Dynamics on Small World Networks of Bi-stable Elements

Pranay Deep Rungta

A dissertation submitted for the partial fulfilment of BS-MS dual degree in Science

Indian Institute of Science Education and Research Mohali

April 2015

Certificate of Examination

This is to certify that the dissertation titled **Dynamics on Small World Networks of Bi-stable Elements** submitted by **Pranay Deep Rungta** (Reg. No. MS10004) for the partial fulfillment of BS-MS dual degree programme of the Institute, has been examined by the thesis committee duly appointed by the Institute. The committee finds the work done by the candidate satisfactory and recommends that the report be accepted.

Dr. Kamal P. Singh

Dr. Rajeev Kapri

Dr. Sudeshna Sinha
(Supervisor)

Dated: April 23, 2015

Declaration

The work presented in this dissertation has been carried out by me under the guidance of Dr. Sudeshna Sinha at the Indian Institute of Science Education and Research Mohali.

This work has not been submitted in part or in full for a degree, a diploma, or a fellowship to any other university or institute. Whenever contributions of others are involved, every effort is made to indicate this clearly, with due acknowledgement of collaborative research and discussions. This thesis is a bonafide record of original work done by me and all sources listed within have been detailed in the bibliography.

Pranay Deep Rungta
(Candidate)

Dated: April 23, 2015

In my capacity as the supervisor of the candidate's project work, I certify that the above statements by the candidate are true to the best of my knowledge.

Dr. Sudeshna Sinha
(Supervisor)

Acknowledgment

It has been a very pleasant experience working with my Supervisor: Prof. Sudeshna Sinha. She has been very helpful and encouraging at every stage of this project. I feel lucky to get the opportunity to work with her. She was very patient, flexible and guiding in making this work a progress. I would also like to thank Dr. Rajeev Kapri who helped me in coding whenever I required help.

I am also very thankful to my colleagues Shweta Kumari, Shivam, Imran and Chandrakala Meena for encouraging me throughout the project. Lastly, I would like to thank my family for the moral support and strength.

List of Figures

Abstract

This work is a development on a previous EPL paper of KP Singh, R Kapri and S Sinha (2012) on the dynamics of a globally coupled system of multi-stable elements. In this work we have investigated the sensitivity of small world networks to heterogeneity. Specifically, we consider a network of bi-stable elements coupled to four neighbours under different connection topologies. We show that as global bias tends to 0^- the network becomes hypersensitive to heterogeneity, even though the elements are connected to only a few other elements. Additionally we find that as the fraction of random links increases, the transition in the collective field gets sharper, for both static and dynamic links. Lastly, as we increase system size, we find again that the transition gets sharper. So it is evident that even a small coupling range, when randomized, can exhibit ultra-sensitivity to heterogeneity, similar to globally coupled systems.

Contents

1. Introduction

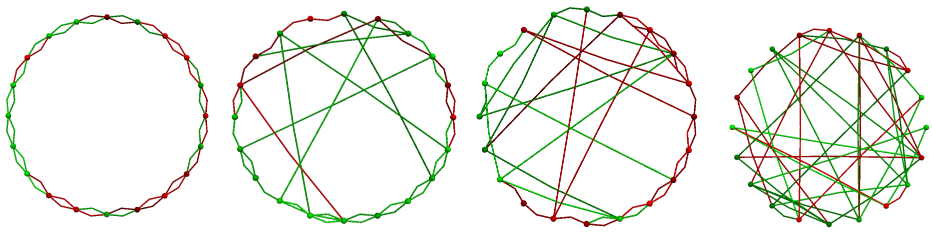


Figure 1.1: Small world network in the increasing order of random links from left to right.

Spatio temporal patterns are widely found in complex interactive systems [1] which can be described through networks. Study of networks can help in better understanding of systems, as microscopic as neurons in the brain, or as macroscopic as socio-economic interactions of humans or the growth of the internet [2]. All these systems have some set of variables that describe the state of the system. These may be as concrete as ion concentrations, or as abstract as information flow in human populations. The change of state of complex systems is marked by very complicated dynamical behavior of these variables.

Different kinds of network try to model one or more aspects of large coupled systems [6]. A popular model of connections is the small world network, interpolating between the completely regular lattice [1] and a completely random network[2]. Such a network consists of N nodes arranged in a ring where fraction $1 - p$ of the nodes are connected to their nearest neighbors, while fraction p are connected to random nodes. We show the schematic of such networks for varying p in Figure 1.1. These random connections provide shortcut paths for the spatial flow of information, leading to more efficient synchronization[5, 4, 7].

The starting point of the work in this thesis are the results of KP Singh, R Kapri and S Sinha (2012) [3] on the dynamics of a globally coupled system of multi-stable elements. The system they explored was heterogeneous, and consisted of a set of nonlinear elements that evolved to different stable states when uncoupled. The focus of that study was the role of heterogeneity in the emergent spatio-temporal patterns.

Specifically, this earlier work studied a collection of N dynamical elements with two distinct stable states x^*_+ and x^*_-. The equation governing the temporal evolution of each node were as follows:

$$\dot{x}_i = G(x_i) + a_i + C(\langle x \rangle - x_i) + b \tag{1.1}$$

where $i = 1, 2, \ldots, N$, with N being system size. $G(x_i)$ was a generic nonlinear function that gives rise to a multi-stable potential. The local parameter a_i may differ from element to element, leading to heterogeneity in the system. It determines the stable state x_i^* of the nodal dynamics in the uncoupled case, by giving a local bias to the dynamical element. Parameter b is a *global bias*, common to the elements, and can be used as a control to obtain different patterns. The elements are coupled through the *mean field* $\langle x \rangle = \frac{1}{N} \sum_{i=1}^{n} x_i$, which is a good indicator of the collective behaviour of the system. Lastly, C gives the strength of coupling.

The heterogeneous system in Ref. [3] consisted of two types of elements: N_0 elements with $a_i = 0$, and these elements were attracted to a stable state x_-^* when uncoupled, and the rest of the elements $N_1 = N - N_0$ had $a_i = 1$, and were attracted to the stable state x_+^*. Intuitively, one would think that contribution by each node is of the order of $O(1/N)$. Thus to draw the entire system towards a certain set, that set would have to be in the majority. However, very interestingly it was found, that under certain conditions, all the elements in the system evolved to the stable state of the minority population, namely the natural state of the set with a smaller number of elements. The underlying reason for this counter-intuitive behavior was the interplay of the relative depths of the different local steady states due to heterogeneity, and the strong global coupling of the system which leads to synchronization.

Further, for suitable global bias b this system could be made *ultra-sensitive* to heterogeneity in the system, and the collective field reflected the presence of the smallest deviation from uniformity in the local parameter a_i. In fact it was observed that in certain systems, even if *one* element had a different a_i, it could lead the entire system to its natural state. Thus, in these conditions the collective field of the system is any extremely sensitive detector of heterogeneity.

2. Model

In the present work we focus on the specific follow-up questions: How sensitive are small world networks to heterogeneity? More specifically, can one find conditions for bi-stable elements connected in small world topologies, where the coupled system becomes ultra-sensitive to any non-uniformity in the constituent elements? The ultra-sensitive system in earlier studies occurred under global coupling. Can one obtain similar sensitivity for systems where the coupling is confined to much smaller neighbourhoods? When coupling is not global, do the relative sizes of the coupled set to the full system size matter?

To answer all these interesting questions, we study the evolution of N dynamical elements, coupled to four neighbours each (namely a degree $k = 4$ network) evolving as follows:

$$\dot{x}_i = G(x_i) + a_i + C(\frac{1}{4}\sum_k x_k - x_i) + b \tag{2.1}$$

with $G(x_i) = x_i - x_i^3$ for $i = 1, \ldots N$. In the interaction term, Σx_k is a sum over the four neighbors of the i^{th} node. Note that the "neighbours" can be the local nearest and next nearest neighbours, or random non-local sites, depending on the topology of the underlying network.In order to probe the effect of local heterogeneity, we consider a network with N_0 nodes with $a_i = 0$, and $N_1 = N - N_0$ nodes with $a_i = 1$.

In order to understand the effect of the global bias better, we examine the potential wells for a system that evolves according to the dynamical equation:

$$\dot{x}_i = (x_i - x_i^3) + a_i + b \tag{2.2}$$

Such nodal dynamics describes a system with no coupling (i.e. $C = 0$) or an uniform synchronized system where $\langle x \rangle = x_i$.

Figures 2.1 - 2.4 show representative cases of the effective potential $V(x) = -\int \dot{x}\, \mathrm{d}x$ for different global bias b, and it is clear from the figures that positive and negative global bias swings the potential in favour of the positive and negative well respectively. Namely when $(a+b) < 0$, the well at $x < 0$ has greater depth than that at $x > 0$, and vice-versa for $(a + b) > 0$.

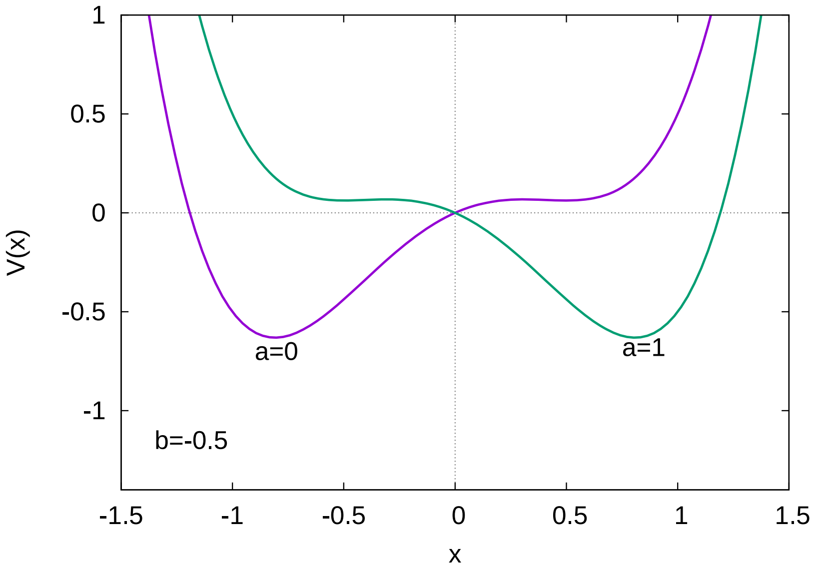


Figure 2.1: Effective potential for a single dynamical element i with $b = -0.5$ with $a_i = 0$ (violet) and $a_i = 1$ (green).

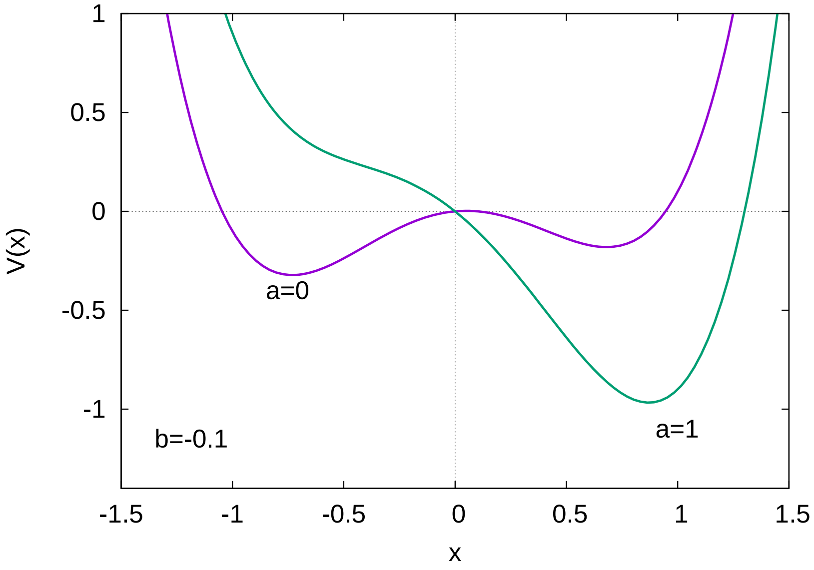


Figure 2.2: Effective potential for a single dynamical element i with $b = -0.1$ with $a_i = 0$ (violet) and $a_i = 1$ (green).

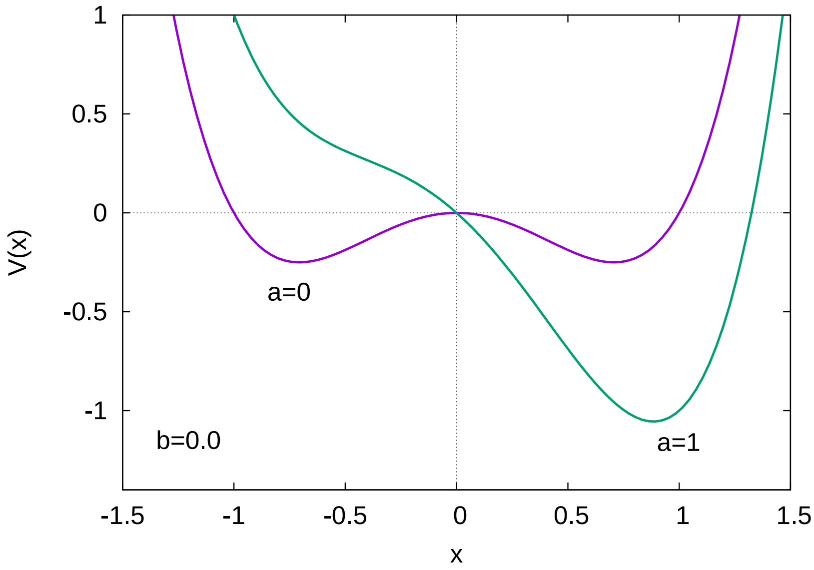


Figure 2.3: Effective potential for a single dynamical element i with $b = 0$ with $a_i = 0$ (violet) and $a_i = 1$ (green).

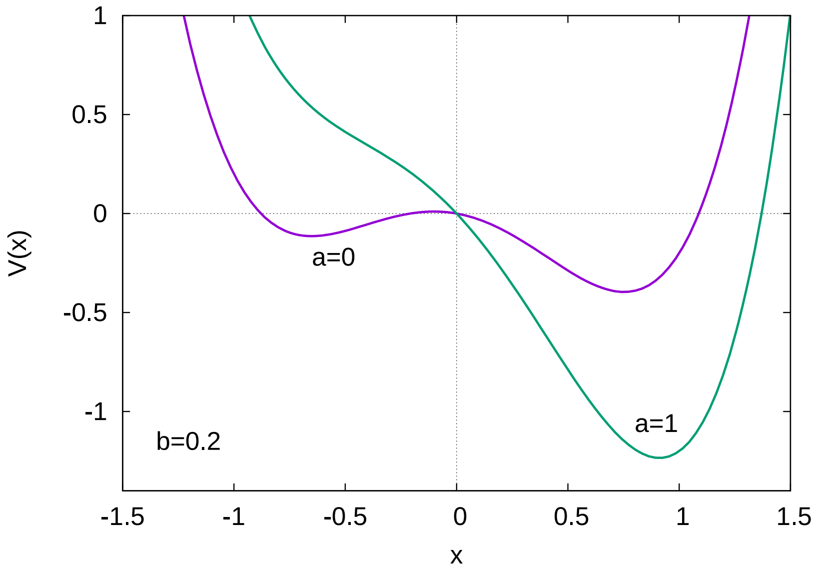


Figure 2.4: Effective potential for a single dynamical element i with $b = 0.2$ with $a_i = 0$ (violet) and $a_i = 1$ (green).

2.1 Study of Spatio Temporal dynamics

The work started with the study of spatio temporal patterns in a small world network of 100 nodes. We simulated the system governed by equation 2.1 for different parameter values. Specifically we consider a_i for N_1 sites and $a_i = 0$ for $N_0 = N - N_1$ sites. We study the strong coupling limit here, with $C = 1$ without any loss of generality. We vary parameter p, namely fraction of random links, from 0 to 1 and global bias b, from -1 to +1.

There is no synchronization when fraction of randoms links is zero (see Figure 2.5). This clearly indicates that when coupling is only restricted to neighbours the network does not synchronize. As fraction of random links increase, there are more shortcut paths for information flow, and global synchronization emerges.

We consider two classes of networks, with fraction p of random links:

1. *Static Networks* where the connections remain unchanged, namely the case of frozen or quenched links.

2. *Dynamic Networks* where the links switch at regular intervals of time, namely the connectivity matrix changes from time to time.

It was observed that synchronization occurred faster when the network has dynamic links rather than static links. The above observation was confirmed by plotting average value of nodes as a function of time ($\langle x \rangle$ vs time as displayed in Figure 2.8)

When there are no random links (i.e. $p = 0$) the system does not synchronize, while a system with a large fraction of random connections (e.g. $p = 0.8$) synchronizes, as is evident in Figures 2.6 and 2.7).

It is evident that dynamical links lead to faster approach to the steady state, namely shorter transience (cf. comparative times taken to reach the synchronized state in Figure 2.8).

As we observe in the next section (see Figure 2.12) that for certain value of N_1(where transition is taking place, i.e. between $N_1 = 0.05$ and $N_1 = 0.2$ in Figure 2.12) there is high dependence on initial conditions, we study the spacetime variation of x in detail for these parameter values(see Figures 2.9 and 2.10).

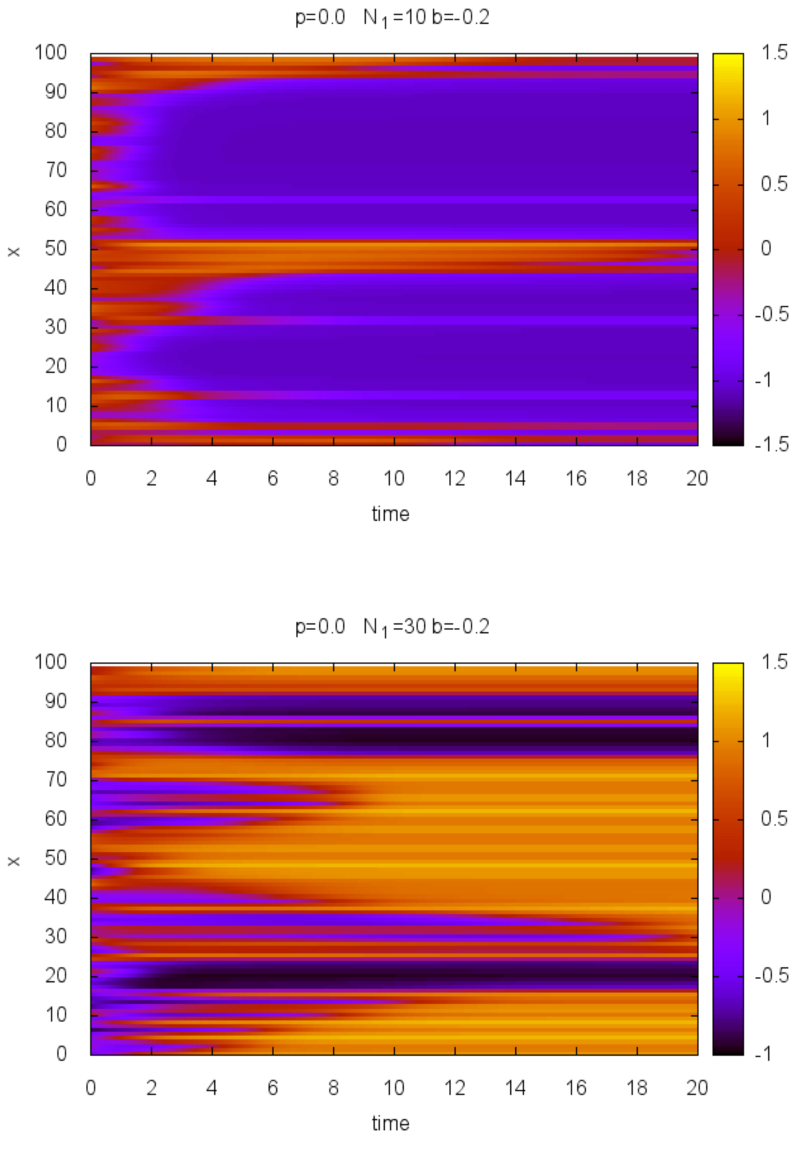
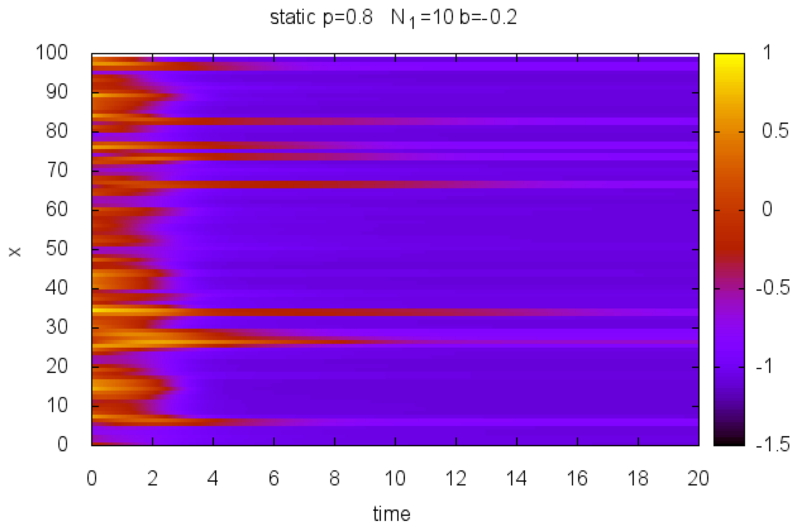


Figure 2.5: Time evolution of coupled bi-stable elements x^i $(i = 1, \ldots, N)$ governed by equation 2.1, where the number of sites with $a_i = 1$ is $N_1 = 10$ (up) and $N_1 = 30$ (down). Here we consider completely regular network without any random links, namely $p = 0$ and the network of size $N = 100$, global bias $b = -0.2$.



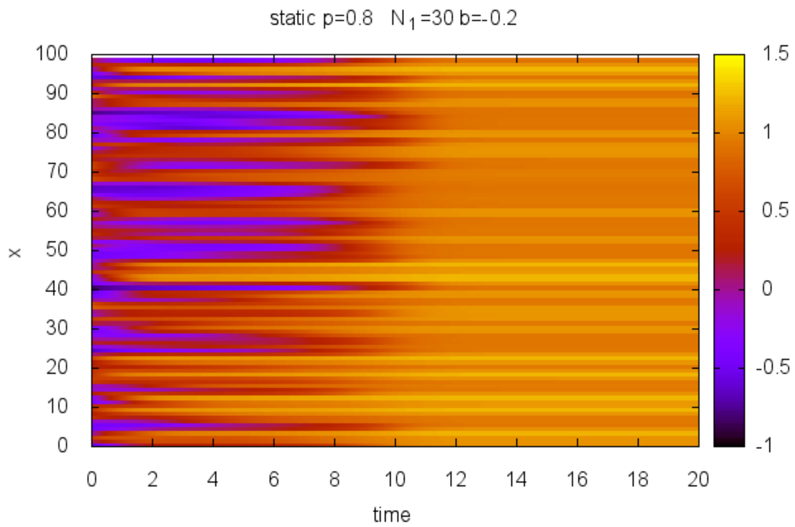


Figure 2.6: Time evolution of coupled bi-stable elements x^i $(i = 1, \ldots, N)$ in a network with quenched random links,governed by equation 2.1. Here the number of sites with $a_i = 1$ is $N_1 = 10$ (up) and $N_1 = 30$ (down) and the network of size $N = 100$, global bias $b = -0.2$, and the fraction of random links $p = 0.8$.

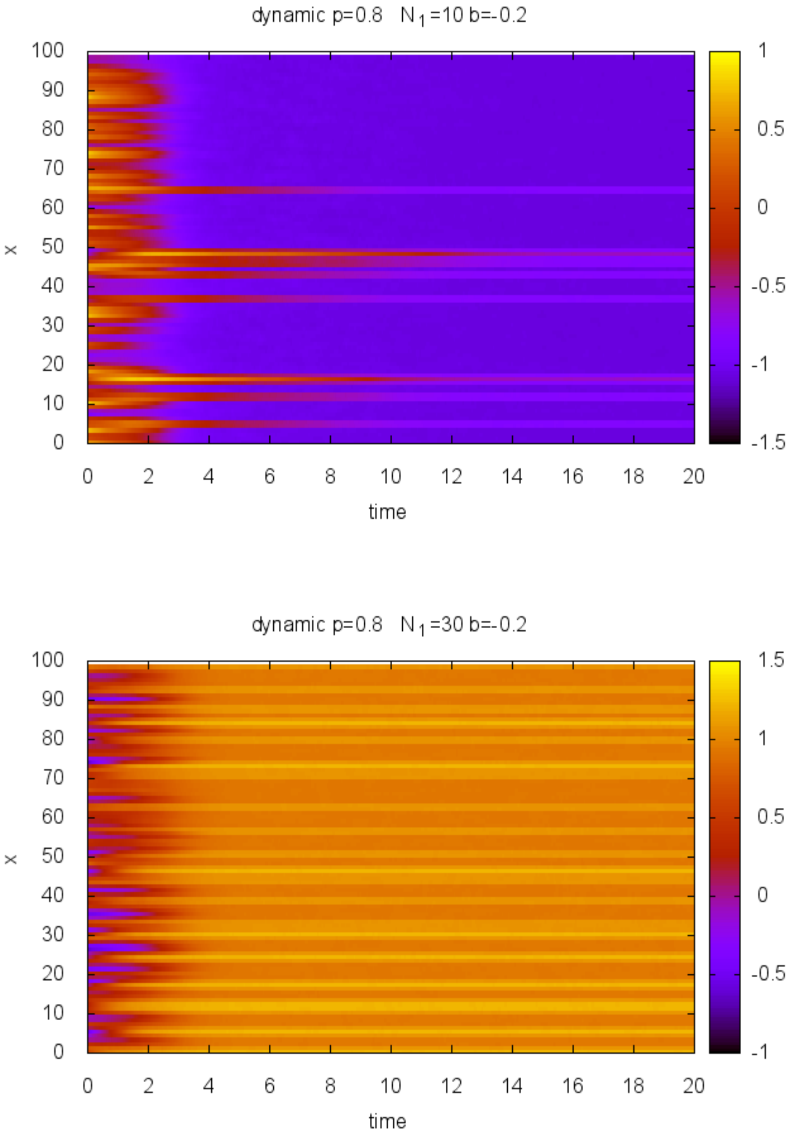


Figure 2.7: Time evolution of coupled bi-stable elements x^i $(i = 1, \dots, N)$ in a dynamically rewired network, where the number of sites with $a_i = 1$ is $N_1 = 10$ (up) and $N_1 = 30$ (down). Here the network of size $N = 100$, global bias $b = -0.2$, and the fraction of random links $p = 0.8$.

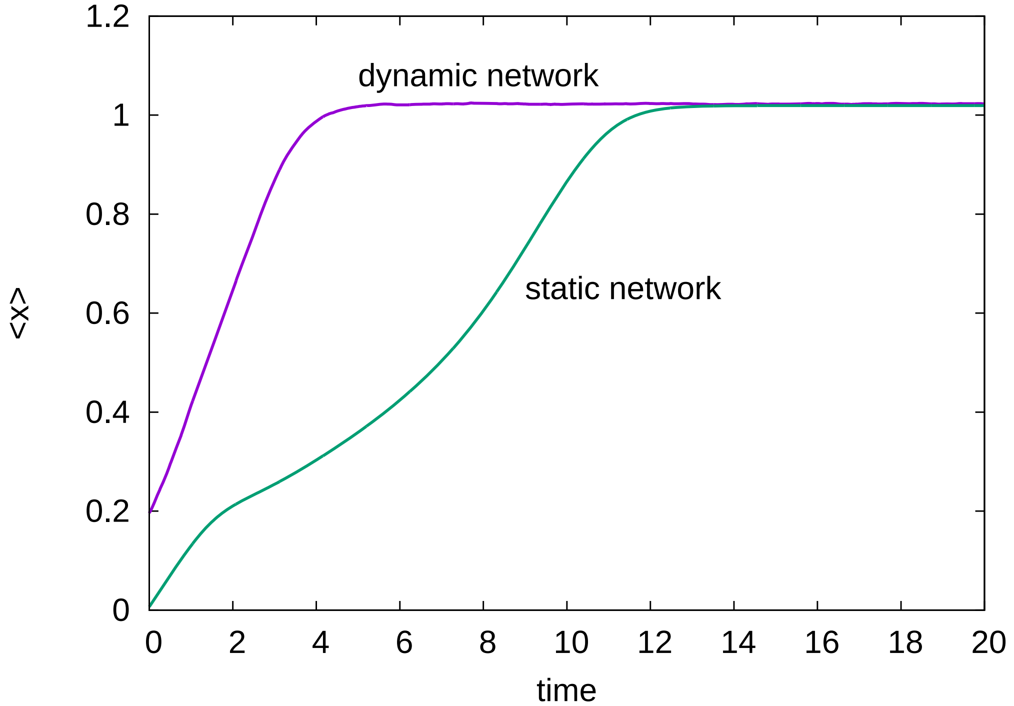


Figure 2.8: Time evolution of the collective field $\Sigma_x x_n^i$. The violet curve is the dynamic rewiring case and the green curve is for a static network. Here the system size is $N = 100$, global bias $b = -0.2$, and the fraction of random links $p = 0.8$, number of sites with $a_i = 1$ is $N_1 = 30$

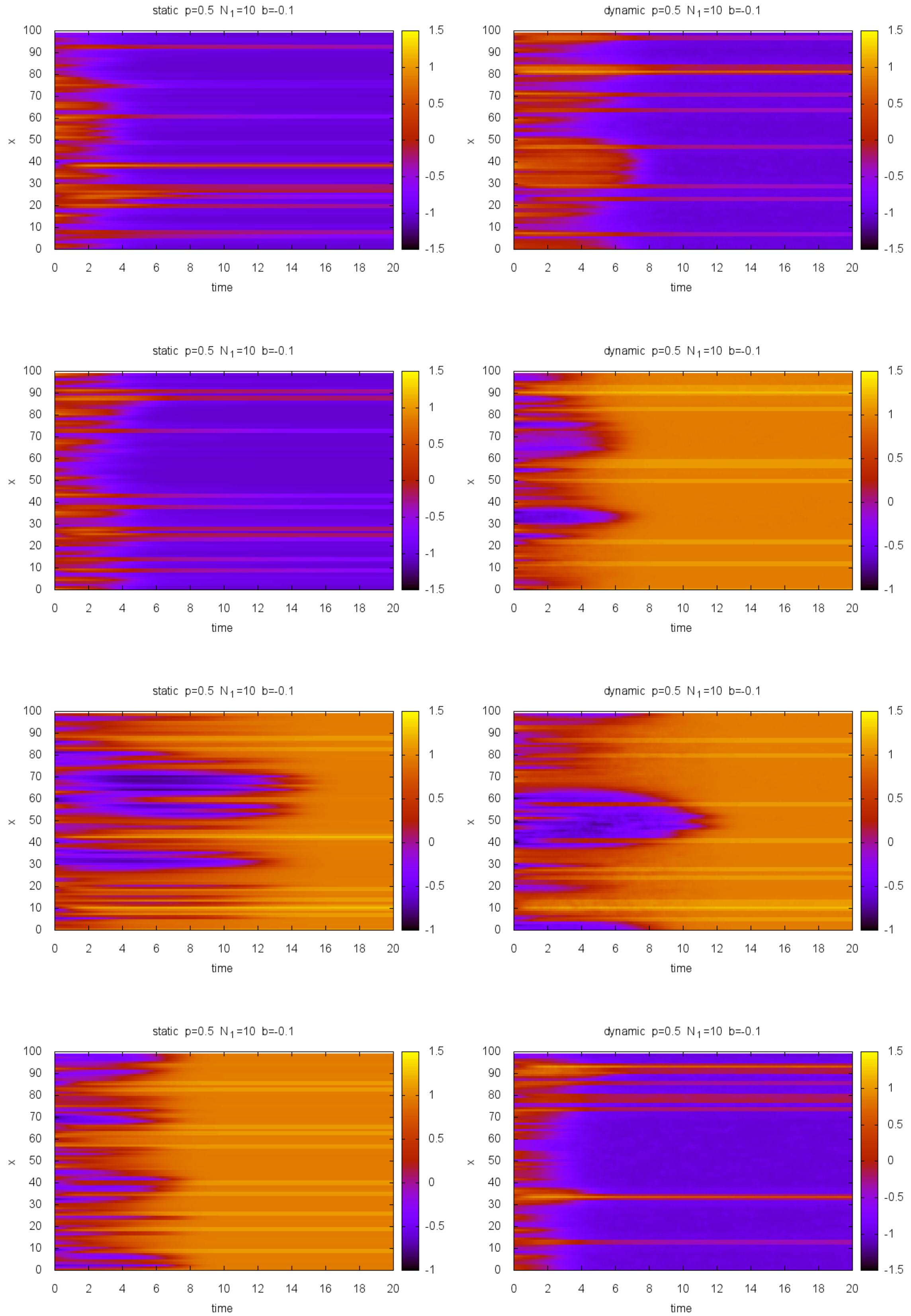


Figure 2.9: Field in a static (left) network and dynamic (right) network for different random realizations with $N_1 = 10$ sites with $a_i = 1$, global bias $b = -0.1$, fraction of random links $p = 0.5$ and system size $N = 100$. Here

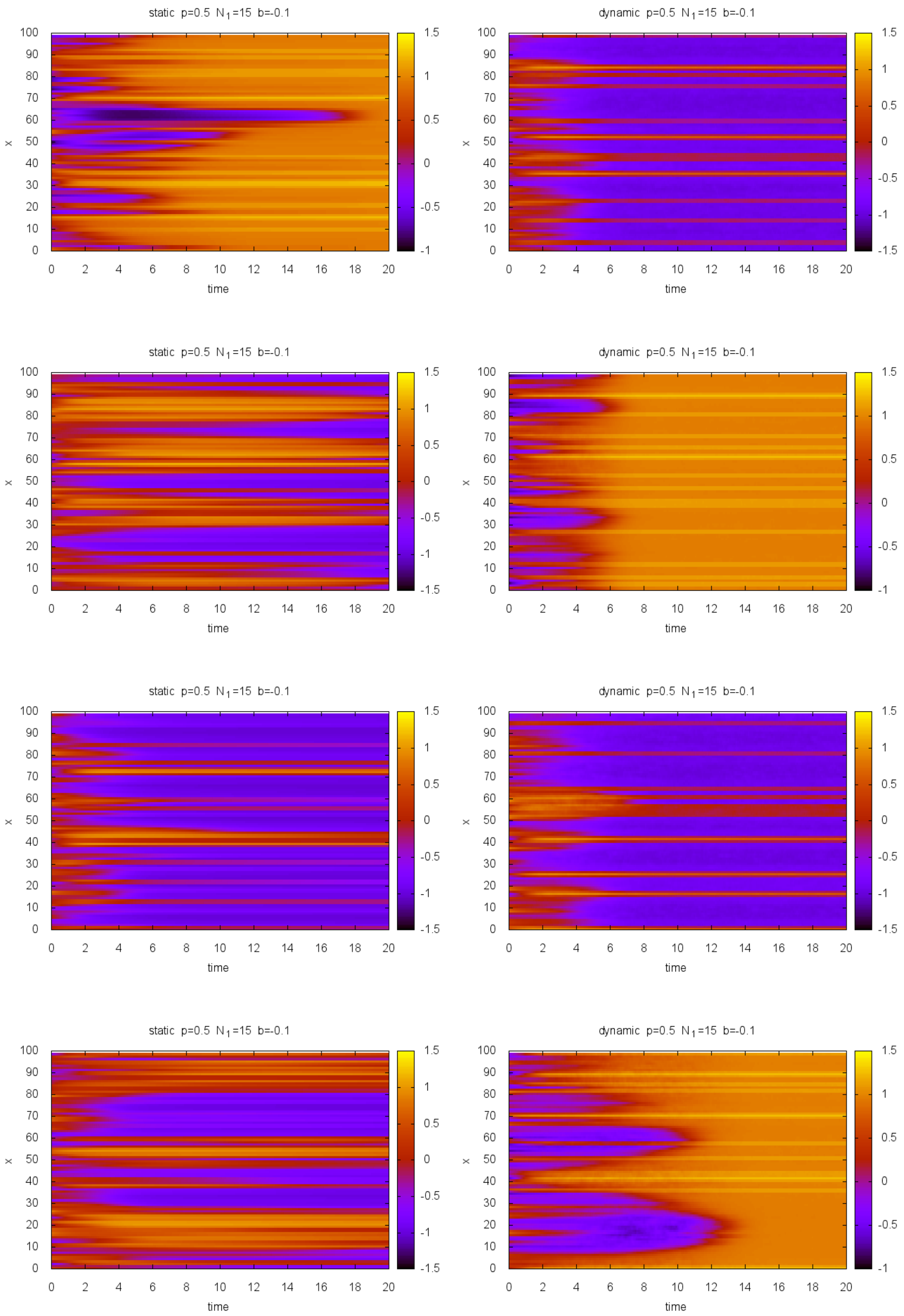


Figure 2.10: Field in a static (left) network and dynamic (right) network for different random realizations with $N_1 = 15$ sites with $a_i = 1$, global bias $b = -0.1$, fraction of random links $p = 0.5$ and system size $N = 100$

2.2 Study of average value of nodes vs N_1

We examine how collective features such as the mean field $\langle x \rangle$ is affected by the majority and the minority elements, namely by the relative magnitudes of N_0 and N_1, in the heterogeneous networks.

1. **Effect of varying global bias :**

   First we vary the global bias b of the system and examine its effect on the transitions in the collective field with respect to changing fraction N_1/N. Note that the global bias changes the nature of the effective local potential at each node. The result of the simulations are represented in Figures 2.11 - 2.14.

   If global bias is positive(i.e. $b > 0$), then we can observe in Figure 2.4 that the nodes with $a_i = 0$ have more stable well on positive side and less stable one on negative side. While for $a_i = 1$ the only stable well is on the positive side. Thus with increasing N_1, more and more nodes are pulled to the stable state on positive side and as shown in Figure 2.11 we get direct proportionality of increment of $\langle x \rangle$ with increasing N_1.

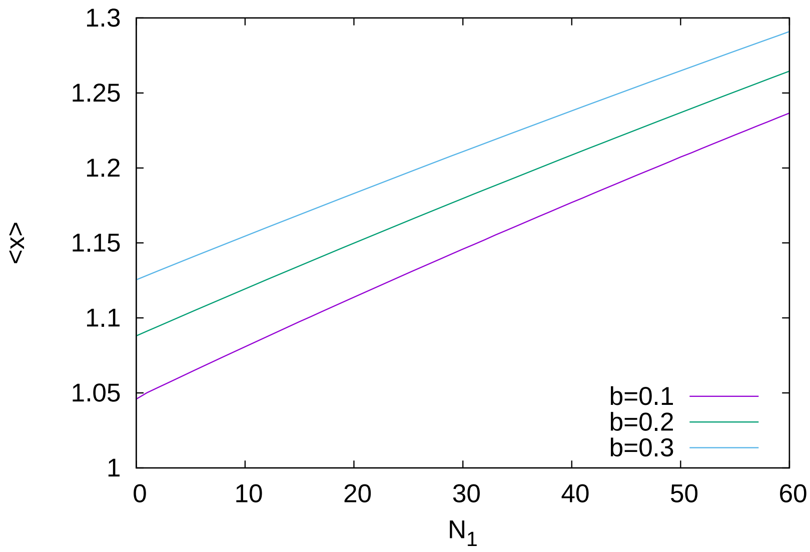


Figure 2.11: Collective field in a static network averaged over 100 random realizations as a function of N_1 (number of sites with $a_i = 1$) for global bias $b = 0.1, 0.2, 0.3$, fraction of random links $p = 0.3$ and system size $N = 100$

On the other hand when global bias is negative, then nodes with $a_i = 0$ has more stable node on negative side of x axis than positive side of x axis(see figure 2.2). While nodes with $a_i = 1$ still has the only stable node on the positive side of x axis. So if there is no coupling, the nodes with $a_i = 0$ would preferentially end up on negative side while the ones with $a_i = 1$ would end up on positive side of x-axis after the transients are over.

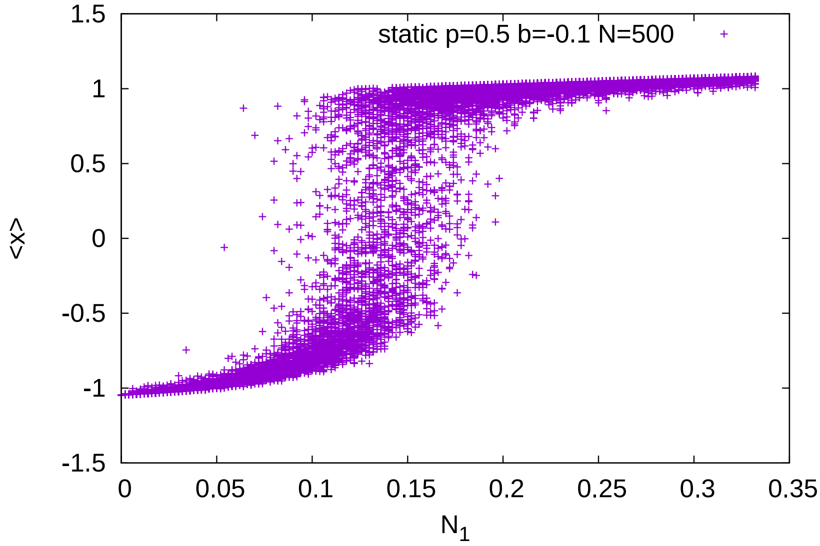


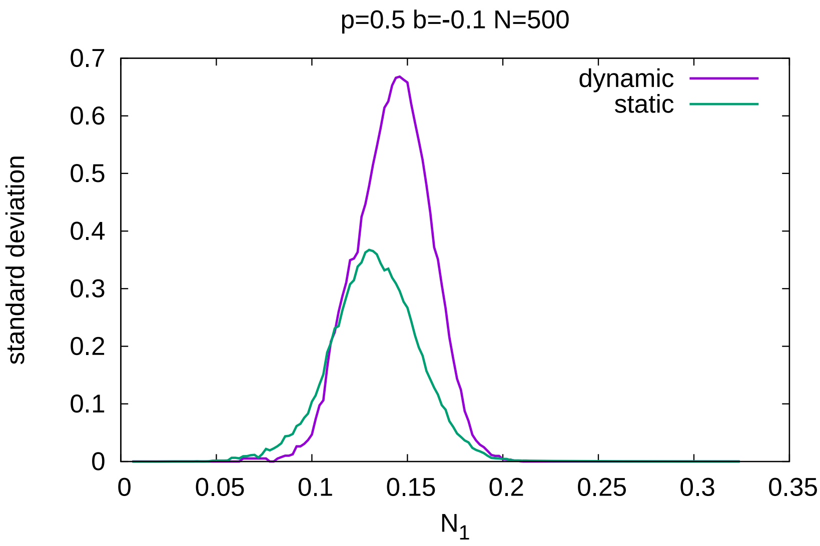
Figure 2.12: Collective field in a static network for 200 random realizations as a function of N_1 (number of sites with $a_i = 1$) for global bias $b = -0.1$, fraction of random links $p = 0.3$ and system size $N = 500$

So, when the network is evolving with equation 2.1, the effect of coupling comes into play and pulls the nodes with $a_i = 1$ from negative side to positive side of x axis(as observed in Fig 2.6 and 2.7).

When we study the $\langle x \rangle$ vs. N_1 we see that the steady state of $\langle x \rangle$ is almost constant for very low values of N_1 and very high values of N_1, but the transition values show high dependence on initial condition. Figure 2.12 shows the spread of $\langle x \rangle$ for different initial condition.

We also studied the standard deviation of $\langle x \rangle$(average value of sites after complete time evolution) averaging over 200 initial condition in order to investigate if there is any difference in static and dynamic networks during transition. The results do not reflect very convincing dependence on the type of network namely static and dynamic(Figure. 2.13).

It is evident that with negative global bias very close to zero ($b \to 0^-$) the transition shifts to $N_1 \to 1$ (Figs.2.14), indicating hypersensitivity of the system. Note that $N_1 = 1$ indicates a situation where the collective field reflects the presence of even a single different element in the network.



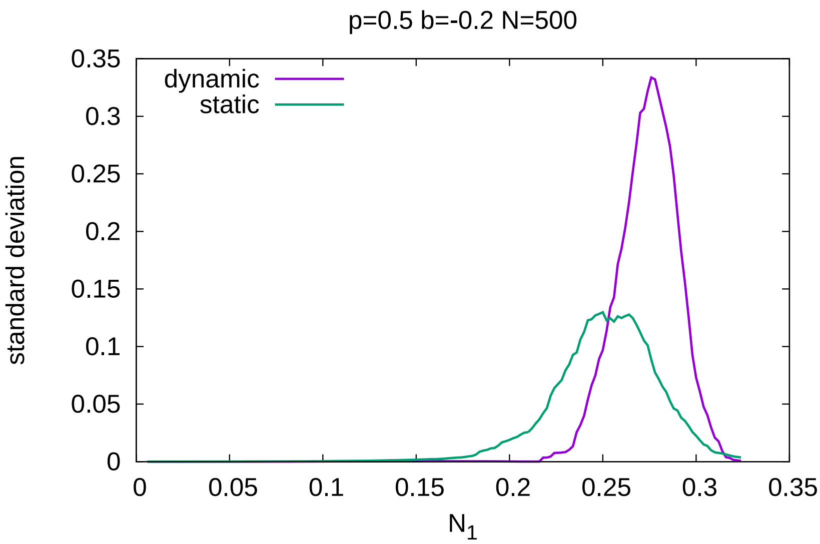
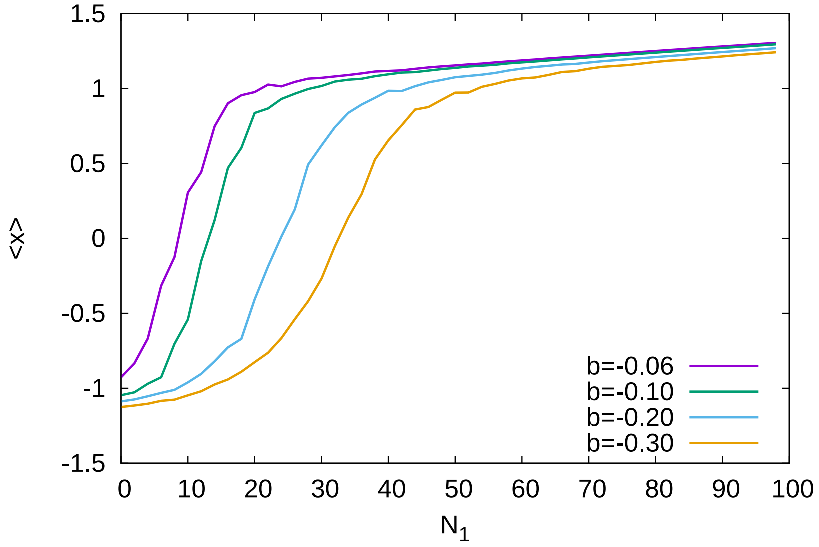


Figure 2.13: Comparison of Standard deviation of average field after time evolution for static network vs dynamic network with respect to changing N_1. Here fraction of nodes having random neighbors is 50% (i.e. $p = 0.5$), number of nodes $N = 500$ and global bias is $-0.1(up)$, $-0.2(down)$



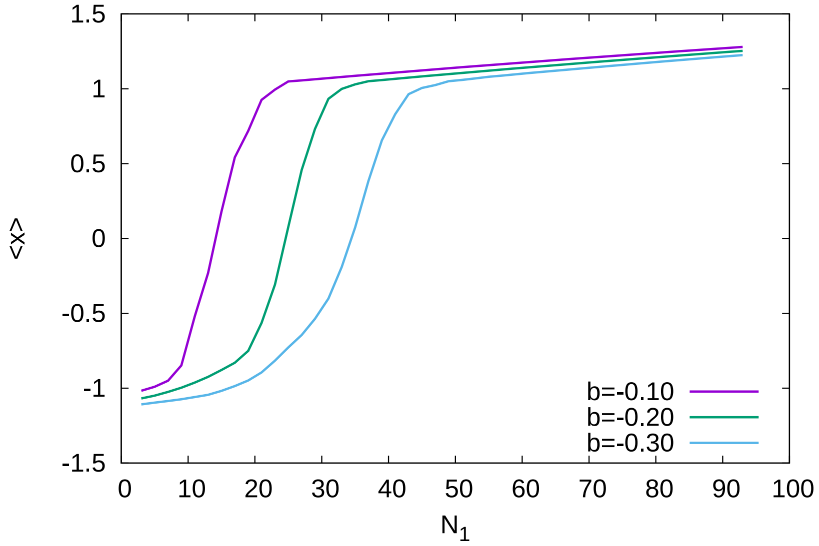


Figure 2.14: Collective field in a network (up: static, averaged over 100 random realizations; down: dynamic, averaged over 10 random realizations), as a function of N_1 (number of sites with $a_i = 1$), for different values of global bias. Here network size $N = 100$, fraction of random links $p = 0.3$.

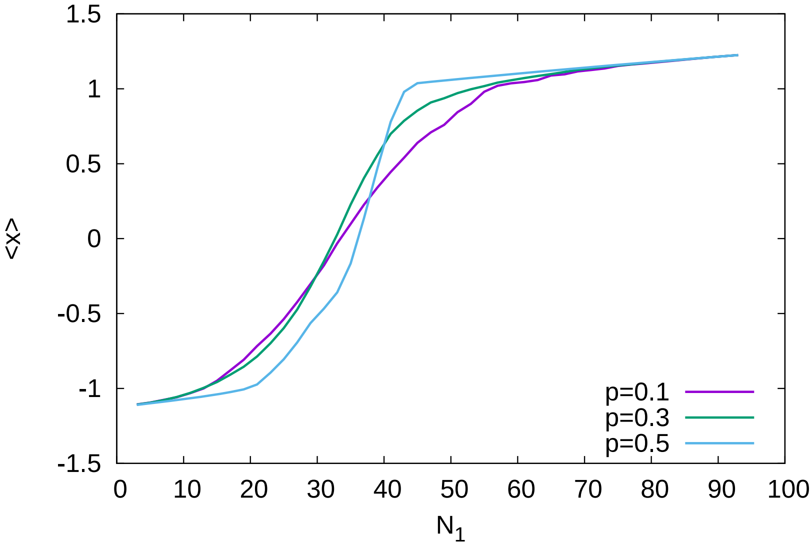
2. **Effect of varying fraction of random links:**

   Further, the fraction of random links p in the system is varied and its effect on the sensitivity of the collective field to N_1 (number of nodes with $a_i = 1$) is studied.

   Figure 2.15 represent the result obtained. For a fixed negative b value, with increasing probability of a link to be random (i.e. increasing number of sites with random connections) the transition becomes sharper, indicating better synchronization.

3. **Effect of varying system size:**

   Larger system sizes give rise to sharper transitions with respect to N_1, as seen in Figure 2.16.



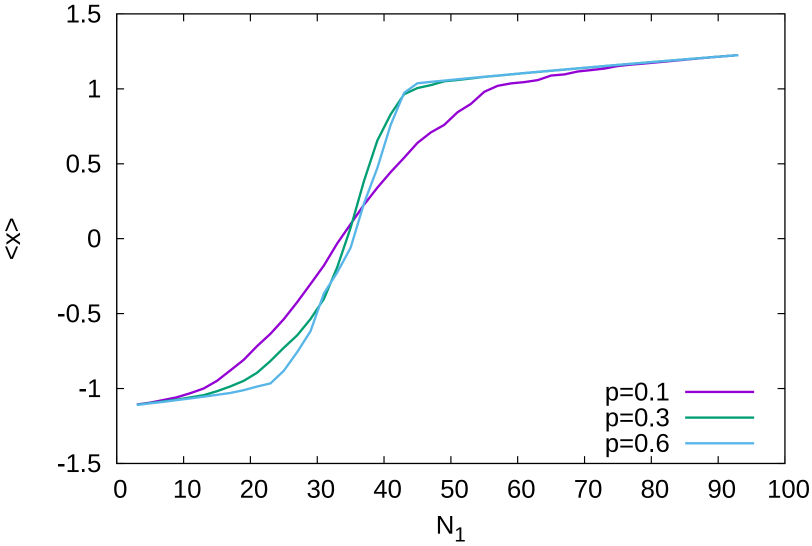


Figure 2.15: Collective field in a network (up: static, averaged over 100 random realizations; down: dynamic, averaged over 10 random realizations), as a function of N_1 (number of sites with $a_i = 1$), for different values of the fraction of random links. Here network size $N = 100$, fraction of random links $b = -0.3$.

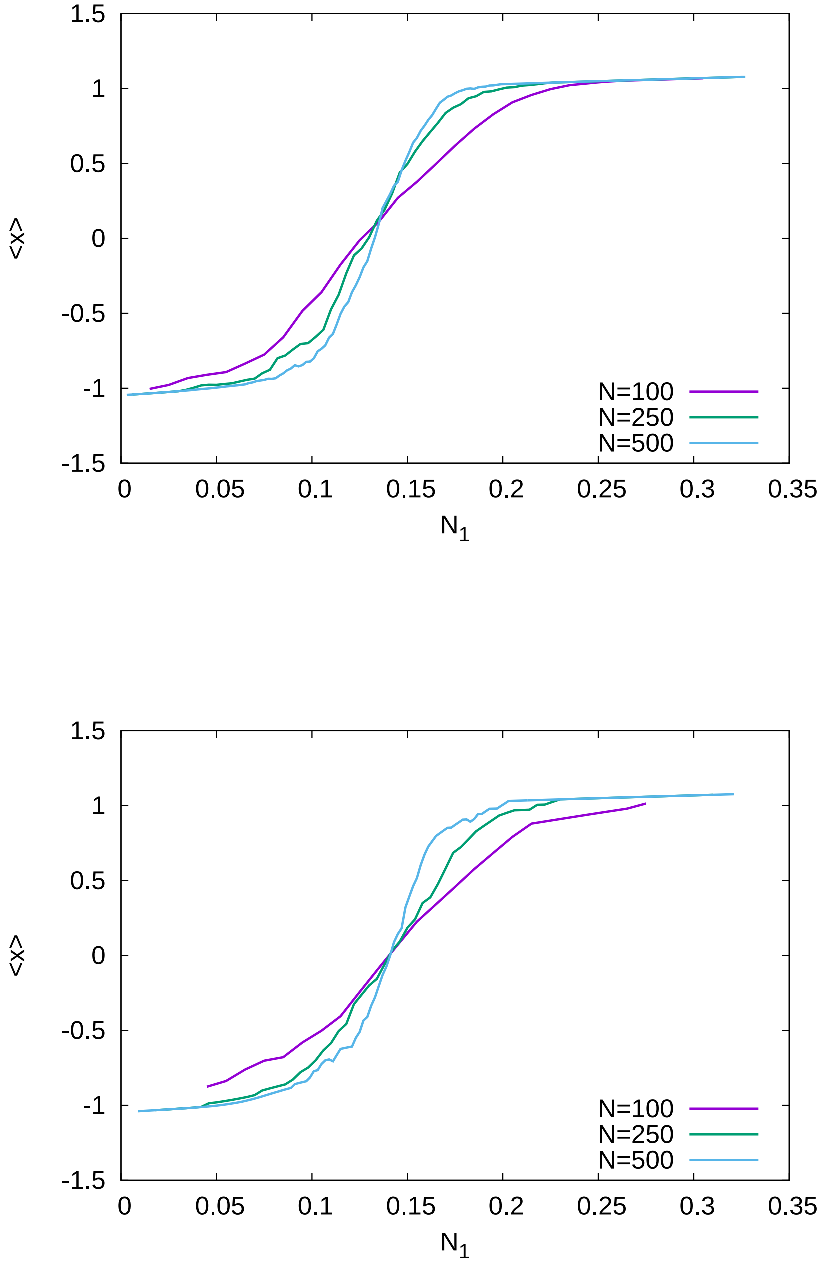


Figure 2.16: Collective field in a network (up: static, averaged over 100 random realizations; down: dynamic, averaged over 10 random realizations), as a function of N_1 (number of sites with $a_i = 1$), for different values of system size. Here network size $p = 0.8$ and global bias $b = -0.3$.

3. Conclusions

To sum up, the significant results obtained are the following:

- Dynamic networks show faster transition than static network in time.
- Decreasing the negative global bias increases the sensitivity of network towards heterogeneity.
- Dynamic networks and static networks do not show appreciable difference in standard deviation of average value of nodes with respect to heterogeneity.
- Increasing the fraction of random links makes the transition sharper, indicating further sensitivity to heterogeneity.
- Increasing the size of system also makes the transition sharper, suggesting that the relative size of coupled elements to the size of system does not influence the systems sensitivity to heterogeneity.

In conclusion, even though the system has such small coupling range (i.e. $k = 4$) which is much smaller than the global limit ($k = N$), it still yields ultra-sensitivity to heterogeneity. We deduce this from investigations of the swing in the collection field as a function of global bias under varying sizes of the minority population. Further, increasing system size makes the transition sharper with the swing occurring at $N_1/N \to 0$ for $b \longrightarrow 0^-$. This also show that the relative size of the couple elements to the size of the system does not influence the ability of the system to respond to small non-uniformity.

Future directions

With the above mentioned results, further interesting investigations can be followed. Some of them are the following:

- How would the system behave if we change the function $G(x)$ in equation 2.1, i.e. how general is the bi-stable potential that we have considered in out work.
- Effect of heterogeneity on a system coupled to two neighbors and then to one neighbor(the limiting case). This will help us gauge the minimum coupling necessary to obtain ultra-sensitivity to non-uniformity in the constituent elements of the coupled system.

A. Programs

The section contains the programs that were used for generating the results shown everywhere. The program consist of two files : one containing the Network class which contains all the variables and functions needed to operate on network, another file imports this file and creates its objects to generate data and get various plots like space-time plot or $\langle x \rangle vs N_1$ plots.

The variables(or identifiers) used throughout the programs are following:

```
#--------Parameters---------
n #No of nodes
n1 #No of sites with a0=1

p #Probablity of random links
c #Coupling constant
b #Global bias

k #No of neighbors for regular sites
kr #No of neighbors for random sites
et #evolution time
dt #Time step size


sle #steps of link evolution for dynamic network
#--------------------------
```

1. The "classes.py" file containing class:

```
from random import *

class Nodes:
    def __init__(self,x=0.0, Dx=0.0, a=0.0, nb=[]):
        self.x = x
        self.Dx = Dx
        self.a = a
        self.nb = nb

    def details(self):
        s= ’a=%s␣␣␣␣nb␣-\t’%self.a
        for i in self.nb:
            s+= str(i) + ’\t’
        return s

class Network:
    # Initializing
    def __init__( self, n=10,n0=2, k=1,kr=2,p=0.0 ):
        self.node = [Nodes( x=(2*random()-1) ) for i in range(n)]
        for i in sample( range(n), n0 ):
            self.node[i].a = 1.0
        self.evolveLinks(p=p,k=k,kr=kr)

    #function to evolve links
    def evolveLinks(self, p=0.0 ,k=1, kr=2):
        n=len(self.node)
        for i in range( len(self.node) ):
            r=random()
            if r>p:
                #regular neighbours
                ion=range(-k,k+1) #ion - index of neighbours
                ion.remove(0)
                index=[]
                for j in ion:
                    index+= [((i+j)%n) ]
                self.node[i].nb=index
            else:
                #site with random neighbours
                s= range(n)
                s.remove(i)
                s=sample(s,kr)
                self.node[i].nb=s

    #function to evolve sites
    def evolveNodes(self, dt=0.01,c=1,b=0.1 ):
        #calculating derivatives at sites
        for i in range( len(self.node) ):
```

```
        nf=0.0
        for nbr in self.node[i].nb :
            nf=nf+ self.node[nbr].x
        nf=nf/ len(self.node[i].nb)
        self.node[i].Dx = (  self.node[i].x - \
                           self.node[i].x**3.0 ) + \
                          c*( nf - self.node[i].x ) + \
                          b+ self.node[i].a
    #evolving sites
    for i in range( len(self.node) ):
        self.node[i].x += self.node[i].Dx * dt

#function to return average value at nodes
def avgx():
    avg=0.0
    for i in range( len(self.node) ):
        avg += self.node[i].x
    avg = avg/len(self.node)
    return s
```

2. Spacetime data is produced by calling the following functions with proper argument

```
from classes import *

def dynamic_spacetime( n,n0, k,kr, p,b,c, dt,et,sle):
    filename='dynamic␣p=%s␣n1=%s␣b=%s.txt'%(p,n0,b)
    f = open(filename,'w')
    swn = Network(n=n, n0=n0, k=k, kr=kr, p=p)

    t=0.0
    while t<=et:
        if (int(t/dt))%sle==0:
            swn.evolveLinks(p=p,k=k, kr=kr)
        for i in range(n):
            f.write( str(t) +'\t'+str(i) +'\t'+ \
                     str(swn.node[i].x) +'\n' )
        f.write('\n')

        swn.evolveNodes(dt,c,b)
        t+=dt
    f.close()
    return filename


def static_spacetime( n,n0, k,kr, p,b,c, dt,et):
    filename='static␣p=%s␣n1=%s␣b=%s.txt'%(p,n0,b)
    f = open(filename,'w')
    swn = Network(n=n, n0=n0, k=k, kr=kr, p=p)

    t=0.0
    while t<=et:
        for i in range(n):
            f.write( str(t) +'\t'+str(i) + \
                     '\t'+str(swn.node[i].x) +'\n' )
        f.write('\n')
        swn.evolveNodes(dt,c,b)
        t+=dt
    f.close()
    return filename
```

3. Function to generate data for $\langle x \rangle$ vs N_1 is following :

   (a) For Static Networks:

```
from classes import *

#-----------------------------------
n = 10 #Total nodes
n0Range = [i for i in range(0,n)]
#-----------------------------------

#function to create one network and
#return average after time evolution
def main(n0, p, c, b ):
    #--------Parameters---------
    els= 10 #Evolution link steps
    k = 2 #No of neighbors for regular sites
    kr = 2*k #No of neighbors for random sites
    et = 2.0 #evolution time
    dt = 0.01 #Time step size
    lp = 100 #No of last time steps
    #    over which avgerage is taken
    #--------------------------
    lt = et-lp*dt

    swn = Network(n=n, n0=n0, k=k, kr=kr, p=p)
    avls=0.0

    i=0
    t=0.0
    while t<=et:
        swn.evolveNodes(dt,c,b)
        if t>lt:
            avls += swn.avgx()
        t+=dt
        i+=1
    avls = avls/lp
    return avls


count=0
#function to produce average values file for n0 vs x
def n0vsX( p, c, b, avgfn ):
    global count

    if avgfn == 0:
        avgfn='average␣b=%0.2f␣p=%0.1f␣c=%0.1f'%(b,p,c)
    avfile = open(avgfn+'.txt','w')
```

```
for n0 in n0Range:
    avls = main(n0=n0, p=p , c=c, b=b)
    avfile.write( '%s\t%s\n'%(n0,avls) )

    #----displaying progress----
    print "\r␣count=%s␣\t␣p=%0.2f␣\t␣b=%0.2f␣\t␣n0=%s␣" \
          %(count,p,b,n0),
    count+=1
    #--------------------------
avfile.close()
```

(b) For Dynamic Networks:

```
from classes import *

#-----------------------------------
n = 10 #Total nodes
n0Range = [i for i in range(0,n)]
#-----------------------------------

#function to create one network and
#return average after time evolution
def main(n0, p, c, b ):
    #--------Parameters---------
    els= 10 #Evolution link steps
    k = 2 #No of neighbors for regular sites
    kr = 2*k #No of neighbors for random sites
    et = 2.0 #evolution time
    dt = 0.01 #Time step size
    lp = 100 #No of last time steps
    #    over which avgerage is taken
    #---------------------------
    lt = et-lp*dt

    swn = Network(n=n, n0=n0, k=k, kr=kr, p=p)
    avls=0.0

    i=0
    t=0.0
    while t<=et:
        if i%els ==0:
            swn.evolveLinks(p ,k , kr)
        swn.evolveNodes(dt,c,b)
        if t>lt:
            avls += swn.avgx()
        t+=dt
        i+=1
    avls = avls/lp
    return avls


count=0
#function to produce average values file for n0 vs x
def n0vsX( p, c, b, avgfn ):
    global count

    if avgfn == 0:
        avgfn='average b=%0.2f p=%0.1f c=%0.1f'%(b,p,c)
    avfile = open(avgfn+'.txt','w')
```

```
for n0 in n0Range:
    avls = main(n0=n0, p=p , c=c, b=b)
    avfile.write( '%s\t%s\n'%(n0,avls) )

    #----displaying progress----
    print "\r count=%s \t p=%0.2f \t b=%0.2f \t n0=%s "\
         %(count,p,b,n0),
    count+=1
    #---------------------------
avfile.close()
```

The data produced with above programs are not averaged over initial condition. Thus to reproduce the graphs presented in this literature one would have to run the program several times and average the data produced.

Bibliography

[1] Kaneko, Kunihiko. "Overview of Coupled Map Lattices." *Chaos 2*, **3**(1992) 279.

[2] D J Watts and S H Strogatz, *Nature*, **393**(1998) 440-2

[3] Sudeshna Sinha, Rajeev Kapri and Kamal P. Singh, *EPL*, **98**(2012) 60004

[4] A Arenas, A Daz-Guilera, J Kurths, Y Moreno, C Zhou. "Synchronization in complex networks" *Physics Reports*, **469, 3**(2008)

[5] A. Pikovsky, M. Rosenblum, & J Kurths "Synchronization" *Cambridge University Press, Cambridge* , UK (2001)

[6] Albert, R; Barabasi, A-L. "Statistical mechanics of complex networks" *Reviews of Modern Physics*, **74**(2002)4797.

[7] Steven Strogatz. "Nonlinear dynamics and Chaos" *ISBN*, **978-0-201-54344-5** (1994) *