Table of Contents

Credit Modol with Firm Dynamics

```
clear all;
close all;
% Second version

Parallel=0 % 2 for GPU, 1 for parallel CPU, 0 for single CPU.

rng('default') % For reproducibility
tic;
```

```
Parallel =

     0
```

Toolkit options

```
tauchenoptions.parallel=Parallel;

mcmomentsoptions.T=10^4;
mcmomentsoptions.Tolerance=10^(-9);
mcmomentsoptions.parallel=tauchenoptions.parallel;

vfoptions.parallel=Parallel;
```

```
simoptions.burnin=10^4;
simoptions.simperiods=10^5; % if iterate=0 then simperiod=10^6
simoptions.iterate=1;
simoptions.parallel=Parallel;

simoptions.maxit=10^4;

heteroagentoptions.verbose=1;
```

Parameters Calibration

```
% Preferences
Params.beta=0.9;% Discount rate

% Firm-level technology
Params.alpha=0.3;  % Capital share
Params.gamma=0.5; % alpha + gamma must be ~= 1
Params.delta=0.05; % Depreciation rate of physical capital
Params.cf=0.5; % Fixed cost of production

% Entry and Exit
Params.ce=0.5; % Fixed cost of entry
Params.lambda=0.1; % Probability of firm exit
% lambda is the average observed exit percentage between 2007--2017
% (https://sidra.ibge.gov.br/Tabela/2718#resultado)
Params.oneminuslambda=1-Params.lambda; % Probability of survival

% Distortions
Params.taurate=0; % This is the rate for the tax.
Params.subsidyrate=0; % This is the rate for the subsidy.
Params.gcost=0.01; % capital adjustment cost parameter

% Initial guesses
Params.p=0.446; % output price
Params.Ne=0.281; % total mass of new entrants

% Declare discount factors
DiscountFactorParamNames={'beta','oneminuslambda'};
% Declare percentage of entrants
EntryExitParamNames.MassOfNewAgents={'Ne'};
% Exogenous survival probability
EntryExitParamNames.CondlProbOfSurvival={'oneminuslambda'};
```

Steady-state interest rate

```
Params.i=1/Params.beta-1; % gross capital return
Params.r=Params.i+Params.delta; % net capital return
```

Exogenous state variables

```
n_s= 20; % firm-specific Productivity level
n_psi = 5; % credit tax
```

```
% Exogenous AR(1) process on (log) productivity
% logz=a+rho*log(z)+epsilon, epsilon~N(0,sigma_epsilon^2)
Params.rho=0.93;
Params.sigma_logz=sqrt(0.53);
Params.sigma_epsilon=sqrt((1-Params.rho)*((Params.sigma_logz)^2));
Params.a=0.098;

tauchenoptions.parallel=Parallel;
Params.q=2; % Hopenhayn & Rogerson (1993) do not report (based on
 Table 4 is seems something around q=4 is used, otherwise don't get
 values of z anywhere near as high as 27.3. (HR1993 have typo and call
 the column 'log(s)' when it should be 's')
[s_grid,
 pi_s]=TauchenMethod(Params.a,Params.sigma_epsilon^2,Params.rho,n_s,Params.q,tauch
 transmatrix]=TauchenMethod_Param(mew,sigmasq,rho,znum,q,Parallel,Verbose),
 transmatix is (z,zprime)
s_grid=exp(s_grid);

% Tax credit
psi_grid = linspace(-1,1,n_psi)';

% Transition matrix
% Note: considering that productivity and taxes are independent
n_z=[n_s,length(psi_grid)];
z_grid=[s_grid; psi_grid];


% transition matrix for the exogenous z and psi variables
pi_z=kron( pi_s,eye(prod(n_psi)))';

% Check transition matrix
for ii = 1: length(pi_z)
A = round(sum(pi_z(:,ii)),5);
if A == 1
else
   error('transition matrix sum is not one')
end
end
pi_z=pi_z';
```

Endogenous state variables

```
% grid for capital
n_a=40;

% steady-state capital without distotions
%%%%% The grid is the same as the Aiygari example
k_ss = ((Params.i+Params.delta/Params.alpha)^(1/Params.alpha-1));
nk1 = floor(n_a/3); nk2=floor(n_a/3); nk3=n_a-nk1-nk2;
a_grid = sort([linspace(0,k_ss,nk1),linspace(k_ss
+0.0001,3*k_ss,nk2),...
```

```
        linspace(3*k_ss+0.0001,15*k_ss,nk3)])';
```

Decision varibles

```
%There is no d variable

d_grid=[];
n_d=0;
```

Check endogenous, exogenous and decision variables

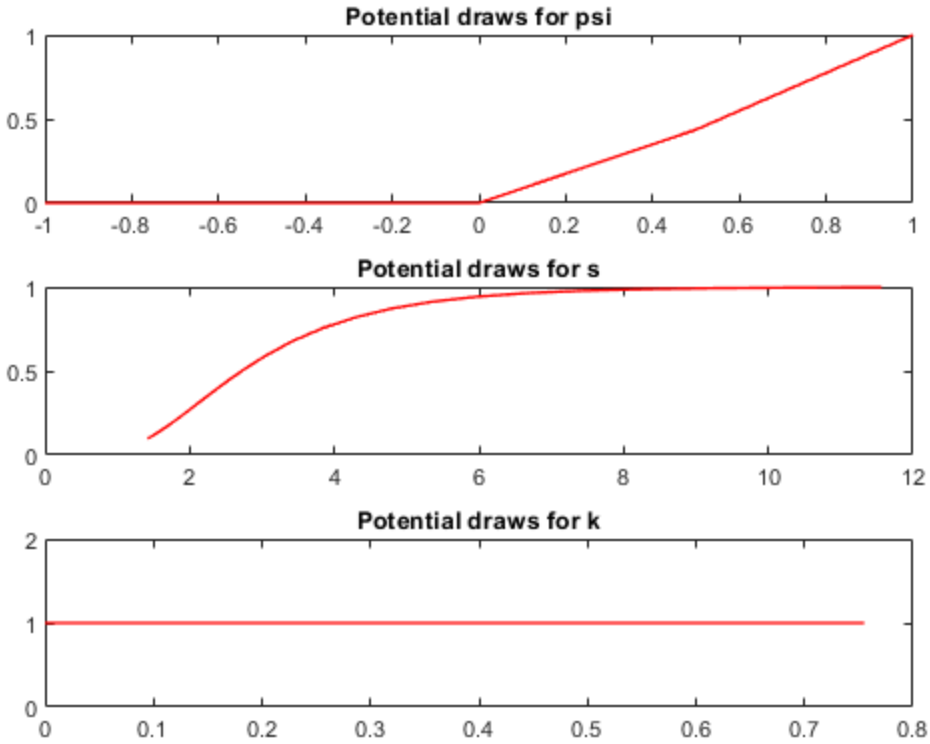
```
disp('sizes')
disp('vector(s) of endogenous state variables')
disp(n_a)
disp('vector(s) of exogenous state variable')
disp(n_z)
disp('vector(s) of decision variabes')
disp(n_d)
```

```
sizes
vector(s) of endogenous state variables
    40

vector(s) of exogenous state variable
    20     5

vector(s) of decision variabes
     0
```

Potential New Entrants Distribution over the states (s, psi, k)

```
% productivity (exogenous state)
cumsum_pistar_s = logncdf(s_grid,1,0.5)';
pistar_s=(cumsum_pistar_s-[0,cumsum_pistar_s(1:end-1)]);

% credit tax (exogenous state)
cumsum_pistar_psi = betacdf(psi_grid,.5,.4)';
pistar_psi =(cumsum_pistar_psi-[0,cumsum_pistar_psi(1:end-1)]);

% capital (endogenous state)
pistar_k = zeros(1,n_a);
pistar_k(1,1) = 1;
cumsum_pistar_k = cumsum(pistar_k);

if (abs(1-sum(pistar_psi)) || abs(sum(pistar_psi)-1)||
abs(sum(pistar_k)-1) > 1e-7)
    error('Draws are NOT a PMD.')
```

```
end


figure(1)
set(groot,'DefaultAxesColorOrder',[0 0 0],...

  'DefaultAxesLineStyleOrder','-|-|--|:','DefaultLineLineWidth',1);
subplot(3,1,1);
plot(psi_grid,cumsum_pistar_psi,'r')
title('Potential draws for psi')
subplot(3,1,2);
plot(s_grid,cumsum_pistar_s,'r')
title('Potential draws for s')
subplot(3,1,3);
plot(a_grid,cumsum_pistar_k,'r')
title('Potential draws for k')
```

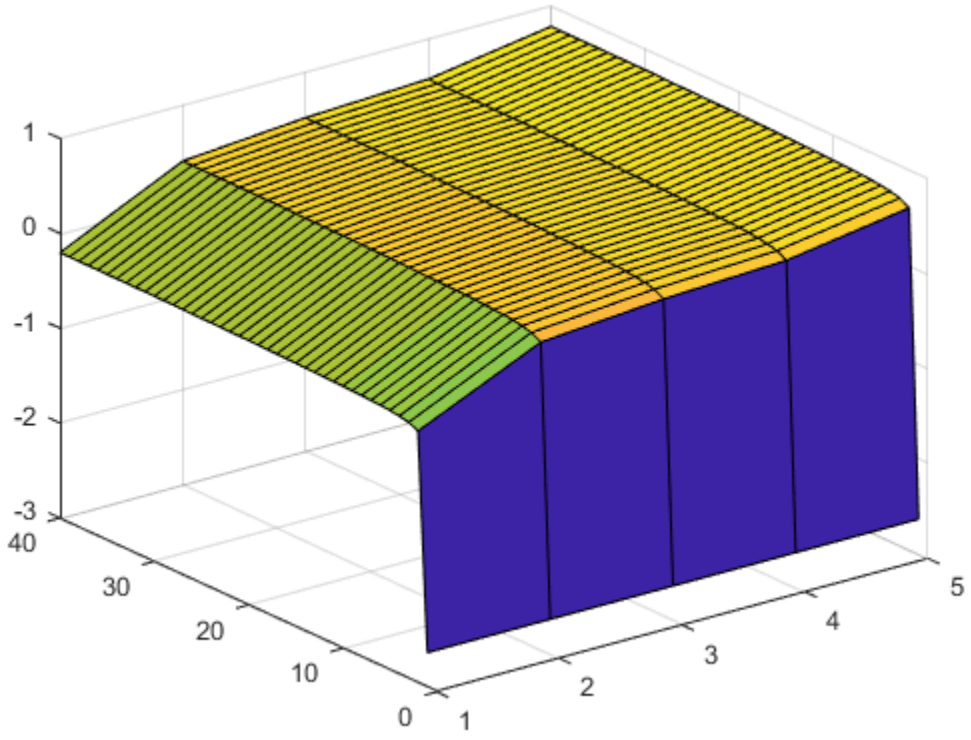


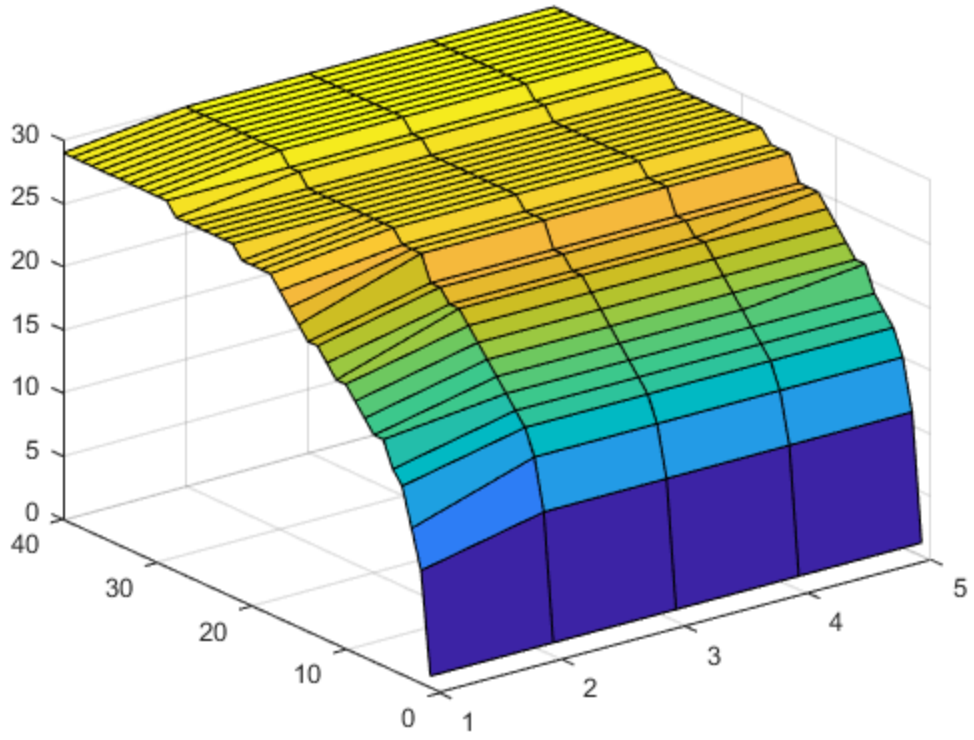
Return Function

```
ReturnFn=@(aprime_val, a_val,s_val, tau_val, p,r,
 alpha,gamma,taurate,subsidyrate, cf, gcost)...
RR2008p_ReturnFn(aprime_val, a_val,s_val, tau_val, p,r,
 alpha,gamma,taurate,subsidyrate, cf, gcost);

ReturnFnParamNames={ 'p','r', 'alpha','gamma','taurate','subsidyrate', 'cf', 'gcos
```

CHECK (to be erase)

```
if vfoptions.parallel==2
    V0=zeros([n_a,n_z,'gpuArray']);
else
    V0=zeros([n_a,n_z]);
end
[V,Policy]=ValueFnIter_Case1(V0, n_d,n_a,n_z,d_grid,...
    a_grid,z_grid, pi_z, ReturnFn, Params,
 DiscountFactorParamNames,...
    ReturnFnParamNames, vfoptions);

figure;
surf(squeeze(V(:,1,:)))

figure;
surf(squeeze(Policy(1,:,1,:)))
```





Aspects of the Endogenous entry

Exit is exogenous with probability lambda

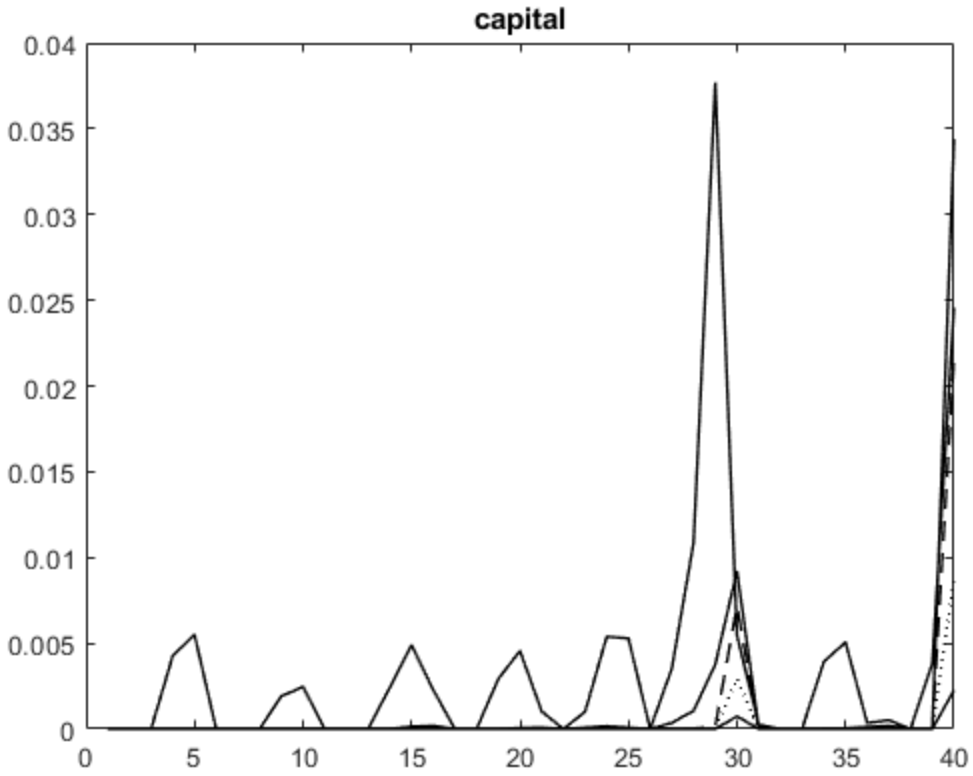
```
simoptions.agententryandexit=1;
simoptions.endogenousexit=0;

% Probability of being in the (s, psi) category
EntryExitParamNames.DistOfNewAgents={'upsilon'};

pistar_psi_s=pistar_s.*(pistar_psi)';
Params.upsilon=NaN(n_psi,n_s,n_a);
 for n=1:n_a
    Params.upsilon(:,:,n)=pistar_psi_s.*pistar_k(n);
 end

disp('upsilon size')
disp(size(Params.upsilon))

disp('sum of upsilon')
disp(sum(Params.upsilon(:)))
```

```
upsilon size
     5    20    40

sum of upsilon
```

```
    0.9981
```

CHECK (to be erased)

```
simoptions.parallel=Parallel

StationaryDist=StationaryDist_Case1(Policy,n_d,n_a,n_z,pi_z,...
    simoptions,Params,EntryExitParamNames);

figure;
surf(squeeze(StationaryDist.pdf(:,1,:)))


plot(squeeze(StationaryDist.pdf(:,1,:)))
title('capital')


simoptions =

  struct with fields:

              burnin: 10000
          simperiods: 100000
             iterate: 1
            parallel: 0
               maxit: 10000
    agententryandexit: 1
       endogenousexit: 0
```




```
%Use the toolkit to find the equilibrium price index
GEPriceParamNames={'p'}%, 'Ne'};

%FnsToEvaluateParamNames(1).Names={};
%FnsToEvaluate={};

heteroagentoptions.specialgeneqmcondn={0,'entry'};

FnsToEvaluateParamNames(1).Names={'alpha','gamma','r','p','taurate'};
FnsToEvaluateFn_nbar
 =@(aprime_val,a_val,z1_val,z2_val,mass,alpha,gamma,r,p,taurate)...
(((1-taurate*z2_val)*p*z1_val*gamma))^(1/(1-gamma))
 *aprime_val^(alpha/(1-gamma));
FnsToEvaluate={FnsToEvaluateFn_nbar};

AggVars=EvalFnOnAgentDist_AggVars_Case1(StationaryDist, Policy,...
    FnsToEvaluate, Params, FnsToEvaluateParamNames, n_d, n_a, n_z,...
    d_grid, a_grid, z_grid,
 simoptions.parallel,simoptions,EntryExitParamNames);

AggVars


GEPriceParamNames =

  1×1 cell array
```

```
    {'p'}


AggVars =

    2.0426


GEPriceParamNames={'p', 'Ne'};
GeneralEqmEqnParamNames(1).Names={};
GeneralEqmEqn_LabourMarket = @(AggVars,GEprices) 1-AggVars;

GeneralEqmEqnParamNames(2).Names={'beta','ce'};
GeneralEqmEqn_Entry = @(EValueFn,GEprices,beta,ce) beta*EValueFn-ce; %
 Free entry conditions (expected returns equal zero in eqm); note that
 the first 'General eqm price' is ce, the fixed-cost of entry.

GeneralEqmEqns={GeneralEqmEqn_LabourMarket,GeneralEqmEqn_Entry};
```

Find equilibrium prices

```
heteroagentoptions.verbose=1;
n_p=0;
% uncomment after erase the 'to be erase' chunks
% initial value function
%if vfoptions.parallel==2
%    V0=zeros([n_a,n_z,'gpuArray']);
%else
%    V0=zeros([n_a,n_z]);
%end

disp('Calculating price vector corresponding to the stationary eqm')
[p_eqm,p_eqm_index,GeneralEqmCondn]=HeteroAgentStationaryEqm_Case1(V0,...
    n_d, n_a, n_z, n_p, pi_z, d_grid, a_grid, z_grid, ReturnFn,...
    FnsToEvaluate, GeneralEqmEqns, Params,
 DiscountFactorParamNames,...
    ReturnFnParamNames, FnsToEvaluateParamNames,
 GeneralEqmEqnParamNames,...
    GEPriceParamNames,heteroagentoptions, simoptions, vfoptions,
 EntryExitParamNames);
```

```
Calculating price vector corresponding to the stationary eqm
Current Aggregates:

AggVars =

    2.0426

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4460
```

```
    0.2810


GeneralEqmConditionsVec =

   -1.0426   -0.4297

Current Aggregates:

AggVars =

    2.2539

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4683
    0.2810


GeneralEqmConditionsVec =

   -1.2539   -0.1370

Current Aggregates:

AggVars =

    2.1447

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4460
    0.2951


GeneralEqmConditionsVec =

   -1.1447   -0.4297

Current Aggregates:

AggVars =

    1.9273

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4237
```

```
    0.2951


GeneralEqmConditionsVec =

   -0.9273   -0.7075

Current Aggregates:

AggVars =

    1.8355

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4237
    0.2810


GeneralEqmConditionsVec =

   -0.8355   -0.7075

Current Aggregates:

AggVars =

    1.6945

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4125
    0.2740


GeneralEqmConditionsVec =

   -0.6945   -0.8403

Current Aggregates:

AggVars =

    1.7936

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4348
```

```
    0.2599


GeneralEqmConditionsVec =

   -0.7936   -0.5706

Current Aggregates:

AggVars =

    1.7176

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4404
    0.2424


GeneralEqmConditionsVec =

   -0.7176   -0.5006

Current Aggregates:

AggVars =

    1.4155

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4070
    0.2353


GeneralEqmConditionsVec =

   -0.4155   -0.9053

Current Aggregates:

AggVars =

    1.4058

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4348
```

```
    0.2037


GeneralEqmConditionsVec =

   -0.4058   -0.5706

Current Aggregates:

AggVars =

    1.2255

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4460
    0.1686


GeneralEqmConditionsVec =

   -0.2255   -0.4297

Current Aggregates:

AggVars =

    1.4784

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4794
    0.1756


GeneralEqmConditionsVec =

   -0.4784    0.0150

Current Aggregates:

AggVars =

    1.4227

Current GE prices and GeneralEqmConditionsVec:

p =

    0.5157
```

```
    0.1458


GeneralEqmConditionsVec =

   -0.4227    0.5349

Current Aggregates:

AggVars =

    0.8777

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4850
    0.1019


GeneralEqmConditionsVec =

    0.1223    0.0924

Current Aggregates:

AggVars =

    0.2985

Current GE prices and GeneralEqmConditionsVec:

p =

    0.5073
    0.0316


GeneralEqmConditionsVec =

    0.7015    0.4114

Current Aggregates:

AggVars =

    1.0752

Current GE prices and GeneralEqmConditionsVec:

p =

    0.5185
```

```
    0.1089


GeneralEqmConditionsVec =

   -0.0752    0.5765

Current Aggregates:

AggVars =

    1.2106

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4641
    0.1537


GeneralEqmConditionsVec =

   -0.2106   -0.1930

Current Aggregates:

AggVars =

    0.6448

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4697
    0.0799


GeneralEqmConditionsVec =

    0.3552   -0.1182

Current Aggregates:

AggVars =

    0.8471

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4721
```

```
    0.1038


GeneralEqmConditionsVec =

    0.1529   -0.0852

Current Aggregates:

AggVars =

    0.4636

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4930
    0.0520


GeneralEqmConditionsVec =

    0.5364    0.2053

Current Aggregates:

AggVars =

    1.0428

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4713
    0.1283


GeneralEqmConditionsVec =

   -0.0428   -0.0958

Current Aggregates:

AggVars =

    1.0846

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4842
```

```
    0.1263


GeneralEqmConditionsVec =

   -0.0846    0.0815

Current Aggregates:

AggVars =

    1.2367

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4706
    0.1527


GeneralEqmConditionsVec =

   -0.2367   -0.1064

Current Aggregates:

AggVars =

    0.9725

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4814
    0.1146


GeneralEqmConditionsVec =

    0.0275    0.0421

Current Aggregates:

AggVars =

    0.9357

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4685
```

```
    0.1165


GeneralEqmConditionsVec =

    0.0643   -0.1340

Current Aggregates:

AggVars =

    1.0464

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4803
    0.1238


GeneralEqmConditionsVec =

   -0.0464    0.0269

Current Aggregates:

AggVars =

    0.9708

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4904
    0.1102


GeneralEqmConditionsVec =

    0.0292    0.1675

Current Aggregates:

AggVars =

    1.0265

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4761
```

```
    0.1237


GeneralEqmConditionsVec =

   -0.0265   -0.0310

Current Aggregates:

AggVars =

    0.9539

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4772
    0.1145


GeneralEqmConditionsVec =

    0.0461   -0.0159

Current Aggregates:

AggVars =

    1.0075

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4719
    0.1236


GeneralEqmConditionsVec =

   -0.0075   -0.0883

Current Aggregates:

AggVars =

    0.9817

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4790
```

```
    0.1168


GeneralEqmConditionsVec =

    0.0183    0.0092

Current Aggregates:

AggVars =

    1.0548

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4779
    0.1261


GeneralEqmConditionsVec =

   -0.0548   -0.0059

Current Aggregates:

AggVars =

    0.9790

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4774
    0.1174


GeneralEqmConditionsVec =

    0.0210   -0.0134

Current Aggregates:

AggVars =

    0.9334

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4803
```

```
    0.1105


GeneralEqmConditionsVec =

    0.0666    0.0269

Current Aggregates:

AggVars =

    1.0034

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4772
    0.1204


GeneralEqmConditionsVec =

   -0.0034   -0.0166

Current Aggregates:

AggVars =

    1.0064

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4788
    0.1199


GeneralEqmConditionsVec =

   -0.0064    0.0061

Current Aggregates:

AggVars =

    1.0200

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4795
```

```
    0.1211


GeneralEqmConditionsVec =

   -0.0200    0.0158

Current Aggregates:

AggVars =

    1.0278

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4769
    0.1235


GeneralEqmConditionsVec =

   -0.0278   -0.0197

Current Aggregates:

AggVars =

    0.9935

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4785
    0.1185


GeneralEqmConditionsVec =

    0.0065    0.0020

Current Aggregates:

AggVars =

    0.9960

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4802
```

```
    0.1180


GeneralEqmConditionsVec =

    0.0040    0.0247

Current Aggregates:

AggVars =

    1.0019

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4779
    0.1198


GeneralEqmConditionsVec =

   -0.0019   -0.0063

Current Aggregates:

AggVars =

    0.9891

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4776
    0.1184


GeneralEqmConditionsVec =

    0.0109   -0.0104

Current Aggregates:

AggVars =

    1.0021

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4785
```

```
    0.1195


GeneralEqmConditionsVec =

   -0.0021    0.0019

Current Aggregates:

AggVars =

    1.0105

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4779
    0.1208


GeneralEqmConditionsVec =

   -0.0105   -0.0063

Current Aggregates:

AggVars =

    0.9977

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1191


GeneralEqmConditionsVec =

    0.0023   -0.0001

Current Aggregates:

AggVars =

    0.9978

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4790
```

```
    0.1188


GeneralEqmConditionsVec =

    0.0022    0.0081

Current Aggregates:

AggVars =

    1.0009

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4782
    0.1195


GeneralEqmConditionsVec =

   -0.0009   -0.0027

Current Aggregates:

AggVars =

    0.9989

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4787
    0.1190


GeneralEqmConditionsVec =

    0.0011    0.0045

Current Aggregates:

AggVars =

    1.0004

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4783
```

```
    0.1194


GeneralEqmConditionsVec =

   1.0e-03 *

   -0.4001   -0.8809

Current Aggregates:

AggVars =

    0.9961

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4781
    0.1190


GeneralEqmConditionsVec =

    0.0039   -0.0029

Current Aggregates:

AggVars =

    1.0006

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1194


GeneralEqmConditionsVec =

   1.0e-03 *

   -0.5738    0.7290

Current Aggregates:

AggVars =

    1.0032

Current GE prices and GeneralEqmConditionsVec:
```

```
p =

    0.4784
    0.1197


GeneralEqmConditionsVec =

   -0.0032   -0.0001

Current Aggregates:

AggVars =

    0.9991

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1192


GeneralEqmConditionsVec =

   1.0e-03 *

    0.8962   -0.0883

Current Aggregates:

AggVars =

    0.9993

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4785
    0.1192


GeneralEqmConditionsVec =

    0.0007    0.0015

Current Aggregates:

AggVars =

    1.0001
```

```
Current GE prices and GeneralEqmConditionsVec:

p =

    0.4783
    0.1194


GeneralEqmConditionsVec =

   1.0e-03 *

   -0.1138   -0.2803

Current Aggregates:

AggVars =

    0.9987

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4783
    0.1192


GeneralEqmConditionsVec =

    0.0013   -0.0011

Current Aggregates:

AggVars =

    1.0001

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

   -0.1025    0.2723

Current Aggregates:
```

```
AggVars =

    1.0011

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1195


GeneralEqmConditionsVec =

   -0.0011    0.0001

Current Aggregates:

AggVars =

    0.9996

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

    0.3996   -0.0462

Current Aggregates:

AggVars =

    1.0006

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1194


GeneralEqmConditionsVec =

   1.0e-03 *
```

```
   -0.5935    0.0381

Current Aggregates:

AggVars =

    0.9998

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

    0.1514   -0.0251

Current Aggregates:

AggVars =

    0.9998

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

    0.1627    0.5276

Current Aggregates:

AggVars =

    1.0000

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
```

```
    0.1193


GeneralEqmConditionsVec =

   1.0e-04 *

   -0.3908   -0.7837

Current Aggregates:

AggVars =

    0.9998

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4783
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

    0.1922   -0.3757

Current Aggregates:

AggVars =

    1.0000

Current GE prices and GeneralEqmConditionsVec:

p =

    0.4784
    0.1193


GeneralEqmConditionsVec =

   1.0e-03 *

   -0.0119    0.1103
```

Value Function, Policy and Firm Distribution in GE

```
disp('Calculating various equilibrium objects')
Params.p=p_eqm.p;
Params.Ne=p_eqm.Ne;
[V,Policy]=ValueFnIter_Case1(V0, n_d,n_a,n_z,[],a_grid,z_grid,
 pi_z,...
    ReturnFn, Params, DiscountFactorParamNames,
 ReturnFnParamNames,vfoptions);

StationaryDist=StationaryDist_Case1(Policy,n_d,n_a,n_z,pi_z,...
    simoptions, Params, EntryExitParamNames);
```

```
Calculating various equilibrium objects
```

Post GE values

```
FnsToEvaluateParamNames(1).Names={'alpha','gamma','r','p','taurate','subsidyrate'}
FnsToEvaluateFn_kbar =
 @(aprime_val,a_val,z1_val,z2_val,mass,alpha,gamma,r,p,...
    taurate,subsidyrate) aprime_val;
FnsToEvaluateParamNames(2).Names={'alpha','gamma','r','p','taurate','subsidyrate'}
FnsToEvaluateFn_output = @(aprime_val,a_val,z1_val,z2_val,mass,
 alpha,gamma,...
    r,p,taurate,subsidyrate)  p*((1-
taurate*z2_val)*z1_val)*(aprime_val^alpha)*...
    ((((((1-taurate*z2_val)*z1_val)*p*gamma))^(1/(1-gamma))
 *aprime_val^(alpha/(1-gamma)))^gamma);
FnsToEvaluateParamNames(3).Names={'alpha','gamma','r','p','taurate'};
FnsToEvaluateFn_nbar
 =@(aprime_val,a_val,z1_val,z2_val,mass,alpha,gamma,r,p,taurate)...
(((1-taurate*z2_val)*p*z1_val*gamma))^(1/(1-gamma))
 *aprime_val^(alpha/(1-gamma));
FnsToEvaluate={FnsToEvaluateFn_kbar,
 FnsToEvaluateFn_output,FnsToEvaluateFn_nbar};

%FnsToEvaluateParamNames(1).Names={'alpha','gamma','r','p','taurate','subsidyrate'

%FnsToEvaluateParamNames(1).Names={};
% Capital
%FnsToEvaluateFn_capital =
 @(aprime_val,a_val,z1_val,z2_val,mass,alpha,gamma,r,p,taurate,subsidyrate)
 aprime_val;
%FnsToEvaluate={FnsToEvaluateFn_capital};

AggVars=EvalFnOnAgentDist_AggVars_Case1(StationaryDist, Policy,...
    FnsToEvaluate, Params, FnsToEvaluateParamNames, n_d, n_a, n_z,...
    d_grid, a_grid, z_grid,
 simoptions.parallel,simoptions,EntryExitParamNames);
```

```
ValuesOnGrid=EvalFnOnAgentDist_ValuesOnGrid_Case1_Mass(StationaryDist.pdf,...
    StationaryDist.mass, Policy, FnsToEvaluate, Params,...
    FnsToEvaluateParamNames,EntryExitParamNames, n_d, n_a, n_z,...
    [], a_grid, z_grid, Parallel,simoptions);


ProbDensityFns=EvalFnOnAgentDist_pdf_Case1(StationaryDist, Policy,
 FnsToEvaluate,...
    Params, FnsToEvaluateParamNames, n_d, n_a, n_z, d_grid, a_grid,
 z_grid,...
    simoptions.parallel,simoptions,EntryExitParamNames);
```

Agggregate Values

```
Output.Y=AggVars(2);
Output.N=AggVars(3);
Output.K=AggVars(1);
Output.KdivY=Output.K/Output.Y;
```

Average values

```
Output.perY=AggVars(2)/StationaryDist.mass;
Output.perN=AggVars(3)/StationaryDist.mass;
Output.perK=AggVars(1)/StationaryDist.mass;

Output.TFP=(Output.Y/Output.N)./((Output.K/Output.N)^Params.alpha);


%%%%%%%%%%%%%%%
nbarValues=shiftdim(ValuesOnGrid(3,:,:,:),1);
normalize_employment=min(min(min(shiftdim(ValuesOnGrid(3,2:end,:,:),1)))); %
 Normalize so that smallest occouring value of nbar in the baseline is
 equal to 1.
nbarValues=nbarValues./normalize_employment;


Partion1Indicator=logical(nbarValues<5);
Partion2Indicator=logical((nbarValues>=5).*(nbarValues<50));
Partion3Indicator=logical(nbarValues>=50);

if ((sum(sum(sum(Partion1Indicator+Partion2Indicator
+Partion3Indicator)))) - prod(n_z)*(n_a) > 1e-3)
    error('error')
end

ShareOfEstablishments(1)=sum(sum(sum(StationaryDist.pdf(Partion1Indicator))));
ShareOfEstablishments(2)=sum(sum(sum(StationaryDist.pdf(Partion2Indicator))));
ShareOfEstablishments(3)=sum(sum(sum(StationaryDist.pdf(Partion3Indicator))));
ShareOfEstablishments(4)=sum(sum(sum(StationaryDist.pdf)));

Output_pdf=shiftdim(ProbDensityFns(2,:,:,:),1);
ShareOfOutput(1)=sum(sum(sum(Output_pdf(Partion1Indicator))));
ShareOfOutput(2)=sum(sum(sum(Output_pdf(Partion2Indicator))));
```

```
ShareOfOutput(3)=sum(sum(sum(Output_pdf(Partion3Indicator))));
ShareOfOutput(4)=sum(sum(sum(Output_pdf)));

Labour_pdf=shiftdim(ProbDensityFns(3,:,:,:),1);
ShareOfLabour(1)=sum(sum(sum(Labour_pdf(Partion1Indicator))));
ShareOfLabour(2)=sum(sum(sum(Labour_pdf(Partion2Indicator))));
ShareOfLabour(3)=sum(sum(sum(Labour_pdf(Partion3Indicator))));
ShareOfLabour(4)=sum(sum(sum(Labour_pdf)));

Capital_pdf=shiftdim(ProbDensityFns(1,:,:,:),1);
ShareOfCapital(1)=sum(sum(sum(Capital_pdf(Partion1Indicator))));
ShareOfCapital(2)=sum(sum(sum(Capital_pdf(Partion2Indicator))));
ShareOfCapital(3)=sum(sum(sum(Capital_pdf(Partion3Indicator))));
ShareOfCapital(4)=sum(sum(sum(Capital_pdf)));

AverageEmployment(1)=sum(sum(sum(nbarValues(Partion1Indicator).*...
StationaryDist.pdf(Partion1Indicator))))/sum(sum(sum(nbarValues.*...
StationaryDist.pdf)));
AverageEmployment(2)=sum(sum(sum(nbarValues(Partion2Indicator).*...
StationaryDist.pdf(Partion2Indicator))))/sum(sum(sum(nbarValues.*...
StationaryDist.pdf)));
AverageEmployment(3)=sum(sum(sum(nbarValues(Partion3Indicator).*...
StationaryDist.pdf(Partion3Indicator))))/sum(sum(sum(nbarValues.*...
StationaryDist.pdf)));
AverageEmployment(4)=sum(sum(sum(nbarValues.*...
StationaryDist.pdf)))/sum(sum(sum(nbarValues.*...
StationaryDist.pdf)));

fprintf('Distribution statistics of benchmark economy  \n');
fprintf('                                <5     5 to 49     >=50
 total\n');
fprintf('Share of establishments  %8.2f  %8.2f  %8.2f  %8.2f  \n',
 ShareOfEstablishments);
fprintf('Share of output          %8.2f  %8.2f  %8.2f  %8.2f\n',
 ShareOfOutput);
fprintf('Share of labour          %8.2f  %8.2f  %8.2f  %8.2f\n',
 ShareOfLabour);
fprintf('Share of capital         %8.2f  %8.2f  %8.2f  %8.2f\n',
 ShareOfCapital);
fprintf('Share of employment      %8.2f  %8.2f  %8.2f  %8.2f\n',
 AverageEmployment);
```

Distribution statistics of benchmark economy

	<5	5 to 49	>=50	total
Share of establishments	*0.33*	*0.28*	*0.39*	*1.00*
Share of output	*0.02*	*0.09*	*0.89*	*1.00*
Share of labour	*0.02*	*0.09*	*0.89*	*1.00*
Share of capital	*0.15*	*0.33*	*0.52*	*1.00*
Share of employment	*0.02*	*0.09*	*0.89*	*1.00*

Display some output about the solution

```
fprintf('The equilibrium output price is p=%.4f \n', Params.p)
```

```
fprintf('The equilibrium value for the mass of entrants is Ne=%.4f
 \n', Params.Ne)

fprintf('Average Labor is n=%.4f \n', Output.perN)
fprintf('Average Capital is k=%.4f \n', Output.perK)
fprintf('Average Output is y=%.4f \n', Output.perY)
fprintf('Total Factor Productivity is TFP=%.4f \n', Output.TFP)


toc;
```

```
The equilibrium output price is p=0.4784
The equilibrium value for the mass of entrants is Ne=0.1193
Average Labor is n=0.8395
Average Capital is k=0.5669
Average Output is y=1.6790
Total Factor Productivity is TFP=2.2500
Elapsed time is 310.492459 seconds.
```

Published with MATLAB® R2018a