Spring 2014

Adaptive State Feedback Control of Lorenz Systems to Its Non-Trivial Equilibrium

Anh V. Tran

ADAPTIVE STATE FEEDBACK CONTROL OF LORENZ SYSTEMS TO ITS NON-TRIVIAL EQUILIBRIUM

by

ANH VUONG TRAN

(Under the Direction of Yan Wu)

ABSTRACT

The complex Lorenz system is a simplified nonlinear dynamical system, which is derived from the Navier-Stokes equations that govern a closed thermal convection loop. The Lorenz system is chaotic for large Rayleigh number. In this chaotic regime, we implement a linear state feedback controller to stabilize the state trajectory at its original nontrivial equilibrium. The state variable for feedback is easily measurable. The system is proved to be globally asymptotically stable with a optimal feedback gain. The stability bound is improved over the previous result. We also established globally stability of the adaptively control system, where the system parameters are unknown. We present numerical simulations to demonstrate the stability, transient and steady state responses, and the performance of the state feedback controller.

Key Words: adaptive control; Lyapunov stability; feedback control;
multi-equilibrium; natural convection loops; Lorenz system

2009 *Mathematics Subject Classification*: Control theory

ADAPTIVE STATE FEEDBACK CONTROL OF LORENZ SYSTEMS TO ITS NON-TRIVIAL EQUILIBRIUM

by

ANH VUONG TRAN

B.S. in Mechanical Engineering

A Thesis Submitted to the Graduate Faculty of Georgia Southern University in Partial Fulfillment

of the Requirement for the Degree

MASTER OF SCIENCE

STATESBORO, GEORGIA

2014

ADAPTIVE STATE FEEDBACK CONTROL OF LORENZ SYSTEMS TO ITS NON-TRIVIAL EQUILIBRIUM

by

ANH VUONG TRAN

Major Professor: Yan Wu

Committee: Frank Goforth

Alex Stokolos

Electronic Version Approved:

May 9, 2014

DEDICATION

I dedicate my work to my to my father, Anh Tran, who has been teaching me many things for many years since I was a kid. He has been strict, yet extremely patient toward my mistakes. He was the one that taught me how to do math when I was at 10, and moreover, taught me how to think logically. He taught me that I can be smart, but I must be nice to other people. Without his endless efforts and advices, I would not make it this far.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Yan Wu for all of his support and guidance throughout the research project. He has spent numerous hours to help me improve and polish my work, to teach me about things that I can never learn in any books but only through personal experience, such as why should I become PhD and what does it takes to do so. Such things are almost never mentioned in any books, and I am very lucky to have Dr. Wu to be my advisor. He has taught me much more than I could have asked for, and I will certainly carry those helpful advices toward my PhD journey in the future.

I would like to thank Dr. Frank Goforth and Dr. Alex Stokolos for their comments and feedback to this thesis and for being my committee members.

I express my deep sense of gratitude toward Dr. Scott Kersey, Dr. Hua Wang, Dr. Sze-Man Ngai, Dr. Frederic Mynard, Dr. Goran Lesaja, Dr. Broderick Oluyede, Dr. Shujun Zheng for their helps in providing necessary materials to expand my knowledge in different areas.

My immense thanks to all the teaching and non-teaching staff members, especially Ms. Nikki Collins of the Department of Mathematical Sciences, Georgia Southern University for their kind help during the course of study.

I extend my special thanks to my graduate friends, particularly Jing Sun, Shuai Yuan, Adrian Joseph, Julia Inozemtseva, Tabitha Williford, William Trott, Valeriia Sherina, etc. for being a part of my graduate school experience. I greatly enjoy my 2 years at Georgia Southern University because of these people.

I sincerely thank Dr. Jeff Knisley at East Tennessee States University for providing his presentation at MAA Southeastern Section 2014 and illustrative examples regarding chaotic behavior of Lorenz system.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF SYMBOLS

- r Rayleigh number
- σ Prandtl number
- b geometric configuration parameters
- μ viscous coupling parameters
- N thermal coupling parameters
- $C^{\pm}$ non-trivial equilibrium points
- C^0 trivial equilibrium point
- k_G global stability threshold of single Lorenz cell
- c_1: $c_1 = \sqrt{b(r-1)}$
- c_2: $c_2 = r - 1$
- T_E external temperature
- q velocity inside the loop
- ϕ angle coordinate. The positive direction is counter-clockwise, and $\phi = 0$ is located at 6 o'clock.
- Γ generalized friction coefficient, corresponding to viscous resistance proportional to the velocity
- K thermal conductivity of the loop

CHAPTER 1

INTRODUCTION

The Lorenz system [1] has been one of the most celebrated chaotic dynamical systems and known as a simplied model of several physical systems. The Oberbeck-Boussineq equations that govern convection fluid flow between two free surfaces maintaing at constant temperature differences can be reduced to a two-dimensional system. This natural convection flow develops a regular pattern of convection cells known as Rayleigh-Benard cells. In 1962, Barry Saltzman [2] derived an infinite set of ordianry differential equations by expanding the stream function and the departure of temperature in double Fourier series form, with functions of t alone for coefficients. In 1963, Lorenz [1] truncated, selected three modes and derived a set of three non-linear first order differential equations, which have bounded solutions and dissipative. There are many similarities between three dimensional and higher dimensional systems, especially the properties of conserving energy in the dissipationless limit and lead to systems that have bounded solutions [3]. Numerous attempts have been made to control the Lorenz system: adaptive control [4], proportional-plus-integral control [5], delayed feedback control [6], impulsive control [7] to mention only a few.

The natural convection in a toroidal loop has been studied extensively due to its wide applications in solar system, thermosyphon, nuclear reactors, geophysical systems, etc. The fluid flows around the loop, which is placed in vertical plane, is driven by the buoyancy and gravity due to vertical temperature difference, formed by heating from below and cooling from above. Experimental works pioneered by Keller [8], Welander [9] and Creveling et al. [10] confirmed the observation of instability in single-phase flow. In 1972, Malkus [11] derived Lorenz model in a closed natural convection loop.

The equations that governs the fluid motion and temperature variation are de-

scribed as:

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = rx - y - xz \\ \dot{z} = xy - bz \end{cases} \tag{1.1}$$

where the dimensionless parameters r, σ and b are Rayleigh number, Prandtl number, and geometric parameter (related to the horizontal wavenumber of convection motion), respectively. For a closed convection loop, the geometric parameter $b = 1$, $x(t)$ is the dimensionless circulating velocity of the fluid flow, $y(t)$ is the horizontal dimensionless temperature difference and $z(t)$ is the dimensionless departure from conductive equilibrium. Fixed points solutions of Lorenz system include the origin C^0, and two other nontrivial equilibrium $\mathcal{C}^{\pm} = (\pm\sqrt{b(r-1)}, \pm\sqrt{b(r-1)}, r-1)$. Over the last two decades, the Lorenz system has been stablized around the origin, but not the other equilibrium points. Controlling the systems around the other two equilibrium points shed a light on adaptive control as well as tracking control problem.

In this thesis, we investigate the flow stability in a toroidal thermosyphon with heating from below and cooling from above, assuming that the temperature at both places are constant. In Chapter 1, the derivation of Lorenz system for thermosyphon is briefly excerpted from Tritton [12]. In Chapter 2, we introduce the linear state feedback controller into the chaotic Lorenz system, followed by proof of global stability for $k > k_G$. Also, a local stability analysis is studied based on the eigenvalues of Jacobian matrices of main and auxiliary equilibrium points. Also, an adaptive controller is proposed upon the development of linear feedback controller to suppress chaotic behavior, with a differential equation on the controller gain. In Chapter 3, we propose the thermally and viscously coupled dual Lorenz system and investigate their equilibrium, as well as their stability with two added coupling parameters. In Chapter 4, we employ MATLAB to demonstrate the suppression of chaotic behav-

ior based on stability analysis of single Lorenz system in previous chapter, and the synchronization of double Lorenz systems. We provide the MATLAB codes at the appendix for the purpose of reproducing numerical experiments.

CHAPTER 2

THE MATHEMATICAL MODEL OF NATURAL CONVECTION LOOP

Following the procedure of Malkus [11] and Tritton [12], we shall very briefly introduce the derivation of Lorenz system to study the interraction between x, y and z variables.

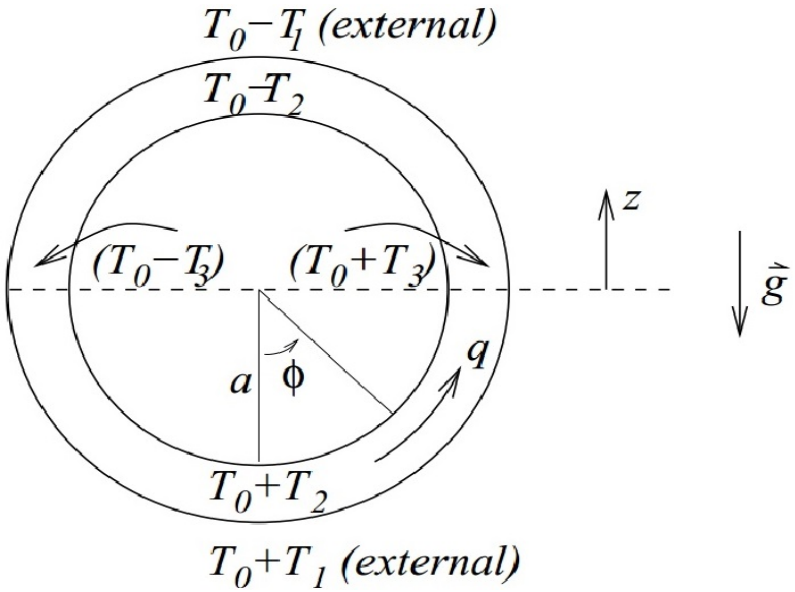

Figure 2.1: Schematic sketch of a simplified toroidal thermosyphon. Photo courtesy of MIT OpenCourseWare.

where ϕ is angular position round the loop, and a is radius of the loop. Assume that the tube's inner radius is much smaller than a and external temperature T_E varies linearly with height, $T_E = T_0 - T_1 \frac{z}{a} = T_0 + T_1 \cos\phi$. Let q and T be the averaged cross-sectionally velocity and temperature inside the loop. $q = q(\phi, t), T = T(\phi, t)$. As in the Rayleigh-Benard problem, we employ Boussineq approximation and therefore assume $\frac{\partial \rho}{\partial t} = 0$. Continuity equation yields $\frac{\partial q}{\partial \phi} = 0$, thus the velocity inside the loop is a function of time $q = q(t)$. The temperature $T(\phi)$ could in reality vary with much complexity, here we assume it depends only on two parameters, T_2 and T_3: $T - T_0 = T_2 \cos\phi + T_3 \sin\phi$, where $T_2 = T_2(t)$ and $T_3 = T_3(t)$ are half the temperature

differences of the top and the bottom, and the mid-height, respectively, and are solely a function of time.

The Navier-Stokes equation for convection yields

$$\frac{\partial q}{\partial t} = -\frac{1}{ua}\frac{\partial p}{\partial \phi} + g\alpha(T - T_0)\sin\phi - \Gamma q \tag{2.1}$$

where $g\alpha(T - T_0)$ is the buoyancy force and Γ is a generalized friction coefficient, corresponding to viscous resistance proportional to velocity. Full simplification of the system leads to

$$\frac{dq}{dt} = -\Gamma q + \frac{g\alpha T_3}{2} \tag{2.2}$$

The full temperature equation in convection for the cross-sectional average within the loop is

$$\frac{\partial T}{\partial t} + \frac{q}{a}\frac{\partial T}{\partial \phi} = K(T_E - T) \tag{2.3}$$

where K is thermal conductivity of the annulus. Proper substitution and separation of $\cos\phi$ and $\sin\phi$ terms give two equations on rate of change of the horizontal and vertical temperatures

$$\frac{dT_3}{dt} - \frac{qT_2}{a} = -KT_3 \tag{2.4}$$

$$\frac{dT_2}{dt} + \frac{qT_3}{a} = K(T_1 - T_2) \tag{2.5}$$

Let $T_4(t) = T_1 - T_2(t)$, and let

$$x = \frac{q}{aK}, y = \frac{g\alpha T_3}{2a\Gamma K}, z = \frac{g\alpha T_4}{2a\Gamma K} \tag{2.6}$$

Define the dimensionless time quantity $t' = tK$. Dropping the prime on t to obtain the Lorenz system with $b = 1$ for closed natural convection loop. Without any loss of generality, we will stick with the geometric parameter b instead of assigning a particular value for this quantity.

It is noticed that high dimensional Lorenz system exists, and the set of equations (1.1) is truncated after 3 most important modes are selected. For the original and

full derivation from Navier-Stokes with infinite set of ordinary differential equation, see [2] and [1], respectively.

CHAPTER 3

STABILITY ANALYSIS OF LINEAR STATE FEEDBACK CONTROL

As we can see in (2.3), the system loses its stability upon the inconsistency of horizontal temperature difference. For the purpose of controlling the system by a linear feedback controller, let us assume that the dynamical equations of the controlled Lorenz system are given by

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = rx - y - xz - k(y - y^*) \\ \dot{z} = xy - bz \end{cases} \tag{3.1}$$

where $y^* = \pm\sqrt{b(r-1)}$ is the nontrivial equilibrium. Our question is that what minimal k value will stabilize the system? And for smaller k value, how will the system behave? We will prove the stability of the translated system case by case; in order to simplify the process, let $c_1 = \sqrt{b(r-1)}$ and $c_2 = r - 1$.

3.1 Translation of the linear feedback control Lorenz system

Before introducing the translated Lorenz systems, we define $C^0 = (0, 0, 0)$ to be the origin of the Lorenz system (3.1), $C^+ = (\sqrt{b(r-1)}, \sqrt{b(r-1)}, r-1)$ and $C^- = (-\sqrt{b(r-1)}, -\sqrt{b(r-1)}, r-1)$ to be the positive and negative nontrivial equilibrium points of (3.1), respectively.

Case 1: $y^* = \sqrt{b(r-1)}$.

Before proposing a Lyapunov function, we translate the system with the transformation $x \to x + c_1, y \to y + c_1$ and $z \to z + c_2$. The translation reserves the dynamical behavior of Lorenz system, and simplifies the object of interest to the origin. The system (3.1) becomes

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = x - y - xz - c_1 z - ky \\ \dot{z} = xy - bz + c_1 x + c_1 y \end{cases} \tag{3.2}$$

If the Lyapunov function is chosen to be

$$V = \frac{1}{2}(x^2 + \alpha y^2 + \alpha z^2) \tag{3.3}$$

where $\alpha = \alpha(r, \sigma, b) > 0$ is a to-be-determined coefficient, then

$$\dot{V} = -\begin{bmatrix} x & y & z \end{bmatrix} Q_+ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3.4}$$

where $Q_+ = \begin{bmatrix} \sigma & -\frac{\sigma+\alpha}{2} & -\frac{\alpha\sqrt{b(r-1)}}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k+1) & 0 \\ -\frac{\alpha\sqrt{b(r-1)}}{2} & 0 & b\alpha \end{bmatrix}$

Case 2: $y^* = -\sqrt{b(r-1)}$.

Followed by Case 1, we translate the system with the transformation $x \to x - c_1, y \to y - c_1$ and $z \to z + c_2$

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = x - y - xz + c_1 z - ky \\ \dot{z} = xy - bz - c_1 x - c_1 y \end{cases} \tag{3.5}$$

and consider the Lyapunov function in (3.3) form, then

$$\dot{V} = -\begin{bmatrix} x & y & z \end{bmatrix} Q_- \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3.6}$$

where $Q_- = \begin{bmatrix} \sigma & -\frac{\sigma+\alpha}{2} & \frac{\alpha\sqrt{b(r-1)}}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k+1) & 0 \\ \frac{\alpha\sqrt{b(r-1)}}{2} & 0 & b\alpha \end{bmatrix}$

It is straightforward to verify that $|Q_+| = |Q_-| = |Q|$. The existence of the coefficient $\alpha > 0$ that make Q_+ and Q_- positive definite concludes the global stability of the equilibrium.

3.2 Global stability of linear feedback control Lorenz system

Observe that from the second equation in (3.1), the number of equilibrium points reduce to one - a unique equilibrium $y = y^* = \pm\sqrt{b(r-1)}$ when $k > k_G = (r-1)/4$

Theorem 3.1. *The system (3.1) is globally uniformly asymptotically stable if* $k > k_G$.

Proof. Let $Q = Q_\pm$

where $Q_\pm = \begin{bmatrix} \sigma & -\frac{\sigma+\alpha}{2} & \mp\frac{\alpha\sqrt{b(r-1)}}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k+1) & 0 \\ \mp\frac{\alpha\sqrt{b(r-1)}}{2} & 0 & b\alpha \end{bmatrix}$

Q is positive definite if and only if all the principal minor of $Q_\pm$ have positive determinants.

$$|Q_1| = \sigma > 0 \tag{3.7}$$

$$|Q_2| = \begin{vmatrix} \sigma & -\frac{\sigma+\alpha}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k+1) \end{vmatrix} = \frac{4\alpha\sigma k - (\alpha-\sigma)^2}{4} > 0 \tag{3.8}$$

$$|Q_3| = \begin{vmatrix} \sigma & -\frac{\sigma+\alpha}{2} & \mp\frac{\alpha\sqrt{b(r-1)}}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k+1) & 0 \\ \mp\frac{\alpha\sqrt{b(r-1)}}{2} & 0 & b\alpha \end{vmatrix} \tag{3.9}$$

$$= -\frac{b\alpha}{4}\{r\alpha^2 + k\alpha[(r-1)\alpha - 4\sigma] + \sigma(-2\alpha+\sigma)\} > 0$$

From (3.8):

$$k > \frac{(\alpha-\sigma)^2}{4\alpha\sigma} > 0 \tag{3.10}$$

From (3.9):

$0 < (r-1)\alpha^2 + (\alpha-\sigma)^2 < k\alpha[4\sigma - (r-1)\alpha]$

which means

$$k > \frac{(r-1)\alpha^2 + (\alpha-\sigma)^2}{\alpha[4\sigma - (r-1)\alpha]} > 0 \tag{3.11}$$

and since $\alpha > 0$, then $4\sigma - (r-1)\alpha > 0$. This implies α has to be in $\left(0, \frac{4\sigma}{r-1}\right)$. Take the right hand side quantity of (3.11) subtract this quantity of (3.10):

$$\frac{(r-1)\alpha^2+(\alpha-\sigma)^2}{\alpha[4\sigma-(r-1)\alpha]} - \frac{(\alpha-\sigma)^2}{4\alpha\sigma} = \frac{(r-1)(\alpha+\sigma)^2}{4\sigma[4\sigma-\alpha(r-1)]} > 0 \tag{3.12}$$

Thus for this range of α, (3.11) implies (3.10). The minimum gain k from (3.11) is k_G, when $\alpha = \frac{2\sigma}{r+1}$. Let

$$V = \frac{1}{2}\{(r+1)x^2 + 2\sigma y^2 + 2\sigma z^2\} \tag{3.13}$$

Since $V(x, y, z)$ is positive definite, radially unbounded and decrescent, and $\dot{V}(x, y, z)$ is negative definite, by LaSalle's Invariance Theorem, the origin of the translated system is globally uniformly asymptotically stable [13]. □

3.3 Local stability of linear feedback control Lorenz system

The question is that how the sytem behaves once k value drops below k_G value? To answer the question, we compute the Jacobian matrix J of the control Lorenz system (3.1) is

$$J = \begin{bmatrix} -\sigma & \sigma & 0 \\ r-z & -1-k & -x \\ y & x & -b \end{bmatrix} \tag{3.14}$$

and thus, the characteristic polynomial of J is computed as

$$p(\lambda) = \lambda^3 + \beta_2\lambda^2 + \beta_1\lambda + \beta_0 \tag{3.15}$$

where $\beta_0 = \sigma[x(x+y)+b(z+k+1-r)]$, $\beta_1 = x^2+(z+k+1-r)\sigma+b(\sigma+k+1)$, $\beta_2 = \sigma+b+1+k$.

The fact that β_2 is always positive implies that the unstable manifold is at most two-dimensional at every point on the state trajectory in phase space. The discriminant of the cubic equation is defined as

$$\Delta = -\beta_2^2\beta_1^2 + 4\beta_2^3\beta_0 + \beta_1^3 - 18\beta_2\beta_1\beta_0 + 27\beta_0^2 \tag{3.16}$$

The Jacobian matrices for $C^{\pm}$ in (3.2) and (3.5) are

$$J^{\pm} = \begin{bmatrix} -\sigma & \sigma & 0 \\ 1 & -1-k & \mp\sqrt{b(r-1)} \\ \pm\sqrt{b(r-1)} & \pm\sqrt{b(r-1)} & -b \end{bmatrix} \tag{3.17}$$

It is noted that the determinant of the Jacobian matrix does not change with respect to the translation. Due to the symmetry of Lorenz system, both the equilibrium points $C^{\pm}$ share the same characteristic equations

$$\lambda(J^-) = \lambda(J^+) = \lambda^3 + \lambda^2(\sigma + b + 1 + k) + \lambda[br + b\sigma + (\sigma + b)k] + b\sigma[2(r-1) + k] \tag{3.18}$$

Pure imaginary solutions occurs if the product of the coefficients of λ^2 and λ equal to the constant term

$$(\sigma + b + 1 + k)[br + b\sigma + (\sigma + b)k] = b\sigma[2(r-1) + k] \tag{3.19}$$

Suppose $\sigma > b + 1$, then

$$k_L = \frac{1}{2}\sqrt{(\sigma + b + 1 + \frac{br}{\sigma+b})^2 + \frac{4b(\sigma-b-1)}{\sigma+b}(r - r_c)} - \frac{1}{2}(\sigma + b + 1 + \frac{br}{\sigma+b}) \tag{3.20}$$

where $r_c = \frac{\sigma(\sigma+b+3)}{\sigma-b-1}$ is the critical value of r. When $k_G > k > k_L$, either all the eigenvalues of the Jacobian $J^{\pm}$ have negative real parts or two of them are positive and one is negative. In fact, numerical simulation shows that the linear state feedback control Lorenz system (3.1) is stablized for very many intial values. The following lemma eliminates the second case and establishes the local stability of $C^{\pm}$.

Theorem 3.2. *For $k > k_L$, $C^{\pm}$ are either stable foci ($0 > Re\{\lambda_{1,2}\} > \lambda_3$) or stable nodes ($0 > \lambda_1 > \lambda_2 > \lambda_3$).*

Proof. At $C^{\pm}$, β_i are

$$\beta_2 = \sigma + b + 1 + k \tag{3.21}$$

$$\beta_1 = br + b\sigma + (\sigma + b)k \tag{3.22}$$

$$\beta_0 = b\sigma\,[2(r-1) + k] \tag{3.23}$$

General theory [14] suggests that $C^{\pm}$ are stable foci if and only if $\beta_2 > 0$ and

$$\beta_2\beta_1 > \beta_0 > \begin{cases} \beta_0^+(\beta_2,\beta_1) > 0, & \text{for } 0 < \beta_1 \leq \frac{\beta_2^2}{3} \\ \frac{\beta_2}{3}\left(\beta_1 - \frac{2\beta_2^2}{9}\right), & \text{for } \beta_1 \geq \frac{\beta_2^2}{3} \end{cases} \tag{3.24}$$

where $\beta_0^{\pm} = \beta_0^{\pm}$ is defined by the following formula

$$\beta_0^{\pm} = \frac{1}{3}\beta_2\beta_1 - \frac{2}{27}\beta_2^3 \pm \frac{2}{27}(\beta_2^2 - 3\beta_1)^{3/2} \tag{3.25}$$

From (3.18), $k > k_L$ implies that $\beta_2\beta_1 > \beta_0$.

$C^{\pm}$ are stable node if and only $\beta_2 > 0$ and

$$0 < \beta_0 < \begin{cases} \beta_0^+(\beta_2,\beta_1), & \text{for } 0 < \beta_1 \leq \frac{\beta_2^2}{3} \\ \frac{\beta_2}{3}\left(\beta_1 - \frac{2\beta_2^2}{9}\right), & \text{for } \beta_1 \geq \frac{\beta_2^2}{3} \end{cases} \tag{3.26}$$

It is clear that β_0 is always positive but unclear how β_0 compares to these values. Either case, the local stability of $C^{\pm}$ is guaranteed by the negative real parts of all eigenvalues. □

Setting $\frac{\beta_2^2}{3} - \beta_1 = 0$ yields

$$k^2 - (\sigma + b - 2)k - [3(br + b\sigma) - (\sigma + b + 1)^2] = 0 \tag{3.27}$$

However, for $k \in (k_L, k_G)$, there are two other auxiliary equilibirum generated by the controller, associated with either C^+ or C^-. Define P_1, P_2 to be two auxiliary equilibrium associated with C^+, N_1, N_2 to be two auxiliary equilibrium associated with C^-. Setting $\dot{x} = \dot{y} = \dot{z} = 0$ in (3.2) and (3.5) gives the post-transled coordinates of P_1^*, P_2^* and N_1^*, N_2^*. Note that the coordinates of these auxiliary equilibrium can be translated back and forth.

$$P_{1,2}^* = P_i^* = \left(\frac{-3c_1 \pm \sqrt{\delta}}{2}, \frac{-3c_1 \pm \sqrt{\delta}}{2}, \frac{-3c_1^2 + \delta \mp 2c_1\sqrt{\delta}}{4b}\right) \tag{3.28}$$

$$N^*_{1,2} = N^*_i = \left(\frac{3c_1 \pm \sqrt{\delta}}{2}, \frac{3c_1 \pm \sqrt{\delta}}{2}, \frac{-3c_1^2 + \delta \pm 2c_1\sqrt{\delta}}{4b}\right) \tag{3.29}$$

where $\delta = c_1^2 - 4bk$

The pre-translated coordinates of P_i and N_i can be solved by setting $\dot{x} = \dot{y} = \dot{z} = 0$ in(3.1)

$$P_{1,2} = P_i = \left(\frac{-c_1 \pm \sqrt{\delta}}{2}, \frac{-c_1 \pm \sqrt{\delta}}{2}, \frac{c_1^2 + \delta \mp 2c_1\sqrt{\delta}}{4b}\right) \tag{3.30}$$

$$N_{1,2} = N_i = \left(\frac{c_1 \pm \sqrt{\delta}}{2}, \frac{c_1 \pm \sqrt{\delta}}{2}, \frac{c_1^2 + \delta \pm 2c_1\sqrt{\delta}}{4b}\right) \tag{3.31}$$

It is noted that when $k \geq k_G$, $\delta \leq 0$ and by Theorem 3.1, the system is globally stable.

Theorem 3.3. *Suppose $k_L < k < k_G$ then*

(a) P_1 is a saddle-focus.

(b) P_2 is either a stable focus, a saddle-focus or a sadlle.

Proof. (a). At P_1, β_i are computed as

$$\beta_2 = \sigma + b + 1 + k \tag{3.32}$$

$$\beta_1 = \ -\tfrac{1}{2}\Big[(b+\sigma)\sqrt{(r-1)(r-1-4k)} + r\sigma - \sigma - b(1+r+2\sigma)\Big] \tag{3.33}$$

$$\beta_0 = \frac{b\sigma}{2}\Big[(r-1-4k) - 3\sqrt{(r-1)(r-1-4k)}\Big] \tag{3.34}$$

Assume $r > r_c = \frac{\sigma(\sigma+b+3)}{\sigma-b-1}$, and observe that $\frac{\sigma(\sigma+b+3)}{\sigma-b-1} > \frac{b+\sigma+2b\sigma}{\sigma-b}$ holds for all σ and b. This implies $r\sigma - \sigma > b(1+r+2\sigma)$, and therefore, $\beta_1 < 0$.

Suppose that $\lambda_1 > \lambda_2 > \lambda_3$ are the roots of (3.15). According to Vieta's formula,

$$\lambda_1 + \lambda_2 + \lambda_3 = -\beta_2 < 0 \tag{3.35}$$

$$\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1 = \beta_1 < 0 \tag{3.36}$$

$$\lambda_1\lambda_2\lambda_3 = -\beta_0 > 0 \tag{3.37}$$

Consider the discriminant of P_1

$$\begin{aligned}\Delta_{P_1} = \; & \tfrac{27}{4}b^2\sigma^2\left[-3\sqrt{(r-1)(-4k+r-1)} - 4k + r - 1\right]^2 \\ & +2b\sigma\left[-3\sqrt{(r-1)(-4k+r-1)} - 4k + r - 1\right](b+k+\sigma+1)^3 \\ & -\tfrac{1}{4}(b+k+\sigma+1)^2\left[-(b+\sigma)\sqrt{(r-1)(-4k+r-1)} + b(r+2\sigma+1) - r\sigma + \sigma\right]^2 \\ & -\tfrac{9}{2}b\sigma\left[-3\sqrt{(r-1)(-4k+r-1)} - 4k + r - 1\right](b+k+\sigma+1) \\ & \left[-(b+\sigma)\sqrt{(r-1)(-4k+r-1)} + b(r+2\sigma+1) - r\sigma + \sigma\right] \\ & +\tfrac{1}{2}\left[-(b+\sigma)\sqrt{(r-1)(-4k+r-1)} + b(r+2\sigma+1) - r\sigma + \sigma\right]^3\end{aligned} \tag{3.38}$$

Since the following inequality holds $\left[-3\sqrt{(r-1)(-4k+r-1)} - 4k + r - 1\right] < 0$, it follows that $\Delta_{P_1} < 0$ for $r > r_C$, therefore, there are three distinct real eigenvalues. This concludes that $\lambda_1 > 0 > \lambda_2 > \lambda_3$ and completes the proof.

(b) At P_2,

$$\beta_2 = \sigma + b + 1 + k \tag{3.39}$$

$$\beta_1 = \; \tfrac{1}{2}\left[(b+\sigma)\sqrt{(r-1)(r-1-4k)} + b(1+r+2\sigma) + \sigma - r\sigma\right] \tag{3.40}$$

$$\beta_0 = \frac{b\sigma}{2}\left[r - 1 - 4k + 3\sqrt{(r-1)(r-1-4k)}\right] \tag{3.41}$$

Let k_S be the unique value that makes $\beta_1 = 0$. If $k < k_S$, then $\beta_1 > 0$; if $k > k_S$, then $\beta_1 < 0$.

$$k_S = \frac{1}{4}\left\{ r - 1 - \frac{[\sigma(r-1) - b(1+r+2\sigma)]^2}{(b+\sigma)^2(r-1)} \right\} = \frac{b(r+\sigma)(r\sigma - b\sigma - b - \sigma)}{(r-1)(b+\sigma)^2} \tag{3.42}$$

According to Vieta's formula,

$$\lambda_1 + \lambda_2 + \lambda_3 = -\beta_2 < 0 \tag{3.43}$$

$$\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1 = \begin{cases} \beta_1 > 0, & \text{if } k < k_S \\ \beta_1 \leq 0, & \text{if } k \geq k_S \end{cases} \tag{3.44}$$

$$\lambda_1\lambda_2\lambda_3 = -\beta_0 < 0 \tag{3.45}$$

If $k > k_S$, it follows that $\lambda_1 \geq \lambda_2 > 0 > \lambda_3$. General theory [14] states that $\mathrm{Re}\{\lambda_{1,2}\} > 0 > \lambda_3$ if and only

$$\beta_0 > \begin{cases} \beta_0^+(\beta_2, \beta_1), & \text{for } \beta_1 < 0 \\ \beta_2\beta_1, & \text{for } \beta_1 \geq 0 \end{cases} \tag{3.46}$$

Indeed, if $k_L < k < k_S$, then there exists a range of k make P_2 a stable-focus. So k have to be greater than k_S. □

The unstable manifolds W_P^u (or W_N^u) are composed of the saddle points P (or N) themselves and two tracjectories that come from P (or N) as $t \to +\infty$. The stable manifold W_P^s (or W_N^s) is given by the eigenvector corresponding to the smallest negative characteristic root. The boundedness of the control Lorenz system is established upon the following theorem

Theorem 3.4. *There exists a bounded ellipsoid $\mathcal{E}$, which is the trapping region for the control Lorenz flow (3.1). ; that is, if a trajectory enters $\mathcal{E}$ at sometime, it will*

stay there and never thereafter leaves. In addition, every trajectory will enter $\mathcal{E}$ *in finite time.*

Proof. Define $V = \frac{1}{\sigma}x^2 + (y \pm c_1)^2 + z^2$ as a Lyapunov function. Then along any trajectory of (3.2) or (3.5),

$$\dot{V} = \ -[x - (y \pm \tfrac{c_1}{2})]^2 - k(y \mp c_1)^2 - b(z - \tfrac{c_1^2}{2b})^2 + \tfrac{k+1}{4}c_1^2 + \tfrac{c_1^4}{4b} \tag{3.47}$$

Define $\mathcal{F} = \{\dot{V} \geq 0\}$. $\mathcal{F}$ is a bounded region. Pick an arbitrary $\epsilon > 0$, and let V_{max} be the maximum of V in $\mathcal{F}$ and $\mathcal{E} = \{V \leq V_{max} + \epsilon\}$, then $\{V\} \subset \mathcal{E}$. Outside $\mathcal{E}$, $\dot{V} \leq -\delta$, for some $\delta > 0$. With an intial condition (x_0, y_0, z_0) outside $\mathcal{E}$, the value of $V(x, y, z)$ will decrease within finite time, and eventually enter the ellipsoid $\mathcal{E}$. All trajectories pass inward the ellipsoid $\mathcal{E}$ so that the trajectory once within $\mathcal{E}$, it will stay there forever. □

Summarize Lemma 3.3 and Lemma 3.4, we now can conclude the following:

Theorem 3.5. *Suppose* $k_G > k > k_L$*, the equilibrium points* $C^{\pm}$ *are locally asymptotically stable. Its basin of attraction is* $\mathbb{R}^3 \backslash \{W^s_{P_1} \cup W^s_{P_2}\}$ *or* $\mathbb{R}^3 \backslash \{W^s_{N_1} \cup W^s_{N_2}\}$

3.4 Global stability analysis of adpative control Lorenz system

Consider the system (3.1) with k as a state variable. The feedback control law thus far requires exact values of r, σ and b. However, determining these values in real life could be impossible. To overcome these drawbacks, an adaptive technique is employed to adjust the feedback gain accordingly. The governing equations for the system are

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = rx - y - xz - k(y - y^*) \\ \dot{z} = xy - bz \\ \dot{k} = \gamma(y - y^*)^2 \end{cases} \tag{3.48}$$

where γ is the adpaptive gain, controlling the speed of the adaptive control process. The equilibrium are $(x^*, y^*, z^*, k^*) = (\pm\sqrt{b(r-1)}, \pm\sqrt{b(r-1)}, r-1, k^*)$ for $\forall k^* \in \mathbb{R}$, and they are unique in the sense of (x, y, z) . Again, we will translate the system to the two equilibrium points and combine them based on their similarity. The adaptive control Lorenz system after translation is described by setting $x \to x + c_1, y \to y + c_1, z \to z + c_1, k \to k + k_G$, the system (3.48):

$$\begin{cases} \dot{x} = \sigma(-x + y) \\ \dot{y} = x - y - xz \mp c_1 z - (k + k_G)y \\ \dot{z} = xy - bz \pm c_1 x \pm c_1 y \\ \dot{k} = \gamma y^2 \end{cases} \tag{3.49}$$

Based on Theorem 3.1, we proposed the following theorem for adpative global stablity of the adaptive control Lorenz system (3.48).

Theorem 3.6. *The trajectory of the adaptive controlled Lorenz system is globally uniformly asymptotically stable to the equilibrium* $(\pm\sqrt{b(r-1)}, \pm\sqrt{b(r-1)}, r-1, k^*)$

Proof. If the Lyapunov function is chosen to be

$$V = \frac{1}{2}(x^2 + \alpha y^2 + \alpha z^2 + \frac{1}{\gamma}\alpha k^2) \tag{3.50}$$

then

$$\dot{V} = -\begin{bmatrix} x & y & z \end{bmatrix} Q \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3.51}$$

where $Q = \begin{bmatrix} \sigma & -\frac{\sigma+\alpha}{2} & \mp\frac{\alpha\sqrt{b(r-1)}}{2} \\ -\frac{\sigma+\alpha}{2} & \alpha(k_G+1) & 0 \\ \mp\frac{\alpha\sqrt{b(r-1)}}{2} & 0 & b\alpha \end{bmatrix}$

By Lemma 1, Lyapunov function of the form (3.50) is positive definite and decrescent, and $\dot{V}$ is negative semidefinite. By Lyapunov theory [15], the equilibrium point of the system (3.48) is uniformly stable, i.e. $x(t), y(t), z(t) \in \mathrm{L}_\infty$ and $k(t) \in \mathrm{L}_\infty$. From (3.51), the square of $x(t), y(t)$ and $z(t)$ are integrable with respect to time, i.e. $x(t), y(t), z(t) \in \mathrm{L}_2$. By Barbalat's lemma, for any initial condition, equation 3.1 implies that $\dot{x}(t), \dot{y}(t), \dot{z}(t) \in \mathrm{L}_\infty$, which in turn implies $x(t), y(t)$ and $z(t) \to 0$ as $t \to \infty$. This implies the trajectory of the controlled Lorenz system is globally asymptotically stabilzed to the equilibrium point stated above. □

CHAPTER 4

SYNCHRONIZATIONS OF IDENTICAL LORENZ SYSTEM

In this chapter, we investigate the bidirectionally coupled Lorenz loops with the introduction of 2 new parameters: N and μ. Following Jackson [16], we present a model built upon a "chain" of vortical fluid cells, each of whose dynamics are described by (1.1), except that they are coupled together by viscous effects and thermal couplings. The below model for Rayleigh-Benard turbulence generalizes Lorenz system and assumes that the basic structure of the fluid motion consists of vortices which persist for a range of r that includes turbulent behavior. The proposed model is presented in Figure 4.1.

Thus, cell i and $i+1$ have opposite sign for $x(t)$, whereas $i-1$ and $i+1$ cell have same signs. When $x_i(t) + x_{i+1}(t) \neq 0$, there will be a viscous force at this interface, which we assume to be $\mu(x_i(t)+x_{i+1}(t))$. It is noted that the positive sign takes place in the coupling because the counter-rotation of adjacent vortices reduces the viscous force, i.e. if $x_i(t) > 0$, then $x_{i+1}(t) < 0$ and the velocity difference is $x_i + x_{i+1}$ rather than $x_i - x_{i+1}$. Also, if there is a temperature difference $T_i(t) + T_{i+1}(t)$ near $\phi = \frac{\pi}{2}$, there will be a heat exchange, modeled by $N\big(T_i(t) + T_{i+1}(t)\big)$, which is proportional to $T_i(t) + T_{i+1}(t)\delta(\phi - \frac{\pi}{2})$

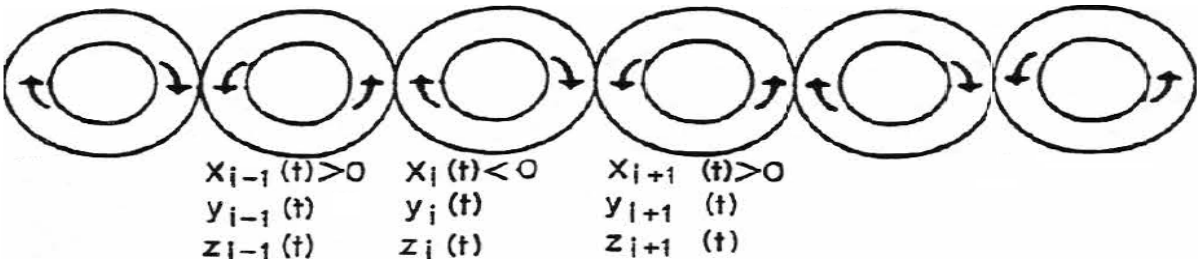

Figure 4.1: Schematic plot of coupled Lorenz-cell configuration. The arrows indicate the flow of direction in each cell. Photo courtesy of E. A. Jackson [16] with slight modification.

Combined these ideas, we model the 1D horizontal Lorenz "chain" as

$$\begin{cases} \dot{x_i} = \sigma(y_i - x_i) - \mu(x_{i+1} + x_{i-1} + 2x_i) \\ \dot{y_i} = rx_i - y_i - x_i z_i - N(y_{i+1} + y_{i-1} + 2y_i) \\ \dot{z_i} = x_i y_i - bz_i \end{cases} \tag{4.1}$$

It is noted that μ and N are related to Reynolds and Nusselt in some way, and therefore, depends on the fluid properties and the system constant σ, r and b; however, in this work, we will assume that those parameters are constant and will investigate the stability of the system based on the range of these paramters.

The set of equations governing the bidirectionally coupled Lorenz system are therefore modeled as

$$\begin{cases} \dot{x_1} = \sigma(-x_1 + y_1) - \mu(x_1 + x_2) \\ \dot{y_1} = rx_1 - y_1 - x_1 z_1 - N(y_1 + y_2) \\ \dot{z_1} = x_1 y_1 - bz_1 \\ \dot{x_2} = \sigma(-x_2 + y_2) - \mu(x_1 + x_2) \\ \dot{y_2} = rx_2 - y_2 - x_2 z_2 - N(y_1 + y_2) \\ \dot{z_2} = x_2 y_2 - bz_2 \end{cases} \tag{4.2}$$

We present a few analogous lemmas and theorems to a single loop system.

Theorem 4.1. *There exists a bounded 6-tupled ellipsoid $\mathcal{E}_2$, which is the trapping region for the bidirectionally coupled Lorenz system flow (4.2). If a trajectory enters $\mathcal{E}_2$ at sometime, it will stay there and never thereafter leaves. In addition, every trajectory will enter $\mathcal{E}_2$ in finite time.*

Proof. Since the proof is very similar to Theorem 3.4, we will neglect the technical details and begin with the Lyapunov function for $\mathcal{E}_2$. In this case, consider

$$V = \frac{1}{2}\Big[\frac{r}{\sigma}\Big(x_1^2 + x_2^2) + (y_1^2 + y_2^2) + (z_1 - 2r)^2 + (z_2 - 2r)^2)\Big]$$

Then, along any trajacetory of (4.2),

$$\dot{V} = -\frac{r\mu}{\sigma}(x_1+x_2)^2 - x_1^2 - x_2^2 - N(y_1^2+y_2^2) - y_1^2 - y_2^2 - b(z_1-r)^2 - b(z_2-r)^2 + 2br^2 \quad (4.3)$$

Outside $\mathcal{E}_2$, $\dot{V} < 0$. With an initial condition outside $\mathcal{E}_6$, the value of $V(x_i, y_i, z_i)$ will decrease within finite time, and eventually enter $\mathcal{E}_2$. □

To establish the synchronization of the bidirectionally coupled Lorenz systems, let $e_x = x_1 + x_2$, $e_y = y_1 + y_2$ and $e_z = x_1 - z_2$. From (4.2),

$$\begin{cases} \dot{e_x} = \sigma(-e_x + e_y) - 2\mu e_x \\ \dot{e_y} = re_x - e_y - (x_1 z_1 + x_2 z_2) - 2Ne_y \\ \dot{e_z} = (x_1 y_1 - x_2 y_2) - be_z \end{cases} \quad (4.4)$$

Following Theorem 4.1, there exist 3 positive constants s_1, s_2, s_3 such that $|x_i(t)| \leq s_1 < \infty$, $|y_i(t)| \leq s_2 < \infty$, and $|z_i(t)| \leq s_3 < \infty$ hold for all $t \geq 0$ and $i = 1, 2$. The synchronization of two identical, bidirectionally coupled Lorenz systems is granted upon the following theorem

Theorem 4.2. *Two bidirectionally coupled Lorenz systems under (4.2) are globally asymptotically synchronized if the following conditions are satisfied:*

(i)

$$\Big(r + \frac{2\mu r}{\sigma}\Big)(2N+1) - \Big(r + \frac{s_3}{2}\Big)^2 > 0 \quad (4.5)$$

(ii)

$$b\Big[\Big(r + \frac{2\mu r}{\sigma}\Big)(2N+1) - (r + \frac{s_3}{2})^2\Big] - (2N+1)\Big(\frac{s_2^2}{4}\Big) > 0 \quad (4.6)$$

Proof. Let $V = \frac{1}{2}\left(\frac{r}{\sigma}e_x^2 + e_y^2 + e_z^2\right)$ then

$$\begin{aligned}\dot{V} &= -(r + \tfrac{2r\mu}{\sigma})e_x^2 + 2re_xe_y - e_y^2 - 2Ne_y^2 - e_x(z_2e_y + y_2e_z) - be_z^2 \\ &\leq -(r + \tfrac{2r\mu}{\sigma})|e_x|^2 + 2r|e_x||e_y| - |e_y|^2 - 2N|e_y|^2 + s_3|e_x||e_y| + s_2|e_x||e_z| - b|e_z|^2\end{aligned} \tag{4.7}$$

If the upper bound of $\dot{V}$ is less than or equal to zero, obviously $\dot{V} \leq 0$. Expressing (4.7) as matrix form, we have

$$\dot{V} \leq -\begin{bmatrix}|e_x| & |e_y| & |e_z|\end{bmatrix}\begin{bmatrix} r + \frac{2\mu r}{\sigma} & -\left(r + \frac{s_3}{2}\right) & -\frac{s_2}{2} \\ -\left(r + \frac{s_3}{2}\right) & 2N + 1 & 0 \\ -\frac{s_2}{2} & 0 & b \end{bmatrix}\begin{bmatrix}|e_x| \\ |e_y| \\ |e_z|\end{bmatrix} \tag{4.8}$$

The aforementioned condition are sufficient and necessary to make the middle matrix positive definite, and therefore $\dot{V} < 0$ holds. Since V is a positive and descrescent function and $\dot{V}$ is negative semidefinite, we can conclude that the two Lorenz systems are globally asymptotically synchronized. □

Indeed, one can choose $s_1 = \frac{2br^2}{\frac{r\mu}{\sigma}+1}$, $s_2 = \frac{2br^2}{N+1}$, $s_3 = r(1+\sqrt{2})$ for the bounded trajectory in Theorem 4.1. As such, possible choices of μ and N for the synchronization problem are $\mu = \sigma r^2$ and $N = r$. Feasible ranges of these parameters are soon to be explored in future work.

CHAPTER 5

NUMERICAL SIMULATION BY MATLAB AND DISCUSSION

5.1 Numerical simulation by MATLAB

To validate the theory, we employed MATLAB to simulate Lorenz system with 4^{th} order Runge - Kutta method in 10 seconds. In the first 5 seconds, the controller is taken off to demonstrate the chaotic behavior of the system. The linear controller is activated at $t = 5s$. The step-size of the simulation is $0.001s$ and the initial value is randomly generate by computer. The Lorenz system variables are $b = 1$, $\sigma = 10$, $r = 100$; for these values, $k_G = 24.75000$, $k_S = 9.08173$, $k_L = 3.03914$.

Figure 5.1 presents the behavior of the control Lorenz system described in the set of equations (3.2) on x, y, z axis, respectively. Figure 5.2 presents the deviation of the flow from the equillibrium desired points $C^{\pm}$ after the controller is activated. Figure 5.3 presents a space plot of control Lorenz flow, from $t = 0s$ to $t = 10s$. Figure 5.4 presents a zoom-in picture of closed feedback control Lorenz flow; the equilibrium point $C^{\pm}$ is plotted as a red star. Figure 5.5 presents the real and imaginary part of Jacobian eigenvalues of C^{+}, P_1 and P_2, respectively with respect to k; in here, the k gain value varies from 0 to $\frac{r-1}{4}$. The simulation results agree with Lemma 3.2 and Lemma 3.3 up on the calculated k_G, k_S and k_L values.

Figure 5.1 to Figure 5.5 feature the closed feedback control Lorenz system described in (3.1) with $b = 1$, $\sigma = 10$, $r = 100$, $k = 10$, the space plot and Jacobian eigenvalues analysis.

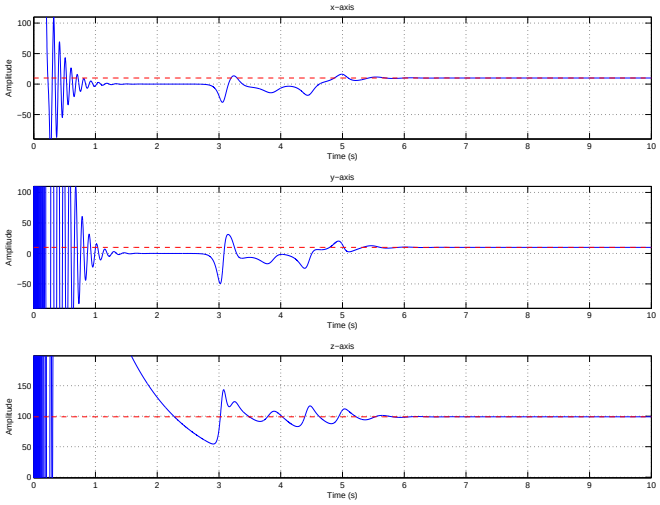

Figure 5.1: Linear feedback control Lorenz system behavior. The controller is activated at $t = 5s$. Red lines denote the desired equilibrium point C^+.

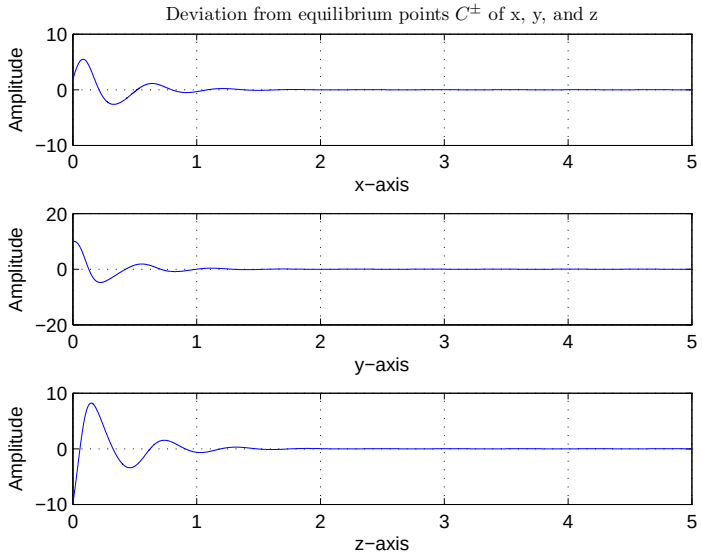

Figure 5.2: Deviation of the flow from C^+ after the linear controller is activated.

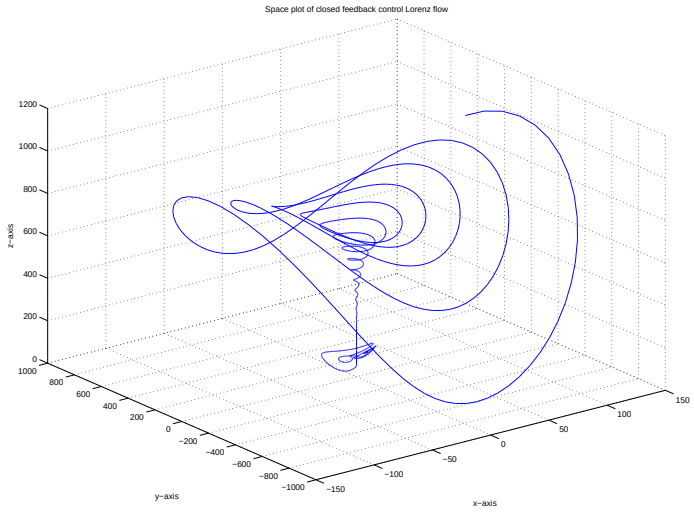

Figure 5.3: Space plot of closed feedback control Lorenz flow.

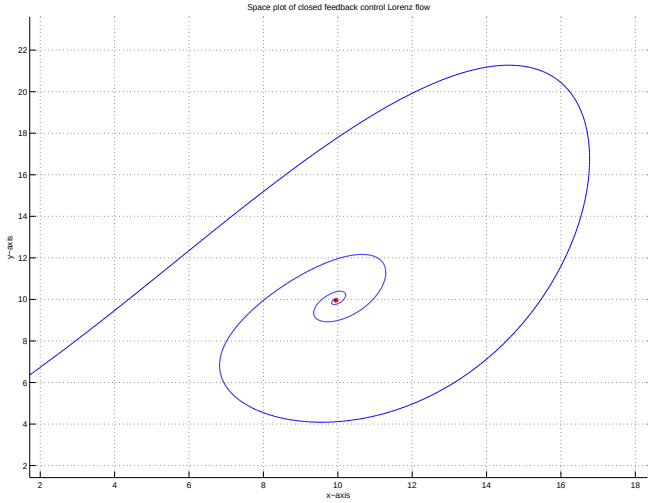

Figure 5.4: Zoom-in space plot after the linear controller is activated. C^+ is plotted as a small red star.

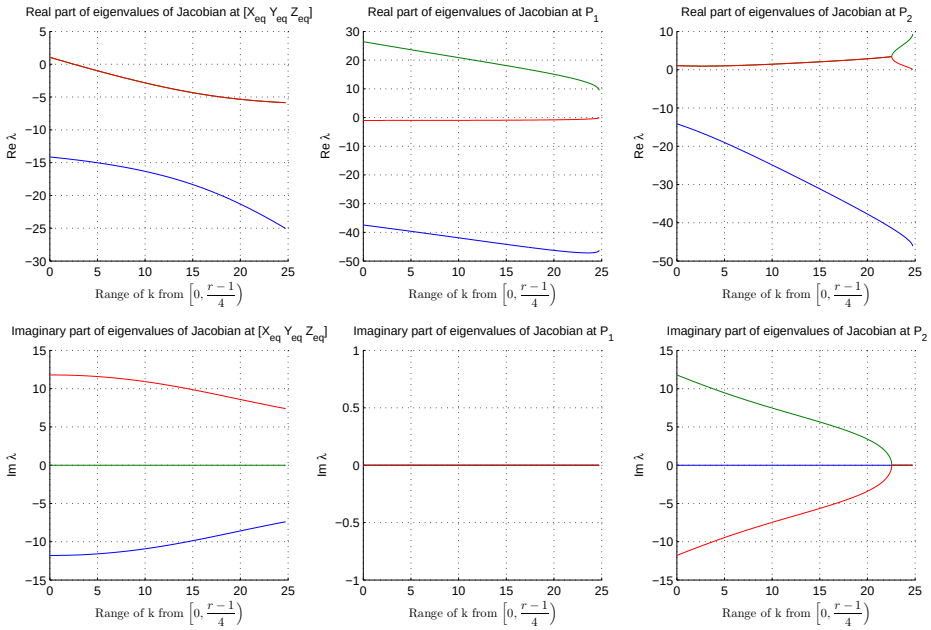

Figure 5.5: Eigenvalues of Jacobian matrix of C^+, P_1 and P_2, respectively.

Simliarly, Figure 5.6 presents the adaptive control system behavior described by the set of equation (3.48). Figure 5.7 plots the deviation from the desired equilibrium point C^+ with respect to time. Figure 5.8 and Figure 5.9 provides a space plot of the adaptive control Lorenz flow. As we observe in Figure 5.6, the k gain value continues rising until the flow stabilizes at C^+.

Figure 5.6 to Figure 5.9 feature the adaptive control Lorenz system described in (3.48), $b = 1$, $\sigma = 10$, $r = 100$, $\gamma = 10^{-2}$ space plot.

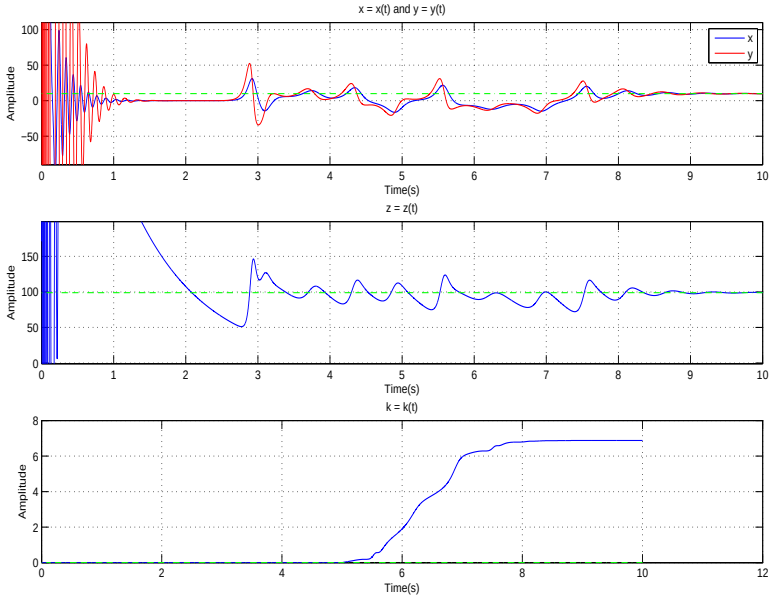

Figure 5.6: Adaptive control Lorenz system behavior. The controller is activated at $t = 5s$. Green line denotes the desired equilibrium point C^+.

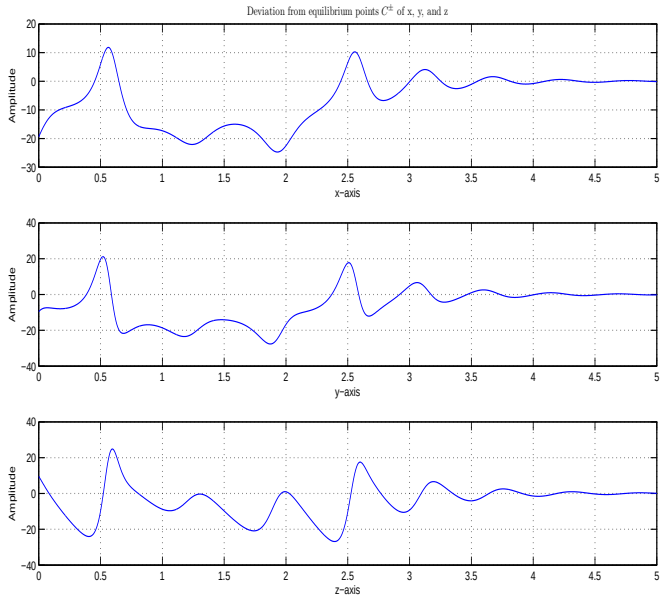

Figure 5.7: Deviation of the flow from C^+ after the adaptive controller is activated.

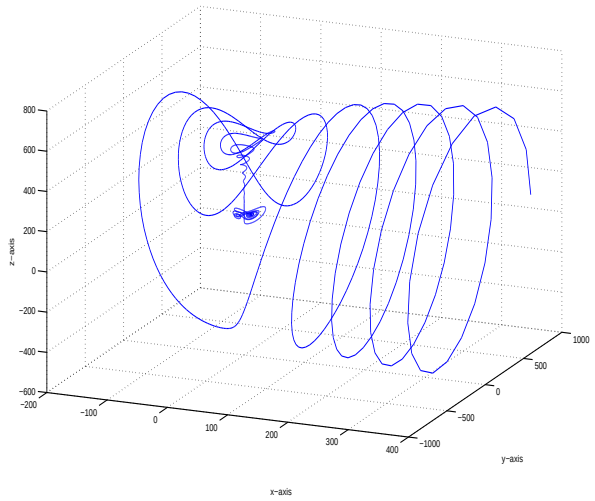

Figure 5.8: Space plot of adaptive control Lorenz flow.

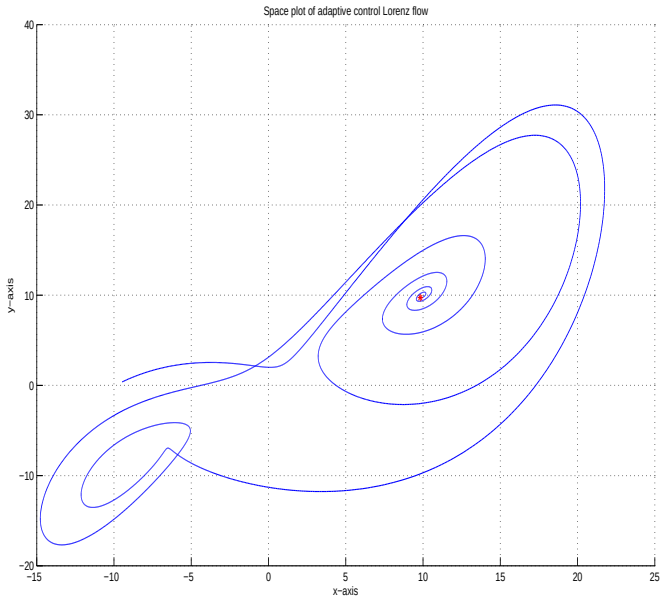

Figure 5.9: Zoom-in space plot after the adaptive controller is activated. C^+ is plotted as a small red star.

Figure 5.10 shows the transient response of the bidirectionally coupled Lorenz system, where the initial values for both loops are randomly chosen. The system constant are $\sigma = 10, b = 1, r = 100$. The coupling constants are assumed to be $\mu = \sigma \cdot r$, $N = r$; as such, the necessary conditions revealed in 4.2 are satisfied. The synchronization numerical results are presented in Figure 5.11; numerical simulations and theoretical results show good agreement and demonstrate that the two Lorenz systems have been asymptotically synchronized using the proposed coupling constant schemes.

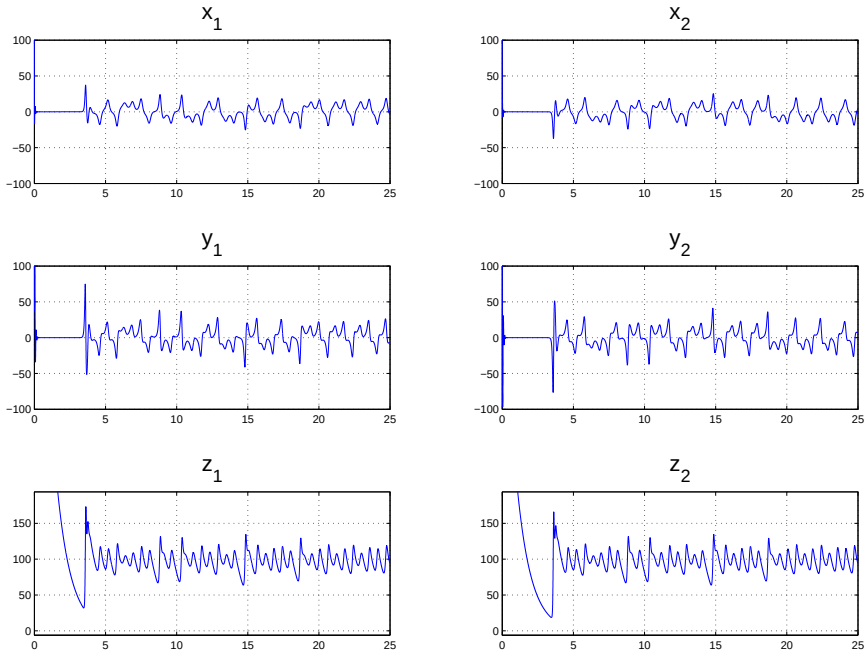

Figure 5.10: Transient response of two bidirectionally coupled Lorenz systems

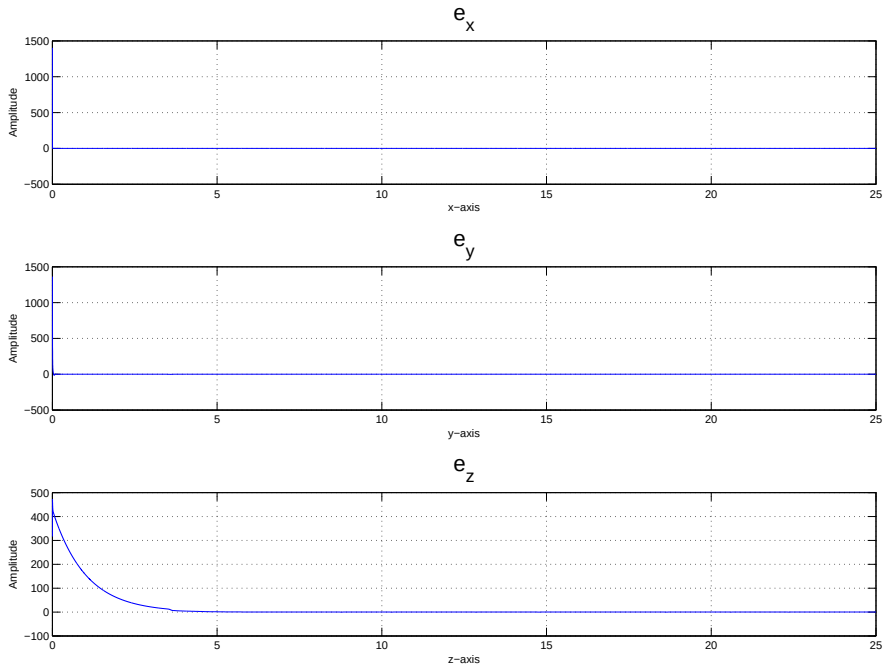

Figure 5.11: State response of bidirectionally coupled synchronization.

5.2 Future work

The results of this thesis point to several directions of future work:

1. Stability of multi-equilibrium of bidirectionally coupled Lorenz systems.

2. Implement a controller or an adaptive controller to either synchronize or drive the state trajectory to the desired equilibrium point

3. Analysis of coupling constants, such as μ and N.

4. Develop a 1D adjacent coupled Lorenz "chain". This has a direct application in thin film fluid motion with perturbed heat flux.

5. Develop a 2D computational fluid dynamics model with vertical coupling included. A possible direction is to use Finite Volume Method, with each cell now is a single Lorenz system

6. Develop a 3D computational fluid dynamics model and compare the simulation results with the experimental results.

In the near future, this topic will soon to be revisited for further development.

REFERENCES

[1] Lorenz, E. N. (1963) Deterministic nonperiodic flow. *J. Atmos. Sci.,* **20**, 130–141.

[2] Saltzmann, B. (1962) Finite amplitude free convection as an initial value problem - I. *J. Atmos. Sci.,* **19**, 329–341.

[3] Roy, D. and Musielak, Z. E. (2007) Generalized Lorenz models and their routes to chaos. I. Energy-conserving vertical mode truncations. *Chaos, Solitions and Fractals,* **32**, 1038–1052.

[4] Liao, T. L. and Lin, S. H. (1999) Adaptive control and synchronization of Lorenz systems. *Journal of Franklin Institute,* **336**, 925–937.

[5] Hartley, T. T. and Mossayebi, F. (1992) A classical approach to controlling the Lorenz equations. *International Journal of Bifurcation and Chaos,* **2**(4), 881–887.

[6] Pyragas, V. and Pyragas, K. (2006) Delayed feedback control of the Lorenz system: An analytical treatment at a subcritical Hopf bifurcation. *Physical Review E,* **73**.

[7] W. Xie, C. Wen, Z. L. (2000) Impulsive control for the stabilization and synchronization of Lorenz system. *Physics Letter A,* **275**, 67–72.

[8] Keller, J. B. (1966) Periodic oscillations in a model of thermal convection. *J. Fluid Mech.,* **26**, 599–606.

[9] Welander, P. (1967) On the oscillatory instability of a differentially heat fluid loop. *J. Fluid Mech.,* **29**(1), 17–30.

[10] H. F. Creveling, J. F. de Paz, J. Y. B. and Schoenhals, R. J. (1975) Stability characteristics of a single-phase free convection loop. *J. Fluid Mech.,* **67**, 65–84.

[11] Malkus, W. R. V. (1972) Non-periodic convection at high and low Prandtl number. *Memoires Societe Royale des Science de Liege,* **IV**, 125–128.

[12] Tritton, D. J. (1988) Physical Fluid Dynamics, Oxford University Press Inc., New York, .

[13] R. M Murray, Z. L. and Sastry, S. S. (1994) A Mathematical Introduction to Robotic Manipulation, CRC Press LLC, .

[14] L. P. Shilnikov, A. L. Shilnikov, D. V. T. and Chua, L. O. (2008) Methods of qualitative theory in nonlinear dynamics, World Scientific, .

[15] Leonov, G. (2008) Strange attractors and classical stability theory. *Nonlinear Dynamics and System Theory,* **8**(1), 49–96.

[16] Jackson, E. A., Kodogeorgiou, A. (1992) A coupled Lorenz-cell model of Rayleigh-Benard turbulence. *Physics Letters A,* **168**(4),270–275.

[17] Yu, P., Liao, X. X., Xie, S. L., Fu, Y. L. (2009) A constructive proof on the existence of globally exponentially attractive set and positive invariant set of general Lorenz family *Communication in Nonlinear Science and Numerical Simulation,* **14**(7), 2886–2896

[18] Shashkov, M. V, Turaev, D. V. (2000) A proof of Shilnikov's Theorem for C^1-Smooth Dynamical System *American Mathematical Society Translations,* **200**(2), 142–163

[19] Rasenat, S., Busse, F. H., Rehberg, I. (1989) A theoretical and experimental study of double-layer convection *Journal of Fluid Mechanics,* **199**, 519–540

[20] Lina, J. S., Yan, J. J. (2009) Adaptive synchronization for two identical generalized Lorenz chaotic systems via a single controller *Nonlinear Analysis: Real World Applications,* **10**(2), 1151–1159

[21] Sprott, J. C. (2009) Simplifications of the Lorenz Attractor *Nonlinear Dynamics, Psychology, and Life Sciences,* **13**(3), 271–278

Appendix A

MATLAB CODES

In this appendix, MATLAB codes written for numerical simulation are documented. The structure design of MATLAB codes provide an accessbile way to change the ODE system if necessary. The codes are sub-divided into 3 main categories:

1. Main program: include options to change system constants, e.g. r, σ, b and coupling constants, e.g. μ, N. It also presents computational time and plots, and the cut-off time, as well as the transient time analysis. 3 main program files are lorsim1.m (feedback control), lorasim1.m (adaptive control) and bidirsim.m (bidirectionally coupled). The main programs present the highest observable level of the system.

2. Solvers: are used for time discretized simulation. Inputs include cut-off time and initial values; solvers repeatedly call the differential equations, which are built as a function of the current 3D coordinates of the system. This is the key to uncouple the complex equation from the solvers. Lorenz system solvers include ftcontrol (feedback control), a1control (adaptive control) and bidir.m (bidirectionally coupled)

3. ODE functions: are fundamental differential equations of a system, e.g. $\dot{x}, \dot{y}, \dot{z}$. ODE file names are lor**.m, where the first * is any number, and the second * is a letter.

A.1 Main programs

A.1.1 lorsim1.m

```
% This program simulates feedback and tracking control for single loop
% Variables:
% sigma, r, b -- Prandtl number, Rayleigh number, geometric configuration coefficient
% M-File called:
% notif -- used for lorsim1. To distinguish with anotif and lorasim1.m
% where 'a' denotes adaptive control
close all; clear all; home;
global sigma b r h xeq yeq zeq % system constant
global k % controller & coupling
% Debug note: the 1st line in Plot section, fix T+h or T. +h Only if h is divisible by T1 and T2
%% Define global value
sigma=10;
b=1;
r=100;
% interesting values:
rc = sigma*(sigma+b+3)/(sigma-b-1);
kG = (r-1)/4;
kS = b*(r+sigma)*(r*sigma-b-sigma)/( (r-1)*(b+sigma)^2 );
kL = 1/2*sqrt( (sigma+b+1+b*r/(sigma+b))^2+4*b*(sigma-b-1)/(sigma+b)*(r-rc))-1/2*(sigma+b+1*b*r/(sigma+b));
%% k value
k = 10;
%% Equilbirum computation
c1=sqrt( b*(r-1)); c2 = (r-1);
xeq = sqrt(b*(r-1)); yeq=xeq; zeq=r-1;

% Auxiliary equilibrium
delta = c1^2-4*b*k;
P1=[ 1/2*(-c1+sqrt(delta)),1/2*(-c1+sqrt(delta)),1/(4*b)*(c1^2+delta-2*c1*sqrt(delta))];
P2=[ 1/2*(-c1-sqrt(delta)),1/2*(-c1-sqrt(delta)),1/(4*b)*(c1^2+delta+2*c1*sqrt(delta))];
%% Define initial condition & step time, time limit, and neccessary variable
X = rand(1,3)*1e3;
% X=[105 35 -75];
% X = P2-5e-2;

x0=X(1); y0 = X(2); z0 = X(3); % initial value
h=0.001; % time step, step size is reciprocal to intial value or r
T=20; % total tracking time
T1=12; % chaotic transient time T1
T2=T-T1; % controlling time T2
CPU_time=cputime;
```

```
%% Simulation and Controller variables
t1=[0:h:T1];
[x1,y1,z1]=lor3d(X,T1);
n=length(x1);
X=[x1(n) y1(n) z1(n)]; % adjust initial condition
t2=[0:h:T2];
[x2,y2,z2]=ftcontrol(X,T2);
%% Plot
% figure; plot3(x,y,z); grid on;
t=[0:h:T+h]; x=[x1 x2]; y=[y1 y2]; z=[z1 z2]; n=length(t);

figure; scl=1; % feedback control use only
title('Closed feedback control Lorenz system.');
    % ymin = mean(x) - sqrt(r)/2; ymax = mean(x) + sqrt(r)/2; % plot rescale
    subplot(3,1,1); plot(t,x); grid on; title('x-axis'); hold on; plot(t,xeq*ones(1,n),'r--');
    axis([t(1) t(n) xeq-r/scl xeq+r/scl]);
    xlabel('Time (s)'); ylabel('Amplitude');
    subplot(3,1,2); plot(t,y); grid on; title('y-axis'); hold on; plot(t,yeq*ones(1,n),'r--');
    axis([t(1) t(n) yeq-r/scl yeq+r/scl]);
    xlabel('Time (s)'); ylabel('Amplitude');
    subplot(3,1,3); plot(t,z); grid on; title('z-axis'); hold on; plot(t,zeq*ones(1,n),'r--');
    axis([t(1) t(n) zeq-r/scl zeq+r/scl]);
    xlabel('Time (s)'); ylabel('Amplitude');

figure; % performance of controller
    title('Deviation from equilibrium points C^± of x, y, and z','interpreter','latex');
    t=[0:h:T2];
    subplot(3,1,1), plot(t,(x2-xeq)); grid on; xlabel('x-axis'); ylabel('Amplitude');
    title('Deviation from equilibrium points C^± of x, y, and z','interpreter','latex');

    subplot(3,1,2), plot(t,(y2-yeq)); grid on; xlabel('y-axis'); ylabel('Amplitude');
    subplot(3,1,3), plot(t,(z2-zeq)); grid on; xlabel('z-axis'); ylabel('Amplitude');

% figure; % scl=0.1; % tracking control use only
% % ymin = mean(x) - sqrt(r)/2; ymax = mean(x) + sqrt(r)/2; % plot rescale
% subplot(3,1,1); plot(t,x); grid on; title('x'); hold on; plot(t,xref*ones(1,n),'r--');
%  axis([t(1) t(n) xref-r/scl xref+r/scl]);
% subplot(3,1,2); plot(t,y); grid on; title('y'); hold on; plot(t,yref*ones(1,n),'r--');
%  axis([t(1) t(n) yref-r/scl yref+r/scl]);
```

```
% subplot(3,1,3); plot(t,z); grid on; title('z'); hold on; plot(t,zref*ones(1,n),'r--');
% axis([t(1) t(n) zref-r/scl zref+r/scl]);
%% post-processing
notif;
jacobian_stable;
```

A.1.2 lorasim.m

```
% This program simulates adaptive control for single loop
close all; clear all; home;
global sigma b r h xeq yeq zeq % system constant
global gamma
% Note: the 1st line in Plot section, fix T+h or T. +h Only if h is divisible by T1 and T2
% Function used: a1control lor4k lor2a lorasim1
%% Define global value
sigma=10;
b=1;
r=100;
% interesting values:
rc = sigma*(sigma+b+3)/(sigma-b-1);
kG = (r-1)/4;
kS = b*(r+sigma)*(r*sigma-b-sigma)/( (r-1)*(b+sigma)^2 );
kL = 1/2*sqrt( (sigma+b+1+b*r/(sigma+b))^2+4*b*(sigma-b-1)/(sigma+b)*(r-rc))-1/2*(sigma+b+1*b*r/(sigma+b));
%% Define initial condition & step time, time limit, and neccessary variable
% X=[0 0 0];
X = rand(1,3)*1e3;
% X=1e2* [-0.074350038733783 0.106052075880238 1.125440830802433 0];

x0=X(1); y0 = X(2); z0 = X(3); % initial value
h=0.001; % time step, step size is reciprocal to intial value or r
T=10; % total tracking time
T1=5; % chaotic transient time t1, can't be 0
T2=T-T1; % controlling time t2
CPU_time = cputime;
%% Equilbirum
xeq = sqrt(b*(r-1)); yeq=xeq; zeq=r-1;

%% Simulation and Controller variables
```

```
t1=[0:h:T1];
[x1,y1,z1]=lor3d(X,T1);

gamma=1e-2;
n1=length(x1); k=0; kL=gamma*(r-1)/(2*(r+1));
% k=gamma*(y1(n1)-yeq)^2;
X=[x1(n1) y1(n1) z1(n1) k]; % adjust initial condition
t2=[0:h:T2];
[x2,y2,z2,k2]=a1control(X,T2);

%% Plot
% figure; plot3(x,y,z); grid on;
t=[0:h:T+h]; x=[x1 x2]; y=[y1 y2]; z=[z1 z2]; n=length(t);
k1 = 0*ones(1,n1); k=[k1 k2];
figure; scl=1; % feedback control use only
    % ymin = mean(x) - sqrt(r)/2; ymax = mean(x) + sqrt(r)/2; % plot rescale

    subplot(3,1,1); plot(t,x); grid on; hold on;
    subplot(3,1,1); plot(t,y,'r'); title('x = x(t) and y = y(t)');
    hold on; plot(t,yeq*ones(1,n),'g-.'); axis([t(1) t(n) yeq-r/scl yeq+r/scl]); legend('x','y');
    xlabel('Time(s)'); ylabel('Amplitude');

    subplot(3,1,2); plot(t,z); grid on; title('z = z(t)');
    hold on; plot(t,zeq*ones(1,n),'g-.'); axis([t(1) t(n) zeq-r/scl zeq+r/scl]);
    xlabel('Time(s)'); ylabel('Amplitude');

    subplot(3,1,3); plot(t,k); grid on; title('k = k(t)');
    hold on; plot(t, kL*ones(1,n),'g-.');
    xlabel('Time(s)'); ylabel('Amplitude');

figure; title('Deviation from equilibrium points C^{\pm} of x, y, and z','interpreter','latex');
    t=[0:h:T2];
    subplot(3,1,1), plot(t,(x2-xeq)); grid on; xlabel('x-axis'); ylabel('Amplitude');
    title('Deviation from equilibrium points C^{\pm} of x, y, and z','interpreter','latex');

    subplot(3,1,2), plot(t,(y2-yeq)); grid on; xlabel('y-axis'); ylabel('Amplitude');
    subplot(3,1,3), plot(t,(z2-zeq)); grid on; xlabel('z-axis'); ylabel('Amplitude');

% figure;
% plot(t,k.*(y-yeq));
```

```
% figure; % performance of controller
% t=[0:h:T2];
% subplot(2,1,1), plot(t,k*(y2-yeq)); title('z eqn controller'); hold on; grid on;
% subplot(2,1,2), plot(t,k*(z2-zeq)); title('y eqn controller'); hold on; grid on;

% figure; % scl=0.1; % tracking control use only
% % ymin = mean(x) - sqrt(r)/2; ymax = mean(x) + sqrt(r)/2; % plot rescale
% subplot(3,1,1); plot(t,x); grid on; title('x'); hold on; plot(t,xref*ones(1,n),'r--');
% axis([t(1) t(n) xref-r/scl xref+r/scl]);
% subplot(3,1,2); plot(t,y); grid on; title('y'); hold on; plot(t,yref*ones(1,n),'r--');
% axis([t(1) t(n) yref-r/scl yref+r/scl]);
% subplot(3,1,3); plot(t,z); grid on; title('z'); hold on; plot(t,zref*ones(1,n),'r--');
% axis([t(1) t(n) zref-r/scl zref+r/scl]);
%% notification
anotif;
```

A.1.3 bidirsim.m

```
% This program simulates the classic and controlled Lorenz system in a
% single loop
close all;
clear all;
home;
global sigma b r h mu N

%% Define global value
% sigma -- Prandtl
% r -- Rayleigh
% b -- geometric configuration coefficient
% N -- Nusselt number
sigma=10; b=1; r=100;
mu=sigma*r^2; % mu = sigma*r
N=r;
%% Synchronization check
s1=2*b*r^2/(r*mu/sigma+1);
s2=2*b*r^2/(N+1);
s3=r*(1+sqrt(2));
```

```
check1=(r+2*sigma*r/sigma)*(2*N+1)-(r+s3/2)^2;
check2=b*( (r+2*mu*r/sigma)*(2*N+1)-(r+s3/2)^2)-b*(r+s3/2)^2-(2*N+1)*(s2^2/4);
fprintf('det-2 determinant: %10.5f\n',(r+2*sigma*r/sigma)*(2*N+1)-(r+s3/2)^2);
fprintf('det-3 determinant: %10.5f\n',b*( (r+2*mu*r/sigma)*(2*N+1)-(r+s3/2)^2)-b*(r+s3/2)^2-(2*N+1)*(s2^2/4));
if r>=sigma*(sigma+3+b)/(sigma-1-b) fprintf('\nThe system is unstable.\n');
    else fprintf('\nThe system is stable.\n');
end
if check1 < 0 || check2 < 0
    fprintf('Synchronization: failed.\n')
else if check1 > 0 && check2 > 0
        fprintf('Synchronization: passed.\n');
    end
end
%% Define initial condition & step time, time limit, and neccessary variable
% X=[ 25 -20; -15 10; 0 7]; % initial value
X=rand(3,2)*1e3;
[m n] = size(X);
h=0.00001; % time step
T=25; % tracking time
CPU_time=cputime;

%% Simulation and Controller variables
t=[0:h:T];
% [x,y,z]=lor3d(x0,y0,z0,T);
% -- lor_k2.m --
% k=1; % gain -- lor_k2.m
% [x,y,z]=clork2(x0,y0,z0,T,k)

% -- lor_til.m --
% k=0; % ratio of supported heat to heat source -- lortil.m
% SH=k*r; % supported heat, compared to Rayleigh number (sigma) -- lortil.m
% [x,y,z]=clortil(x0,y0,z0,T,SH);

% -- lorcpl2.m --
[x,y,z]=bidir(X,T);
%% Plot

% figure; plot3(x,y,z); grid on;
figure; scale=r;
    for i=1:n
```

```
        subplot(3,n, i ); plot(t,x(i,:)); grid on; title('x'); axis([0 T -scale +scale]);
        subplot(3,n, n+i ); plot(t,y(i,:)); grid on; title('y'); axis([0 T -scale +scale]);
        subplot(3,n,2*n+i); plot(t,z(i,:)); grid on; title('z'); axis([0 T z(i,T/h+1)-scale z(i,T/h+1)+scale]);
    end
    % retitle
    subplot(3,2,1), title('x_{1}','FontSize',20);
    subplot(3,2,2), title('x_{2}','FontSize',20);
    subplot(3,2,3), title('y_{1}','FontSize',20);
    subplot(3,2,4), title('y_{2}','FontSize',20);
    subplot(3,2,5), title('z_{1}','FontSize',20);
    subplot(3,2,6), title('z_{2}','FontSize',20);
figure; % controller performance
    title('Deviation from equilibrium points C^{\pm} of x, y, and z','interpreter','latex');
    t=[0:h:T];
    subplot(3,1,1), plot(t,(x(1,:)+x(2,:))); grid on; title('e_{x}','FontSize',20);
    xlabel('x-axis'); ylabel('Amplitude'); % x1+x2
    subplot(3,1,2), plot(t,(y(1,:)+y(2,:))); grid on; title('e_{y}','FontSize',20);
    xlabel('y-axis'); ylabel('Amplitude'); % y1+y2
    subplot(3,1,3), plot(t,(z(1,:)-z(2,:))); grid on; title('e_{z}','FontSize',20);
    xlabel('z-axis'); ylabel('Amplitude'); % z1-z2
%% Post-processing
fprintf('Total simulated time: %10.0f seconds\n', T);
fprintf('Computational time: %10.5f seconds\n', cputime-CPU_time);
fprintf('System constants:\n r = %10.0f\n sigma = %10.0f\n b = %10.0f\n', r,sigma, b );
fprintf('Coupling constants:\n mu = %10.0f\n N = %10.0f\n', mu, N);
```

A.2 Solvers

A.2.1 ftcontrol.m

```
function [x,y,z]=ftcontrol(X,T)
% This program simulate feedback and tracking control
% T -- time limit
% h -- step time

global h
% Define initial conditions
```

```
x(1)=X(1);y(1)=X(2); z(1)=X(3);
x0=X(1);y0=X(2);z0=X(3);
k1x(1)=x0; k2x(1)=x0; k3x(1)=x0; k4x(1)=x0;
k1y(1)=y0; k2y(1)=y0; k3y(1)=y0; k4y(1)=y0;
k1z(1)=z0; k2z(1)=z0; k3z(1)=z0; k4z(1)=z0;
for i=1:1:T/h
    k1x(i) = h * lor1( x(i), y(i), z(i) );
    k1y(i) = h * lor2t( x(i), y(i), z(i) );
    k1z(i) = h * lor3t( x(i), y(i), z(i) );

    k2x(i) = h * lor1( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );
    k2y(i) = h * lor2t( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );
    k2z(i) = h * lor3t( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );

    k3x(i) = h * lor1( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );
    k3y(i) = h * lor2t( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );
    k3z(i) = h * lor3t( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );

    k4x(i) = h * lor1( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );
    k4y(i) = h * lor2t( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );
    k4z(i) = h * lor3t( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );

    x(i+1) = x(i) + 1/6*( k1x(i) + 2*k2x(i) + 2*k3x(i) + k4x(i) );
    y(i+1) = y(i) + 1/6*( k1y(i) + 2*k2y(i) + 2*k3y(i) + k4y(i) );
    z(i+1) = z(i) + 1/6*( k1z(i) + 2*k2z(i) + 2*k3z(i) + k4z(i) );
end
end
```

A.2.2 a1control.m

```
function [x,y,z,k]=a1control(X,T)
% This program simulate feedback and tracking control
% T -- time limit
% h -- step time

global sigma b r h k xeq yeq zeq
% Define initial conditions
x(1)=X(1);y(1)=X(2); z(1)=X(3); k(1)=X(4);
```

```
x0=X(1);y0=X(2);z0=X(3); k=X(4);
k1x(1)=x0; k2x(1)=x0; k3x(1)=x0; k4x(1)=x0;
k1y(1)=y0; k2y(1)=y0; k3y(1)=y0; k4y(1)=y0;
k1z(1)=z0; k2z(1)=z0; k3z(1)=z0; k4z(1)=z0;
k1k(1)=k ; k2k(1)=k ; k3k(1)=k ; k4k(1)=k ;
for i=1:1:T/h
    k1x(i) = h * lor1( x(i), y(i), z(i) );
    k1y(i) = h * lor2a( x(i), y(i), z(i), k(i) );
    k1z(i) = h * lor3( x(i), y(i), z(i) );
    k1k(i) = h * lor4k( x(i), y(i), z(i) );

    k2x(i) = h * lor1( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );
    k2y(i) = h * lor2a( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2, k(i)+k1k(i)/2);
    k2z(i) = h * lor3( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );
    k2k(i) = h * lor4k( x(i)+k1x(i)/2, y(i)+k1y(i)/2, z(i)+k1z(i)/2 );

    k3x(i) = h * lor1( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );
    k3y(i) = h * lor2a( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2, k(i)+k2k(i)/2);
    k3z(i) = h * lor3( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );
    k3k(i) = h * lor4k( x(i)+k2x(i)/2, y(i)+k2y(i)/2, z(i)+k2z(i)/2 );

    k4x(i) = h * lor1( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );
    k4y(i) = h * lor2a( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i), k(i)+k3k(i));
    k4z(i) = h * lor3( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );
    k4k(i) = h * lor4k( x(i)+k3x(i), y(i)+k3y(i), z(i)+k3z(i) );

    x(i+1) = x(i) + 1/6*( k1x(i) + 2*k2x(i) + 2*k3x(i) + k4x(i) );
    y(i+1) = y(i) + 1/6*( k1y(i) + 2*k2y(i) + 2*k3y(i) + k4y(i) );
    z(i+1) = z(i) + 1/6*( k1z(i) + 2*k2z(i) + 2*k3z(i) + k4z(i) );
    k(i+1) = k(i) + 1/6*( k1k(i) + 2*k2k(i) + 2*k3k(i) + k4k(i) );
end
end
```

A.2.3 bidir.m

```
% This function simulates the bidirectionally coupled loop
% Uncontrolled
function [x,y,z]=bidir(X,T)
```

```
% X -- initial condition
% T -- time limit
% h -- step time

[m,n]=size(X);
if (m==3)==0
    return;
end
global h
% Define initial conditions
for i=1:n
x(i,1) = X(1,i); y(i,1) = X(2,i); z(i,1) = X(3,i);
k1x(i,1)=X(1,i); k2x(i,1)=X(1,i); k3x(i,1)=X(1,i); k4x(i,1)=X(1,i);
k1y(i,1)=X(2,i); k2y(i,1)=X(2,i); k3y(i,1)=X(2,i); k4y(i,1)=X(2,i);
k1z(i,1)=X(3,i); k2z(i,1)=X(3,i); k3z(i,1)=X(3,i); k4z(i,1)=X(3,i);
end
% Simulate time sequence
clear i j
for j=1:T/h
% k1x(1,j) = h * lor1( x(1,j), y(1,j), z(1,j) );
    k1x(1,j) = h * lor1c( x(1,j), y(1,j), z(1,j), x(2,j) );
    k1y(1,j) = h * lor2c( x(1,j), y(1,j), z(1,j), y(2,j) );
    k1z(1,j) = h * lor3( x(1,j), y(1,j), z(1,j) );

% k1x(2,j) = h * lor1( x(2,j), y(2,j), z(2,j) );
    k1x(2,j) = h * lor1c( x(2,j), y(2,j), z(2,j), x(1,j) );
    k1y(2,j) = h * lor2c( x(2,j), y(2,j), z(2,j), y(1,j) );
    k1z(2,j) = h * lor3( x(2,j), y(2,j), z(2,j) );

% k2x(1,j) = h * lor1( x(1,j)+k1x(1,j)/2, y(1,j)+k1y(1,j)/2, z(1,j)+k1z(1,j)/2 );
    k2x(1,j) = h * lor1c( x(1,j)+k1x(1,j)/2, y(1,j)+k1y(1,j)/2, z(1,j)+k1z(1,j)/2, x(2,j)+ k1x(2,j)/2 );
    k2y(1,j) = h * lor2c( x(1,j)+k1x(1,j)/2, y(1,j)+k1y(1,j)/2, z(1,j)+k1z(1,j)/2, y(2,j)+ k1y(2,j)/2 );
    k2z(1,j) = h * lor3( x(1,j)+k1x(1,j)/2, y(1,j)+k1y(1,j)/2, z(1,j)+k1z(1,j)/2 );

% k2x(2,j) = h * lor1( x(2,j)+k1x(2,j)/2, y(2,j)+k1y(2,j)/2, z(2,j)+k1z(2,j)/2 );
    k2x(2,j) = h * lor1c( x(2,j)+k1x(2,j)/2, y(2,j)+k1y(2,j)/2, z(2,j)+k1z(2,j)/2, x(1,j)+ k1x(1,j)/2 );
    k2y(2,j) = h * lor2c( x(2,j)+k1x(2,j)/2, y(2,j)+k1y(2,j)/2, z(2,j)+k1z(2,j)/2, y(1,j)+ k1y(1,j)/2 );
    k2z(2,j) = h * lor3( x(2,j)+k1x(2,j)/2, y(2,j)+k1y(2,j)/2, z(2,j)+k1z(2,j)/2 );

% k3x(1,j) = h * lor1( x(1,j)+k2x(1,j)/2, y(1,j)+k2y(1,j)/2, z(1,j)+k2z(1,j)/2 );
```

```
    k3x(1,j) = h * lor1c( x(1,j)+k2x(1,j)/2, y(1,j)+k2y(1,j)/2, z(1,j)+k2z(1,j)/2, x(2,j)+ k2x(2,j)/2 );
    k3y(1,j) = h * lor2c( x(1,j)+k2x(1,j)/2, y(1,j)+k2y(1,j)/2, z(1,j)+k2z(1,j)/2, y(2,j)+ k2y(2,j)/2 );
    k3z(1,j) = h * lor3( x(1,j)+k2x(1,j)/2, y(1,j)+k2y(1,j)/2, z(1,j)+k2z(1,j)/2 );

% k3x(2,j) = h * lor1( x(2,j)+k2x(2,j)/2, y(2,j)+k2y(2,j)/2, z(2,j)+k2z(2,j)/2 );
    k3x(2,j) = h * lor1c( x(2,j)+k2x(2,j)/2, y(2,j)+k2y(2,j)/2, z(2,j)+k2z(2,j)/2, x(1,j)+ k2y(1,j)/2 );
    k3y(2,j) = h * lor2c( x(2,j)+k2x(2,j)/2, y(2,j)+k2y(2,j)/2, z(2,j)+k2z(2,j)/2, y(1,j)+ k2y(1,j)/2 );
    k3z(2,j) = h * lor3( x(2,j)+k2x(2,j)/2, y(2,j)+k2y(2,j)/2, z(2,j)+k2z(2,j)/2 );

% k4x(1,j) = h * lor1( x(1,j)+k3x(1,j), y(1,j)+k3y(1,j), z(1,j)+k3z(1,j) );
    k4x(1,j) = h * lor1c( x(1,j)+k3x(1,j), y(1,j)+k3y(1,j), z(1,j)+k3z(1,j), x(2,j)+ k3x(2,j)/2 );
    k4y(1,j) = h * lor2c( x(1,j)+k3x(1,j), y(1,j)+k3y(1,j), z(1,j)+k3z(1,j), y(2,j)+ k3y(2,j)/2 );
    k4z(1,j) = h * lor3( x(1,j)+k3x(1,j), y(1,j)+k3y(1,j), z(1,j)+k3z(1,j) );

% k4x(2,j) = h * lor1( x(2,j)+k3x(2,j), y(2,j)+k3y(2,j), z(2,j)+k3z(2,j) );
    k4x(2,j) = h * lor1c( x(2,j)+k3x(2,j), y(2,j)+k3y(2,j), z(2,j)+k3z(2,j), x(1,j)+ k3x(1,j) );
    k4y(2,j) = h * lor2c( x(2,j)+k3x(2,j), y(2,j)+k3y(2,j), z(2,j)+k3z(2,j), y(1,j)+ k3y(1,j) );
    k4z(2,j) = h * lor3( x(2,j)+k3x(2,j), y(2,j)+k3y(2,j), z(2,j)+k3z(2,j) );

    for i=1:n
    x(i,j+1) = ( x(i,j) + (k1x(i,j) + 2*k2x(i,j) + 2*k3x(i,j) + k4x(i,j))/6 );
    y(i,j+1) = ( y(i,j) + (k1y(i,j) + 2*k2y(i,j) + 2*k3y(i,j) + k4y(i,j))/6 );
    z(i,j+1) = ( z(i,j) + (k1z(i,j) + 2*k2z(i,j) + 2*k3z(i,j) + k4z(i,j))/6 );
    end
end
```

A.3 ODE functions

A.3.1 lor1.m

```
% This function is the first differential equation from Lorenz system
% Uncoupled
% sigma -- Prandtl number
function xdev = lor1(x,y,z)
global sigma
xdev = sigma * (-x+y);
```

A.3.2 lor2.m

```
% This function is the second differential equation from Lorenz system
% Uncoupled
% r -- Rayleigh number
function ydev = lor2(x,y,z)
global r
ydev = r*x-y-x*z;
```

A.3.3 lor3.m

```
% This function is the third differential equation from Lorenz system
% Uncoupled
% b -- geometric configuration coefficient
function zdev = lor3(x,y,z)
global b
zdev = x*y-b*z;
```