Solving the Solow Growth Model by Linear Approximation

Fabrice Collard

1 The model

We consider the central problem of a planner that aims at maximizing the discounted sum of expected utility of a infinitely lived household with respect to its consumption :

$$\max E_0 \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\sigma}}{1-\sigma}$$

This household can either consume C_t at time t, or save an amount of good that it invests to form capital, it will use in production according to a technology represented by the production function $f(k_t)$, where k_t denotes physical capital. Its intertemporal budget constraint then resumes to :

$$k_{t+1} = A_t k_t^{\alpha} + (1-\delta)k_t - c_t$$

A_t is an exogenous stochastic shock affecting technology. A_t is assumed to follow a stationary AR(1) process :

$$\log(A_t) = \rho \log(A_{t-1}) + (1-\rho)\log(\overline{A}) + \varepsilon_t$$

with $|\rho| < 1$. $\overline{A}$ denotes the unconditional mean of A_t. ε_t is a Gaussian white noise with zero mean and standard deviation σ_ε : $\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$

The first order conditions are given by[1] :

$$c_t^{-\sigma} = \lambda_t \quad (1)$$

$$\lambda_t = \beta E_t \left[\lambda_{t+1} \left(\alpha \frac{y_{t+1}}{k_{t+1}} + 1 - \delta \right) \right] \quad (2)$$

$$k_{t+1} = i_t + (1-\delta)k_t \quad (3)$$

$$y_t = c_t + i_t \quad (4)$$

$$y_t = A_t k_t^{\alpha} \quad (5)$$

$$\lim_{t \to \infty} E_t \left[\beta^t \lambda_t k_{t+1} \right] = 0 \quad (6)$$

The last condition is the transversality condition. It states that the discounted marginal value of capital tends to zero in the long run. In a finite horizon model, this would be interpreted as a zero saving behavior in the last period. Since the agent would be death in the next period, he has no incentive to accumulate so that $k_{T+1} = 0$.

[1] λ_t denotes the lagrangean multiplier associated to the law of motion of capital.

2 Solving the model by Farmer's method

The previous rational expectations non linear dynamic system admits no analytical solution, unless in very special cases. In order to solve it, we use a log–linear approximation of the system around the deterministic steady state. This later is given by :

$$s_k = \left(\frac{k}{y}\right)^{\star} = \frac{\alpha\beta}{1-\beta(1-\delta)} \tag{7}$$

$$s_i = \left(\frac{i}{y}\right)^{\star} = \delta s_k \tag{8}$$

$$s_c = \left(\frac{c}{y}\right)^{\star} = 1 - s_i \tag{9}$$

$$\tag{10}$$

2.1 The log–linearization method

Each of the first order condition can be written as :

$$E_t\varphi(x) = 0$$

where x is a vector of control and state variables, dated according to the case in t or $t+1$. Let us denote the steady state value of x by $x^{\star}$. It is then noteworthy that $\varphi(.)$ satisfies :

$$\varphi(x^{\star}) = 0$$

The log–linear approximation of $\varphi(.)$ is then given by :

$$E_t\varphi(x) = E_t\left\{\varphi(x^{\star}) + \sum_i \left(\frac{\partial\varphi(x)}{\partial\log(x_i)}\right)_{x=x^{\star}} (\log(x_i) - \log(x_i^{\star})) + O(\|x\|^2)\right\}$$

where $O(\|x\|^2)$ converges to zero in probability.

Let $\hat{x}_i$ be the percentage deviation of x_i relative to its steady state value $x_i^{\star}$, $\hat{x}_i$ is thus defined by :

$$\hat{x}_i = \log\left(\frac{x_i}{x_i^{\star}}\right) \simeq \frac{x_i - x_i^{\star}}{x_i^{\star}}$$

Since $\hat{x}_i$ can be interpreted as the percentage deviation of x_i relative to its steady state, the coefficient beside this deviation can be interpreted as elasticities.

Using the fact that $\varphi(x^{\star}) = 0$ and neglecting the terms of order greater or equal than 2, the log–linear approximation is given by :

$$E_t\varphi(x) \simeq E_t\left\{\sum_i \left(\frac{\partial\varphi(x)}{\partial\log(x_i)}\right)_{x=x^{\star}} \hat{x}_i\right\}$$

Thus, the log–linear approximation of the growth model is given by :

$$-\sigma\hat{c}_t = \hat{\lambda}_t \tag{11}$$

$$\hat{y}_t = s_c\hat{c}_t + s_i\hat{\imath}_t \tag{12}$$

$$\hat{k}_{t+1} = \delta\hat{\imath}_t + (1-\delta)\hat{k}_t \tag{13}$$

$$\hat{\lambda}_t = E_t\hat{\lambda}_{t+1} + \frac{\alpha\beta}{s_k}E_t\hat{y}_{t+1} - \frac{\alpha\beta}{s_k}E_t\hat{k}_{t+1} \tag{14}$$

$$\hat{y}_t = \hat{a}_t + \alpha\hat{k}_t \tag{15}$$

$$\hat{a}_t = \rho\hat{a}_{t-1} + \varepsilon_t \tag{16}$$

This can be rewritten according to the following matrix equations :

$$M_{cc}C_t = M_{cs}S_t \tag{17}$$

$$M_{ss0}S_{t+1} + M_{ss1}S_t = M_{sc0}C_{t+1} + M_{sc1}C_t + M_{se}E_{t+1} \tag{18}$$

Interpreting the previous system as a state–space system allows to assimilate equation (17) as the measurement equation of the system : It links the control variables, represented by the vector C_t, to the state variables, represented by the vector S_t. In our problem, the control variables vector is given by :

$$C_t = \{\hat{c}_t, \hat{y}_t, \hat{i}_t\}'$$

whereas the state variables vector is given by :

$$S_t = \{\hat{k}_t, \hat{A}_t, \hat{\lambda}_t\}'$$

The second equation can be interpreted as the state equation. It accounts for the dynamic link between control variables, state variables and the surprises, represented by[2] :

$$E_t = \{E_t\hat{x}_{t+1} - \hat{x}_{t+1}; \hat{\varepsilon}_t; \}; \text{ for } x \in \{k, \lambda, y\}$$

In fact, we will not use it in our code, since in the case considered no sunspots arise.

It must be emphasized that the state variables are ranked in a particular order, which is a key point in the solution method, as it will become clear in a moment. We first introduce the endogenous backward state, namely capital stock in our model, then the exogenous shocks, namely a_t, and finally the forward variables, namely the shadow price λ_t. This is essential since our method will rest on a partition of matrices, conditioned by the status of all variables.

2.2 Solving the rational expectations log–linearized model

The first step is to rewrite the model in a form that only involves state variables. This is done using equation (17) :

$$C_t = M_{cc}^{-1}M_{cs}S_t$$

Plugging this expression in (18), we get :

$$S_{t+1} = W_S S_t + W_E E_{t+1}$$

where :

$$W_S = -\left(M_{ss0} - M_{sc0}M_{cc}^{-1}M_{cs}\right)^{-1}\left(M_{ss1} - M_{sc1}M_{cc}^{-1}M_{cs}\right)$$

$$W_E = \left(M_{ss0} - M_{sc0}M_{cc}^{-1}M_{cs}\right)^{-1}M_{se}$$

Blanchard et Kahn [1980] established that existence and uniqueness of a solution path rely on a condition on the position of eigenvalues of W_S relative to the unit circle. Let N_B and N_F denote respectively the number of Backward looking state variables and Forward looking state variables[3]. Then let M_I and M_O be the number of eigenvalue Inside and Outside of the unit circle, then :

> If $N_B = M_I$ and $N_F = M_O$ then there exists a unique path that is the solution of the rational expectations problem, converging to the steady state of the model.

This configuration corresponds to a saddle path. It exists a unique λ_0 such that the path $\{\hat{\lambda}_t\}_{t=0}^{\infty}$ satisfies the transversality condition. In fact, the method we are using tries to find this value λ_0..

[2] $\hat{\varepsilon}_t$ denote the vector of innovations for the exogenous shocks.

[3] In our model $N_B = 2$ (k_t and a_t) and $N_F = 1$, (λ_t).

The diagonalization of W_S leads to :

$$W_S = P D P^{-1}$$

where D is the matrix of eigenvalues and P is the associated eigenvectors matrix. We first sort the eigenvalues by ascending order, and rearrange the eigenvector matrix according to the new order. This leads to the following partition of matrices P and P^{-1} :

$$P = \begin{pmatrix} P_{BB} & P_{BF} \\ P_{FB} & P_{FF} \end{pmatrix} ; \; P^{-1} = \begin{pmatrix} P^{\star}_{BB} & P^{\star}_{BF} \\ P^{\star}_{FB} & P^{\star}_{FF} \end{pmatrix}$$

This partition relies on the position of eigenvalues relative to the unit circle. Indeed, a B index means that the corresponding eigenvalue is of modulus less than one, whereas an index F means that the eigenvalue is outside of the unit circle.[4].

We then rewrite the system in an other basis, allowing to obtain a simple diagonal system. We thus define :

$$\tilde{S}_t = P^{-1} S_t$$

so that :

$$P^{-1} S_{t+1} = P^{-1} W_S P P^{-1} S_t + P^{-1} W_E E_{t+1}$$

thus :

$$\tilde{S}_{t+1} = D \tilde{S}_t + R E_{t+1}$$

The partition of R is in the lines of that of P :

$$R = \begin{pmatrix} R_{B:} \\ R_{F:} \end{pmatrix}$$

For the state vector :

$$\tilde{S}_t = \begin{pmatrix} \tilde{S}_{B;t} \\ \tilde{S}_{F;t} \end{pmatrix}$$

Thus the law of motion of forward looking variables can be written :

$$\tilde{S}_{F;t+1} = D_F \tilde{S}_{F;t} + R_{F:} E_{t+1}$$

The conditional expectation at time t of $\tilde{S}_{F;t+1}$ is given by :

$$E_t \tilde{S}_{F;t+1} = D_F \tilde{S}_{F;t}$$

since D_F is a diagonal matrix, iterating on the process leads to :

$$\tilde{S}_{F;t} = \lim_{j \to \infty} D_F^{-j} E_t \tilde{S}_{F;t+j}$$

$\tilde{S}_{F;t}$ is thus bounded if the limit of $D_F^{-j} E_t \tilde{S}_{F;t+j}$ is finite as $j \to \infty$. If the transversality condition is satisfied, we have[5] :

$$\lim_{j \to \infty} D_F^{-j} E_t \tilde{S}_{F;t+j} = 0$$

This implies :

$$\tilde{S}_{F;t} = P^{\star}_{FB} S_{B;t} + P^{\star}_{FF} S_{F;t} = 0$$

This equation allows us to define the initial condition on $S_{F;t}$ compatible with the transversality conditions and the initial conditions of predetermined variables:

$$S_{F;t} = (P^{\star}_{FF})^{-1} P^{\star}_{FB} S_{B;t}$$

[4] This is why the order in which variables are inserted in the system is important. When Blanchard and Kahn's conditions are satisfied, the partition corresponds to a decomposition between backward looking and forward looking state variables.

[5] Actually this condition correspond to ruling out the bubbles.

Defining $Q = \left(I \,\vdots\, -P^{\star}_{FF}{}^{-1} P^{\star}_{FB}\right)'$, we get :

$$E_t\left(\widehat{S}_{B,t+1}\right) = W_{S,B\cdot}\, Q\, \widehat{S}_{B,t} = M_{SS}\, \widehat{S}_{B,t}$$

Since the surprises on predetermined variables are null[6], the surprises only rely on the innovation of the exogenous shocks. Thus :

$$\widehat{S}_{B,t+1} = M_{SS}\, \widehat{S}_{B,t} + M_{SE}\, \widehat{\varepsilon}_{t+1}$$

where $M_{MSE} = R_{B\cdot} Q$.

The only thing that we still have to do is to obtain the solution for the measurement equation :

$$C_t = M_{cc}^{-1} M_{cs}\, Q\, S_{B,t} = \Pi\, S_{B,t}$$

Then, we are done !

2.3 Impulse response functions

Knowing the policy rules associated to each variable, it is possible to generate the dynamics of the model after a shock. This is the impulse response function analysis. They are obtained using the state space representation we obtained in the previous analysis.

Consider a 1% deviation shock in the technology shock. The instantaneous response of the state vector will be given by :

$$\widehat{S}_{B,t} = M_{SE} \times 1$$

We then just iterate on the state equation to get the response of the state vector at period $t + j$:

$$\widehat{S}_{B,t+j} = M_{SS} \widehat{S}_{B,t+j-1} \quad j > 0$$

The impulse response of the control variables are then given by :

$$\widehat{C}_{t+j} = \Pi\, \widehat{S}_{B,t+j}$$

2.4 Simulations and quantitative evaluation

The RBC literature insists on the comparison of moments generated by the model to their empirical counterpart. This is done by simulation. Using a random number generator, we get a sequence of innovations for the exogenous shocks. Then using the state equation, we generate a simulated path for the states :

$$\widetilde{S}_{B,t+1} = M_{SS}\, \widetilde{S}_{B,t} + M_{SE}\, \widetilde{\varepsilon}_{t+1}$$

where $\widetilde{\ }$ denote a simulated variable. The use of the measurement equation then allows us to obtain a simulated path for the control variables :

$$\widetilde{C}_t = \Pi\, \widetilde{S}_{B,t}$$

It is then possible, using the definition of the log–linearization, to get the "original" series :

$$\widehat{x}_t = \log\left(\frac{x_t}{x^{\star}}\right) \Longrightarrow x_t \simeq x^{\star}(1 + \widehat{x}_t)$$

Then the series are logged and filtered with the Hodrick–Prescott filter. This filter eliminates the trend component and the long run movements of the series. For a series X, the filter

[6] Just recall that for a predetermined variable $E_t \widehat{S}_{t+1} = \widehat{S}_{t+1}$

determines the trend component x^T that minimizes the sum of the gap between the trend and the logged series, subject to the constraint that the second order differences in the trend component (namely the acceleration of the trend) are not too high :

$$\min_{\{x^T_\tau\}_{\tau=1}^{t}} \sum_{\tau=1}^{t} \left(x_\tau - x^T_\tau\right)^2$$

$$s.c. \quad \sum_{\tau=2}^{t-1} \left[\left(x^T_{\tau+1} - x^T_\tau\right) - \left(x^T_\tau - x^T_{\tau-1}\right)\right]^2 \leq \mu$$

μ allows to control for the trend component properties generated by the filter. The lower μ is, the smoother the fluctuations in the trend component are. Thus, $\mu = 0$ leads to a deterministic trend, corresponding to the ordinary least square estimator of the trend. Whereas, when $\mu \to \infty$, the trend component corresponds to the input series. The previous program is solved considering the lagrangean form :

$$\min_{\{x^T_\tau\}_{\tau=1}^{t}} \sum_{\tau=1} \left(x_\tau - x^T_\tau\right)^2 + \lambda \sum_{\tau=2}^{t-1} \left[\left(x^T_{\tau+1} - x^T_\tau\right) - \left(x^T_\tau - x^T_{\tau-1}\right)\right]^2$$

where λ is the lagrangean multiplier associated to the constraint[7]. The cyclical component associated to the logged series x_t is then given by :

$$\hat{x}_t = x_t - x^T_t$$

Then, we compute a set of statistics for the so generated series. This operation is done a high number of times, we then obtain a distribution on each moment, for which we can compute the average and the standard deviation. This average is then compared to its empirical counterpart.

[7] Usually $\lambda = 1600$. See Hodrick and Prescott [1980] or King and Rebelo [1993] for further details.

3 An application to the growth model :

The matrices associated to the system (17)–(18) are given by :

Measurement equation :

$$M_{cc} = \begin{pmatrix} -\sigma & 0 & 0 \\ 0 & 1 & 0 \\ -s_c & 1 & -s_i \end{pmatrix} \quad M_{cs} = \begin{pmatrix} 0 & 0 & 1 \\ \alpha & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

State equation :

$$M_{ss0} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{\alpha\beta}{\gamma s_k} & 0 & 1 \end{pmatrix} \quad M_{ss1} = \begin{pmatrix} (\delta - 1) & 0 & 0 \\ 0 & -\rho & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$M_{sc0} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{\alpha\beta}{\gamma s_k} & 0 \end{pmatrix} \quad M_{sc1} = \begin{pmatrix} 0 & 0 & \delta \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$M_{se} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Solving the model we then obtain :

$$\begin{pmatrix} \hat{k}_{t+1} \\ \hat{a}_{t+1} \end{pmatrix} = M_{SS} \begin{pmatrix} \hat{k}_t \\ \hat{a}_t \end{pmatrix} + M_{SE}\,\varepsilon_{t+1}$$

with

$$M_{SS} = \begin{pmatrix} \pi_{kk} & \pi_{ka} \\ 0 & \rho \end{pmatrix} ; \; M_{SE} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

and :

$$\begin{pmatrix} \hat{c}_t \\ \hat{y}_t \\ \hat{i}_t \end{pmatrix} = \Pi \begin{pmatrix} \hat{k}_t \\ \hat{a}_t \end{pmatrix}$$

with :

$$\Pi = \begin{pmatrix} \pi_{ck} & \pi_{ca} \\ \pi_{yk} & \pi_{ya} \\ \pi_{ik} & \pi_{ia} \end{pmatrix}$$

As an illustrative example we set the following values for the model :

Table 1 : Calibration

β	σ	α	δ	ρ	σ_ε
0.99	2	0.4	0.025	0.95	0.01

Table 2 : Steady State

$y^\star$	$c^\star$	$i^\star$	$k^\star$
5.0640	3.6213	1.4427	57.7077

Then the previous system is given by :

$$\begin{pmatrix} k_{t+1} \\ a_{t+1} \end{pmatrix} = \begin{pmatrix} 0.9792 & 0.0665 \\ 0.0000 & 0.9500 \end{pmatrix} \begin{pmatrix} k_t \\ a_t \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \varepsilon_{t+1}$$

and :

$$\begin{pmatrix} c_t \\ y_t \\ i_t \end{pmatrix} = \begin{pmatrix} 0.4918 & 0.3381 \\ 0.4000 & 1.0000 \\ 0.1696 & 2.6613 \end{pmatrix} \begin{pmatrix} k_t \\ a_t \end{pmatrix}$$

4 Impulse response analysis :

5 Quantitative evaluation :

Table 2 : Quantitative Analysis

	U.S. Economy			Model		
	std(x)	std(x)/std(y)	Corr(x,y)	std(x)	std(x)/std(y)	Corr(x,y)
y	1.76	1.00	1.00	1.29	1.00	1.00
c	1.29	0.73	0.85	0.45	0.35	0.97
i	8.60	4.89	0.82	3.44	2.67	0.99

6 Matlab Code :

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Solving the Growth Model by Farmer's Method %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear; % Clear the memory
clc; % clear the text screen (clg for graphic screen)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Algorithm Parameters %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ncont=3; % # of controls
nback=1; % # of Backward endogenous state variables
nshoc=1; % # of exogenous shocks
ntotb=nback+nshoc; % Total # of Backward state variables
nforw=1; % # of Forward endogenous state variables
nstat=ntotb+nforw; % Total # of state variables
long=120; % Length of simulated series
slong=200; % Length of simulation
tronc=50; % Truncature of simulations
nsim=100; % # of simulations
nrep=200; % IRF horizon
Mcc=zeros(ncont,ncont); % Matrix Control-Control
Mcs=zeros(ncont,nstat); % Matrix Control-State
Mss0=zeros(nstat,nstat); % Matrix State-State
Mss1=zeros(nstat,nstat); % Matrix State-State lagged
Msc0=zeros(nstat,ncont); % Matrix State-Control
Msc1=zeros(nstat,ncont); % Matrix State-Controle lagged
Mse=zeros(nstat,nshoc); % Matrix State-Shocks
select=[1:3]; % Selection of variables of interest
indy=2; % Index of Y(t) in the selection
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Structural Parameters of the Economy %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

beta=0.99; % Discount factor
sigma=2; % Elasticity of utility
alpha=0.4; % Elasticity of output with regard to capital
delta=0.025; % Depreciation rate
rho=0.95; % Autocorrelation
stda=0.01; % Standard deviation of the shock
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Steady state %
```

```
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ksy=(alpha*beta)/(1-beta*(1-delta));
ysk=1/ksy;
isy=delta*ksy;
csy=1-isy;
yss=ksy^(alpha/(1-alpha));
kss=ksy*yss;
css=csy*yss;
iss=isy*yss;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Matrices Coe±cients %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%
% Consumption
%
cc=-sigma;
cl=1;
%
% Production function
%
yy=1;
yk=alpha;
ya=1;
%
% Investment
%
ic=-csy;
iy=1;
ii=-isy;
%
% Law of motion of capital
%
kk=1;
kkl=delta-1;
kil=delta;
%
% Euler equation
%
ll=-1;
lll=1;
lk=alpha*beta*ysk;
ly=lk;
%
% Technological shock
%
aa=1;
aal=-rho;
ae=1;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% De¯ning the matrices %
% %
% %
% Mcc X(t) = Mcs S(t) %
% Mss0 S(t+1) + Mss1 S(t) = Msc0 X(t+1) + Msc1 X(t) + e(t+1) %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%
% Mcc
%
Mcc(1,1)=cc;
Mcc(2,2)=yy;
Mcc(3,1)=ic;
Mcc(3,2)=iy;
Mcc(3,3)=ii;
%
% Mcs
%
Mcs(1,3)=cl;
Mcs(2,1)=yk;
Mcs(2,2)=ya;
%
% Mss0
%
Mss0(1,1)=kk;
Mss0(2,2)=aa;
Mss0(3,1)=lk;
Mss0(3,3)=ll;
%
% Mss1
%
Mss1(1,1)=kkl;
Mss1(2,2)=aal;
Mss1(3,3)=lll;
%
% Msc0
%
Msc0(3,2)=ly;
%
% Msc1
%
Msc1(1,3)=kil;
%
% Mse
%
Mse(2,1)=ae;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Solving the system (Farmer)%
% %
% -1 %
% X(t) = Mcc Mcs S(t) %
% %
% -1 -1 %
% S(t+1)=(Mss0-Msc0 Mcc Mcs)(Msc1 Mcc Mcs-Mss1)S(t) %
% -1 %
% +(Mss0-Msc0 Mcc Mcs)e(t+1) %
% %
% S(t+1) = W S(t) + R e(t+1) %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%
% First : eliminate the control in the state equation
%
M0=inv(Mss0-Msc0*inv(Mcc)*Mcs);
M1=(Mss1-Msc1*inv(Mcc)*Mcs);
W=-M0*M1;
%
% Computation of eigenvalues and eigenvectors
%
[P,MU] = eig(W); % P-> Eigenvectors, MU -> Eigenvalues
AMU=abs(MU); % Compute the modulus of eigenvalues
%
% Sorting the eigenvalues
%
[MU,k] = sort(diag(AMU)); % sort the vector diag(AMU) in MU, k is the indexes
P=P(:,k); % rearrange the P matrix according to the sorting
Q=inv(P); % Compute the inverse of P
%
% Just compute the last step of Farmer's method
%
M=[eye(ntotb);-inv(Q(ntotb+1:nstat,ntotb+1:nstat))*Q(ntotb+1:nstat,1:ntotb)];
MSS=W(1:ntotb,:)*M;
M2=M0*Mse;
MSE=M2(1:ntotb,1:nshoc);
PI=inv(Mcc)*Mcs*M;
%
% We are Done !!!!
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% So we now get : %
% %
% X(t)=Pi S(t) %
% %
% S(t+1)=MSS S(t) + MSE e(t+1) %
% %
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp('Policy Functions :')
disp('==================');
disp('');
disp('K(t+1) - A(t+1) as functions of (K(t),A(t)) :');
disp(MSS);
disp('');
disp('');
disp('C(t) - Y(t) - I(t) as functions of (K(t),A(t)) :');
disp(PI);
disp('');
disp('');
disp('Hit a Key');pause;clc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Impulse response functions to a technological shock %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

CA=[1]; % Size of the shock
REPSA=MSE*CA; % Instantaneous response
for i=2:nrep; %
 REPSA=[REPSA MSS*REPSA(:,i-1)]; % Just iterate on the state equation
end; %
REPXA=PI*REPSA; % IRF of control variables
%
% Display the graphs
%
clg;
subplot(221);plot(REPXA(1,:),'w');title('IRF(C,A)');xlabel('Quarters');
subplot(222);plot(REPXA(2,:),'w');title('IRF(Y,A)');xlabel('Quarters');
subplot(223);plot(REPXA(3,:),'w');title('IRF(I,A)');xlabel('Quarters');
subplot(224);plot(REPSA(1,:),'w');title('IRF(K,A)');xlabel('Quarters');
%print -dps growth.eps;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Simulation %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc;
HP=hpmat(long,1600); % Just obtain the HP ¯lter matrix
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %
% Loop of Simulation %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%
% The deterministic component
```

```
%
% Trick : kron performs the kronecker product of 2 matrices.
%
determ=kron(log([css;yss;iss]),ones(1,long));
%
% stochastic Simulation
%
SHP=[];SRHP=[];CHP=[];
for j=1:nsim;
 disp(sprintf('Simulation : %4.0f',j));
%
% Random generator
%
 CHOC=[stda]*randn(nshoc,slong);
%
% Initialize the states
%
 SSIM=MSE*CHOC(1,1);
%
% Iterate on the state equation to obtain series of the state variables
%
 for i=2:slong;
 SSIM=[SSIM MSS*SSIM(:,i-1)+MSE*CHOC(:,i)];
 end;
%
% Build selected controls
%
 XSIM=PI(select,:)*SSIM(:,tronc:tronc+long-1)+determ;
%
% Obtain their HP-¯ltered representation
%
 HPSIM=XSIM'-HPnXSIM';
%
% Centering (Actually HP-¯ltered variables are centered)
%
 HPSIM=HPSIM-kron(mean(HPSIM),ones(length(HPSIM),1));
%
% Store the moments in di®erent matrices
%
% For example : at iteration "i", SHP has i rows. then at iteration "i+1"
% the SHP equals the SHP matrix to whoich we had another
% line containing std(HPSIM).
%
 SHP=[SHP;std(HPSIM)];
 SRHP=[SRHP;std(HPSIM)/std(HPSIM(:,indy))];
 VCOV=HPSIM'*HPSIM;
 CHP=[CHP;VCOV(indy,:)./sqrt(diag(VCOV)'*VCOV(indy,indy))];
end;

clc;
disp('Quantitative Evaluation :');
disp('=========================');
disp('');
disp('');
```

```
disp(sprintf('%5.0f Simulations',nsim));
disp('');
disp('Standard deviations : C - Y - I');
disp('');
disp([mean(SHP);std(SHP)]);
disp('');
disp('Relative Standard deviations : C - Y - I');
disp('');
disp([mean(SRHP);std(SRHP)]);
disp('');
disp('Correlation with Y : C - Y - I');
disp('');
disp([mean(CHP);std(CHP)]);
```