# Find and Describe Balanced Growth Path

play_with_bgp.m

*by* Jaromir Benes

May 27, 2015

Summary

The SPBC.model is a BGP model: It does not have a stationary long run. Instead, it has two unit roots, introduced through the productivity process, and the general nominal price level. To deal with BGP models, there is absolutely no need to stationarise them. They can be worked with directly in their non-stationary forms.

## Contents

1 Clear Workspace . . . . . 2
2 Load Solved Model Object . . . . . 2
3 Compute Two Different Points on BGP . . . . . 2
4 Solve Model Around Different Points . . . . . 4
5 Help on IRIS Functions Used in This File . . . . . 6

# 1 Clear Workspace

Clear workspace, close all graphics figures, clear command window, and check the IRIS version.

```
clear;
close all;
clc;
irisrequired 20140315;
```

# 2 Load Solved Model Object

Load the solved model object built in `read_model`. Run `read_model` at least once before running this m-file.

```
load read_model.mat m;
```

# 3 Compute Two Different Points on BGP

Compute two different points on the BGP corresponding to two different levels of productivity, `A`. The resulting steady-state levels of other varibles are always in constant proportion to the level of `A` (here, they simply double). The steady-state growth rates remain, obviously, unchanged. Whenever some variables are fixed in `sstate`, the steady state must be solved for non-recursively, i.e. with `'blocks=' false` 1; this is the default setting, and can be therefore omitted.

```
oo = {'tolX=',1e-16,'tolFun=',1e-16};

m1 = m;
m1.A = 2;
m1 = sstate(m1, ...
    'growth=',true,'blocks=',false,'fixLevel=','A','display=','final', ...
    'optimSet=',oo); 1
chksstate(m1);

m2 = m;
m2.A = 4;
m2 = sstate(m2, ...
    'growth=',true,'blocks=',false,'fixLevel=','A','display=','final', ...
    'optimSet=',oo); 1
chksstate(m2);

```

```
54 disp('Productivity level and gross rate of growth')
55 m1.A
56 m2.A
57
58 disp('Output level and gross rate of growth')
59 m1.Y
60 m2.Y
61
62 disp('Real wage level')
63 real(m1.W) / real(m1.P) %#ok<NOPTS>
64 real(m2.W) / real(m2.P) %#ok<NOPTS>
```

```
Local minimum possible.

lsqnonlin stopped because the relative size of the current step is less than
the selected value of the step size tolerance.




Local minimum found.

Optimization completed because the size of the gradient is less than
the selected value of the function tolerance.




Local minimum possible.

lsqnonlin stopped because the relative size of the current step is less than
the selected value of the step size tolerance.




Local minimum found.

Optimization completed because the size of the gradient is less than
the selected value of the function tolerance.



Productivity level and gross rate of growth
ans =
```

```
   2.0000 + 1.0074i
ans =
   4.0000 + 1.0074i
Output level and gross rate of growth
ans =
   3.1039 + 1.0074i
ans =
   6.2078 + 1.0074i
Real wage level
ans =
    3.4627
ans =
    6.9254
```

## 4 Solve Model Around Different Points

It does not matter which point on the BGP is used to solve the model. They give the same solution. Illustrate this fact here by comparing the covariance matrices of the model variables, and a shock simulation.

```
m1 = solve(m1);
m2 = solve(m2);

C1 = acf(m1);
C2 = acf(m2);

index = isfinite(C1);
maxabs(C1(index),C2(index))

d1 = zerodb(m1,1:20);
d1.Er(1) = 0.01;
s1 = simulate(m1,d1,1:20,'deviation',true);
s1 = dbextend(d1,s1);

d2 = zerodb(m2,1:20);
d2.Er(1) = 0.01;
s2 = simulate(m2,d2,1:20,'deviation',true);
s2 = dbextend(d2,s2);

[s1.Y,s2.Y,s1.Y-s2.Y] %#ok<NOPTS>
maxabs(s1,s2)
```

```
ans =
```

```
   9.1660e-13
ans =
        tseries object: 24-by-3
        -3:        1           1           0
        -2:        1           1           0
        -1:        1           1           0
         0:        1           1           0
         1:  0.99514     0.99514           0
         2:  0.99254     0.99254  1.1102e-16
         3:  0.99153     0.99153  1.1102e-16
         4:  0.99158     0.99158  1.1102e-16
         5:  0.99231     0.99231  1.1102e-16
         6:  0.99342     0.99342  1.1102e-16
         7:  0.99468     0.99468  1.1102e-16
         8:  0.99596     0.99596  1.1102e-16
         9:  0.99713     0.99713  2.2204e-16
        10:  0.99816     0.99816  3.3307e-16
        11:  0.99901     0.99901  3.3307e-16
        12:  0.99967     0.99967  2.2204e-16
        13:   1.0002      1.0002  2.2204e-16
        14:   1.0005      1.0005  4.4409e-16
        15:   1.0007      1.0007  4.4409e-16
        16:   1.0008      1.0008  4.4409e-16
        17:   1.0008      1.0008  4.4409e-16
        18:   1.0008      1.0008  4.4409e-16
        19:   1.0007      1.0007  4.4409e-16
        20:   1.0006      1.0006  4.4409e-16
    'Output'    'Output'    ''
        user data: empty
        export files: [0]
ans =
      Short: 1.5765e-14
       Infl: 5.6621e-15
     Growth: 1.8610e-14
       Wage: 3.3085e-14
          Y: 4.4409e-16
          N: 2.2204e-16
          W: 5.5511e-16
          Q: 5.5511e-16
          H: 6.6613e-16
          A: 1.1102e-16
          P: 2.2204e-16
          R: 2.2204e-16
         Pk: 9.9920e-16
         Rk: 8.8818e-16
     Lambda: 4.4409e-16
```

```
        dP: 1.1102e-16
       d4P: 2.2204e-16
        dW: 1.1102e-16
       RMC: 6.6613e-16
        Mp: 0
        Mw: 0
        Ey: 0
        Ep: 0
        Ea: 0
        Er: 0
        Ew: 0
     alpha: 0
      beta: 0
     gamma: 0
     delta: 0
         k: 0
        pi: 0
       eta: 0
       psi: 0
       chi: 0
       xiw: 0
       xip: 0
      rhoa: 0
      rhor: 0
    kappap: 0
    kappan: 0
    Short_: 0
     Infl_: 0
   Growth_: 0
     Wage_: 0
    ttrend: 0
```

## 5 Help on IRIS Functions Used in This File

Use either `help` to display help in the command window, or `idoc` to display help in an HTML browser window.

```
help model/sstate
help model/solve
help model/acf
help model/subsasgn
help model/subsref
help model/zerodb
```

```
help model/simulate
```