Image Representation Through Patterns of Synchronisation and Generalised Turing Patterns in Networks of Dynamical Systems

A thesis submitted towards partial fulfilment of
BS-MS Dual Degree Programme

by

YAGYIK GOSWAMI

under the guidance of

Prof. GOVINDAN RANGARAJAN

INDIAN INSTITUTE OF SCIENCE-BANGALORE

Indian Institute of Science Education and Research
Pune

Certificate

This is to certify that this thesis entitled "Image Representation Through Patterns of Synchronisation and Generalised Turing Patterns in Networks of Dynamical Systems" submitted towards the partial fulfilment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research Pune represents original research carried out by Yagyik Goswami at Indian Institute of Science-Bangalore, under the supervision of Prof. Govindan Rangarajan during the academic year 2014-2015.

Student
YAGYIK GOSWAMI

G. Rangarajan

Supervisor
PROF. GOVINDAN
RANGARAJAN

PROF. G. RANGARAJAN
DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF SCIENCE
BANGALORE - 560 012

Declaration

I hereby declare that the matter embodied in the report entitled "Image Representation Through Patterns of Synchronisation and Generalised Turing Patterns in Networks of Dynamical Systems" are the results of the investigations carried out by me at the Department of Mathematics, Indian Institute of Science-Bangalore, under the supervision of Prof. Govindan Rangarajan and the same has not been submitted elsewhere for any other degree.

Student
YAGYIK GOSWAMI

G. Rangarajan

Supervisor
PROF. GOVINDAN
RANGARAJAN

PROF. G. RANGARAJAN
DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF SCIENCE
BANGALORE - 560 012

Acknowledgements

I would like to thank Prof. Govindan Rangarajan for guidance and encouragement and for giving me a clear direction in this project. I would like to thank Prof. M. S. Santhanam for his kind advice throughout the duration of the project. I would especially like to thank Indian Institute of Science Education and Research, Pune for enabling me to pursue my academic goals and Indian Institute of Science Bangalore for hosting me and giving me access to their facilities. I would like to thank Meghna Manae for valuable input in putting together this report. Lastly and most importantly, I am grateful to my parents for all their support.

Abstract

The phenomenon of synchronisation in coupled dynamical systems, and in particular, the conditions under which such coupled systems achieve synchrony is a topic of current interest and potential applications. The main objective of the present project has been to explore a technique by which an image can be represented with a network of coupled dynamical systems in terms of the coupling matrix of the network. The central concept is the use of the Lyapunov exponent in determining the bounds within which a network of dynamical systems will show globally synchronous behaviour. The concept of a Lyapunov spectrum has been used to determine the conditions for which perturbations around the synchronous trajectory damp over time, implying a stable globally synchronous state. For coupling schemes outside these bounds one observes specific patterns, termed Generalised Turing Patterns; this work presents application of this method to represent a given image with a network of coupled dynamical systems. The effectiveness of the method is shown through a few test cases. The method can be an effective tool in image reconstruction/retrieval; certain directions of improvement are outlined.

Contents

Chapter 1

Introduction

Broadly, the focus of this project is the phenomenon of synchronisation in coupled dynamical systems. Specifically, the conditions under which such coupled systems achieve synchrony is a topic which has been extensively covered in recent years in notable works such as [17, 19] among many others. The goal of this project is to study the phenomenon of synchronisation built around existing literature and to explore the behaviour of these systems outside this regime as well; conditions that lead to the formation of Generalised Turing Patterns, as studied in [5, 24]. This project builds on the work [5], which describes how a network of oscillators can form Generalised Turing Patterns, patterns that emerge for specific coupling regimes for which global synchrony is not possible. Non-linear differential equations first caught general interest when their utility in describing models of population growth and decay was first brought to light. The Logistic equation [12] as a model of population growth and decay captured a variety of behaviour that was commonly observed in time-varying populations such as oscillations that switch between a few finite values, steady populations, oscillations that damp over time to give steady values, wildly fluctuating populations and extinction.

Since then, non-linear dynamical systems, both discrete-time and continuous have been used as models to describe a variety of natural phenomena, such as the Lorenz equations as an atmospheric model[9] and the Rössler equations[20] to model chemical reactions.

Notably, the behaviour of such dynamical systems when they are coupled to each other has also been a topic of interest. Here, coupling implies that the time-evolution of one such system from t to $t + \delta t$ is dependent on the value of another system at time t or at $t - T$ through some function. A prominent example of this is the case of theoretical neuroscience. A number of non-linear models of action potentials in neurons currently exist in literature, The Hodgkin-Huxley[6], Fitzhugh-Nagumo[18] and Morris-Lecar[15] being examples of some, and their behaviour under various types of coupling has been extensively studied in recent years.

Synchronisation in coupled non-linear systems has held the interest of researchers for decades now. In these systems, one has N m-dimensional dynamical systems whose state update or derivative functions are modified so that the state vector of each is dependent on either the state vector or some subset of the components of the state vector of at least one other such system. Interestingly, such synchronisation is possible even when the individual dynamical systems are executing chaotic trajectories[16]. Synchronisation of many forms occurs; the most commonly studied is the phenomenon of global synchronisation which this project focuses on. In a globally synchronised network of oscillators, as time progresses, each of the N oscillators in the network begin to execute identical trajectories, in spite of differences in the initial conditions and even the parameters of the individual members/nodes of the network. Other forms of synchronisation include

1. Partial synchronisation, where only a fraction of the nodes synchronise [21]

2. Delay synchronisation, where some subset of the nodes is consistently behind another subset of the nodes, i.e., $x(t) = x'(t + T)$ for all t>0. This is typically seen when a delay is incorporated into the function that specifies how a given node is coupled to another. [11]

3. Cluster synchronisation, where different groups or clusters of nodes synchronise amongst themselves[1, 10], and

4. Phase synchronisation, where two subsets of nodes may be in phase in their trajectories but may not have the same amplitudes.[14]

In 1998, Pecora and Carrol put forward a method by which the stability of globally synchronised networks of dynamical systems could be evaluated [17]. This method was based on linearising the state vector of each node about the synchronisation manifold, i.e., the trajectory the synchronised system follows, and calculating the Lyapunov spectrum of this system. They showed in their work the dependence of the Lyapunov exponents on the eigenvalues of the coupling matrix by which one obtained bounds for stability for the eigenvalues of the connectivity matrix.

The Lyapunov exponent or, as in the case of a many-dimensional dynamical system, the Lyapunov spectrum can be used to determine how small perturbations around a given trajectory evolve over time [3]. In this context, that concept has been used to determine the conditions for which perturbations around the synchronous trajectory damp over time, implying a stable globally synchronous state [17]. For coupling schemes outside these bounds one observes specific patterns, termed Generalised Turing Patterns [5]. In the work [5, 24], the authors explored the behaviour of such coupled systems outside the regime of stable global synchronisation. The condition for stability of the synchronised state is best described as being the case where all perturbations transverse to the synchronised manifold damp down over time such that all trajectories converge on the synchronisation manifold.

In their work, the authors described how specific coupling or connectivity matrices satisfy the conditions of having one eigenvalue outside the regime of synchronisation so that the system upon time evolution generates a specific pattern.

They go on to propose a method by which one could theoretically generate a matrix that produces a desired image. The main objective of this project is to further develop and to implement this method.

Chapter 2

Theory

2.1 Dynamical Systems

The properties of dynamical systems is central to the theory on which this project is based. The formation of patterns that are observed under specific cases of coupling is also due to the various limiting behaviour that the individual dynamical systems exhibit under certain parameter ranges.

Typically, a dynamical system can be defined with the following iterative equation [22, 5, 1, 17]:

$$x_{n+1} = f(x_n) \tag{2.1}$$

$$\dot{x}(t) = f(x(t)) \tag{2.2}$$

For the discrete and continuous time cases, respectively. Here, x is an m-dimensional state vector and $f(x)$ is a linear or non-linear function of the state vector.

2.2 Behaviour of Dynamical Systems

For different parameter sets, dynamical systems display a range of behaviour asymptotically [22]. Broadly, the categories are:

1. Fixed points - stable, unstable and neutral A fixed point x^* is defined as a point in the phase space where $x_{n+1} = f(x*) = 0$. A stable fixed point is one such that when the map is applied to a point x' in the vicinity of x^*, one gets x^* asymptotically, that is,

$$\lim_{x \to \infty} f^n(x' - x^*) = 0$$

as an example for the discrete case. This vicinity is known commonly as the basin of attraction. For an unstable fixed point, this limit goes to either infinity or negative infinity. For a neutral fixed point, the RHS is a non-zero constant.

2. Periodic orbits - stable or unstable - a periodic orbit is described by the following constraint, $x(t) = x(T + t)$ for the continuous case and $x(n) = x(n + m)$ where T and m are the periods for the orbit.

3. Chaotic dynamics - deterministic trajectories with high sensitivity to initial conditions and to perturbations.

4. Chaotic attractors - In this case, points in the basin of attraction that are arbitrarily close become arbitrarily far apart over evolution by the map. However, they always remain in the basin of attraction.

2.2.1 The Dynamical Systems considered in this study

1. The one-dimensional discrete Logistic Map [12] - the equation for the Logistic map used here was

$$x(n+1) = f(x(n)) = 1 - a[x(n)]^2 \tag{2.3}$$

   As the parameter a is varied from low to high ($0 < a < \infty$) the system's asymptotic behaviour transitions from a single fixed point to an m-point limit cycle where m is a natural number greater than 1 and finally to chaotic dynamics for $a > 1.5$. This transition occurs in a number of steps, termed bifurcations.

2. The three-dimensional continuous Rössler system of differential equations [20] - the Rössler system is described by the following differential equations

$$\begin{aligned} \frac{dx}{dt} &= -y - z \\ \frac{dy}{dt} &= x + ay \\ \frac{dz}{dt} &= b + z(x - c) \end{aligned} \tag{2.4}$$

   The Rössler system is well-known for producing this strange attractor. Shown also in 3 dimensions below in Figure 2.1.

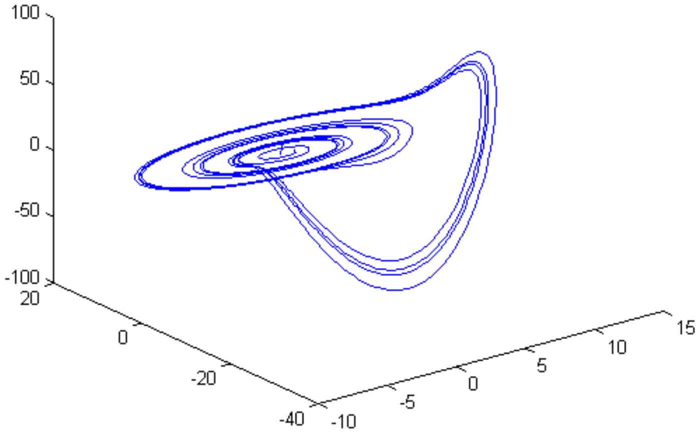


Figure 2.1: The Rössler Attractor in 3D

The plots above were produced using the parameter set $a = 0.2, b = 0.2, c = 5.7$.

2.3 Coupled Dynamical Systems

In a system of coupled oscillators, the RHS of the dynamical equation, in each case, will be modified to incorporate the effects of the other oscillators. In general, such a system is described as [5]:

$$\dot{x}^i(t) = \mathbf{f}(x^i(t)) + \sum_j G_{ij}(x^j(t)) \tag{2.5}$$

Likewise, for the discrete time case, the coupled dynamical system takes the form

$$x^i(n+1) = \mathbf{f}(x^i(n)) + \frac{1}{N}\sum_{j=1}^{N} G_{ij}\mathbf{H}(x^j(n)) \tag{2.6}$$

In these cases, $H(x)$ is the coupling function that describes the dependence of $x_i(n+1)$ on $x_j(n)$. For the purposes of this study, $H : R_M \rightarrow R_M$. G is an $N \times N$ coupling matrix that describes the links from and given node i to its neighbours j. N is the number of such nodes in the network.

2.3.1 Normal Modes

In coupled systems, such as below, one can evaluate the behaviour of the system rather than that of each individual node. The normal modes of the system of oscillators themselves oscillate with independent frequencies [13]. We consider the case of two coupled pendulums as in [13], whose displacements are denoted by x and y. The coupling is implemented by attaching a spring of spring constant k between the two bobs.

$$m\ddot{x} = -mg\frac{x}{l} + k(y-x) \tag{2.7}$$

$$m\ddot{y} = -mg\frac{y}{l} - k(y-x) \tag{2.8}$$

Adding and subtracting the two equations above gives two equations in terms of two new variables, $(x+y)$ and $(x-y)$ where the first describes the position of the centre of mass and the second the distance between them.

$$m(\ddot{x}+\ddot{y}) = -m\frac{g}{l} + (x+y) \tag{2.9}$$

$$m(\ddot{x}-\ddot{y}) = (-m\frac{g}{l} - 2k)(x-y) \tag{2.10}$$

The frequency with which the quantity $x+y$ oscillates is the same as that of the unperturbed individual oscillators. It describes the two pendulums swinging in unison, that is, in synchrony. The concept of synchrony applies as well to many coupled oscillators or dynamical systems.

2.4 Coupling and Synchronisation

In this context, the phenomenon of synchronization can be described as the case where each x^i oscillates in unison with the other oscillators in the system (this is the case of global synchronization). Cases where a fraction of the oscillators in the system synchronise is termed partial synchronization [21]. Synchronisation can occur in a variety of forms; complete synchronization means that all the oscillators oscillate with the same frequency and have the same magnitude at the same time, phase synchronization means that the oscillators are in phase but their amplitudes may differ, delay synchronization is the case where the oscillators possess the same frequency but may have a constant phase difference between them.

Interestingly, global synchronization is seen even in coupled chaotic systems [16], such as a system of coupled Lorenz oscillators, or coupled Logistic Maps. The focus of this work is the different patterns of synchronized behavior that can be seen in coupled chaotic systems. One such case is multi-cluster synchronization [1, 10], where different groups of oscillators are synchronized within themselves but not with other groups. One other case is the emergence of patterns analogous to Turing patterns [4, 23]for parameter values where the globally synchronized state is unstable.

While studying the different patterns of oscillatory behavior, one sees that the nature of the coupling within the system between the different oscillators plays an important role in the global behavior of the system. It helps to visualize the system as a network of oscillators with each oscillator as a separate node and the coupling strength between the different nodes as weighted links. This allows one to explore the effects of various known network geometries, and in some cases, call upon known results in group theory and Linear Algebra on the adjacency matrix that describes the network.

For a system of many oscillators connected on a network, the conditions for global synchronization are expressed in terms of the coupling matrix. For cases where the system exhibits fixed point dynamics, the synchronised state is the one for which all oscillators converge on the same state.

The study of the conditions for synchronisation are detailed below. We consider the case of coupled continuous dynamical systems here, however, the same procedure is applicable to the discrete time case as well.

Proceeding through the steps of the work in [5] similar to the steps in [24] and [17], one considers the generic form of the coupled dynamical system,

$$\dot{x}^i(t) = \mathbf{f}(x^i(t)) + \sum_j G_{ij}(x^j(t)) \tag{2.11}$$

Taking the deviations at each node around the synchronised trajectory, $z_i = x_i - x_{sync}$,one gets the linearised form, where the vector $\mathbf{Z}$ is the $N \times m$ dimensional matrix composed of the vectors of deviations.

$$\dot{Z} = DfZ + DuZG^T \tag{2.12}$$

Here, one proceeds by denoting the left eigenvectors of G as ekL, and multiplying throughout. One gets a new vector $\phi = Z * ekL$. Using the eigenvalue equation on the second term on the right hand side, we replace the matrix G with the eigenvalue γk where the index k is for different such eigenvalues of the coupling matrix. The resultant equations is

$$\phi_k = [Df + \gamma_k Du]\phi_k \tag{2.13}$$

This equation now has a linear matrix form and describes how deviations from the synchronised trajectory behave over time. This linearised equation can be used to compute the Lyapunov exponents for the system. The eigenvalue 0 corresponds to the synchronised case with chaotic dynamics. This is evident from the fact that in this case, the coupled system and the uncoupled system are identical. The remaining eigenvalues are for eigenvectors transverse to the synchronised trajectory. If they are all negative, the system is stable.

For the synchronised manifold, one has the eigenvector of the form $(1, 1, 1, 1, \ldots 1, 1)$. The corresponding eigenvalue is 0. For the eigenvalue 0, one has the N state vectors each propagating according to the equation of the individual uncoupled system, that is, unperturbed by the adjacent oscillators.

One can generalise this to the following case

$$\sum_j G_i j = g, \qquad \forall i \tag{2.14}$$

Here, each individual node receives the same input from its neighbours and one gets a synchronised state. In general, the eigenvector takes the form $(v, v, v, v, \ldots, v, v,)$ and the value of g is one eigenvalue. The dynamical equation simplifies to

$$\dot{x} = f(x) + gu(x) \tag{2.15}$$

For which, $g = 0$ gives us the unperturbed, uncoupled dynamics and the corresponding eigenvector. One can linearise the equation about the synchronised manifold to get the form described above from which stability can be evaluated.

The event plot above (Figure 2.2) describes how a system of 5 coupled Rössler systems approaches global synchronization. The parameters are $a = 0.2, b - 0.2, c = 5.7$. The system is evolved over 20000 time steps of length 0.005 increment.

For different coupling schemes, it is possible to obtain cases of partial or no synchronization as well.

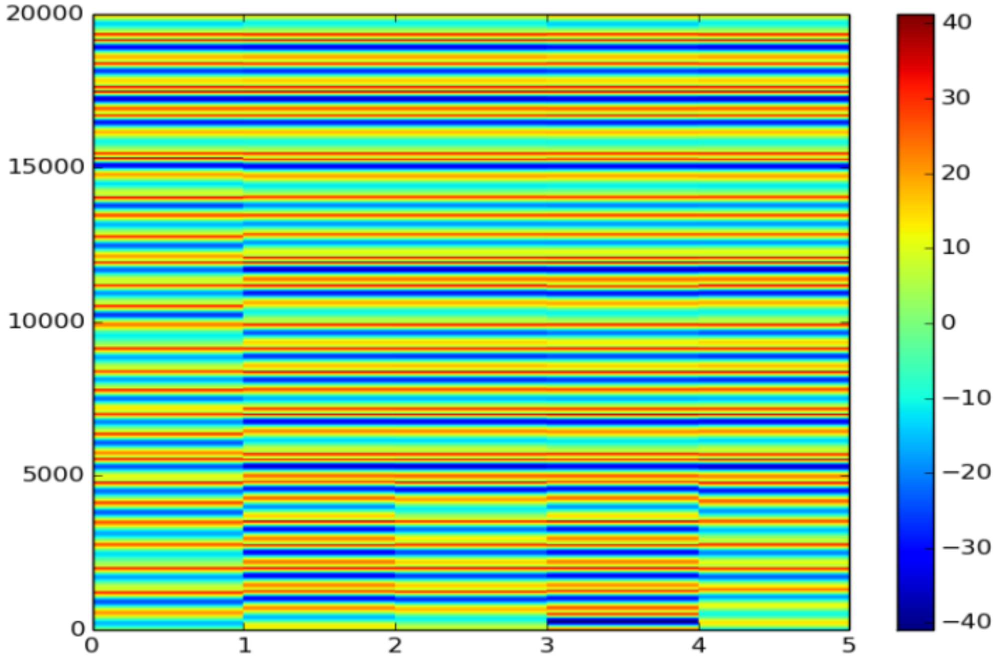


Figure 2.2: Global Synchronisation in a network of coupled Rössler Systems -The parameters are a=0.2, b-0.2, c=5.7. The system is evolved over 20000 time steps of length 0.005 increment. The coupling scheme for the case above is symmetric all-to-all. The eigenvalues of the matrix G are [0,-5,-5,-5,-5]. In this case, the maximal Lyapunov exponent of the uncoupled nodes is 0.075

.

2.5 Lyapunov Exponents and Stability of the Synchronised State

The Lyapunov exponents or Lyapunov spectrum of the system is central to the study of stability and of the chaotic trajectories of dynamical systems.

The Lyapunov exponent measures sensitive dependence to initial conditions. Essentially, the Lyapunov exponent is a measure of the asymptotic behaviour of a small difference in initial conditions over time or over repeated iterations [3]. If the difference grows then the system is sensitive to initial conditions, which is indicative of chaotic dynamics of the map in question. If the difference shrinks to zero asymptotically, the system is describing a either a stable periodic orbit or a stable fixed point.

In the context of coupled systems, the Lyapunov exponent can be used to check for divergent or damped perturbations transverse to the synchronised manifold [17, 1, 5]. The system is linearised about the synchronised manifold and the asymptotic behaviour of perturbation vectors transverse to the synchronised manifold is quantified to evaluate the stability of the synchronised state.

For a system given by $x'(t) = f(x(t))$, the Jacobian matrix J is defined as

$$J^t(x_0) = \left.\frac{df^t(x)}{dx}\right|_{x_0} \tag{2.16}$$

The matrix given by

$$L(x_0) = \lim_{t\to\infty}(J^t Transpose(J^t))^{\frac{1}{2t}}$$

can be decomposed using QR decomposition or an equivalent method to give its eigenvalues. The spectrum of Lyapunov exponents is obtained in terms of these eigenvalues

$$\lambda_i(x_0) = log\Lambda_i(x_0) \tag{2.17}$$

For the discrete case, we get instead the Jacobian of f(x(n)) and the matrix L as

$$\lim_{n\to\infty}[J(x_n, y_n)J(x_n - 1, y_n - 1)\dots J(x_1, y_1)]^{1/n} \tag{2.18}$$

From which the spectrum of Lyapunov exponents is obtained.

For the coupled case, we have obtained the Lyapunov exponent in terms of the linearised equation in equation X . Thus, we get our Lyapunov exponent in terms of the eigenvalues of the coupling matrix G given by λ.

$$\mu_i(\lambda) = h_i + \frac{1}{N}Re(\lambda), \qquad i = 1, 2, \ldots, M \tag{2.19}$$

Rearranging the above equation, and using the constraint that the maximal Lyapunov exponent has to be less than zero, we get the stability condition below as a constraint on the eigenvalues of the coupling matrix. Here, the h_i are the Lyapunov exponents of the uncoupled system.

$$|\lambda + N| < Nexp(-h_{max}), \qquad \lambda \neq 0 \tag{2.20}$$

For the case of nearest neighbour coupling, for which results have been presented here, the eigenvalues of the coupling matrix are given as.

$$\lambda_l = -\frac{2N}{P}\sum_{p=1}^{p} a_p sin^2\frac{\pi p(l-1)}{N}, \qquad l = 0, 1, \ldots, N-1 \tag{2.21}$$

Thus, the constraint on the eigenvalues can be translated into constraints on the a_p's by substituting for λ.
These constraints describe the coupling scheme required to obtain global synchronisation. In addition to global synchronisation, it also possible to obtain partial synchronisation, phase-delayed synchronisation where the phase difference between different components is constant and also multi-cluster synchronisation.
The main focus of this project is Generalised Turing Patterns. These are patterns that emerge out of coupled systems outside the parameter regime of stability.

2.6 Turing Patterns

Turing patterns have been studied for a considerable amount of time. One of the oldest and most well-known examples is the case of the Reaction-Diffusion equation. [4, 23]

$$(\partial_t u \;\; \partial_t v) = \begin{pmatrix} D_u & 0 \\ 0 & D_v \end{pmatrix}\begin{pmatrix} \partial_{xx}u \\ \partial_{xx}v \end{pmatrix} + \begin{pmatrix} F(u,v) \\ G(u,v) \end{pmatrix} \tag{2.22}$$

The equation above is familiar as the heat equation, however, the second term on the RHS describes how the two components are coupled. The solution of the independent heat equation is given as

$$u(x,t) = \sum_{n=1}^{\infty} D_n sin\left(\frac{n\pi x}{L}\right) e^{-\frac{n^2\pi^2 at}{L^2}} \tag{2.23}$$

where

$$D_n = \frac{2}{L}\int_0^L f(x) sin\left(\frac{n\pi x}{L}\right) dx \tag{2.24}$$

Here, f(x) describes the initial state of the system, that is, $u(x,0) = f(x)$. It is assumed that for this case the boundary conditions are $u(0,t) = u(L,t) = 0$ for all $t > 0$.

With appropriate selection of $f(x)$, D_n takes 0 values for all n except $n = 1$. This gives a Gaussian bell-shaped solution to $u(x,t)$. As time progresses, the bell levels out to a constant function. For cases where the function $u(x,t)$ describes aprobability distribution function, the process of levelling out of the Gaussian describes diffusion.

In the case of coupled equations, such as the reaction diffusion equation, for coupling strengths outside the regime of stability one observes steady patterns in the spatial coordinates of the system, rather than the constant, diffused state. The stability of the system is evaluated by using linear stability analysis on the matrix form of the RHS and its eigenvalues.

2.7 Generalised Turing Patterns

The following section is about the study of patterns of oscillations seen in networks of oscillators where the coupling parameters are such that the globally synchronous case is unstable. All the results are generated calling upon code written to caluclate the Lyapunov exponent as mentioned in the previous sections.

In addition, using the linearised forms as derived above in the stability analysis of cluster synchronisation, one can obtain bounds on the eigenvalues of the graph in terms of the maximal Lyapunov exponent of the uncoupled system.

The focus thus far has been on the single dimensional Logistic discrete map and the many-dimensional continuous Rössler map. In these two cases, symmetric nearest neighbour coupling in 1-D ring and 2-D lattice configurations has been implemented.

The matrix H that determines the functional dependence of each oscillator on its neighbours has been set to the identity in the Rössler case and to $f(x)$ in the Logisitic map case.

For the Rössler map with $H(x) = Ix$, the Jacobian of this function is simply the identity itself. Thus, the linearised equation obtained above becomes convenient to evaluate since the identity matrix commutes with any other matrix.

The relation obtained is:

$$\mu_i(\lambda) = h_i + \frac{1}{N} Re(\lambda), \qquad\qquad i = 1, 2, \ldots, M \tag{2.25}$$

For the discrete system, since $H(x) = f(x)$, the relation for the stability bounds on the eigenvalue can be determined from the equation below (put equation reference here)

$$|\lambda + N| < Nexp(-h_{max}), \qquad\qquad \lambda \neq 0 \tag{2.26}$$

For cases of general coupling functions or general coupling matrices, the Lyapunov exponents for the system do not have convenient forms by which they can be expressed as in the one above.

In these cases, the trajectory of the coupled system given some initial conditions needs to be calculated and stored. Thereafter, the linearised system can be calculated at each time step by using the stored values to calculate the Jacobian at different points and follow the rest of the procedure to obtain the Lyapunov exponents.

For the case of the Logistic map, specific patterns were generated by using the known stability bounds of the uncoupled system.

To have an analytical understanding of how different coupling schemes generate different patterns, some specific cases were implemented.

For the Logistic map with symmetric nearest neighbour coupling, one can see that the coupling matrix is shift-invariant. Under this condition, the eigenvectors of the coupling matrix take the following form:

$$\mathbf{e_l} = \left(exp\left(2\pi i \frac{l}{N}\right), exp\left(4\pi i \frac{l}{N}\right), \ldots, exp\left(2N\pi i \frac{l}{N}\right)\right)^T \tag{2.27}$$

Here, l is the index over the N eigenvectors of the $N \times N$ coupling matrix, G.

The eigenvalues can be determined using the eigenvectors and the matrix with the following relation.

$$\lambda_l = \frac{e_l^T G e_l}{e_l^T w_l} \tag{2.28}$$

Finally, substituting the following shift invariant coupling matrix G with symmetric nearest neighbour coupling,

$$\mathbf{G} = \frac{N}{2P} \begin{bmatrix} -2\sum_{p=1}^{P} & a_1 & \ldots & a_p & 0 & \ldots & 0 & a_p & \ldots & a_1 & \\ a_1 & -2\sum_{p=1}^{P} & a_1 & \ldots & a_p & 0 & \ldots & a_p & \ldots & a_2 & \\ \ldots & & & \ldots & & & \ldots & & & & \\ a_1 & \ldots & a_p & 0 & \ldots & 0 & a_p & \ldots & a_1 & -2\sum_{p=1}^{P} \end{bmatrix} \tag{2.29}$$

One obtains the following form of the eigenvalues, which can be substituted to obtain the stability bounds on the coupling strengths themselves in terms of the maximal Lyapunov exponent, rather than the eigenvalues.

$$\lambda_l = -\frac{2N}{P} \sum_{p=1}^{p} a_p sin^2 \frac{\pi p(l-1)}{N}, \qquad l = 0, 1, \ldots, N-1 \tag{2.30}$$

Using the scheme described above, one can set coupling strengths to realise different patterns. One has a set of inequalities in P variables each obtained by setting each eigenvalue within the stability bounds. For a coupling matrix that has one non-zero eigenvalue that is, the completely connected graph, only two patterns can be realised. Likewise, for a system with two distinct non-zero eigenvalues, a total of four patterns can be realised, two of one type and two of another. For the cases shown below, the two types are the short and long-wavelength patterns.

Patterns such as the ones described below are generated by putting one or more eigenvalues outside the stability boundaries so that the

1. the deviations grow with time in the directions corresponding to those eigenvalues

2. the deviations are bounded by the non-linearity of the system after long time so that the desired pattern is obtained.

Chapter 3

Methods

3.1 Calculating the Lyapunov Exponent

The calculation of the Lyapunov exponent for general many-dimensional dynamical systems has been well-documented [2] and the code used here to calculate it has drawn from a number of sources

1. The dynamical system is evolved for a given time length T and stored.
2. Another loop is run in which, at each time step, the Jacobian of the function $J(x_n)$ is calculated using the stored value of x_n
3. Using this Jacobian, an orthonormal basis of perturbations around the state vector is propagated.
4. For an m-dimensional state vector, an orthonormal basis of $n \leq m$ perturbation vectors can be propagated in this manner to give the n largest Lyapunov exponents.
5. The $m \times n$ matrix of propagated perturbations can then be decomposed using QR decomposition to generate another set of orthonormal perturbation vectors, as well as the eigenvalues of the Jacobian update operation.
6. The logarithm of the eigenvalues is summed at each time step. The new orthonormal basis generated is propagated using the Jacobian in the next time step and the logarithm of the resulting eigenvalues also summed.
7. At the end of T time steps, the accumulated sum of logarithms is divided by T to give the m-dimensional vector of Lyapunov exponents. To approximate the $\lim_{x\to\infty}$, a very large T is typically used.

3.2 Algorithm to Propagate the Coupled Dynamical System

The basis of the algorithm is the use of the RK4 numerical integration routine to solve the system of differential equations for the continuous time case and the discrete time increment function for the discrete time case.

The coupling matrix G is initialised with either 1-D or 2-D nearest neighbour or next-nearest neighbour coupling.

Following this, for a fixed number of time-steps, the system is propagated and a snapshot of the final state after these time steps is plotted.

For the continuous time Rössler case, a standard RK4 routine was used

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1) \\ k_3 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2) \\ k_4 &= hf(x_n + h, y_n + k_3) \\ y_{n+1} &= y_n + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4 + O(h^5) \end{aligned} \tag{3.1}$$

Here, $\mathbf{y_n}$ denotes the state vector at time $x = x_{initial} + nh$, h is the infinitesimal time-increment. THe boldface $\mathbf{y_n}$ denotes the state vector

The state vector $\mathbf{y} = (x, y, z)$ from the 3-D Rössler system

$$\begin{aligned} \frac{dx}{dt} &= -y - z \\ \frac{dy}{dt} &= x + ay \\ \frac{dz}{dt} &= b + z(x - c) \end{aligned} \tag{3.2}$$

For the discrete time case, $\mathbf{x_{n+1}} = \mathbf{f}(\mathbf{x_n})$ and the system is solved by calling the increment function at each discrete time step. For the system used, that is, the Logistic map, the equation is

$$x_i(n+1) = f(x_i(n), x_j(n)) = 1 - a[x_i(n)]^2 + \sum_{j=1}^{N} G_{ij} H_j(x_j(n)) \qquad (3.3)$$

The coupling matrix is symmetric nearest neighbour coupling as shown above. The matrix looks like

$$\mathbf{G} = \frac{N}{2P} \begin{bmatrix} -2\sum_{p=1}^{P} & a_1 & \dots & a_p & 0 & \dots & 0 & a_p & \dots & a_1 \\ a_1 & -2\sum_{p=1}^{P} & a_1 & \dots & a_p & 0 & \dots & a_p & \dots & a_2 \\ \dots & & & \dots & & & \dots & & & \\ a_1 & \dots & a_p & 0 & \dots & 0 & a_p & \dots & a_1 & -2\sum_{p=1}^{P} \end{bmatrix} \qquad (3.4)$$

For values of a_p falling within certain bounds, which can be obtained from the bounds on the eigenvalues in this case

$$|\lambda + N| < Nexp(-h_{max}), \qquad \lambda \neq 0 \qquad (3.5)$$

And the relation between the eigenvalues and the values of a_p

$$\lambda_l = -\frac{2N}{P} \sum_{p=1}^{p} a_p sin^2 \frac{\pi p(l-1)}{N}, \qquad l = 0, 1, \dots, N-1 \qquad (3.6)$$

One gets bounds on the eigenvalues for the stable regime.

By setting an eigenvalue outside these bounds, one can destabilise the corresponding eigenvector or eigenmode, in the case of symmetric coupling, to produce a pattern.

For each given eigenvalue of the coupling matrix G, we get an inequality in P variables. We get N such inequalities for the N eigenvalues. This set of inequalities determines whether we see synchrony or asynchrony in the network.

3.3 Algorithm to Reproduce and Image Using a Matrix

The next area of study was the use of specific patterns and creating a matrix that would generate the given pattern. One can use the fact that one of the eigenvectors of the coupling matrix has to be (1,1,1,...1,1) for the synchronised manifold, with corresponding eigenvalue 0. Seeing that a particular eigenvector of the coupling matrix would grow exponentially with time if the corresponding eigenvalue was outside the stability bound, we take the second eigenvector to correspond to the desired spatial pattern and its corresponding eigenvalue to be outside the stability bound.

Thus, one has two eigenvectors and eigenvalues, and can arbitrarily set the remaining eigenvalues within the stability bounds. Using the two known eigenvectors, one can create an orthogonal basis of eigenvectors and use the eigenvalues to create a matrix G which will realise the desired pattern.

Therefore, given a vector **v** that represents a given pattern to be realized, one can generate the coupling matrix that will produce this pattern when the system is evolved through the following steps.

1. For **v** of length N, generate N eigenvalues such that the first is 0, the second lies outside the stability bounds as specified by the dynamical equation of each node, (computed using the Lyapunov exponent) and the remaining are selected randomly from within the bounds.

2. The null-space of the synchronized manifold, $(1,1,1,1,...,1,1,)^T$ and the vector **v** complete the basis set that can be used to compute the coupling matrix G for which the instability is along the vector **v** so that the corresponding pattern is seen.

3. The linearly independent basis isorthogonalised using the Gram Schmidt method to produce a set of orthogonal eigenvectors **v**.

   For the Gram-Schmidt method, we take a linearly independent finite set of vectors and define the following projection operator

$$proj_u(\mathbf{v}) = \frac{\langle \mathbf{v}, \mathbf{u} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle}\mathbf{u} \tag{3.7}$$

   Using this, in general, each u_i is modified as

$$U_i = v_i \sum_{j=1}^{i-1} proj_{u_j}(v_i) \tag{3.8}$$

Here, the v_i denote the vectors that make up the linearly independent basis and the u_i denote the new orthogonal vectors produced.

4. The matrix G is given by $\mathbf{UWU^{-1}}$. Where W is the diagonal matrix of eigenvalues as initially generated.
5. To represent an image in terms of a matrix, first the image is modified so that the resolution is as desired by averaging the pixel values over the appropriate window. Then the image matrix is reshaped to produce an eigenvector. Next the vector of ones of the same length as this modified image vector is used to produce a matrix G using steps 1-4.

Chapter 4

Results

For the case where the dynamical system is in the fixed point regime, we consider the Logistic map $x(n+1) = f(x(n)) = 1 - a * x(n)^2$. A network of size $N = 5$ was used.

The Lyapunov exponent for the system with $a = 0.5$ is -0.32.
The Lyapunov exponent for the chaotic case, $a = 1.9$ is 0.54.
We use the relation between the eigenvalues of G and the maximal Lyapunov exponent.

$$|\lambda + N| < exp(-h_{max}) \tag{4.1}$$

And the relation between the coupling strengths a_p where $p = 1, 2$ to get a set of inequalities in terms of the maximal Lyapunov exponent.
For $N = 5$ and $h_{max} = -0.32$ we have the following relation

$$-1.36 < 1 - a_1 sin^2(\pi l/5) - a_2 sin^2(2\pi l/5) < 1.36, \qquad l = 1, 2 \tag{4.2}$$

Likewise, for $N = 5$ and $h_{max} = 0.54$ we have the relation below.

$$-0.578 < 1 - a_1 sin^2(\pi l/5) - a_2 sin^2(2\pi l/5) < 0.578, \qquad l = 1, 2 \tag{4.3}$$

For two values of l we get two sets of inequalities with which we can set the values of a_1 and a_2 to get specific patterns. Since we can vary both a_1 and

a_2 independently to be outside the bounds of stability, we have two possible patterns that we can form. In the spatial sense, one is a long wavelength pattern while the other is a short wavelength pattern.

In the plots generated below, the five greyscale bars are each for a different oscillator. The vertical axis is for time.The figures below for the fixed point and chaotic case are reproductions of the figures in the paper [5]. The figures on the Rössler system have no analogues in the paper [5] but are produced on the same principle.
For $a = 0.5$, the system settles to a fixed point. The short wavelength pattern can be obtained by putting $a_1 = 3$, $a_2 = 1$.It is shown in Figure 4.1.

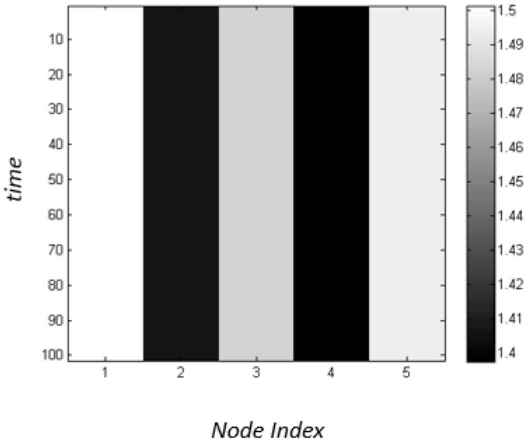


Figure 4.1: For $a = 0.5$ in the Logistic map, the system settles to a fixed point. The short wavelength pattern for fixed point dynamics can be obtained by putting $a_1 = 3$, $a_2 = 1$.

Putting $a_1 = 1$, $a_2 = 3$, we get the so-called long-wavelength pattern, shown below in Figure 4.2.

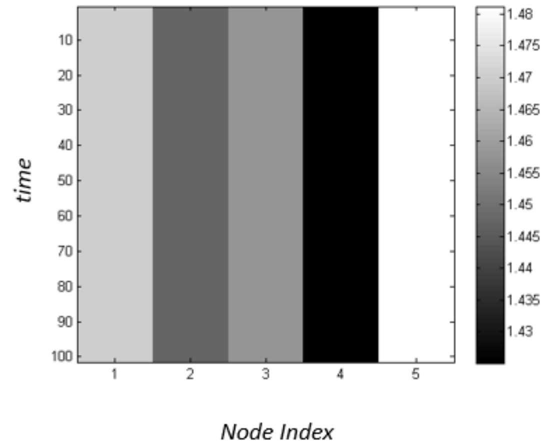


Figure 4.2: Setting the parameter $a = 0.5$ for the Logistic map and putting $a_1 = 1$, $a_2 = 3$, we get the so-called long-wavelength pattern under fixed point dynamics

One can see that due to the coupling strengths being stronger for the immediate neighbours in the one case, that configuration leads to the short wavelenght pattern where immediate neighbours are strongly correlated. For the other case where the next-nearest neighbour is strongly coupled, we see the long-wavelength pattern where the correlated nodes include the next-nearest neighbours. The symmetric neighbour coupling is a good case-study to consider the effects of coupling.

Next, the case of chaotic dynamics was considered.

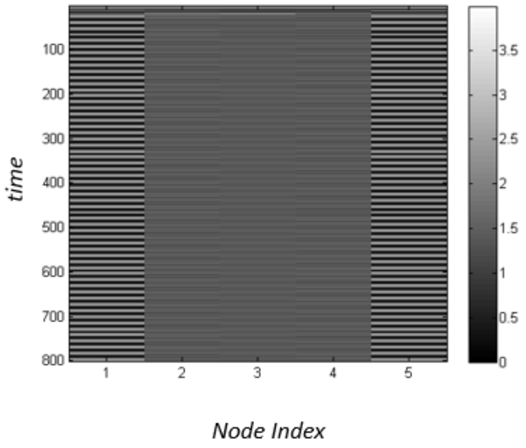


Figure 4.3: Here, the parameter $a = 1.9$ for the Logistic map equation, the coupling strengths are $a_1 = 1$, $a_2 = 3$. One observes a pattern similar to the long wavelength case Figure 4.2 except that the time trace shows chaotic dynamics instead of time invariant fixed point dynamics asymptotically.

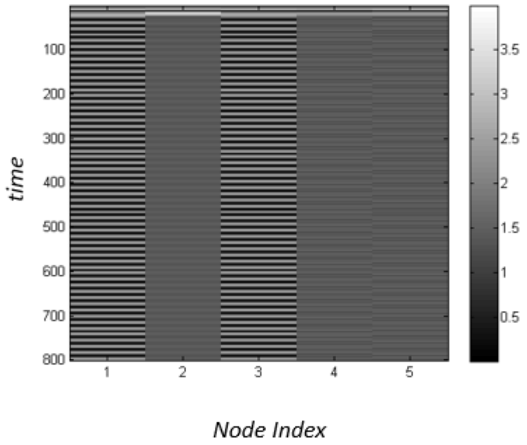
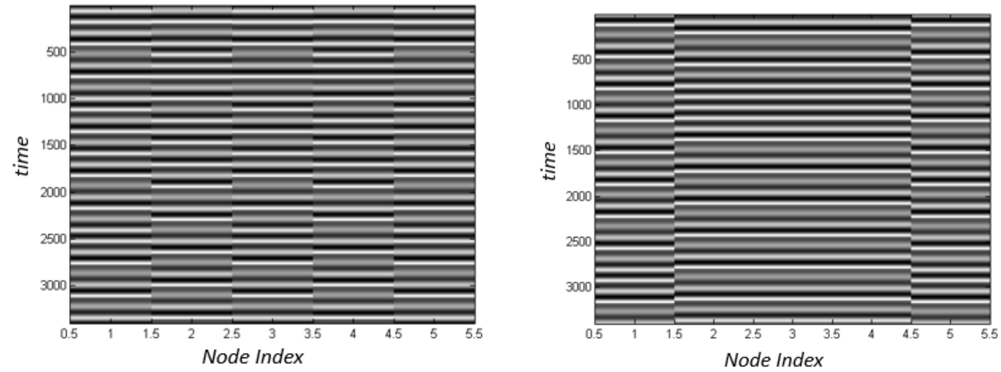


Figure 4.4: Here, the parameter $a = 1.9$ for the Logistic map equation, the coupling strengths are $a_1 = 3$, $a_2 = 1$. One observes a pattern similar to the short wavelength case Figure 4.1 except that the time trace shows chaotic dynamics instead of time invariant fixed point dynamics asymptotically.

4.1 Applied to the 3-D Rössler system

In the figures Figure 4.5a and Figure 4.5b, the Rössler has been used and the first dimension plotted against time for the N oscillators. The patterns described are respectively the long and short wavelength ones.



(a) The pattern is [1,0,1,0,1].

(b) The pattern is [1,0,0,0,1].

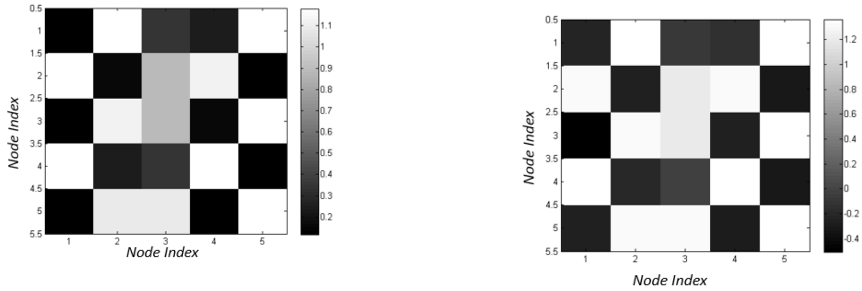
Figure 4.5: The nodes follow Rössler system dynamics with parameters, $a = 0.2$, $b = 0.2$ and $c = 5.7$. Two patterns are produced on a network of 5 nodes using the method described in Section 3.

4.2 Using 2-D or biperiodic Coupling

Similarly, a 2-D lattice coupling case was checked Figure 4.6a and reproduced using the method described in Section 3 as Figure 4.6b. In this case, a snapshot of the oscillator amplitudes of theN^2 oscillators is taken and plotted below. One can see roughly a checkerboard pattern emerge. Here, $N = 5$. This means there are 25 oscillators in the network.

The first one is from the $[-2.4, -1]$ nearest neighbour 2-d lattice map, that is, the coupling strength between two elements connected laterally is -2.4 and that between two elements connected vertically is -1. It should be noted that for 2-D coupling, one is essentially implementing 1-D coupling with the nearest neighbour and the N+1th neighbour. A=0.5 so the dynamics is in the stable regime. The image on the right is produced after using the pattern in the first to reproduce the image. Once can see small variations in the scales and the relative values of different cells, however, the similarity is marked. This same procedure was repeated with the case of nearest and next-nearest

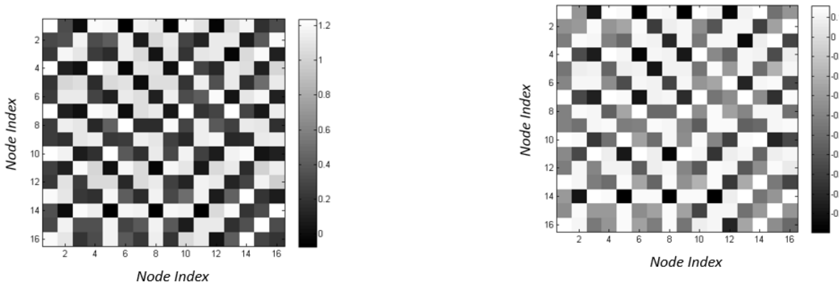
2-d coupling.
In this case Figure 4.7a and Figure 4.7b, each node was connected to the 4 closest nodes both laterally and vertically. The lateral coupling strengths were $[-18.4, -10]$, the vertical were $[-12, -8]$. The parameter $a = 0.5$ and the number of oscillators was 256 on a $16 * 16$ network.



(a) The pattern produced using the nearest neighbour coupling scheme.

(b) A reproduction of the adjacent image

Figure 4.6: This figure is produced from the $[-2.4, -1]$ nearest neighbour 2-d lattice map, that is, the coupling strength between two elements connected laterally is -2.4 and that between two elements connected vertically is -1.here, too, the fixed point dynamics of $a = 0.5$ was used in the Logistic map for the nodes. The second image is a reproduction produced using the algorithm in Section 3.



(a) Here each node was connected to the 4 closest nodes both laterally and vertically.

(b) This one is a reproduction of the adjacent figure using the algorithm described in Section 3.

Figure 4.7: Here each node was connected to the 4 closest nodes both laterally and vertically. The lateral coupling strengths were $[-18.4, -10]$, the vertical were $[-12, -8]$. The parameter $a = 0.5$ and the number of oscillators was 256 on a $16 * 16$ network.

For the figures above Figure 4.6a and Figure 4.6b followed by Figure 4.7a and Figure 4.7b, the general form of the time evolution equation of the system is:

$$\mathbf{x}_{j,k}(n+1) = \mathbf{f}(\mathbf{x}_{j,k}(n)) + \frac{1}{2P}\sum_{p=1}^{p}[a_p[\mathbf{f}(\mathbf{x}_{j+p,k}(n)) + \mathbf{f}(\mathbf{x}_{j-p,k}(n))]$$
$$+b_p[\mathbf{f}(\mathbf{x}_{j+p,k}(n)) + \mathbf{f}(\mathbf{x}_{j-p,k}(n))] - 2(a_p+b_p)\mathbf{f}(\mathbf{x}_{j,k}(n)], \quad j,k = 1,2,\ldots,L \tag{4.4}$$

Where the dual index specifies the 2-D nature of the network where each node has 4 neighbours instead of 2. As in the 1-D case, finding the eigenvectors, that is, the fourier modes of the matrix, we get the eigenvalues as earlier and they obey the following constraint based on the maximal Lyapunov exponent $h_m ax$.

$$\left|1 - \frac{2}{P}\sum_{p=1}^{p}(a_p sin^2(\pi pl/L) + b_p sin^2(\pi pm/L))\right| < exp(-h_1),$$
$$l,m = 1,2,\ldots,L/2or(L-1)/2 \tag{4.5}$$

Here we have $l+m$ constraints and we work with the assumption that the extent of connection P along the lateral direction is the same as along the vertical, meaning that if a node (j,k) is connected to its neighbours $j \pm 1$ and $j \pm 2$ then it will be connected to $k \pm 1$ and $k \pm 2$ as a result. This can be generalised to abandon this constraint by setting P to be the max of either of these two cases, so that the a_p or b_p can be set to zero where necessary and the general form of the constraint is preserved.

4.3 Application to Encoding an Image

The method was now used on a real image. The resolution of the image was reduced by averaging over groups of pixels for ease of computation. The image used is a common one in these contexts Figure 4.8.

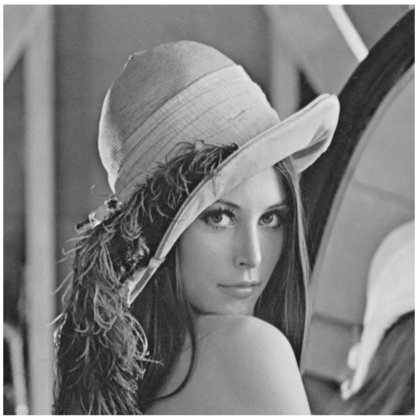
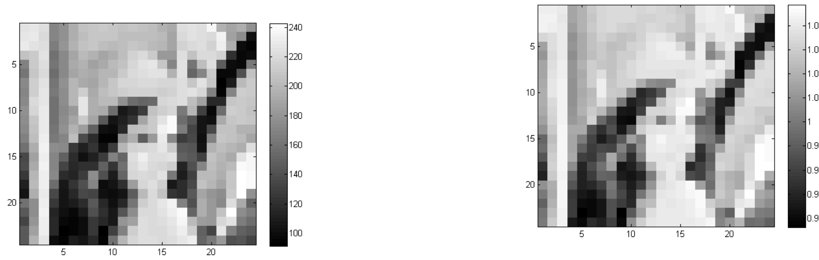


Figure 4.8: The image of Lenna in 256*256 resolution and in monochrome.

The low resolution version was used to produce a 24*24 length image vector. This is seen in Figure 4.9a and Figure 4.9b.
Correspondingly, this vector and the 24*24 vector of ones were used to create a new orthonormal basis and the matrix of connections with the eigenvalue corresponding to the second eigenvector, i.e., the image itself set to destabilise the system along this eigenvector to produce the image asymptotically Figure 4.9b



(a) The image of Lenna downsampled
The image of Lenna was downsampled to 24*24

(b) The downsampled image of Lenna reproduced
The downsampled image was used as the eigenvector

Figure 4.9: The image of Lenna was downsampled to 24*24 and The downsampled image was used as the eigenvector and a coupling matrix that would reproduce the image asymptotically when nodes following fixed point dynamics of the Logistic map were coupled.

The method becomes very computationally inefficient at higher resolutions. Parallel computing methods are not immediately applicable since the computation at a given node is dependent on the computation at its adjacent nodes.
It is possible to splice the image and to perform the matrix representation and image reproduction method on each splice in parallel to get the higher resolution image back. However, the recovered image differs from the original as can be seen below in Figure 4.10.

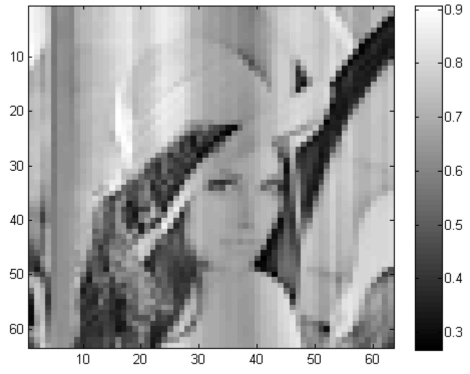


Figure 4.10: The image of Lenna was downsampled to 64*64. Each individual strip was separately reproduced using the algorithm described in Section 3 and the final image was reassembled.

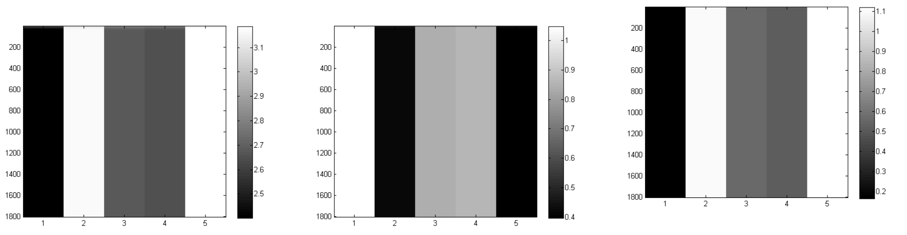
One notices differences in Figure 4.10 from the original Figure 4.8 in that some strips are on average a different shade from that in the original. This is because each strip is computed separately and the basis of eigenvectors used to reproduce it have been normalised independent of the other strips as opposed to the non-parallel case where there was only one instance of normalisation.

4.4 Selective Pattern Formation and Pattern Inversion

A point that deserves mention here is the case of multiple eigenvalues being set outside the bounds of stability. Consider for example if two distinct patterns were used, along with the synchronised manifold, to generate the set of eigenvectors of the coupling matrix, one could then experiment with the eigenvalues used to complete the coupling matrix and observe the following :

1. If the eigenvalue corresponding to the first pattern is set unstable and all the others stable then the first pattern is observed.
2. If the eigenvalue corresponding to the first pattern is set stable, the second stable and all the rest unstable, the second pattern is observed.
3. If the eigenvalues of both patterns are set unstable then only the second pattern is seen.
4. If the eigenvalues of both patterns are set unstable and also set one other eigenvalue unstable then a pattern distinct from any of the three eigenvalues is observed. This combination is not the same as the algebraic sum of the patterns (under whatever scaling is used to quantify them) or the linear superposition of them

These observations are useful in determining constraints under which the method may be employed to reproduce the image.
Another interesting phenomenon that could be observed in the case of the Logistic map under the stable regime was that for the same eigenvalues and eigenvectors, one could either obtain the original pattern used to produce the coupling matrix representations or an inversion of the pattern depending on the initial values used



(a) The inversion triplet The original pattern

(b) The inversion triplet The pattern inverted

(c) The inversion triplet The original pattern reproduced

Figure 4.11: The figure above shows in the three panels, first a given pattern produced using fixed point dynamics of the Logistic map for the nodes, second, an inverted pattern produced and third, the original pattern reproduced again.

In all the three cases Figure 4.11a, Figure 4.11b and Figure 4.11c, the coupling matrix was identical as well as the parameter for the Logistic equation as well. Examination shows that initial conditions play a role in determining the nature of the image reproduced according to the rules listed below.
If one considers a given node and its asymptotic value according to the pattern,

1. If the asymptotic value is of high magnitude and the difference between the initialisation and the asymp totic mean of all the nodes is low in magnitude, one observes an inversion in the final image.

2. If the asymptotic value is of low magnitude and the difference between the initialisation and the asymptotic mean of all the nodes is low in magnitude, one observes the original image itself.

3. If the initialisation is outside the basin of attraction for the given attractor, one gets unbounded asymptotic values.

The Logistic equation itself for the parameter value used converges to a single fixed point over time. This fixed point is stable. The alternate fixed point arises only when coupling is incorporated and that made it hard to study the inversion analytically to draw conclusions.

Chapter 5

Discussion

This report represents a method and the implementation to represent images using networks of coupled dynamical systems. As shown, it is possible given an image of size N, to generate a matrix of couplings between nodes that will reproduce the desired image when each node represents a non-linear dynamical system in a network defined by the above-mentioned matrix of coupling strengths of sizye N^2.

A central concept in this study, the Lyapunov exponent is predominantly employed as a means to quantify sensitive dependence on initial conditions. It measures how the relative distance between two nearby trajectories evolves over time as the two trajectories are evolved using the dynamical map. It is therefore one of the most commonly used methods to identify chaotic dynamics. An important feature of the work is the use of the Lyapunov exponent as in earlier works [17] and in other articles since then, to measure how perturbations transverse to the synchronised manifold in a system evolved over time, i.e., whether these perturbations grow in magnitude or decreased. This provides a quantitative handle on the stability of the synchronised manifold.

Another highlight of the work is the use of a number of mathematical tools useful in studying matrix forms of networks, namely adjacency lists. The eigenvalues of the coupling/adjacency matrix are the primary handle on the existence of a trajectory for global synchronisation and on the stability analysis of this trajectory.

The method opens the door to many possibilities in the question of training a network for robust image retreival and image encryption. In particular, employing techniques in compressive or sparse sensing [7] on the coupling matrix is one possibility for encrypting the information in the image. A sparse matrix is one whose entries are predominantly zero. It is possible to compress the information of such matrices to much smaller sizes than the original matrix. This technique is used in compressive image transmission by considering a transform of the image that satisfies this criteria, such as the Fourier transform.

The receiver needs to be aware of the dynamical system and parameters thereof for the nodes. Accordingly, by reconstructing the coupling matrix using sparse representations and sparse sensing techniques such as FOCUSS, the image that is the original message can be reconstructed at the receivers end.

One drawback of this method is its computational cost, and the cost of transmitting the information. While most image compression and transmission involves sending less information than the image itself, in this case, a coupling matrix of size N^2 is required to represent an image of size N. Here too, methods like compressive sensing can make this a technique with real-world application; this involves considering only a part of the coupling matrix rather than the entire matrix. The FOCUSS algorithm is applicable in this context. However, it uses a dictinoary matrix to reconstruct the original data from the set of sparse samples. Often, the only way to find the appropriate dictionary matrix is by using a training algorithm in conjunction with the FOCUSS algorithm which minimizes the error iteratively[7, 8]. Therefore, even in cases of convergence, the converged dictionary matrix may not reproduce the original coupling matrix exactly.

References

[1] R. E. Amritkar and G. Rangarajan. Stability of multi-cluster synchronization. *International Journal of Bifurcation and Chaos*, 2009.

[2] P.; Abarbanel-H. Brown, R.; Bryant. Computing the lyapunov spectrum of a dynamical system from an observed time series. *Physical Review A*, 43(6), 1991.

[3] R.; Abarbanel-H. Bryant, P.; Brown. Lyapunov exponents from observed time series. *Physical Review Letters*, 1990.

[4] A. Kolmogorov et al. A study of the diffusion equation with an increase in the amount of substance, and its applications to a biological problem. *Moscow Univ. Bull. Math*, 1937.

[5] Y. Chen G. Rangarajan and M. Ding. Generalized turing patterns and their selective realization in spatiotemporal systems. *Physics Letters A*, 2003.

[6] A. L. Hodgkin and A. F. Huxley. A quantitive description of membrane current and its application to conduction and excitation in a nerve. *J Physiol.*, 1952.

[7] Bhaskar D. Rao Irina F. Goroditnitsky. Sparse signal reconstruction from limited data using focuss: A reweighted minimum norm algorithm. *IEEE Transcations on Signal Processing*, 45, 1997.

[8] Kenneth Kreutz-Delgado Joseph F. Murray. Learning sparse overcomplete codes for images. *VLSI Signal Processing Systems*, 46, 2007.

[9] Edward Norton Lorenz. Deterministic nonperiodic flow. *Journal of Atmospheric Sciences*, 20:130–158, 1963.

[10] Aaron M. Hagerstrom Thomas E. Murphy Louis M. Pecora, Francesco Sorrentino and Rajarshi Roy. Cluster synchronization and

isolated desynchronization in complex networks with symmetries. *Nature Communications*, 2014.

[11] A. S. Pikovsky M. G. Rosenblum and J. Kurths. From phase to lag synchronization in coupled chaotic oscillators. *Phys. Rev. Lett*, 1997.

[12] R. May. Simple mathematical models with very complicated dynamics. *Nature*, 261(5560):459–467, 1976.

[13] L Meirovitch. *Elements of Vibration Analysis, second edition.* McGraw-Hill Science-Engineering-Math, 1986.

[14] Arkady S. Pikovsky Michael G. Rosenblum and JÃijrgen Kurths. Phase synchronization of chaotic oscillator. *Phys. Rev. Lett.*, 1996.

[15] C. Morris and H. Lecar. Voltage oscillations in the barnacle giant muscle fiber. *Biophys. J.*, 1981.

[16] Louis M. Pecora and Thomas L. Caroll. Synchronisation in chaotic systems. *Phys. Rev. Lett.*, 1990.

[17] Louis M. Pecora and Thomas L. Caroll. Master stability functions for synchronized coupled systems. *Phys. Rev. Lett.*, 1998.

[18] Fitzhugh R. Mathematical models of threshold phenomena in the nerve membrane. *Bull. Math. Biophysics*, 1955.

[19] G. Rangarajan and M. Din. Stability of synchronized chaos in coupled dynamical systems. *Physics Letters A*, 2002).

[20] O. E. Rossler. An equation for continuous chaos. *Phys. Lett. A*, 57:397–398, 1976.

[21] R. E Amritkar S. Jalan and C. K. H. Synchronized clusters in coupled map networks -i. numerical studies. *Phys. Rev. E*, 2005.

[22] S. H. Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology and chemistry.* Perseus publishing, 2001.

[23] A. M. Turing. The chemical basis of morphogenesis. *Phil. Transact. Royal Soc.*, 1952.

[24] G. Rangarajan Y. Chen and M. Ding. Stability of synchronized dynamics and pattern formation in coupled systems: Review of some recent results. *Communications in Nonlinear Science and Numerical Simulation*, 2006.

Appendix A

Code

This appendix contains some of the code used to generate the figures in this report.

Code to calculate the Lyapunov Exponent

```
function mu = LyapExpo(a,b,c,Ndim,dt,tmax)
Ndir = Ndim                                  ;
% number of vectors that are being evaluated to calculate the Lyapunov
exp
                            ;
nmax = round(tmax/dt)                       ;% nbre of time steps
;
v = randn(Ndim,1)                           ;
%========================
% Initialize everything
%=========================
Y      = zeros(Ndim,Ndir)      ;
%% Ndir specifies the directions along which Lyapunove exponent will
be calculate
%% for Ndir = Ndim, the dimensionality of the dynamical system, the
entire spectrum of the uncoupled Lyapunov exponent is calculated.
weight = zeros(Ndir,nmax)      ;
mu     = zeros(Ndir,nmax)      ;
uu     = zeros(Ndim,nmax)      ;
Edir   = randn(Ndim,Ndir)      ;
uu(:,1) = v                    ;
for itime = 2 : nmax
  v = rk4_step(a,b,c,dt,v)  ;
  uu(:,itime) = v            ;
end % this bit creates a Rossler time series
for  itime = 1 :  nmax
     for idir = 1 : Ndir
        Y(:,idir) =  jac_step(a,b,c,dt,uu(:,itime),Edir(:,idir))  ;  %
Tangent linear model
        % here, at each time step the state vector is used, to
calculate
        % the jacobian at each time stepand propagate the perturbation
        % using the linearised model
     end
     [QQ TT] = qr(Y)                  ;%GramShmidt orthogonalization
     %% QQ is the orthonormal basis and TT is the triangular matrix
with the eigenvalues
     weight(:,itime) = diag(TT)       ;%use the eigenvalues and take
sum
    for idir = 1 : Ndir
        tmp = 0.0 ;
        for j = 1 : itime
           tmp = tmp + log(weight(idir,j))   ; %accumulate sume of
abs(eigenvalues)
        end
        mu(idir,itime) = real(tmp)/(dt * itime)    ; %divide by
current time
        %the final column of mu is the asymptotically converged value
of
```

```
        %the Lyapunov exponent, the entire time series has been stored
to
        %see when the ecxponent converges
    end
    Edir = QQ   ;% reset the perturbations to an orthonormal basis
that will be propagated using the jacobian again
end
end
```

Code to calculate the increment for the discrete time Logistic map

```
function output = increment_rhs(x,a,sigma,G,H,P)
n = size(G);
output = zeros(n(1),1);
dim =1;
for i =1:n(1)
    output(i) = 1 - a*x(i).^2;
    accum1 = 0;
    for j=1:n(1)
        %below for H(x) = f(x)
        accum1 = accum1 + (sigma/(n(1)))*G(i,j)*(1 - a*x(j).^2);

        %below for H(x) = x in 1D
        %accum1 = accum1 + sigma*(1/n(1))*G(i,j)*x(j)

        %below for H(x)  = x in many D - note that dim!=1 and outpute
is of
        %dimension dim., as is accum1.

%         accum2 = zeros(1,dim);
%         for k =1:dim
%             for l = 1:dim
%             accum2(1,k) = accum2(1,k) + H(k,l)*x_base(j,l);
%             end
%         end
%         accum1 = accum1 + sigma*(1/n(1))*G(i,j)*accum2;

    end
    output(i) = output(i) + accum1;
end
end
```

Code to calculate the derivatives for the continuous time Rossler system

```
function output = derivatives(a,b,c,sigma,G,H,x_base)
%% derivatives for the 3-d Rossler system with H(x) = I(NxN)
n = size(G);
```

```
m = size(x_base);
members = m(1);
dim = m(2);
output = zeros(n(1),dim);
for i =1:n(1)
    output(i,1)  = - x_base(i,2) - x_base(i,3);
    output(i,2) = x_base(i,1) + a*x_base(i,2);
    output(i,3) = b + x_base(i,3)*(x_base(i,1) - c);
    accum1 = zeros(1,dim);
   %%code block below includes effect of coupling with H(x) = I(NxN)
    for j=1:n(1)
        accum2 = zeros(1,dim);
        for k =1:dim
            for l = 1:dim
            accum2(1,k) = accum2(1,k) + H(k,l)*x_base(j,l);
            end
        end
        accum1 = accum1 + sigma*(1/n(1))*G(i,j)*accum2;
    end
    output(i,:) = output(i,:) + accum1(1,:);
end
end
```

Code to calculate the RK4 increment for the continous time Rossler system

```
function x = rk4_step(dt,a,b,c,sigma,G,H,x)
%function that computes rk4 with the drivatives that include the
coupling
k1 = derivatives(a,b,c,sigma,G,H,x);
spam1 = x + 0.5*k1*dt;
k2 = derivatives(a,b,c,sigma,G,H,spam1);
spam2 = x + 0.5*k2*dt;
k3= derivatives(a,b,c,sigma,G,H,spam2);
spam3 = x + k3*dt;
k4 = derivatives(a,b,c,sigma,G,H,spam3);
kk = (1/6)*(k1+2*(k2+k3)+k4);
x = x+dt*kk;
end
```

Code to plot a pattern based on a given coupling matrix or to reproduce a predetermined pattern for the continuous time Rossler system.

```
% % % %%% 1-d scene
% ap = [8];
% P = length(ap);
% G = G_writer(ap,5);


%%%% 2-d scene
% % % ap = [-18.4,-10];
% % % bp = [-12,-8];
% % % P = length(ap);
% % % G = coupler_2d(ap,bp,16);

%%%%%% the image is read from a .png file and downsampled to produce
the
%%%%%% image vector
% % % resolution = 8;
% % % image = imread('lena.png');
% % % image_prep(resolution,image);
% % % patt_vec1 = csvread('patt_hat3.csv');
% % %
% % % %%%%%% the coupling matrix is generated using the image vector,
synch
% % % %%%%%% manifold and by initialising the eigenvalues according to
the
% % % %%%%%% stability bounds
% % %
% % % [m,n] = size(patt_vec1);
% % % patt_vec = reshape(patt_vec1,n*m,1)
h_max = 0.07140;
lower_bound =  -n*m*(1 + exp(-h_max));
upper_bound = n*m*(exp(-h_max) - 1);
stable = upper_bound - upper_bound/2;
unstable = upper_bound + 2*upper_bound;
patt_vec = [1,0,0,0,1]';
unstable = 0;
stable = -23.3;
G = construct_matrix_2c(h_max,patt_vec,unstable,stable);
dlmwrite('second_G.csv',G,',')

[n1,n2] = size(G);
dim = 3;
T=4000;
H = eye(dim);
a=0.2;
b=0.2;
c=5.7;
sigma=1;
dt=0.05;
x=zeros(n1,dim,T);
x(:,:,1) = (02*rand(n1,dim)-1);
time_trace = zeros(1,T);
time_trace(1,1)=1;
for t=2:T
    time_trace(t) = time_trace(t-1)+1;
    x(:,:,t) = rk4_step(dt,a,b,c,sigma,G,H,x(:,:,t-1));
    if mod(t,50) ==0
        t
    end
end
% size(x(1,1,:)) %% plotting the first dimension of N oscillators vs
time
image = imagesc(transpose(squeeze(x(:,1,600:T))));
colormap(gray);
% % plot(squeeze(x(1,2,:)),squeeze(x(1,1,:)))

%%%% for the Lenna image
```

Code to produce a given pattern using a coupling matrix or to reproduce a pre-determined pattern for the discrete time logistic map

```
clear all

%%%%G can be read from a csv file, or constructed using an eigenvector
and
%%%%the max Lyap exponent or by constructing a nearest neighbour/
lattice
%%%%coupling

%%%%% csv read
%G = csvread('coupling_mat_SW.csv');
 %G = 2.*G
%%%% constuct from eigenvector
%%% long wavelength
%G = construct_matrix_2c(-0.39,[-0.4802,-
0.5399,0.1465,0.6304,0.2431]')
%%% short wavelength
% patt_vec = csvread('patt_hat.csv');
% % G = construct_matrix_2c(-0.39,[-0.6321,0.4984,-0.1743,-
0.2163,0.5244]')
% G = construct_matrix_2c(-0.39,patt_vec)

%P=2;

%%%%%%%% written using the nearest-neighbour coupling strengths 1-d
case
%for short wavelength at fixed point dyn, ap = [3,1]
%G = G_writer([3,1],5)
%  G = G_writer_eval([1,1,2],5,-0.39);
P = 2;
%%%%%%%%%%% written using nearest-neighbour coupling, 2-d case
% for (2,2) mode and chaotic dyn (a=1.9), ap=0.9 and bp = 0.9
ap = [0.9];
bp = [0.9];
```

```
P = length(ap);
G = coupler_2d(ap,bp,5);

n = size(G);
T=2000;  %%% time steps
dim=1; %%% not equal to 1 for many imensional systems
H = eye(dim); %%%% H atrix for a matrix transform coupling function

%%%%%%%% inititalise parameters
a=0.5; %%%%% 1-d logistic map
%%%%%%%%%%%%%%%%%%%%

replicates = 1;
sigma=0.1; %  global coupling constant

x=zeros(replicates,n(1),T);
z=zeros(replicates,n(1),T);
for i = 1:replicates
x(i,:,1) = 0.05*(2*rand(n(1),1)-1); % initial conditions
%  x(i,:,1) = -0.2*[1,-2,3,-3,2,-1,1,-1,3,-3];
x(i,:,1);
time_trace = zeros(1,T);
time_trace(1,1)=1;
for t=2:T
%     if t <= 1000
%         G = G_writer([3,1],5);
%     else
%         G = csvread('looped_mat.csv');
%     end
    time_trace(t) = time_trace(t-1)+1;
    x(i,:,t) = increment_rhs(x(i,:,t-1),a,sigma,G,H,P);
    %%%% absolute value of the deviations from the mean, as described
in
    %%%% the paper
    z(i,:,t) = abs(x(i,:,t) - sum(x(i,:,t)));
%     if mod(t,50) ==0
%         t;
%     end
end
% x(i,:,end);
end
% zz = zeros(replicates,floor(n(1)/2),T);
% c=1;
% for i = 1:2:n(1)
%     zz(1,c,:) = z(1,i,:);
%     c=c+1;
% end
b = zeros(replicates,floor(sqrt(n(1))),floor(sqrt(n(1))),T);
c=1;

%%%%% block below is to take an N8N snapshot in the case of 2d
coupling at
```

```
%%%%% some given time 't' or of the time average of all the values

for i = 1:replicates
    c=1;
    for j=1:floor(sqrt(n(1)))
        for k=1:floor(sqrt(n(1)))
            for t=1:T
                b(i,j,k,t) = x(i,c,t);
%                 if x(i,c,t) > 0
%                     b(i,j,k,t) = 1;
%                 else
%                     b(i,j,k,t) = 0;
%                 end
            end
            c=c+1;
        end
    end
end

%%%%% take time average of the N*N values for each replicate
 bb = mean(b,4);
bb = b(:,:,:,400);
plot_data = squeeze(mean(bb(:,:,:),1));
figure
image = imagesc(plot_data(:,:));
%%%%%%%%%%%%%%%%%%%%


%%%% block below is for the 1-d case, plotting the absolute value of
% %%%% deviations
% plot_data = squeeze(mean(z,1));
% pattern = plot_data(:,2000)
% spam = isnan(pattern)
% % if sum(spam) == 0
% % dlmwrite('patt_hat.csv',pattern,'delimiter',',');
% end
%size(plot_data)
% figure
% image = imagesc(transpose(plot_data(:,200:2000)))
%%%%%%%%%%%%%%%%%

colorbar('eastoutside')
colormap(gray)
%plot(x(1,1:100))
% hold on
```

Code to downsample a given image

```
function im = image_prep(t_res,image)
%%% this function downsamples the original image according to the
desired
```

```
%%% resolution t_res and displays the downsampled image at the end
%%% windows of length t_res are successively chosen from the original
image
%%% and the average value of all cells in a given window is used for
the
%%% new image.
info = imfinfo('lena.png');
x_res = info.Width;
y_res = info.Height;
sv = floor(x_res/t_res);
n = x_res;
m = t_res;
im = zeros(t_res-1);
c1 = 1;
for i = 1:sv:n-sv
    c2 = 1;
    for j = 1:sv:n-sv
        window = image(i:i+sv,j:j+sv);
        value = mean(mean(window));
        im(c1,c2) = value;
        c2 = c2+1;
    end
    c1 = c1+1;
end
imagesc(im);
colormap(gray);
colorbar('eastoutside');
dlmwrite('patt_hat3.csv',im,'delimiter',',');
end
```