

```

1  var x xp b k kp q mu phi CC YY;
2
3  varexo ed;
4
5  parameters alpha m Kb beta betap a c z Rs qs bs ks kps xps phis mus qs;
6
7  load param_km;
8  set_param_value('alpha',alpha);
9  set_param_value('m',m);
10 set_param_value('Kb',Kb);
11 set_param_value('beta',beta);
12 set_param_value('betap',betap);
13 set_param_value('a',a);
14 set_param_value('c',c);
15 set_param_value('z',z);
16 set_param_value('Rs',Rs);
17 set_param_value('qs',qs);
18 set_param_value('bs',bs);
19 set_param_value('ks',ks);
20 set_param_value('kps',kps);
21 set_param_value('xps',xps);
22 set_param_value('phis',phis);
23 set_param_value('mus',mus);
24 set_param_value('qs',qs);
25
26 model;
27
28 % (1) Euler equation bonds farmer
29 1 + phi = (beta*(1+phi(+1)) + mu)/betap;
30
31 % (2) Euler equation capital farmer
32 q*(1+phi) + beta*c*phi(+1) = beta*(1+phi(+1))*((1+ed(+1))*(a + c) + q(+1)) + mu*q(+1);
33
34 % (3) Budget constraint
35 q*(k - k(-1)) + b(-1)/betap + x = (1+ed)*(a+c)*k(-1) + b;
36
37 % (4) Borrowing constraint
38 b = betap*q(+1)*k;
39
40 % (5) Euler equation gatherer
41 q = betap*((1+ed(+1))*alpha*(z + kp)^(alpha-1) + q(+1));
42
43 % (6) Resource constraint
44 x + m*xp = (1+ed)*(a+c)*k(-1) + m*(1+ed)*(z + kp(-1))^(alpha);
45
46 % (7) Capital market-clearing
47 k + m*kp = Kb;
48
49 % (8) non-tradeable constraint
50 x = c*k(-1);
51
52 % (9) Aggregate consumption
53 CC = x + m*xp;
54
55 % (10) Aggregate output
56 YY = CC;
57
58 end;
59
60 initval;
61 x = c*ks;
62 k = ks;
63 kp = kps;
64 q = qs;
65 b = bs;
66 mu = mus;
67 phi = phis;
68 xp = xps;
69 end;

```

```
70
71 steady;
72
73 shocks;
74 var ed = 0.0011^2;
75 end;
76
77 stoch_simul(order=1, irf=12, nograph, ar=0);
78
```