Econ. 511b Spring 1997 C. Sims

Stochastic Decentralization Exercise

Consider an economy in which there are identical individuals, each of whom maximizes

$$E\left[\sum_{t=0}^{\infty} \beta^t \frac{C_t^{1-\gamma}}{1-\gamma}\right] \tag{1}$$

subject to

$$C_t + Q_t\left(S_t - S_{t-1}\right) = Y_t + \pi_t + \delta_t S_{t-1} \ . \tag{2}$$

The individual chooses C and S, with the usual requirement that choices at t depend only on information available at t, and takes Q, Y, δ and π as exogenously given processes. Y is an endowment process, π is the process of payments of profits by firms to owners, δ is the process of dividends per share on the traded security, and Q is the price of the security.

Firms, of which there are just as many as there are individuals, maximize a function of profits:

$$E\left[\sum_{t=0}^{\infty} \varphi\left(\pi(t)\right)\beta^t\right] \tag{3}$$

subject to

$$\pi(t) = A_t K_{t-1}^{\alpha} + (1-\mu)K_{t-1} - K_t + Q_t\left(S_t - S_{t-1}\right) - \delta_t S_{t-1} \ . \tag{4}$$

The firm chooses π, K, and S, taking the processes for A, Q, and δ as exogenously given. About the function φ we know that $\varphi' > 0, \varphi'' < 0$. The nature of the security is defined by specifying $\delta_t = A_t$, all t. That we use the same symbol S for firm liabilities in the form of the security and for individual assets in the form of the security is a shortcut to imposing market clearing (since we have the same numbers of firms and individuals). The same is true for our use of the same symbol for π in both problems. Assuming $\alpha = .35$, $\beta = .96$, $\gamma = 3$, $\mu = .1$ and $\bar{A} = EA_t = 1$, find the steady state of this system and linearize the model around the steady state. Assuming A is i.i.d., find a stochastic difference equation that characterizes the model solution by relating the variables in the model to their lagged values and to the current shock A. Derive the social resource constraint from the firm and consumer budget constraints, and set up the planner's problem of maximizing (1) subject to the social resource constraint. In the planner's problem, no price variables or securities holdings should appear. Find the FOC's of the planner's problem, linearize them about steady state, and solve the resulting linearized model. Compare the solution to that for the decentralized economy. For a given path of A, are the time paths of C and K in the two linearized solutions the same? [Additional hard question, optional: Would the two time paths be the same in the original nonlinear model?]

Below you will find a transcript of a Matlab session that sets up this model for solution using `gensys`. The session shows techniques for setting up g0 and g1 parametrically, so that you can change parameter values and resolve if necessary. It also shows some errors and backtracking to correct them, so you can see how that is done. Note that in some cases long lines have “wrapped” below, without the “…” at the end of the line that is necessary to make this legal in Matlab input.

```
» beta=.95;delta=.07;alpha=.3;K=((1/beta-1+delta)/alpha)^(1/(alpha-1))
K =
    3.5894
» C=K^alpha+1-delta*K
C =
    2.2160
» Q=beta/(1-beta)
Q =
   19.0000
» pi=K^alpha-delta*K
pi =
    1.2160
» dpi=1/pi
dpi =
    0.8224
» ddpi=-dpi*dpi
ddpi =
   -0.6763
» g0=[1 0 0 1 -K^alpha -1;
      ]
g0 =
    1.0000         0         0    1.0000   -1.4673   -1.0000
» gamma=2;
» g0=[0 1 0 0 1 -K^alpha -1; 0 -gamma*C^(-1+gamma) beta*C^(-gamma) 0 0 beta*C^-gamma 0;
      ddpi 0 0 0 0 beta*alpha*K^(alpha-1) 0; ddpi*Q 0 beta*dpi 0 0 beta*dpi 0;
      1 0 0 -Q 1 -K^alpha 0]
g0 =
         0    1.0000         0         0    1.0000   -1.4673   -1.0000
         0   -4.4320    0.1935         0         0    0.1935         0
   -0.6763         0         0         0         0    0.1165         0
  -12.8495         0    0.7812         0         0    0.7812         0
    1.0000         0         0  -19.0000    1.0000   -1.4673         0
» psi=g0(:,6:7)
psi =
   -1.4673   -1.0000
    0.1935         0
    0.1165         0
```

```
    0.7812         0
   -1.4673         0
» g0=g0(:,1:5)
g0 =
         0    1.0000         0         0    1.0000
         0   -4.4320    0.1935         0         0
   -0.6763         0         0         0         0
  -12.8495         0    0.7812         0         0
    1.0000         0         0  -19.0000    1.0000
» g1=[0 0 0 0 1/beta;0 -gamma*Q*C^(-gamma-1) C^-gamma 0 0;ddpi 0 0 0 -beta*dpi*alpha*(alpha-1)*K^(alpha-2);

]
g1 =
         0         0         0         0    1.0526
         0   -3.4920    0.2036         0         0
   -0.6763         0         0         0    0.0187
» g1=[g1;ddpi 0 dpi 0 0;0 0 0 -Q-1 1/beta]
g1 =
         0         0         0         0    1.0526
         0   -3.4920    0.2036         0         0
   -0.6763         0         0         0    0.0187
   -0.6763         0    0.8224         0         0
         0         0         0  -20.0000    1.0526
» g0(2,2)=-gamma*C^(-1-gamma);
» g0
g0 =
         0    1.0000         0         0    1.0000
         0   -0.1838    0.1935         0         0
   -0.6763         0         0         0         0
  -12.8495         0    0.7812         0         0
    1.0000         0         0  -19.0000    1.0000
» pi=[0 0 0;eye(3);0 0 0]
pi =
     0     0     0
     1     0     0
     0     1     0
     0     0     1
     0     0     0
» const=zeros(5,1);
» help gensys
```

```
 function [G1,C,impact,fmat,fwt,ywt,gev,eu]=gensys(g0,g1,c,psi,pi,div)
 System given as
         g0*y(t)=g1*y(t-1)+c+psi*z(t)+pi*eta(t),
 with z an exogenous variable process and eta being endogenously determined
 one-step-ahead expectational errors.  Returned system is
        y(t)=G1*y(t-1)+C+impact*z(t)+ywt*inv(I-fmat*inv(L))*fwt*z(t+1) .
  If z(t) is i.i.d., the last term drops out.
  If div is omitted from argument list, a div>1 is calculated.
  eu(1)=1 for existence, eu(2)=1 for uniqueness.  eu(1)=-1 for
  existence only with not-s.c. z; eu=[-2,-2] for coincident zeros.
  By Christopher A. Sims
  Corrected 10/28/96 by CAS

» [G1,SC,impact,fmat,fwt,ywt,gev,eu]=gensys(g0,g1,const,psi,pi)
G1 =
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
SC =
     0
     0
     0
     0
     0
impact =
     0     0
     0     0
     0     0
     0     0
     0     0
fmat =
  Columns 1 through 4
   0.0512- 0.0000i   0.0220+ 0.0090i   0.8674+ 0.0009i  -0.0365- 0.0000i
   0.0000+ 0.0000i   0.9763- 0.1527i -14.8008+ 0.0155i   0.0404- 0.0000i
   0.0000+ 0.0000i   0.0000+ 0.0000i   0.9763+ 0.1527i  -0.0001- 0.0004i
   0.0000+ 0.0000i   0.0000- 0.0000i   0.0000+ 0.0000i   0.9500
   0.0000+ 0.0000i   0.0000+ 0.0000i   0.0000- 0.0000i   0.0000
  Column 5
  -0.9478- 0.0008i
  -0.0483+ 0.0004i
  -0.0382+ 0.0005i
```

```
    0.0000
    0.9500
fwt =
  -1.0190- 0.0009i  -0.6901- 0.0006i
   0.7166+ 0.0015i  -0.0623+ 0.0003i
  -0.2108- 0.0074i  -0.0270+ 0.0006i
  -0.0343            0.0234
   0.9965            0.6518
ywt =
  Columns 1 through 4
   0.0011- 0.0000i   0.0017- 0.0103i  -0.9999+ 0.0001i   0.0028
   0.6866- 0.0006i  -0.0180- 0.0000i   0.0008- 0.0002i   0.0261
   0.0188- 0.0000i  -0.9989- 0.0010i  -0.0017- 0.0103i  -0.0049
  -0.0381+ 0.0000i  -0.0056+ 0.0000i   0.0027- 0.0001i   0.9993
  -0.7258+ 0.0006i  -0.0426- 0.0001i  -0.0009- 0.0004i  -0.0279
  Column 5
        0
   0.7263
  -0.0424
   0.0000
   0.6861
gev =
  -0.1287+ 0.0001i  -2.5129+ 0.0020i
  -0.8057+ 0.1323i  -0.8262+ 0.0063i
  -0.6585- 0.1087i  -0.6754- 0.0057i
  18.9507           19.9481
  -1.3886           -1.4617
eu =
     0
     1
```

Note: This means that something's wrong. We have non-existence.

```
» gev(:,1).\gev(:,2)
ans =
  19.5314+ 0.0000i
   0.9998+ 0.1564i
   0.9998- 0.1564i
   1.0526
   1.0526
» abs(ans)
ans =
   19.5314
    1.0119
    1.0119
    1.0526
```

```
    1.0526
```

So all roots exceed one, which can't be right. Repair incorrect g0(2,2) to fix this.

```
» g0(1:2,:)=[0 1 0 0 1 -K^alpha -1; 0 -gamma*Q*C^(-1-gamma) beta*C^(-gamma) 0 0 beta*C^(-gamma) 0;
]
??? In an assignment  A(matrix,:) = B, the number of columns in A and B
must be the same.


» g0(1:2,:)=[0 1 0 0 1 ; 0 -gamma*Q*C^(-1-gamma) beta*C^(-gamma) 0 0 ];
» g0
g0 =
         0    1.0000         0         0    1.0000
         0   -3.4920    0.1935         0         0
   -0.6763         0         0         0         0
  -12.8495         0    0.7812         0         0
    1.0000         0         0  -19.0000    1.0000
» [G1,SC,impact,fmat,fwt,ywt,gev,eu]=gensys(g0,g1,const,psi,pi)
G1 =
    0.0000    0.0000    0.0000   -5.4023    0.3577
    0.0000    0.0000    0.0000   12.3900   -0.5326
    0.0000    0.0000    0.0000  223.6456  -10.5636
    0.0000    0.0000    0.0000    0.1162    0.0469
    0.0000    0.0000    0.0000  -12.3900    1.5852
SC =
     0
     0
     0
     0
     0
impact =
   -0.4986   -0.0697
    0.7424   -0.1135
   14.7246   -1.1469
   -0.0653   -0.0503
   -2.2096   -0.8865
fmat =
    0.7265   -0.0018    0.0716
    0.0000    0.9500    0.0000
    0.0000    0.0000    0.9500
fwt =
    0.0538   -0.0702
   -0.0700    0.0001
    0.0179    0.0498
ywt =
  1.0e+002 *
```

```
   0.0072- 0.0000i   0.0515+ 0.0000i   0.0005- 0.0000i
   0.0066- 0.0000i  -0.1178+ 0.0000i  -0.0086- 0.0000i
   0.1183- 0.0000i  -2.1266+ 0.0000i   0.0173- 0.0000i
   0.0000- 0.0000i   0.0089+ 0.0000i   0.0005- 0.0000i
  -0.0066- 0.0000i   0.1178+ 0.0000i   0.0086- 0.0000i
gev =
  -0.9528- 0.1245i  -0.8358+ 0.0872i
   1.1973- 0.1565i   1.0503+ 0.1095i
   0.4860            0.6690
  18.9629           19.9609
  -3.2780           -3.4505
eu =
     1
     1
» gev(:,1).\gev(:,2)
ans =
   0.8507- 0.2027i
   0.8507+ 0.2027i
   1.3764
   1.0526
   1.0526
» abs(ans)
ans =
    0.8745
    0.8745
    1.3764
    1.0526
    1.0526
»
```