臺灣二ＯＯ六年國際科學展覽會

科　　　別：物理科

作 品 名 稱 ：同步現象的研究

得 獎 獎 項 ：佳作

學校 / 作者 ：國立臺灣師範大學附屬高級中學　章學元
國立臺灣師範大學附屬高級中學　劉傑瑋

章學元，師大附中 1086 班學生，在高一的時候參與國科會高中基礎科學資優人才培育計畫，備選爲中研院胡進錕教授的輔導學生。從一年級下學期開始每個禮拜四下午三堂專題研究的時間，都準時到教授的會議室報到。從一開始對教授的研究一知半解，大部分的時間都花在閱讀書籍以及中研院一些實驗報告，到後來也慢慢開始試著自己寫程式。原本對程式也不怎麼在行，遇到問題的時候都到書店去查程式語法書，在不斷地摸索中，終於可以寫出符合自己實驗的程式。過程是漫長且辛苦，卻充滿樂趣的。經過五六個月的接觸統計物理，與嘗試實驗，我們最後才選定了我們的研究主題—同步現象的研究。

除了同步的模型，在中研院統計物理所，我們還接觸了許多不同的模型，大開眼界，像是沙粒落下的模型、晶格連結模型、蛋白質旋轉模型、螺絲模型……其中印象最深，最有心得的就是 Link 提到的 small word。這些都是學校課程中無法學到的，是很棒的收穫，一種別於一般的研究方式。

我們整個實驗都是在用電腦跑數據，論文上的圖表也都是數據和斜率，花了很大的精神才了解得到的實驗結果與意義，懂了之後很有成就感。我們的工具就只有一台筆記電腦和兩台桌上型的舊電腦，好在性能不錯，運算一次兩百個點兩千個時刻，只需要 12 分鐘……有時候一個小時過去只計算了一組數據，可是我們必須做 10-20 組取平均。做實驗有時候覺得很枯燥，因爲程式寫好了，運算只需要按兩三個按鈕，其他的都是等待。整個實驗過程，最讓人回味的就是寫程式的過程了。不斷的測試與等待，十分鐘過去出來的卻都是錯誤數據，十分耐人尋味。總算是有了小小的成果，在這整整兩年的時間……在課業與實驗之間……默默等待。

劉傑瑋，目前就讀師大附中，1087 班學生，在高一的時候參與國科會高中基礎科學資優人才培育計畫，備選爲中研院胡進錕教授的輔導學生。從一年級下學期開始每個禮拜四下午三堂專題研究的時間，都準時到教授的會議室報到。一開始我們是對教授的研究一知半解，大部分的時間都花在閱讀書籍以及中研院一些實驗報告，到了後來就常是去沿著前人的研究繼續的作下去。經過了半年的接觸統計物理，與嘗試實驗，我們最後才選定了我們的研究主題—同步現象的研究。這些過程是漫長但有趣的。我們整個實驗都是在用電腦跑數據，論文上的圖表也都是數據和斜率，花了很大的精神才了解實驗結果與意義，懂了之後很有成就感。

Abstract

Synchronized/Simultaneous Research

In our daily life, objects and the contacts between objects they will have mutually affect each other, some initially chaotic systems after a sufficient amount of time will mutually correct each other, and finally achieve synchronization (example: the speed of bird and fish migration, market prices, infantry…), although some are unable to achieve this. We will illustrate and explain the synchronization system, its process and discover the conditions for synchronization. Using linking concepts, we will integrate the coupled map lattices with global coupling and coupled map lattices with intermediate-range models into a synchronization mode in order to simulate a synchronization system.

We first used a small system of $n \leqq 50$ to obtain results that will demonstrate the linking concepts:

1. The more chaotic a system, a longer period of time is required for synchronization.
2. An increase in the number of individual objects requires an increase in the range of concepts and the amount of time in order to achieve an in depth synchronization.
3. Initial concept values which randomly effect synchronization critical point conditions are not obvious in a mathematically incorrect graph. In a closer look, when we increased the synchronization to $n \leqq 400$ and the number of times to t-->100,000 we discovered:

1. Using the function G(x) we hoped the results from the graph after apply the function and correction able to overlap and test with “Scaling and Universality in Transition to Synchronous Chaos with Local-Global Interactions”, but the part which overlapped the measurements was not identical:
2. We can use the significance of the critical point and the Interactive Process to find the approximate value of the critical value up to 4 digits following the decimal point.
3. We can also use the approximate value to find out the range for the simultaneous conditions and the various points on the system itself, as well as obtain a negative correlation between them, and then it can be similarly expressed with using a curve.

A computer can calculate values with this kind of enumerating method, even without any special resolution capabilities to quickly obtain large amounts of approximate values of simultaneous conditions, this is especially true when calculating unfamiliar systems.

指導教授：
國立台灣師大附中

中文摘要

同步現象的研究

日常生活中，物件與物件的接觸，彼此會互相影響，有些原本雜亂的系統再經過充裕時間的互相修正後，最後竟能達成同步(例如：鳥群、魚群遷徙的速度、市場價格、行軍步伐…)，有些則不能。因此，我們試著利用描述同步系統的模型，觀察系統同步的過程，並且找出同步的條件。由連結的觀點，我們將 Coupled map lattices with global coupling 和 Coupled map lattices with intermediate-range 模型的優點整合成 Synchronization mode 去模擬同步系統。

我們先用小系統(n≦50)得到能印證連結觀點的結果：

(一)、系統越雜亂，就需要稍長的時間同步；

(二)、個體數越多時，各點需要更大範圍的點數去影響於每單位時間內以及更深的影響才能同步；

(三)、起始值隨機影響同步臨界條件並不明顯，在誤差範圍內。

更進一步，我們將系統推向 n≦400 點,t→100,000 次，我們發現：

(一)、在”G(x)”我們希望能將圖形經過函數修正之後能疊和，驗證”Scaling and Universality In Transition to Synchronous Chaos with Local-Global Interactions ”中的結果，但只有部分疊和，尺度不相同；

(二)、可以直接利用臨界點的意義用十分逼近法求出臨界值的近似值到小數後四位；

(三)、我們用近似值也能發現同步條件與系統各點本身可跳躍的數值範圍是負相關，可用曲線去近似。

這種窮舉方式，交由電腦運算，不需要特別的解析能力就能夠快速且大量求得同步條件的近似值，尤其在運算不熟悉的系統時。

壹、前言

『將許多自然界的現象：生物、物理、社會，轉換成模型進行研究。』這是我們從 Link 這本書上得到的有趣點子。將個體化成點，彼此間的影響以線表示，彼此相連，或是有其他更複雜的形式。如果要將一個社會結構轉換成模型，則將不同個體的人化爲不同的點彼此以線連接，接著考慮影響目標物的因素以及想要觀察的模型特色，找出其中點與點之間造成影響的函數，寫成程式，即可進行模擬。在這樣隨機的分配下有許多細節雖未加以詳細考慮(例如系統中每個人的年齡、性別、職業)，卻能不影響整體結果，這也是爲什麼此研究方式可以解決各式複雜問題。

我們最早的想法是利用這樣的方法去研究同步現象，找出達成同步的條件，後來也對同步模型做些深入的討論。了解方程式 $f(\chi_{t+1})=4\lambda\chi_t(1-\chi_t)$ 以及 $f(\chi_{t+1})=1-a\chi_t^2$ 兩者相同、異同之處。這些函數是我們從論文上看到已大家都討論過的，我們希望能夠以我們自己的方法，還有目前的能力去嘗試驗證。我們也將一些不同的研究方式做整合，希望能作更深入的討論。

會選擇研究同步現象，則是看到一篇關於螢火蟲發光同步的文章，引起我們的興趣。雖然沒辦法親眼看到螢火蟲是否真的有同步的現象，但是我們想對同步模型進行研究。

貳、研究方法

我們討論同步現象，是用數值去描述個體的狀態。起始的時候用亂數給每個點一個起始值，爲了之後電腦運算方便，我們將範圍設定在[-1,1]所有實數，遞迴函數的值域也不會超過這個範圍。每個數值都代表一個狀態，在電腦上，如果兩數之差小於 10^{-8} 就可以視爲相等了。如果兩個點的數值跑程式經過一段時間的回饋能相等了，再繼續跑函數出來的值也會相同，所以我們就可以說他們達到同步。因此我們就利用程式，可以討論同步的快慢，以及執行初所給的變數條件，對同步的影響。

我們選的 f(x)內函數不能是收斂的狀態，不然即使沒有經過同步也會收斂在同一數值上，無法判定是否是受到同步模型的影響。

一、 各點本身的內函數

因此我們定義模型中每一個點是對前一時刻值的遞迴函數：

$$X_{t+1}=f(X_t) \qquad -1\le x\le 1$$

我們選來討論的波動函數：

$f(x)=4*\lambda*x*(1-x)$

$f(x)=1-a*x2$

$$G(x)=\begin{cases} ax & -0.5\sim 0.5 \\ -a\,(1-x) & 0.5\sim 1 \\ -a(1+x) & -1\sim -0.5 \end{cases} \qquad 0\le a\le 2$$

二、 同步模型：

前人已提出各種研究的模型

Coupled map lattices with global coupling

K.Kaneko 1989 提出

Coupled map lattices with intermediate-ran

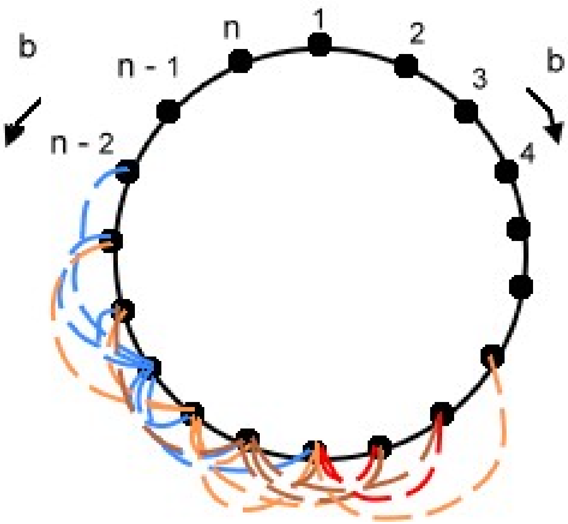
S.Sinha,et 1992 提出

R.Kozma 1998

P.M Gade and C.-K. Hu 1999

我們將這些模型的優點整合成

Synchronization mode

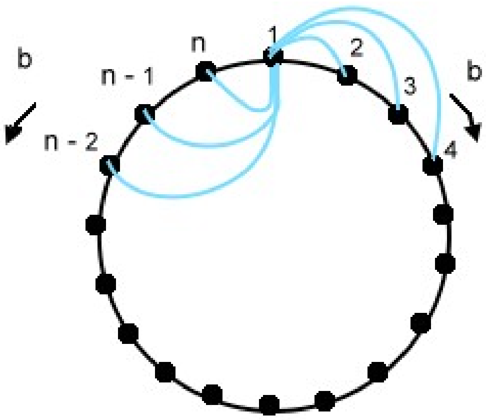


$$\chi_{i,t+1}=(1-\varepsilon-r)*f(\chi_{i,t})+\varepsilon*(f(\chi_{i+1,t})+f(\chi_{i-1,t}))/2+r*\Sigma\{f(\chi_{i+k,t})+f(\chi_{i-k,t})\}/(2b)$$

意義：

在一個平面上定義 n 個點，然後彼此相鄰的 2b 個點互相相連，以此線表示會受到影響。然後我們要定義函數：單看一個點本身就要有個函數

($X_{i,t+1}=f(X_{i,t})$)，I 表示位置。然後我們在給這些線的影響下個定義，給他一個函數。為了定義方便，姑且我們就先把點排列在一個圓上，如此一來相鄰的 2b 個點也好定義。



以這張圖為例 b=3 n=18

圖一

$\varepsilon*(f(\chi_{i+1,t}+f(\chi_{i-1,t})/2$ 的這一項， ε 為倍率，讓隔壁的影響加大。

$r*\Sigma\{f(\chi_{i+k,t}+\chi_{i-k,t}))/(2b)$的這一項，就是往左往右 2b 範圍取平均乘以 r 倍率再加上去。此項是最主要的回饋。有關此模型，可以參考 link 一書。這並不是最好的用來描述社會的模型，也不適合網路，但卻是簡單而比較容易定義的，其他都需要加入一定比例的隨機聯繫。

三、研究設備及器材

用電腦去計算函數，依據不同目的，寫了兩種程式：

1. 研究內函數本身(討論本身 λ)

用 c++算出最後收斂的值或是發散的範圍

```
#include<stdio.h>
#include<math.h>


double a = 0.0;
double f(double x ){
  return 1-a*x*x; '在這裡可以更改目標函數
}
int main()
{

  int i;
  double y,x;
  char c;

  fprintf(stdout,"please input the value of a for your function:");
  fscanf(stdin,"%lf",&a);
  fprintf(stdout,"please input the value of x for your function:");
  fscanf(stdin,"%lf",&x);

  fprintf(stdout,"eacho:input a=%lf   x=%lf \n",a,x);
  i=1;
  y=f(x);
  fprintf(stdout,"x_0=%lf\n",y);

while(!((((y-x)<0.000001)&&(y>x)||(x>y)&&((x-y)<0.000001)))){
    x=y;
    y=f(x);
    fprintf(stdout,"x\_%d=%lf\n",i++,y);
    }
fprintf(stdout,"\nthe fixed point we find is: %lf\n",y);
fprintf(stdout,"press any key to stop");
fscanf(stdin,"%c",&c);
}
```

2 同步模型

有時候需要同步過程，可以加入存檔功能，把過程數據都紀錄下來。只是要等結果，就把檔案部分抽走，執行上比較有效率。爲了存取方便，我們用 VB 寫。

```
Private Sub Command1_Click()
Dim fs As New FileSystemObject
Dim txtf As TextStream, txtf2 As TextStream
Dim s   As String
a = Text1
b = Text2
n = text3
e = Text5
Label9 = 2 * b / (n + 1)
Dim ii, tt As Integer
r = Text6
For tt = 0 To 1999
'For num = 1 To 10
'num = 1
For i = 0 To n
Select Case x(i)
Case 0.5 To 1
y(i) = a * (1 - x(i))
Case 0 To 0.5
y(i) = a * x(i)
Case -0.5 To 0
y(i) = a * x(i)
Case -1 To -0.5
y(i) = -a * (1 + x(i))
End Select
Next i
```

```
For i = 0 To n
sum = 0
For ii = 1 To b
If (i + ii) > n Then
j = i + ii - n - 1
Else
j = i + ii
End If
If (i - ii) < 0 Then
k = i - ii + n + 1
Else
k = i - ii
End If
sum = sum + y(k) + y(j)
Next ii
ii = 1
If (i + ii) > n Then
j = i + ii - n - 1
Else
j = i + ii
End If
If (i - ii) < 0 Then
k = i - ii + n + 1
Else
k = i - ii
End If
    z(i) = (1 - e - r) * y(i) + (e / 2) * (y(j) + y(k)) + r *
(sum / (2 * b))
    x(i) = z(i)
Next i
sum = 0
For i = 0 To n
sum = sum + z(i)
Next i
range = sum / (n + 1)
sum = 0
For i = 0 To n
y(i) = (z(i) - range) * (z(i) - range) / (n + 1)
sum = sum + y(i)
Next i
'state(num) = sum
'Next num
'sum = 0
'For num = 1 To 10
'sum = state(num) + sum
'Next num
'range = sum / 10
range = sum
Set txtf = fs.OpenTextFile("c:\test\test1.txt")
Set txtf2 = fs.CreateTextFile("c:\test\test2.txt")

If Not txtf.AtEndOfStream Then
Text7.Text = txtf.ReadAll
End If
Dim kk As String
s = Text7.Text
kk = range
Text4.Text = s + vbCr + vbLf + kk
Text4.Text = range
s = Text4.Text
txtf2.Write s
txtf2.Close
txtf.Close
Set txtf = fs.CreateTextFile("c:\test\test1.txt")
txtf.Write s
txtf.Close
Next tt
End Sub
```

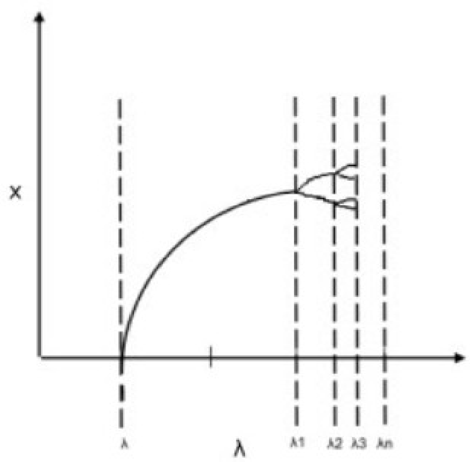
參、研究結果與討論

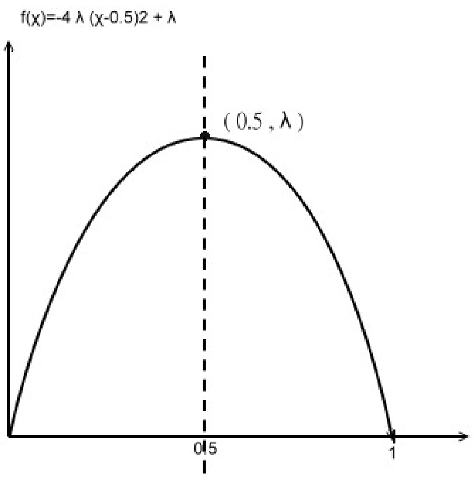
一、討論內函數本身

剛開始我們對於這些方程式都還不太熟悉，所以我們就先寫程式來觀察方程式數值收斂跳躍的情形

1. $f(\chi)=4\lambda\chi(1-\chi)$

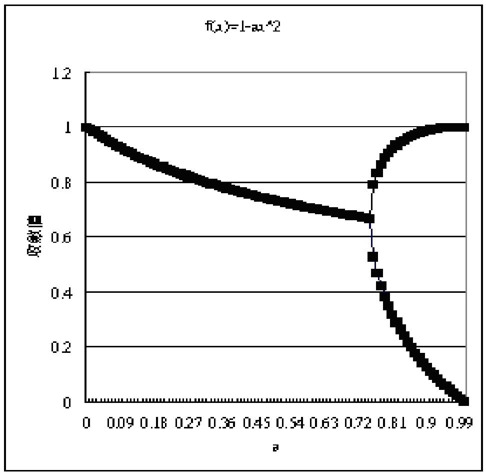
方程式[1]來說，在$0.25 \le \lambda < 0.75$開始才會有收斂的情形產生，並且收斂在一點，在$0.75 \le \lambda < 0.864$其跳躍的值就會變成兩個，在$0.864 \le \lambda < 0.8921$時跳躍值點會變成四個，之後跳躍點會愈來愈多，最後在0.92～0.99時λ會混沌。〈如圖二〉。





2. $f(\chi)=1-a\chi^2$

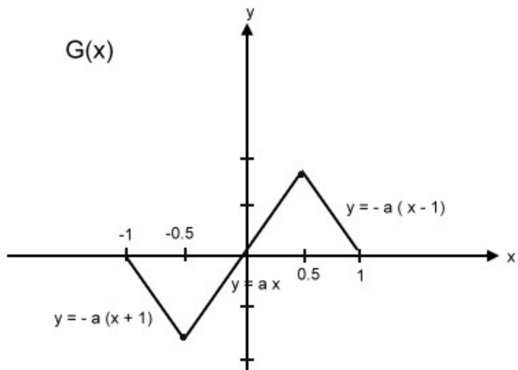
方程式[2]較為簡易，其所收斂的點在 $0 \le a \le 0.75$ 只有一個，在 $0.75<a \le 1$ 時會收斂於兩個〈如圖三〉。



圖三

3. $G(x)=\begin{cases} ax & -0.5\sim0.5 \\ -a(1-x) & 0.5\sim1 \\ -a(1+x) & -1\sim-0.5 \end{cases}$

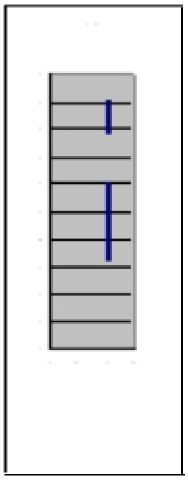
$0\le a \le 2$

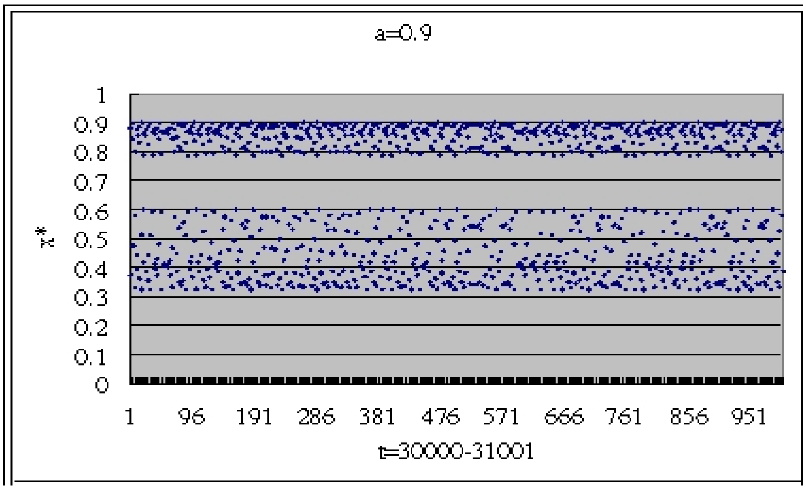


方程式[1]會是比較好的研究對象，因爲他λ的範圍比較大。

4.嘗試投影方式

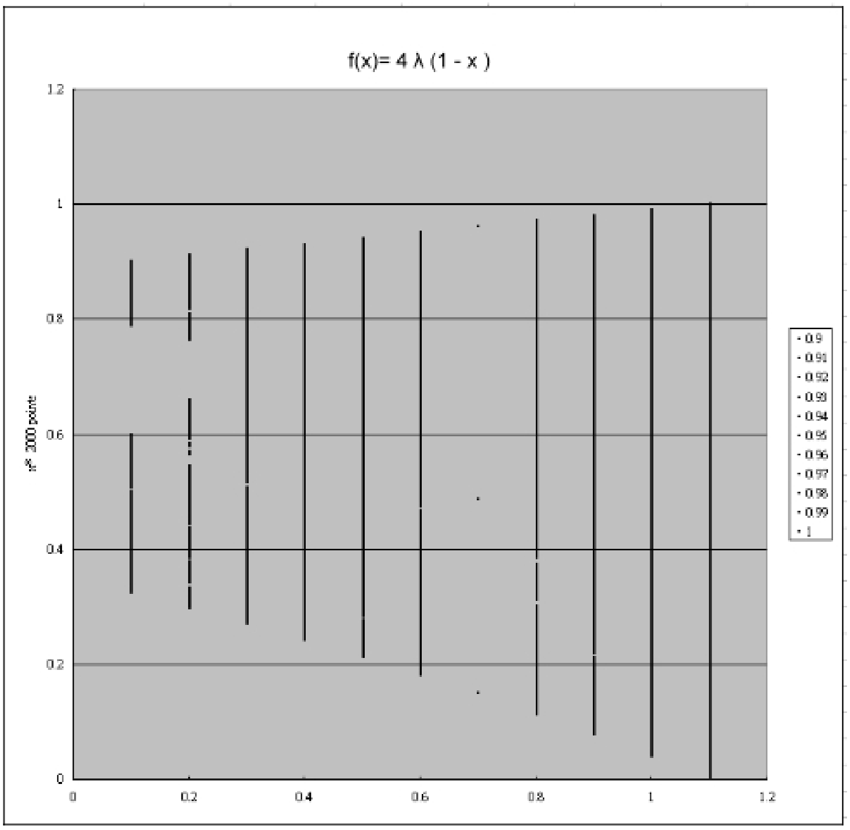
在完全發散之後，雖然無限跳躍而不會回到原本的數值，但是不同λ還是會有不同的範圍。要求出這個範圍除了用數學去解析函數，我們想到利用投影的方式也可以表示求出。





執行一千次看數值分布情形也就差不多了，然後找最大最小值或是投影到同一個x座標。T 選用30000之後的數值，因爲起始的時候數值還爲開始跳躍，這從數學解析函數可以解釋。或是用幾何的方式去了解函數意義再用微分討論斜率也就可以得到λ收斂範圍發散範圍。

這個圖表就是我們投影出來的範圍，線中間有間斷，是因爲只有執行1000個點的原因。
0.96的結果很特別，這並不是程式錯誤，實際計算也是這個結果。



最左邊第一條是λ=0.9，再來是0.91依序類推。
意義：如果選用越大的λ想當然就越難同步，回饋的影響就更爲重要，或是函數本身離散到用一般的手段都無法同步。

二、同步過程的研究

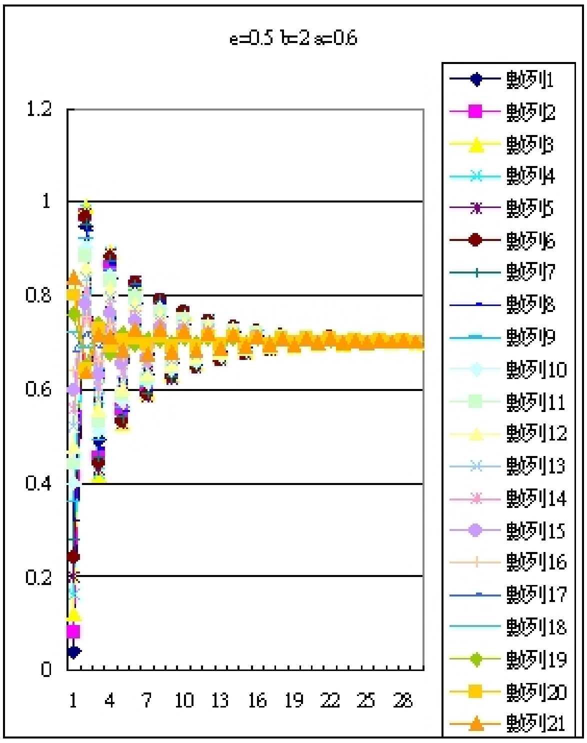
而在對兩個方程式都基本的研究過之後，我們認爲方程式[2]做起來較爲單純。我們一開始還不很了解點的數值是怎樣變動的，所以我們用最簡單的函數 $f(x)=1-ax^2$ 來做討論，並且只取了 20 個點。原本程式不怎麼會寫的時候，只能取 10 幾個點，現在要多少點都沒問題了。我們把 20 個點的數值變動都做成圖表，再配合不同範圍的變數，來討論同步過程。原本程式不週全的時候，我們數據必須用手寫的紀錄，現在都用自動存檔的方式了。

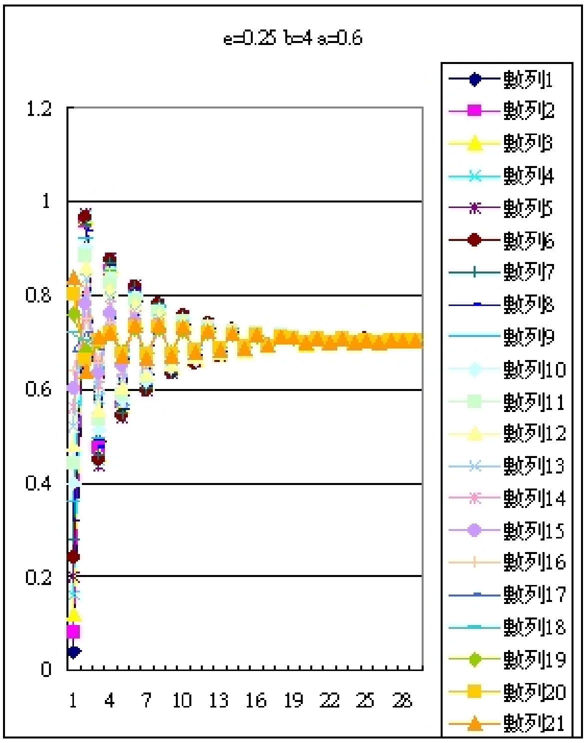
1. $f(x)=1-a*x^2$ 的同步觀察

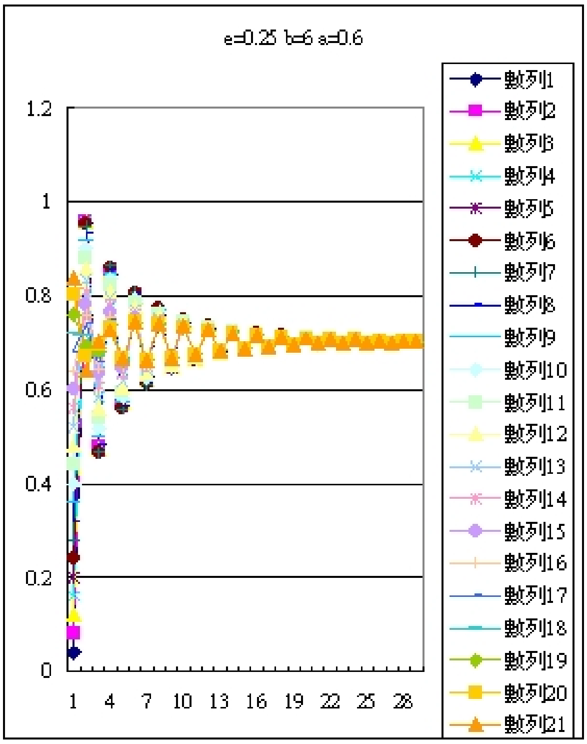
e=r 整體影響的倍率。

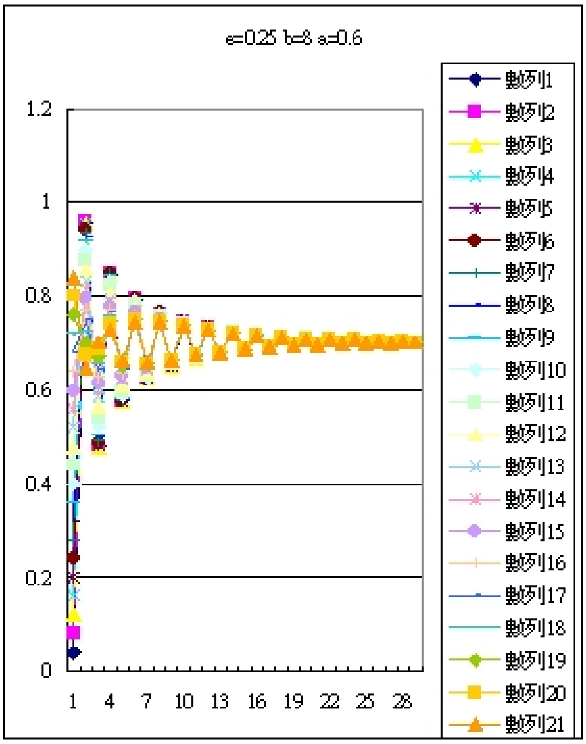
a=λ 影響函數收斂或是跳躍兩點。

b=有 2b 的範圍有能力影響一點，使其同步。





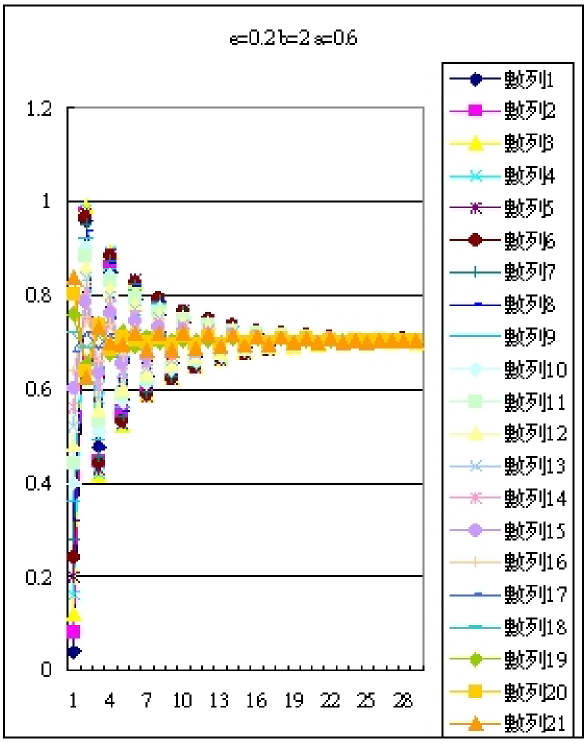


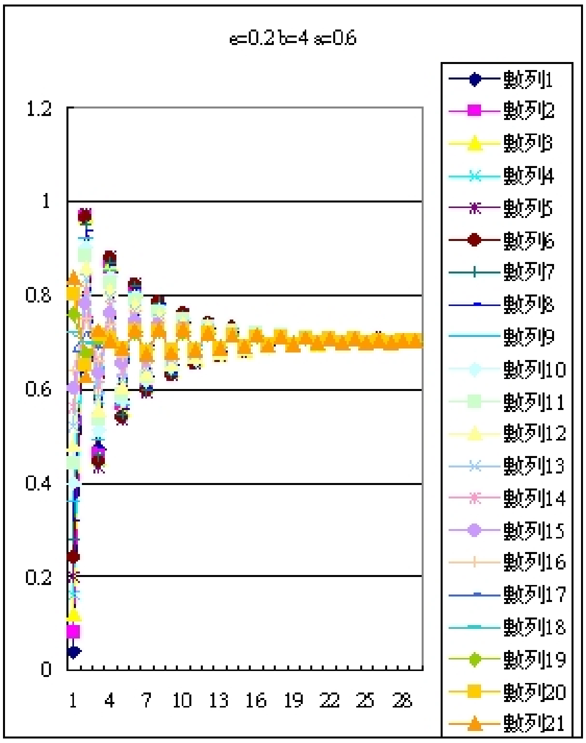


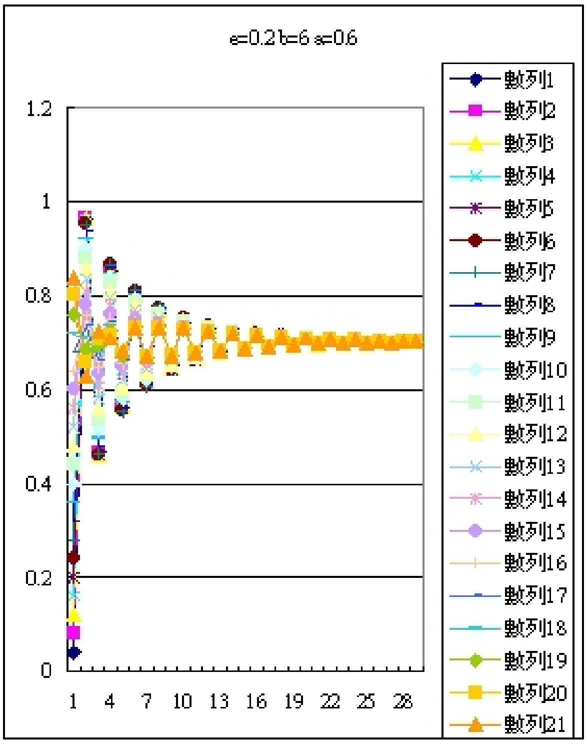
這四張圖示在變數 e,a 固定時

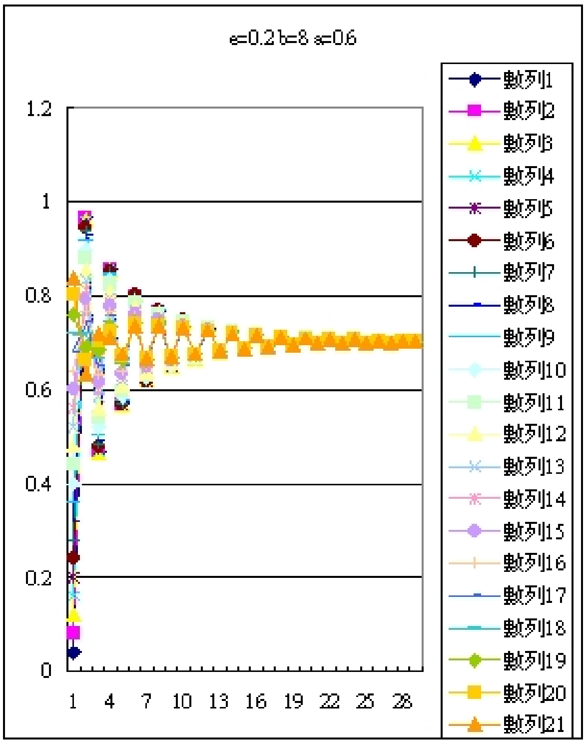
b 分別在 2. 4. 6. 8 的圖形

我們可以看出，b 值越大，其同步過程的速度會越快

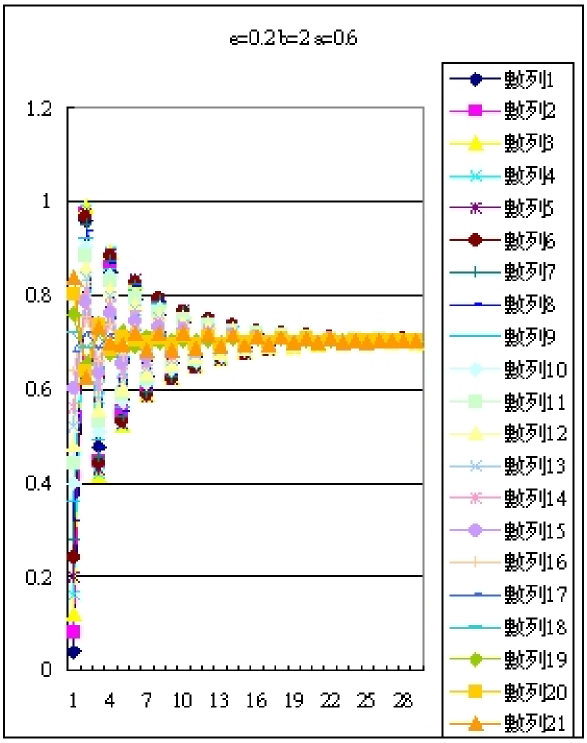


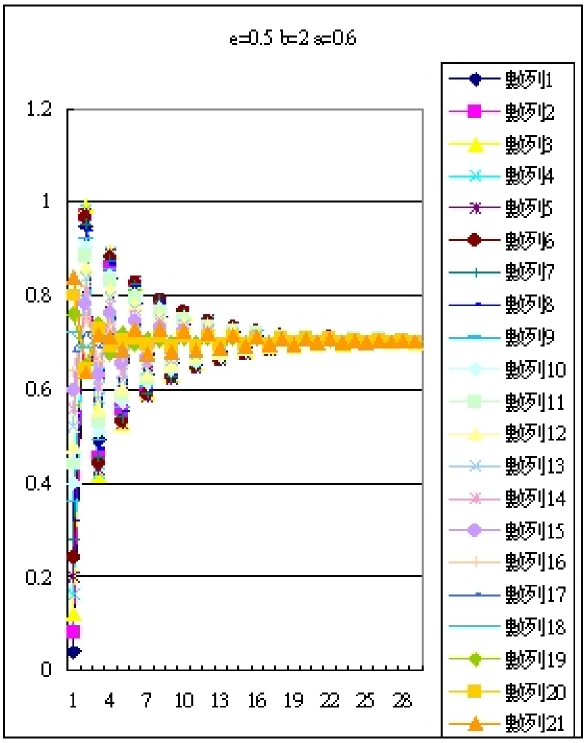


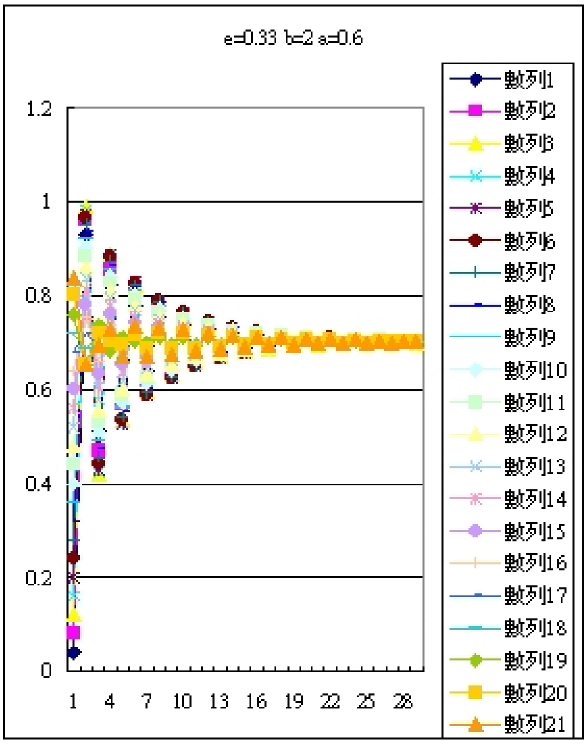


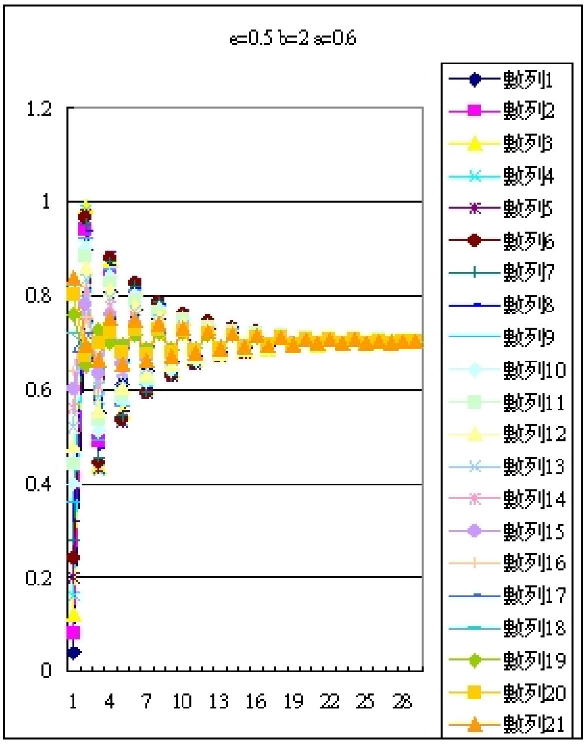


這四張圖示也是在變數 e,a 固定時
b 分別在 2. 4. 6. 8 的圖形，但 a 值不同與之前
我們可以依然可看出，b 值越大，其同步過程的速度會越快





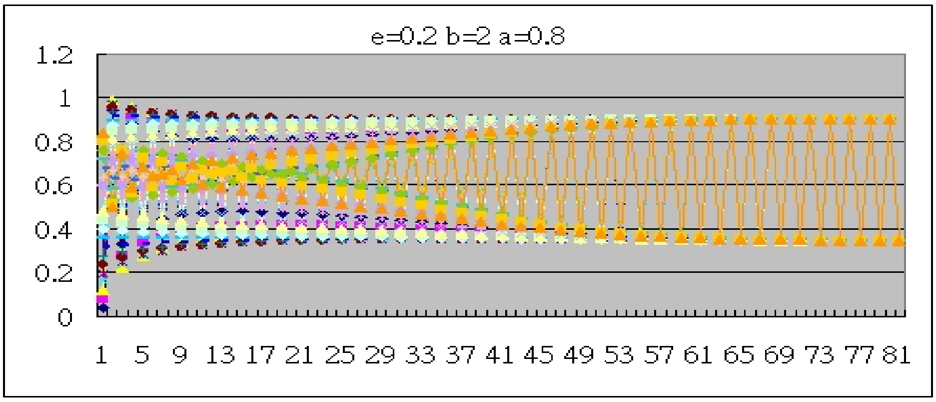


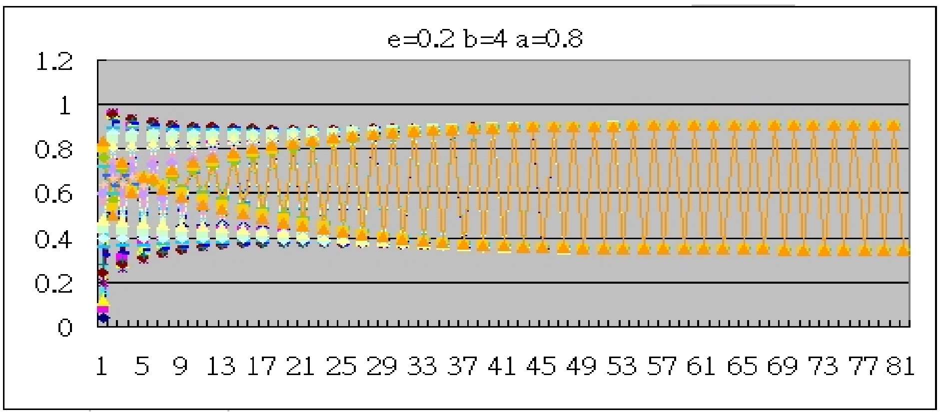


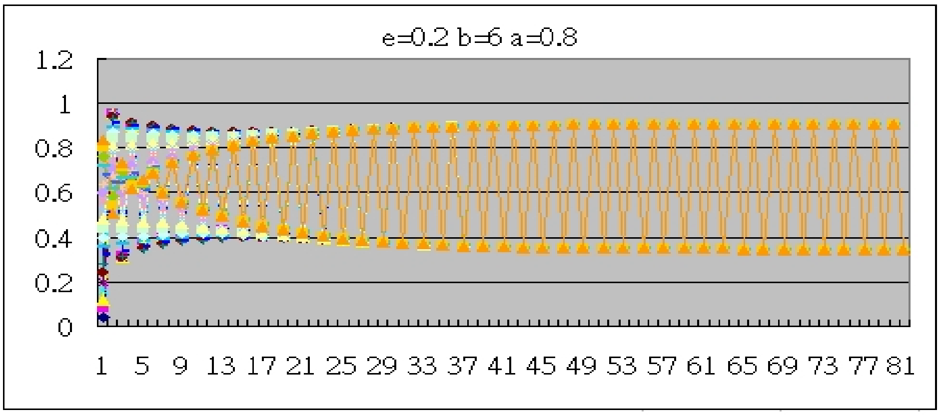
這邊是將 b,a 固定

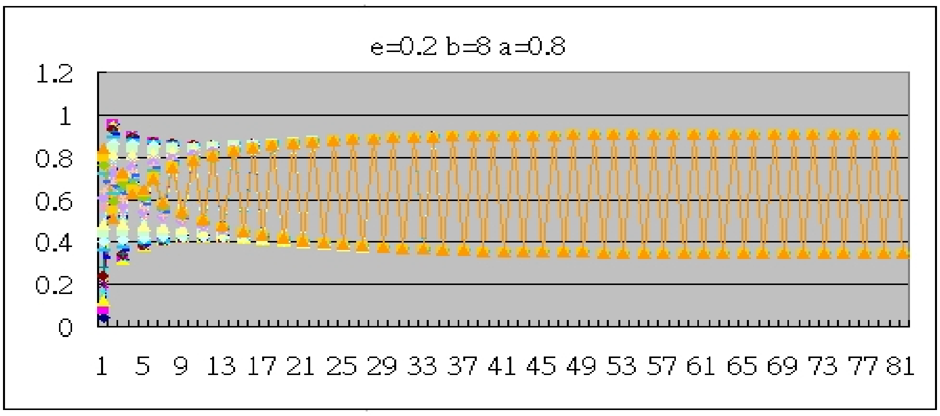
e 分別爲 1/2. 1/3. 1/4. 1/5 時的圖形

可看出 e 值越大，函數同步就越快



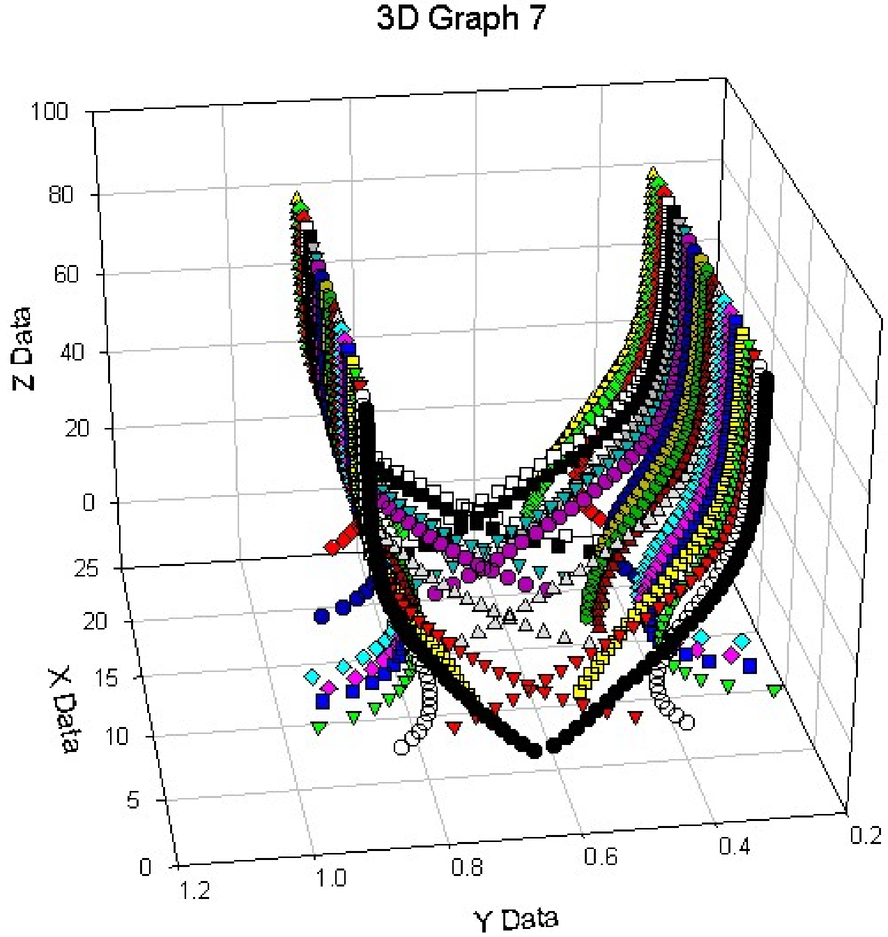


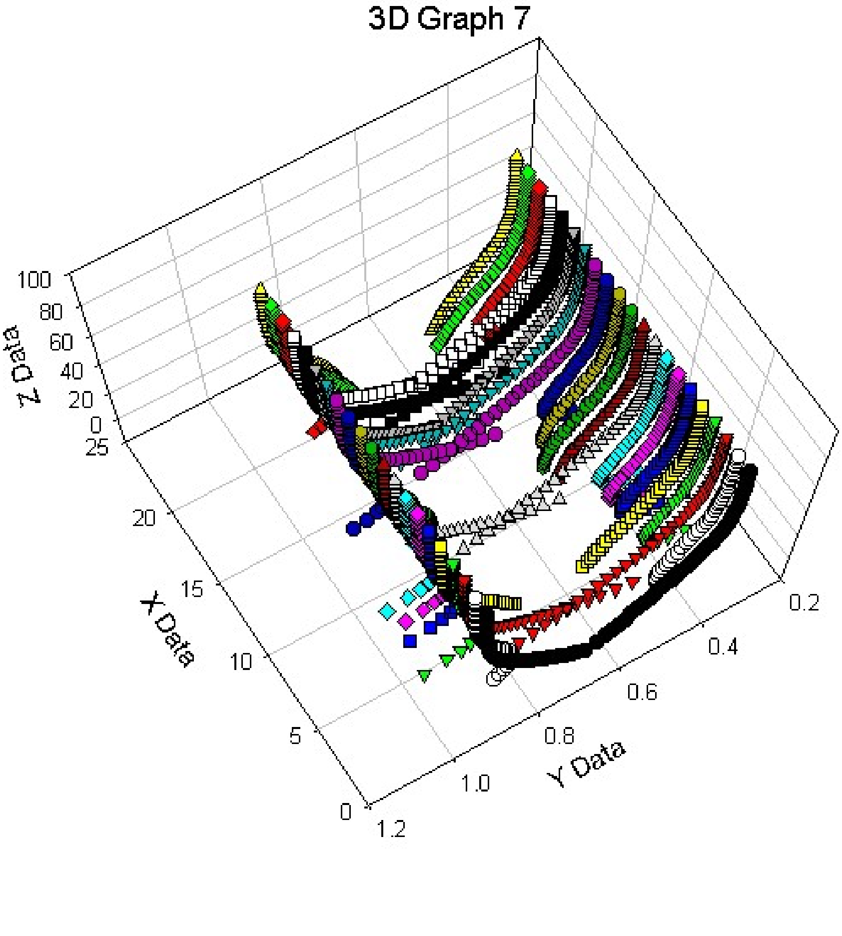




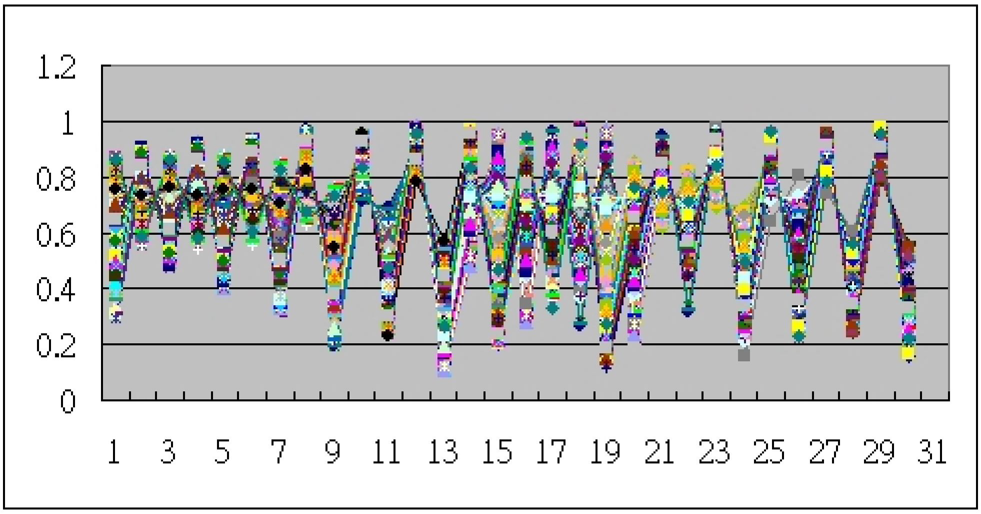
而這四張圖所表示的是將 a,e 固定，而當 a 爲 0.8 時，根據我們前面研究函數的結果會收斂於兩點，我們發現當b值越大時，數列被周圍影響的情況就越顯著。

下面兩張圖是用立體的方式觀察同步的過程(z 軸是時間 x 軸是點的座標 y 是點的數值)





2.比較複雜的函數 f(X)=4 λ *x*(1-x)

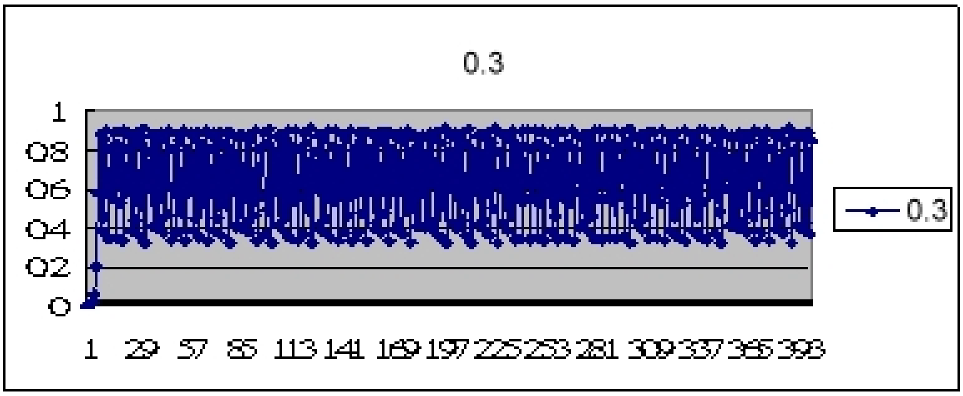


上面每一條線都代表著一個起始值以及之後被其他值影響的跳動情形。而觀察過上面的圖形做了比較之後，我們發現，當 λ,e 值都相同時，b 值愈大，其同步前的跳動就愈大。x 本來都已經要趨向於收斂值了，可是在其他點的影響之下，又被拉開，這種情況在每個圖形中都可以明顯的看出，而又以 b, r 值愈大的情形下，被拉開的情況愈明顯。

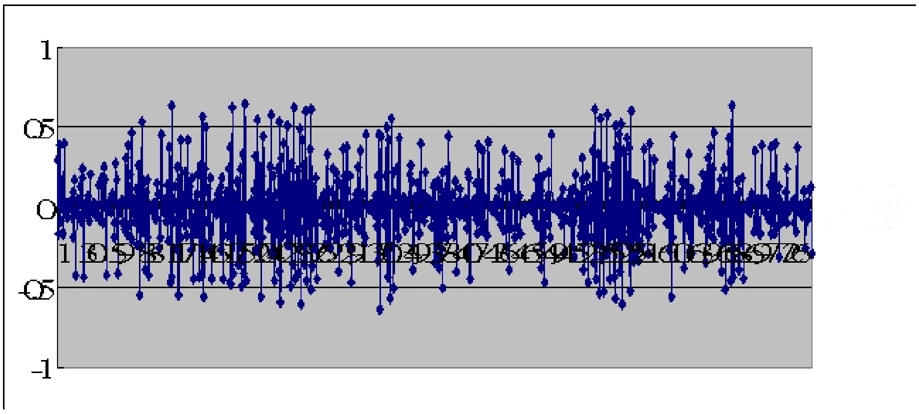
a.受到本身前一刻時間的值影響

這一部分教授只有提到一些，而且有關的論文只看了一篇，主要是再說很複雜的系統如果經過這樣的回饋去運算，可以得到可以討論的結果。這一部分我們嘗試運算看看，作爲一個練習。

以 f(x)=4λ*x*(1-x)爲例：



b.每隔一段時間 x 的差：



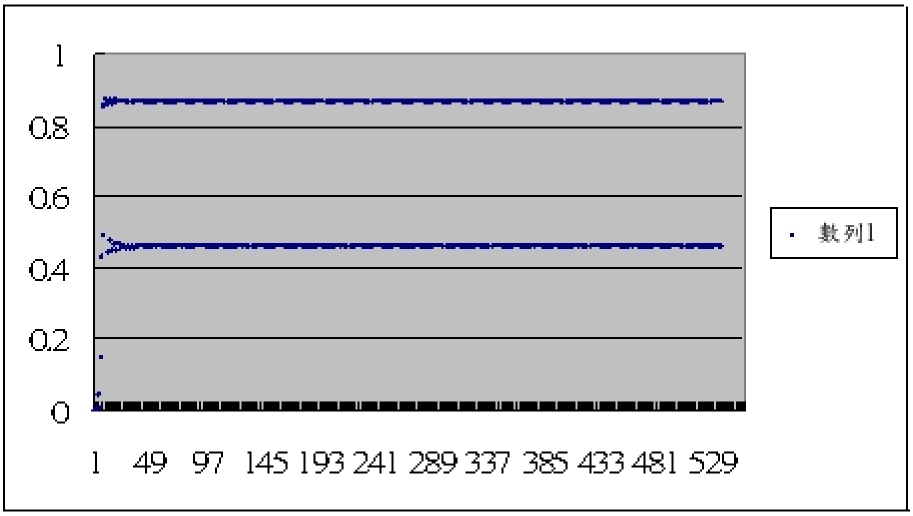
這張圖數據是先用原函數，運算幾千項然後截取一段數據，再將 t 時刻的值跟 t-k 時刻的數值做一些回饋。我們希望能整理出所以然，利用這個方式分析原本混亂的軌跡。結果不彰。(k 隨意找….)

c.代入 f(x)之後再與前一刻的值取回饋後得到的圖形

$X_{t+1}=(1-r)*f(X_t)+r*X_{t+1}$

這樣的方式可以得到比較有章法的圖形去描述這個系統，可以用最後同步的兩個數值還有同步的時間以及 r 表示這個點的資訊。

而且我們也可以從下面這樣的數據到推回函數所使用的λ不過根據不同的r有著小於0.001的誤差



3.根據我們所看的一些論文我們進行以下的實驗：

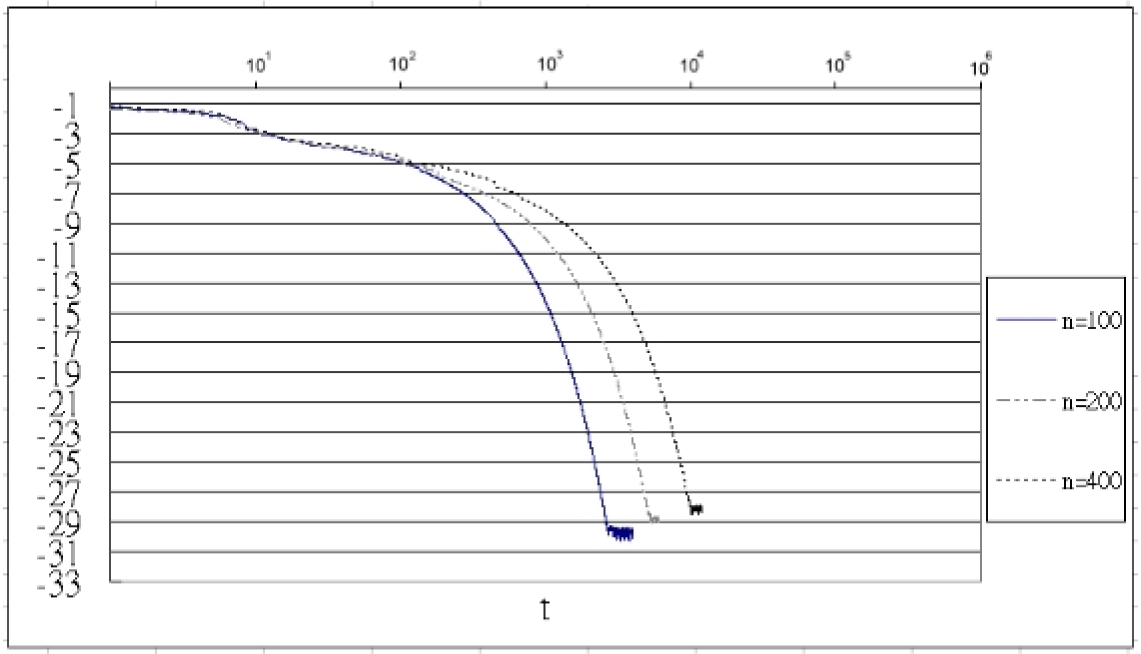
a.G(x)

我們要把點的數目推向 200 以上，對於同步的定義要重新修正，前面是要數值完全相同，因爲不太可能同時比對那麼多筆數據，因此改爲看同步的程度。我們定義標準差平方

$\theta^2=\Sigma(X_i - <t>)^2$ ，< t >爲此刻所有點的平均。而在電腦中同步程度小於 10^{-30} 我們就視爲達到同步。一次跑同一個數列，而一個數列在相同變因下做了 10 次之後取平均，這是因爲一開始隨機數值會有誤差。固定變因分別爲：a=2, n/2b=1, e=0.1,r=r*=0.5

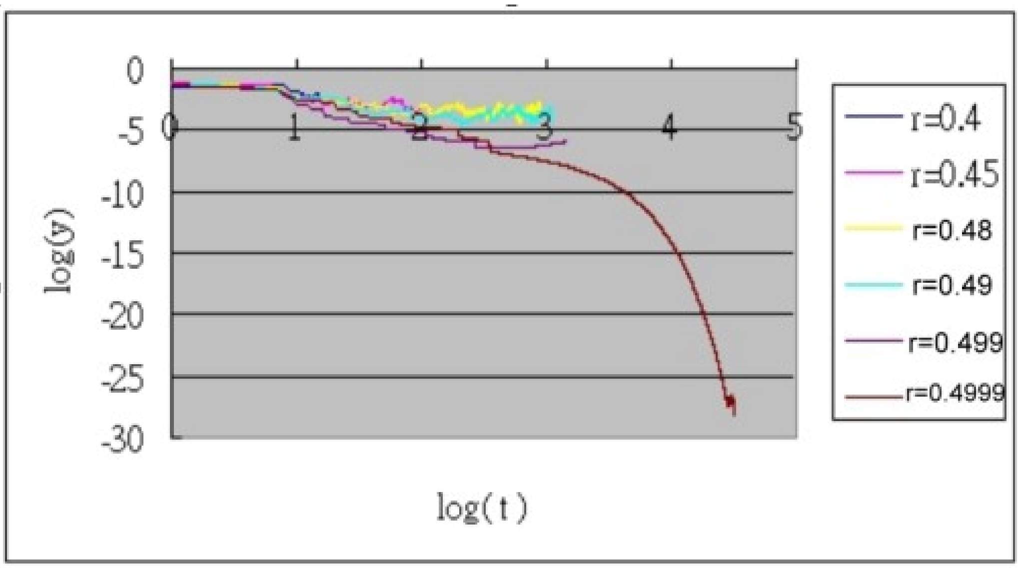
改變不同的 n ，點的數目。50,100,200,400……

然後對時間 t 取 log 以及對標準差平方 θ^2 取 log 這樣做出的圖表才會清楚明白方便討論。



物件數目越多，同步速度越慢。我們還希望能將圖形經過某種處理之後疊和，滿足良好的物理意義。可是只有部分能疊和，可能跟程式數字有效位數不夠精密有關係。

b.不同 r 值對同步程度的影響

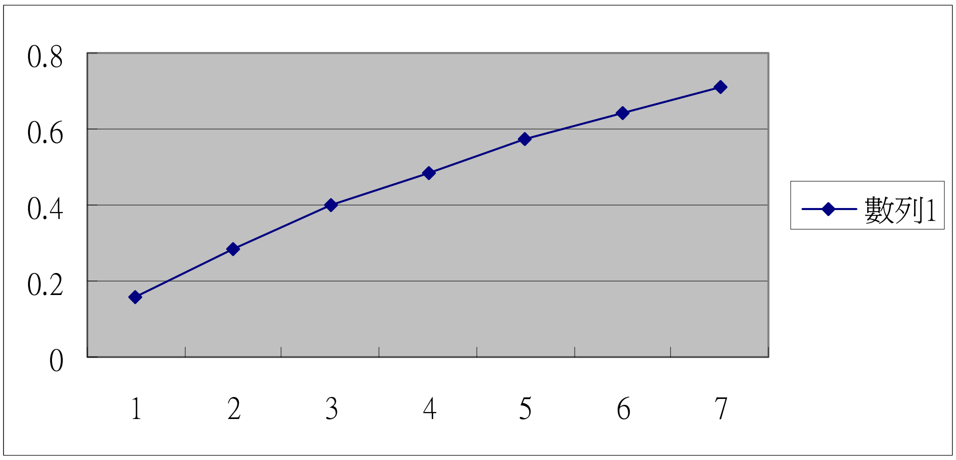


利用這種窮舉方式十分逼近法找出 r*值，可以準確到小數第四位。

r*的意義就是可以達成同步條件的最小值，所以是研究的重點。

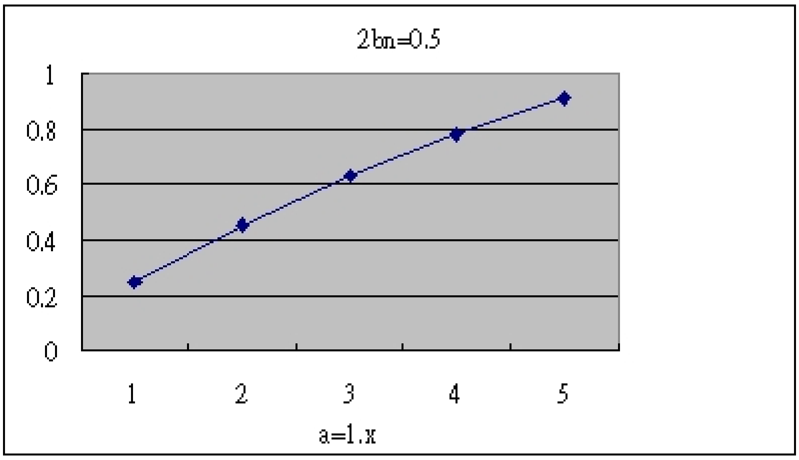
c.我們下面就討論不同的 2b/n 範圍，找到 G(x)對應的 r*。

2b/n=0.666



x 座標是 a = 1.x　　y 座標是 r*　　e 固定等於 0.1

2b/n=0.5

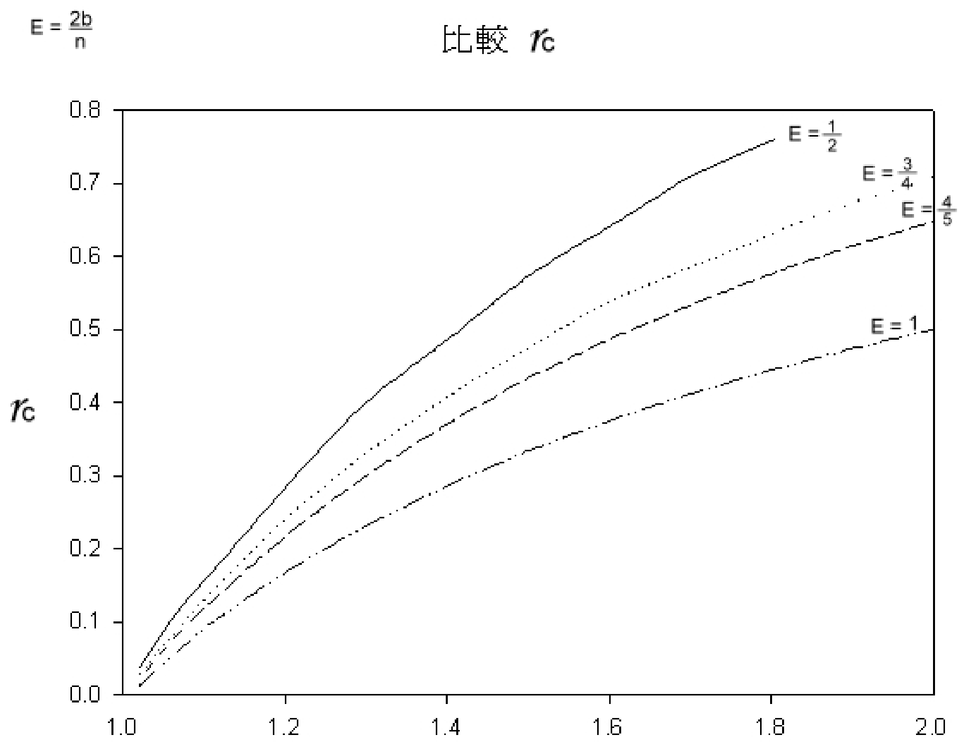


範圍比較小 r*變得更大

座標單位同上，e=0.1

然後我們發現 r*應該是有個函數關係。

在 2b/n=0.5 的時候，a 超過 1.5 就無法同步 已經 r*超過 0.91，本身函數倍率變成零。除非能有一個系統，完全只受到別人的影響，而自己的下一刻值自己忽略卻又能影響別人，才能達成同步。



橫座標代表的是 G(x)函數的內變數 a , y 軸代表的是 r*值。不同的曲線分別代表不同的 2b/n(影響範圍 E) 。

我們可以看出影響範圍 E 越大，所需要的同步條件 r*就越低，越容易同步。而當 G(x)的內變數 a 越大，使得數值分布越複雜，也導致所需同步條件 r*提高，增加同步的難度。

中研院的一篇研究結果

是利用解析的方式求出來的，我們用來比較我們的結果。

‘2b/n=1

‘r*=1-1/a

不過並沒有提到其他比例，因爲他的模型是整體同時影響的，而我們多加入 b 這個變數。

三、討論

1. 方程式[1]會是比較好的研究對象，因爲他 λ 的範圍比較大。在完全發散之後，雖然無限跳躍而不會回到原本的數值，但是不同 λ 還是會有不同的範圍。
2. 在我們第一個同步過程研究中，在 a < 0.75 時函數本身收斂無法判斷是否同步，甚至同步模型還會影響個別收斂速度；a > 0.75 時 b 值越大時，數列被周圍影響的狀況就越明顯， e 越大同步越快。

3. 在"比較複雜的函數"中我們發現，波動函數本身 $f(\chi)=4\lambda\chi(1-\chi)$對數值的離散影響很強，λ不能太大，不然很難達到同步〈r*都很大〉。由於這部分的運算所需時間太長，所以我們最後選用 G(x)討論。
4. G(x)的函數我們沒有特別加以研究，是因爲構造比較簡單，只是線性關係遞迴的函數，稍微把函數式看懂，用圖形解析就可以知道λ對應的範圍，以及數值跳躍的方式。
5. 在"G(x)"中我們會發現物件數目越多，同步速度越慢。我們還希望能將圖形經過某種處理之後疊和，滿足良好的物理意義。可是只有部分疊和，可能跟程式數字有效位數不夠精密有關係。
6. .r*原本是要用數學去證明導出式子，我們沒有這樣的功力，但是發現直接利用 r*的性質也可以找到。r*値是一個零界點，未超過這個值以前，系統在如何運作最終都無法達到同步，一旦滿足這個值系統就能達到同步。
7. 我們發現 r*- λ 應該是有個函數關係，而且會是用線性去推，加入修正逼近。
8. 我們在這實驗中其實只是把電腦當成計算機，設計了很多迴圈給電腦計算，不過卻是相當花時間，每個點都要計算一次下一刻的直，然後要累加每個點周圍的平均去回饋，計算非常繁複，而且通常一個圖表定義平面 1000 個點然後要跑 16000 次程式左右。這麼多筆數據當然不能用一開始的方法複製貼上，最後才寫出能儲存數字的程式，讓我們進度快了很多，不然當初根本做不了這麼多討論。
9. 有存檔的程式速度相對會比較慢，討論 R*我們只用看最後擺盪的值或是討論函數本身我們只看最後幾項收斂或跳躍的值，也就不用存檔，讓實驗速率快一些。
10. 同步模型的意義，可以看成一個人生活在 n 個人的社會上，最有關聯的是周遭的 2b 個人他們的思想會影響到這個人，影響的程度爲 r。他也會有一兩個超級好友或是父母老師，給他們的影響比較大的人加上 e=0.1 的倍率。
11. 我們選用的這個模型，還不能完全描述現實社會狀況，現實的小世界還需要加上隨機連線，這比例很低的隨機連線卻是在圓上相對兩點要互相影響的捷徑。如果是這樣的模型，2b/n=0.5 以下的 r*可以下降很多。可是這樣的程式不好寫，不太能寫出隨機的感覺，反而會是很特意，等我們能力夠了再去嘗試討論。

肆、結論與應用

一、我們先用小系統(n≦50)得到能印證連結觀點的結果：

1.系統越雜亂，就需要稍長的時間同步；

2.個體數越多時，各點需要更大範圍的點數去影響於每單位時間內以及更深的影響才能同步；

3.起始值隨機影響同步臨界條件並不明顯，而且當點數 n 越多影響越微。

二、更進一步，我們將系統推向 n≦400 點,t→100,000 次，我們發現：

1.在”G(x)”我們希望能將圖形經過函數修正之後能疊和，驗證”Scaling and Universality In Transition to Synchronous Chaos with Local-Global Interactions ”中的結果，但只有部分疊和，尺度不相同；

2.可以直接利用臨界點的意義用十分逼近法求出臨界值的近似值到小數後四位，可用曲線去近似， 因爲十分圓滑；

3.我們用近似值也能發現同步條件與系統各點本身可跳躍的數值範圍(G(x)內變數 a)是負相關，與影響範圍 E(2b/n)也是負相關 ，可以描述現實現象。

這種窮舉方式，交由電腦運算，不需要特別的解析能力就能夠快速且大量求得同步條件的近似值，尤其在運算不熟悉的系統時。

在研究過程中我們發現利用電腦模型寫出的程式，可以將現實狀況的情形加以描述。

只要你的模型能夠描述你的對象，並且將所需要的資料統計並數據化。例如研究 AIDS 擴散情形，你將可能需要一個巨觀的性聯絡網以及統計出的感染機率，也需要其他可能傳染途徑的連結網以及其統計出來的感染機率，再加上防護措施保護其中環節以及其保護效果。越詳細的數據資料就越能模擬系統，而其中個體的差異是可忽略的。

你可以預測如果致病率降低到多少的程度，AIDS 族群將會在系統中隨著患者的老死衰退。等計算出這個臨界值，再交給防範愛滋病的研究人員努力，作爲一個指標。加上我們可以用電腦運算的方式找出臨界值，而不需要特別去解析，這樣在應用上將會更方便更容易推廣。

伍、參考文獻

1. **Robert M. May** , Simple mathematical models with very complicated dynamics, 149 ~ 157 Nature Vol. 261 June 10 (1976)
2. **Chin-Kun Hu** , Synchronization in Nonlinear Couple Systems , October 1 (2002)
3. **Prashant M. Gade and Chin-Kun Hu**, Synchronization and coherence in thermodynamic couple map lattices with intermediate-range coupling, Phys. Rev. E 60,4966-4969(1999)
4. **Prashant M. Gade and Chin-Kun Hu**, Scaling and universality in transition to synchronous chaos with local-global interactions5. **Albert-László Barabási**, Linked

評語

同步現象是自然界中有趣重要的主題，本作品以電腦模擬前人的模型以及各種模型的組合。學生自行撰寫程式演算所整合的模型，是很好的驗證作品。