

Synchronisation by mutual time delayed
feedback: An analytic case study for coupled
maps

Matthew James Morrison 110260068
MTHM038

Supervisor: Dr Wolfram Just
School of Mathematical Sciences

2015/16

Specific Learning Difficulties Cover Note

Student ID: 110260068

Assessors and Examiners:

Guidelines for Markers Assessing Coursework and Examinations of Students Diagnosed with Specific Learning Difficulties

As far as possible examiners are asked to mark exam scripts sympathetically, ignoring dyslexic/dyspraxic errors and focusing on content and understanding of the subject.

Dyslexia and Dyspraxia may affect student performance in the following ways:

- Spelling, grammar and punctuation may be inaccurate
- Organisation of ideas may be confused affecting the overall structure of written work
- Handwriting may be untidy with inappropriate capital letters
- Proof reading may be weak with errors undetected.
- The use of spell and grammar-checkers is not always effective as word substitutions, phonetic equivalent and American spelling errors may occur

Under examination conditions, these difficulties are likely to be exacerbated. Errors are likely to become more marked towards the end of scripts.

Useful approaches can include:

- Read passage quickly for content
- Include positive/constructive comments
- Use clear English and when correcting explain what is wrong and give examples
- Use non-red coloured pens for comments/corrections.

For more information regarding marking guidelines see DDS webpage <http://qm-web.dds.qmul.ac.uk/staff/index.html> and open link [Institutional Marking Practices for Dyslexic Students](#)

Disability and Dyslexia Services:

Simon Jarvis
Head of Disability and Dyslexia Services
s.jarvis@qmul.ac.uk

Grazia Bevere
Senior Dyslexia Tutor
g.bevere@qmul.ac.uk

Alteration or misuse of this document will result in disciplinary action

Abstract

The control of unstable fixed point solutions in discrete time dynamical systems, with time delayed feedback control, will be explored by analytical methods. By means of linear stability, we will compute the relevant coupling gains needed for successful control. This concept will be applied to study the synchronisation properties of two mutually coupled non identical maps with time delayed feedback control.

DECLARATION

I (**Matthew James Morison**),

hereby declare that this thesis is entirely my own work, and that all sources have been fully acknowledged. I am aware that plagiarism is a serious offence, and, if detected, could result in a mark of zero for the entire project (and possible expulsion from the MSc degree programme as well, which may prevent me from resubmitting at a later date).



13/09/2016

Signed

Date

ACKNOWLEDGEMENT:

I would like to dedicate this to my parents Eleanor and John (Mummy and Daddy). You have both been a saving grace throughout my academic career and life. I hope to one day be able to support you in the numerous ways you

have supported me.

All my love - Mattz.

I would also like to especially thank my supervisor Dr Wolfram Just, whom I have also had as a lecturer for the past two years. His teaching and philosophies have been nothing short of inspiring, making 9am lectures less of a chore to attend, and making mathematics more interesting than it already is. Indeed, throughout this thesis, his thoughtful guidance has been a great help - Thank you.

“The difference in judgement between you and I originates from different rules derived from past experience.” - Rivaille (Shingeki no Kyojin)

Contents

1	Introduction	8
2	Discrete Dynamical Systems and Fixed Points	9
2.1	Preliminaries	9
2.2	Solving Linear systems	11
2.2.1	One-Dimensional Case	12
2.2.2	Two-Dimensional Case	12
3	1-D Case with Time-Delayed Feedback Control (TDFC)	14
3.1	Deriving the Characteristic Equation	14
3.2	Results	16
4	Jury's Criterion	19
4.1	Polynomial Roots	19
4.2	Jury's Stability Test	19
5	2-D Case with TDFC	22
5.1	Deriving the Characteristic Equation	22
5.2	Application of Jury's Test	23
5.2.1	Pictorial overview of (C1-C5)	25
5.2.2	Analysis of coupled system with 'zero control'	27
5.2.3	Graphs of regions of stability with $-1 \leq K_1 K_2 \leq 0.95$ (in steps of 0.05)	28
5.2.4	Analysis of regions of stability with $-1 \leq K_1 K_2 \leq 1$	32
5.2.5	Results	35
6	Conclusion	36
A	Appendix - Matlab codes for Figures	38

1 Introduction

Having the means to influence the behaviour of something over a period of time is what is known as ‘control’. A dynamical system is something that changes over a period of time[1]. It achieves this by interacting with its environment, and then displaying emerging behaviour indicative of some form of stimulation. These stimulants are known as inputs, and the emerging behaviour outputs. Sometimes these outputs are not desirable and we wish to exert control in order to bring these systems to an ‘ideal state’. Consider for instance driving in a 70mph zone with cruise control, and wanting to reach one’s destination with urgency, whilst remaining within the boundaries of the law. The desirable speed is 70mph. If one’s speed is too fast or too slow, or if the trajectory of the car encounters inclines and declines, the cruise control will make necessary adjustments in order to bring the speed to 70mph. This is an example of stability. The cruise control maintains the speed in the ideal state whilst perturbations are acting on it. In addition to this, if for example the car is going 140mph, the cruise control will not exert the breaks harshly, as this could cause damage to the passengers and waste energy, amongst other safety issues. But it would slow down the vehicle in a steady and timely manner. This is known as flexibility. It is more beneficial to let the system slowly bring itself to the desired state using as little energy as possible [16]. This concept serves as an analogy for the essence of this thesis.

In this dissertation we take difference equations (maps) and study the stability of their fixed point. We choose difference equations as opposed to differential equations (continuous maps), as we gain an insight into some of the principal elements of differential equations with more ease. Our aim is to be able to control unstable solutions of these discrete maps by implementing a mechanism called **time delayed feedback control (TDFC)** - which was first proposed by Pyragas in 1992 [8], thanks to Ott’s seminal work [18] - in order to bring our system to a desired state, namely one that is stable.

The second chapter gives us a brief overview of dynamical systems, and provides the foundation of notation and definitions upon which the thesis is built.

The third and fifth chapters give us the one-dimensional (1-D) and two-dimensional (2-D) analytical case study of our systems with TDFC, and uses graphical evidence to explore conditions for which our systems can be stabilised.

The fourth chapter presents Jury's Criterion, which adds a powerful tool to our arsenal, enabling us to tackle the 2-D study case in a more elegant fashion.

2 Discrete Dynamical Systems and Fixed Points

This chapter gives a brief overview of discrete dynamical systems, and provides us with the foundation upon which this thesis is built. It is based mainly on material found in these books [2, 3, 4, 5], which we have adapted in order to tailor it to the purposes of this work.

2.1 Preliminaries

There are two types of dynamical systems, namely continuous-time and discrete-time, of which the latter will be our main focus. A **discrete-time dynamical system (map)** is a system that evolves over time according to a specific rule. For example imagine different sized populations of bacteria in different petri dishes which double in size per time interval. A map is of the form:

$$\vec{x}_{n+1} = f(\vec{x}_n), \quad f : \mathbb{R}^m \mapsto \mathbb{R}^m, \quad \vec{x}_n \in \mathbb{R}^m, \quad n \in \mathbb{N}_0 \quad (1)$$

Here \vec{x}_n would be the vector denoting the populations of the bacteria at time n , and f would denote the doubling function.

We will be studying **linear maps** on the reals, which, by definition, have output values proportional to their input values, and pass through the origin (maps whose output values are proportional to their input values, but do not pass through the

origin are called 'affine'[6]). Linear maps are of the form:

$$\vec{x}_{n+1} = A\vec{x}_n, \quad A \in \mathbb{R}^{m \times m}, \quad \vec{x}_n \in \mathbb{R}^m, \quad n \in \mathbb{N}_0 \quad (2)$$

where we are given \vec{x}_0 , called an **initial vector**, and A is the $m \times m$ matrix,

$$A = (a_{ij}) = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mm} \end{pmatrix} \quad (3)$$

Where a_{ij} represents the element in the i -th row and j -th column.

The **determinant** of a square matrix A is denoted $\det(A)$. It associates a scalar to a matrix, which can be used to provide information about systems of linear equations[9].

The determinant of a 2×2 matrix A is given by the formula:

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21} \quad (4)$$

The determinant of a 3×3 matrix A is given by the formula:

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \quad (5)$$

The determinant of a 4×4 matrix A is given by the formula:

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix} = \quad (6)$$

$$a_{11} \begin{vmatrix} a_{22} & a_{23} & a_{24} \\ a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} & a_{24} \\ a_{31} & a_{33} & a_{34} \\ a_{41} & a_{43} & a_{44} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} & a_{24} \\ a_{31} & a_{32} & a_{34} \\ a_{41} & a_{42} & a_{44} \end{vmatrix} - a_{14} \begin{vmatrix} a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{vmatrix}$$

2.2 Solving Linear systems

One of our main areas of interest concerning maps is fixed points and their stability. Here we demonstrate how to solve 1-D and 2-D linear systems.

A **fixed point** occurs when $f(\vec{x}_*) = \vec{x}_*$, i.e., if $\vec{x}_0 = \vec{x}_*$ then $\vec{x}_n = \vec{x}_*$, for all $n \geq 1$. In terms of the example of bacteria, we can view this as having 0 bacteria in all our petri dishes, as our initial vector (i.e., bacteria will not spontaneously appear). For all linear maps the origin $\vec{x}_n = \vec{0}$ is a fixed point.

Given an initial vector \vec{x}_0 , we look at the iterates of Eq. (2) to find a general solution:

$$\begin{aligned} \vec{x}_1 &= A\vec{x}_0 \\ \vec{x}_2 &= A\vec{x}_1 = A^2\vec{x}_0 \\ \vec{x}_3 &= A\vec{x}_2 = A^2\vec{x}_1 = A^3\vec{x}_0 \\ \vec{x}_4 &= A\vec{x}_3 = A^2\vec{x}_2 = A^3\vec{x}_1 = A^4\vec{x}_0 \\ &\vdots \\ \vec{x}_n &= A^n\vec{x}_0 \end{aligned} \quad (7)$$

We determine the stability of a fixed point by analysing at the behaviour of points of the map in a small neighbourhood of the fixed point, and note whether repeated iterations converge towards or diverge away from it. The fixed points are called:

1. **stable/attracting** if the iterations converge towards it.

2. **unstable/repelling** if the iterations diverge away from it.

2.2.1 One-Dimensional Case

In the 1-D case, the general solution given by Eq. (7) translates to,

$$x_n = a^n x_0, \quad x_0, a \in \mathbb{R} \quad (8)$$

Example 1. Consider the real valued linear map

$$f(x_n) = ax_n \quad (9)$$

with a being constant. We have that $x_* = 0$ is a fixed point of the map, and $f^n(x_n) = a^n x_0$ is a solution of the map. Assuming $x_0 \neq 0$, it can be seen that if:

1. $|f'(x_*)| = |a| < 1$, then repeated iterations tend to 0, and the system is **stable**.
2. $|f'(x_*)| = |a| = 1$, then repeated iterations stay fixed, and the system is **not stable**.
3. $|f'(x_*)| = |a| > 1$, then repeated iterations will diverge away from 0, and the system is **unstable**.

2.2.2 Two-Dimensional Case

In a 2-D map, we have a system as follows,

$$\begin{aligned} x_{n+1} &= f(x_n, y_n) \\ y_{n+1} &= g(y_n, x_n) \end{aligned} \quad x_n, y_n \in \mathbb{R}; f, g : \mathbb{R} \mapsto \mathbb{R} \quad (10)$$

The system is said to be **mutually coupled** if an x is in the y equation and a y is in the x equation. We add a small pertubation, letting $x_n = x_* + \delta x_n$ and $y_n = y_* + \delta y_n$ be a point close to the fixed point $(x_*, y_*) = (0, 0)$. We then use Linearisation to expand Eq.(10) in the neighbourhood of the fixed point[15], and

see whether future iterations converge towards, or diverge away from it. We have:

$$\begin{aligned} f(x_n, y_n) &= f(x_*, y_*) + f_x(x_*, y_*)\delta x_n + f_y(x_*, y_*)\delta y_n \\ g(x_n, y_n) &= g(x_*, y_*) + g_x(x_*, y_*)\delta x_n + g_y(x_*, y_*)\delta y_n \end{aligned} \quad (11)$$

Where $f_x(x_n, y_n) = \frac{\partial f}{\partial x}$ evaluated at the point (x_n, y_n) . Noting that $x_n = x_* + \delta x_n$ and $y_n = y_* + \delta y_n$ implies $x_{n+1} = x_* + \delta x_{n+1}$ and $y_{n+1} = y_* + \delta y_{n+1}$, we now substitute Eq. (11) into Eq. (10) yielding:

$$\begin{aligned} x_* + \delta x_{n+1} &= f(x_*, y_*) + f_x(x_*, y_*)\delta x_n + f_y(x_*, y_*)\delta y_n \\ y_* + \delta y_{n+1} &= g(x_*, y_*) + g_x(x_*, y_*)\delta x_n + g_y(x_*, y_*)\delta y_n \end{aligned} \quad (12)$$

We know that $x_* = f(x_*, y_*)$ and $y_* = g(x_*, y_*)$, so they cancel out and Eq. (12) becomes the linearised system:

$$\begin{aligned} \delta x_{n+1} &= f_x(x_*, y_*)\delta x_n + f_y(x_*, y_*)\delta y_n \\ \delta y_{n+1} &= g_x(x_*, y_*)\delta x_n + g_y(x_*, y_*)\delta y_n \end{aligned} \quad (13)$$

By defining $\vec{\delta x}_n := (\delta x_n, \delta y_n)^T$, We can transform Eq. (13) into a matrix equation:

$$\begin{pmatrix} \delta x_{n+1} \\ \delta y_{n+1} \end{pmatrix} = \underbrace{\begin{pmatrix} f_x(x_*, y_*) & f_y(x_*, y_*) \\ g_x(x_*, y_*) & g_y(x_*, y_*) \end{pmatrix}}_{\mathbf{J}} \begin{pmatrix} \delta x_n \\ \delta y_n \end{pmatrix} \quad (14)$$

Where \mathbf{J} is known as the **Jacobian matrix**. It is the multivariable version of $f'(x_*)$ for the 1-D case.

In order to determine the stability of (10) we compute the eigenvalues (λ) of the Jacobian as follows [9]:

$$|J - \lambda I| = \begin{vmatrix} f_x(x_*, y_*) - \lambda & f_y(x_*, y_*) \\ g_x(x_*, y_*) & g_y(x_*, y_*) - \lambda \end{vmatrix} = 0 \quad (15)$$

$$P(\lambda) = \lambda^2 - (f_x(x_*, y_*) + g_y(x_*, y_*))\lambda - f_y(x_*, y_*)g_x(x_*, y_*) = 0$$

In order for a map, including Eq.(10) to be stable we require that **all eigenvalues**

are less than 1 in magnitude.[13]

Example 2. Consider the following system

$$\begin{aligned}x_{n+1} &= f(x_n, y_n) = 3x_n + y_n \\y_{n+1} &= g(x_n, y_n) = 2x_n + 4y_n\end{aligned}\tag{16}$$

The fixed point is at $(x_*, y_*) = (0, 0)$. The corresponding Jacobian matrix is

$$\begin{vmatrix}3 - \lambda & 1 \\2 & 4 - \lambda\end{vmatrix} = \lambda^2 - 7\lambda + 10 = (\lambda - 5)(\lambda - 2) = 0\tag{17}$$

The system has eigenvalues $\lambda_1 = 5$ and $\lambda_2 = 2$, which means that the fixed point $(x_*, y_*) = (0, 0)$, is unstable as they are both larger than 1 in magnitude.

In essence this thesis requires us to take systems and reduce them to eigenvalue problems, then impose restrictions in order to ensure that the corresponding eigenvalues are less than 1 in absolute value in order to achieve stability.

3 1-D Case with Time-Delayed Feedback Control (TDFC)

3.1 Deriving the Characteristic Equation

Suppose we have a machine that produces some unstable time dependent output. For the purposes of this thesis we require this output to be periodic. We do this by measuring the signal to take what one calls ‘time delayed difference’[8], that is, one takes the a proportion of the difference of the previous output with the current output and feeds this back into the system[17]. For correct choices of time delay, the system becomes periodic, which means the control goes to zero [10],[19]. A positive consequence is that this does not alter the dynamics of the unstable system, but it causes the trajectories of neighbouring points to converge towards it. Another benefit of such a method is that one need not know any

detailed information about the system, just the previous output[20]. Keeping with the theme of ‘flexibility’, one wishes to effectively stabilise something with as minimal energy as possible.

As before, we are dealing with linear maps of the form in Eq.(2) but in 1-D, i.e. maps of the form:

$$x_{n+1} = f(x_n) = ax_n, \quad a, x_n \in \mathbb{R}, \quad n \in \mathbb{N}_0 \quad (18)$$

Now we assume that our system is unstable, i.e. $|a| > 1$. We are in the situation where our machine produces a signal and we want to introduce feedback, which will stabilise our system. The time delayed difference we shall introduce to our system is $K(x_n - x_{n-1})$ [17], with $K \in \mathbb{R}$ as our **control gain**. The system now reads:

$$x_{n+1} = f(x_n) + K(x_n - x_{n-1}) = ax_n + K(x_n - x_{n-1}) \quad (3.1)$$

The aim of this section is to find out for what values of K stabilises this system. This will eventually surface as a solution to an eigenvalue problem.

Firstly note that adding the time delayed difference increases the dimension of our system by unity[11]. Taking the fixed point of the system without TDFC $x_* = 0$, we linearise about x_* with $\delta x_n = x_n - x_*$ to get the linearised system:

$$\begin{pmatrix} \delta x_{n+1} \\ \delta x_n \end{pmatrix} = \begin{pmatrix} a + K & -K \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \delta x_n \\ \delta x_{n-1} \end{pmatrix} \quad (19)$$

Note that $K(x_n - x_{n-1}) = K(\delta x_n - \delta x_{n-1})$ [12].

Next, we find the determinant of the following matrix:

$$\begin{vmatrix} a + K - \lambda & -K \\ 1 & -\lambda \end{vmatrix} \quad (20)$$

which gives rise to the characteristic equation:

$$P(\lambda) = -\lambda(a + K - \lambda) + K = \lambda^2 - (a + K)\lambda + K = 0 \quad (21)$$

Now, utilising the quadratic equation, we can see that:

$$\lambda_{1/2} = \frac{a + K \pm \sqrt{(a + K)^2 - 4K}}{2} \quad (22)$$

In order for stability to be achieved, we require $|\lambda_{1/2}| < 1$. In (Figure 1)¹ we provide a graphical representation of the following inequalities:

$$\begin{aligned} |\lambda_1| &= \left| \frac{a + K + \sqrt{(a + K)^2 - 4K}}{2} \right| < 1 \\ |\lambda_2| &= \left| \frac{a + K - \sqrt{(a + K)^2 - 4K}}{2} \right| < 1 \end{aligned} \quad (23)$$

3.2 Results

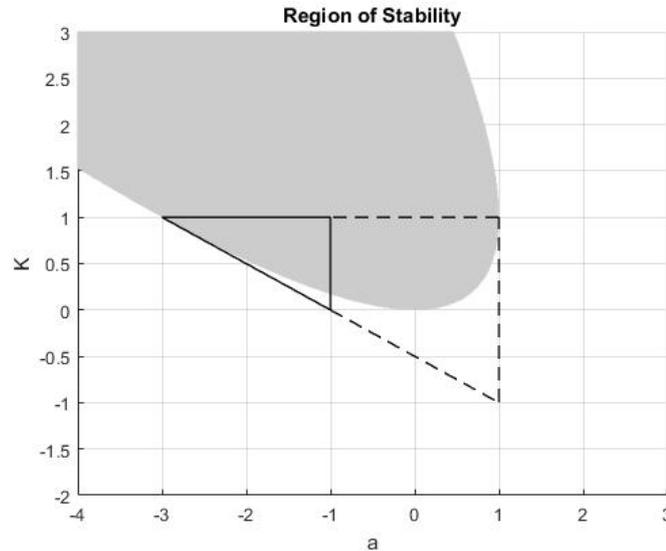


Figure 1: **Plot of region of stability for 1-D case:** The boundary of the largest triangle represents the border line cases $|\lambda_{1/2}| = 1$. Stability is achieved in the interior. Furthermore, we introduce TDFC when $|a| > 1$, which is represented by interior of the smaller right-angled triangle with solid lines. The shaded area is indicative of when the eigenvalues are complex conjugates, i.e. the discriminant $(a + K)^2 - 4K < 0$.

(Figure 1) provides us with graphical proof which shows that when our system is

¹See Appendix for Matlab code

unstable, there exists some $K \in (0, 1)$ for which the system can be stabilised if $a \in (-3, -1]$.

Example 3. Let $a = -2$. This corresponds to a half of a co-ordinate, which when paired a complementary control gain co-ordinate K , would lie within the boundaries of the solid triangle in (Figure 1).

$$f(x_n) = -2x_n \quad (24)$$

Let the initial seed $x_0 = 1$. We have:

$$x_0 = 1 \quad (25)$$

$$x_1 = -2 \times 1 = -2$$

Now we are in the predicament where our system is unstable as $|a| > 1$, so we introduce a complementary feedback of $K = 0.75$ and our system will stabilise.

Our system now reads:

$$f(x_n) = -2x_n + 0.75(x_n - x_{n-1}), \quad n = 1, 2, 3, \dots \quad (26)$$

Continuing our iterates now with TDFC, we have:

$$\begin{aligned} x_2 &= ax_1 + K(x_1 - x_0) = 4 + (0.75 \times -3) = 1.75 \\ x_3 &= ax_2 + K(x_2 - x_1) = -3.5 + (0.75 \times 3.75) = -0.6875 \\ x_4 &= ax_3 + K(x_3 - x_2) = 1.375 + (0.75 \times -2.4375) = -0.453125 \\ x_5 &= ax_4 + K(x_4 - x_3) = 0.90625 + (0.75 \times 0.234375) = 1.08203125 \\ x_6 &= -1.0127 \\ x_7 &= 0.4543 \\ x_8 &= 0.1916 \\ x_9 &= -0.5802 \end{aligned} \quad (27)$$

In (Figure 2)² we have plotted the first 50 iterates of this example, and we can see,

²See Appendix for Matlab code

that even though our system was initially unstable, after introducing the TDFC mechanism, the trajectory converges to the fixed point, and hence the system has been stabilised.

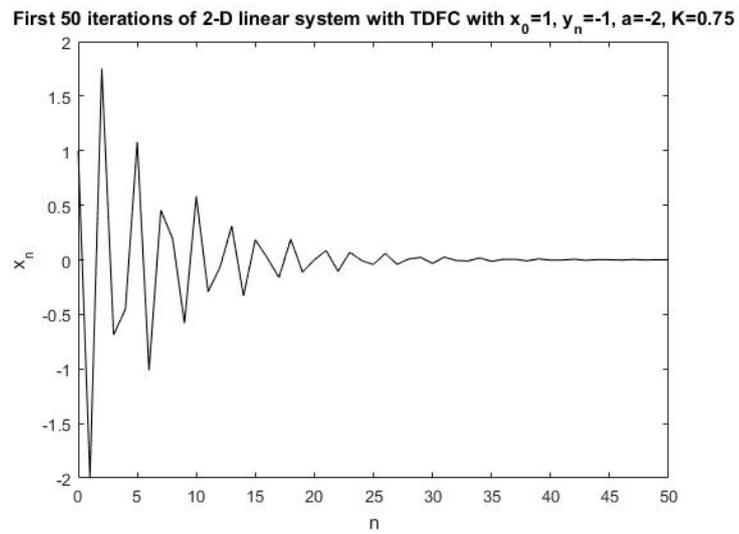


Figure 2: Plot of the first 50 iterates of 2-D system with TDFC with $x_0 = 1, a = -2$ and $K = 0.75$

4 Jury's Criterion

4.1 Polynomial Roots

Finding the roots of the characteristic equation for the 1-D case requires no more than the quadratic equation, which is short, and most mathematicians will have memorised. However, for the higher dimensional case, one needs a more sophisticated method, as it involves finding the roots of a quartic equation, which is long and cumbersome. Jury's Criterion provides an alternative solution. It gives us necessary and sufficient criteria for polynomials to have all their roots confined within the unit circle[7]. In context, this ensures that the eigenvalues of our system are less than one in magnitude, and guarantees stability.

4.2 Jury's Stability Test

For the characteristic polynomial

$$P(\lambda) = a_0\lambda^0 + a_1\lambda^1 + a_2\lambda^2 + \dots + a_n\lambda^n \quad a_n > 0$$

We construct Jury's Table [7] as follows:

λ^0	λ^1	λ^2	\dots	λ^{n-k}	\dots	λ^{n-2}	λ^{n-1}	λ^n
a_0	a_1	a_2	\dots	a_{n-k}	\dots	a^{n-2}	a_{n-1}	a_n
a_n	a_{n-1}	a_{n-2}	\dots	a_k	\dots	a_2	a_1	a_0
b_0	b_1	b_2	\dots	b_{n-k}	\dots	b_{n-2}	b_{n-1}	
b_{n-1}	b_{n-2}	b_{n-3}	\dots	b_{k-1}	\dots	b_1	b_0	
c_0	c_1	c_2	\dots	c_{n-k}	\dots	c_{n-2}		
c_{n-2}	c_{n-3}	c_{n-4}	\dots	c_{k-2}	\dots	c_0		
\vdots	\vdots	\vdots	\vdots	\ddots				
r_0	r_1	r_2	r_3					
r_3	r_2	r_1	r_0					
s_0	s_1	s_2						
s_2	s_1	s_0						

Table 1: Jury's Table

The first row is composed of the coefficients of the characteristic polynomial, and every even row is identical to the row above it, but in reverse order. The elements in odd rows greater than one are calculated by the following determinants:

$$b_k = \begin{vmatrix} a_0 & a_{n-k} \\ a_n & a_k \end{vmatrix}, c_k = \begin{vmatrix} b_0 & b_{n-k-1} \\ b_{n-1} & b_k \end{vmatrix}, d_k = \begin{vmatrix} c_0 & c_{n-k-2} \\ c_{n-2} & c_k \end{vmatrix}, \dots$$

Theorem 4.1. Jury's Criteria

In order for the characteristic polynomial

$$P(\lambda) = a_0\lambda^0 + a_1\lambda^1 + a_2\lambda^2 + \dots + a_n\lambda^n, \quad a_n > 0, n \geq 2$$

to have all its roots confined within the unit circle, the necessary and sufficient conditions of Jury's Test are:

- 1) $P(1) > 0$
- 2) $(-1)^n P(-1) > 0$
- 3) $a_n > |a_0|$
- 4) $|b_0| > |b_{n-1}|$
- 5) $|c_0| > |c_{n-2}|$
- 6) $|d_0| > |d_{n-3}|$
- 7) $|e_0| > |e_{n-4}|$
- \vdots
- $n + 1) |s_0| > |s_2|$

For a proof of the theorem, please see [14]

Example 4. *Using the quadratic characteristic Eq. (21) from the previous chapter $P(\lambda) = \lambda^2 - (a + K)\lambda + K$, we use Jury's Criteria, to verify the stability conditions. The Criteria read:*

- 1) $P(1) = 1 - a > 0$
- 2) $P(-1) = 1 + a + 2K > 0$
- 3) $|K| < 1$

Plotting these inequalities, we gain the same region of stability (depicted in black) as in (Figure 1).

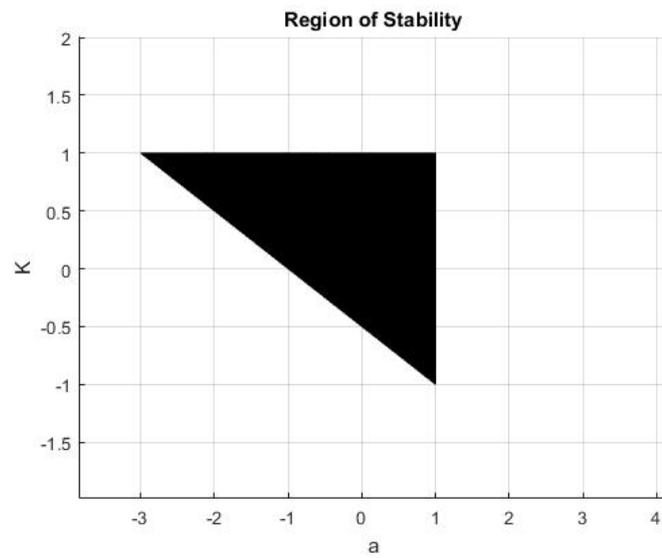


Figure 3: Plot of region of stability for 1-D case using Jury's Criterion

5 2-D Case with TDFC

5.1 Deriving the Characteristic Equation

Now we take two linear maps and couple them via TDFC, and we look to find under what conditions the system can be stabilised. With $a, b, x_n, y_n, K_1, K_2 \in \mathbb{R}$, we have the following system:

$$\begin{aligned}x_{n+1} &= ax_n + K_1(y_n - y_{n-1}) \\y_{n+1} &= by_n + K_2(x_n - x_{n-1})\end{aligned}\tag{5.1}$$

By adding the TDFC into the system we have gone from a 2 to a 4-dimensional system, with a fixed point $(x_*, y_*) = (0, 0)$, and as before, we let $x_n = x_* - \delta x_n$ and $y_n = y_* - \delta y_n$ be in the neighbourhood of our fixed point and linearise about that point to get the system:

$$\begin{pmatrix} \delta x_{n+1} \\ \delta x_n \\ \delta y_{n+1} \\ \delta y_n \end{pmatrix} = \underbrace{\begin{pmatrix} a & 0 & K_1 & -K_1 \\ 1 & 0 & 0 & 0 \\ K_2 & -K_2 & b & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_A \begin{pmatrix} \delta x_n \\ \delta x_{n-1} \\ \delta y_n \\ \delta y_{n-1} \end{pmatrix}\tag{28}$$

Next we calculate the determinant $\det(A - \lambda I)$ in order to bring it to an eigenvalue problem.

$$\begin{vmatrix} a - \lambda & 0 & K_1 & -K_1 \\ 1 & -\lambda & 0 & 0 \\ K_2 & -K_2 & b - \lambda & 0 \\ 0 & 0 & 1 & -\lambda \end{vmatrix} = (a - \lambda) \begin{vmatrix} -\lambda & 0 & 0 \\ -K_2 & b - \lambda & 0 \\ 0 & 1 & -\lambda \end{vmatrix} + K_1 \begin{vmatrix} 1 & -\lambda & 0 \\ K_2 & -K_2 & 0 \\ 0 & 0 & -\lambda \end{vmatrix} + K_1 \begin{vmatrix} 1 & -\lambda & 0 \\ K_2 & -K_2 & b - \lambda \\ 0 & 0 & 1 \end{vmatrix} = \tag{29}$$

$$\begin{aligned}
 & (\lambda^2 - a\lambda) \begin{vmatrix} b - \lambda & 0 \\ 1 & -\lambda \end{vmatrix} + K_1 \begin{vmatrix} -K_2 & 0 \\ 0 & -\lambda \end{vmatrix} + K_1\lambda \begin{vmatrix} K_2 & 0 \\ 0 & -\lambda \end{vmatrix} + K_1 \begin{vmatrix} -K_2 & b - \lambda \\ 0 & 1 \end{vmatrix} + \\
 & \qquad \qquad \qquad K_1\lambda \begin{vmatrix} K_2 & b - \lambda \\ 0 & 1 \end{vmatrix} \\
 & = (\lambda^2 - a\lambda)(\lambda^2 - b\lambda) + K_1(K_2\lambda - K_2\lambda^2) + K_1(K_2\lambda - K_2) = \\
 & P(\lambda) = \lambda^4 - (a + b)\lambda^3 + (ab - K_1K_2)\lambda^2 + 2K_1K_2\lambda - K_1K_2 = 0
 \end{aligned}$$

5.2 Application of Jury’s Test

We have that the characteristic equation for the two dimensional system Eq.(5.1) is given by:

$$P(\lambda) = \lambda^4 - (a + b)\lambda^3 + (ab - K_1K_2)\lambda^2 + 2K_1K_2\lambda - K_1K_2 = 0 \quad (30)$$

Now we construct the Jury’s Table as in chapter 4 as follows:

λ^0	λ^1	λ^2	λ^3	λ^4
a_0	a_1	a_2	a_3	a_4
a_4	a_3	a_2	a_1	a_0
b_0	b_1	b_2	b_3	
b_3	b_2	b_1	b_0	
c_0	c_1	c_2		
c_2	c_1	c_0		

Table 2: Jury’s Table for quartic characteristic Eq(30)

with:

$$a_0 = -K_1K_2,$$

$$a_1 = 2K_1K_2,$$

$$a_2 = ab - K_1K_2,$$

$$a_3 = -(a + b),$$

$$a_4 = 1,$$

$$b_0 = \begin{vmatrix} -K_1K_2 & 1 \\ 1 & -K_1K_2 \end{vmatrix} = K_1^2K_2^2 - 1,$$

$$b_1 = \begin{vmatrix} -K_1K_2 & -(a + b) \\ 1 & 2K_1K_2 \end{vmatrix} = a + b - 2K_1^2K_2^2,$$

$$b_2 = \begin{vmatrix} -K_1K_2 & ab - K_1K_2 \\ 1 & ab - K_1K_2 \end{vmatrix} = K_1^2K_2^2 + (1 - ab)K_1K_2 - ab, \quad (31)$$

$$b_3 = \begin{vmatrix} -K_1K_2 & 2K_1K_2 \\ 1 & -(a + b) \end{vmatrix} = (a + b - 2)K_1K_2,$$

$$c_0 = \begin{vmatrix} K_1^2K_2^2 - 1 & (a + b - 2)K_1K_2 \\ (a + b - 2)K_1K_2 & K_1^2K_2^2 - 1 \end{vmatrix} =$$

$$K_1^4K_2^4 - (a^2 + b^2 + 2ab - 4a - 4b + 6)K_1^2K_2^2 + 1,$$

$$c_2 = \begin{vmatrix} K_1^2K_2^2 - 1 & a + b - 2K_1^2K_2^2 \\ (a + b - 2)K_1K_2 & K_1^2K_2^2 + (1 - ab)K_1K_2 - ab \end{vmatrix} =$$

$$K_1^4K_2^4 + (2a + 2b - ab - 3)K_1^3K_2^3 - (ab + 1)K_1^2K_2^2 -$$

$$(a^2 + b^2 + ab - 2a - 2b + 1)K_1K_2 + ab$$

Utilising Jury's Criterion from chapter 4, the necessary and sufficient conditions in order for this mutually coupled linear system to have eigenvalues within the unit circle, and hence achieve stability are:

$$1. P(1) > 0 \implies ab - a - b + 1 > 0 \quad (\text{C1})$$

$$2. P(-1) > 0 \implies ab + a + b - 4K_1K_2 + 1 > 0 \quad (\text{C2})$$

$$3. |a_0| < a_4 \implies |K_1K_2| < 1 \quad (\text{C3})$$

$$4. |b_0| > |b_3| \implies |K_1^2K_2^2 - 1| > |(a + b - 2)K_1K_2| \quad (\text{C4})$$

$$5. |c_0| > |c_2| \implies |K_1^4K_2^4 - (a^2 + b^2 + 2ab - 4a - 4b + 6)K_1^2K_2^2 + 1| > \\ |K_1^4K_2^4 + (2a + 2b - ab - 3)K_1^3K_2^3 - (ab + 1)K_1^2K_2^2 - \\ (a^2 + b^2 + ab - 2a - 2b + 1)K_1K_2 + ab| \quad (\text{C5})$$

It can be seen that through Eqs.(C1-C5), K_1 and K_2 are always present multiplied together. Therefore taking into account Eq.(C3) = $|K_1K_2| < 1$, we can treat the control gains as a single variable. This also means that if one does find conditions where stability is successful i.e, for $K_1K_2 = -0.2$, then we can choose varying controls e.g. $K_1 = -2$ and $K_2 = 0.1$, or $K_1 = 20$ and $K_2 = -0.01$. In addition to this the controls could be exchanged ($K_1 := K_2$) and ($K_2 := K_1$), and this would not effect the stability of our system.

5.2.1 Pictorial overview of (C1-C5)

Now in (Figure A)³ we shall provide snapshots of the criteria established above.

³See Appendix for Matlab code

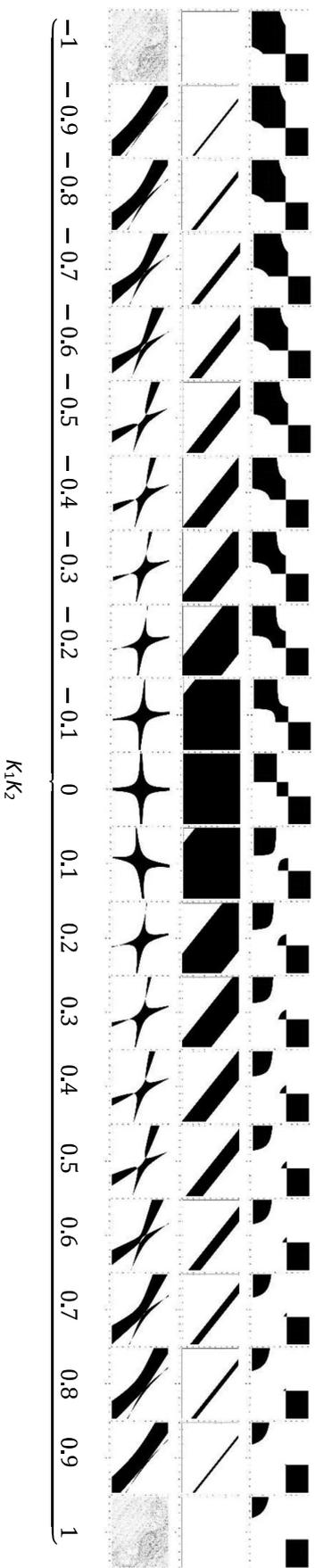


Figure A: Inequality plots of individual Jury Criteria with increasing magnitudes of K_1K_2

In (Figure A) we have 63 inequality plots. The axis are too small to be distinguished but a detailed explanation follows: beginning from left to right, in the first row we have the inequality plots depicting the intersection of Eq.(C1) and Eq.(C2). In the second row we have the inequality plots for Eq.(C4). In the third row we have the inequality plots for Eq.(C5), and in the fourth row we have the corresponding values of K_1K_2 for each column, which administers Eq.(C3). Each plot has a as the horizontal-axis and b as the vertical-axis, both ranging from -5 to +5 with the origin at the centre of each plot. The black regions portray the areas in which the respective criteria are met, therefore, graphing the intersections of the first three rows of each column would give the region of stability for the coupled system Eq.(5.1), according to the respective values of K_1K_2 .

When $|K_1K_2| = 1$, we can see that stability is not achieved, as the leftmost and rightmost plots of the second row have no black regions. This adheres to Eq.(C3).

5.2.2 Analysis of coupled system with ‘zero control’

Beginning with $K_1 = 0, K_2 = 0$ or $K_1 = K_2 = 0$ Eqs.(C1-C5) of Jury’s Criteria read:

1. $P(1) = ab - a - b + 1 > 0$
2. $P(-1) = ab + a + b - +1 > 0$
3. $0 < 1$
4. $1 > 0$
5. $1 > |ab|$

We can see that 3 and 4 are equivalent Boolean expressions which when evaluated are true. In (Figure 4)⁴ the **white region** represents the intersection of 1, 2 and 5, which shows us that when at least one control gain is zero, stability is achieved when $|a| < 1$ and $|b| < 1$. This result is reasonable, as without the presence of control, an unstable signal will not be able to be stabilised.

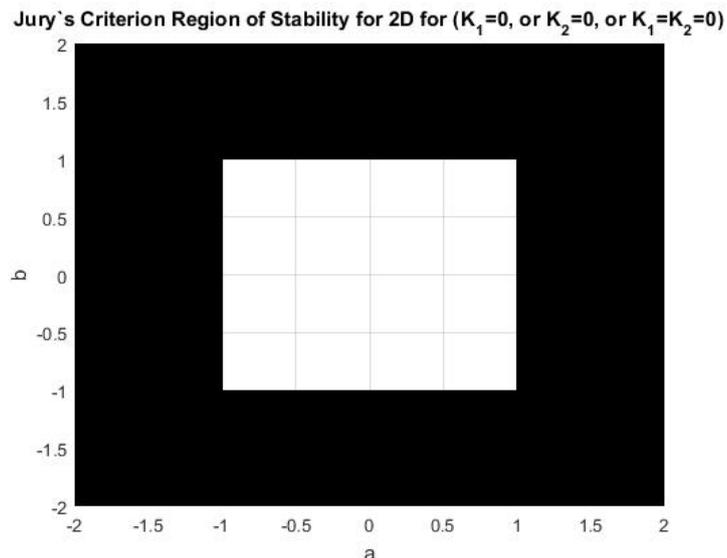


Figure 4: Plot of region of All Criteria

⁴See Appendix for Matlab code

5.2.3 Graphs of regions of stability with $-1 \leq K_1K_2 \leq 0.95$ (in steps of 0.05)

We shall now present 40 plots, over 5 figures in order to give an overview of the regions of stability for varying control parameters. In each plot from here on, the region where the criteria are met are the **white regions**.

In (Figure 5)⁵ We have used Eqs.(C1-C5) to plot the region of stability for the system Eq.(5.1) for the respective values of the control gains.

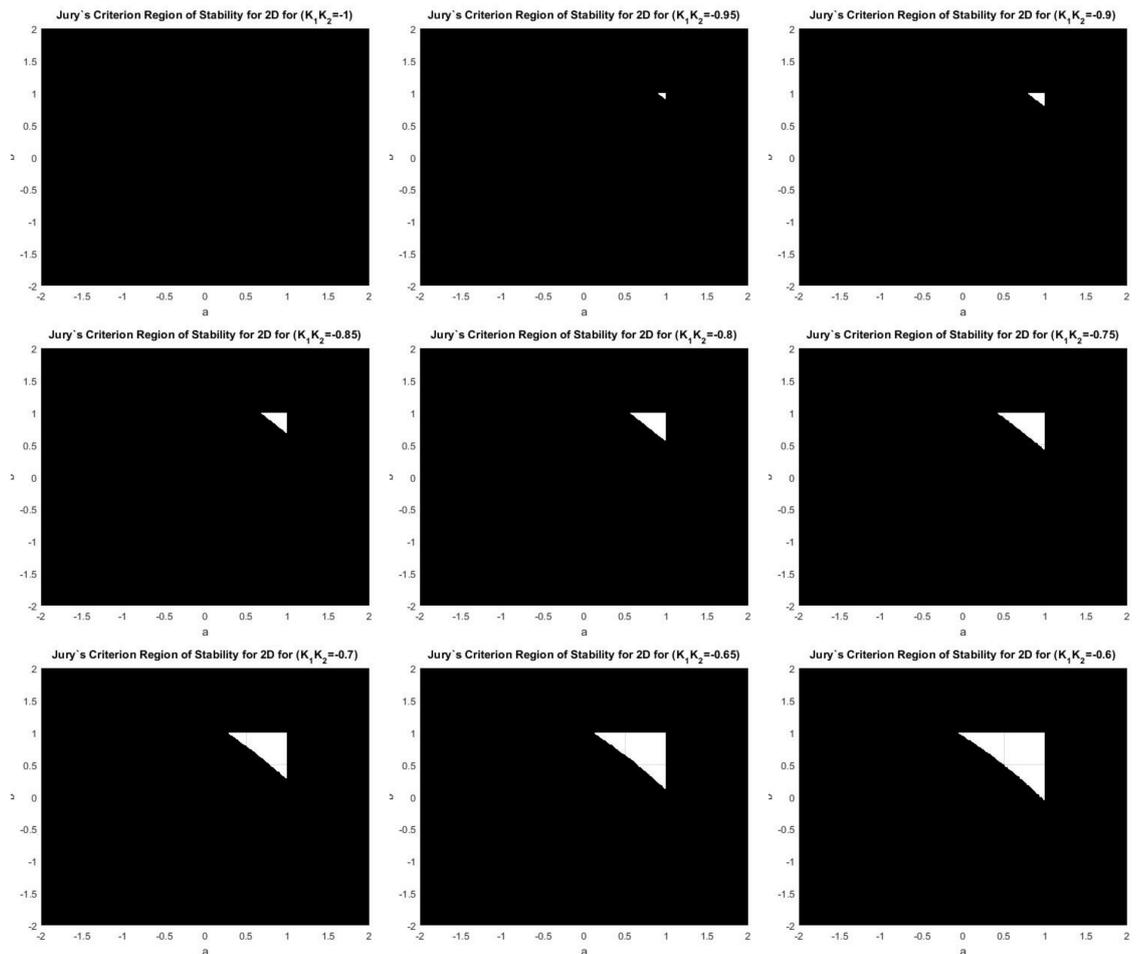


Figure 5: **Plot of region of stability for 2-D case with $-1 \leq K_1K_2 \leq -0.6$:** The figure represents the stability region of Eq.(5.1) for $-1 \leq K_1K_2 \leq -0.6$ in steps of 0.05, with the top left figure representing the region of stability when $K_1K_2 = -1$, and the bottom right figure representing the region of stability when $K_1K_2 = -0.6$.

⁵See Appendix for Matlab code

In (Figure 6)⁶ We have used Eqs.(C1-C5) to plot the region of stability for the system Eq.(5.1) for the respective values of the control gains.

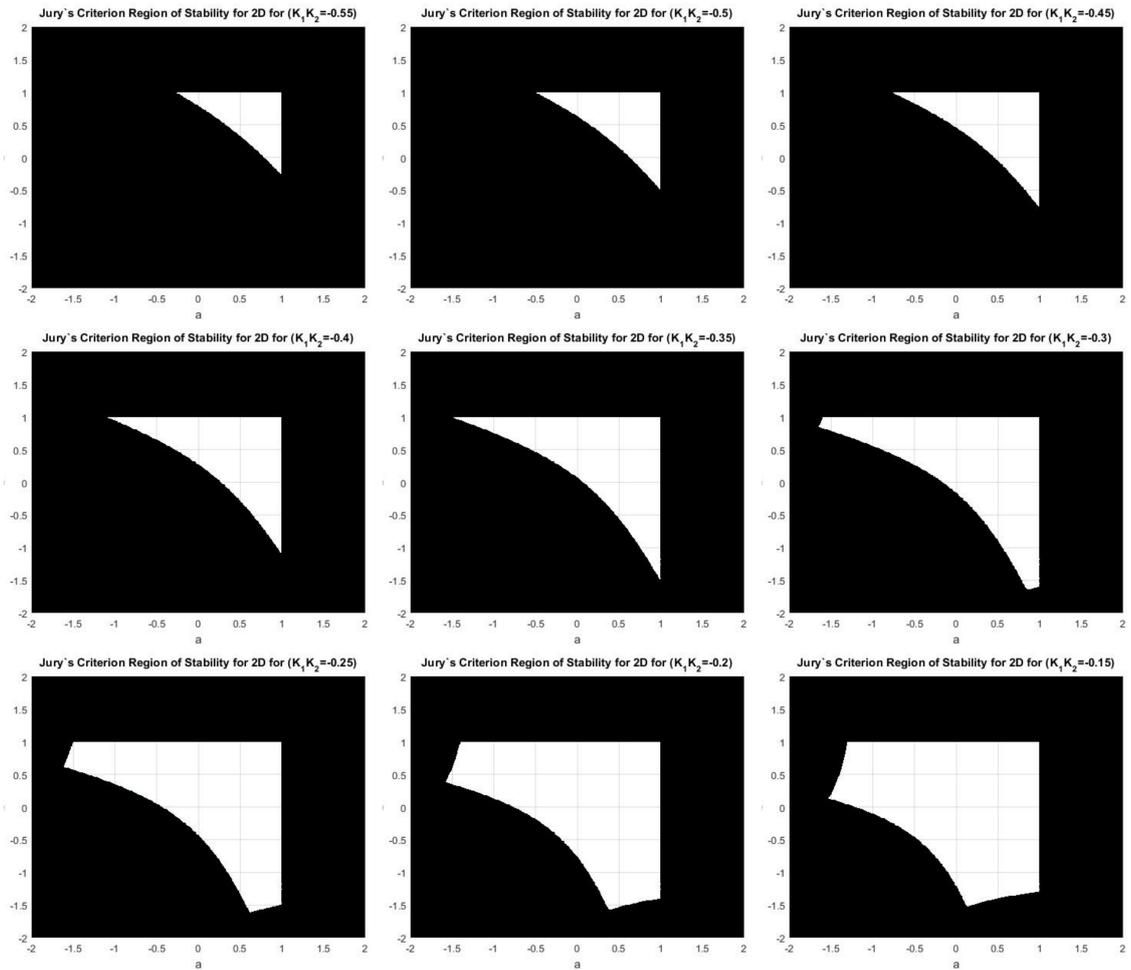


Figure 6: **Plot of region of stability for 2-D case with $-0.55 \leq K_1K_2 \leq -0.15$:** The figure represents the stability region of Eq.(5.1) for $-0.55 \leq K_1K_2 \leq -0.15$ in steps of 0.05, with the top left figure representing the region of stability when $K_1K_2 = -0.55$, and the bottom right figure representing the region of stability when $K_1K_2 = -0.15$.

⁶See Appendix for Matlab code

In (Figure 7)⁷ We have used Eqs.(C1-C5) to plot the region of stability for the system Eq.(5.1) for the respective values of the control gains.

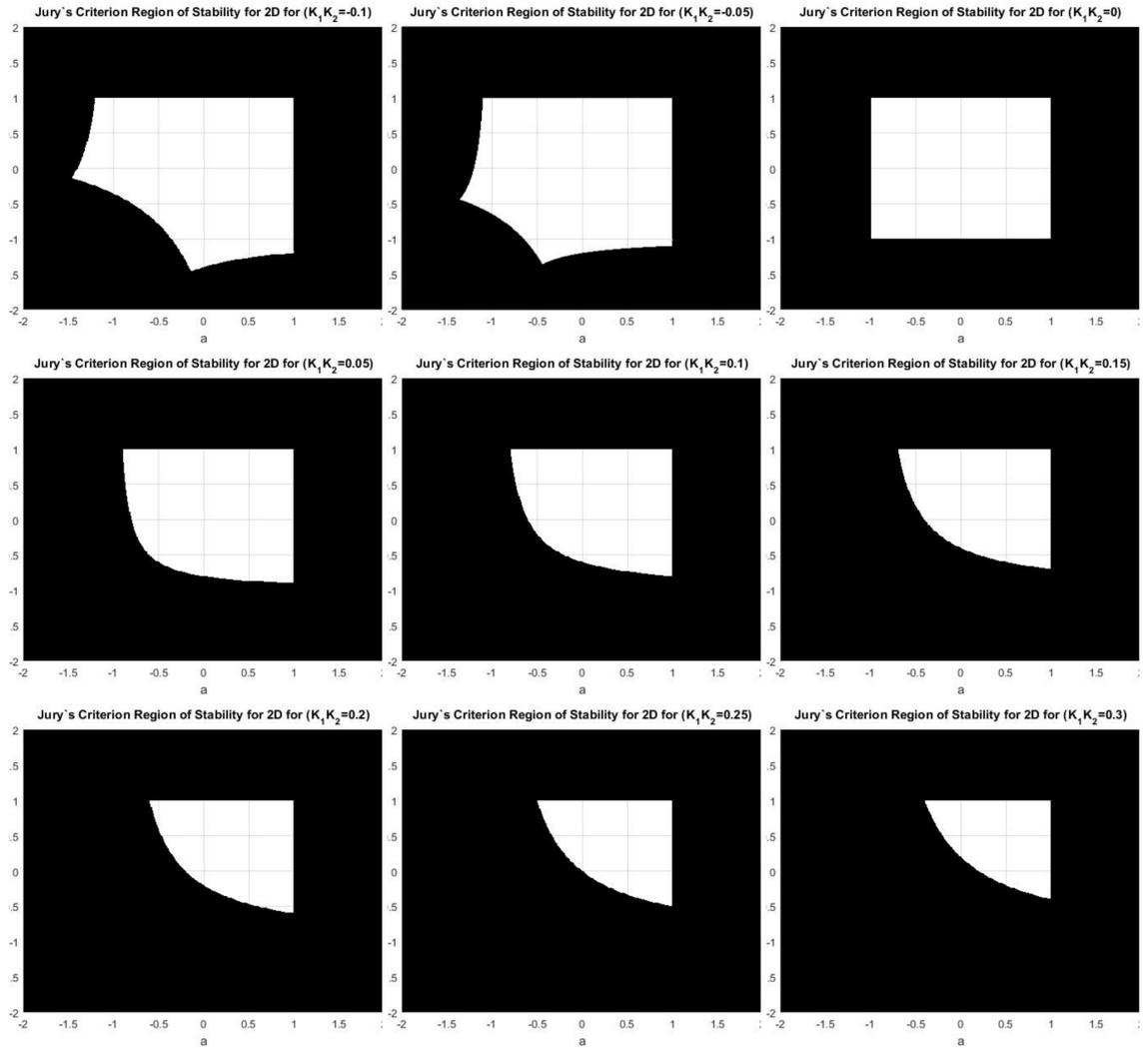


Figure 7: **Plot of region of stability for 2-D case with $-0.1 \leq K_1K_2 \leq 0.3$:** The figure represents the stability region of Eq.(5.1) for $-0.1 \leq K_1K_2 \leq 0.3$ in steps of 0.05, with the top left figure representing the region of stability when $K_1K_2 = -0.1$, and the bottom right figure representing the region of stability when $K_1K_2 = -0.3$.

⁷See Appendix for Matlab code

In (Figure 8)⁸ We have used Eqs.(C1-C5) to plot the region of stability for the system Eq.(5.1) for the respective values of the control gains.

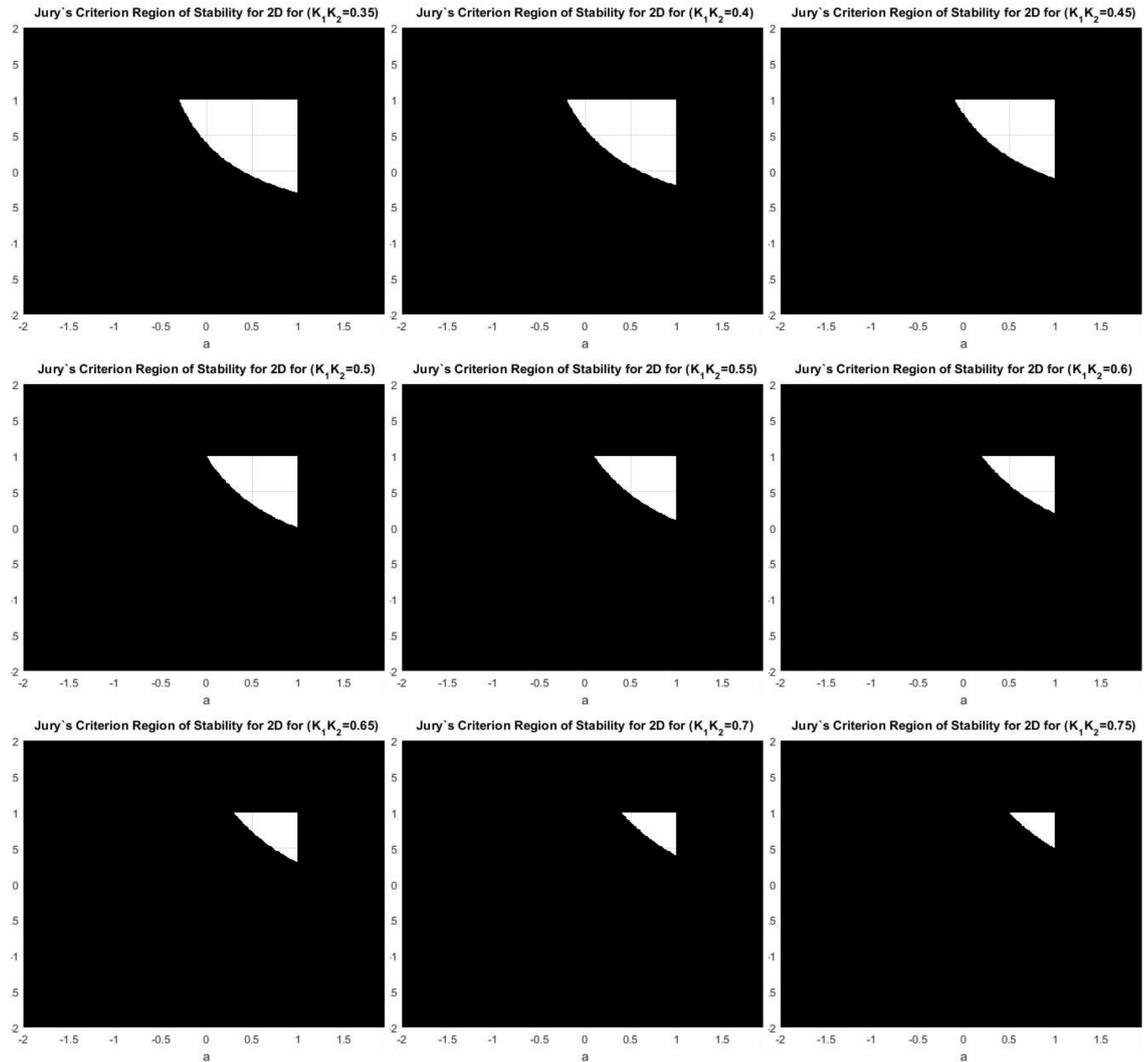


Figure 8: **Plot of region of stability for 2-D case with $0.35 \leq K_1K_2 \leq 0.75$:** The figure represents the stability region of Eq.(5.1) for $0.35 \leq K_1K_2 \leq 0.75$ in steps of 0.05, with the top left figure representing the region of stability when $K_1K_2 = 0.35$, and the bottom right figure representing the region of stability when $K_1K_2 = 0.75$.

⁸See Appendix for Matlab code

In (Figure 9)⁹ We have used Eqs.(C1-C5) to plot the region of stability for the system Eq.(5.1) for the respective values of the control gains.

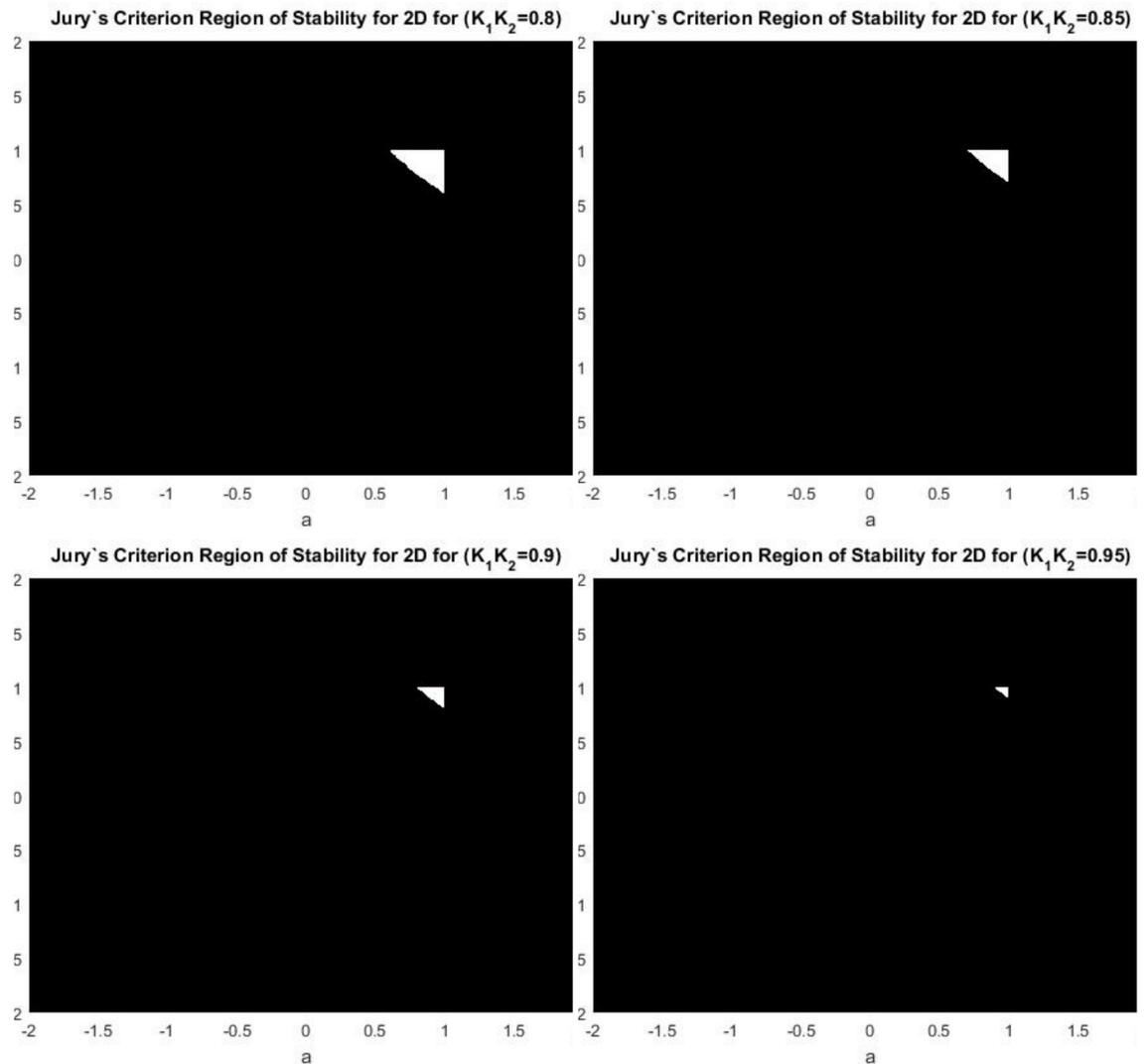


Figure 9: **Plot of region of stability for 2-D case with $0.8 \leq K_1K_2 \leq 0.95$:** The figure represents the stability region of Eq.(5.1) for $0.8 \leq K_1K_2 \leq 0.95$ in steps of 0.05, with:

The top left figure representing the region of stability when $K_1K_2 = 0.8$,

The top right figure representing the region of stability when $K_1K_2 = 0.85$,

The bottom left figure representing the region of stability when $K_1K_2 = 0.9$,

The bottom right figure representing the region of stability when $K_1K_2 = 0.95$.

When $K_1K_2 = 1$ we know from (Figure A), that stability cannot be achieved.

5.2.4 Analysis of regions of stability with $-1 \leq K_1K_2 \leq 1$

(Figures 5-9) indicate that we have the largest region of stability when $K_1K_2 = 0$, and this decreases as the magnitude of K_1K_2 increases. We can see from (Figure

⁹See Appendix for Matlab code

5) and up until the third plot in (Figure 6), in other words when, $-1 < K_1K_2 \leq -0.45$, the system can only be stabilised for $|a| < 1$ and $|b| < 1$, in which case stability would not be required, as the system is stable without control. (Figures 5-9) also indicate that stability cannot be achieved for $a \geq 1$ and $b \geq 1$. Moreover, as the magnitude of K_1K_2 increases, the regions of stability go from a square ($K_1K_2 = 0$), and then undergo some sort of ‘distortion’ (for negative values only) before the region shrinks in towards the point $(a, b) = (1, 1)$ whilst resembling an isosceles right angled triangle (with a concave/convex hypotenuse). The two equal sides of this triangle are represented by $a < 1$ and $b < 1$, and the right angle is the point $(a, b) = (1, 1)$. The major difference between the negative and positive values of K_1K_2 , is that for positive values, the region shrinks in a what seems to be uniform manner, and doesn’t undergo any distortion.

(Figures 5-9) suggest that Eq.(5.1) cannot be stabilised for $|a| \geq 1$ and $|b| \geq 1$, however, from (Figure 6-plot 4) to (Figure 7-plot 2), in other words, from $-0.4 \leq K_1K_2 \leq 0.05$, the ‘distortion’ that occurs, enables us to stabilise some values for $|a| \geq 1$ and $|b| < 1$, or vice versa. For an example, we take a look at (Figure 10) which is an enlarged version of (Figure 6-plot 8):

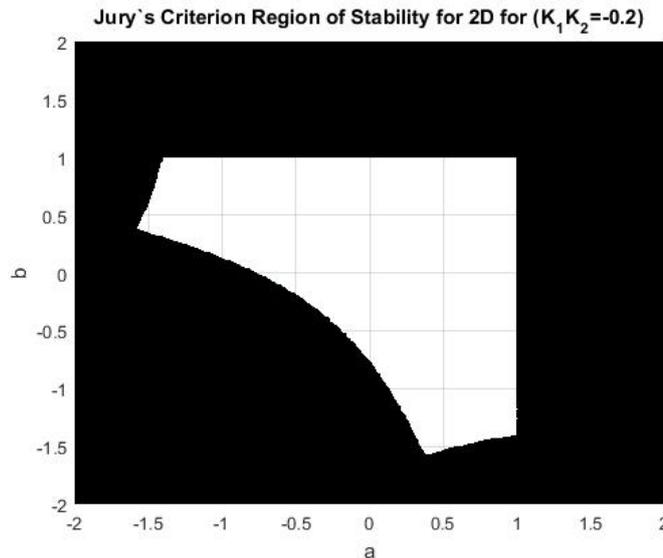


Figure 10: Plot of region of stability for 2-D case with $K_1K_2 = -0.2$

We can see from (Figure 10), that there are values of $|a| \geq 1$ and $|b| < 1$ (or vice versa), which the graph indicates, that stability would be successful.

Example 5. For Eq(5.1), Letting $a = -1.55$, $b = 0.4$, $K_1 = -2$, $K_2 = 0.1$ we

have the coupled map:

$$\begin{aligned}x_{n+1} &= -1.55x_n - 2(y_n - y_{n-1}) \\y_{n+1} &= 0.4y_n + 0.1(x_n - x_{n-1})\end{aligned}\tag{5.2}$$

Jurys Criterion for the system reads:

$$(C1) = -0.62 + 1.55 - 0.4 + 1 = 1.53 > 0$$

$$(C2) = -0.62 - 1.55 + 0.4 + 0.8 + 1 = 0.03 > 0$$

$$(C3) = |-0.2| < 1$$

$$(C4) = |(-0.2)^2 - 1| > |0.1(-6.3 - 1 - 2)| \implies 0.96 > 0.63$$

$$(C5) = |(-0.2)^4 - (11.9225)(-0.2)^2 + 1| > |(-0.2)^4 + (-4.68)(-0.2)^3 - (0.38)(-0.2)^2 - (5.2425)(-0.2) - 0.62| \implies 0.5427 > 0.45234$$

As all the criteria have been met, the four eigenvalues of the system are all smaller than one in unity, and stabilisation of the system is guaranteed.

In (Figure 11)¹⁰ we have plotted the first 500 iterates of the example above, with x_n in black and y_n in blue. It can be seen, that even though our system was initially unstable, after introducing the TDFC mechanism, the trajectories oscillate and converge to the fixed point, and hence the coupled system has been stabilised.

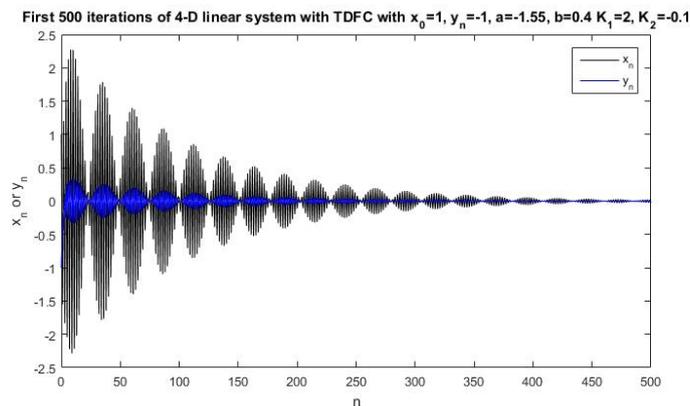


Figure 11: Plot of the first 500 iterates of 4-D system with TDFC with $x_0 = 1, y_0 = -1, a = -1.55, b = 0.4, K_1 = -2$ and $K_2 = 0.1$

¹⁰See Appendix for Matlab code

5.2.5 Results

For the system:

$$\begin{aligned}x_{n+1} &= ax_n + K_1(y_n - y_{n-1}) \\ y_{n+1} &= by_n + K_2(x_n - x_{n-1})\end{aligned}\tag{5.1}$$

Our graphical analysis indicates that for all values of $|K_1K_2| < 1$ the system can be stabilised with TDFC, although for the majority of cases $|a| < 1$ and $|b| < 1$, so TDFC would not be useful. In addition to this:

- When $K_1 = 0$ or $K_2 = 0$ the fixed point $(x_*, y_*) = (0, 0)$ is stable if and only if $|a| < 1$ and $|b| < 1$.
- When $-0.4 \leq K_1K_2 \leq 0.05$ there are values for some $|a| \geq 1$ and $|b| < 1$, or vice versa such that the system can be stabilised.
- The system cannot be stabilised if $|a| \geq 1$ **and** $|b| \geq 1$.
- The system cannot be stabilised if $a \geq 1$ **or** $b \geq 1$.

6 Conclusion

By using analytical methods, we have explored the control of unstable fixed point solutions in discrete time dynamical systems, with TDFC. By means of linear stability, we found that for the 1-D case Eq(3.1), in order for stability to be achieved, we must have:

$$|\lambda_{1/2}| = \left| \frac{a + K \pm \sqrt{(a + K)^2 - 4K}}{2} \right| < 1$$

which led to the conclusion that we can find a suitable control gain K which could stabilise an originally unstable signal of $-3 < a \leq -1$.

This concept was then applied to a 2-D case Eq(5.1), where we used Jury's Criterion to find the necessary and sufficient conditions to restrict the eigenvalues of the quartic equation to less than one in magnitude are. These criteria are:

- $ab - a - b + 1 > 0$
- $ab + a + b - 4K_1K_2 + 1 > 0$
- $|K_1K_2| < 1$
- $|K_1^2K_2^2 - 1| > |(a + b - 2)K_1K_2|$
- $|K_1^4K_2^4 - (a^2 + b^2 + 2ab - 4a - 4b + 6)K_1^2K_2^2 + 1| > |K_1^4K_2^4 + (2a + 2b - ab - 3)K_1^3K_2^3 - (ab + 1)K_1^2K_2^2 - (a^2 + b^2 + ab - 2a - 2b + 1)K_1K_2 + ab|$

By treating K_1K_2 as a single variable, we were able to produce graphical plots of these criteria, with varying values of K_1K_2 , and this led to the conclusion that the system can be stabilised for some values of $|a| \geq 1$ and $|b| < 1$, (intuitively one would assume $a > -3$, as this was a limit in the 1-D case) but not for any $|a| \geq 1$ and $|b| \geq 1$, or $a \geq 1$ and any b (or vice versa).

This topic could be explored further in many ways. One could explore the effect of changing the values of K_1 and K_2 , (where the value of K_1K_2 would

remain constant) and seeing whether iterations converge at different rates to the fixed point. One could also go about formalising and extending the analysis of the graphical conjectures e.g. from the results section we have that “When $-0.4 \leq K_1K_2 \leq 0.05$ there are values for some $|a| \geq 1$ and $|b| < 1$, or vice versa such that the system can be stabilised”. When $K_1K_2 = -0.01$, we have (Figure 12)¹¹, which suggests that the inequality is $-0.4 \leq K_1K_2 < 0$. One

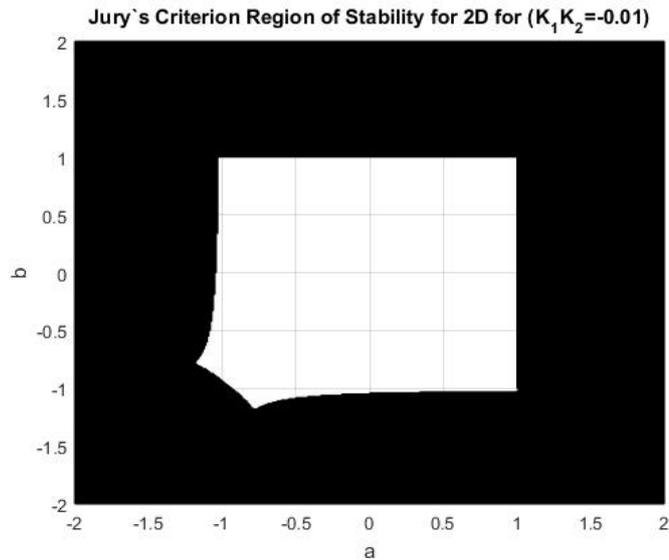


Figure 12: Region of 2D stability with $K_1K_2 = -0.01$

may also explore systems with a differing version of TDFC for example by using $K(x_{n-1} - x_{n-2})$, in order to see whether other unstable solutions can be stabilised.

¹¹See Appendix for Matlab code

A

Appendix - Matlab codes for Figures

This section reveals the Matlab coding used to create many of the figures used in the thesis.

Matlab Code for (Figure 1)

This Matlab code was used to get the region of stability:

```
area = -5:0.01:5,-5:0.01:5;
[a K] = meshgrid(area);
lamda_1 = abs(a + K + ((a + K).^2 - 4.*K).^(1/2)) < 2;
lamda_2 = abs(a + K - ((a + K).^2 - 4.*K).^(1/2)) < 2;
lamda_1 = double(lamda_1);
lamda_2 = double(lamda_2);
lamda_1(lamda_1 == 0) = NaN;
lamda_2(lamda_2 == 0) = NaN;
eigenvalues = lamda_1.*lamda_2;
surf(a,K,eigenvalues)
view(0,90)
xlabel('a')
ylabel('b')
title('Region of Stability')
```

This Matlab code was then used to recreate the area of stability as seen in (Figure 1):

```
axis([-4 3 -2 3])
line([-3 -1],[1 0],[1 1], 'LineStyle', '-')
hold on
```

```
line([-1 1],[0 -1],[1 1],'LineStyle','-')
hold on
line([1 1],[-1 1],[1 1],'LineStyle','-')
hold on
line([1 -1],[1 1],[1 1],'LineStyle','-')
hold on
line([-1 -3],[1 1],[1 1],'LineStyle','-')
hold on
line([-1 -1],[0 1],[1 1],'LineStyle','-')
grid on
hold on
area = -5:0.01:5;
[a K] = meshgrid(area);
inequalities = ((a + K).^2 - 4 * K > 0);
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, K, points)
view(0,90)
xlabel('a')
ylabel('K')
title('Region of Stability')
```

Matlab Code for (Figure 2)

This Matlab code was used to produce (Figure 2):

```
x = zeros(51,1);
a = -2;
K = 0.75;
x(1) = 1;
```

```
x(2) = a.*x(1);
for i = 3:1:51
    x(i) = a.*x(i-1) + K.*(x(i-1) - x(i-2));
end
z = 0:1:50;
plot(z,x)
xlabel('n')
ylabel('x_n')
title('First 50 iterations of 2-D linear system with TDFC with  $x_0 = 1, y_n = -1, a = -2, K = 0.75'$ )
```

Matlab Code for (Figure A)

This Matlab code was used to get the regions of inequalities for row 1:

```
area = -5 : 0.01 : 5;
[a, b] = meshgrid(area);
for K = -1 : 0.1 : 1;
inequalities = (a.*b - a - b + 1 > 0)&(a.*b + a + b - 4.*K + 1 > 0);
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, b, points);
view(0,90)
xlabel('a')
ylabel('b')
figure
end
```

This Matlab code was used to get the regions of inequalities for row 2:

```
area = -5 : 0.01 : 5;
[a, b] = meshgrid(area);
for K = -1 : 0.1 : 1;
inequalities = (abs(K.^2 - 1) > abs(K .* (a + b - 2)));
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, b, points);
view(0,90)
xlabel('a')
ylabel('b')
figure
end
```

This Matlab code was used to get the regions of inequalities for row 3:

```
area = -5 : 0.01 : 5;
[a, b] = meshgrid(area);
for K = -1 : 0.1 : 1;
inequalities = (abs(K.^4 - (a.^2 + b.^2 + 2 .* a .* b - 4 .* a - 4 .* b + 6) .* K.^2 + 1) >
abs(K.^4 + (2 .* a + 2 .* b - a .* b - 3) .* K.^3 - (a .* b + 1) .* K.^2 - (a.^2 + b.^2 +
a .* b - 2 .* a - 2 .* b + 1) .* K + a .* b));
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, b, points);
view(0,90)
xlabel('a')
ylabel('b')
figure
end
```

Matlab Code for (Figures: 4-9, & 12)

This Matlab code was used to get the regions of stability for (Figures: 4-9):

```

area = -2 : 0.01 : 2;
[a, b] = meshgrid(area);
for K = -1 : 0.05 : 1;
inequalities = (a.*b-a-b+1 > 0)&(a.*b+a+b-4.*K+1 > 0)&(abs(K.^2-1) >
abs(K.*(a+b-2)))&(abs(K.^4-(a.^2+b.^2+2.*a.*b-4.*a-4.*b+6).*K.^2+1) >
abs(K.^4+(2.*a+2.*b-a.*b-3).*K^3-(a.*b+1).*K^2-(a.^2+b.^2+
a.*b-2.*a-2.*b+1).*K+a.*b));
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, b, points);
view(0,90)
xlabel('a')
ylabel('b')
title(['Jury's Criterion Region of Stability for 2D for (K1K2= ' num2str(K) ')'])
figure
end

```

The Matlab code used to get the region of stability for (Figure 12) is:

```

area = -2 : 0.01 : 2;
[a, b] = meshgrid(area);
K = -0.01;
inequalities = (a.*b-a-b+1 > 0)&(a.*b+a+b-4.*K+1 > 0)&(abs(K.^2-1) >
abs(K.*(a+b-2)))&(abs(K.^4-(a.^2+b.^2+2.*a.*b-4.*a-4.*b+6).*K.^2+1) >
abs(K.^4+(2.*a+2.*b-a.*b-3).*K^3-(a.*b+1).*K^2-(a.^2+b.^2+
a.*b-2.*a-2.*b+1).*K+a.*b));

```

```
points = zeros(length(area));
points(inequalities) = NaN;
surf(a, b, points);
view(0,90)
xlabel('a')
ylabel('b')
title(['Jury's Criterion Region of Stability for 2D for ( $K_1K_2=$ ' num2str(K) ' ')'])
```

Matlab Code for (Figure 11)

The Matlab code used in order to achieve this plot is:

```
x = zeros(500,1);
y = zeros(500,1);
a = -1.55;
b = 0.4;
K1 = -2;
K2 = 0.1;
x(1) = 1;
y(1) = -1;
x(2) = a.*x(1);
y(2) = b.*y(1);
for i = 3:1:501
x(i) = a.*x(i-1) + K1.*(y(i-1) - y(i-2));
y(i) = b.*y(i-1) + K2.*(x(i-1) - x(i-2));
end
z = 0:1:500;
plot(z,x,'k',z,y,'b')
xlabel('n')
ylabel('xn or yn')
```

```
title('First 500 iterations of 4-D linear system with TDFC with  $x_0 = 1, y_n =$   
 $-1, a = -1.55, b = 0.4K_1 = 2, K_2 = -0.1$ ')  
legend('x_n', 'y_n')
```

References

- [1] Just, Wolfram. "MTH744P-Dynamical Systems, 2015–2016". Queen Mary University of London - Lecture Notes (2015) p1-3.
- [2] Devaney, Robert L. "An introduction to chaotic dynamical systems". Vol. 13046. Reading: Addison-Wesley, (1989).
- [3] Alligood, Kathleen T. Tim D. Sauer, and James A. Yorke., "Chaos." Chaos. Springer Berlin Heidelberg, (1997).
- [4] Strogatz, Steven H. "Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering". Westview press, (2014).
- [5] Hsu, Sze-Bi. "Ordinary differential equations with applications". World scientific, (2006).
- [6] Vivaldi, Franco. "Mathematical writing for undergraduate students". School of Mathematical Sciences. Queen Mary, University of London. Citado na pág (2014) p81.
- [7] Phillips, Charles L. and Royce D. Habor., "Feedback control systems". Simon & Schuster, (1995) p529-530.
- [8] Pyragas, Kestutis. "Continuous control of chaos by self-controlling feedback". Physics letters A 170.6 (1992): p421-428.
- [9] Bandtlow, Oscar. "MTH5112-Linear Algebra I, 2009–2010". Queen Mary University of London - Lecture Notes (2010) p23-26.
- [10] Socolar, Joshua ES. "Time-Delay Control for Discrete Maps". Handbook of Chaos Control (2008) p29-30.
- [11] Just, Wolfram. "On global properties of time-delayed feedback control". Handbook of chaos control (2006) p95.

-
- [12] Ushio, Toshimitsu. "Limitation of delayed feedback control in nonlinear discrete-time systems". *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 43.9 (1996) p815-816.
- [13] Jenkinson, Oliver. "MTH5112-Chaos and Fractals, 2014–2015". Queen Mary University of London - Lecture Notes (2014).
- [14] Keel, Lee H., Bhattacharyya, Shankar P. "A new proof of the Jury test". *American Control Conference, 1998. Proceedings of the 1998. Vol. 6. IEEE, 1998*
- [15] Weir, Maurice D., Hass, Joel. "Thomas' Calculus". Twelfth Edition (2010) p795
- [16] Macau, Elbert E.N., Grebogi, Celso. "Controlling Chaos". *Handbook of Chaos Control* (2008) p3-6.
- [17] Pyragas, K., and A. Tamaševičius. "Experimental control of chaos by delayed self-controlling feedback." *Physics Letters A* 180.1 (1993) p99-102.
- [18] Ott, Edward., Grebogi, Celso. Yorke, James A. "Controlling chaos." *Physical review letters* 64.11 (1990) p1196.
- [19] Hövel, P., Schöll, E. "Control of unstable steady states by time-delayed feedback methods." *Physical Review E* 72.4 (2005) 046203.
- [20] Pratchett, Terry. "Time-Delayed Feedback Control." Springer (2010).