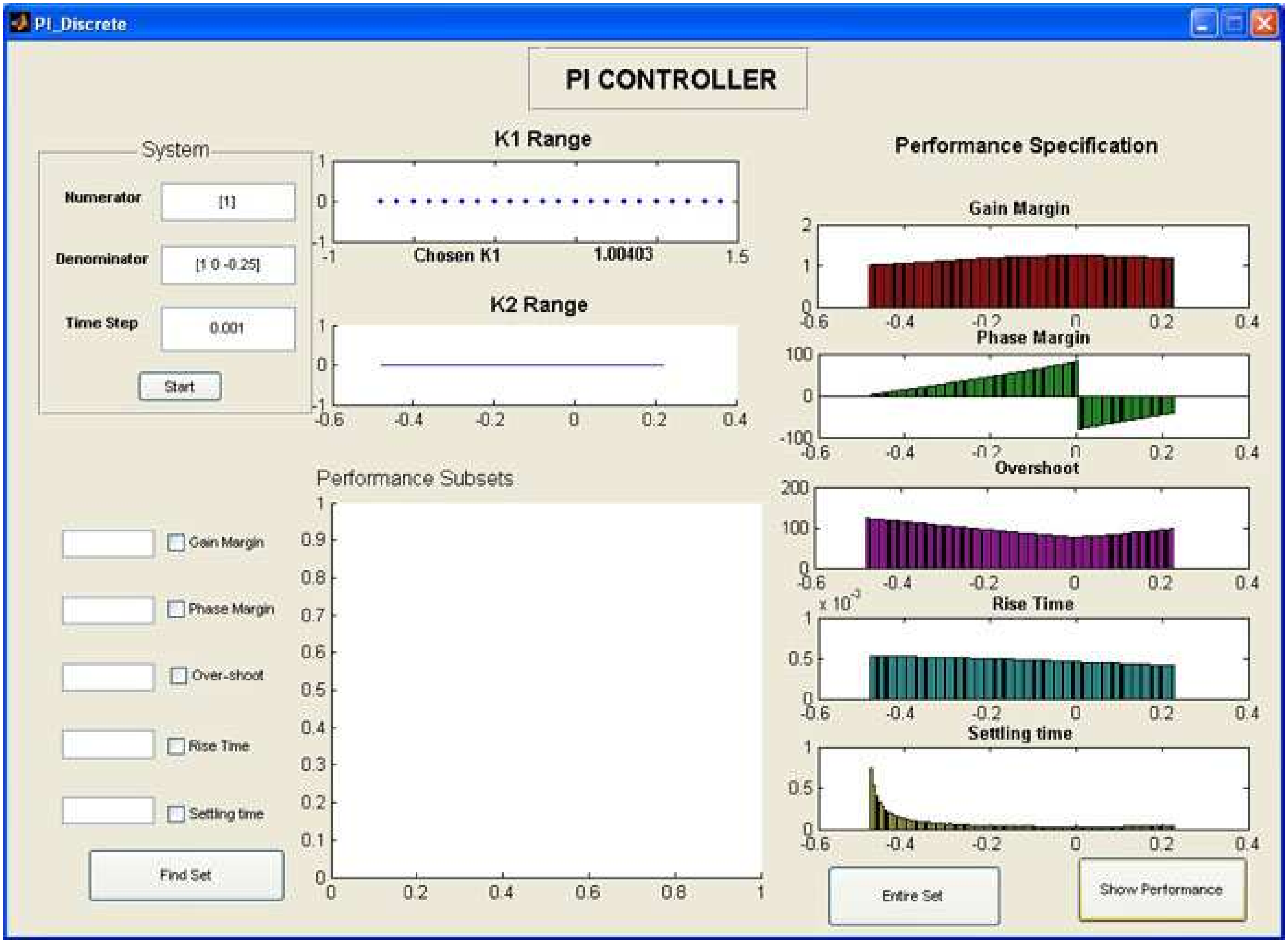

Fig. 7. GUI for PI controller design

- **Performance Specifications** Various important time and frequency domain performance criteria are considered. These involve phase margin, gain margin, over shoot, rise time and settling time. In case of P, PD and PI controllers, for a chosen value of K_1, on clicking the 'View Performance ' button, the variation of the above mentioned performance criteria along the set of stabilizing values of K in case of P and K_2 in case of PI and PD are displayed. This gives a general idea of how the performance varies along the set. This is of great help to the designer who can see all the performances in one screen and hence can decide on the best possible value for the controller. In case of PID, for a chosen value of K_3, a convex $K_1 - K_2$ set is obtained. On clicking the 'View Performance ' button, a cursor appears on the $K_1 - K_2$ set. On choosing a particular value of K_1 and K_2 in that set, all the performance appear as sticks on the GUI. As the designer clicks on different $K_1 - K_2$ values, the performance specifications change. The increasing or decreasing values of the performance along any particular direction gives the designer a good insight of the controller values ideal for the system. It also gives the idea of the performance that can be expected out of the PID controller. For example, after exploring different $K_1 - K_2$ values for different K_3, the designer may see that Gain Margin greater than 10db is not possible to achieve with PID controllers alone. Further, as the designer clicks on any controller value, the step response, the error signal and the control signal for that chosen controller value is displayed. The designer gets to see the transient response and the control signal from which it can be decided if these signals are desirable or feasible in real applications.

- **Performance Subset** The other possible scenario is taken into account in this block. Suppose the desired performance specifications are known and the designer wants to find the controller parameters achieving this set. The designer has to enter the desired performance and check the corresponding box. It is considered that the designer will need gain and phase margin greater than the specified value while rise time, overshoot and settling time should be less than the specified value. On clicking the 'Find Subset ' button, the subset satisfying the desired values are shown. If no set exists achieving the given performance specifications, an empty set is shown.

Apart from this, the entire $2-D$ set in case of PI and PD controllers and the $3-D$ set for PID controller can be seen on clicking the appropriate button. For PID, both $K_1 - K_2 - K_3$ and $K_p - K_i - K_d$ sets are observed.

2. Functions Used

The functions provide the building blocks for the program. The basic functions for all the controllers are similar and are described below.

- **Digi_K.m, Digi_PI.m, Digi_PD.m, Digi_PID.m :** These are the main functions respectively for computing the stabilizing sets. Digi_K.m finds the entire set of K values stabilizing the system while the other 3 find a range of value of the free parameter and on choosing a particular value gives the stabilizing set in other parameters.

- **z2tcheby.m** When a polynomial and its degree are passed on as input parameters, this function gives the Tchebychev representation, $R(u)$ and $T(u)$ of the given polynomial.

- **create_poly.m** This function gives the polynomials $P_1(u)$, $P_2(u)$ and $P_3(u)$ when the numerator and denominator of the transfer function is given.

- **signature.m** Calculates the signature of the rational function ν for a given transfer function. When type= 0 or 1, it is for P or PI/PD controller respectively and when type= 2, it is PID controller.

- **RangeofK.m** This function determines the valid ranges of the free parameter based on the number of real, distinct roots of odd multiplicities in $[-1, 1]$ of the imaginary part of the rational function ν.

- **RootsatK.m** Given the imaginary part of ν and a particular value of the free parameter, this function gives the real, distinct roots of odd multiplicities in $[-1, 1]$ of the imaginary part of the rational function ν.

- **sgnT.m** This function evaluates the sign of $T(u, K3)$ that is $Sgn[T^p(-1)]$ where p is no. of times -1 is a root of $T(u, K3)$.

- **stringgen.m** This function generates the strings $[1, -1, ..]$ which satisfy the signature criterion for the given value. The inputs to the function are number of real roots between -1 and 1 and $2.signature.Sgn[T^p(-1)]$.

- **inequalsol.m** This function solves the linear inequalities and finds the edges of the polygon which satisfy the given inequalities.

- **lineq1d.m** This function solves 1 dimensional linear inequalities.

- **viewperformance.m** Given the chosen controller parameters and the transfer function, this function finds the values of the performance indices like gain margin, phase margin etc and displays them.

- **determsubset.m, determsubset_K.m, determsubset_PI.m, determsubset_PD.m** These functions grid the stabilizing set in a number of points. The performance specifications is then evaluated for each point on the grid and compared with the given performance indices. If the point satisfies all the given criteria, then it is marked.

C. Illustrative Examples

In this section various aspects of performance based controller design are discussed in different problems and their solutions are obtained using the GUI based software.

1. Constant Gain Controller Examples

Example III.5 Consider a discrete time transfer function

$$P(z) = \frac{-0.2z - 0.3}{z^3 - 0.4z^2 - 0.15z - 0.2}$$

The sampling time, $T = 0.001s$. When the numerator, denominator and sampling time is input to the GUI, the stabilizing set of gain values is obtained as shown in Fig. 8. The stabilizing values lie between $-2.7 < K < 0.5$. The variation of various performance indices like gain margin, phase margin, overshoot, rise time, settling time and steady state error along the range of K is also shown.

Further, consider that some design constraints are imposed.

Phase Margin$> 50^\circ$

Settling Time$< 0.05s$

Gain Margin$> 3db$

The subsets achieving these criteria are shown in Fig. 9, Fig. 10 and Fig. 11. It is observed that as more conditions are imposed the set shrinks and and is a subset of the previous set.

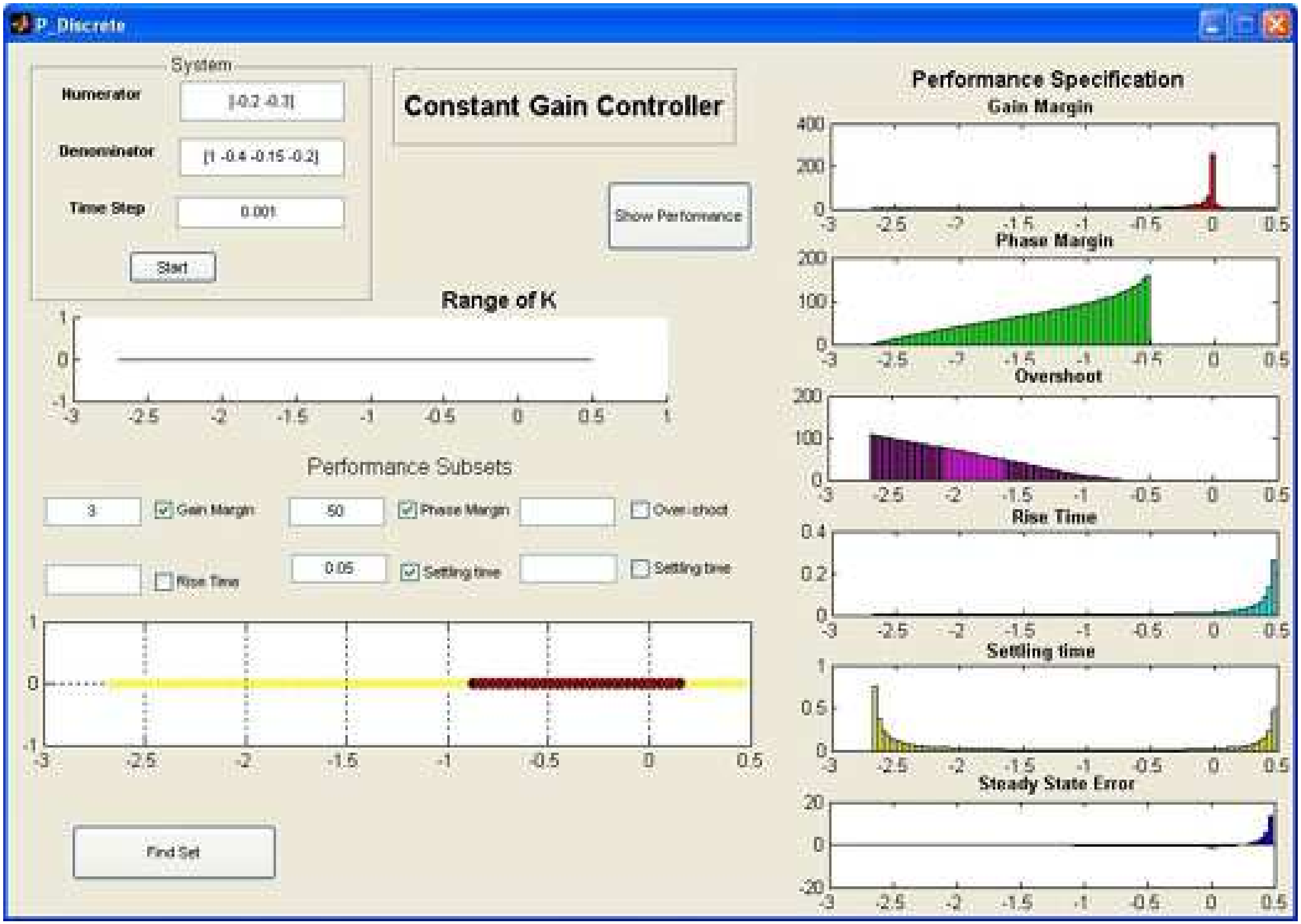

Fig. 8. GUI for P controller design in example III.5

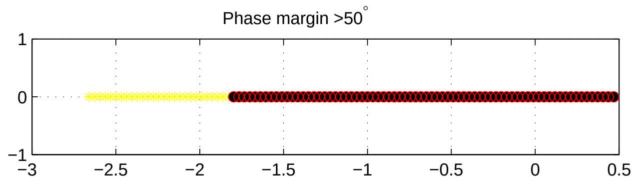

Fig. 9. The stabilizing set for example III.5 with phase margin $> 50°$

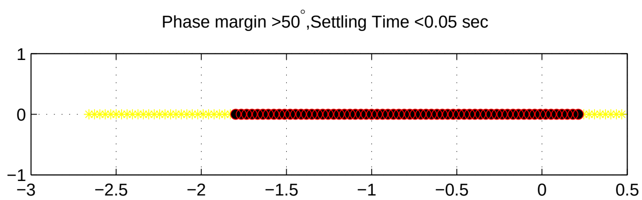

Fig. 10. The stabilizing set for example III.5 with phase margin $> 50°$ and settling time$< 0.05s$

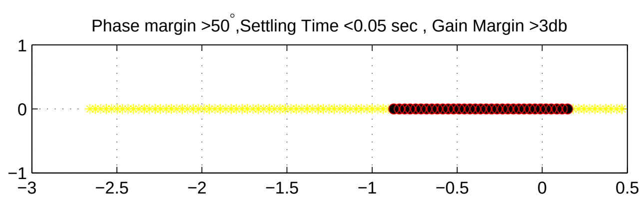

Fig. 11. The stabilizing set for example III.5 with phase margin $> 50°$, settling time$< 0.05s$ and gain margin$> 3db$

Example III.6 In this example, consider a discrete time transfer function

$$P(z) = \frac{0.8z + 0.5}{z^4 - 0.3z^3 + 0.7z^2 + 0.9z + 0.25}$$

The sampling time, $T = 0.001s$. The GUI is as shown in Fig. 12.

When the conditions, Gain Margin$> 0.8db$ and Phase Margin$> 2^{\circ}$ are imposed, it is observed that two disjoint sets are obtained which satisfy these conditions as shown in Fig. 13.

2. PD and PI Controller Examples

Example III.7 For designing a PD controller,consider the plant

$$P(z) = \frac{z - 0.2}{z^3 + 0.7z^2 + 0.3z + 0.8}$$

The sampling time, $T = 0.001s$. When the numerator, denominator and sampling time is input in the GUI as shown in Fig. 14, a set of valid values of K_1 is obtained. $-1.5 < K_1 < 0.8$. When a particular value of K_1 say $K_1 = -0.5$ is chosen, the set of stabilizing values of K_2 is obtained to be $-2.7 < K_2 < -1.3$. Variation of various performance indices along this range of K_2 is also displayed. The entire $K_1 - K_2$ set is as shown in Fig. 15. The subset satisfying the conditions Gain Margin$> 1db$ and Rise Time$< 0.001s$ is also shown in Fig. 14.

Example III.8 Consider the plant

$$P(z) = \frac{-0.2}{z^3 + 0.7z^2 + 0.3z + 0.8}$$

The sampling time, $T = 0.001s$. The objective is to find the entire set of stabilizing values for a PI controller and then find the subset achieving desired performance

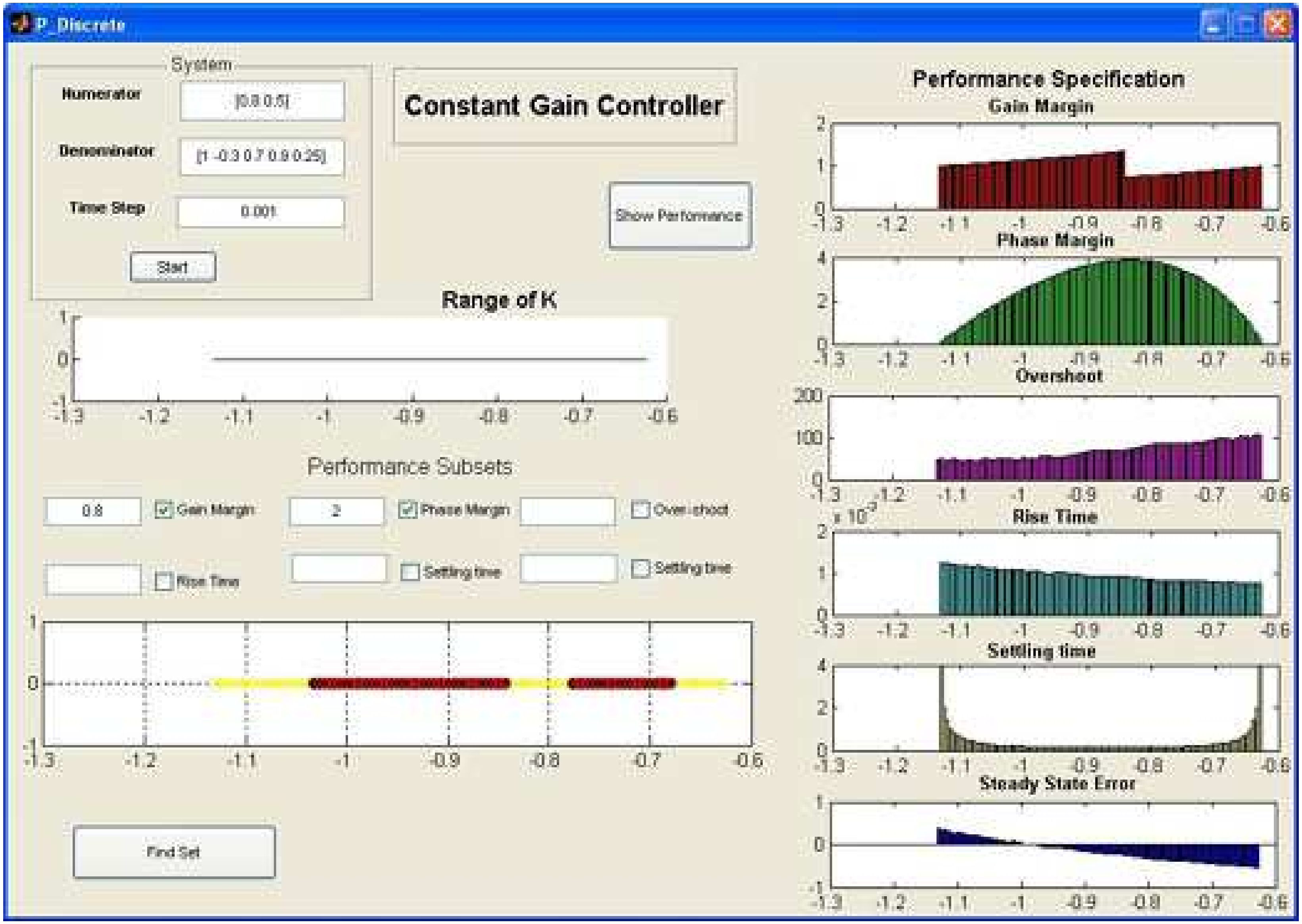

Fig. 12. GUI for P controller design in example III.6

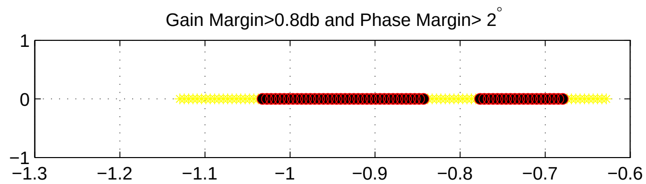

Fig. 13. The stabilizing set for example III.6 with phase margin $> 2°$ and gain margin$> 0.8db$

specifications. When the numerator, denominator and sampling time is input in the GUI as shown in Fig. 16, a set of valid values of K_1 is obtained. $-2.8 < K_1 < 5.7$. When a particular value of K_1 say $K_1 = 4$ is chosen, the set of stabilizing values of K_2 is obtained to be $0.18 < K_2 < 0.46$. Variation of various performance indices along this range of K_2 is also displayed. The entire $K_1 - K_2$ set is as shown in Fig. 17. The subset satisfying the conditions Gain Margin$> 1db$ and Settling Time$< 0.25s$ is also shown in Fig. 16.

3. PID Controller Examples

In this section, we consider two examples for PID controller design.

Example III.9 Consider the plant

$$P(z) = \frac{z^3 + 0.25z^2 + 0.5z}{z^3 + 10.7z^2 + 10z + 0.5}$$

The sampling time, $T = 0.01s$. The GUI for the example is shown in Fig. 18. On entering the numerator, denominator and sampling time, a range of K_3 values is obtained for which the plant can be stabilized. For this example, the entire range of scanning i.e.from -50 to 50 is a valid range. On selecting a particular value of K_3, say $K_3 = -2$, the $K_1 - K_2$ set of stabilizing values is obtained. In this case, 2 distinct regions are obtained. This is because there are two *strings* satisfying the required

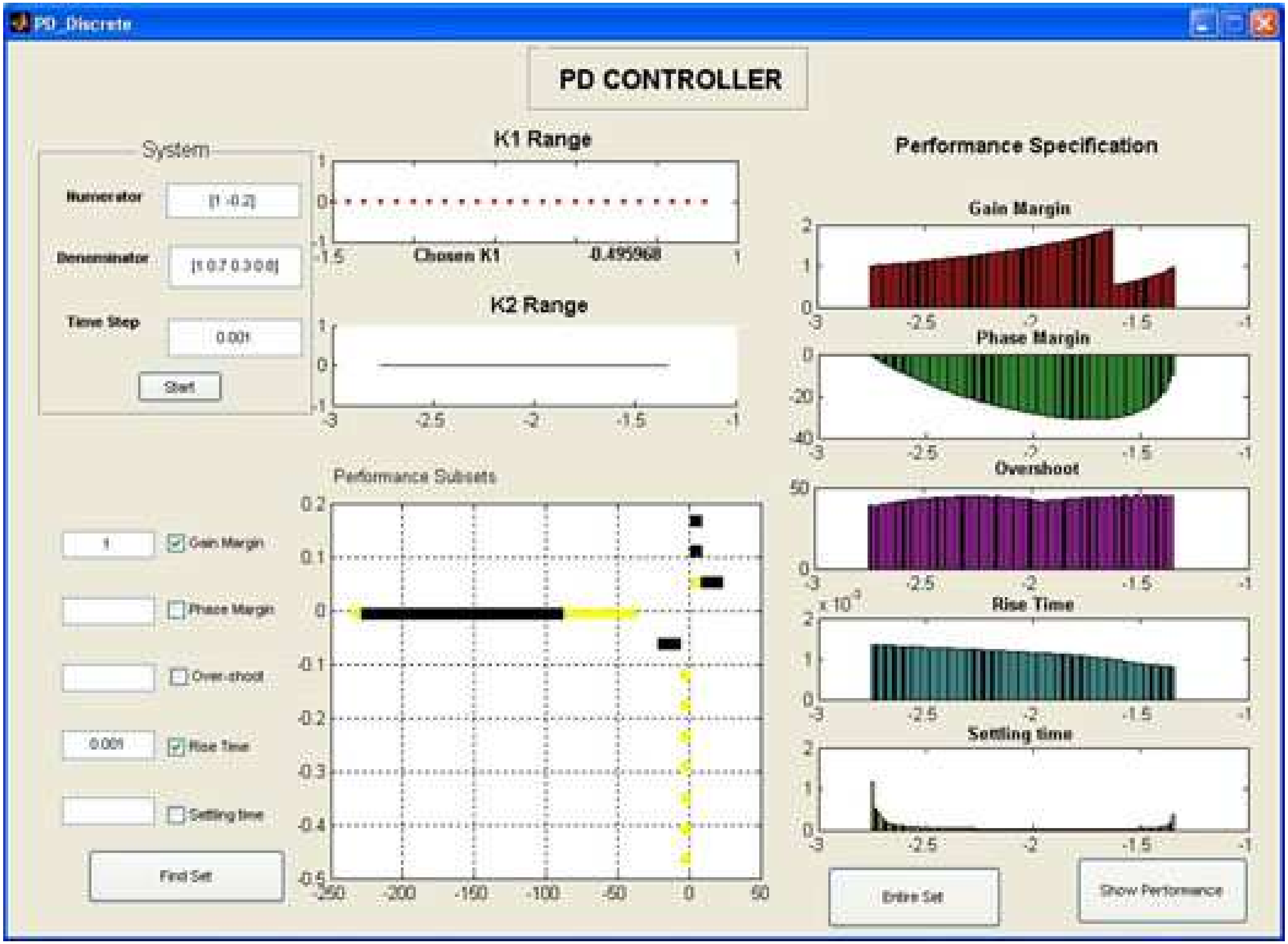

Fig. 14. GUI for PD controller design in example III.7

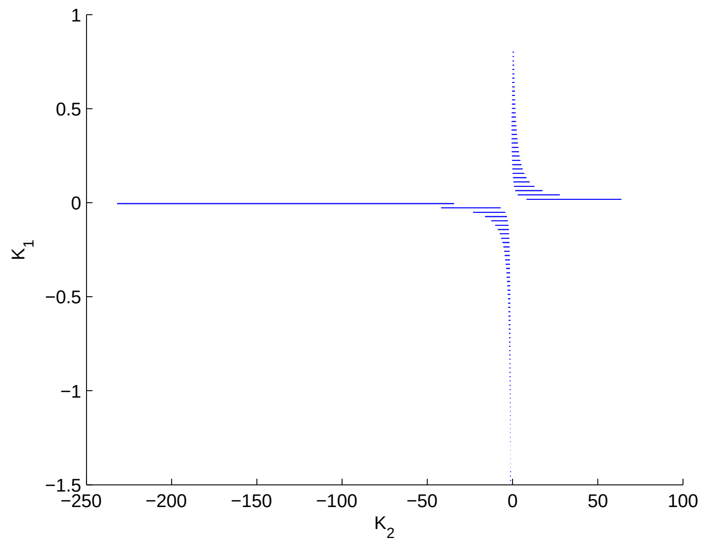

Fig. 15. The entire $K_1 - K_2$ set of stabilizing values for example III.7

signature. They are $[1,\ -1,\ -1]$ and $[-1,\ -1,\ 1]$. The $K_1 - K_2$ set is unbounded and is bounded by a square of length 1000 centered at origin. On selecting a particular value of $K_1 = -300$ and $K_2 = -250$ for $K_3 = -2$, various performance indices are displayed in the GUI. The step response for the above controller parameter values is shown in Fig. 19

The 3-D stabilizing set is shown in Fig. 20. The corresponding $K_p - K_i - K_d$ set is shown in Fig. 21. It is observed that 2 disjoint sets are formed which stabilizes the plant. Further, when some performance criteria are specified, say Phase Margin $> 30°$ and Settling time$< 0.4s$, the subset satisfying these conditions are obtained as shown in Fig. 22

Example III.10 Consider another example for design PID controllers. Let the plant be

$$P(z) = \frac{z^3 + 3z^2 - 2z + 1}{z^4 + 0.8z^3 + 0.6z^2 + 0.25z + 0.8}$$

The sampling time, $T = 0.001s$. The GUI is as shown in Fig. 23. On selecting $K_3 =$

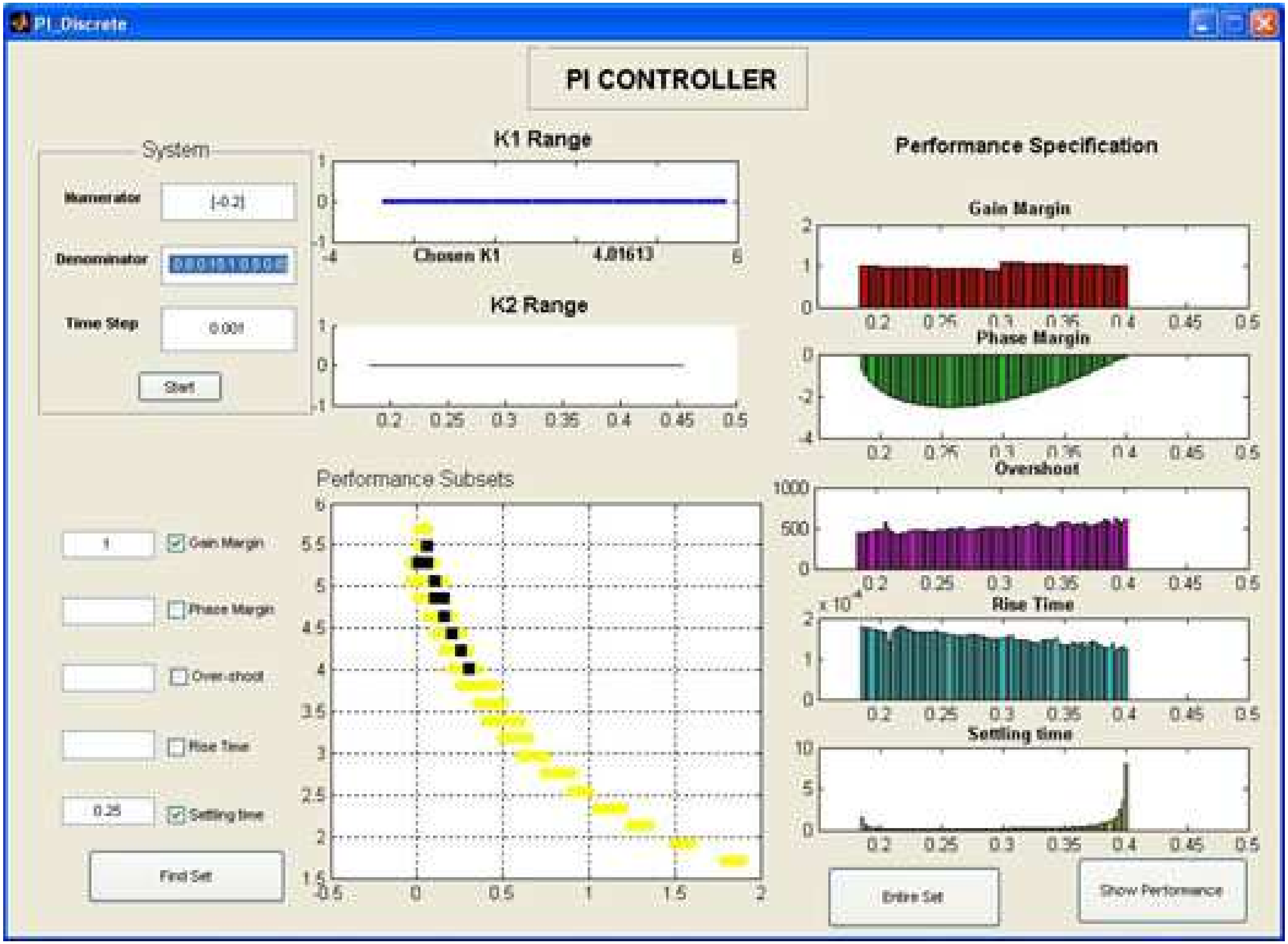

Fig. 16. GUI for PI controller design in example III.8

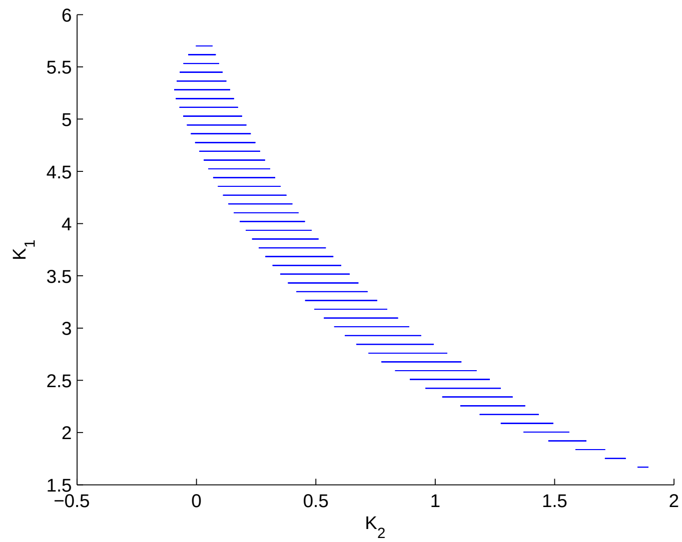

Fig. 17. The entire $K_1 - K_2$ set of stabilizing values for example III.8

-0.1, the $K_1 - K_2$ set is obtained. To find the subset for which Gain Margin$> 1db$, the data is fed to the GUI and the subset obtained is as shown in Fig. 24. If further constraint Rise Time$< 0.01s$ is imposed, the set shrinks as shown in Fig. 25. The set further shrinks and becomes two disjointed subsets when the condition Settling Time$< 0.16s$ is imposed as shown in Fig. 26. However even these 2 subsets are a subset of the previous set. This example shows that though the stabilizing set is continuous, the performance subsets can be disjoint.

These examples described above show various aspects of Controller design through the GUIs developed.

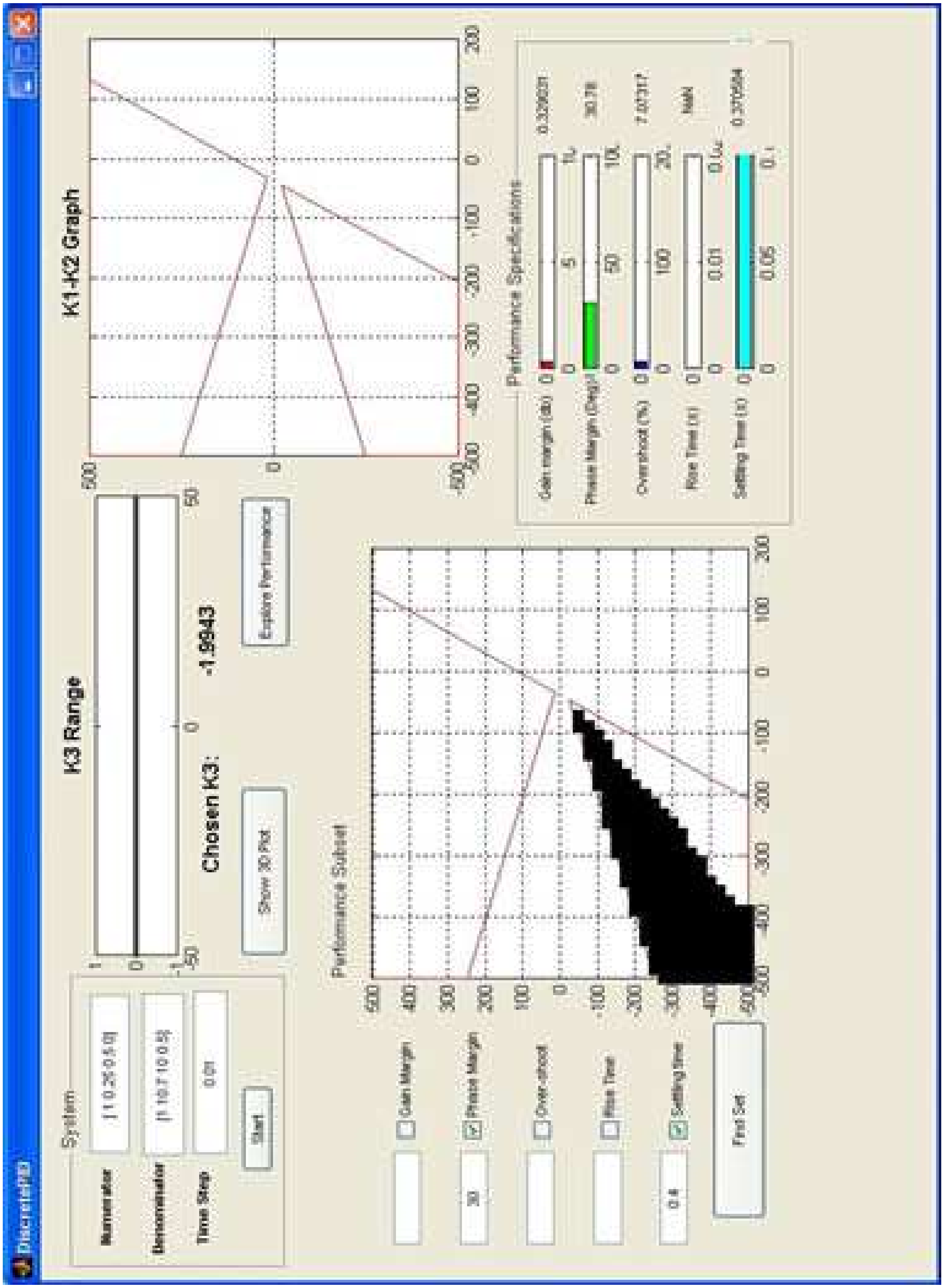

Fig. 18. GUI for PID controller design in example III.9

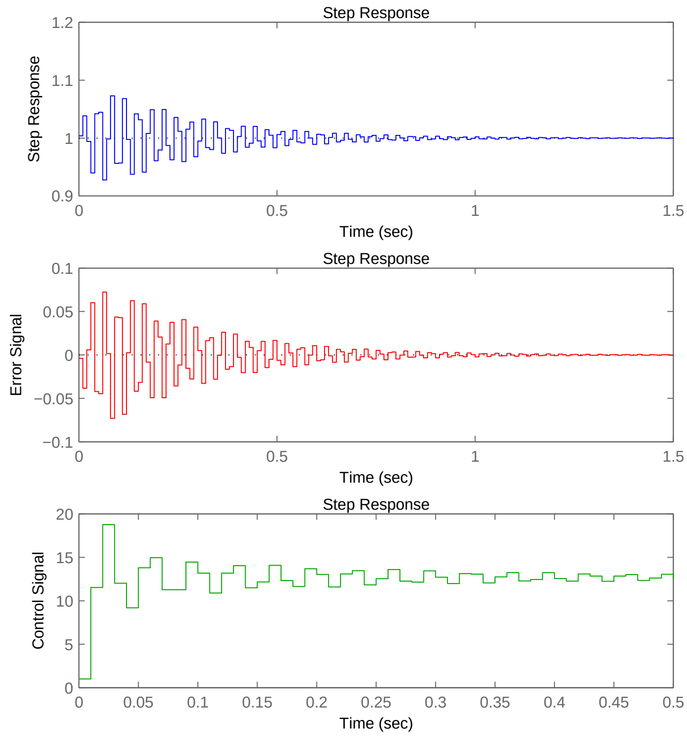

Fig. 19. Step response for $K_1 = -300, K_2 = -250$ and $K_3 = -2$ for the plant in example III.9

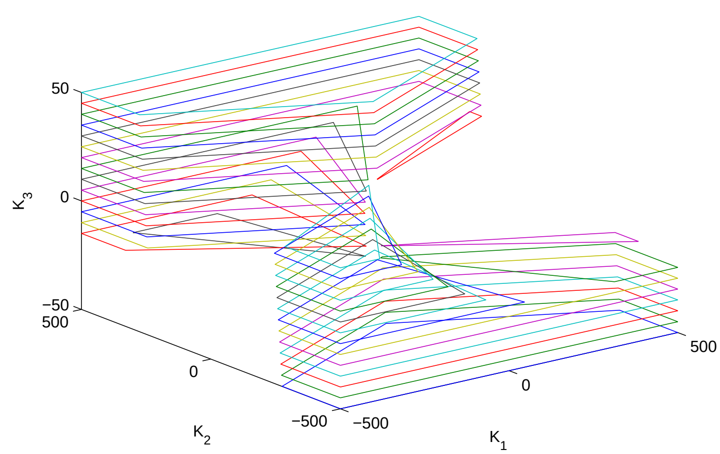

Fig. 20. 3D $K_1 - K_2 - K_3$ stabilizing set for the plant in example III.9

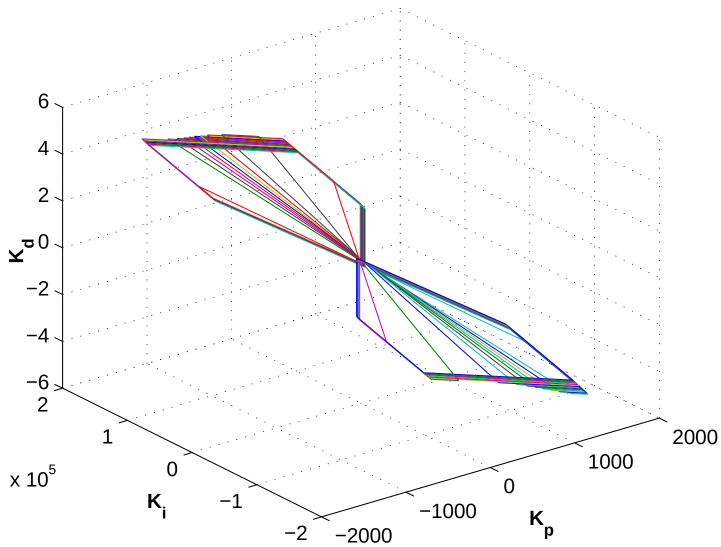

Fig. 21. 3D $K_p - K_i - K_d$ stabilizing set for the plant in example III.9

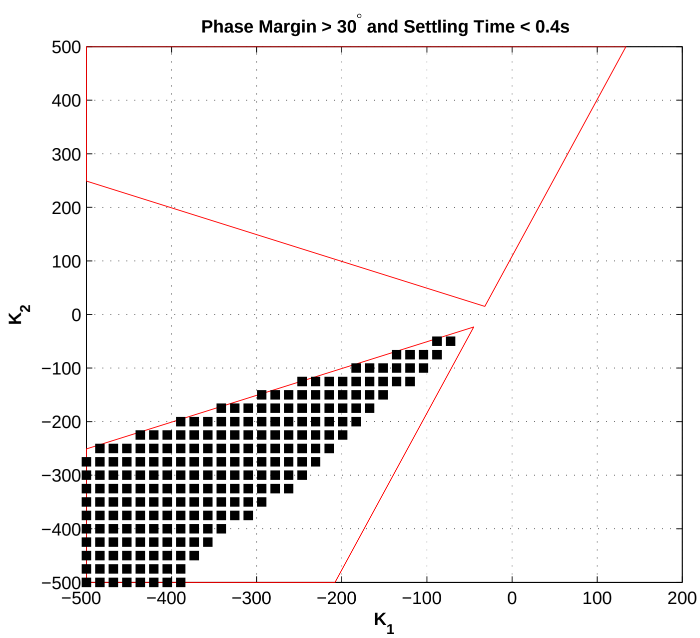

Fig. 22. Subset satisfying phase margin $> 30°$ and settling time$< 0.4s$ for the plant in example III.9

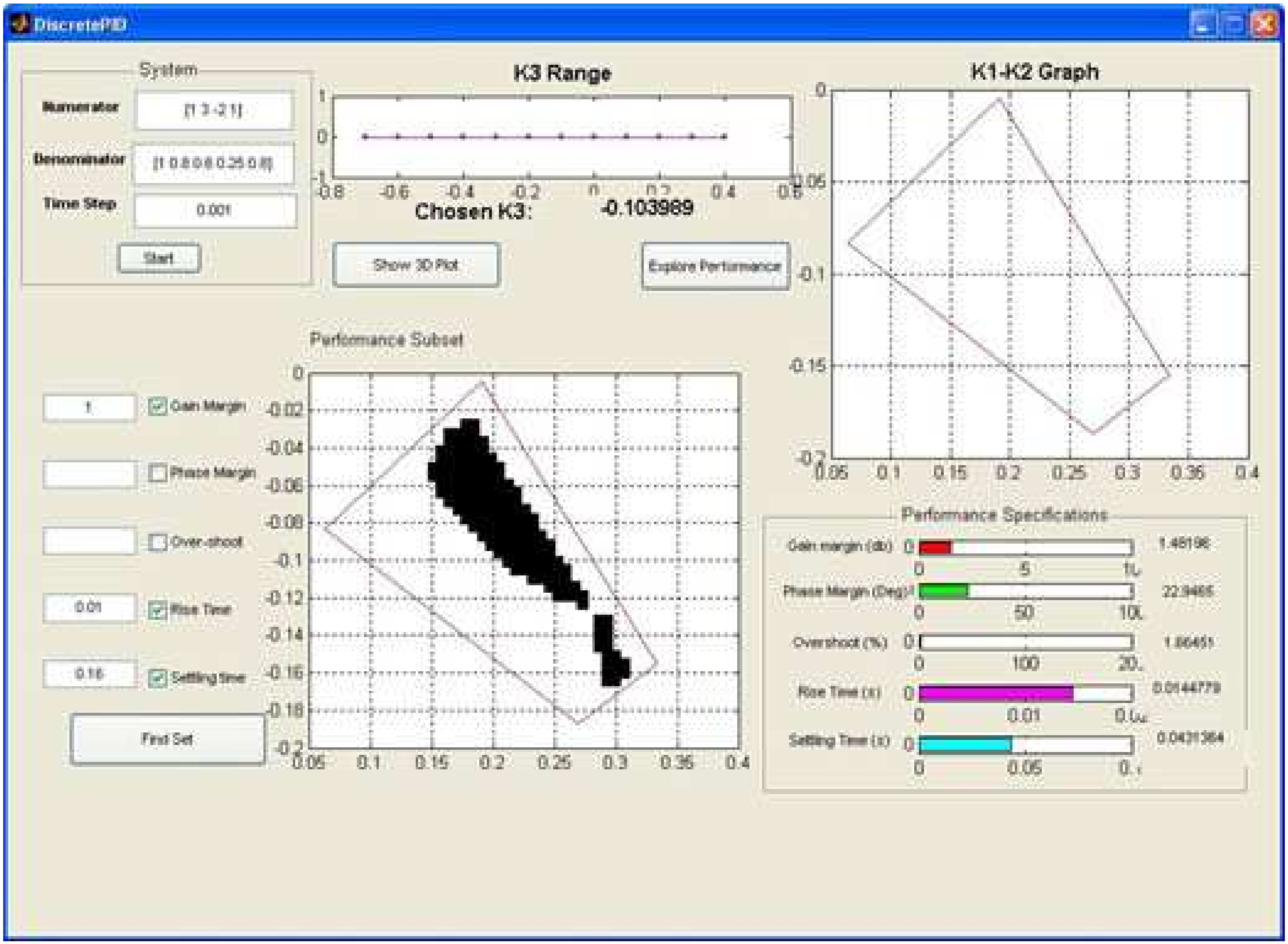

Fig. 23. GUI for PID controller design in example III.10

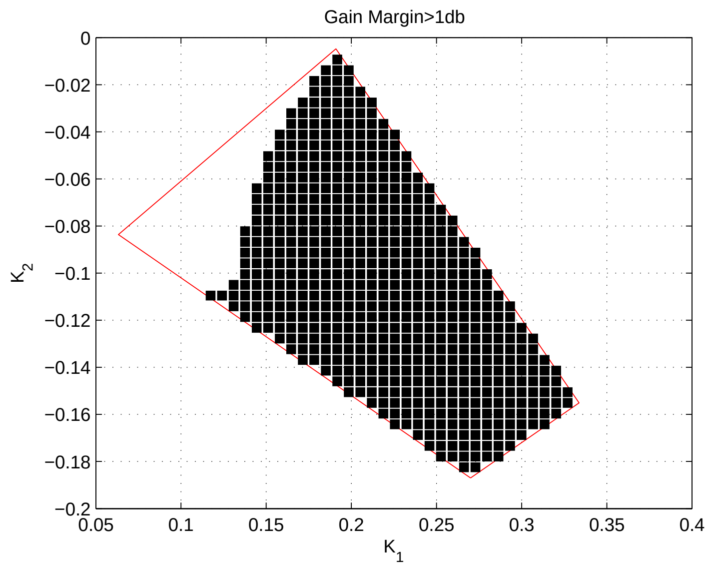

Fig. 24. Subset satisfying gain margin $> 1db$ for the plant in example III.10

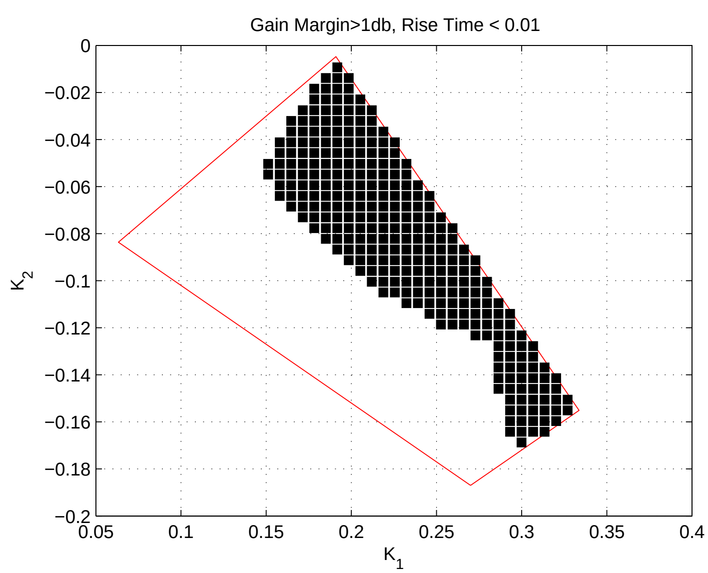

Fig. 25. Subset satisfying gain margin $> 1db$ and rise time$< 0.01s$ for the plant in example III.10

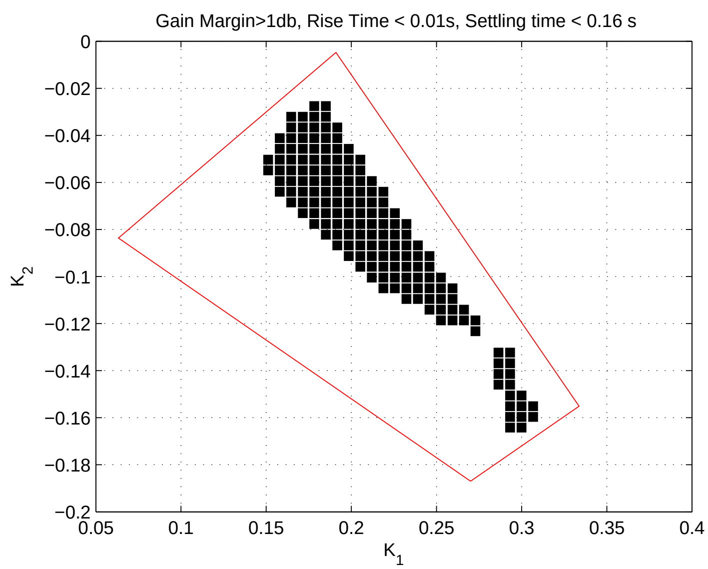

Fig. 26. Subset satisfying gain margin $> 1db$, rise time$< 0.01s$ and settling time$< 0.16s$ for the plant in example III.10

CHAPTER IV

DATA BASED DESIGN OF DISCRETE TIME PID CONTROLLERS *

A. Introduction

In model based methods, there has always been the problem of identifying the system before any design can be done. In this chapter, a new method of obtaining the set of controller parameters for the discrete-time PID controller is discussed which is based on directly on the input output data obtained from the plant. This method doesn't need any identification of the plant before the design is carried out.

First, the step response data of the system is obtained experimentally. Next, the Markov parameters are obtained by using simple transformations. The data is truncated to a certain number of samples and the z-transform of the output is approximated. The unit circle evaluation of the latter is obtained using the Tchebyshev representations as discussed in [18].

Now using the results described in [15], the set of PID controllers are obtained as linear inequalities in two variables while one of them is kept constant. By sweeping over the third parameter the entire stabilizing set can be obtained.

Various performance specifications like gain margin, phase margin and overshoot can be achieved on the sets thus obtained. The results are illustrated by examples. A GUI based program developed for this case has been discussed at the end of the chapter.

---

*Reprinted with permission from "Data based design of digital PID controllers" by S. Mitra, L.H. Keel and S.P. Bhattacharyya in *Proceedings of 2007 American Controls Conference*, New York, NY, Copyright[2007] by American Automatic Control Council

B. Formulation of the Problem

Consider the unit step response of a discrete time system. The output of the step signal can be described as

$$y_s[k] = [y_0, y_1, y_2, ..., y_k, ...] \tag{4.1}$$

where k is a positive integer. For a stable system, as k increases, the response attains a steady state value. Taking z-transform, (4.1) can be written as

$$Y_s(z) = y_0 + y_1 z^{-1} + y_2 z^{-2} + \cdots + y_k z^{-k} + \cdots \tag{4.2}$$

It is also known that

$$Y(z) = H(z)U(z) \tag{4.3}$$

where $Y(z)$ is the output of a plant described by transfer function $H(z)$ when an input $U(z)$ is applied. For a unit step input, $U(z) = z/(z-1)$, therefore the output $Y_s(z)$ can be written as

$$\begin{aligned} Y_s(z) &= H(z)U(z) \\ &= H(z)\left[\frac{z}{z-1}\right] \end{aligned} \tag{4.4}$$

From (4.2) and (4.4),

$$\begin{aligned} H(z) &= \frac{Y(z)}{U(z)} \\ &= Y(z)\left[\frac{z-1}{z}\right] \\ &= Y(z)\left(1 - z^{-1}\right) \\ &= \left(1 - z^{-1}\right)\left(y_0 + y_1 z^{-1} + y_2 z^{-2} + \cdots + y_k z^{-k} + \cdots\right) \end{aligned} \tag{4.5}$$

Also, it is known [20] that any plant $H(z)$ can be expressed in terms of its Markov parameters

$m_0, m_1, \cdots, m_k, \cdots$ as

$$H(z) = m_0 + m_1 z^{-1} + m_2 z^{-2} + \cdots + m_k z^{-k} + \cdots \tag{4.6}$$

Equating (4.5) and (4.6) we obtain

$$m_0 = y_0, \quad m_1 = y_1 - y_0, \quad m_2 = y_2 - y_1, \cdots, m_k = y_k - y_{k-1}, \cdots \tag{4.7}$$

Alternatively, the impulse response of a system can be obtained directly and can be approximated as the system transfer function [21]. Denoting the impulse response of a system as y_i,

$$y_i[k] = [m_0,\ m_1,\ m_2,\ \ \cdots,\ m_k,\ \ \cdots] \tag{4.8}$$

where k is a positive integer.

Note that the Markov parameters approach zero with increasing value of k, because the system is stable.

Taking the z-transform of the above sequence,

$$Y_i[z] = H(z) = m_0 + m_1 z^{-1} + m_2 z^{-2} + \cdots + m_k z^{-k} + \cdots. \tag{4.9}$$

Now, for both the cases, the plant transfer function $H(z)$ can be approximated by truncating the series up to n points. That is, $H(z) \approx P_n(z)$ where

$$\begin{aligned} P_n(z) &= m_0 + m_1 z^{-1} + m_2 z^{-2} + ... + m_n z^{-n} \\ &= \frac{m_0 z^n + m_1 z^{n-1} + ... + m_n}{z^n}. \end{aligned} \tag{4.10}$$

Lemma IV.1 The relative degree of a system, r, is the number of leading zeros in its impulse response.

Proof IV.1 On expanding any rational function as a power series of the variable, the first r non-zero term in the series is always the relative degree $(n-l)$ of the function where m is the degree of numerator and l is the degree of the denominator.

Now, consider that this plant is being stabilized by a PID controller. As shown in [15], the controller is defined as,

$$\begin{aligned} C(z) &= K_P + K_I T \frac{z}{z-1} + \frac{K_D}{T}\frac{z-1}{z} \\ &= \frac{K_2 z^2 + K_1 z + K_0}{z(z-1)} \end{aligned} \tag{4.11}$$

where

$$\begin{aligned} K_P &= -K_1 - 2K_0, \\ K_I &= \frac{K_0 + K_1 + K_2}{T} \\ K_D &= K_0 T. \end{aligned} \tag{4.12}$$

Let the plant $P_n(z)$ is given by

$$P_n(z) = \frac{N(z)}{D(z)}. \tag{4.13}$$

Rest of the analysis and the algorithm is similar to the one described in previous chapter and [15].

C. An Illustrative Example

1. Finding the Stabilizing Set

To illustrate with an example, take the step response of the system used in [15]. The step response of the system is

$$y_s[k] = [0,\ 0,\ 1,\ 1,\ 1.25,\ 1.25,\ 1.3125,\ 1.3125,\ 1.328125,\ 1.328125,\ 1.33203125, \cdots]$$

The sampling time $T = 0.001$sec. Using 4.5, The markov parameters of the system can be computed to be

$$m[k] = [0,\ 0,\ 1,\ 0,\ 0.25,\ 0,\ 0.0625,\ 0.015625,\ 0,\ 0.00390625, \cdots]$$

Consider data points up to $n = 3$. Writing the equivalent z-transform, we obtain

$$\begin{aligned} P_n(z) &= 0 + 0.z^{-1} + 1.z^{-2} + 0.z^{-3} \\ &= \frac{1}{z^2}. \end{aligned} \tag{4.14}$$

The signature of the above plant is

$$i_\delta + i_{Nr} - (l+1) = n + 2 + z_0 - l - 1 = r + z_0 + 1 = 2 + 0 + 1 = 3,$$

where the value of r is obtained using Lemma IV.1 and z_o is the number of zeros outside the unit circle for the approximated plant.

Now to obtain the stabilizing set for this plant, choose a particular value of K_3, say $K_3 = 1.2$. Then the real roots of $T(u, K_3)$ in $(-1, 1)$ are

$$-0.3618 \quad \text{and} \quad -0.1382.$$

Furthermore,

$$\text{sgn}[T(-1)] = 1.$$

Using the signature formula,

$$\begin{aligned} 3 &= \frac{1}{2}\text{sgn}[T(-1)]\Big(\text{sgn}[R(-1)] - 2\text{sgn}[R(-0.3618)] \\ &\quad +2\text{sgn}[R(-0.1382)] - \text{sgn}[R(+1)]\Big). \end{aligned}$$

Here there is only one sequence of signs satisfying the above equation.

$$\text{sgn}[R(-1)] = 1$$

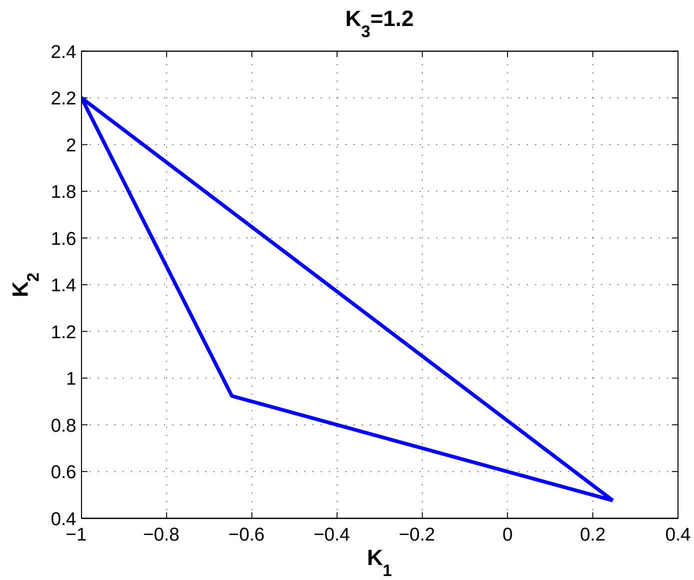

Fig. 27. The stabilizing set in K_1-K_2 space when K_3=1.2

$$\begin{aligned}\text{sgn}[R(-0.3618)] &= -1\\ \text{sgn}[R(-0.1382)] &= 1\\ \text{sgn}[R(+1)] &= -1\end{aligned}$$

so that

$$2(i_1 - i_2) = 6.$$

From this sequence, the following inequalities are obtained.

$$\begin{aligned}K_1 + 2K_2 &> 1.2\\ K_1 + 0.7236K_2 &< 0.5919\\ K_1 + 0.2674K_2 &> -0.3913\\ K_1 - 2K_2 &< 0.8\end{aligned}$$

Solving these inequalities, the stabilizing region in K_1-K_2 space is obtained as shown in Fig. 27. As K_3 is varied, the entire set is obtained as shown in Fig. 28.

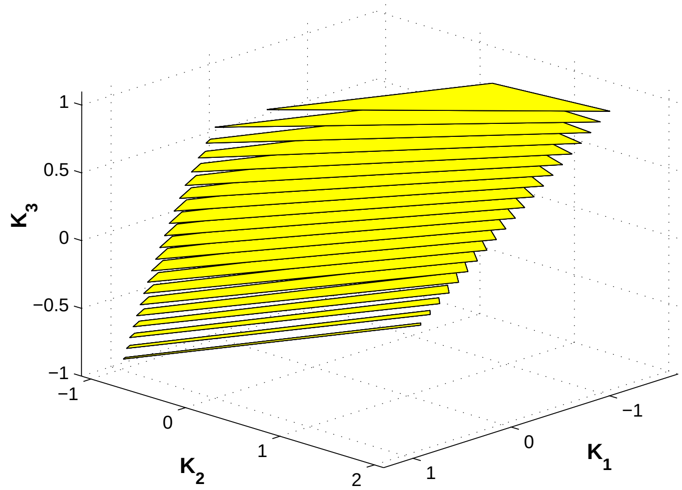

Fig. 28. The 3D stabilizing set for the given example

Continuing with $n = 5$, $n = 7$, $n = 10$, the respective stabilizing sets are obtained in a similar way. The results for all the system with $K_3 = 1.2$ is shown in Fig. 29.

It is observed that as n increases, the sets get closer and closer to the actual stabilizing region.It is also seen that the sets for $n = 10$ and actual system almost match each other. For more than 10 samples, for example 20 points, the area remains exactly the same. The thin line in Fig.29 is barely visible as it coincides with $n = 10$. This shows convergence of region with respect to number of terms in the Markov parameters. It is seen that a good choice of n is always after the step response has almost reached steady state value.

2. Set Satisfying Performance Requirements

The subsets achieving some performance specifications on the PID stabilizing sets obtained above can also be computed. In this case we consider gain margin and phase margin of the open loop and overshoot of the closed loop.

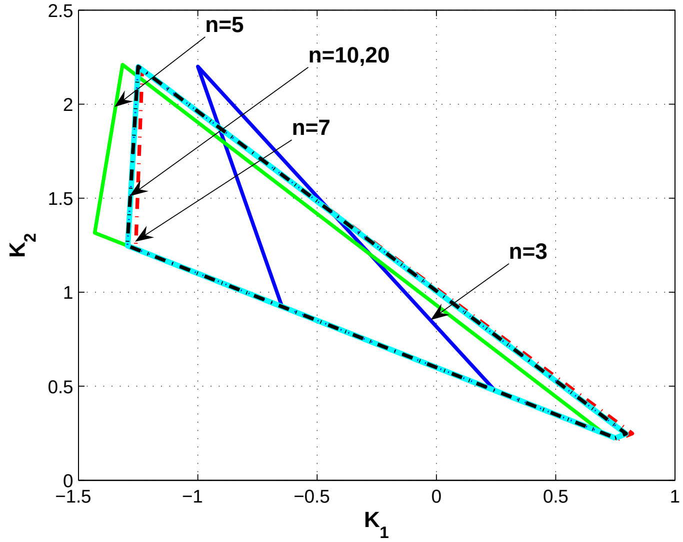

Fig. 29. Stabilizing set at $K_3 = 1.2$ for $n = 3, 5, 7, 10$ and actual set

For the gain margin and the phase margin, the approximated open loop system $G_n(z)$ is

$$G_n(z) = C(z)P_n(z). \tag{4.15}$$

From the frequency response of the above rational function, the gain and phase margin is obtained. To obtain the time response and hence the over-shoot, the close the loop with unity feedback is computed. The feedback system is

$$G_{CL}(z) = \frac{G_n(z)}{1 + G_n(z)}. \tag{4.16}$$

Next the step response of the above rational function is obtained and the overshoot is computed from this data. Fig. 30, Fig. 31 and Fig. 32 show the subsets achieving gain margin more than 1 db, phase margin more than 20 degrees and overshoot less than 100% corresponding to $n = 5$, $n = 7$ and $n = 10$ respectively.

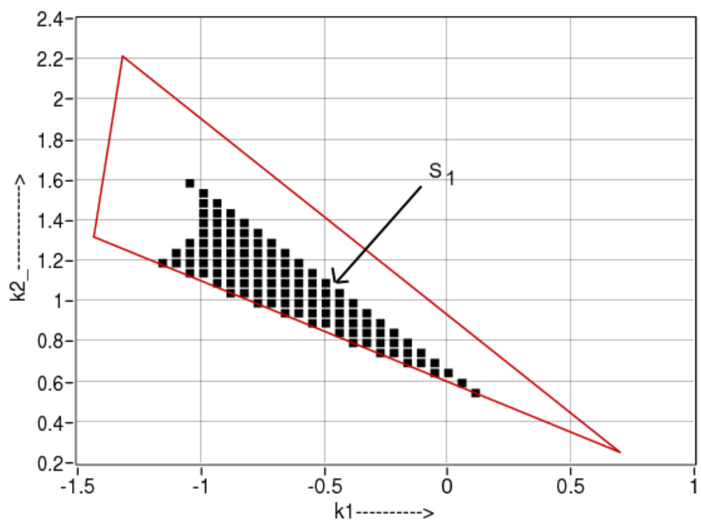

Fig. 30. The shaded region indicates a gain margin greater than $1db$, phase margin greater than 20 degrees and overshoot less than 100% for that region. This was obtained for approximation $n = 5$

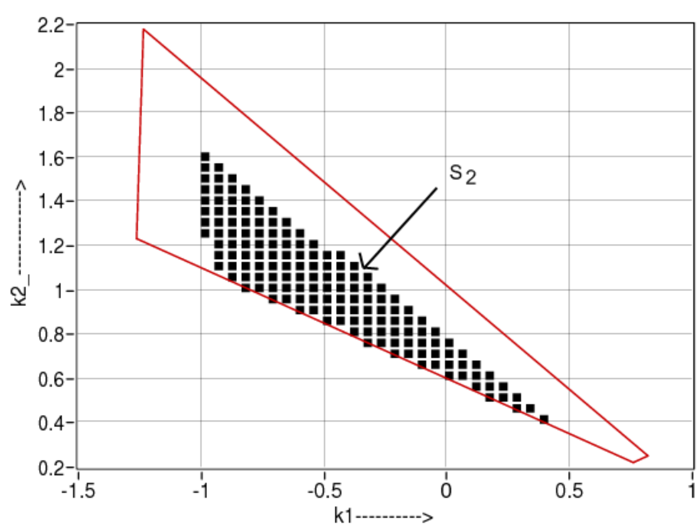

Fig. 31. The shaded region indicates a gain margin greater than $1db$, phase margin greater than 20 degrees and overshoot less than 100% for that region. This was obtained for approximation $n = 7$

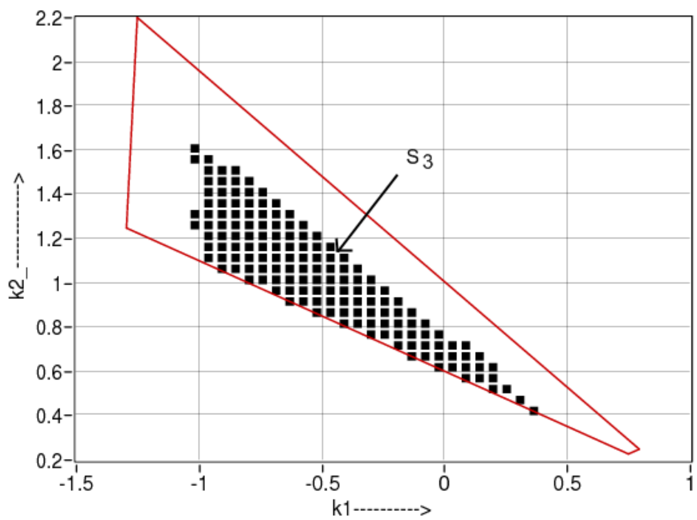

Fig. 32. The shaded region indicates a gain margin greater than $1db$, phase margin greater than 20 degrees and overshoot less than 100% for that region. This was obtained for approximation $n = 10$

D. GUI for Data Based Discrete Time PID Controllers

The GUI for data based design is very similar to the GUI for model based PID Controller design. Here, instead of the numerator and denominator, the step response data of the plant is the input as a text file. Once the user uploads the step response data, the GUI displays the step response as a plot. The user can select the number of points after which the data is to be truncated. Once the number of data points and the sampling time is given, the GUI gives a feasible set of K_3 values for which valid $K_1 - K_2$ set may exist. On selecting a particular K_3, the $K_1 - K_2$ stabilizing set is obtained. As in model based GUI, the performance specifications can be obtained in two different ways. This is explained below with the help of the following example.

Example IV.1 Consider the step response data of the plant as shown in Fig. 33. The GUI is as shown in Fig. 34.

On entering the number of points as 10 and sampling time of $0.001s$, the set of

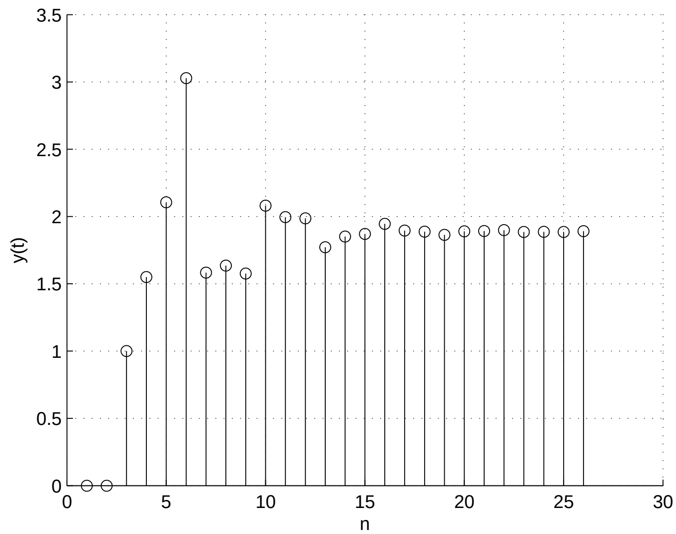

Fig. 33. Data for the plant in example IV.1

valid K_3 is displayed. On selecting $K_3 = 0.2$, the corresponding $K_1 - K_2$ stabilizingset is obtained. On selecting $K_1 = 0.0$ and $K_2 = 0.2$, various time and frequency performance parameters like gain margin, rise time, overshoot etc. are displayed for this chosen controller. The step response, error signal and control signal for this system is also obtained as shown in Fig. 35.

Further on specifying some performance objectives, say Gain Margin $> 1db$,Phase Margin$> 30°$ and Overshoot$< 50\%$, the subset achieving these performances are as shown in Fig. 36.

The 3-D stabilizing set is shown in Fig. 37. The corresponding $K_p - K_i - K_d$ set is shown in Fig. 38.

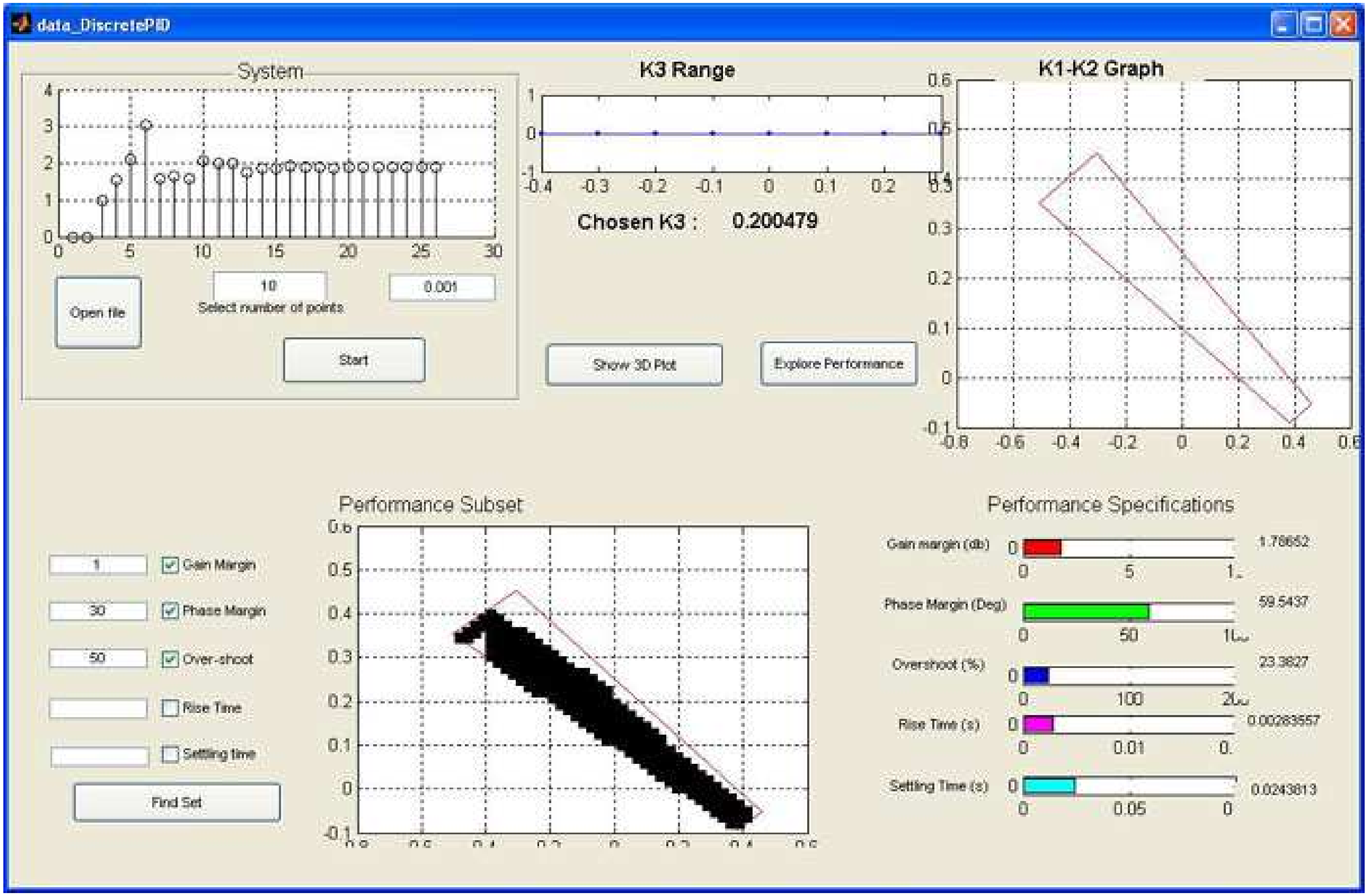

Fig. 34. GUI for data based PID controller design in example IV.1

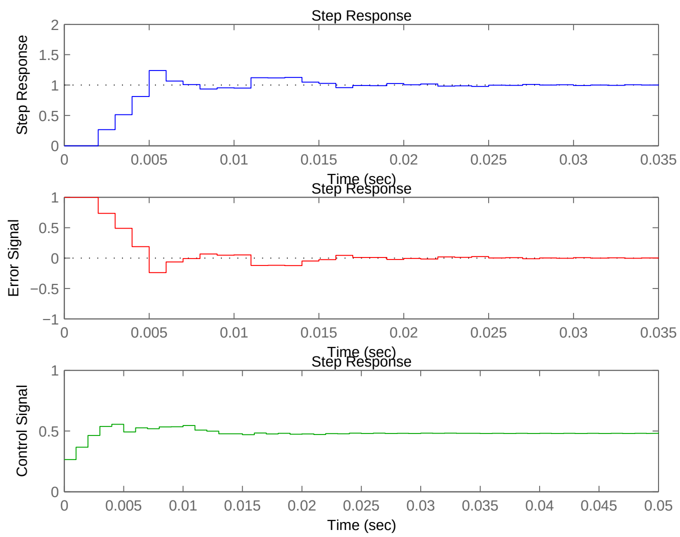

Fig. 35. Step response for $K_1 = 0.0$,$K_2 = 0.2$ and $K_3 = 0.2$ for the plant in example IV.1

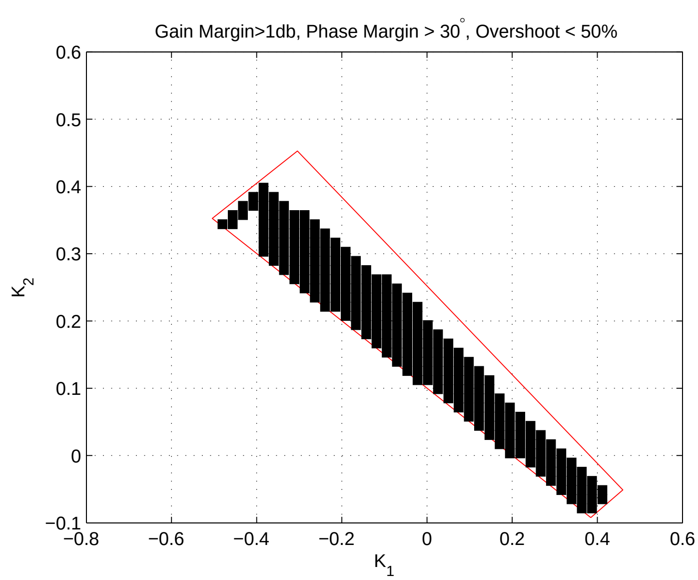

Fig. 36. Subset satisfying gain margin $> 1db$, phase margin$> 30°$ and overshoot$< 50\%$ for the plant in example IV.1

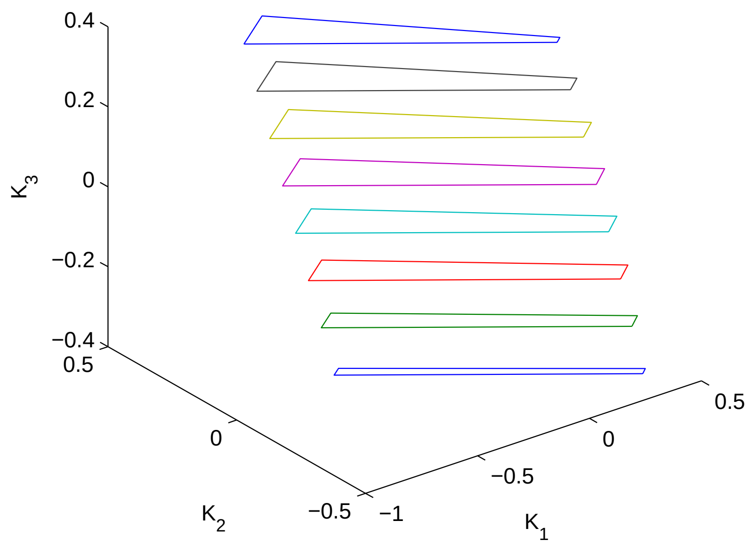

Fig. 37. 3D $K_1 - K_2 - K_3$ stabilizing set for the plant in example IV.1

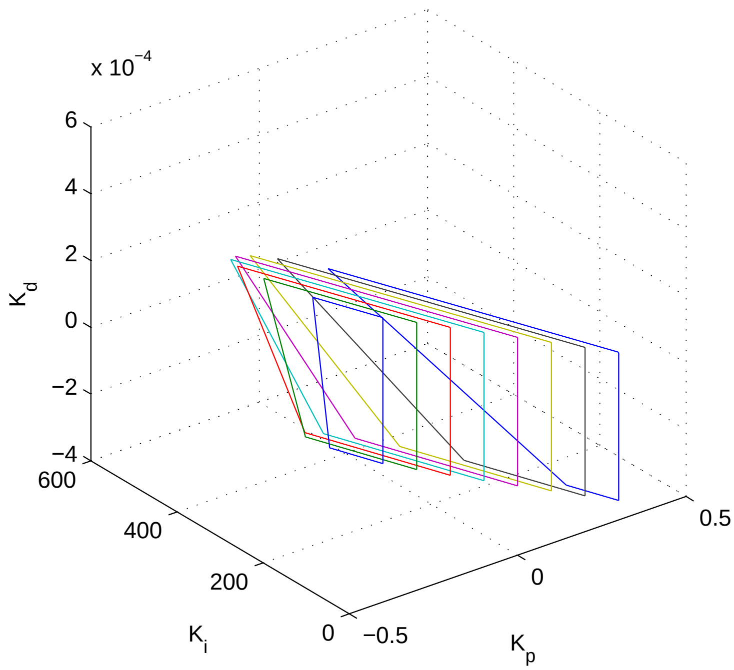

Fig. 38. 3D $K_p - K_i - K_d$ stabilizing set for the plant in example IV.1

CHAPTER V

ROBUST, DATA BASED DESIGN OF PID CONTROLLERS

A. Introduction

The model free part of classical control theory mainly rests on the Nyquist criterion and Bode plots. See [22] ,[23] . The Nyquist criterion predicts the closed loop stability based on the frequency response measurements made on open loop system. Bode and others developed a graphical approach to reshape the open loop frequency response by a simple cascaded compensator to achieve closed loop stability margins.

Recently a new approach is developed by Keel and Bhattacharyya [16] introduced a model free PID controller design which was based on frequency response data. In this section this theory is extended to robust stabilization for continuous time plants.

In this, the PID controller parameters are evaluated assuming some uncertainty in the measurement of the frequency response data. This robust stabilization problem under data uncertainty is transformed to a linear programming problem with interval coefficients. The set thus obtained will robustly stabilize the given set of plants. In the next few sections, first the necessary background regarding stabilization of a continuous time plant with just the frequency response data is provided. Later it is illustrated how this can be extended to robust stability with an example.

B. Background for Data Based Design

In this section the algorithm for finding the entire stabilizing set of PID controllers based on the frequency response data of the plant is described. Detailed theory can be obtained in [16] . It is assumed that the only information available to the designer is the frequency response of the plant $P(j\omega)$ for $\omega \in [0, \infty)$ when the plant is stable or the knowledge of a stabilizing controller $C(s)$ and the frequency response of the corresponding closed loop system $G(\jmath\omega)$ for $\omega \in [0, \infty)$.

Let

$$\begin{aligned} P(\jmath\omega) &= |P(\jmath\omega)|e^{\jmath\phi(\omega)} = P_r(\omega) + \jmath P_i(\omega) \\ &= |P(\jmath\omega)|\ cos(\phi) + \jmath|P(\jmath\omega)|\sin(\phi) \end{aligned} \tag{5.1}$$

where $|P(\jmath\omega)|$ denotes the *magnitude* and $\phi(\omega)$ the *phase* of the plant, at the frequency ω. Let the PID controller be of the form

$$C(s) = \frac{K_i + K_p s + K_d s^2}{s(1+sT)}, \quad T > 0 \tag{5.2}$$

where T is assumed to be fixed and small.

Also define

$$F(s) := s(1+sT) + \left(K_i + K_p s + K_d s^2\right) P(s).$$

and

$$\bar{F}(s) = F(s)P(-s).$$

Then

$$\begin{aligned} \bar{F}(\jmath\omega) &= F(\jmath\omega)P(-\jmath\omega) \\ &= \bar{F}_r\left(\omega, K_i, K_d\right) + \jmath\omega\bar{F}_i\left(\omega, K_p\right) \end{aligned}$$

The algorithm is as follows:

For stable systems: 0.1 Determine relative degree of the plant $r_P := n - m$ from the high frequency slope of the Bode magnitude plot of $P(\jmath\omega)$ where n and m

are degree of denominator and numerator of plant respectively.

$$r_P = -\frac{\text{High frequency slope (in dbs/decade)}}{20} \tag{5.3}$$

0.2 Let $\Delta_0^\infty[\phi(\omega)]$ denote the net change of phase in radians, of $P(\jmath\omega)$ for $\omega \in [0, \infty)$. Determine the number of right half plane zeros z^+ from

$$\Delta_0^\infty[\phi(\omega)] = -\left[r_P + 2z^+\right]\frac{\pi}{2} \tag{5.4}$$

For unstable systems: 0.1 Compute the frequency response $P(\jmath\omega)$ as

$$P(\jmath\omega) = \frac{G(\jmath\omega)}{C(\jmath\omega)(1 - G(\jmath\omega))} \tag{5.5}$$

0.2 Determine the relative degree of the plant r_P from the high frequency slope of the Bode magnitude plot of $P(\jmath\omega)$.

0.3 Determine the number of right half plane zeros and relative degree of the controller, z_c^+ and r_C respectively from $C(s)$.

0.4 Compute $\sigma(G)$ as

$$\sigma(G) = \frac{2}{\pi}\Delta_0^\infty \angle G(\jmath\omega). \tag{5.6}$$

0.5 Compute z^+ as

$$z^+ = \frac{1}{2}\left[-r_P - r_C - 2z_c^+ - \sigma(G)\right] \tag{5.7}$$

1. Fix $K_p = K_p^*$, solve

$$\begin{aligned} K_p^* &= -\frac{P_r(\omega) + \omega T P_i(\omega)}{|P(\jmath\omega)|^2} \\ &= -\frac{\cos\phi(\omega) + \omega T \sin\phi(\omega)}{|P(\jmath\omega)|} =: g(\omega) \end{aligned} \tag{5.8}$$

and let $\omega_1 < \omega_2 < \cdots < \omega_{l-1}$ denote the distinct frequencies of odd multiplicities which are solutions of (5.8).

2. Set $\omega_0 = 0$, $\omega_l = \infty$ and $j = \text{sgn}[\bar{F}_i(-\infty^-, K_p^*)]$. Determine all strings of integers $i_t \in \{+1, -1\}$ such that:

$$\left.\begin{array}{l} \text{For } n-m \text{ even :} \\ [i_0 - 2i_1 + \\ \cdots + (-1)^{i-1} 2i_{l-1} + (-1)^l i_l](-1)^{l-1} j \\ \text{For } n-m \text{ odd :} \\ [i_0 - 2i_1 + \\ \cdots + (-1)^{i-1} 2i_{l-1}](-1)^{l-1} j \end{array}\right\} = r_P + 2z^+ + 2. \tag{5.9}$$

3. For the fixed $K_p = K_p^*$ chosen in Step 1, solve for the stabilizing (K_i, K_d) values from

$$[\; K_i \; -K_d\omega_t^2 + \frac{\omega_t \sin\phi(\omega_t) - \omega_t^2 T \cos\phi(\omega_t)}{|P(\jmath\omega_t)|}] i_t > 0 \tag{5.10}$$

for $t = 0, 1, \cdots, l$.

4. Repeat the previous three steps by updating K_p over prescribed ranges. The ranges over which K_p must be swept is determined from the requirements that (5.9) are satisfied for at least one string of integers.

C. Data Robust PID Design

The motivation for robust design comes from the fact that in reality there is uncertainty in the measured data $P(\jmath\omega)$. Thus equation (5.8) and equation (5.10) have uncertainties. We can convert equation (5.10) into an inequality with interval coefficients [24] and can proceed towards the solution with the help of the following theorem.

Theorem V.1 Consider the interval inequality

$$[y - \mathbf{m}x - \mathbf{c}]\, i > 0 \tag{5.11}$$

where $\mathbf{m} \in [m^- m^+]$ and $\mathbf{c} \in [c^- c^+]$ are slope and intercept of the straight line equation above and are intervals varying from a minimum value to maximum value. $i = \{-1, 1\}$ such that $i = 1$ means $y > \mathbf{m}x + \mathbf{c}$ and $i = -1$ means $y < \mathbf{m}x + \mathbf{c}$. Then the region which will satisfy all the inequalities of (5.11) will be the intersection of the regions described by

$$\begin{aligned} \left[y - m^- x - c^-\right] i &> 0 \\ \left[y - m^- x - c^+\right] i &> 0 \\ \left[y - m^+ x - c^-\right] i &> 0 \\ \left[y - m^+ x - c^+\right] i &> 0 \end{aligned} \tag{5.12}$$

Proof: Consider the 'm-c' plane. The intervals of m and c form a rectangle in the above plane. Further, consider a fixed m. As can be seen from the Fig. 39 ,the area which satisfies all the inequalities $y > \mathbf{m}x + \mathbf{c}$ is bounded by c^+. Now let us vary m. It can be seen in figure that as m varies from m^- to m^+, the area described by (5.11) with $i = 1$ is bounded by the lines $y = m^- x + c^+$ and $y = m^+ x + c^+$. Similarly it can be shown that when $i = -1$, the inequality $y < \mathbf{m}x + \mathbf{c}$ is bounded by the lines $y = m^- x + c^-$ and $y = m^+ x + c^-$. This shows that it is enough to evaluate the inequalities at the vertices of the m-c rectangle in order to determine the solution that will satisfy the equation (5.11) .

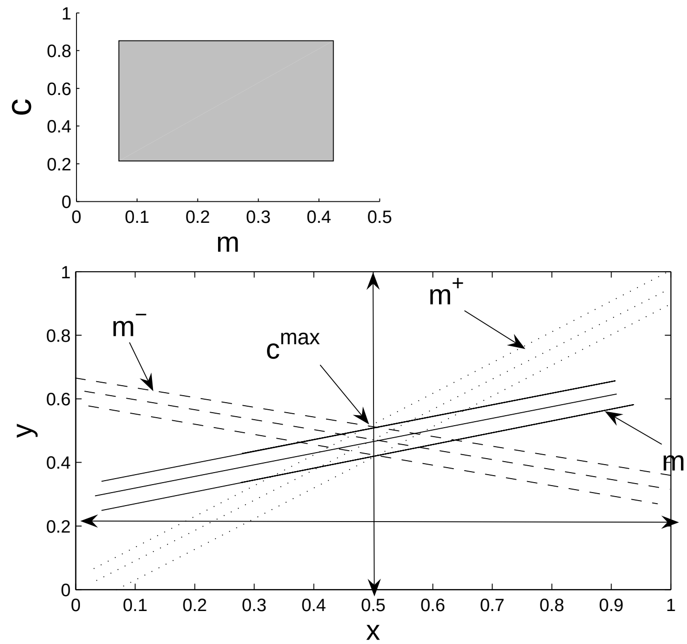

Fig. 39. Top:'m-c' plot. Bottom:'x-y' plot with varying slope m and intercept c

The objective now is to find the intervals for the equation (5.10), This is illustrated with the help of the following example.

Consider the frequency response $P(\jmath\omega)$ as shown in Fig.(40).For sake of simplicity, consider that the number of right half zeros of the plant is known to be 2 . Otherwise RHP poles can be determined along the equation (5.4) or equation (5.7).Let there be an uncertainty of $\pm 20\%$ around the real and imaginary part of the response . It is also assumed that there are no $\jmath\omega$ axis zeros in this example. If there are $\jmath\omega$ zeros in the plant, slightly perturb the plant can be slightly perturbed to get rid of them. If $\jmath\omega$ zeros are unavoidable, those terms can be lumped with the controller.

Denoting the maximum and minimum of the real and imaginary parts as $P_r^{max}(\jmath\omega)$, $P_r^{min}(\jmath\omega)$, $P_i^{max}(\jmath\omega)$ and $P_i^{min}(\jmath\omega)$ respectively. We now compute the $g(\jmath\omega)^{max}$ for

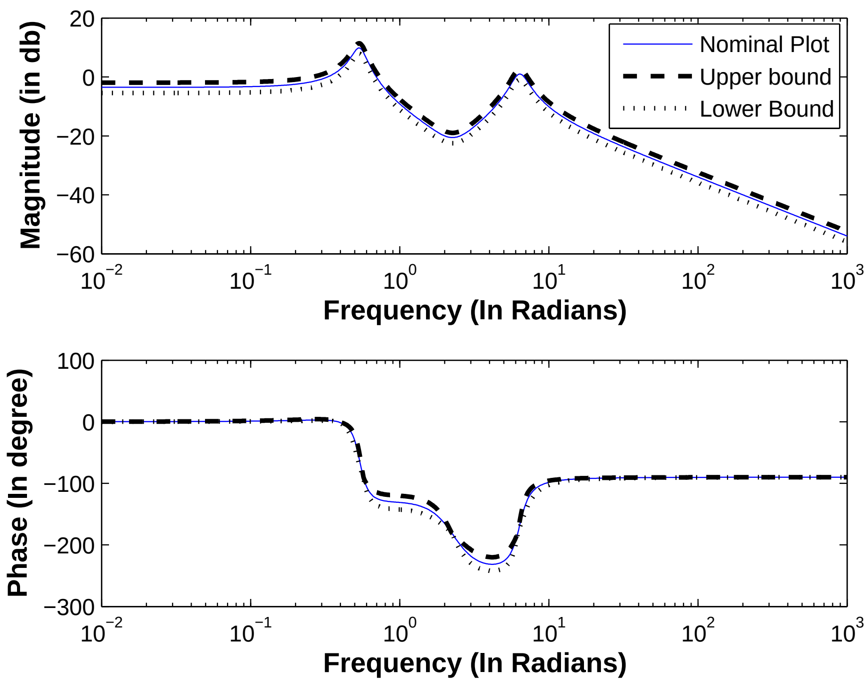

Fig. 40. Bode plot with uncertainty bound

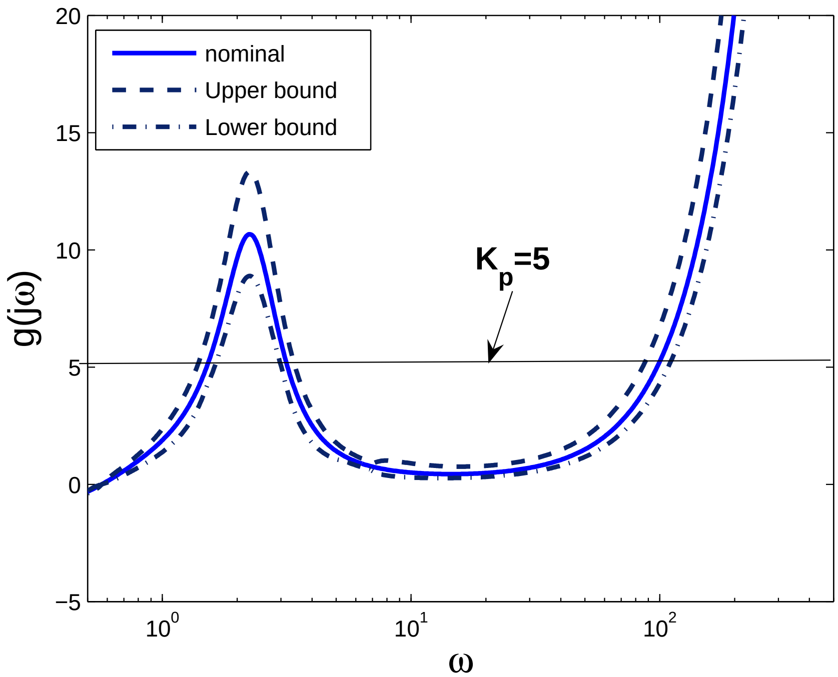

Fig. 41. $g(\jmath\omega)$-ω plot. The $K_p = 5$ line is also shown

the above data as follows.

$$\begin{aligned} g(\jmath\omega)^{max} &= \mathbf{max}[g(\omega, P_r^{max}(\jmath\omega), P_i^{max}(\jmath\omega)) \\ &, \quad g(\omega, P_r^{max}(\jmath\omega), P_i^{min}(\jmath\omega)) \\ &, \quad g(\omega, P_r^{min}(\jmath\omega), P_i^{max}(\jmath\omega)) \\ &, \quad g(\omega, P_r^{min}(\jmath\omega), P_i^{min}(\jmath\omega))] \end{aligned} \tag{5.13}$$

for $0 \leq \omega \leq \infty$ and $g(\jmath\omega)$ is evaluated from (5.8) where $T = 0.001s$ .Similarly $g(\jmath\omega)^{min}$ can also be obtained. These are shown in Fig. 41.

Now, the high frequency slope is -20db/decade. Therefore, from equation (5.3), $r_P = 1$. The right hand side of the equation (5.9) is given by

$$r_P + 2z^+ + 2 = 1 + 2.2 + 2 = 7 \tag{5.14}$$

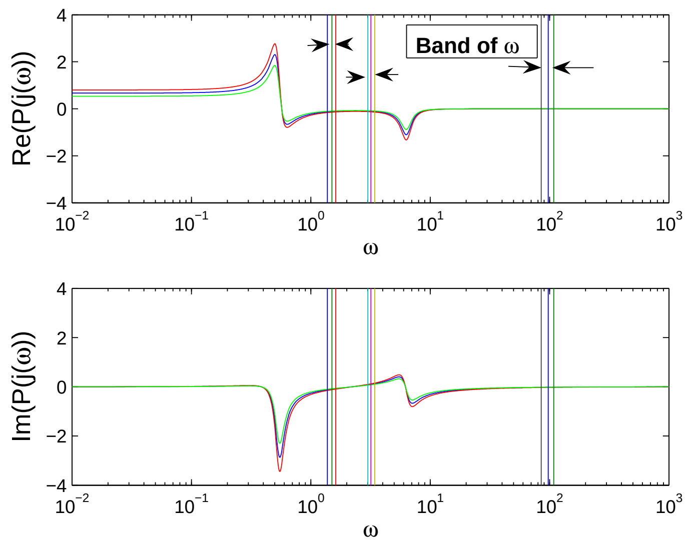

Fig. 42. 'P_r-ω' and 'P_i-ω' plot

where $z^+ = 2$ was obtained from the fact that the plant has 2 RHP zeros as mentioned above. It is observed that $K_p = 5$ cuts the $g(\jmath\omega)$ plot at 3 frequencies. Using equation (5.9) for the case where $n - m$ is odd, the only possible string satisfying the equation is

$$\mathcal{F} = \{i_0, i_1, i_2, i_3\} = \{1, \ -1, \ 1, \ -1\}. \tag{5.15}$$

Let the roots be ω_1, ω_2 and ω_3. From Fig. 41 it is observed that $\omega_t^- \leq \omega_t \leq \omega_t^+$ and for a fixed ω_t define

$P_r(\omega_t)^- := min(P_r(\omega_t))$, $P_r(\omega_t)^+ := max(P_r(\omega_t))$

$P_i(\omega_t)^- := min(P_i(\omega_t))$, $P_i(\omega_t)^+ := min(P_r(\omega_t))$.

These bounds are evaluated by finding the values from the 'P_r-ω' and 'P_r-ω' graphs as shown in Fig.42 and a zoomed version in Fig.43.

$$P_r(\omega_t)^{min} : \ = \min_{\omega_t^- \leq \omega_t \leq \omega_t^+} P_r(\omega_t)^-$$

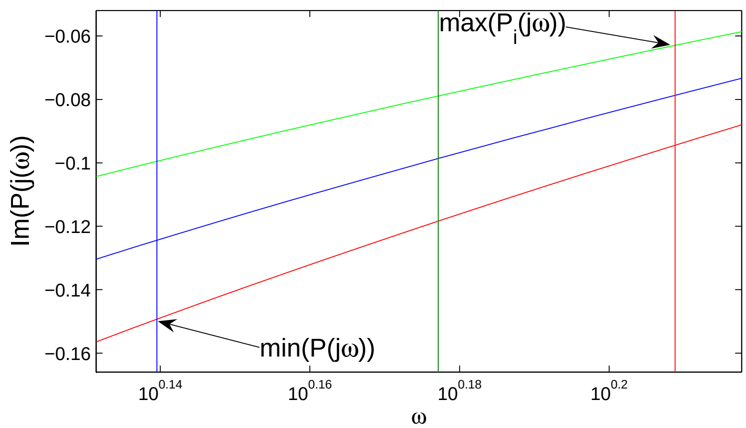

Fig. 43. Zoomed version of Figure (42) for ω_1

$$\begin{aligned} P_r(\omega_t)^{max} : &= \max_{\omega_t^- \le \omega_t \le \omega_t^+} P_r(\omega_t)^+ \\ P_i(\omega_t)^{min} : &= \min_{\omega_t^- \le \omega_t \le \omega_t^+} P_i(\omega_t)^- \\ P_i(\omega_t)^{max} : &= \max_{\omega_t^- \le \omega_t \le \omega_t^+} P_i(\omega_t)^+ \end{aligned} \tag{5.16}$$

Defining the constant coefficient, b_t of (5.10) as

$$b_t := \frac{-\omega_t P_i(\omega_t) + \omega_t^2 T P_r(\omega_t)}{|P(\jmath\omega_t)|} \tag{5.17}$$

Now comparing equation (5.11) with equation (5.10) and equation (5.17), it is observed that $\mathbf{m} = [(\omega_t^-)^2, (\omega_t^+)^2]$ and $\mathbf{c} = [b_t^-, b_t^+]$ where b_t^- and b_t^+ are the minimum and maximum values of b_t at w_t. These b_t^- and b_t^+ depend on the quadrant at which w_t is in the 'P_r-P_i' graph. This graph is shown in Fig.44.

For example if w_t lies in first quadrant i.e. $P_r > 0$ and $P_i > 0$, then

$$b_t^+ = \frac{-\omega_t^- P_i(\omega_t)^{min} + (\omega_t^2)^+ T P_r(\omega_t)^{max}}{|(P_r(\omega_t)^{min})^2 + (P_r(\omega_t)^{min})^2|}$$

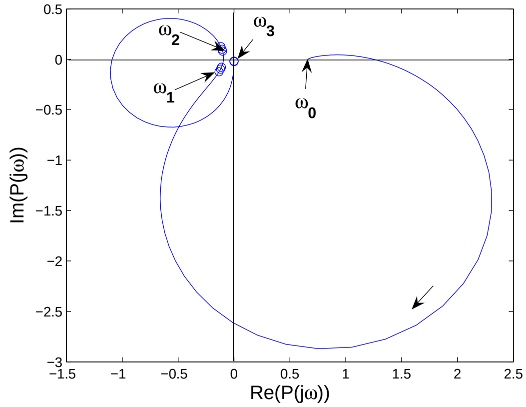

Fig. 44. Nyquist plot of the given plant

$$b_t^- = \frac{-\omega_t^+ P_i(\omega_t)^{max} + (\omega_t^2)^- T P_r(\omega_t)^{min}}{|(P_r(\omega_t)^{max})^2 + (P_r(\omega_t)^{max})^2|}. \tag{5.18}$$

Carrying out the calculations it is found that for this example,

$\omega_t^- = [0\ 1.3790\ 3.0052\ 85.1289]$

$\omega_t^+ = [0\ 1.6174\ 3.4297\ 108.3767]$

$b_t^- = [0\ 1.7982\ \ -47.6582\ 1574.0669]$

$b_t^+ = [0\ 20.1039\ \ -4.6160\ 13934.0711]$

Using equation (5.10) and theorem V.1, following inequalities are to be solved.

$$\begin{aligned}
\left(k_i - (\omega_t^-)^2 k_d - b_t^-\right) i_t &> 0 \\
\left(k_i - (\omega_t^-)^2 k_d - b_t^+\right) i_t &> 0 \\
\left(k_i - (\omega_t^+)^2 k_d - b_t^-\right) i_t &> 0 \\
\left(k_i - (\omega_t^+)^2 k_d - b_t^+\right) i_t &> 0
\end{aligned} \tag{5.19}$$

where $t = 0,\ 1,\ 2,\ 3$

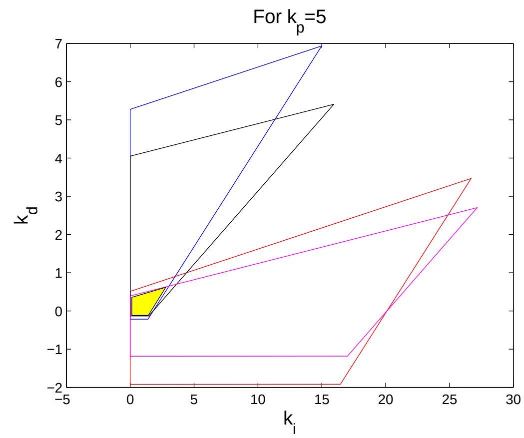

Fig. 45. 'K_i-K_d' plot. All possible inequality regions are shown. The shaded area is the intersection of all regions

Solving these inequalities for each of the four ω_t's, the region that robustly stabilizes the given plant is obtained for $K_p = 5$ as shown in Figure 45(shaded region).

CHAPTER VI

THE COMMERCIAL PRODUCT IN LABVIEW

A PID design software similar to the ones described in previous chapters was also developed in LabVIEW. This software has been productized by National Instruments and is released in their latest control systems toolkit version 3.0 along with LabVIEW 8.5. In this chapter a brief overview of the LabVIEW program designed called a 'VI' designed by them is presented.

A. Description of the VI

The VI is designed for PID controller design for discrete time systems. The name of the vi is: CD Design PID For Discrete Systems. The VI calculates the proportional gain K_p, integral gain K_i, and derivative gain K_d that stabilize the specified controller model(s). When a specified gain margin and/or phase margin specification is given, the vi finds out the controller parameter values satisfying these performance criteria. The VI is only for discrete single-input single-output (SISO) models. However, more than one SISO model at a time can be specified by using the two-dimensional transfer.

The main inputs and outputs of the VI are as follows. A detailed help is given in [25] **Model** specifies a state-space/transfer fucntion/ zero-pole representation of the controller model for which this VI calculates the PID gains.

Num K Grid Points indicates the number of points into which the VI grids the range of K_3. The default value is 50.

Num Search Points specifies the number of $K_1 - K_2$ sets of PID gain values to be chosen at each possibleK_3 value.The default value is 50.

Min Gain and Phase Margins denote the optional performance constraints on the PID controller model. If a value of 0 for either of these constraints, is specified, the VI considers that as no argument. A nonzero value for either of the parameters returns PID gain values that satisfy the performance constraint(s).

Design PID Gains returns the PID gain values that are closest to the centroid of all intersecting planes. If nonzero Min Gain and Phase Margins are specified, these gain values also satisfy those performance constraints.

Stable Set Boundary returns the boundary od stabilizing PID gain values. All gain values within the stable boundary are guaranteed to be stable for the PID controller. **Stable Set Interior Points** returns the sets of gain values that lie within the Stable Set Boundary. All gain values are guaranteed to meet any performance criteria specified by using the Min Gain and Phase Margins parameter.

For more than one transfer functions, the VI finds the set of K_3 values which is common for all the plants, and then finds the $K_1 - K_2$ polygon which is the intersection of the stabilizing set of each of the transfer functions.

B. Examples

Three examples are shown to illustrate different aspects of the VI.

Example VI.1 Consider a liquid level control system as described in [26]. The system gain is 0.001476. It has 2 zeros at -0.31461, -4.4523 and 3 poles at 1,0.5353 and 0.9512.Let the sampling time be $50ms$. The stabilizing set of PID values is shown in Fig. 46. The central point has the values $K_p = 5.1375$,$K_i = 57.1102$ and $K_d = 2.8565$.

Example VI.2 Consider a set of plants given by

$$P_1(z) = \frac{1}{z^2 - 0.25}$$

$$P_1(z) = \frac{1}{z^2 - 0.375}$$

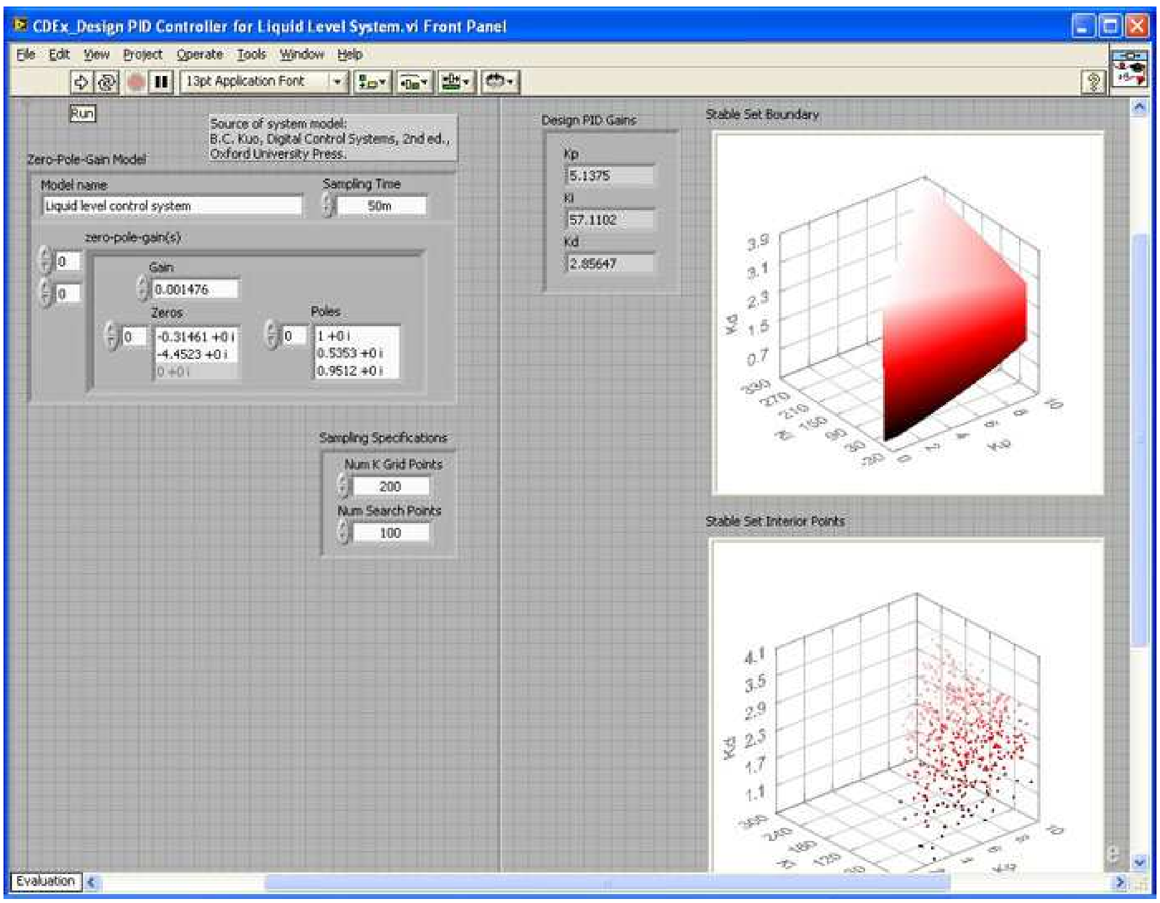

Fig. 46. LabVIEW VI showing results for example VI.1

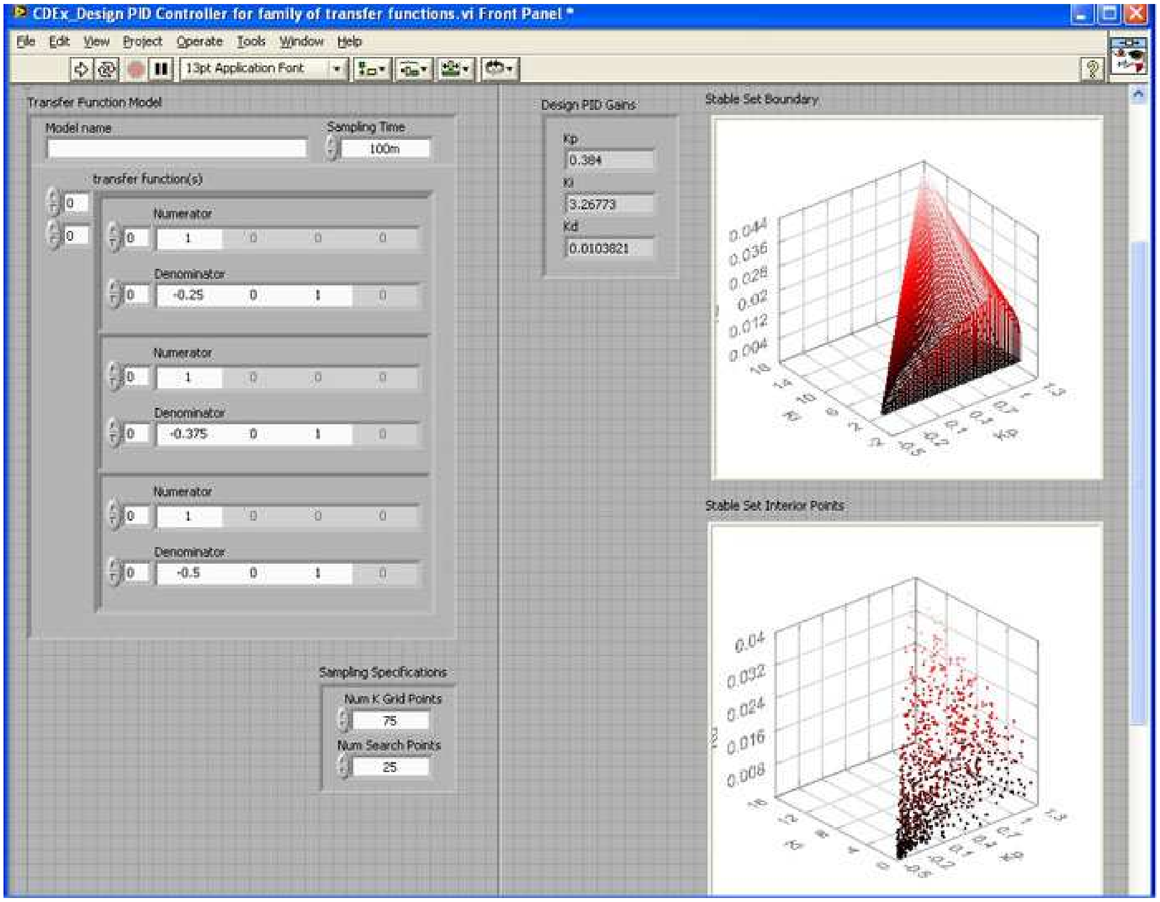

Fig. 47. LabVIEW VI showing results for example VI.2

and

$$P_1(z) = \frac{1}{z^2 - 0.5}$$

The sampling time is $100ms$. The set of stabilizing values for all the plants is as shown in Fig. 47.

Example VI.3 Consider a plant

$$P_1(z) = \frac{1}{z^2 - 0.25}$$

Let the sampling time be $100ms$. It is required to find the PIDvalues which have gain margin$> 1db$ and phase margin$> 10°$. The above data is input to the VI. The output

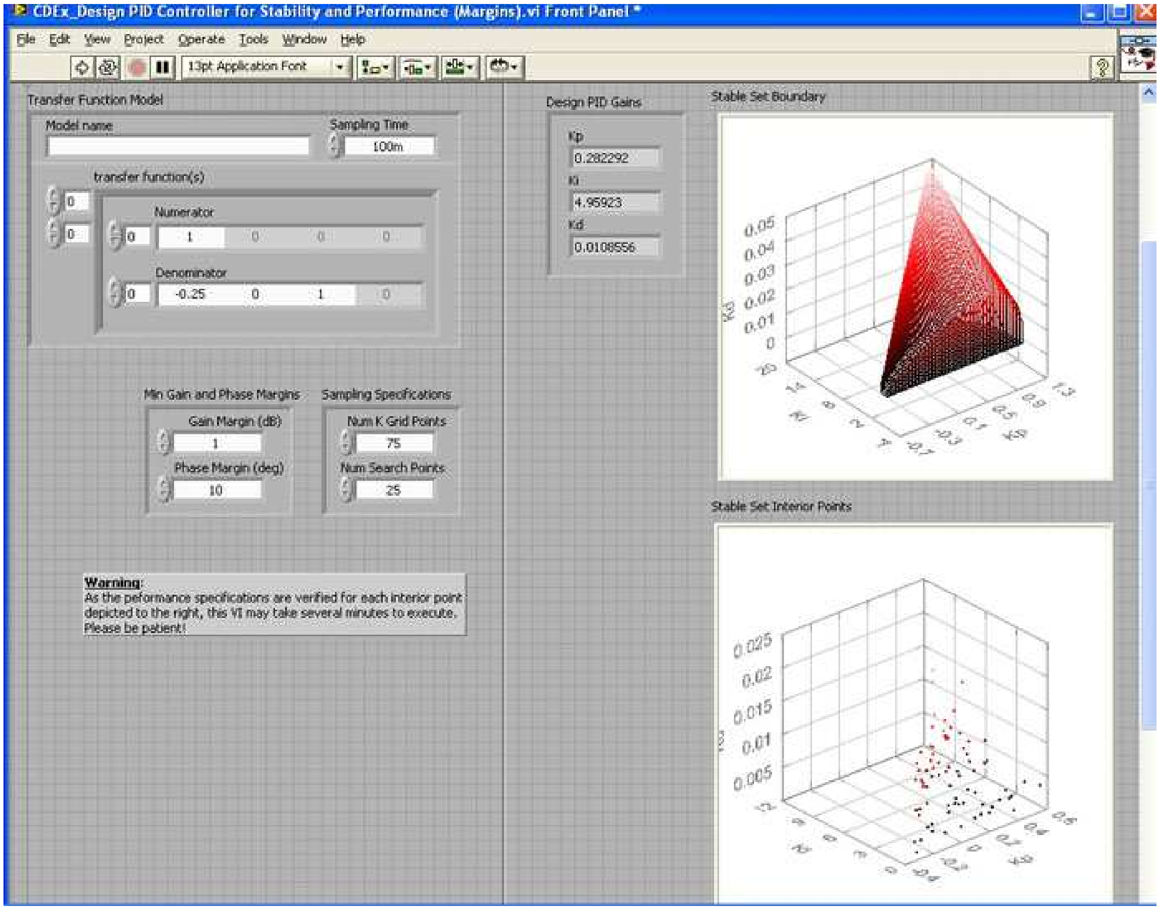

Fig. 48. LabVIEW VI showing results for example VI.3

is as shown in Fig. 48. The median point has the values $K_p = 0.2822$,$K_i = 4.9523$ and $K_d = 0.0108$.

CHAPTER VII

CONCLUSION AND FUTURE WORK

A. Summary

In this thesis, many aspects of synthesis and design of PID controllers have been discussed. The thesis presents computer aided design of discrete time controllers, data based design of discrete PID controllers and Data-robust design of PID controllers. These topics are very important in control system literature where identification of the plant and robust stability have always been important areas of research.

In the Computer Aided Design of discrete time controllers, a GUI based software is presented. Various ways of designing controllers based on performance has been introduced. The user can either explore the performance by interactively picking controllers one by one from the entire set and visualizing its performance or can specify some performance constraints and obtaining the resulting set.

In the data based design, a new way of designing PID Controllers based on input-output data has been developed. With this, the intermediate step of identification of model from data is removed saving considerable effort.the data required is step response data which is easier to obtain in case of discrete time system than frequency response data as introduced in [27]. Further, a GUI is developed for interactive design in this case too.

In data-robust design, the problem of uncertainty in measured data is considered. The design method developed finds the stabilizing set which can robustly stabilize the plant. It has been put forward as an application to interval linear programming.

B. Future Work

In future, a complete interactive menu driven software for general 3-term controller can be designed. The user will have option of choosing discrete time or continuous time system, model based or data based, delay or delay free design and a variety of controller structures to choose from. Some works has already been done in this area [28],[29]. Many new performance constraints can be included based on the need of the industry.

This software can be applied for automating the design in some practical scenario. By integrating this with hardware, a novel method can be developed which will take input as the data from the system or model provided by the user and output the most suitable value of the controller in runtime so that the new controller can be implemented instantly.

For data based design, the current method is only for stable systems. For the future, unstable system design can be considered. Robust design for data corrupted with noise for discrete time system also needs to be studied.Further, performance specification on the robustly stabilized data can also be looked upon.

Three term controllers are widely used in the industry. Any progress in this field will be well acknowledged both in the industry and the academia alike.

REFERENCES

[1] R.E. Kalman, "Contribution to the theory of optimal control," *Boletin de la Sociedad Mathematica Mexicana*, vol. 5, pp. 102–119, 1960.

[2] K. Astrom and T. Hagglund, *PID Controllers: Theory, Design, and Tuning*, Instrument Society of America, Research Triangle Park, NC, 1995.

[3] "Special Issue on PID 2006," *IEEE Control Systems Magazine*, February 2006.

[4] D. Shiokata S. Hara and T. Iwasaki, "Fixed order controller design via generalized kyp lemma," in *Proceedings of the IEEE Conference on Control Applications*, Taipei, Taiwan, September 2004.

[5] A. Hansson D. Henrion and R. Wallin, "Reduced lmis for fixed order polynomial controller design," in *Proceedings of the Symposium on Mathematical Theory of Networks and Systems (MTNS)*, Leuven, Belgium, July 2004.

[6] H.H. Huang W.M. Haddad and D.S. Bernstein, "Robust stability and performance via fixed-order dynamic compensation: the discrete time case," *IEEE Transactions on Automatic Control*, vol. 38, pp. 776–782, July 1993.

[7] T. Iwasaki and R.E. Skelton, "All fixed order h-infinity controllers: observer based structure and covariance bound," *IEEE Transactions on Automatic Control*, vol. 40, pp. 512–516, March 1995.

[8] P. Dorato, "Quantified multivariable polynomial inequalities: The mathematics of (almost) all practical design problems," in *Proceedings of the Sixth IEEE Mediterranean Conference on Control and Systems*, Alghero, Italy, June 1998.

[9] E.N. Gryazina and B.T. Polyak, "On the root invariant regions structure for linear systems," in *Proceedings of the 16th IFAC World Congress*, Prague, Czech, July 2005.

[10] Yu. I. Neimark, *Stability of Linearized Systems*, Leningrad Aeronautical Engineering Academy, Leningrad, Russia, 1949.

[11] D.D. Siljak, *Nonlinear Systems: The Parameter Analysis and Design*, Wiley, NY, 1969.

[12] E. Feron S. Boyd, L. El Ghaoui and V. Balakrishnan, "Linear matrix inequalities in system and control theory," in *SIAM*, Philadelphia, PA, 1994.

[13] Arthur Author and Joe Author, "Optimum settings for automatic controllers," *Trans. ASME*, vol. 64, pp. 759–768, 1942.

[14] A. Datta, M. Ho, and S.P. Bhattacharyya, *Structure and synthesis of PID Controllers*, Springer-Verlag, Berlin, Germany, 2000.

[15] J.I. Rego L.H. Keel and S.P. Bhattacharyya, "A new approach to digital PID controller design," *IEEE Transactions on Automatic Control*, vol. 64, pp. 687–692, April 2003.

[16] L.H. Keel and S.P. Bhattacharyya, "PID controller synthesis free of analytical models," in *Proceedings of the 16th IFAC World Congress*, Prague, Czech, July 2005.

[17] L.H. Keel S. Mitra and S.P. Bhattacharyya, "Data based design of digital PID controller," in *Proceedings of the 2006 American Control Conference*, New York, NY, July 2007.

[18] L.H. Keel and S.P. Bhattacharyya, "Root counting, phase unwrapping, stability and stabilization of discrete-time systems," *Linear Algebra and its Applications*, vol. 351–352, pp. 501–508, 2002.

[19] G. Szego G. Polya, *Problems and Theorems in Analysis II*, Springer, NY, 1974.

[20] J.-N. Juang, *Applied System Identification*, PTR Prentice Hall, Englewood Cliffs, NJ, 1994.

[21] J.G. Proakis and D.G. Manolakis, *Digital Signal Processing,Principles, Algorithms and Applications*, Prentice Hall, NJ, 3rd edition, 2006.

[22] H. Nyquist, "Regulation theory," *Bell System Technical Journal*, vol. 11, pp. 126–147, 1932.

[23] H.W. Bode, *Network Analysis and Feedback Amplifier Design*, Van Nostrand, Princeton, NJ, 1945.

[24] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis, with examples in parameter and state estimation, robust control and robotics*, Springer, Londres, 2001.

[25] LabVIEW Help, *CD Design PID For Discrete Systems (Control Design Toolkit) documentation*, National Instruments, Austin, Texas, 2007.

[26] B.C. Kuo, *Digital Control Systems*, Oxford University Press, New York, NY, 2nd edition, 1995.

[27] L.H. Keel and S.P. Bhattacharyya, "Data driven synthesis of three term digital controllers," in *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, June 2006.

[28] Bharat Narsimhan, "An automated virtual tool to compute the entire set of proportional integral derivative controllers for a continuous linear time invariant system," M.S. thesis, Texas A&M University, College Station, TX, December 2007.

[29] Indu Ramamurthi, "A versatile simulation tool for virtual implementation of proportional integral derivative (PID) controllers," M.S. thesis, Texas A&M University, College Station, TX, May 2007.

VITA

Name: Sandipan Mitra

Address: Prasanta Smriti, Kalibari Rd, Noapara, Barasat,
Kolkata - 700124, India

E-mail ID: sandipan.mitra@gmail.com

Education: Master of Science
Electrical Engineering, Dec 2007
Texas A&M University
College Station, Texas, USA

Bachelor of Engineering
Electrical and Electronics Engineering, June 2005
National Institute of Technology Karnataka, Surathkal,India

Experience: Software Developer
Infosys Technologies Limited, India
(August 2004 - July 2005)