Chaos in Exchange Rate Models

Dartmouth College
Department of Mathematics
Daniel Salas

November 23, 2015

Abstract

This paper analyzes the dynamic behavior of exchange rate models. Chaos is easily found in discrete models, but not in most continuous models below 4 dimensions. The paper concludes with a brief look at the real data and the challenges it presents.

1 Introduction

Exchange rates are a hot topic in economics. Currency manipulation (government intervention in the exchange rate instead of letting it 'float'), exchange rate crises, and currency unions are a few areas of interest. Large changes in exchange rates can have dramatic impacts on many countries' economies. In finance, the exchange rate is an important factor in the return on foreign assets. As a result, many people (especially Forex traders) would like to predict exchange rates.

Unfortunately, their exact behavior over short time horizons is still unknown. Most economists have put their focus toward stochastic models, but some still try and explain the behavior with nonlinear dynamics note. Over long time horizons, relatively simple models that use comparative statics can accurately predict the equilibrium exchange rate. But over short time horizons, exchange rates have drastically different behavior. One observation that has led economists to believe in some nonlinear dynamics is the evidence of exchange rate overshooting. The Dornbusch model (1976) was the first model to imply the existence of exchange rate overshooting. As a result, the principles of the Dornbusch model are the foundation for many linear and nonlinear dynamic models. First, I will show that even simple models of exchange rates can display 'random' behavior that is actually just chaotic and determinant behavior. Then I will explore the dynamics of simple dynamic ODE models. Lastly, I will briefly look at time-delay embedding in real data.

2 One Dimensional Discrete Map

First, I looked at the behavior of a 1 dimensional nonlinear map. For iterative maps, it is possible to get chaos with only 1 dimension (logistic map with a = 4) and that's what one can find using a very simple exchange rate model. The derivation of the model is short and requires some heavy assumptions.

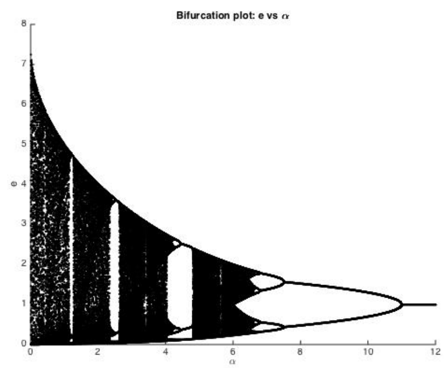

Figure 1: Bifurcation of α.

Exchange rates in each period are determined by the previous period's exchange rate and various parameters. The parameters are assumed to be constant. The relationship between the change in the speculators demand for foreign currency and the equation for the trade balance are combined to yield the next period's exchange rate in terms of the previous exchange rate and various parameters. First, the speculators' demand for foreign currency is quantified:

$$S_t = \alpha(e_f/e_t - 1). \tag{1}$$

α is a sensitivity parameter defined to be greater than zero (varying this sensitivity gives chaotic behavior). e_t is the current nominal exchange rate. e_f is the equilibrium exchange rate (fixed point in the map). The intuition is that the demand is proportional to the relative deviation of the equilibrium exchange rate to the current exchange rate. When α is small, speculators don't adjust their demand to deviations very strongly, thus the pressure on the next exchange rate to change is small. As α gets larger, the deviations are met with larger changes in demand, keeping the exchange rate within a tighter band around the equilibrium. As shown in the bifurcation plot, when alpha is sufficiently large (approximately 11), the speculators' demand responds so strongly that the exchange rate is quickly brought to its equilibrium level. The uninteresting case is when $\alpha = 0$. Then speculators play no role in determining the exchange rate since they never want foreign currency.

Next, the trade balance is defined

$$T_t = \beta(e_t - e_f) + \gamma(e_{t-1} - e_f). \tag{2}$$

The relative size of γ to β is important. When gamma is larger, more trade is done using the previous exchange rate (common for purchase agreements to be made well in advance of the actual purchase at the current exchange rate). When β is larger, the opposite is true. As both get large, the volume of trade increases. As shown in the bifurcation plots, when γ/β gets larger/smaller the chaotic behavior increases. When the γ is small enough relative to β, the exchange rate settles into the equilibrium rate instead of chaotically orbiting around it. Lastly, if all constants (α, β, γ) are increased and held to the same ratio, the behavior doesn't change. The bifurcation plot is just a set of horizontal line(s) with then

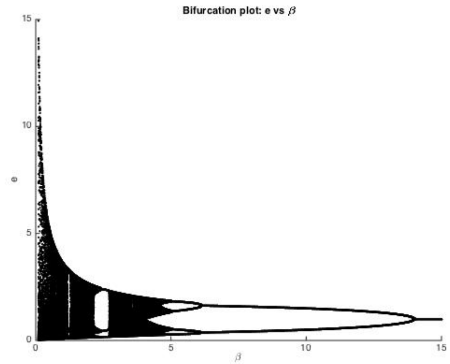

Figure 2: Bifurcation of β. Blows up near zero because of β term in denominator.

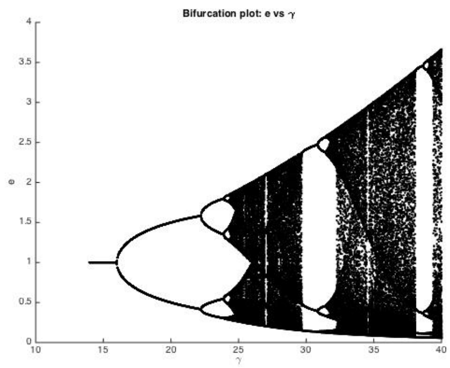

Figure 3: Bifurcation of γ.

number of lines depending on the period of the sink. The period is differs by changing the ratio $\alpha : \beta : \gamma$.

Setting $T_t = \Delta S_t$ yields a quadratic relationship between the current exchange rate and the previous exchange rate with the following positive root of e_t:

$$\beta e_{t-1} e_t^2 - ((\beta+\gamma)e_f e_{t-1} - \gamma e_{t-1}^2 - \alpha e_f)e_t - \alpha e_f e_{t-1} = 0$$

$$e_t = \frac{(\beta+\gamma)e_f e_{t-1} - \gamma e_{t-1}^2 - \alpha e_f + \sqrt{((\beta+\gamma)e_f e_{t-1} - \gamma e_{t-1}^2 - \alpha e_f)^2 + 4\beta e_{t-1}\alpha e_{t-1}}}{2\beta e_{t-1}} \tag{3}$$

The intuition for setting the change in demand equal to the trade balance is that the value of goods being sold by a country must be equal to the money flowing into the country. This is related to the idea of balance of payments. The balance of payments means that the current account (value of exported products in home currency - value of imported goods in home currency) should equal the capital account (the foreign purchases of domestic assets with home

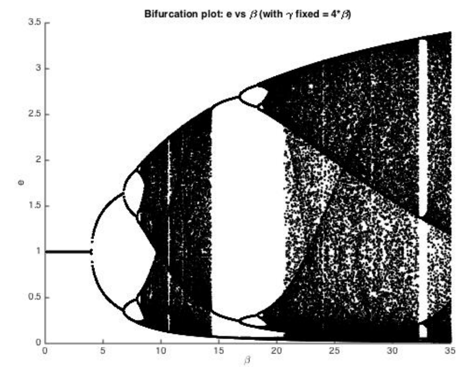

Figure 4: Bifurcation of β holding fixed $\gamma = 4\beta$.

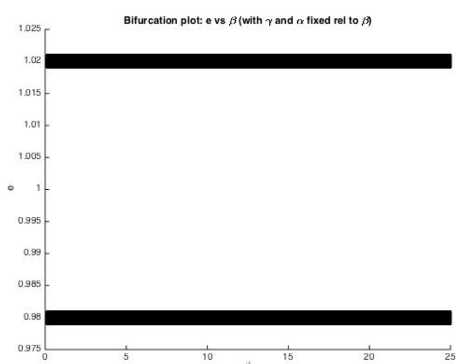

Figure 5: Bifurcation of β holding fixed $\gamma = 4\beta$ and $\alpha = 6\beta/4$. The sink is period 2 in this case.

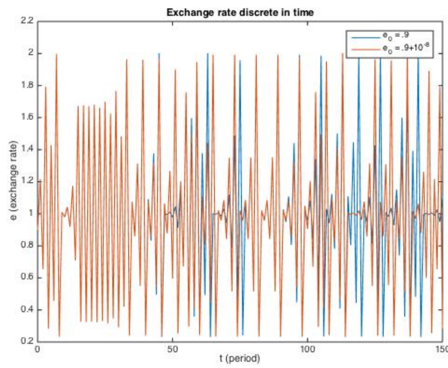

Figure 6: Time series illustrates sensitive dependence and unpredictable movements.

currency - the domestic purchases of foreign assets with foreign currency). The following thought experiment explains the reasoning behind the assumption.

> If the dollar is worth more than the euro, then Europe will buy less of the US's exports since they are relatively more expensive. In turn, the US will buy more of Europe's exports since they are relatively cheaper (the US imports more). Thus, the US's net exports are negative. In the capital market, the return (yield, interest rate, etc.) on European assets (bonds, stocks, etc.) is weaker because the euro is less valuable than the dollar. So, people in both countries would shift their purchases in the capital market to US assets since these assets now have a higher return. The balance of payments implies that the shifts in the net exports (part of the current account) and investment in assets (capital account) should be equal. (Assume net exports are a positive value and an increase in domestic purchases of foreign assets (capital outflow) is a positive value. Also, for simplicity, assume the world is made up of only two separate trading partners (US and Europe).)

From map defined in (3), we assume when the parameters are not varied, they are $\alpha = 6$, $\beta = 4$, $\gamma = 26$, $e_0 = .9$, and $e_f = 1$. These parameters are used in the time series plot, calculating the Lyapunov exponent, and the time delay embedding. The time series plot illustrates the sensitive dependence. The second point iterated is only 10^{-8} greater than e_0, but it reaches a maximum distance away from e_0 after only 50 iterations. The Lyapunov exponent is numerically calculated to be approximately $.3491 (> 0$ necessary for chaos). Lastly, the plot of the time delay embedding contrasted with the time series plot emphasizes that while the two points seem to deviate from each other randomly, they are both following the same exact relationship between e_{n+1} and e_n.

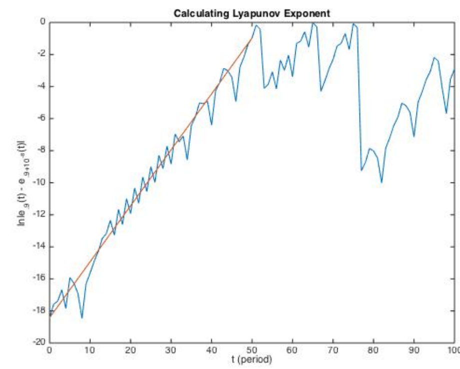

Figure 7: Calculating the Lyapunov exponent ($h = .3491$)

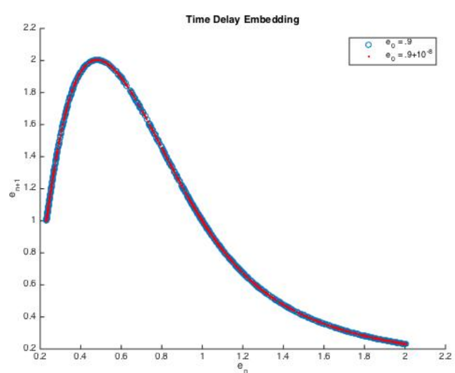

Figure 8: Time delay embedding shows the shape of the iterating function.

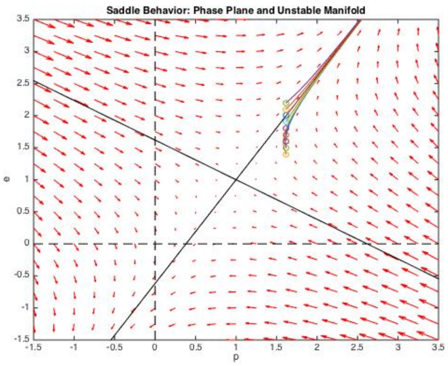

Figure 9: Phase plane and behavior near unstable manifold

3 Giavazzi's Simplified Dornbusch Model

Giavazzi (2007) shows the Dornbusch model can be simplified to the following 2D system of linear ODEs. Since the model is 2D and linear, it does not exhibit chaotic behavior (Alligood). Even though it is not chaotic, the model does have some interesting properties and implications that are useful to know before expanding into a 3D nonlinear system. e = exchange rate, p = price level, m = quantity of money in economy, r = the interest rate, and r_f = the foreign interest rate. All variables will be in log terms (real money = $ln(M/P) = m - p$). First, Giavazzi derives the $\dot{e}$ relationship from the money market equilibrium. The money market equilibrium requires that money demand equal a fixed money supply, $m^S - p = \bar{m} - p = m^D - p = -\lambda r + \psi y$. λ, $\psi > 0$ and they represent the sensitivity of money demand to income y and interest rate r. Then assume that the interest parity condition holds, $r = r_f + \mathrm{E}(\dot{e})$ (no arbitrage in capital market). Using Dornbusch's assumption that speculators have perfect foresight, $\mathrm{E}(\dot{e}) = \dot{e}$, so the expected change = the actual change. After combining these equations, $\dot{e}$ has solution:

$$\dot{e} = -\frac{\bar{m}}{\lambda} + \frac{p}{\lambda} \tag{4}$$

Then Giavazzi looks at a sluggish price level proportional (δ) to the discrepancy between aggregate demand and aggregate supply. Holding aggregate supply fixed at 0, and assuming the demand relation is $\dot{p} = \delta(y^D - y^S)$, the $\dot{p}$ equation is

$$\dot{p} = \delta(e - p). \tag{5}$$

The equilibrium point is $\bar{e} = \bar{p} = \bar{m}$ by solving $\dot{e} = \dot{p} = 0$, thus in matrix form the relation is

$$\begin{bmatrix} \dot{p} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} -\delta & \delta \\ 1/\lambda & 0 \end{bmatrix} \begin{bmatrix} p - \bar{p} \\ e - \bar{e} \end{bmatrix}.$$

I found the eigenvalues to be $\mu_{1,2} = -\frac{\delta}{2} \pm \frac{\sqrt{\delta^2 + 4\delta/\lambda}}{2}$, and confirmed Giavazzi's result that the eigenvectors (the manifolds) are $\vec{v}_{1,2} = [\lambda\mu_{1,2}\ 1]$ (second is stable). Since δ and λ are assumed to be greater than zero, the equilibrium point will always be a saddle since there will always be one negative and one positive eigenvalue of the linear ODE. Below, I have shown the phase plane and the behavior

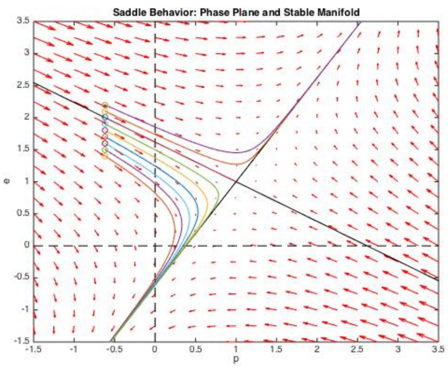

Figure 10: Phase plane and behavior near stable manifold

of solutions near the stable and unstable manifold (I modified a phase plane code from http://www-users.math.umd.edu/ petersd/246/matlabode2.html). Since, the equilibrium solution is always a saddle, very small deviations away from the stable manifold lead to solutions that diverge to ∞ or $-\infty$.

4 Nonlinear 3D system of ODEs

Asada et. al (2003) expand upon many models including the Dornbusch model. The nonlinear model proposed (Asada et. al p.227) gives rise to some interesting behavior. Unfortunately, this model does not have chaotic behavior, but it gets pretty close. The derivation of this model is much more complex, but the assumptions and relationships are still similar. The 3rd variable in this ODE system is the adjustment of the expected exchange rate. Previously, the expectations were assumed to be perfect, but now the expectations change along with the price level and the exchange rate. The dynamic behavior is described by the system:

$$\dot{e} = \beta_e(\bar{r}^* + \beta_\epsilon(e_0 - e) - \frac{\phi\bar{y} + p - \bar{m}}{\alpha_r})$$

$$\dot{p} = \beta_p((\gamma - 1)\bar{y} + \delta(e - p) + \bar{u})$$

$$\dot{\beta_\epsilon} = \beta_{\beta_\epsilon}(v(\beta_\epsilon) - \beta_\epsilon), v(\beta_\epsilon) = \beta_p\delta(1 + \frac{1}{\alpha_r\beta_\epsilon}) \tag{6}$$

All of the β's are sensitivity parameters, and the exchange rate and price level changes are determined by a few more relationships. The equations are derived by Asada et. al by modifying some of Dornbusch's assumptions. For example, the $v(\beta_\epsilon)$ is introduced by Asada to consider the relationship between "the believed and the actual adjustment speed of exchange rates" (p. 226). The time series plots of this model show an overshooting behavior consistent with the Dornbusch model and some oscillation around the equilibrium that looks chaotic!

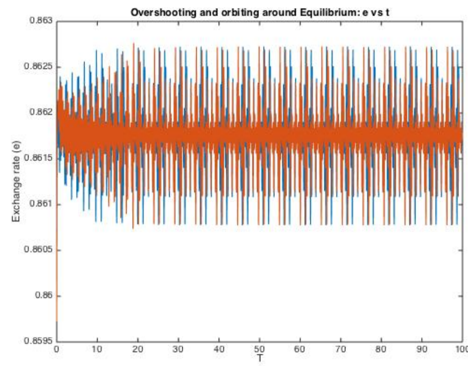

Figure 11: e vs t. Overshooting occurs quickly, then violent oscillations around the equilibrium. Initial conditions vary by .01

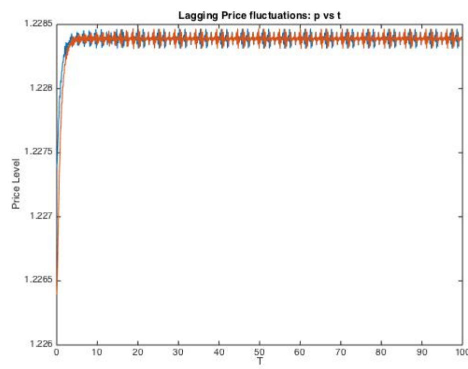

Figure 12: p vs t. Overshooting occurs quickly, then violent oscillations around the equilibrium. Initial conditions vary by .01

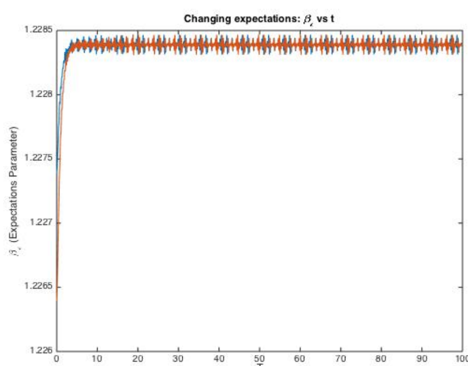

Figure 13: β_ϵ vs t. Overshooting occurs quickly, then violent oscillations around the equilibrium. Initial conditions vary by .01

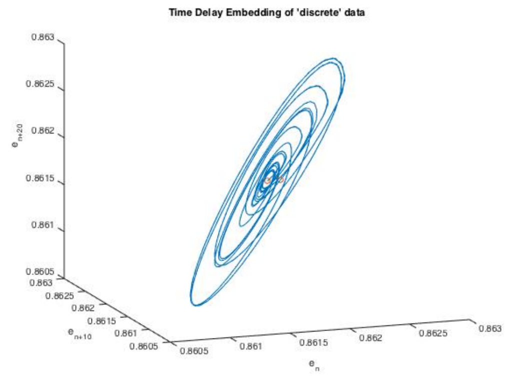

Figure 14: Time Delay Embedding using e_n, e_{n+1}, e_{n+2}

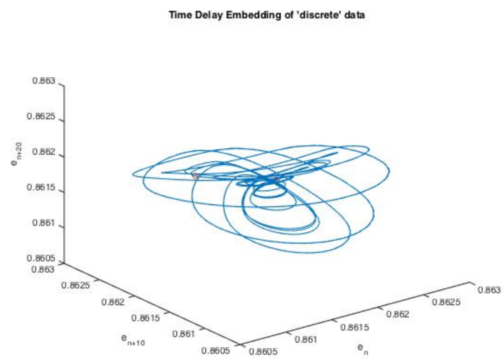

Figure 15: Time Delay Embedding using e_n, e_{n+10}, e_{n+20}

Out of curiosity, I looked at the "discrete" values of the exchange rate (continuous relationship calculated numerically so not really continuous, but the ODE system is meant to be interpreted continuously). The spacing between "discrete" values is 1 and 10 indices.

The time delay pattern starts looking like a rose bud with one time unit separation, at 10 time units the petals start to blossom or spread, and finally when the time step between observations is 100 units the pattern is a large 3 dimensional tangle.

Unfortunately, this model doesn't show chaotic behavior. The attractor is a straight line in 3 dimensions. Using a modified code provided by Professor Barnett (lyapflow.m), the Lyapunov exponents with these parameters are calculated to be all negative ($h_1 = -.572, h_2 = -4.728, h_3 = -13.891$). The parameters can be manipulated to give rise to positive exponents, but this occurs by making β_e too large. The model breaks down well before the Lyapunov exponents are positive (points shoot off to infinity or negative infinity). Asada et. al find chaotic attractors in the 8D, 10D, and 12D models that take a few chapters to explain.

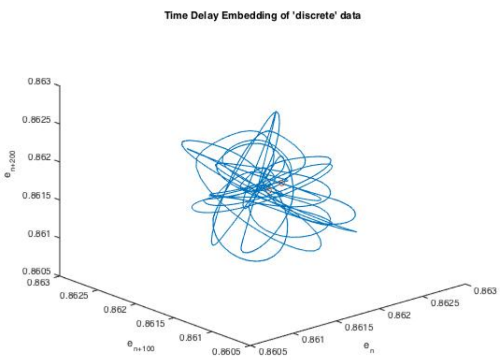

Figure 16: Time Delay Embedding using $e_n, e_{n+100}, e_{n+200}$

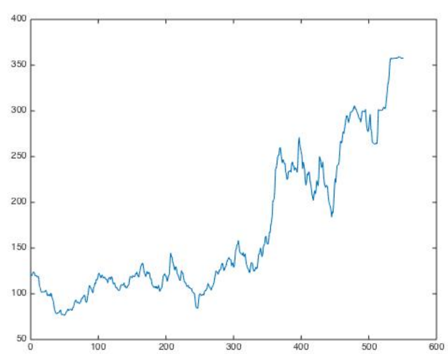

Figure 17: Real Data Time Series: US/Japan. Time units are 1 month

5 Chaos in Real Data?

As seen in the time delay plots of the dollar/yen exchange rate, there doesn't seem to be an underlying pattern.

The time delay plots indicate that the best guess of the next exchange rate is the current exchange rate and some noise. I tried looking at the deviation of the current exchange rate as a new variable $w_t = e_t - \bar{e}_t$, where $\bar{e}_t \approx \frac{1}{9}\sum_{i=t-4}^{t+4} e_i$. This resulted in an incomprehensible blob. Even thought the blob doesn't show a clear pattern, it is more promising because it suggests that maybe there is some relationship, we just can't tell what it is. The relation is possibly more than 3 dimensions and I could be omitting some important variable like prices, or interest rates.

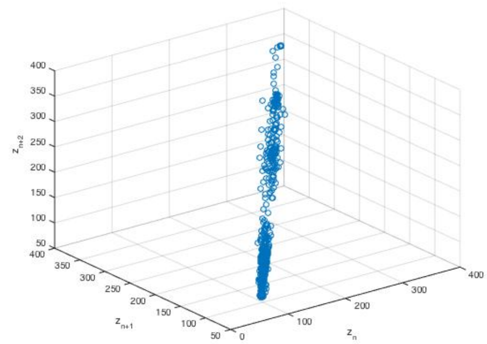

Figure 18: Real Data Time Delay Embedding: US/Japan. e_t, e_{t+1}, e_{t+2}. Time units are 1 month. Relationship is a noisy $x = y = z$ relationship with $x = z$ a little bit less accurate.

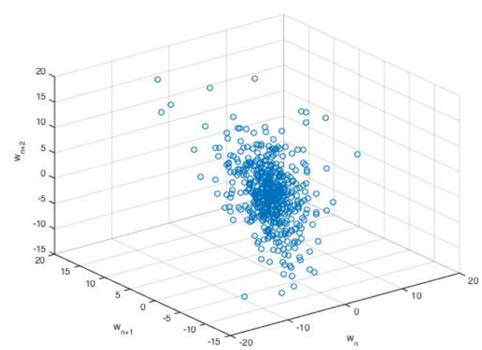

Figure 19: Real Data Time Delay Embedding: US/Japan. w_t, w_{t+1}, w_{t+2}. Time units are 1 month and $w_t = e_t - \bar{e}_t$

References

[1] Alligood, Sauer, Yorke. *Chaos: An Introduction to Dynamical Systems.* Springer-Verlag 1996. Chapters 1-13.

[2] Asada, Toichiro; Carl Chiarella, Peter Flaschel, Reiner Franke. *Open Economy Macro-Dynamics: An Integrated Disequilibrium Approach.* Springer-Verlag Berlin Heidelberg 2003. Chapter 5.

[3] Barnett, Alex. Dartmouth College Department of Mathematics.

[4] Brock W., Hsieh D., LeBaron B. *Nonlinear Dynamics, Chaos, and Instability.* MIT Press 1991.

[5] De Grauwe P. and Dewachter H. "A Chaotic Model of the Exchange Rate: The Role of Fundamentalists and Chartists." *Open economies review 4*: 351-379. Kluwer Academic Publishers 1993.

[6] Dornbusch, Rudiger. "Expectations and Exchange Rate Dynamics." Massachusetts Institute of Technology. *Journal of Political Economy.* December 1976.

[7] Engel, Mark, West. "Factor Model Forecasts of Exchange Rates." September 2010.

[8] Federici D. and Gandolfo G. "Chaos and the exchange rate." *Journal of International Trade & Economic Development 11:2 111-142* University of Rome 'La Sapienza', Italy. 2002.

[9] Giavazzi, Francesco. "A Primer on Differential Equations." Universita' Bocconi September 2007.

[10] Karaguler, Turban. "Chaos Theory and Exchange Rate Problem." Silesian University.

[11] Lopez-Suarez C.F. and Rodriguez-Lopez J.A. "Nonlinear Exchange Rate Predictability." *Journal of International Money and Finance.* September 2011.

[12] Sargent, Thomas J. *Dynamic Macroeconomic Theory* Harvard University Press 1987.

[13] Vandrovych V. "Nonlinearities in Exchange-Rate Dynamics: Chaos?" Brandeis University. November 4, 2006.

[14] Phase Plane Matlab code from http://www-users.math.umd.edu/ petersd/246/matlabode2.html

[15] US/Japan Exchange rate data from http://fxtop.com/en/historical-exchange-rates

[16] Template: https://www.overleaf.com/latex/templates/quantum-hall-effect-report-template/hqsgsyxsvphs

6 MATLAB

6.1 One Dimensional Map

```
% chaos 1d ex rate
% Daniel Salas
% M53 Project code
% November 23, 2015

clear
clc
close all;

% bifurcation plots take a lot of time...so set these to zero if you don't
% want to see them
bifgamma = 0;
bifbeta = 0;
bifalpha = 0;
bifratio = 0;
bifratio2 = 1;

N = 1100; % N-1 = number of iterations
e = zeros(N,1); % array for exchange rates
e(1) = .9; % initial exchange rate
B = 4;  % beta
g = 26; % gamma
ee = 1; % equilibrium exchange rate
a = 6;  % alpha

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bifurcation Plot of Alpha
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if bifalpha == 1
B = 4;
a = 6;
g = 26;

f = @(e) ((B+g).*ee.*e-g.*e.^2-a.*ee)./(2.*B.*e)+sqrt(((B+g).*ee.*e-g.*e.^2-a.*
for i = 1:N-1
    e(i+1) = f(e(i));
end
figure(1)
scatter(a.*ones(200,1),e(N-199:N),'.k')
xlabel('\alpha')
ylabel('e')
title('Bifurcation plot: e vs \alpha')
hold on

for a = 0.01:.01:12
```

```
f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
scatter(a*ones(200,1),e(N-199:N),'.k')
end
hold off
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Time Series Plot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


B = 4;
a = 6;
g = 26;
e1 = zeros(N,1);
e1(1) = .9;
e2 = zeros(N,1);
e2(1) = .9+10^(-8);
f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e1(i+1) = f(e1(i));
    e2(i+1) = f(e2(i));
end
figure(2)
plot(0:150,e1(1:151))
hold on
plot(0:150,e2(1:151))
xlabel('t (period)')
ylabel('e (exchange rate)')
title('Exchange rate discrete in time')
legend('e_0 = .9','e_0 = .9+10^{-8}')
hold off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculating the Lyapunov Exponent
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


figure(3)
plot(0:100,log(abs(e2(1:101)-e1(1:101))))
title('Calculating Lyapunov Exponent')
xlabel('t (period)')
ylabel('ln|e_{.9}(t) - e_{.9+10^{-8}}(t)|')
hold on
y2 = log(abs(e2(51)-e1(51)));
```

```
y1 = log(abs(e2(1)-e1(1)));
plot([0 50],[y1 y2]);
hold off
slope = (y2-y1)/50

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Time Delay Embedding
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


 figure(4)
 scatter(e1(1:N-1),e1(2:N))
 hold on
 scatter(e2(1:N-1),e2(2:N),'.r')
 title('Time Delay Embedding')
 xlabel('e_{n}')
 ylabel('e_{n+1}')
 legend('e_0 = .9','e_0 = .9+10^{-8}')
 hold off


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bifurcation Plot of Beta
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 if bifbeta == 1

B = 4;
a = 6;
g = 26;

f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
figure(5)
scatter(B.*ones(200,1),e(N-199:N),'.k')
xlabel('\beta')
ylabel('e')
title('Bifurcation plot: e vs \beta')
hold on

for B = 0.1:.01:15


f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
scatter(B*ones(200,1),e(N-199:N),'.k')
```

```
end
hold off
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bifurcation Plot of Gamma
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if bifgamma == 1
B = 4;
a = 6;
g = 26;

f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end

figure(6)
scatter(g.*ones(200,1),e(N-199:N),'.k')
xlabel('\gamma')
ylabel('e')
title('Bifurcation plot: e vs \gamma')
hold on

for g = 14:.05:40


f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
scatter(g*ones(200,1),e(N-199:N),'.k')
end
hold off
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bifurcation Plot of Gamma/Beta
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if bifratio == 1

B = 4;
a = 6;
g = 4*B;

f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
```

```
figure(7)
scatter(B.*ones(200,1),e(N-199:N),'.k')
xlabel('\beta')
ylabel('e')
title('Bifurcation plot: e vs \beta (with \gamma fixed = 4*\beta)')
hold on

for B = 0.1:.05:35
g = 4*B;

f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
scatter(B*ones(200,1),e(N-199:N),'.k')
end
hold off



end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bifurcation Plot of Gamma/Beta/Alpha
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if bifratio2 == 1

B = 4;
a = 6;
g = 4*B;

f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end

figure(8)
scatter(B.*ones(200,1),e(N-199:N),'.k')
xlabel('\beta')
ylabel('e')
title('Bifurcation plot: e vs \beta (with \gamma and \alpha fixed rel to \beta)
hold on

for B = .1:.01:25
g = 4*B;
a = 6/4*B; % keep ratio between alpha,beta, and gamma fixed (hint: no chaos)
```

```
f = @(e) ((B+g)*ee*e-g*e^2-a*ee)/(2*B*e)+sqrt(((B+g)*ee*e-g*e^2-a*ee)^2+4*B*e*a
for i = 1:N-1
    e(i+1) = f(e(i));
end
scatter(B.*ones(200,1),e(N-199:N),'.k')
end
hold off


end
```

6.2 Giavazzi's 2D map: Phase Plane Code

```
% Modified Code from http://www-users.math.umd.edu/~petersd/246/matlabode2.html
clear; close all; clc
% Daniel Salas
% Phase Plane M53 project
% November 23, 2015
h = .01;
m = 1;
d = 2;
l = .5;
u1 = -d/2 + sqrt(d^2+4*d/l)/2;
u2 = -d/2 - sqrt(d^2+4*d/l)/2;
yo = (1+h).*[l*u1;1];

f = @(t,y) [d*(y(2)-y(1));(y(1)-m)/l];

figure(1)
vectfield(f,m-2.5:.25:m+2.5,m-2.5:.25:m+2.5)
xlabel('p')
ylabel('e')
title('Saddle Behavior: Phase Plane and Unstable Manifold')
hold on
plot([m-2.5,m+2.5],[0,0],'--k')
plot([0,0],[m-2.5,m+2.5],'--k')
plot(10.*[-l*u1,l*u1]+m,10.*[-1,1]+m,'-k')
plot(10.*[-l*u2,l*u2]+m,10.*[-1,1]+m,'-k')
for y20=0.4:0.1:1.2
  [ts,ys] = ode45(f,[0,10],[l*u1;y20]+m);
  plot(ys(:,1),ys(:,2))
  scatter(ys(1,1),ys(1,2),'o')
end
axis([m-2.5 m+2.5 m-2.5 m+2.5])
hold off

figure(2)
vectfield(f,m-2.5:.25:m+2.5,m-2.5:.25:m+2.5)
xlabel('p')
ylabel('e')
```

```
title('Saddle Behavior: Phase Plane and Stable Manifold')
hold on
plot([m-2.5,m+2.5],[0,0],'--k')
plot([0,0],[m-2.5,m+2.5],'--k')
plot(10.*[-l*u1,l*u1]+m,10.*[-1,1]+m,'-k')
plot(10.*[-l*u2,l*u2]+m,10.*[-1,1]+m,'-k')
for y20=0.4:0.1:1.2
  [ts,ys] = ode45(f,[0,10],[l*u2;y20]+m);
  plot(ys(:,1),ys(:,2))
  scatter(ys(1,1),ys(1,2),'o')
end
axis([m-2.5 m+2.5 m-2.5 m+2.5])
hold off

%vectfield  vector field for system of 2 first order ODEs
%   vectfield(func,y1val,y2val) plots the vector field for the system of
%   two first order ODEs given by func, using the grid of y1val and
%   y2 values given by the vectors y1val and y2val. func is either a
%   the name of an inline function of two variables, or a string
%   with the name of an m-file.
%   By default, t=0 is used in func. A t value can be specified as an
%   additional argument: vectfield(func,y1val,y2val,t)
function vectfield(func,y1val,y2val,t)
if nargin==3
  t=0;
end
n1=length(y1val);
n2=length(y2val);
yp1=zeros(n2,n1);
yp2=zeros(n2,n1);
for i=1:n1
  for j=1:n2
    ypv = feval(func,t,[y1val(i);y2val(j)]);
    yp1(j,i) = ypv(1);
    yp2(j,i) = ypv(2);
  end
end
quiver(y1val,y2val,yp1,yp2,'r');
axis tight;

%vectfieldn  vector field for system of 2 first order ODEs,
%            with arrows normalized to the same length
%   vectfield(func,y1val,y2val) plots the vector field for the system of
%   two first order ODEs given by func, using the grid of y1val and
%   y2 values given by the vectors y1val and y2val. func is either a
%   the name of an inline function of two variables, or a string
%   with the name of an m-file.
%   By default, t=0 is used in func. A t value can be specified as an
%   additional argument: vectfield(func,y1val,y2val,t)
function vectfieldn(func,y1val,y2val,t)
```

```
if nargin==3
   t=0;
end
n1=length(y1val);
n2=length(y2val);
yp1=zeros(n2,n1);
yp2=zeros(n2,n1);
for i=1:n1
   for j=1:n2
      ypv = feval(func,t,[y1val(i);y2val(j)]);
      yp1(j,i) = ypv(1);
      yp2(j,i) = ypv(2);
   end
end
len=sqrt(yp1.^2+yp2.^2);
quiver(y1val,y2val,yp1./len,yp2./len,.6,'r');
axis tight;
```

6.3 Asada's Nonlinear Map

Plots

```
% Models 2D
clc
clear
close all;

h = .001;
m = 2;
d = 3;
rf = 0;
eo = 1.1;
ar = 3;
u = 1.3;
g = .9;
yb = 2;
Bp = .3;
Be = 10;
BBe = 10;
ph = .8;





% Equilibrium points
z1 = Bp*d/2+sqrt((Bp*d)^2+4*Bp*d/ar)/2;
y1 = (rf+z1*(eo-(-u/d-(g-1)*yb/d))-(ph*yb-m)/ar)/(1/ar+z1);
x1 = y1 - u/d - (g-1)*yb/d;

z2 = Bp*d/2-sqrt((Bp*d)^2+4*Bp*d/ar)/2;
```

```
y2 = (rf+z2*(eo-(-u/d-(g-1)*yb/d))-(ph*yb-m)/ar)/(1/ar+z2);
x2 = y2 - u/d - (g-1)*yb/d;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


Xo1 = [x1;y1;z1]-h;
Xo2 = [x2;y2;z2]-h;
Xo1h2 = Xo1 -h;
Xo2h2 = Xo2-h;
t = [0:.01:100];
f = @(t,X) [Be*(rf+X(3)*(eo-X(1))-(ph*yb+X(2)-m)/ar);...
            Bp*((g-1)*yb+d*(X(1)-X(2))+u);...
            BBe*(Bp*d*(1+1/(ar*X(3)))-X(3))];
[T1,X1] = ode45(f,t,Xo1);
[T2,X1h2] = ode45(f,t,Xo1h2);
N = numel(X1(:,1));

dX1 = zeros(N,3);

for i = 1:N
    dX1(i,:) = f(1,X1(i,:));
end

figure(1)
plot(T1,X1(:,1))
hold on
plot(T1,X1h2(:,1))
xlabel('T')
ylabel('Exchange rate (e)')
title('Overshooting and orbiting around Equilibrium: e vs t')
hold off

figure(2)
plot(T1,X1(:,2))
hold on
plot(T1,X1h2(:,2))
xlabel('T')
ylabel('Price Level')
title('Lagging Price fluctuations: p vs t')
hold off

figure(3)
plot(T1,X1(:,2))
hold on
plot(T1,X1h2(:,2))
xlabel('T')
ylabel('\beta_{\epsilon} (Expectations Parameter)')
title('Changing expectations: \beta_{\epsilon} vs t')
hold off
```

```
% Time Delay Embedding
M = 900; % how many points
sp = 100; % spacing between points
figure(4)
plot3(X1(N-(M+2*sp):N-2*sp,1),X1(N-(M+sp):N-sp,1),X1(N-M:N,1))
hold on
scatter3([X1(N-(M+2*sp),1) X1(N-2*sp,1)],[X1(N-(M+sp),1) X1(N-sp,1)],[X1(N-M,1)
xlabel('e_{n}')
ylabel('e_{n+100}')
zlabel('e_{n+200}')
title('Time Delay Embedding of ''discrete'' data')
hold off



% figure(9)
% plot(T,X1(:,1))
% xlabel('T')
% ylabel('Exchange rate (e)')
% title('Close to stable manifold (roundoff error): e vs p')
```

Modified lyapflow codes:

```
% lyapunov exponents in a flow in R^3, eg Lorenz attractor
% Needs the function lorenz_time1map.m which returns the time-1 map
% barnett 11/19/07
clear
close all;
clc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% My additions
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
h = .01; % .01
m = 2; %2
d = 3; %3
rf = 0; % 0
eo = 1.1; %1.1
ar = 3; %3
u = 1.3; % 1.3
g = .9; % .9
yb = 2; % 2
Bp = .1; %.1
Be = 10; %10, good .001
BBe = 10; %10
ph = .8; %.8

c = [m;
     d;
```

```
      rf;
      eo;
      ar;
      u;
      g;
      yb;
      Bp;
      Be;
      BBe;
      ph];

% Equilibrium points
z1 = Bp*d/2+sqrt((Bp*d)^2+4*Bp*d/ar)/2;
y1 = (rf+z1*(eo-(-u/d-(g-1)*yb/d))-(ph*yb-m)/ar)/(1/ar+z1);
x1 = y1 - u/d - (g-1)*yb/d;

z2 = Bp*d/2-sqrt((Bp*d)^2+4*Bp*d/ar)/2;
y2 = (rf+z2*(eo-(-u/d-(g-1)*yb/d))-(ph*yb-m)/ar)/(1/ar+z2);
x2 = y2 - u/d - (g-1)*yb/d;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



Xo1 = [x1;y1;z1]-h;
Xo1h2 = [x1;y1;z1]-2*h;
Xo2 = [x2;y2;z2]+h;

t = [0:.01:100];
F = @(t,X) [Be*(rf+X(3)*(eo-X(1))-(ph*yb+X(2)-m)/ar);...
              Bp*((g-1)*yb+d*(X(1)-X(2))+u);...
              BBe*(Bp*d*(1+1/(ar*X(3)))-X(3))];



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

   % vec flow field
yo = Xo1;                                  % IC
[ts, ys] = ode45(F, [0 300], yo); % numerically solve in t domain

pts = numel(ys(:,1));
figure;plot3(ys(pts-100:pts,1), ys(pts-100:pts,2), ys(pts-100:pts,3), '-'); axi

% test lorenz time-1 map
[x, J] = exlorenz_time1map(yo,c)



% Measure Lyapunov exponents...
% ------ Re-orthogonalizing version, repeated averaging
```

```
M = 20;                                  % how many averaging loops
N = 50;                                       % how many its per meas step
x = Xo1;
h = zeros(3,1);                          % place to store averaged lyap exps
for m=1:M
  J = eye(3);                        % Id is where Jacobean starts
  for n=1:N
    [x Jx] = exlorenz_time1map(x,c);
    J = Jx*J;                              % update Jacobean
    [Q,R] = qr(J);                         % re-orthogonalize
    J = Q*diag(diag(R));                   % but keep them correct lengths
  end
  rN = abs(diag(R))                        % print out progress
  h = h + log(rN)/N;                       % sum up lyap exps from each run
end
h = h/M                              % final answer: mean (over runs) of lyap exps

% you can see the middle lyap exp is v close to 0 - in fact all flows have
% a zero exponent if they are bounded and hit no equilibrium points.

%exlorenz_time1map.m

function [x, DFx] = exlorenz_time1map(xo,c)
% evolve Lorenz flow for 1 time unit, including finding the jacobean matrix

     m=c(1);
     d=c(2);
     rf=c(3);
     eo=c(4);
     ar=c(5);
     u=c(6);
     g=c(7);
     yb=c(8);
     Bp=c(9);
     Be=c(10);
     BBe=c(11);
     ph=c(12);

F = @(t,X)  [Be*(rf+X(3)*(eo-X(1))-(ph*yb+X(2)-m)/ar);...
             Bp*((g-1)*yb+d*(X(1)-X(2))+u);...
             BBe*(Bp*d*(1+1/(ar*X(3)))-X(3))]; % vec flow field
Df = @(y) [-Be*y(3) -Be/ar Be*(eo-y(1));...
           Bp*d -Bp*d  0;...
           0 0 BBe*(-Bp*d/(ar*y(3)^2)-1)]; % DF at y

J0 = eye(3);   % initial Jac matrix

% 12-component ODE flow given by 3 components of solution and 9 components
% of the J matrix (J satisfies the ODE dJ/dt = Df.J)
G = @(t,z)  [F(t,z(1:3,:)); Df(z(1:3,:))*z(4:6,:); Df(z(1:3,:))*z(7:9,:); ...
```

```
            Df(z(1:3,:))*z(10:12,:) ];

[ts, xs]  = ode45(G, [0 1], [xo; J0(:)],odeset('abstol',1e-3));
% numerically solve in t domain
x = reshape(xs(end,1:3), [3,1]);     % extract the answer at the final time t=1
J = xs(end,4:end);   % same for the J components
DFx = reshape(J,[3 3]);    % send J out as a 3x3 matrix.
```

6.4 Real Data Plots

```
% math exp with dollar yen
% data from http://fxtop.com/en/historical-exchange-rates.php?A=1&C1=USD&C2=JPY
 clear
 clc
 close all;

M = [122.50176
 120.022268
 120.175646
 123.128729
 123.397559
 123.730861
 120.87394
 119.617625
 120.340684
 118.671782
 118.279355
 119.263228
 116.282652
 107.989769
 107.283792
 102.966938
 101.723027
 102.056261
 101.763971
 102.530867
 102.354868
 102.019534
 103.942137
 103.385798
 100.028352
 97.781882
 99.203056
 97.930827
 99.695216
 97.366185
 101.018171
 97.904468
 94.869816
```

93.117683
89.047084
83.612486
81.017717
78.981996
78.171123
78.692963
78.985946
79.240284
79.72825
81.296217
82.477133
78.458511
76.967734
77.811325
77.470746
76.653714
76.79352
76.995641
79.402519
80.448579
81.174937
83.391242
81.724659
82.625214
82.628231
83.290905
82.513022
81.794824
84.376573
85.332527
87.541869
90.919565
92.150954
93.48913
90.678521
90.2073
91.31662
89.814057
89.170045
90.376697
91.44315
94.838413
94.473409
96.599155
96.609295
98.73546
97.835667
92.542258
90.421242

91.161063
96.819209
100.120255
106.617581
109.276898
106.826127
106.906652
104.33583
102.580483
100.878621
107.116111
107.810744
112.259666
110.949462
115.941668
115.007921
116.747369
121.587087
122.632672
120.804112
118.882925
117.231931
120.546543
120.449895
117.183837
117.312999
118.670875
117.083237
115.941304
115.664792
114.722965
111.758494
117.025997
117.274318
117.910723
115.525045
118.579271
118.447248
114.904814
111.029225
110.625366
111.938081
108.692609
106.66044
107.312797
105.181809
104.92694
103.360992
103.781079
104.768539

108.89041
110.086791
110.497178
109.322561
109.439853
112.015272
107.717074
108.56468
106.582616
106.341606
107.801899
109.249395
109.580249
114.961343
118.829985
118.701607
118.372443
117.279024
119.94969
118.610842
119.377724
118.738412
122.034356
121.490168
123.915857
120.708722
118.955994
118.045675
123.329232
126.364753
130.745773
131.021657
133.594942
132.600256
127.258888
122.362743
121.280899
118.778284
121.4507
124.573848
122.242748
121.814769
123.753905
121.346998
116.178879
116.783487
112.109344
108.90605
108.452308
106.758782

108.124842
107.903854
106.10919
108.279072
105.529986
106.38179
109.449633
105.09912
102.575551
104.700703
106.026134
107.010158
113.251455
119.538746
120.764182
122.050127
119.721926
119.639197
116.716847
113.194605
117.144498
120.346141
120.775036
134.571939
144.679549
140.629012
140.081734
134.878018
132.228814
128.999937
125.689921
129.526867
129.509902
125.181885
120.961006
120.725384
117.819213
115.235649
114.286681
119.119025
125.555351
122.669101
122.929094
117.76466
113.911394
112.267102
112.357758
109.870911
107.865047
109.243838

108.860999
106.327634
107.200532
105.885116
105.61657
105.523571
101.869461
101.847809
100.731317
100.432867
94.612133
87.23088
84.555966
85.024827
83.841584
90.601336
98.239413
99.732727
100.148581
98.002763
98.471022
98.833657
99.886705
98.582343
102.646814
103.766884
103.327049
105.093329
106.310547
111.386134
109.95955
107.854021
106.958025
105.391873
103.803775
107.593089
107.304622
110.314339
112.30052
116.978721
120.907447
125.052771
123.992987
123.880529
121.168197
122.667609
126.300924
125.688681
126.915488
130.823213

133.434685
132.875089
127.565603
125.151675
128.066785
129.630486
130.710026
134.550988
136.825919
137.903912
139.865458
138.067781
137.242536
137.310958
130.373455
133.716559
133.855728
128.992193
129.797831
138.717491
147.519992
149.155896
153.785213
153.810891
158.390939
153.198067
145.580979
144.94094
143.611349
143.580133
142.120616
145.174348
141.249727
140.533841
144.009306
137.835909
132.069783
130.302498
127.701089
127.032378
123.615549
123.174428
129.031436
134.43109
133.707458
133.087382
127.196178
124.725675
124.998402
127.174456

129.239742
127.218386
128.23464
135.324063
143.343953
143.229484
147.645525
150.248765
144.467267
140.507903
142.917926
151.415226
153.495171
154.860579
162.269232
162.869942
156.327132
154.664649
154.096102
158.624531
167.620251
166.84966
175.049839
178.598838
184.553017
200.29578
202.688562
203.943295
214.814032
236.739103
237.188061
241.426105
248.87877
251.583255
251.540621
258.399539
260.200188
253.758022
247.775761
243.01291
246.743635
245.074542
242.332984
242.756627
233.162972
230.459206
224.950794
225.390121
233.604967
233.621819

234.189389
234.91624
232.816753
242.466437
244.293054
240.352217
240.047592
234.763035
237.448114
237.779223
235.913568
232.725418
241.605633
264.716095
270.987332
262.585421
258.634407
254.607594
251.011234
236.600951
244.182826
240.793183
235.377297
224.189587
218.603252
223.294665
231.314074
229.425498
233.479311
232.146941
223.807552
220.258967
214.81364
208.705727
205.368235
201.940441
209.111974
212.882023
209.045043
214.442584
223.929536
220.759039
217.991732
228.113398
250.185617
248.261321
243.861834
237.677148
239.673798
244.352925

230.000919
222.12764
217.7105
216.236107
218.783691
218.187954
215.892145
206.301069
200.305081
197.262452
195.888825
191.333127
183.796296
189.938206
188.394057
199.513293
214.099014
225.942913
221.502923
231.462133
239.972164
240.897659
240.970512
244.582524
254.706805
266.729604
266.460586
264.609633
272.806813
277.36373
275.059771
280.079147
284.678208
290.773953
294.618223
294.888861
290.773709
287.145824
290.422104
294.506534
298.978198
298.859769
298.92309
300.322251
301.365302
304.489592
305.503218
302.206506
302.094604
299.449339

297.792211
296.004382
293.312742
291.132407
291.928226
287.502341
291.489179
299.442719
299.958486
299.576983
299.134599
299.006078
301.771725
289.693247
282.471106
278.37914
277.123647
281.632681
290.591913
296.363051
279.822637
277.573853
266.116993
265.055586
264.679738
263.73707
263.599277
264.30301
265.032037
263.770016
276.677956
301.569801
300.474877
300.391943
300.862869
300.764135
300.8179
300.611792
302.533537
304.067943
303.180602
302.291925
305.107897
312.550486
320.085217
328.751364
331.134762
338.362326
351.698443
357.503614

```
357.425501
357.055715
357.279053
357.28324
357.312873
357.961206
357.665591
357.561898
357.651666
357.882079
358.097067
359.326465
358.802558
358.890533
358.029968
357.581236
357.613592
357.743517
357.645345];



z1 = M(1:numel(M)-2);
z2 = M(2:numel(M)-1);
z3 = M(3:numel(M));
w = zeros(numel(M)-8,1);

for i = 5:numel(M)-4
    a = (M(i-4)+M(i-3)+M(i-2)+M(i-1)+M(i)+M(i+1)+M(i+2)+M(i+3)+M(i+4))/9;
    w(i-4) = M(i)-a;
end
sp = 1;
w1 = w(1:numel(w)-2*sp);
w2 = w(1+sp:numel(w)-sp);
w3 = w(1+2*sp:numel(w));

figure(1)
scatter3(z1,z2,z3)
xlabel('z_{n}')
ylabel('z_{n+1}')
zlabel('z_{n+2}')

figure(2)
plot(0:numel(M)-1,M)

figure(3)
scatter3(w1,w2,w3)
xlabel('w_{n}')
ylabel('w_{n+1}')
zlabel('w_{n+2}')
```

```
numel(w)
numel(M(5:numel(M)-4))

figure(4)
scatter3(M(5:numel(M)-4),w,M(6:numel(M)-3))
xlabel('z_{n}')
ylabel('w_{n}')
zlabel('z_{n+1}')

numel(M)
```