COMPLEX ROOT LIMITS IN PID-CONTROLLED SYSTEMS: AN APPROACH USING D-DECOMPOSITION

A Thesis

by

HUNTER MATT WILLIAMS

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

MASTER OF SCIENCE

Chair of Committee,	Shankar P. Bhattacharyya
Committee Members,	Aniruddha Datta
	Swaroop Darbha
	Mehrdad Ehsani
Head of Department,	Miroslav Begovic

May 2020

Major Subject: Electrical Engineering

ABSTRACT

This thesis develops a new approach to the design of Proportional Integral Derivative (PID) controllers, solving an open problem in identifying feasible pole placement for controlled systems that allows for optimization of transient time-domain characteristics such as damping ratio and response oscillation. It also offers an alternative to previously developed methods for reducing overshoot and settling time of the step response. The method relies on the novel application of previously developed techniques.

Fundamentally, the method uses a variant of Neimark's D-Decomposition technique to place all closed-loop poles of a system in a specified region of the complex plane.

DEDICATION

To my family.

ACKNOWLEDGMENTS

Professor S. P. Bhattacharyya, my advisor and committee chair, has taught, guided, and supported me throughout my time at Texas A&M and he has my deepest appreciation and gratitude, and who provided me with an interesting and meaningful problem to address.

I would like to extend my sincere appreciation to my committee members: Dr. A. Datta, Dr. S. Darbha and Dr. M. Ehsani. They have provided patient support throughout my studies.

I would like to extend gratitude to the United States Military Academy under COL James Raftery for granting me the chance to obtain my graduate degree while on active duty, as well as the critical advice and guidance of Dr. Lisa Shay and MAJ David Sedivy during the initial application process. I would like to thank COL Christopher Korpela, LTC David Diaz, and LTC Jason Pieri for recommending me to the program.

My father, Matt Williams, mother, Ginger, and brother, Dillon have all given me their unconditional support.

My colleague and friend, Dr. Sangjin Han, provided me with invaluable insight when trying different means to approach the problem.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a thesis committee consisting of Professor Shankar P. Bhattacharyya, Professor Aniruddha Datta, and Professor Mehrdad Ehsani of the Department of Electrical and Computer Engineering and Professor Swaroop Darbha of the Department of Mechanical Engineering.

All work for the thesis was completed by the student, under the advisement of Professor Shankar P. Bhattacharyya of the Department of Electrical and Computer Engineering.

Funding Sources

The U.S. Army's Advanced Civilian Schooling program funded my course of study at Texas A&M.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

1. INTRODUCTION

The first requirement of any controller design is stability. For Proportional Integral (PI) and Proportional Integral Derivative (PID) controllers, there exists a method to obtain a set of all stabilizing controllers, the stabilizing set $\mathcal{S}$, represented as an enclosed region in the space of controller parameters. Further refinements to this method allow for the selection of regions within $\mathcal{S}$ that meet certain criteria beyond stability. One of these refinements, σ-Hurwitz stability, allows for specification of regions with closed-loop poles a certain distance away from the imaginary axis i.e. of placing limits on the real magnitude of closed-loop poles. This thesis develops a method to place a corresponding limit on the imaginary magnitude of a system's closed-loop poles with the goal of allowing the designer to limit response oscillation and increase damping ratio.

This chapter provides an introduction and summary of the stabilizing set method – also known as the "signature" method – for continuous time systems as well as an overview of currently available performance refinements taking advantage of this method. This includes a previously unpublished method for finding anti-σ-Hurwitz stability.

Chapter 2 explains the history of d-decomposition and develops the modifications necessary for our method. This is followed by a detailed example showing how to use d-decomposition to find a performance region, $\overline{\mathcal{S}}$, within the larger stabilizing region $\mathcal{S}$.

Chapter 3 develops the method outlined in Chapter 2 with further examples, including both a Proportional Integral (PI) and Proportional Integral Derivative (PID) controller for different plants. The Edge Theorem is used to prove the location of poles within the specified $\overline{\mathcal{S}}$. The use of non-constant limits on the imaginary magnitude of system poles is illustrated using the simpler PI example.

In Chapter 4, the applications of the method established in the previous chapter are examined. Specifically, the ability to limit imaginary root magnitudes allows the designer to reduce system's damping ratio and the frequency of oscillations in the step response.

Chapter 5 contains concluding remarks.

1.1 Stabilizing Set for Continuous Time Systems

The stabilizing set method produces a space of all possible stabilizing controllers $\mathcal{S}$ in the controller parameter space.

Consider the LTI continuous-time unity feedback system shown in 1.1.

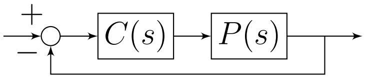

Figure 1.1: Unity feedback control loop.

$P(s)$ is the plant transfer function and $C(s)$ is the controller transfer function.

$$P(s) = \frac{N(s)}{D(s)}, C(s) = \frac{K_d s^2 + K_p s + K_i}{s}, \tag{1.1}$$

where $N(s)$ is of degree m and $D(s)$ is of degree n. Assume $N(s)$ and $D(s)$ are coprime – that is, have no common roots. This system has a closed-loop characteristic polynomial of

$$\delta(s) = sD(s) + (K_d s^2 + K_p s + K_i)N(s) \tag{1.2}$$

This method uses the Hurwitz criterion for stability, in which all roots of the characteristic polynomial must have negative real values. The stabilizing set $\mathcal{S}$ then, is the set of all Hurwitz-stable PID controllers for a given plant:

$$\mathcal{S} := \{(k_p, k_i, k_d) \colon \delta(s, k_p, k_i, k_d) \text{ is Hurwitz}\}\,. \tag{1.3}$$

To calculate $\mathcal{S}$, consider the complex plane $\mathbb{C}$. This is divided into the open left half-plane (LHP) $\mathbb{C}^-$ and the open right half-plane (RHP) $\mathbb{C}^+$. The net phase change of $\delta(j\omega)$ for $-\infty < \omega < \infty$ given real coefficients for $\delta(j\omega)$ is

$$\Delta_{\omega=0}^{\infty}\angle\delta(j\omega) = \frac{\pi}{2}(l-r). \tag{1.4}$$

where l and r are the roots of the characteristic equation located in the LHP and RHP, respectively.

$l - r$ is the *Hurwitz signature*, denoted as

$$\text{sig}(\delta) := l - r. \tag{1.5}$$

For stability, this must meet the condition

$$\text{sig}(\delta(s, k_p, k_i, k_d)) = n + 1,\ \ \forall(k_p, k_i, k_d)\ \in\ \mathcal{S}. \tag{1.6}$$

For the next step, controller gains must be separated into real and imaginary parts. To do this, write the modified polynomial

$$\nu(s) := \delta(s)N(-s). \tag{1.7}$$

Which will produce

$$\nu(s) := \nu_{even}(s^2, k_i, k_d) + sv_{odd}(s^2, k_p) \tag{1.8}$$

Assume $N(s)$ does not have roots on the imaginary axis. z^- and z^+ denote the zeros of $P(s)$ on the LHP and RHP, respectively. $\delta(s)$ is Hurwitz stable if and only if

$$\text{sig}(\nu) = n + 1 - z^- + z^+ \tag{1.9}$$

which becomes

$$\text{sig}(\nu) = n + 1 - m + 2z^{+}. \tag{1.10}$$

The stabilizing set $\mathcal{S}$ is thus described as

$$\mathcal{S} := \{(k_p, k_i, k_d)\text{: (1.21) is satisfied.}\}$$

With this, we can now apply the following steps to identify the complete stabilizing set $\mathcal{S}$

- Begin with

  $$\nu(s) = \nu_{even}(s^2) + s\nu_{odd}(s^2).$$

  which becomes

  $$q(\omega, K_p) := \text{Im } \nu(j\omega), \tag{1.11}$$

  $$p(\omega, K_i, K_d) := \text{Re } \nu(j\omega), \tag{1.12}$$

- Identify the range of k_p by finding k_p such that the number of roots $l-1$ meets the signature requirement at the extreme case:

  $$l - 1 \geq \begin{cases} \dfrac{n - m - 1 + 2z^{+}}{2}, & \text{if } n + m \text{ odd,} \\ \dfrac{n - m - 2 + 2z^{+}}{2}, & \text{if } n + m \text{ even.} \end{cases} \tag{1.13}$$

  This is best done by rearranging $q(w)$ as $k_p(\omega)$ and plotting $k_p(\omega)$.

- Fix $k_p = k_p^*$ and let $0 < \omega_1 < \omega_2 < \cdots < \omega_{l-1}$ denote the positive finite frequencies which are zeros of

$$\nu_{odd}(-\omega^2, k_p) = 0 \tag{1.14}$$

of odd multiplicities. $\omega_0 := 0$ and $\omega_l := \infty$.

- Find $j = \text{sgn}[\nu_{odd}(0^+, k_p^*)]$ and calculate sets of integers $i_0, i_1, \ldots$ such that

$$\begin{aligned} & n - m + 1 + 2z^+ \\ &= \begin{cases} j(i_0 + 2\sum\limits_{t=1}^{l-1}(-1)^t i_t), & \text{if } n+m \text{ odd,} \\ j(i_0 + 2\sum\limits_{t=1}^{l-1}(-1)^t i_t + (-1)^l i_l), & \text{if } n+m \text{ even.} \end{cases} \end{aligned} \tag{1.15}$$

- Let I_1, I_2, $\ldots$ denote distinct strings of $\{i_t\}_{t=0}^{t=l-1}$ or $\{i_t\}_{t=0}^{t=l}$ satisfying (1.15). Each string I_j produces a stabilizing set in (k_i, k_d) space at $k_p = k_p^*$. This is the intersection of linear inequalities

$$i_t \cdot \nu_{even}(-\omega_t^2, k_i, k_d) > 0 \tag{1.16}$$

$\forall t \in \{1, \ldots, l-1\}$ if $n+m$ is odd, or $\forall t \in \{1, \ldots, l\}$ if $n+m$ is even, where each $i_t \in \{-1, +1\}$.

- This intersection is either an empty set or a convex polygon, $S_j(k_p^*)$. The stabilizing set for a fixed k_p^* is the union of these sets:

$$S(k_p^*) = \bigcup_j S_j(k_p^*). \tag{1.17}$$

- The complete stabilizing set in (k_p, k_i, k_d) is then found by sweeping k_p over the range found above. Practically, this will be done in discrete intervals.

Applying this method to the plant and PI controller

$$P(s) = \frac{s-2}{s^2+4s+3}, C(s) = \frac{K_p s + K_i}{s} \tag{1.18}$$

which will be used in several succeeding examples produces the space in Fig 1.2, with the roots along the boundary of $\mathcal{S}$ show in Fig 1.3

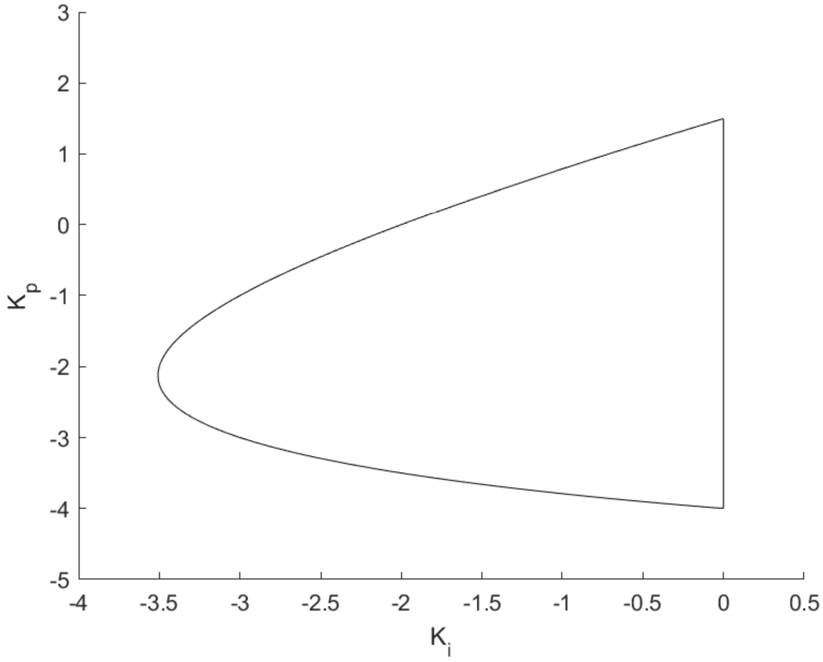

Figure 1.2: Stabilizing Set $\mathcal{S}$, PI System

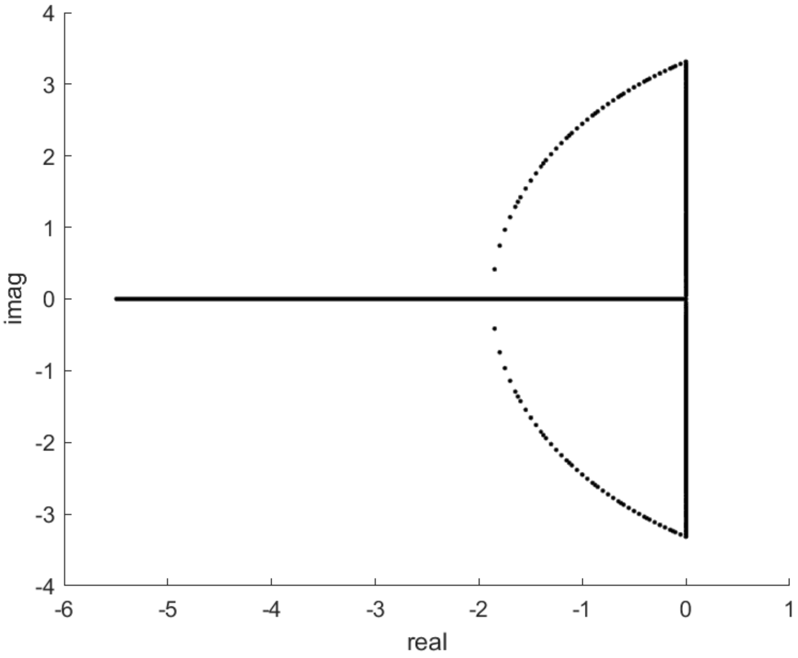

Figure 1.3: Poles along boundary of $\mathcal{S}$, PI System

1.2 Set of Anti-Stable Controllers

With slight modification, the stabilizing set method can be used to find A, the *anti-Hurwitz-stable* set. This concept is used extensively in this thesis in conjunction with the σ-Hurwitz criterion illustrated below.

To find the A, all that is necessary is to invert the signature equation:

$$\text{antisig}(\delta) := r - l. \tag{1.19}$$

Which for $\nu(s)$ becomes the new anti-stability condition

$$\text{sig}(\nu) = n + 1 + z^{-} - z^{+} \tag{1.20}$$

or

$$\mathrm{sig}(\nu) = n + 1 + m - 2z^{+}. \tag{1.21}$$

Then, proceed with the steps for calculating $\mathcal{S}$ as above. This will instead yield $\mathcal{A}$, the space of controllers which ensure that all closed-loop roots are in the RHP.

1.3 Gain, Phase, and H_∞ Margins

Although stability is the first requirement for controlled systems, in practice the ability to optimize for performance measures is also vital. In the years following the development of the stabilizing set method, additional refinements were made that allowed a designer to delineate subsets of $\mathcal{S}$ that met certain criteria.

Detailed explanations of these procedures are beyond the scope of this paper, although illustrative examples are given in the figures below. What is notable about each of these methods is that they rely on first finding $\mathcal{S}$, then intersecting $\mathcal{S}$ with some other region of the parameter space that may or will include unstable controller gain values in order to produce a final result that guarantees both the achievement of the specified margin *and* stability. Illustrations are given below for gain, phase, and H_∞ margins, along with iterated Nyquist plots demonstrating the achievement of the desired performance measures. All figures are original, although Figs 1.4-1.7 were derived using the methods in [1], and Figs 1.8 and 1.9 using the calculations in [2].

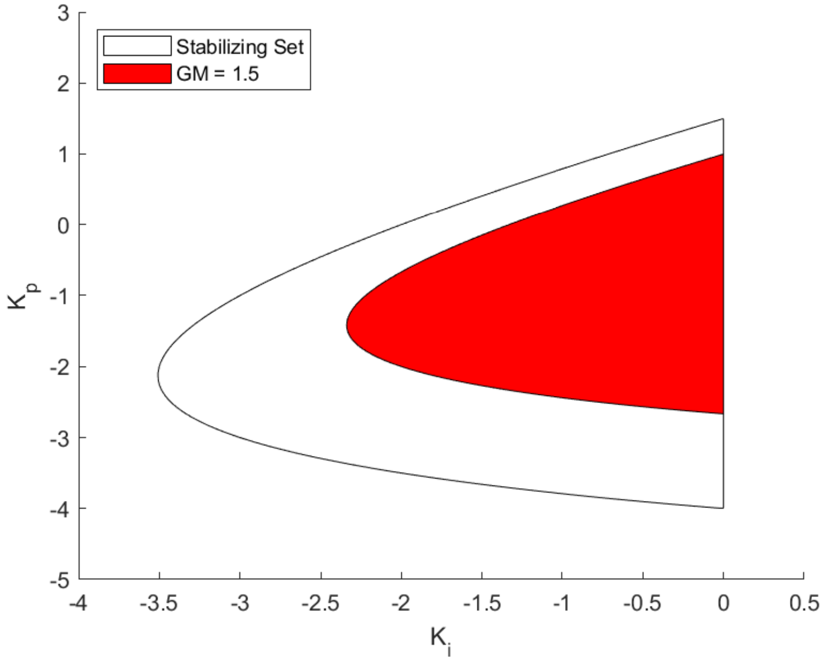

Figure 1.4: Gain Margin criterion, GM ≥ 1.5

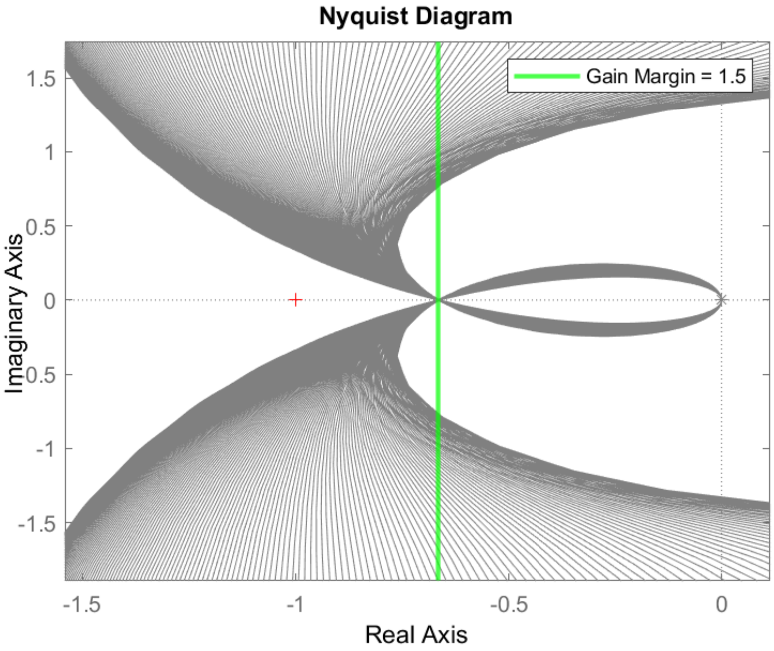

Figure 1.5: Nyquist plots demonstrating GM compliance

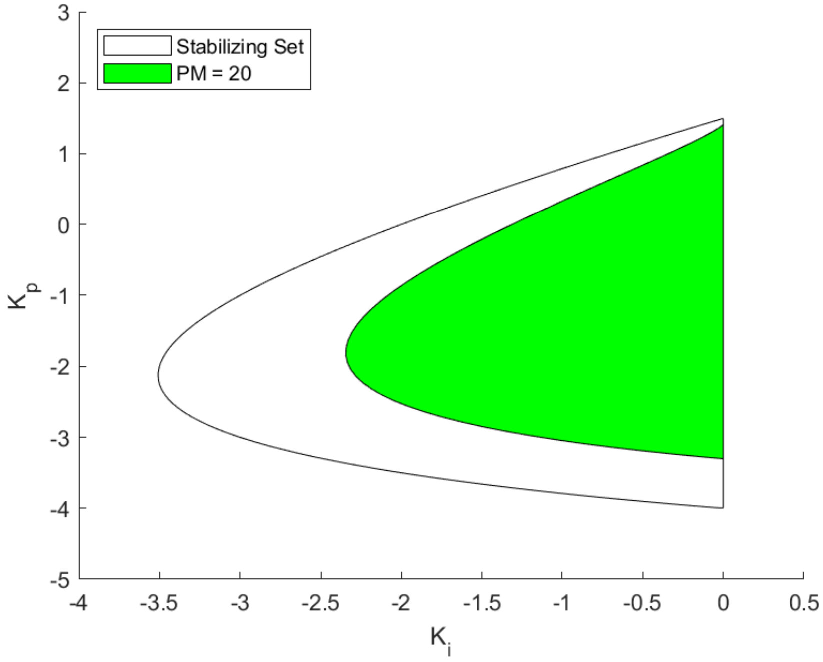

Figure 1.6: Phase Margin criterion, PM $\geq 20^0$

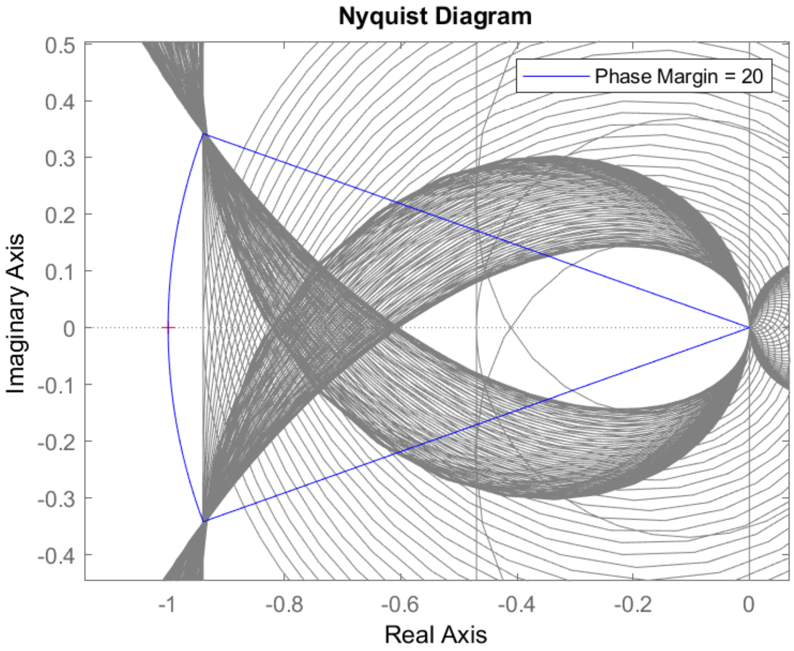

Figure 1.7: Nyquist plots demonstrating PM compliance

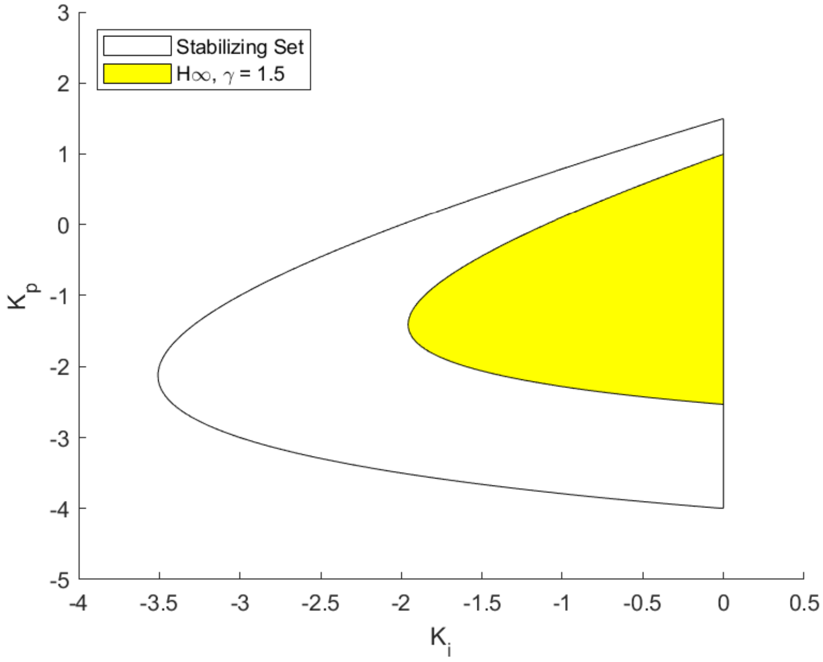

Figure 1.8: H_∞ Margin criterion, $\gamma \geq 3$

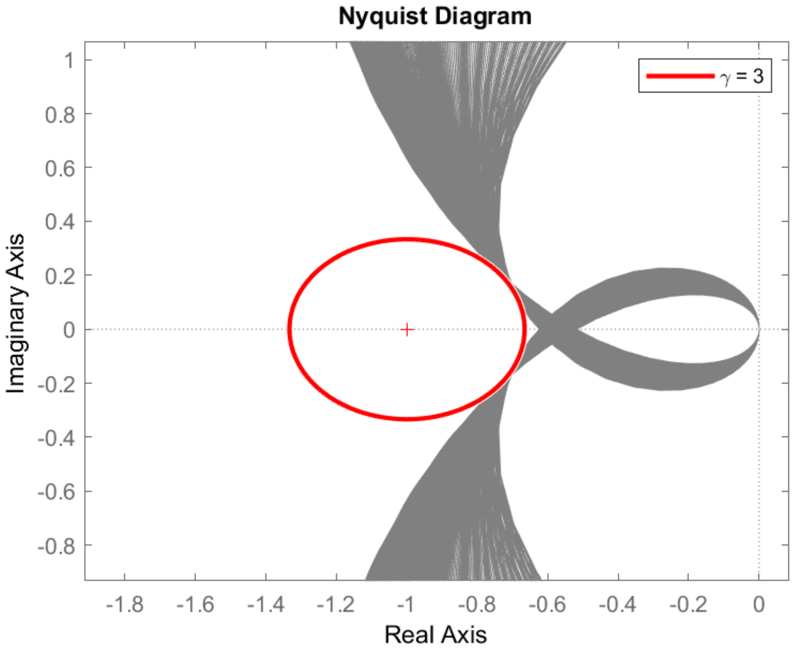

Figure 1.9: Nyquist plots demonstrating H_∞ compliance

1.4 σ-Hurwitz Stability

A Hurwitz stable polynomials has all roots in the LHP, i.e. the real magnitudes of all roots are less than zero. In the more generalized case of σ-Hurwitz stability, the roots are instead less than some value σ, and standard Hurwitz stability is considered a special case in which $\sigma = 0$.

Recently, a method was developed to find the σ-stable set $\mathcal{S}_\sigma$. The closed-loop roots of $\mathcal{S}_\sigma$ are constrained to be a distance of at least σ from the imaginary axis. Using the method from [3] to find $\mathcal{S}_\sigma$ for $\sigma = -.1$ of the same $P(s)$ used previously generates the parameter space in Fig 1.10 and rootspace in Fig 1.11 shown below.

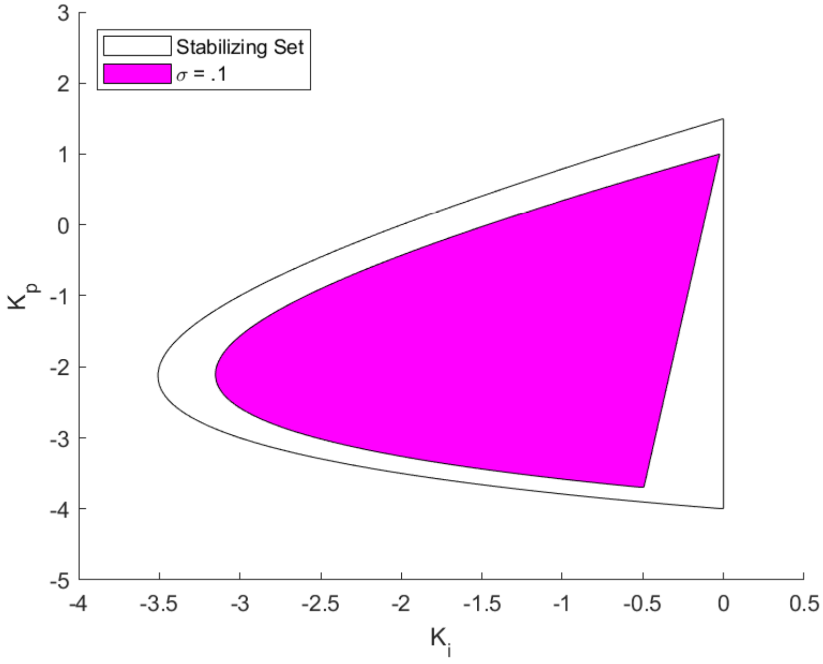

Figure 1.10: $\mathcal{S}_\sigma$ with $\sigma = -.1$

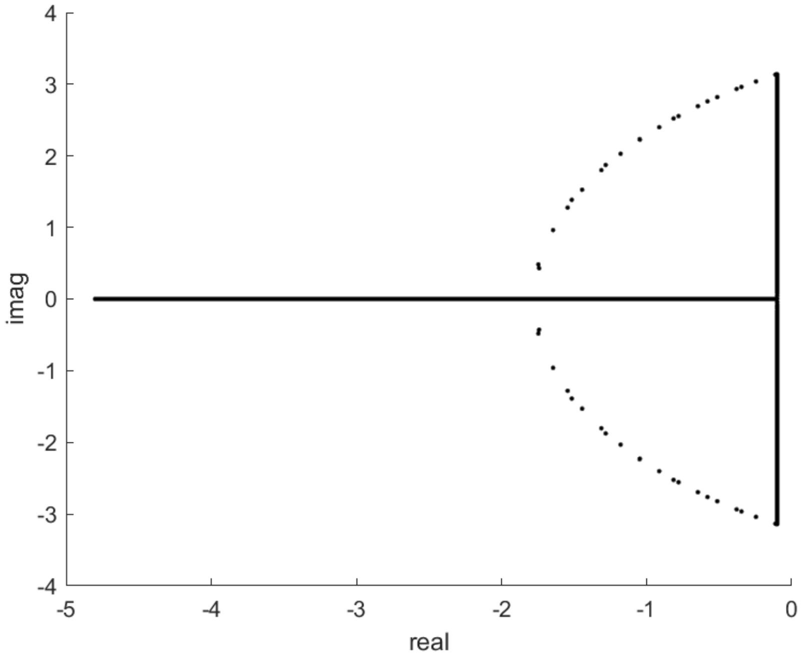

Figure 1.11: Poles along $\mathcal{S}_\sigma$ boundary showing constraint at $\sigma = -.1$

By combining the anti-Hurwitz modification and the σ-Hurwitz method, it is possible to produce $\mathcal{A}_\sigma$, a space in which all roots are *greater than* σ. While this obviously must be intersected with a Hurwitz-stable region to ensure system stability, it allows the designer to establish both a left *and* right limit along the real axis for a set's root space. An example is given below, with $\mathcal{A}_\sigma$ intersected with the $\mathcal{S}_\sigma$ region previously shown in Fig 1.12. The poles around the edges of the resultant are shown in Fig 1.13.

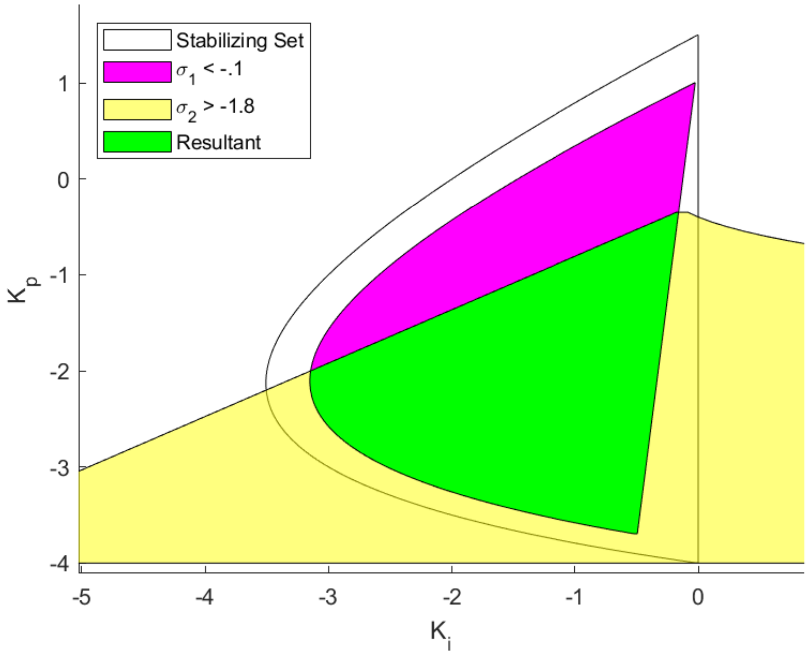

Figure 1.12: Intersection of $\sigma_1 \leq -.1$ and $\sigma_2 \geq -1.8$

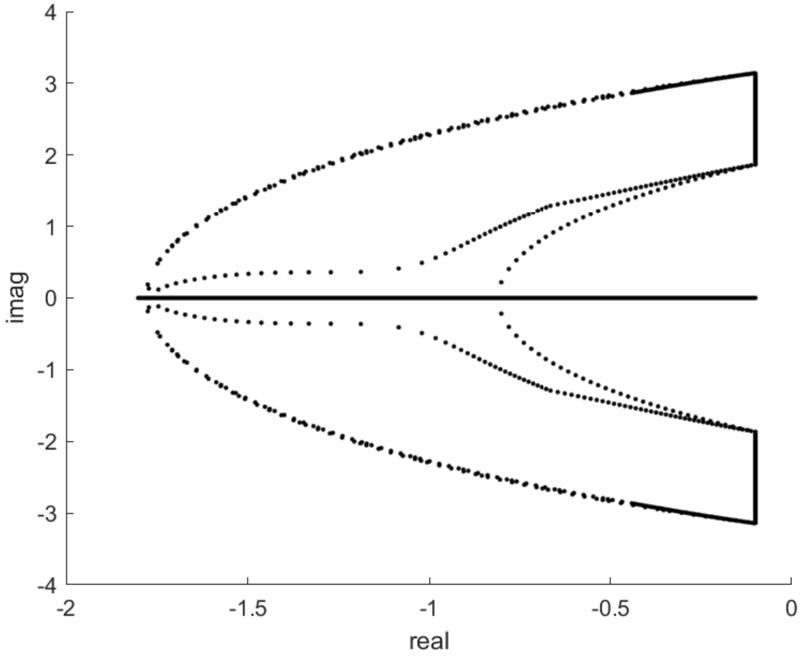

Figure 1.13: Plot showing $-1.8 \leq \sigma \leq -.1$ pole constraint

This provides a partial solution to pole placement using the stabilizing set method, although it provides no way to control the placement of poles based on their imaginary magnitudes. This is the problem that will be addressed in this thesis.

1.5 Notes and References

The plant used as an example is from [4], describing the response of a machine lathe.

The stabilizing set was first introduced in [5]. Bhattacharyya, Datta and Keel developed the method in [6]. Diaz-Rodriguez developed the gain and phase margin techniques in [1], and Han developed the H_∞ and σ-Hurwitz techniques in [2],[3]. [3] also contains a condensed overview of the stabilizing set.

All of these methods are collected together in [7].

Despite the ubiquity of higher-order controllers in literature, low order controllers remain the most common in application and PID controllers continue to be used in a diverse range of current applications, including unmanned aerial vehicles [8], chemical processes [9], automotive design [10], and industrial robotics [11].

2. DEVELOPMENT OF METHOD

2.1 Introduction

D-decomposition was originally stated in [12] in a Russian-language article. Here it was used to show that the characteristic equation $\delta(j\omega)$ of a linear system would, from $-\infty < \omega < \infty$, form a series of curves and a line (at $j\omega = 0$) that partitioned the parameter space into defined root distribution invariant regions. Later, in English language literature, [13] noted that this technique could be used to determine the stability region of a given characteristic polynomial, as well as estimate some characteristics of the response. Most application of d-decomposition since have focused on using it to identify stability regions. [14] generalized the findings of [13], producing a so-called "parameter plane" system which is similar to the method we used below. Simply put, [14] noted that d-decomposition could be applied not simply to $\delta(j\omega)$ but to $\delta(s+j\omega)$. This allowed for the determination of frequency response in a two-pole system. Since then, d-decomposition has been primarily employed as a means of finding the stability region for a given system, although [15] developed a method to determine H_∞ regions of first-order controllers based on d-decomposition.

2.2 D-Decomposition

D-decomposition is a method of mapping a characteristic polynomial in parameter space as a function of ω. Doing so results in a partition of the parameter space into root distribution invariant regions, which is to say invariant with respect to the *location* of roots of the polynomial. Most applications of d-decomposition use the method to search for the stability region. However, given the ability to calculate $\mathcal{S}$ as shown previously using the stabilizing set method, this is of little interest. Fortunately, this tool has other uses relevant to the problem addressed by this thesis.

For partitioning a plant and PI controller

$$P(s) = \frac{N_e(s^2) + sN_o(s^2)}{D_e(s^2) + sD_o(s^2)}, \tag{2.1}$$

$$C(s) = \frac{K_p s + K_i}{s}, \tag{2.2}$$

where N_e, D_e and N_o, D_o are even- and odd-ordered terms of $N(s)$ and $D(s)$, respectively, first find the characteristic closed-loop equation $\delta(j\omega)$.

$$\begin{aligned}\delta(j\omega) =&[-\omega^2 N_o(-\omega^2)K_p + N_e(-\omega^2)K_i - \omega^2 D_o(-\omega^2)] \\ &+ j\omega[N_e(-\omega^2)K_p + N_o(-\omega^2)K_i + D_e(-\omega^2)].\end{aligned} \tag{2.3}$$

This can be made more tractable by putting it into matrix form:

This can be put into matrix form as

$$\begin{bmatrix} \omega^2 N_o(-\omega^2) & -N_e(-\omega^2) \\ N_e(-\omega^2) & N_o(-\omega^2) \end{bmatrix} \begin{bmatrix} K_p \\ K_i \end{bmatrix} = \begin{bmatrix} -\omega^2 D_o(-\omega^2) \\ -D_e(-\omega^2) \end{bmatrix} \tag{2.4}$$

Note that at $\omega = 0$, this produces the line

$$N_e(0)K_i = 0 \tag{2.5}$$

More generally, any point mapped from the complex plane $\mathbb{C}$ to parameter space touches the real axis of $\mathbb{C}$, it will produce a line in parameter space.

Using the plant with PI controller from the previous section:

$$P(s) = \frac{s-2}{s^2+4s+3}, C(s) = \frac{K_p s + K_i}{s} \tag{2.6}$$

We can use this method to produce the following decomposition:

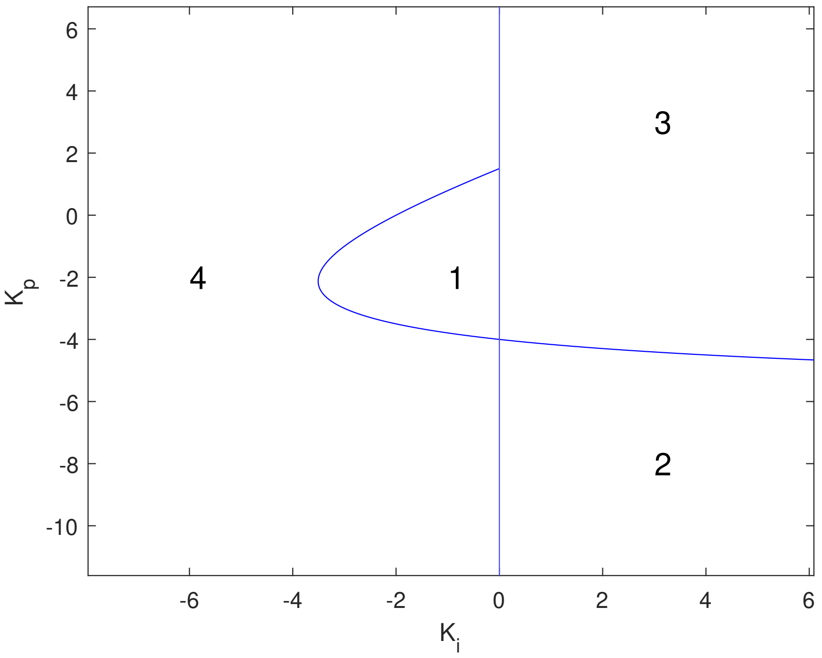

Figure 2.1: D-Decomposition of $P(s)$

As illustrated in Fig 2.1, there are four root distribution invariant regions. The curve and line are simply the imaginary axis mapped from $\mathbb{C}$ into (K_p, K_i) space. So, if there exists *any* region of parameter space that meets the Hurwitz stability criterion of all roots being to the left of the imaginary axis, and any region where this criterion is not met, moving from the one region to another must involve a crossing of the imaginary axis – in parameter space, a crossing from one of the labeled regions to another. However, one must find the roots of a point in each of these regions in order to discover on which side of $\mathbb{C}$ the roots of a region actually lie.

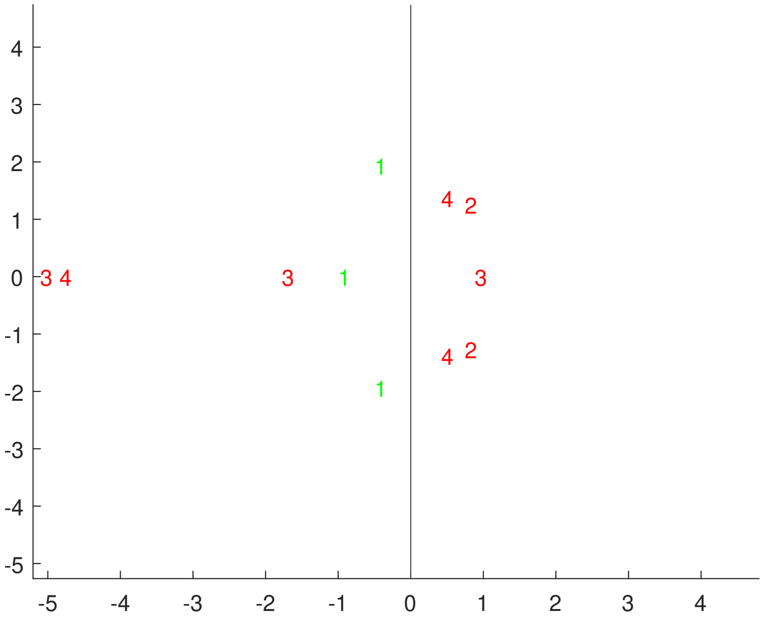

Figure 2.2: Testing Roots of Each Invariant Region

As Fig 2.2 shows, only Region 1 of Fig 2.1 has roots which meet the Hurwitz stability criterion. Again, it is readily apparent that for either the real or complex-pair roots of Region 1 to violate the stability criterion would involve crossing the imaginary axis in $\mathbb{C}$ and thus a corresponding crossing into another region in parameter space. Fig 2.3 highlights this region; this is the same result found using the stabilizing set method and shown in Fig 1.2 in the previous section.

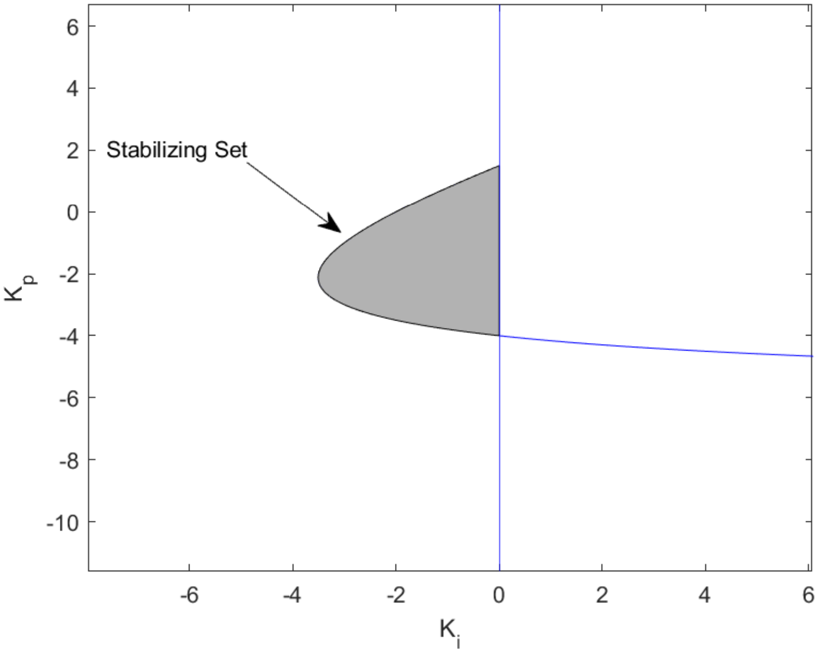

Figure 2.3: D-Decomposition of $P(s)$, with $\mathcal{S}$

We have thus identified $\mathcal{S}$ using an alternative method. However, the ability to partition parameter space into root distribution invariant regions allows for other conditions than simple Hurwitz stability to be applied. It should also be noted that the regions of parameter space that *do not* provide stability are of little interest to the designer.

2.3 Generalized D-Decomposition

The preceding method of decomposition allows s to traverse only along the imaginary axis. However, we are interested in finding regions with nonzero real values ($\sigma \neq 0$). For this, the even-and-odd method of decomposition given above is no longer feasible and we need a new approach.

First, start with a more general plant:

$$P(\sigma + j\omega) = \frac{N_r + jN_i}{D_r + jD_i} \tag{2.7}$$

where N_i, N_r, D_r, D_i are the respective real and imaginary numerical values of $N(\sigma + j\omega)$ and $D(\sigma + j\omega)$ for any given σ and ω.

Using this method gives us the complex characteristic polynomial

$$\begin{aligned}\delta(K_p, K_i) = K_p[(\sigma N_r - \omega N_i) + j(\sigma N_i + \omega N_r)] \\ +K_i(N_r + jN_i) + (\sigma + j\omega)(D_r + jD_i)\end{aligned} \tag{2.8}$$

Which can be separated into real and imaginary parts and rewritten as

$$\begin{bmatrix} \sigma N_r - \omega N_i & N_r \\ \sigma N_i + \omega N_r & N_i \end{bmatrix} \begin{bmatrix} K_p \\ K_i \end{bmatrix} = \begin{bmatrix} -\sigma D_r + \omega D_i \\ -\sigma D_i - \omega D_r \end{bmatrix} \tag{2.9}$$

The solutions for K_p and K_i represent a single point in (K_p, K_i) parameter space with the given closed-loop root, $s = \sigma + j\omega$ although, as previously noted, when $\omega = 0$ the mapping to parameter space is a line rather than a point. This is suitable for PI controllers.

Returning to the plant and PI controller in 2.6, we can apply this generalized method.

Take the characteristic polynomial $\delta(s)$ substitute $s = \sigma + j\omega$, and decompose it into real and imaginary parts.

$$\begin{gathered}\begin{bmatrix} \sigma^2 - 2\sigma - \omega^2 & \sigma - 2 \\ 2\sigma\omega - 2\omega & \omega \end{bmatrix} \begin{bmatrix} K_p \\ K_i \end{bmatrix} = \\ \begin{bmatrix} -\sigma^3 - 4\sigma^2 + 3\sigma\omega^2 - 3\sigma + 4\omega^2 \\ -3\omega\sigma^2 - 8\omega\sigma + \omega^3 - 3\omega \end{bmatrix}\end{gathered} \tag{2.10}$$

For this example we will set $\sigma = [-1.6, -.1]$ and $\omega_0 = 1.8$ i.e. $\omega = [-1.8, 1.8]$ – since $\delta(s)$ has all real coefficients this will be symmetrical across the real axis. Sweep $j\omega$ from [-1.8,1.8] with $\sigma = -.1$, then $\sigma = [-1.6, -.1]$ while $j\omega = 1.8$. Then, $j\omega = [0, 1.8]$ with $\sigma = -1.6$ and finally $\sigma = [-1.6, -.1]$ while $j\omega = 1.8$. This region is shown graphically in Fig 2.4.

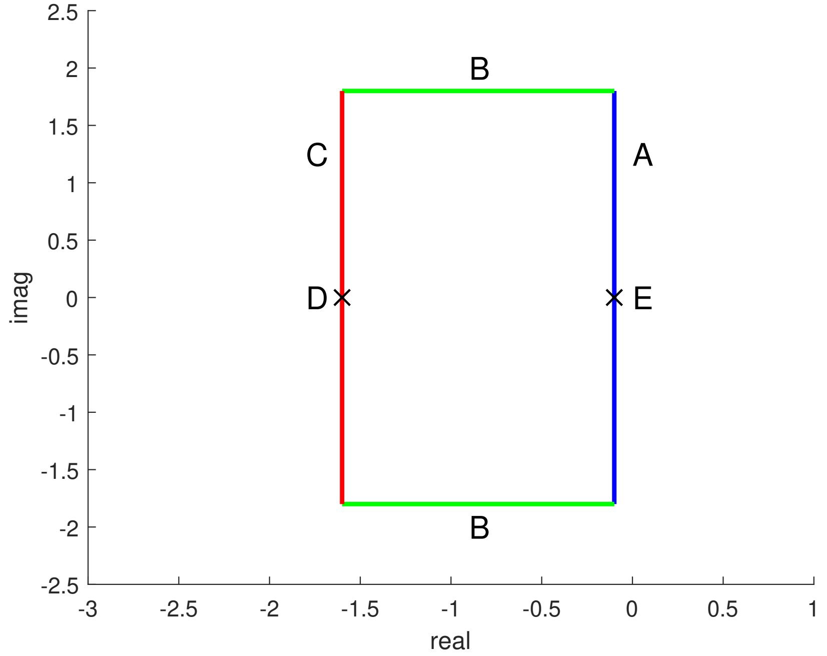

Figure 2.4: Decomposition input, Rectangular Constraint Region

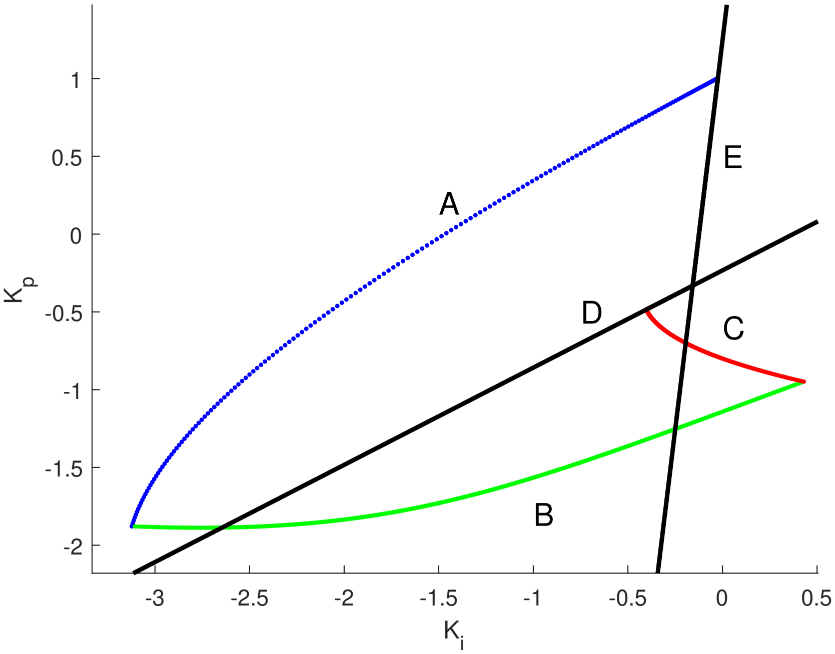

Figure 2.5: Map to parameter space, Rectangular Constraint

The parameter-space mapping shown in Fig 2.5, like the first d-decomposition illustration, also divides the parameter space into root distribution invariant regions. These are labeled in Fig 2.6.

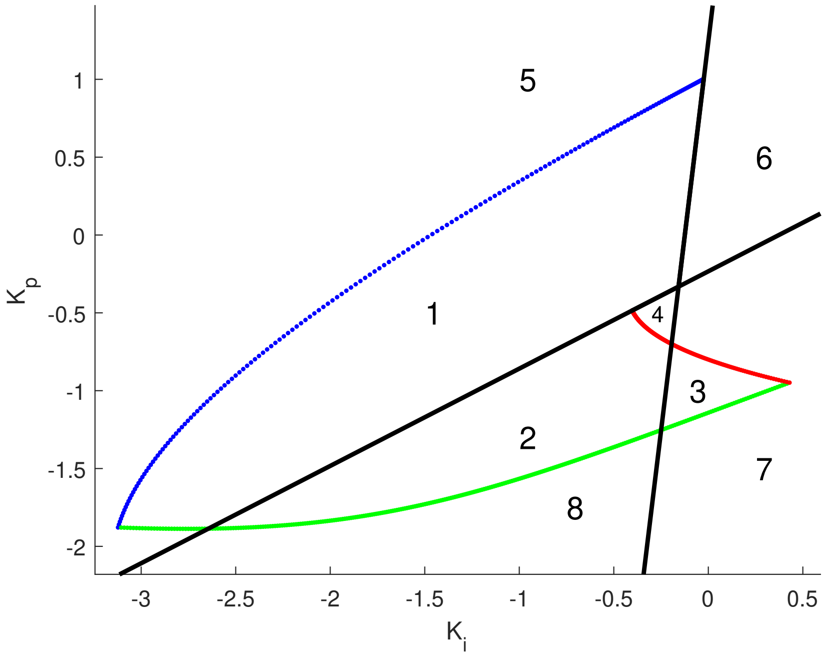

Figure 2.6: Root Distribution Invariant Regions, Rectangular Constraint

We must select a test (K_p, K_i) within each of these regions and find the poles at this point to determine which, if any, of these regions corresponds to the interior of the box drawn in Fig 2.4.

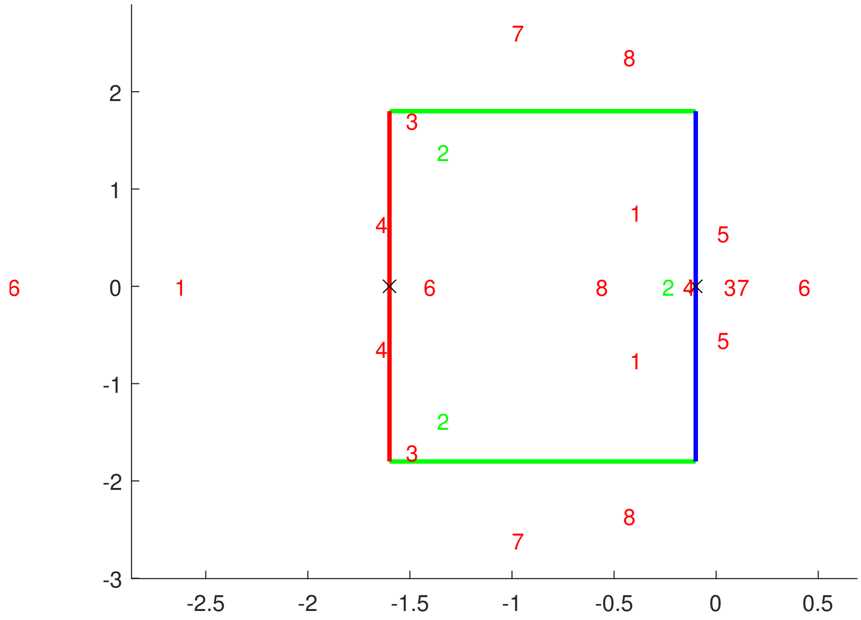

Figure 2.7: Testing of System Poles, Rectangular Constraint

The results shown in Fig 2.7 show that Region 2 contains all of the poles within its boundaries. We thus reject the other regions and use Region 2 as our final result as illustrated in Fig 2.8.

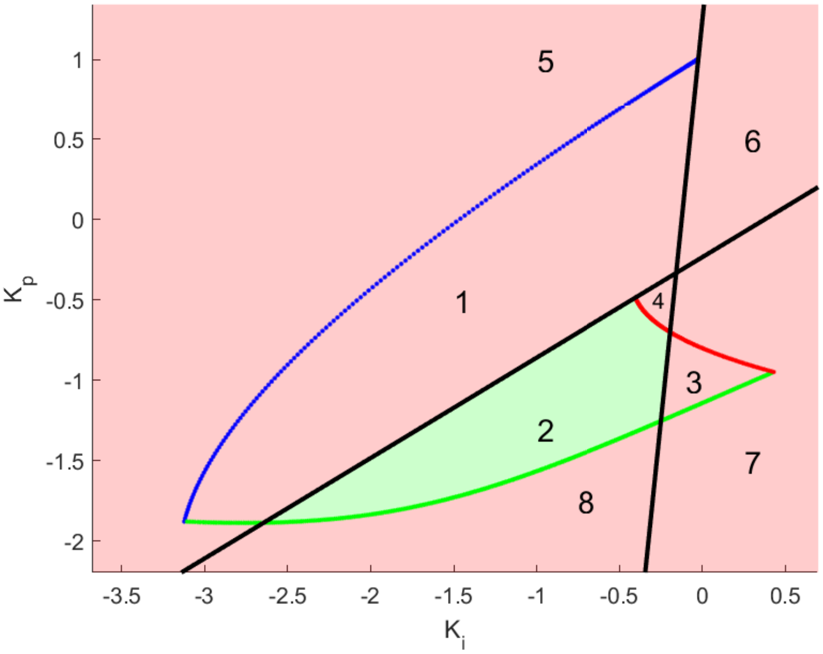

Figure 2.8: Accepted (green) and discarded regions (red), Rectangular Constraint

2.4 PID D-Decomposition

For PID controllers we need an additional modification.

Recall from Section 1 that the stabilizing set includes the identification of the stabilizing range of k_p. We can thus treat the introduction of the third gain term in the same way as in finding $\mathcal{S}$, that is, by evaluating at a fixed k_p^* within the range of k_p.

Substituting $s = \sigma + j\omega$ as in the PI case, obtain the characteristic polynomial

$$\begin{aligned}\delta(K_i, K_d) = {} & K_p^*[(\sigma N_r - \omega N_i) + j\,(\sigma N_i + \omega N_r)] \\ & + \left(K_i + K_d[\left(\sigma^2 - \omega^2\right) + j\,(2\sigma\omega)]\right)(N_r + jN_i) \\ & + (\sigma + j\omega)(D_r + jD_i)\end{aligned} \tag{2.11}$$

Which can be represented in matrix form as

$$\begin{bmatrix} \left(\sigma^2 - \omega^2\right) N_r - 2\sigma\omega^2 N_i & N_r \\ \left(\sigma^2 - \omega^2\right) N_i + 2\sigma\omega^2 N_r & N_i \end{bmatrix} \begin{bmatrix} K_d \\ K_i \end{bmatrix} = \begin{bmatrix} -\sigma D_r + \omega D_i - K_p^*(\sigma N_r - \omega N_i) \\ -\sigma D_i - \omega D_r - K_p^*(\sigma N_i + \omega N_r) \end{bmatrix} \tag{2.12}$$

This method is suitable for the decomposition of systems using PID controllers, as will be demonstrated in the next section.

2.5 Edge Theorem

The Edge Theorem is an important property that allows us to securely make assumptions about the roots of a parameter space without having to calculate them for the entire region.

As stated in [6], for a given polynomial

$$\delta(s) = \delta_n(s)^n + \delta_{n-1}(s)^{n-1} + \cdot + \delta_1(s) + \delta_0 \tag{2.13}$$

For a polytope Ω - a convex hull of a finite number of points, then assuming the sign of δ_n is constant over Ω and letting $R(\Omega)$ be the root space of Ω, then the boundary of $R(\Omega)$ is contained in the root space only of the exposed edge of Ω.

Even more simply, as reflected in the title of [16], "It Suffices to Check the Edges".

Given the Routh-Hurwitz criterion, δ_n will not change within the stability region $\mathcal{S}$. As a result, we can use the Edge Theorem to quickly determine the bounds of root space for polytopes inscribed within $\mathcal{S}$ and any subsets of $\mathcal{S}$.

A graphical illustration of the Edge Theorem is shown below; the root space shown in Fig 2.10 corresponds to the polytope inscribed in Fig 2.9.

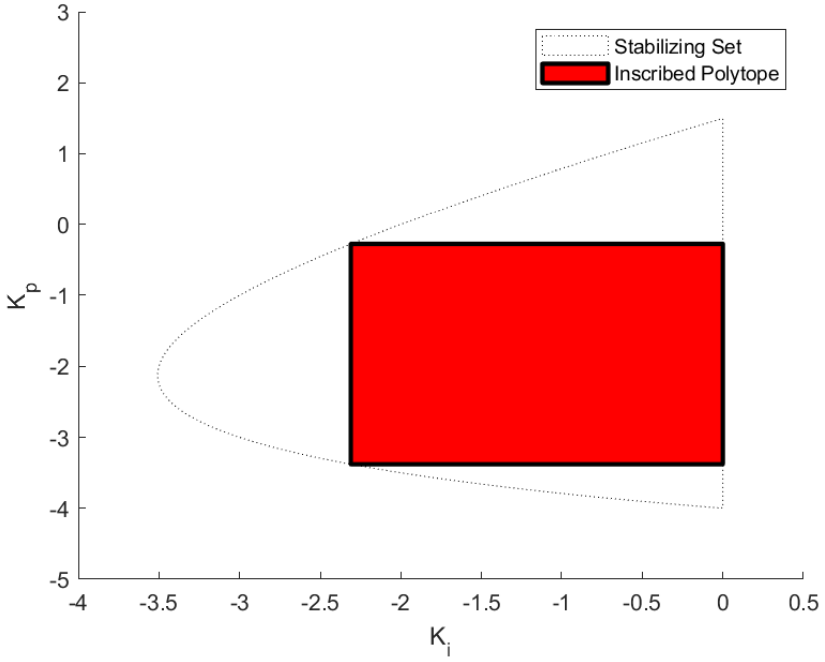

Figure 2.9: Polytope inscribed in $\mathcal{S}$

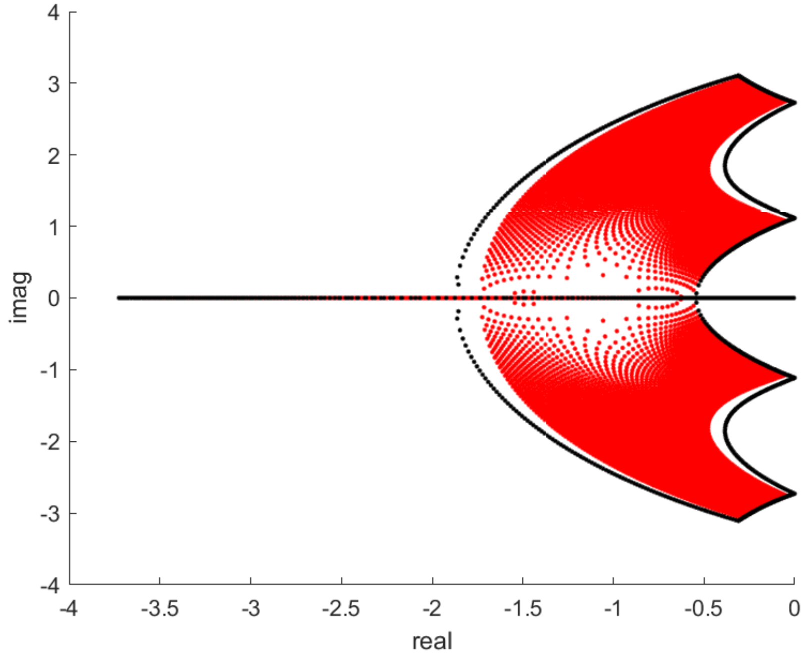

Figure 2.10: Interior and Exterior Roots of polytope from Fig 2.9

2.6 Notes and References

The Edge Theorem was stated and proved in [16].

D-Decomposition was first stated in [12] (English language: [17]. It was first used as a method of graphical analysis in [13]. Nonzero values for σ were first used in [14], which referred to it as the "parameter plane" method.

The initial method for even-and-odd decomposition is from [6].

More recently, it has been used to search for stability regions in PI [18] and in fractional-order PID controllers [19] although a pole-placement for dominant pole systems was described in [20], and for H_∞ regions in [15].

3. LIMITING COMPLEX POLES

3.1 Introduction

This chapter will explain how to use d-decomposition to confine the roots of a controlled system to a complex-plane performance region that is a subset of the stabilizing parameter space region $\mathcal{S}$. This performance region will be referred to as $\overline{\mathcal{S}}$. The Edge Theory can be used to conclusively verify the root locations of a polytope inscribed within $\overline{\mathcal{S}}$. The use of a polytope (axis-parallel box in parameter space) also allows for simple, parameter-independent ranges to be specified for an actual controller that that meets the performance constraint of $\overline{\mathcal{S}}$. The method will initially be applied to a PI example with a constant imaginary root limit ω_0. However, the root limit does not have to be constant and further examples will show this. A PID example is also given. This method allows for the damping and the reduction in frequency of step response oscillations.

3.2 Method

The method, which will be exemplified and explained in more detail below, is as follows:

1. Determine $\mathcal{S}$ using the stabilizing set method; this will check that the system can be stabilized at all.

2. Set an initial performance region, $\overline{\mathcal{S}}$ in the complex plane in terms of σ and ω and use d-decomposition to map this into parameter space.

3. Test one point of each root invariant region in $\overline{\mathcal{S}}$ to determine the corresponding parameter space.

4. If $\overline{\mathcal{S}}$ is achievable, apply additional constraints if desired.

5. To check results, inscribe a polytope within $\overline{\mathcal{S}}$ and find the roots along the edges.

To demonstrate this method, we will continue using the example from the previous chapter:

$$P(s) = \frac{s-2}{s^2+4s+3}, C(s) = \frac{K_p s + K_i}{s} \tag{3.1}$$

1. Find $\mathcal{S}$. This has already been done previously, see Fig 1.2.
2. Set initial performance region $\overline{\mathcal{S}}$. We will once again use $\sigma = [-1.6, -.1]$ and $\omega_0 = 1.8$
3. Test one point of each root invariant region in $\overline{\mathcal{S}}$ to determine the corresponding parameter space. This is shown in Figs 2.4 through 2.8. $\overline{\mathcal{S}}$ in parameter space is shown below, along with $\mathcal{S}$, in Fig 3.1.

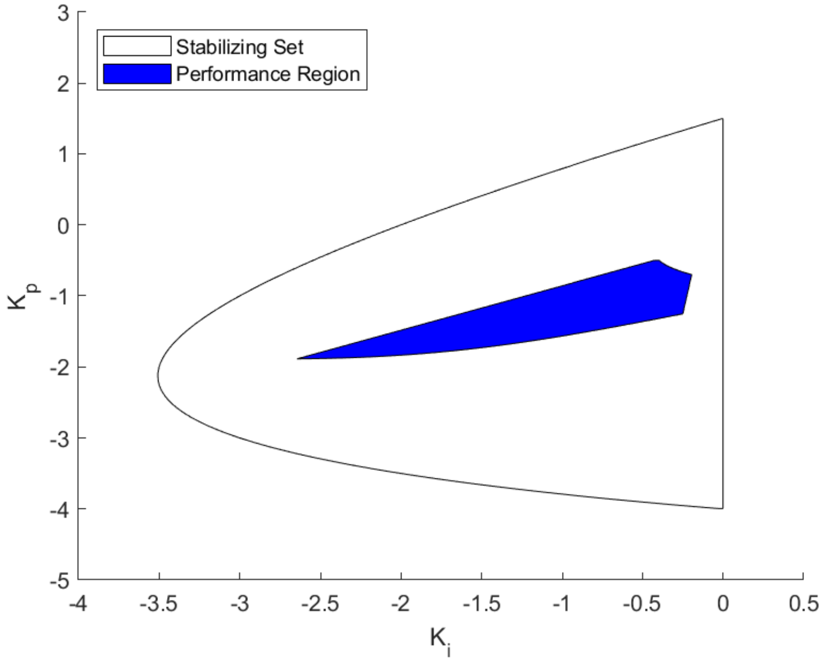

Figure 3.1: $\overline{\mathcal{S}}$ showing enclosure within $\mathcal{S}$, Rectangular Constraint

We can find the poles along the edges of $\overline{\mathcal{S}}$, shown in Fig. 3.2 along with a red box showing the original input, which now forms the constraint on the system's poles. The process of testing and discarding regions in Step 3 removed that part of the parameter space which was not achievable for this given controller-plant combination.

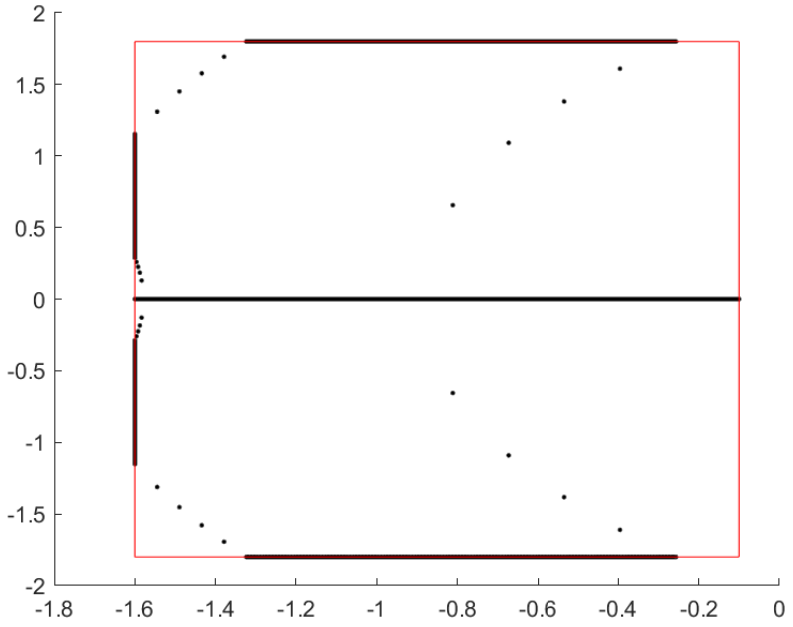

Figure 3.2: Roots along edges of $\overline{\mathcal{S}}$, Rectangular Constraint. Red line shows original d-decomposition input.

4. There are no further constraints to apply.

5. As a last step, we inscribe a polytope into the resulting $\overline{\mathcal{S}}$ in parameter space, shown in Fig 3.3. Aside from allowing us to *provably* see the location of all poles by checking only along the edges of the polytope, it has the further practical benefit of specifying parameter-independent ranges for the controller gains within $\overline{\mathcal{S}}$. The roots along the edges of the inscribed polytope are shown in Fig 3.4.

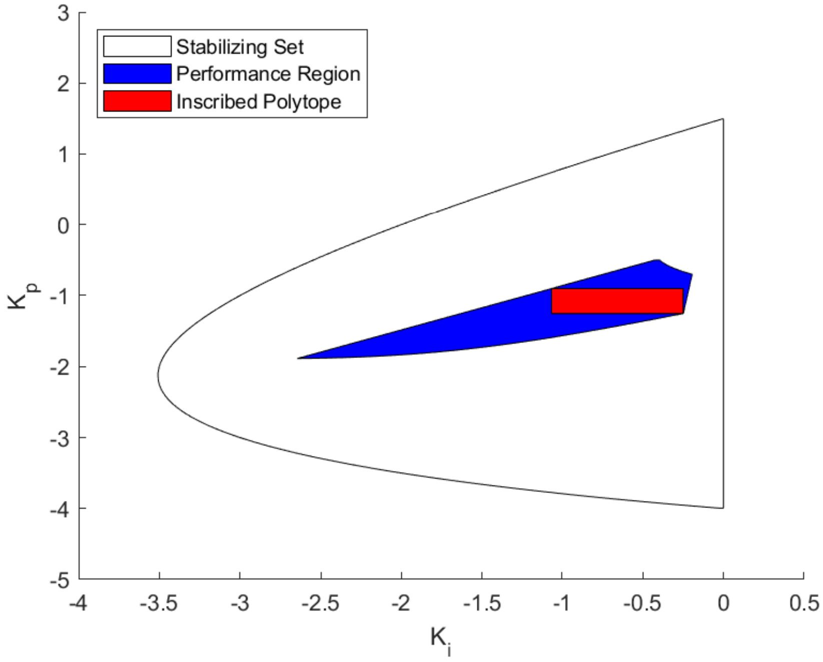

Figure 3.3: $\overline{S}$ with inscribed polytope, Rectangular Constraint

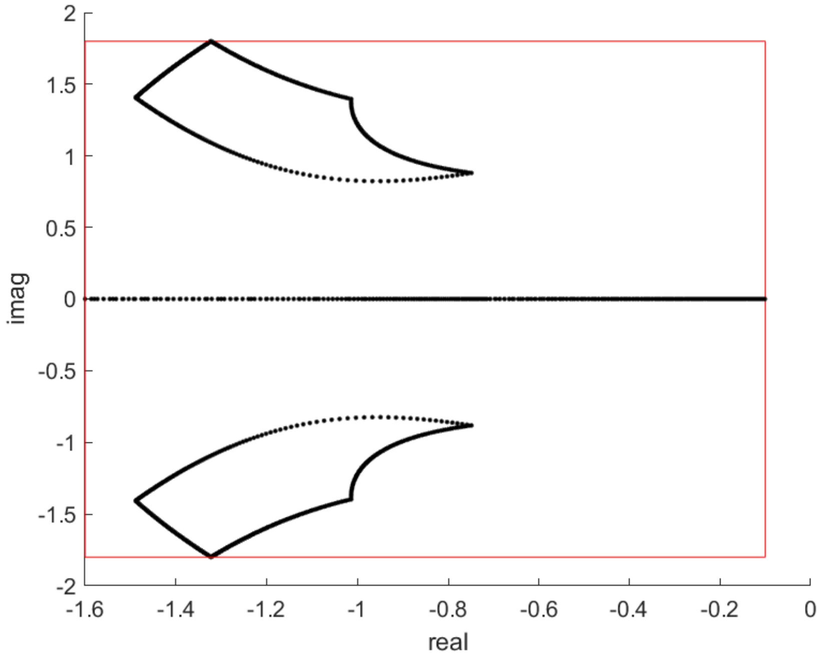

Figure 3.4: Polytope edge roots from Fig 3.4

This method can be used to constructively find a minimum value for ω by successively reducing ω_0 as shown in Figs. 3.5 and 3.6.

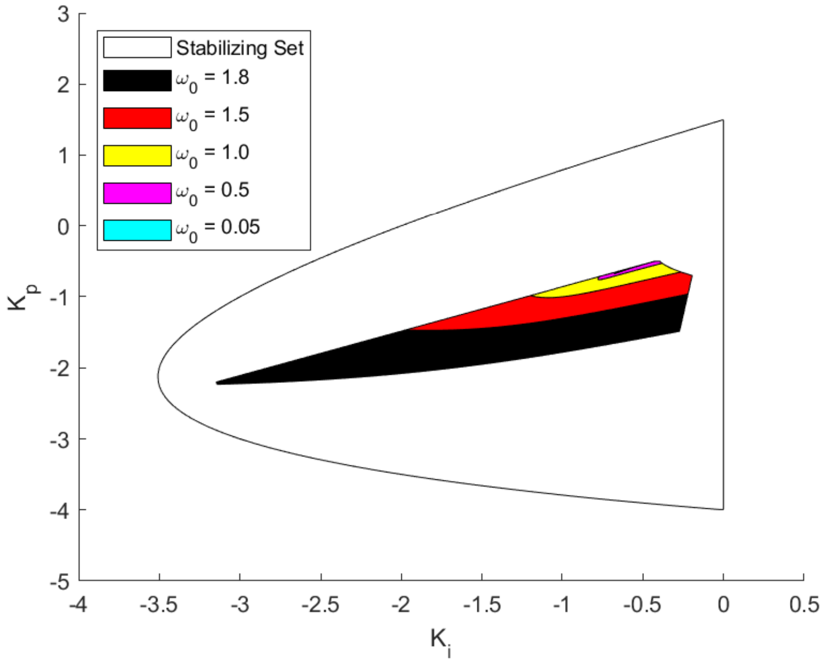

Figure 3.5: Shrinking ω_0, Rectangular Constraint

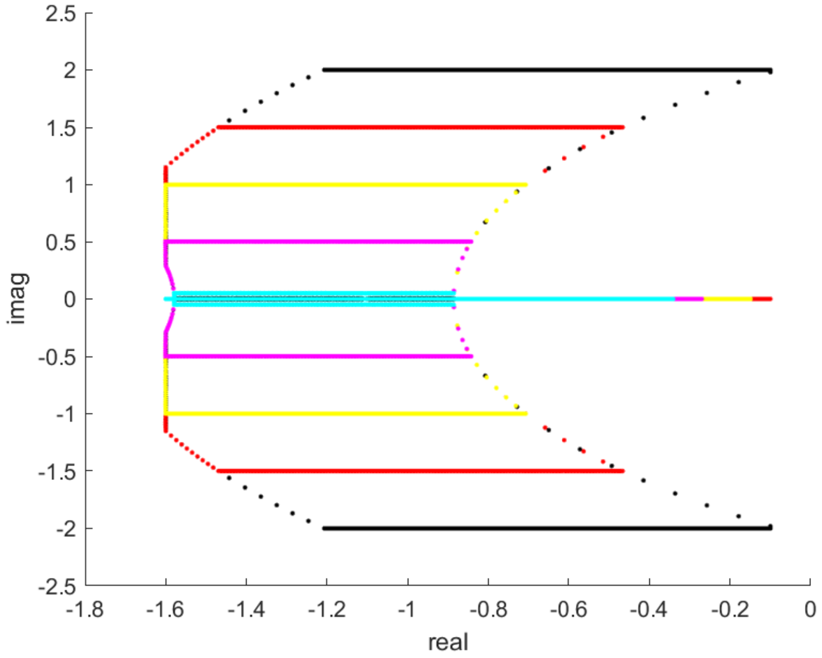

Figure 3.6: Root Space corresponding to Fig 3.5

3.3 Non-Constant Values for ω_0

ω_0 does not need to be a constant value.This allows the creation of arbitrary shapes in root space (although they must be symmetrical about the real axis given real coefficients of $\delta(s)$).

Using the same PI-controlled plant as before, we can set $\omega_0 = \sqrt{1.5^2 - (\sigma + 1.6)^2}$ to map a circle into parameter space located at $\sigma_0 = -1.6$ with a radius of 1.5 as shown in Fig 3.7.

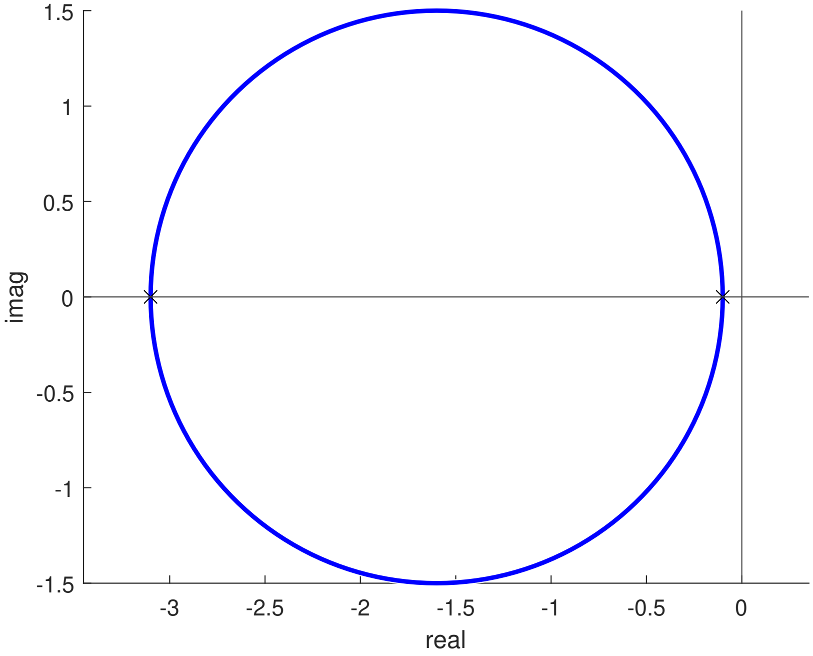

Figure 3.7: D-Decomposition input for $\overline{\mathcal{S}}$, Circular Constraint Region

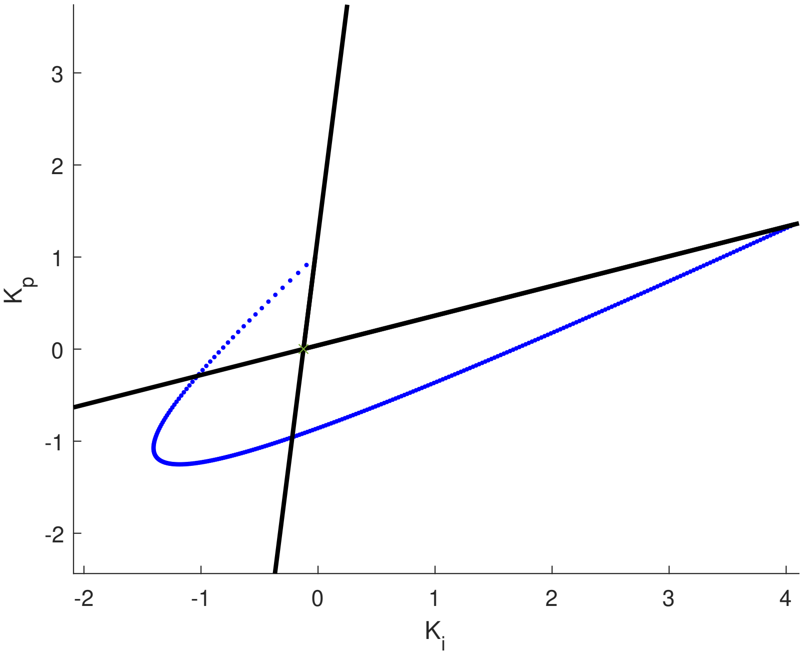

Figure 3.8: Parameter space output for circular $\overline{\mathcal{S}}$ from Fig 3.7

As before, each partitioned region of Fig 3.8 must be tested. Doing so identifies the space that corresponds to the performance region in parameter space; this is highlighted in Fig 3.9.

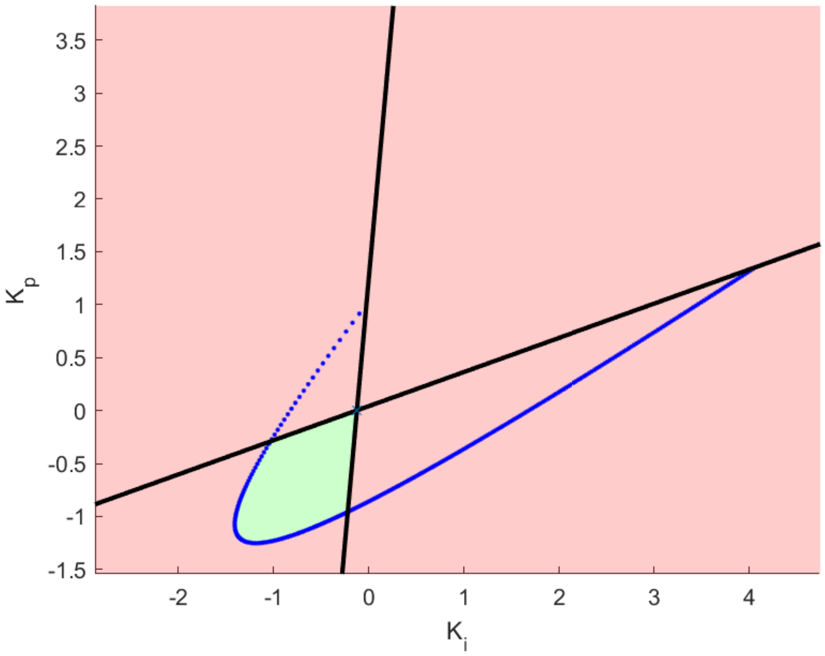

Figure 3.9: Compliant (green) and discarded (red) regions from Fig 3.8

As with the previous example, the performance region $\overline{\mathcal{S}}$ is plotted along with $\mathcal{S}$ for comparison in Fig **??**, as are the poles around the boundaries of $\overline{\mathcal{S}}$ in Fig 3.11.

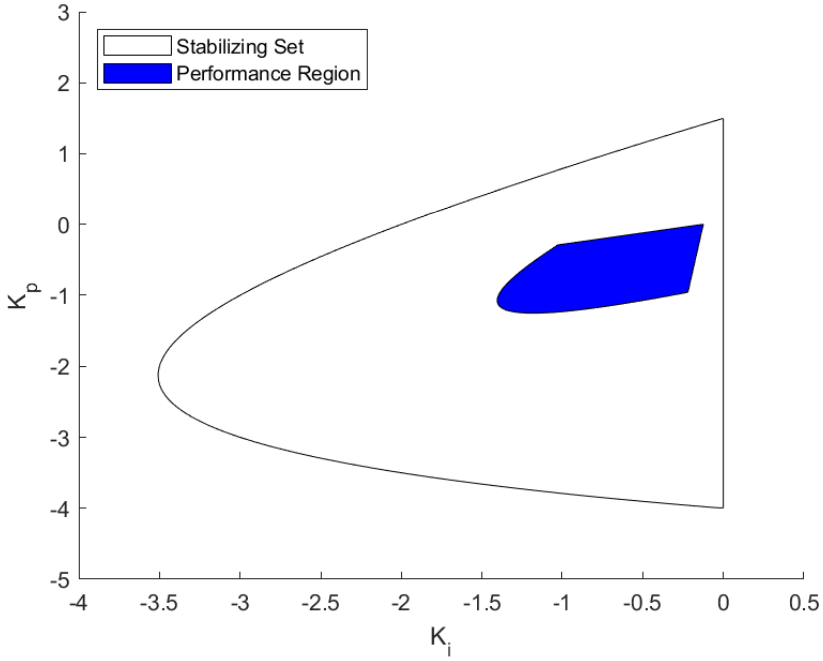

Figure 3.10: $\overline{\mathcal{S}}$ and $\mathcal{S}$, circular constraint

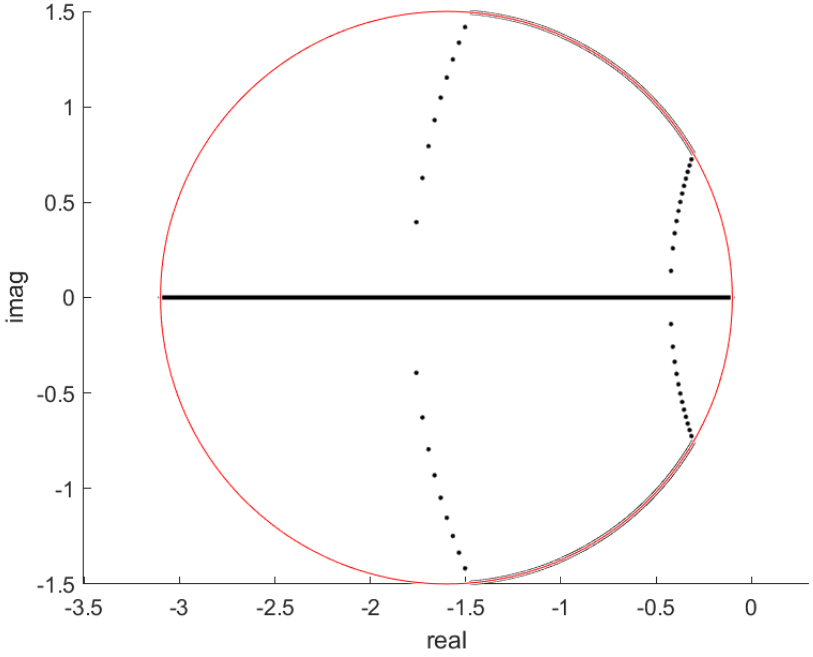

Figure 3.11: Poles along edges of $\overline{\mathcal{S}}$ from Fig 3.10, Circular Constraint

Inscribing a polytope within this region (see Fig 3.12) allows confirmation that the roots enclosed within the polytope meet the constraints as shown in Fig 3.13.

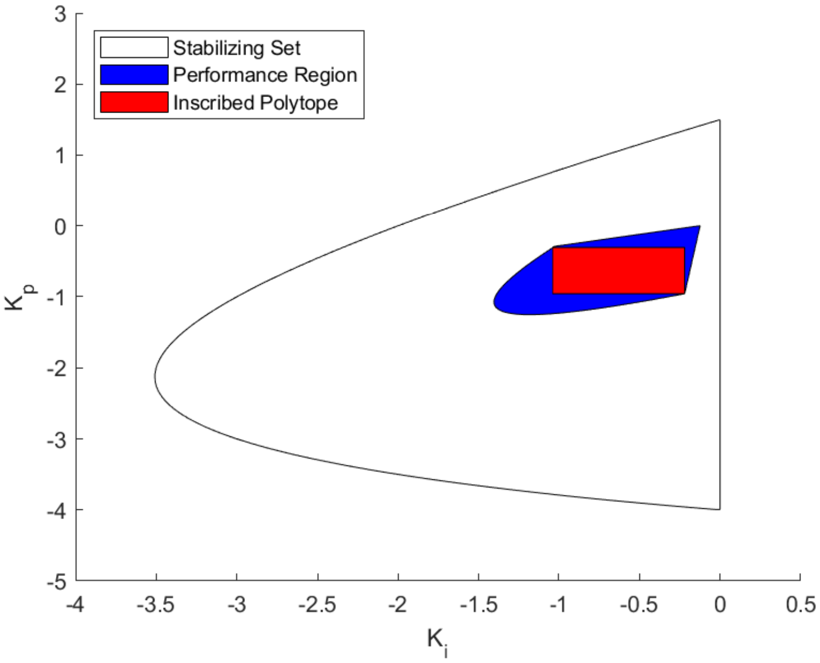

Figure 3.12: $\overline{S}$ with inscribed polytope, Circular Constraint

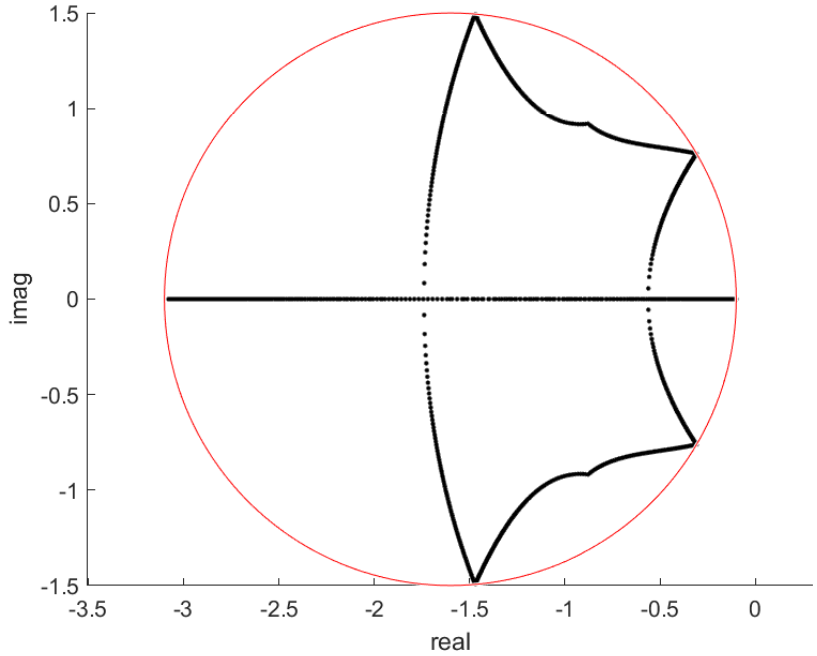

Figure 3.13: Poles along edges of polytope in Fig 3.12, Circular Constraint

A more useful shape is a Quadrant-II line with negative slope, reflected over the real axis, forming a triangle. As shown in the next section, the reducing the slope of this line can select a space of controllers with increased damping ratio.

Using the same PI-controlled plant as previous, set $\omega_0 = -m\sigma$ where m is the slope of a line. We also include real root part limits of $\sigma = [-1.6, 0]$. The figures below give the decomposition input (Fig 3.14), output to parameter space (Fig 3.15), and results of root invariant region testing (Fig 3.16).

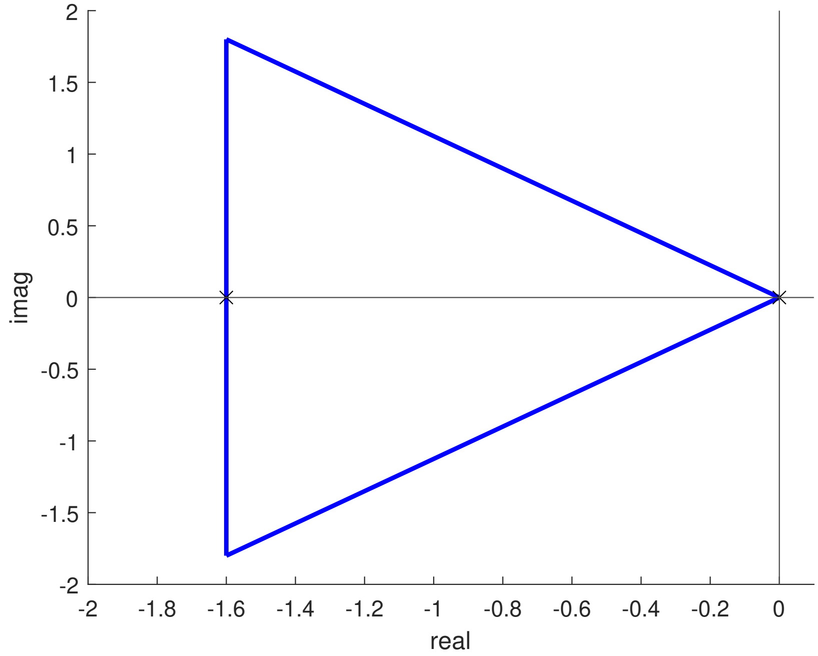

Figure 3.14: Decomposition input, Triangle with $m = -1$

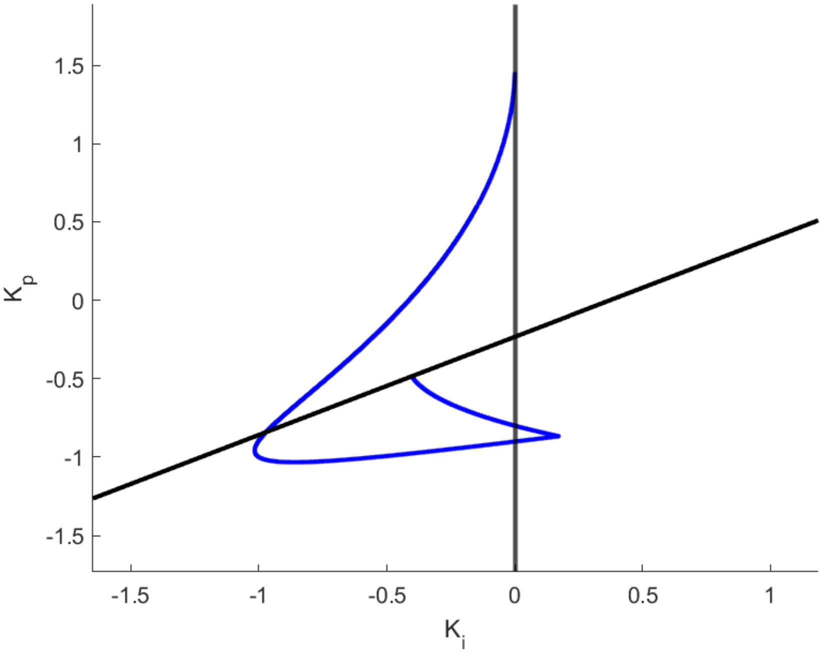

Figure 3.15: Map to parameter space, Triangular Region from Fig 3.14

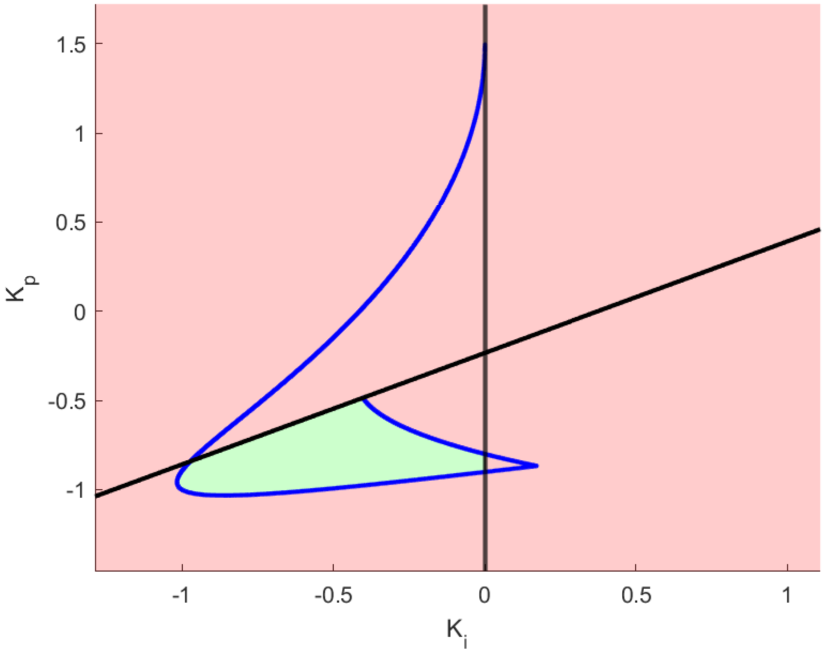

Figure 3.16: Included and discarded regions from Fig 3.15 after testing

We plot the performance region and $\mathcal{S}$ in Fig 3.17 below, along with roots along the boundaries as previous in Fig 3.18. Notice how with a right-hand real limit at $\sigma = 0$, $\overline{\mathcal{S}}$ shares a boundary with $\mathcal{S}$.

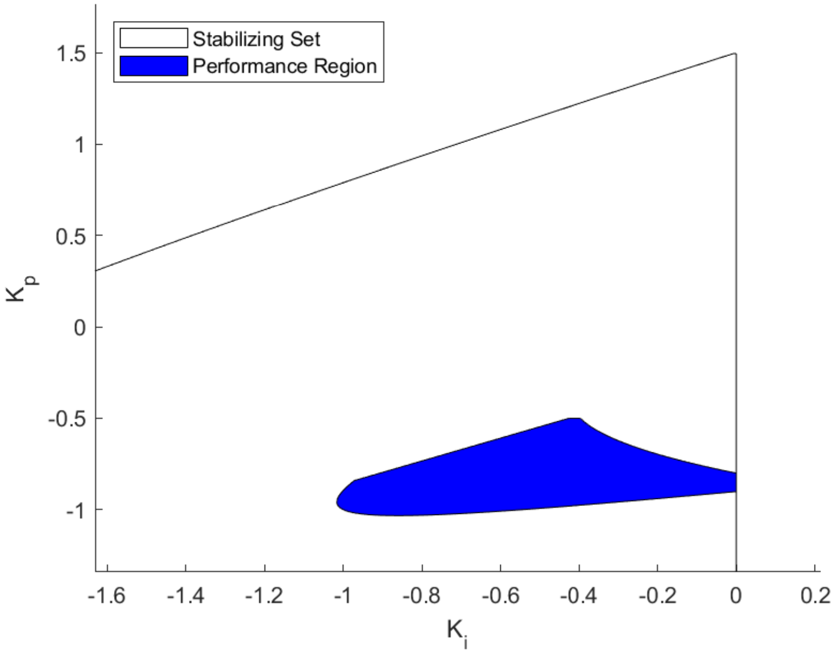

Figure 3.17: Comparison of $\overline{\mathcal{S}}$ with $\mathcal{S}$, Triangular Constraint

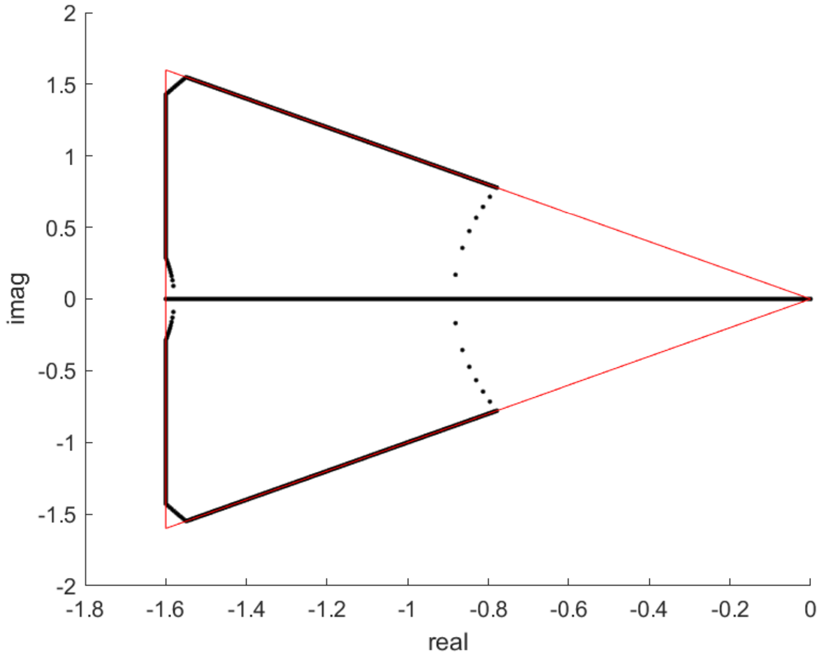

Figure 3.18: Roots along boundary of $\overline{\mathcal{S}}$ from Fig 3.18, Triangular Constraint

And once again, we inscribe a polytope within $\overline{\mathcal{S}}$ and find the roots along the edges, shown in Fig 3.19 and Fig 3.20, respectively.

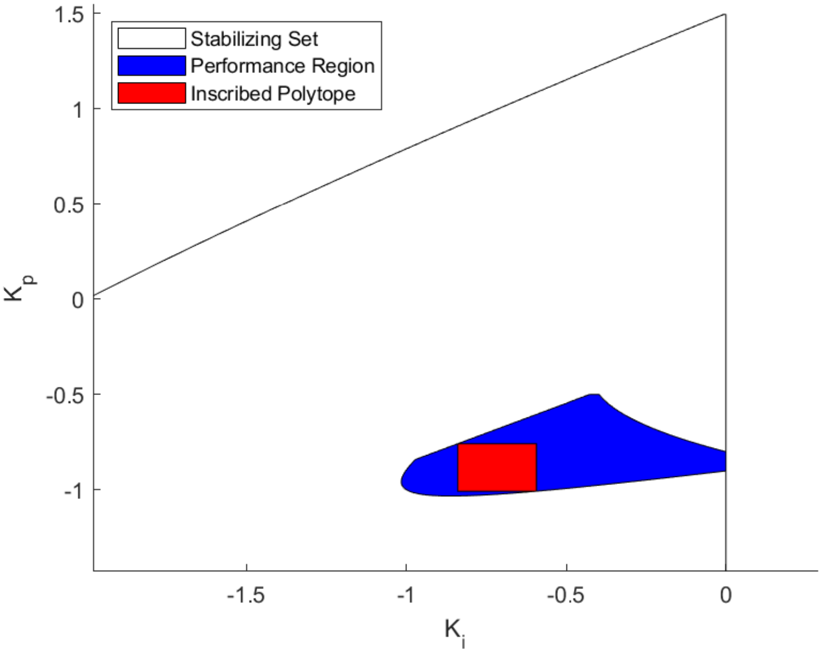

Figure 3.19: $\overline{S}$ and inscribed polytope, Triangular Region

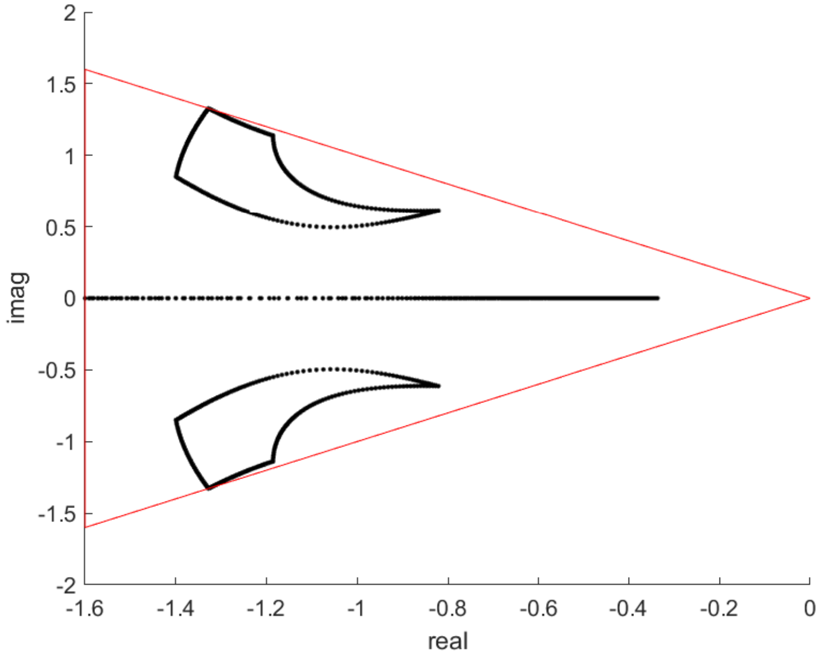

Figure 3.20: Roots along boundary of polytope, Triangular Constraint from Fig 3.19

3.4 PID Example

For a PID example, we will use the plant and PID controller

$$
\begin{aligned}
P(s) &= \frac{N(s)}{D(s)} \\
N(s) &= -2.1 \times 10^{-6}s^5 + 6.3 \times 10^{-4}s^4 - .0882s^3 \\
&\quad + 7.056s^2 - 317.5s + 6350 \\
D(s) &= 2.35 \times 10^{-13}s^8 + 5.567 \times 10^{-8}s^7 \\
&\quad + 1.727 \times 10^{-5}s^6 + .002509s^5 + .211s^4 \\
&\quad + 10.34s^3 + 255.2s^2 + 1742s
\end{aligned}
\tag{3.2}
$$

$$
C(s) = \frac{K_p s + K_i + K_d s^2}{s}
\tag{3.3}
$$

$P(s)$ is a fifth-order Padé approximation of

$$
P_0(s) = \frac{K_p s + K_i + K_d s^2}{s(2.35 \times 10^{-8}s^2 + 5.56 \times 10^{-3} + .00576} e^{.1s}
\tag{3.4}
$$

This system has a large number of complex pole pairs, intended to demonstrate the feasibility of this method in higher order, PID-controlled systems for which other methods may fail. The specific characteristics of this system limit the applicability of σ limits, so our performance region will have only ω_0, with $\sigma = [-\infty, 0)$.

1. Determine $\mathcal{S}$. Note that $K_p = [0, 4.65]$. This is shown in Fig 3.21, with root space plot in Fig 3.22.

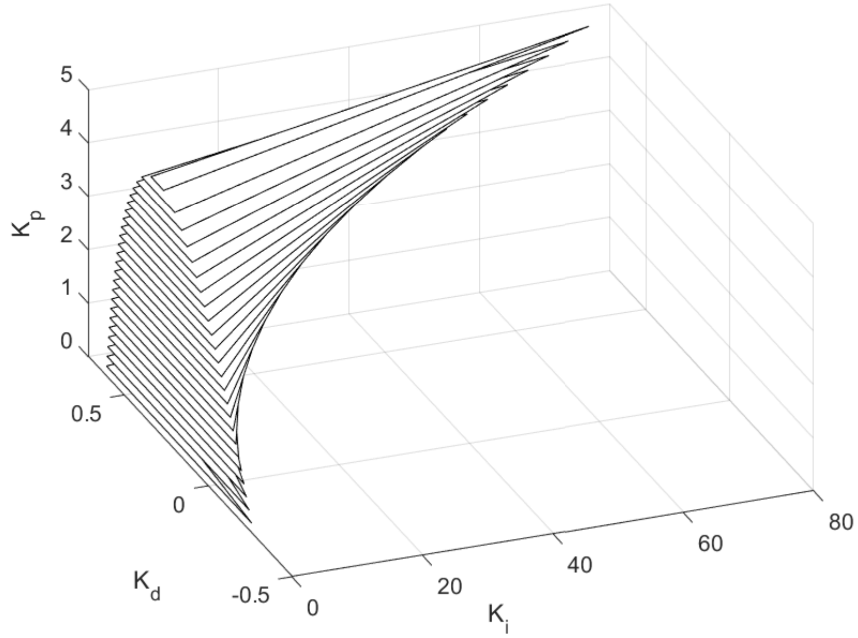

Figure 3.21: $\mathcal{S}$ for PID System

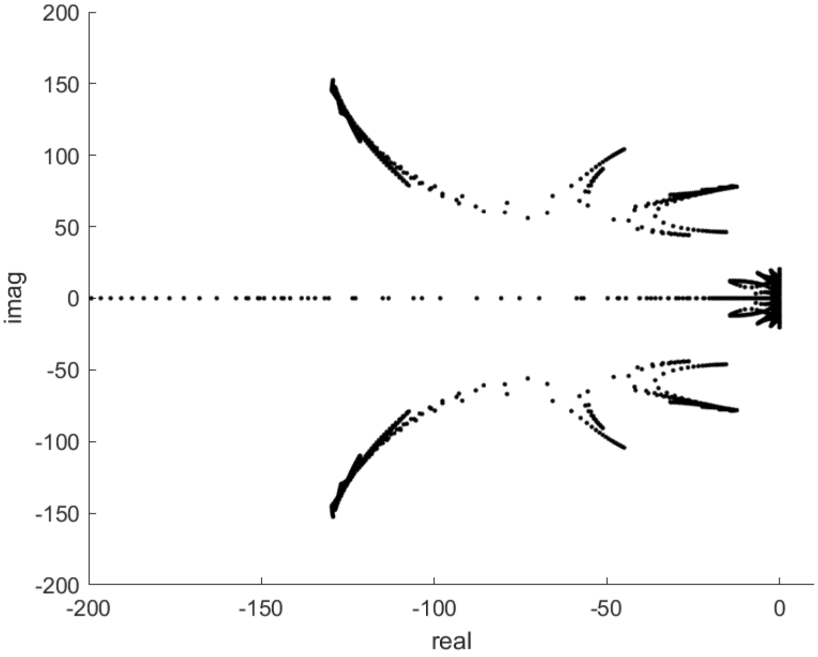

Figure 3.22: Poles along boundary of $\mathcal{S}$ for PID System

2. Set an initial performance region, $\overline{\mathcal{S}}$. Our region will be $\omega_0 = 110, \sigma = [-\infty, 0)$.

3. For each evaluated K_p^*, test each root invariant region to determine the corresponding parameter space. The results of this are shown in Fig 3.23 and the roots along the boundary of this three-dimensional space in Fig 3.24.

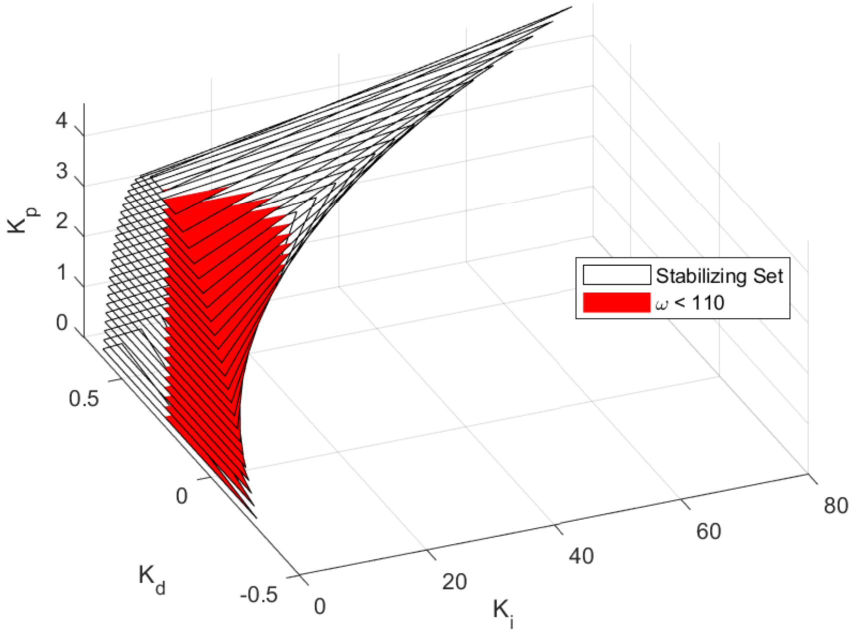

Figure 3.23: $\mathcal{S}$ with $\overline{\mathcal{S}}$ in red, PID System

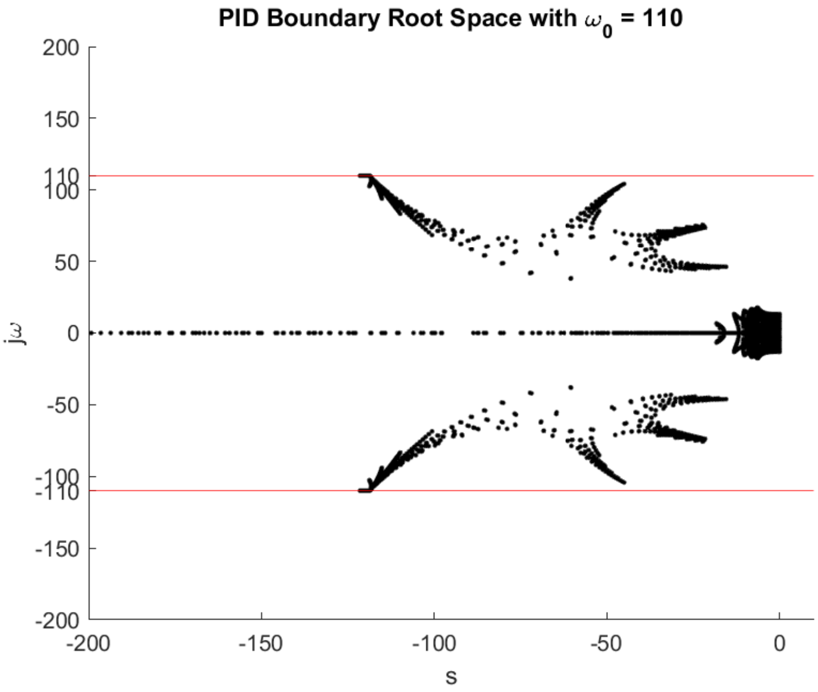

Figure 3.24: Root Space showing ω_0 limit in PID system

4. Inscribe a polytope – three dimensions for this controller – indicated in Fig 3.25 in cyan. The roots along the edges of this polytope confirm the results as shown in Fig 3.26.

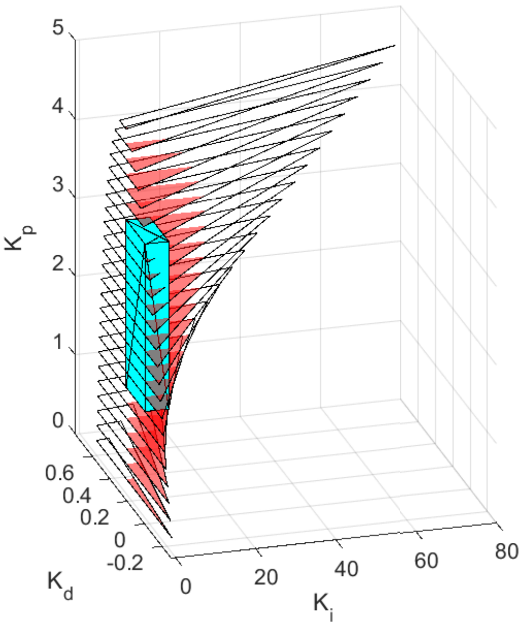

Figure 3.25: $\overline{\mathcal{S}}$ with polytope inscribed, PID System

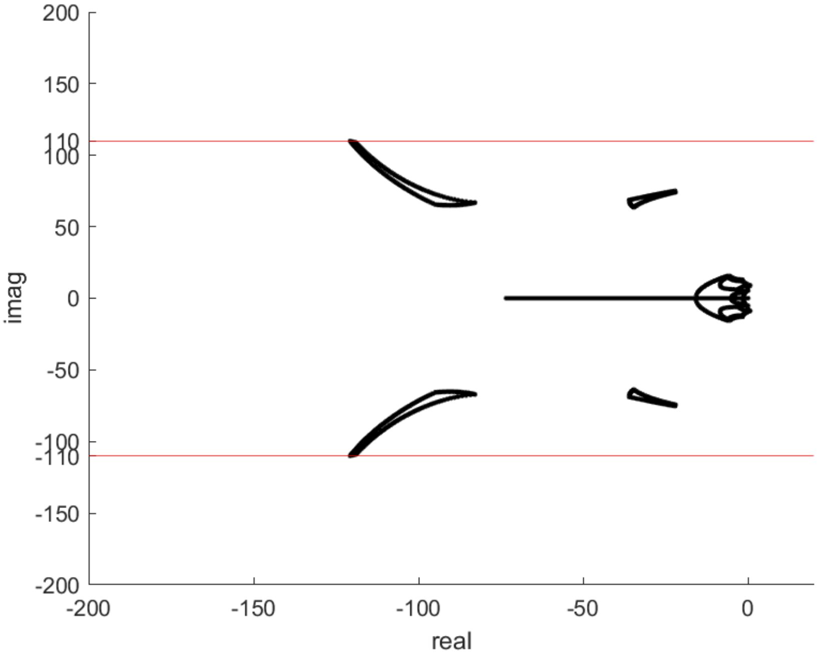

Figure 3.26: Root Space of polytope edges, PID System

3.5 Notes and References

The PID plant used in the example above is adapted from [21].

4. APPLICATIONS

4.1 Introduction and Notes

In the previous chapter, we saw how to place a system's closed-loop roots within a specified performance region, $\overline{\mathcal{S}}$, of the complex plane. Now, we will examine the effects of this on time response.

Figure 4.1 shows a range of unit step responses for controller gain values within $\mathcal{S}$. Each figure below shows fifty values for (K_p, K_i) selected at regular intervals from within the given region.

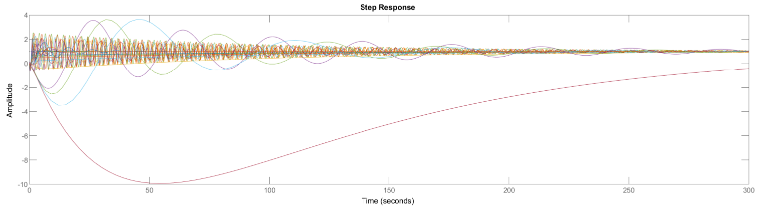

Figure 4.1: Selected Unit Step Responses of $\mathcal{S}$

Clearly, stability alone guarantees only one thing in terms of performance: finite settling time. The practical application of our method is removing undesirable responses. Although it is possible to reduce controller gains to a single point, the power of this method is that it allows the expression of a space of controllers rather than a single value. This allows the designer to make tradeoffs with other design constraints.

By moving roots away from the imaginary axis, σ-Hurwitz stability reduces settling time as shown in Fig 4.2. Partly for clarity and partly for continuity with the previous examples given, unless otherwise specified all of the following have been intersected with $\mathcal{S}_\sigma, \sigma = -.1$ to ensure removal of responses with very high settling times.

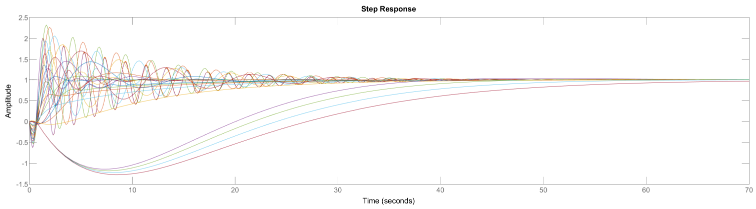

Figure 4.2: Unit Step Responses of $\mathcal{S}_\sigma$, $\sigma = -.1$

4.2 Time Response of Box Constraints

In this section, unit step responses in the time domain are considered for root box constraints with varying ω_0 and $\sigma = [-1.6, -.1]$ (Fig 4.3).

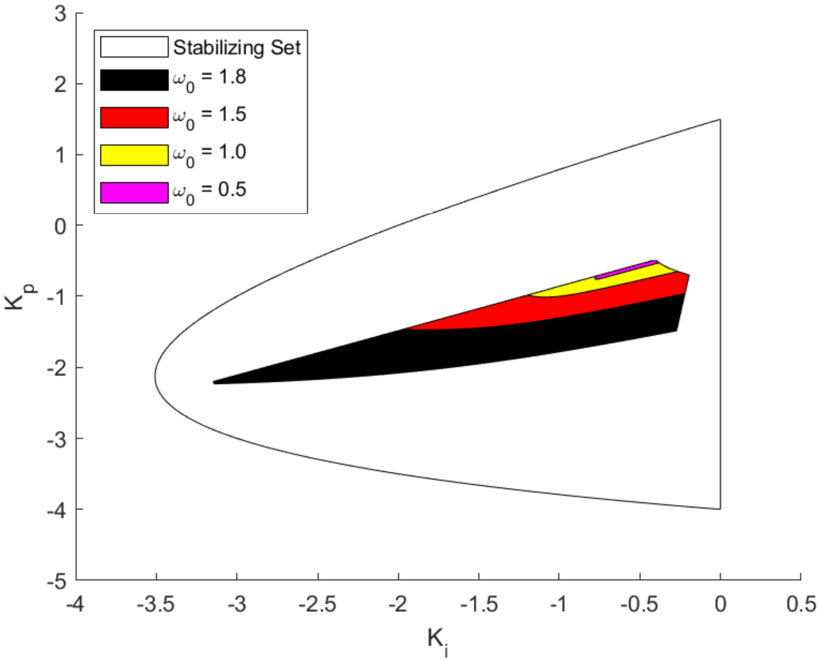

Figure 4.3: Box Constraints with varying ω_0

The elimination of higher-frequency responses is already clearly evident in comparing Fig. 4.2 to Fig. 4.4, with the latter being $\omega_0 = 1.8$ as labeled. As ω_0 is decreased in Figs 4.5 through 4.7, response oscillation approaches zero.

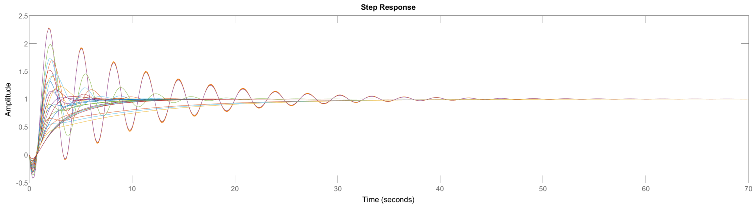

Figure 4.4: Selected Unit Step Responses from Box Constraint, $\omega_0 = 1.8$

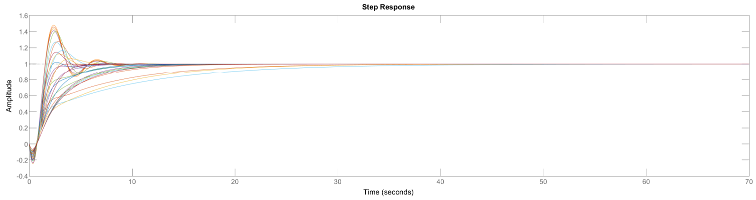

Figure 4.5: Selected Unit Step Responses from Box Constraint, $\omega_0 = 1.5$

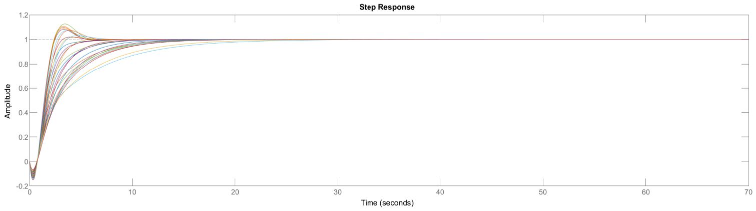

Figure 4.6: Selected Unit Step Responses from Box Constraint, $\omega_0 = 1.0$

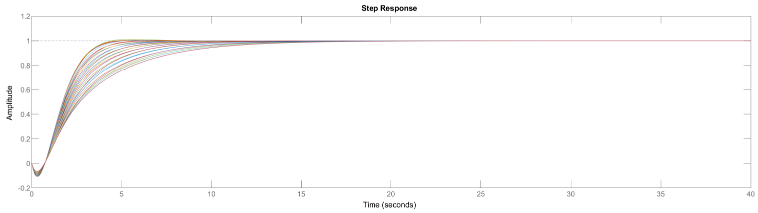

Figure 4.7: Selected Unit Step Responses from Box Constraint, $\omega_0 = 0.5$

4.3 Time Response of Triangle Constraints

In this section, samples unit step responses in the time domain are considered for triangular root constraints with varying ω_0, expressed in terms of slope m, and $\sigma = [-1.6, -.1]$ (Fig 4.8). As previously, each figure shows unit step responses for fifty values of (K_p, K_i) selected at regular intervals within $\overline{\mathcal{S}}$.

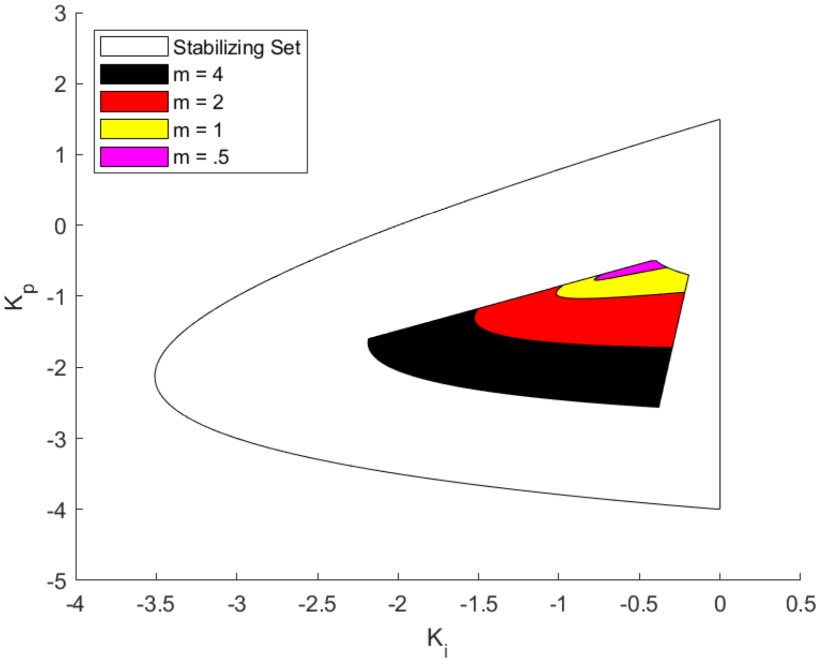

Figure 4.8: Triangle Constraints with varying m

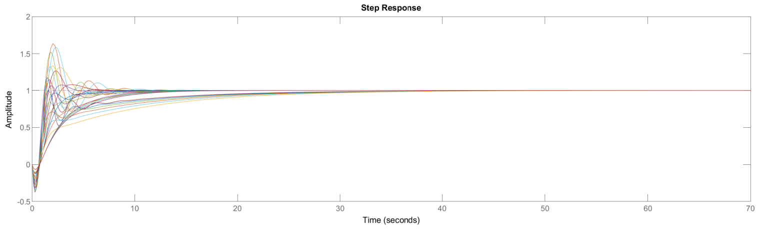

Figure 4.9: Selected Unit Step Responses from Triangle Constraint, $m = -4$

Comparison of Fig 4.9 and Fig 4.2 shows a faster response decay, while comparison with Fig 4.4 demonstrates the presence of higher-frequency oscillations in the triangle-constrained space. Figures 4.10 to 4.12 shows that as the magnitude of m decreases, so does the decay rate of the step response.

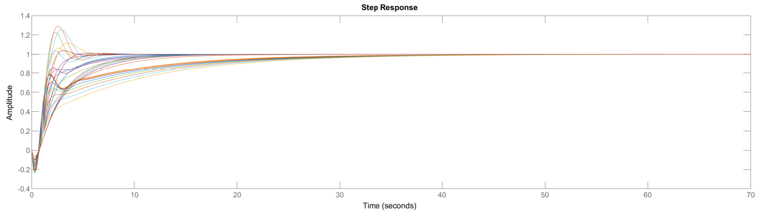

Figure 4.10: Selected Unit Step Responses from Triangle Constraint, $m = -2$

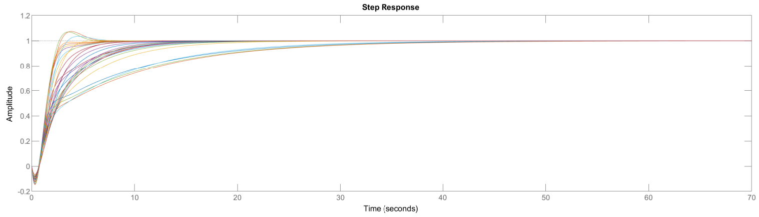

Figure 4.11: Selected Unit Step Responses from Triangle Constraint, $m = -1$

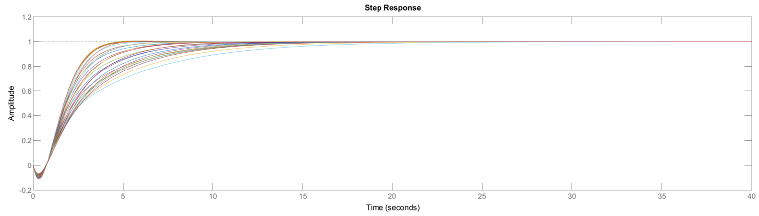

Figure 4.12: Selected Unit Step Responses from Triangle Constraint, $m = -0.5$

The triangle-constraint regions have a larger area than the corresponding box-constraint regions in this instance, although this is not necessarily always true.

As can be seen from time domain figures of transient responses in this chapter, this method is best used constructively. In systems with multiple complex pole pairs, straightforward calculation of response oscillation and damping ratio may not be feasible. However, using this method allows the designer to use a single variable (ω_0 or m) which can be used as part of multi-objective optimization.

5. CONCLUSIONS

This thesis presented a method for confining the poles of a PI or PID-controlled continuous-time linear time invariant system into an arbitrary region of the complex plane, $\overline{\mathcal{S}}$. This is a novel application of Neimark's D-decomposition: specify $\overline{\mathcal{S}}$ on the complex plane, and use a modified D-decomposition approach to mape this region into parameter space. This mapping will divide the parameter space into regions, which can then be tested to determine whether $\overline{\mathcal{S}}$ is achievable for the given plant and controller. Taking advantage of the Edge Theorem, a polytope may be inscribed into the resulting parameter space region both for convenience and provide additional verification that the desired root space constraint is met for all points within the polytope. This allows the designer to reduce the response oscillation frequency and increase damping ratio, both time domain characteristics.

The method presented here is for single-input, single-output (SISO) continuous-time plants. Additionally, it is a model-based method, and so is dependent on the designer having an accurate characterization of the plant being controlled. The most logical next development for this method is use in discrete-time problems, and for multi-input/multi-output (MIMO) systems.

REFERENCES

[1] I. D. Díaz-Rodríguez and S. P. Bhattacharyya, “Pi controller design in the achievable gain-phase margin plane,” in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 4919–4924, IEEE, 2016.

[2] S. Han, L. H. Keel, and S. P. Bhattacharyya, “PID controller design with an H_∞ criterion,” in *Proc. 3rd IFAC Conf. Adv. Proportional Integral Derivative Control*, pp. 400–405, 2018.

[3] S. Han and S. Bhattacharyya, “Pid controller synthesis using a σ-hurwitz stability criterion,” *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 525–530, 2018.

[4] M. Tomizuka and S. Zhang, “Modelling and Conventional/Adaptive PI Control of a Lathe Cutting Process,” in *1985 American Control Conference*, pp. 745–750, June 1985.

[5] M.-T. Ho, A. Datta, and S. Bhattacharyya, “A linear programming characterization of all stabilizing pid controllers,” in *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, vol. 6, pp. 3922–3928, IEEE, 1997.

[6] S. P. Bhattacharyya, A. Datta, and L. H. Keel, *Linear control theory: structure, robustness, and optimization*, vol. 33. CRC press, 2009.

[7] I. D. Diaz-Rodriguez, S. Han, and S. Bhattacharyya, *Analytical Design of PID Controllers*. Springer, 2019.

[8] H. S. Khan and M. B. Kadri, “Attitude and altitude control of quadrotor by discrete pid control and non-linear model predictive control,” in *2015 International Conference on Information and Communication Technologies (ICICT)*, pp. 1–11, IEEE, 2015.

[9] N. Pachauri, V. Singh, and A. Rani, “Two degree of freedom pid based inferential control of continuous bioreactor for ethanol production,” *ISA transactions*, vol. 68, pp. 235–250, 2017.

[10] D. Hrovat and J. Sun, “Models and control methodologies for ic engine idle speed control design,” *Control Engineering Practice*, vol. 5, no. 8, pp. 1093–1100, 1997.

[11] R. Sharma, P. Gaur, and A. Mittal, “Performance analysis of two-degree of freedom fractional order pid controllers for robotic manipulator with payload,” *ISA transactions*, vol. 58, pp. 279–291, 2015.

[12] Y. I. Neimark, “Stability of linearized systems,” *LKVVIA, Leningrad*, p. 140, 1949.

[13] D. Mitrović, “Graphical analysis and synthesis of feedback control systems i-theory and analysis,” *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry*, vol. 77, no. 6, pp. 476–487, 1959.

[14] D. Siljak, “Generalization of mitrovic's method,” *IEEE Transactions on Applications and Industry*, vol. 83, no. 74, pp. 314–320, 1964.

[15] R. N. Tantaris, L. H. Keel, and S. P. Bhattacharyya, “H/sub/spl infin//design with first-order controllers,” *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1343–1347, 2006.

[16] A. C. Bartlett, C. V. Hollot, and H. Lin, “Root locations of an entire polytope of polynomials: It suffices to check the edges,” *Mathematics of Control, Signals and Systems*, vol. 1, no. 1, pp. 61–71, 1988.

[17] Y. I. Neimark, “D-partition and robust stability,” *Computational Mathematics and Modeling*, vol. 9, no. 2, pp. 160–166, 1998.

[18] E. N. Gryazina and B. T. Polyak, “Stability regions in the parameter space: D-decomposition revisited,” *Automatica*, vol. 42, no. 1, pp. 13–26, 2006.

[19] S. E. Hamamci, “An algorithm for stabilization of fractional-order time delay systems using fractional-order pid controllers,” *IEEE Transactions on Automatic Control*, vol. 52, no. 10, pp. 1964–1969, 2007.

[20] P. D. Mandić, T. B. Šekara, M. P. Lazarević, and M. Bošković, “Dominant pole placement with fractional order pid controllers: D-decomposition approach,” *Isa Transactions*, vol. 67, pp. 76–86, 2017.

[21] H.-C. Yu, T.-C. Chen, and C.-S. Liu, "Adaptive fuzzy logic proportional-integral-derivative control for a miniature autofocus voice coil motor actuator with retaining force," *IEEE Transactions on Magnetics*, vol. 50, no. 11, pp. 1–4, 2014.