FF_VFI_AZ_BISEC_VEC Dynamic Savings Problem Vectorized Continuous Exact

back to Fan's Intro Math for Econon, Matlab Examples, or Dynamic Asset Repositories

This is the example vignette for function: **ff_vfi_az_bisec_vec** from the **MEconTools Package.** This function solves the dynamic programming problem for a (a,z) model. Households can save a, and face AR(1) shock z. The problem is solved over the infinite horizon. This is a vectorized code, it is much faster for larger state-space problems then looped code.

The code uses continuous choices, solved with bi(multi)section. The state-space is on a grid, but choice grids are in terms of percentage of resources available, which is individual specific, to save and solved exactly up to ((1/(2)^16)*100=0.001525878) percentage of cash on hand. THe **ff_vfi_az_vec** from the **MEconTools Package** solves the same problem using vectorized common grid code where the choice set and state space share the same grid.

This is the vectorized code, its speed is much faster than the looped code. The function is designed to have small memory footprint and requires low computing resources, yet is fast.

Links to Four Code:

Four Core Savings/Borrowing Dynamic Programming Solution Functions that are functions in the **MEconTools Package.** :

- Common Choice and States Grid ***Loop***: **ff_vfi_az_loop**, slow should use for testing new models
- Common Choice and States Grid ***Vectorized***: **ff_vfi_az_vec**, fast good for many purposes
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand ***Loop***: **ff_vfi_az_bisec_loop**, high precision even with small grid
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand ***Vectorized***: **ff_vfi_az_bisec_vec**, precision and speed

Test FF_VFI_AZ_BISEC_VEC Defaults

Call the function with defaults. By default, shows the asset policy function summary. Model parameters can be changed by the mp_params.

```
%mp_params
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('fl_crra') = 1.5;
mp_params('fl_beta') = 0.94;
% call function
ff_vfi_az_bisec_vec(mp_params);
```

```
Elapsed time is 0.668798 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
          i    idx    ndim    numel    rowN    colN     sum      mean       std      coefvari    min      max  
          _    ___    ____    _____    ____    ____    _____    ______    ______    ________    ___    ______
```

```
    ap    1     1      2       700      100       7      16866     24.094     14.071     0.58399       0      50.252
```

```
xxx TABLE:ap xxxxxxxxxxxxxxxxxx
               c1          c2          c3          c4          c5          c6         c7   
            _______     _______     _______     _______     _______     _______     ______

    r1            0           0           0           0     0.13188     0.66203     1.9859
    r2      0.25914     0.26426     0.29511     0.39221     0.57697      1.1208     2.4569
    r3      0.65371     0.66543     0.70966     0.82502       1.029       1.582     2.9298
    r4       1.0748      1.0921      1.1447      1.2698      1.5151      2.0481     3.4046
    r5       1.5152      1.5319      1.5903       1.721      2.0011      2.5252     3.8802
    r96      45.561      45.615      45.712      45.887      46.192      46.835     48.252
    r97      46.049      46.104      46.201      46.377      46.681      47.325     48.743
    r98       46.54      46.593       46.69      46.866      47.171      47.815     49.235
    r99      47.029      47.082      47.179      47.356      47.661      48.304     49.734
    r100     47.518      47.572       47.67      47.845       48.15      48.793     50.252
```

Test FF_VFI_AZ_BISEC_VEC Speed Tests

Call the function with defaults. By default, shows the asset policy function summary. Model parameters can be changed by the mp_params.

```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_timer') = true;
mp_support('ls_ffcmd') = {};
```

A grid 50, shock grid 5:

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 50;
mp_params('it_z_n') = 5;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 0.336083 seconds.
```

A grid 750, shock grid 15:

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 750;
mp_params('it_z_n') = 15;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 23.612756 seconds.
```

A grid 600, shock grid 45:

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 600;
mp_params('it_z_n') = 45;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 46.281741 seconds.
```

Test FF_VFI_AZ_BISEC_VEC Control Outputs

Run the function first without any outputs;

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 50;
mp_params('it_z_n') = 5;
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_timer') = false;
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
```

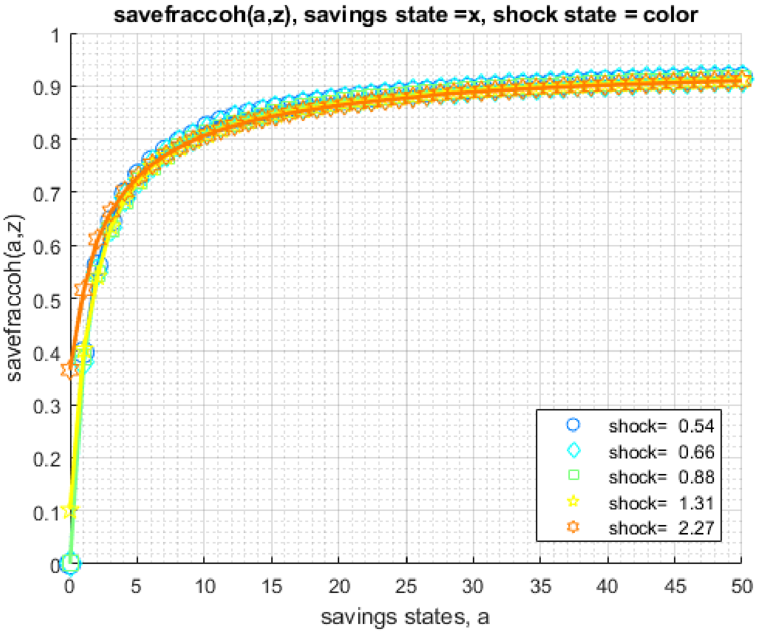
Run the function and show policy function for savings choice. For ls_ffcmd, ls_ffsna, ls_ffgrh, can include these: 'v', 'ap', 'c', 'y', 'coh', 'savefraccoh'. These are value, aprime savings choice, consumption, income, cash on hand, and savings fraction as cash-on-hand.

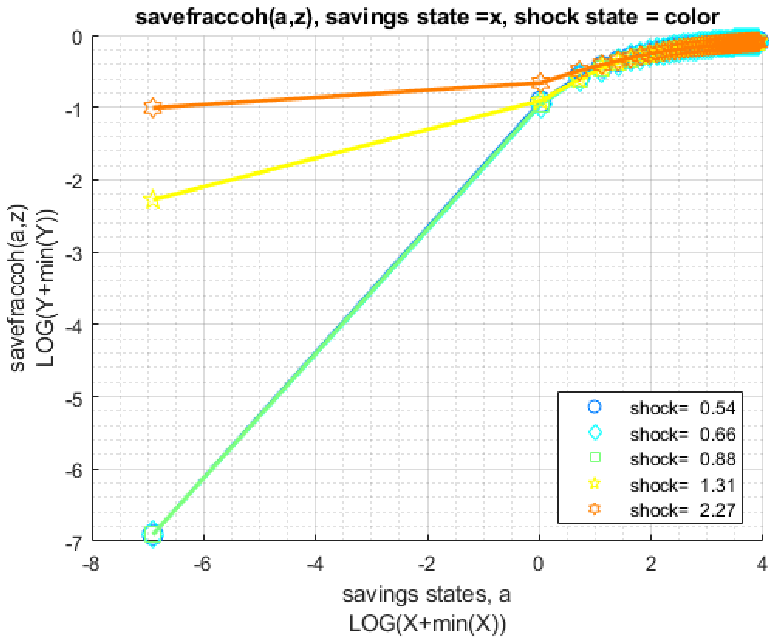
```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
% ls_ffcmd: summary print which outcomes
mp_support('ls_ffcmd') = {};
% ls_ffsna: detail print which outcomes
mp_support('ls_ffsna') = {'savefraccoh'};
% ls_ffgrh: graphical print which outcomes
mp_support('ls_ffgrh') = {'savefraccoh'};
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 0.470134 seconds.
xxx  ff_vfi_az_vec, outcome=savefraccoh  xxxxxxxxxxxxxxxxxxxxxxxxxx
   group      a      mean_z_0_54195    mean_z_0_66401    mean_z_0_88162    mean_z_1_3095    mean_z_2_2745
   _____    ______    ______________    ______________    ______________    _____________    _____________

     1         0              0                 0                 0            0.10165          0.36531
     2    1.0204        0.39833           0.38191           0.38826            0.40373          0.51573
     3    2.0408        0.56236           0.54875           0.54619            0.54161          0.61104
     4    3.0612        0.64616           0.63545            0.6306            0.63591          0.66349
     5    4.0816        0.69783            0.6891            0.6837             0.6873          0.70155
     6     5.102        0.73323            0.7259           0.72068            0.72184           0.7302
     7    6.1224        0.75917           0.75291           0.74803            0.74784          0.75257
     8    7.1429        0.77909           0.77363           0.76911            0.76817          0.77058
     9    8.1633        0.79493           0.79011           0.78593            0.78452          0.78541
    10    9.1837        0.80787           0.80354           0.79969            0.79801          0.79783
    11    10.204        0.82343           0.81474           0.81114             0.8093          0.80839
    12    11.224        0.83412           0.82591           0.82084            0.81895          0.81748
    13    12.245        0.84116           0.83759           0.82917            0.82725          0.82542
    14    13.265        0.84733           0.84434           0.83637            0.83448          0.83241
    15    14.286        0.85282           0.84998           0.84272            0.84083          0.83857
    16    15.306         0.8577           0.85508           0.84947            0.84647          0.84406
    17    16.327        0.86213           0.85966           0.85685            0.85151          0.84901
    18    17.347        0.86613           0.86378           0.86143              0.856          0.85349
    19    18.367        0.86976            0.8675           0.86521            0.86008          0.85755
    20    19.388        0.87305           0.87092           0.86869            0.86384          0.86128
    21    20.408        0.87608           0.87403            0.8719            0.86726          0.86472
    22    21.429        0.87885            0.8769           0.87486             0.8704           0.8679
    23    22.449        0.88145           0.87955            0.8776            0.87333          0.87083
    24    23.469        0.88383           0.88203           0.88013            0.87601          0.87354
    25     24.49        0.88602           0.88432           0.88248            0.87852          0.87608
    26     25.51         0.8881           0.88645           0.88468            0.88087          0.87843
    27    26.531        0.89002           0.88844           0.88673            0.88361          0.88126
    28    27.551        0.89185           0.89033           0.88865            0.88685          0.88444
    29    28.571        0.89353           0.89207           0.89045            0.88895          0.88636
    30    29.592        0.89515           0.89371           0.89216            0.89063          0.88813
    31    30.612        0.89664           0.89527           0.89375            0.89225          0.88978
```

```
32      31.633      0.89808         0.89674         0.89524         0.89378         0.89136
33      32.653      0.89942         0.89811         0.89667         0.89521         0.89286
34      33.673      0.90067         0.89942         0.89802         0.89658         0.89429
35      34.694      0.90189         0.90067          0.8993          0.8979         0.89564
36      35.714      0.90305         0.90186         0.90052         0.89915         0.89692
37      36.735      0.90473         0.90299         0.90168         0.90034         0.89814
38      37.755      0.90662         0.90406         0.90278         0.90147         0.89933
39      38.776      0.90763         0.90507         0.90382         0.90256         0.90043
40      39.796      0.90852         0.90668         0.90482         0.90357          0.9015
41      40.816      0.90937         0.90833          0.9058         0.90458         0.90253
42      41.837      0.91019         0.90919         0.90671         0.90552         0.90351
43      42.857      0.91099         0.91001          0.9076         0.90641         0.90446
44      43.878      0.91172         0.91077         0.90842         0.90729         0.90534
45      44.898      0.91245         0.91154         0.90925         0.90812          0.9062
46      45.918      0.91315         0.91224         0.91041         0.90891         0.90702
47      46.939      0.91383         0.91294         0.91181         0.90971         0.90784
48      47.959       0.9145         0.91361         0.91264         0.91044         0.90861
49       48.98      0.91511         0.91425         0.91328         0.91114         0.90934
50          50      0.91572         0.91486         0.91392         0.91181         0.91004
```





Run the function and show summaries for savings and fraction of coh saved:

```
mp_params('it_a_n') = 100;
mp_params('it_z_n') = 9;
mp_support('ls_ffcmd') = {'ap', 'savefraccoh'};
mp_support('ls_ffsna') = {};
mp_support('ls_ffgrh') = {};
mp_support('bl_vfi_store_all') = true; % store c(a,z), y(a,z)
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 0.716025 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean        std      coefvari    min
                   _    ___    ____    _____    ____    ____    _____    _______     ______    ________    ___    _

    ap             1     1      2       900     100      9      21835     24.261     14.095     0.58096     0
    savefraccoh    2     2      2       900     100      9     754.27    0.83808     0.1259     0.15023     0     0.

xxx TABLE:ap xxxxxxxxxxxxxxxxxx
                c1         c2         c3         c4         c5          c6          c7          c8         c9
             _______    _______    _______    _______    _______    ________    _______    _______    ______

    r1             0          0          0          0          0    0.082559    0.50504     1.2988     3.1416
    r2       0.26067    0.25936    0.26888    0.30308    0.39296     0.52492    0.96211     1.7672     3.6183
    r3       0.65383    0.65589    0.67297    0.71974    0.82473      1.0101     1.4185     2.2377     4.0955
    r4        1.0734     1.0789     1.1015     1.1556     1.2679      1.4909     1.8821     2.7095     4.5736
    r5        1.5151     1.5159     1.5427     1.6019       1.72      1.9489      2.349     3.1825     5.0521
    r96       45.547      45.58     45.636      45.73     45.888      46.134     46.603      47.52      49.54
    r97       46.036     46.069     46.126      46.22     46.377      46.622     47.092     48.009     50.057
    r98       46.525     46.559     46.615      46.71     46.867      47.112     47.583     48.501     50.575
    r99       47.014     47.049     47.104     47.198     47.357      47.601     48.072     48.992     51.092
    r100      47.503     47.537     47.593     47.687     47.845      48.091     48.561     49.495      51.61

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
                c1         c2         c3         c4         c5          c6          c7          c8          c9
             _______    _______    _______    _______    _______    ________    _______    _______    _______
    r1          0          0          0          0          0    0.056268    0.24587    0.41301    0.58272
    r2    0.23098      0.217    0.20843    0.21203    0.23925     0.26445     0.3741    0.48253    0.61235
    r3    0.39717    0.38292    0.37227    0.36965    0.38179     0.40361    0.45915    0.53532    0.63728
    r4    0.49605    0.48369    0.47368    0.46883    0.47347     0.49364    0.52177    0.57677    0.65861
    r5    0.56502    0.55159    0.54262    0.53709    0.53825     0.55086    0.56947    0.61021    0.67704
    r96   0.91477    0.91422    0.91361    0.91294    0.91221      0.9109    0.90961    0.90818    0.90781
    r97   0.91508    0.91453    0.91395    0.91328    0.91254     0.91123    0.90998    0.90855    0.90867
    r98   0.91538    0.91486    0.91425    0.91361    0.91288     0.91157    0.91035    0.90894    0.90952
    r99   0.91569    0.91517    0.91456    0.91392    0.91322      0.9119    0.91068    0.90934    0.91035
    r100  0.91596    0.91544    0.91486    0.91422    0.91352     0.91224    0.91102    0.90992    0.91117
```

Test FF_VFI_AZ_BISEC_VEC Change Interest Rate and Discount

Show only save fraction of cash on hand:

```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
mp_support('ls_ffcmd') = {'savefraccoh'};
mp_support('ls_ffsna') = {};
mp_support('ls_ffgrh') = {};
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 750;
mp_params('it_z_n') = 9;
mp_params('fl_a_max') = 50;
mp_params('st_grid_type') = 'grid_powerspace';
```

Solve the model with several different interest rates and discount factor:

```
% Lower Savings Incentives
mp_params('fl_beta') = 0.80;
mp_params('fl_r') = 0.01;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 4.541023 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean       std       coefvari    min
                   _    ___    ____    _____    ____    ____    ______    _______    _______    ________    ___

    savefraccoh    1     1      2      6750     750      9      3318.4    0.49162    0.27766    0.56478      0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
                c1         c2         c3         c4         c5         c6          c7          c8         c9
              _______    _______    _______    _______    _______    _______    ________    _______    _______

    r1              0          0          0          0          0          0    0.023584     0.1329    0.29705
    r2              0          0          0          0          0          0    0.023584     0.1329    0.29705
    r3              0          0          0          0          0          0    0.023584     0.1329    0.29705
    r4              0          0          0          0          0          0    0.023584     0.1329    0.29705
    r5              0          0          0          0          0          0    0.023584     0.1329    0.29705
    r746      0.80439    0.80299    0.80094    0.79856    0.79588    0.79222     0.78825    0.78263    0.77647
    r747      0.80467    0.80314     0.8011    0.79875    0.79606     0.7924     0.78846    0.78285    0.77687
    r748      0.80491    0.80329    0.80125     0.7989    0.79621    0.79255     0.78864    0.78315    0.77732
    r749      0.80515    0.80341    0.80137    0.79905     0.7964    0.79273     0.78883    0.78352    0.77769
    r750      0.80534    0.80357    0.80152     0.7992    0.79655    0.79292     0.78904    0.78388     0.7779
```

```
% Higher Savings Incentives
mp_params('fl_beta') = 0.95;
mp_params('fl_r') = 0.04;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 17.994960 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean        std      coefvari    min
                   _    ___    ____    _____    ____    ____    ______    _______    _______    ________    ___    _

    savefraccoh    1     1      2      6750     750      9      4493.5    0.66571    0.28784    0.43238      0     0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
              c1         c2         c3         c4          c5         c6         c7         c8         c9
            _______    _______    _______    _______    ________    _______    _______    _______    _______

    r1            0          0          0          0    0.032007    0.15008    0.31087    0.48467    0.64488
    r2            0          0          0          0    0.032007    0.15008    0.31087    0.48467    0.64488
    r3            0          0          0          0    0.032007    0.15008    0.31087    0.48467    0.64488
    r4            0          0          0          0    0.032007    0.15008    0.31087    0.48467    0.64488
    r5            0          0          0          0    0.032007    0.15008    0.31087    0.48467    0.64488
    r746     0.9289     0.9285    0.92805    0.92734     0.92664    0.92594    0.92515    0.92545    0.92777
    r747    0.92902     0.9286    0.92814    0.92747     0.92673    0.92606     0.9253    0.92573    0.92802
    r748    0.92911    0.92869    0.92826    0.92756     0.92686    0.92618    0.92545      0.926    0.92829
    r749    0.92921    0.92881    0.92835    0.92768     0.92698    0.92631    0.92564    0.92631    0.92857
    r750     0.9293     0.9289    0.92844    0.92777     0.92707    0.92643    0.92591    0.92658    0.92881
```

Test FF_VFI_AZ_BISEC_VEC Changing Risk Aversion

Here, again, show fraction of coh saved in summary tabular form, but also show it graphically.

```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
mp_support('ls_ffcmd') = {'savefraccoh'};
mp_support('ls_ffsna') = {};
mp_support('ls_ffgrh') = {'savefraccoh'};
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 750;
mp_params('it_z_n') = 9;
mp_params('fl_a_max') = 50;
mp_params('st_grid_type') = 'grid_powerspace';
```

Solve the model with different risk aversion levels, higher preferences for risk:

```
% Lower Risk Aversion
mp_params('fl_crra') = 0.5;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

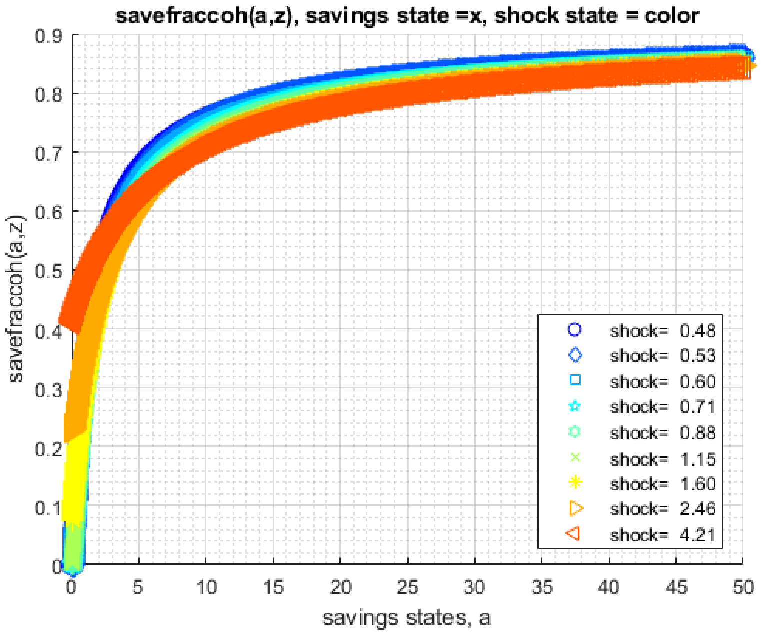
```
Elapsed time is 13.815578 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean        std      coefvari    min
```

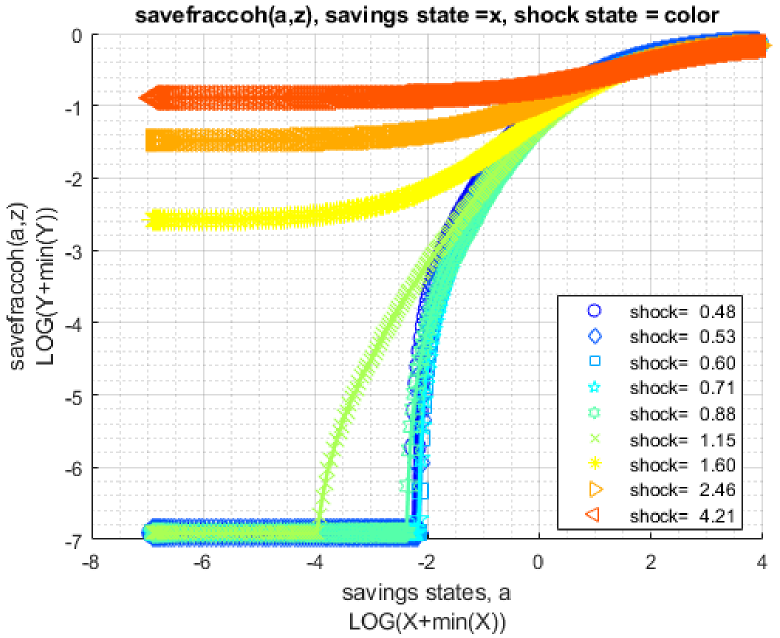
```
                 _     ___     ____     _____     ____     ____     ______     _______     _______     ________     ___     _

    savefraccoh  1      1       2       6750       750       9      3735.7     0.55343     0.28972      0.5235       0      0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
                  c1          c2          c3          c4          c5          c6            c7          c8          c9
              _______     _______     _______     _______     _______     _______     ________     _______     _______

    r1              0           0           0           0           0           0     0.074609     0.22661     0.41036
    r2              0           0           0           0           0           0     0.074609     0.22661     0.41036
    r3              0           0           0           0           0           0     0.074609     0.22661     0.41036
    r4              0           0           0           0           0           0     0.074609     0.22661     0.41036
    r5              0           0           0           0           0           0     0.074609     0.22664     0.41039
    r746      0.85941     0.85828     0.85703      0.8556     0.85398     0.85178      0.84907     0.84583     0.84211
    r747      0.85957     0.85844     0.85719     0.85575     0.85413     0.85194      0.84925     0.84602     0.84229
    r748      0.85969     0.85859     0.85734      0.8559     0.85429     0.85212      0.84943      0.8462     0.84248
    r749      0.85984     0.85871     0.85749     0.85606     0.85447     0.85227      0.84962     0.84638     0.84266
    r750      0.85999     0.85889     0.85761     0.85621     0.85462     0.85246      0.84977     0.84657     0.84284
```





When risk aversion increases, at every state-space point, the household wants to save more.

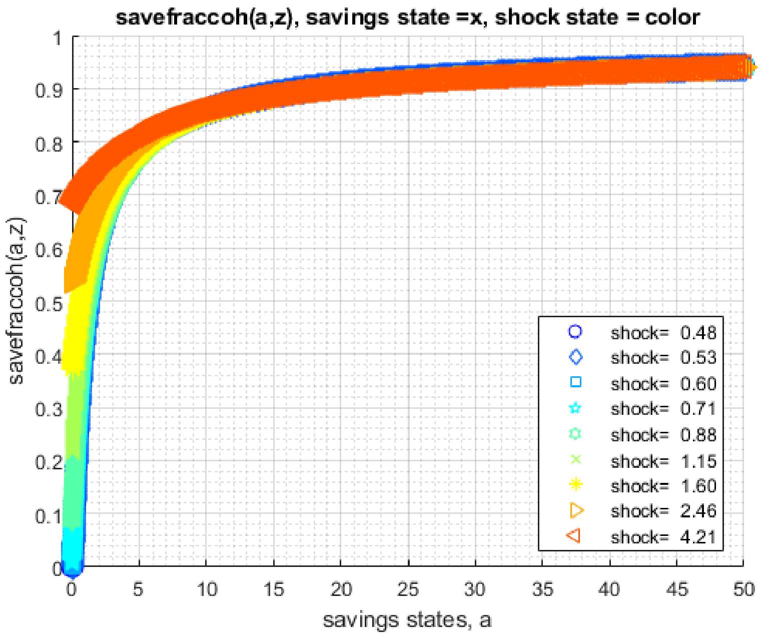
```
% Higher Risk Aversion
mp_params('fl_crra') = 5;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

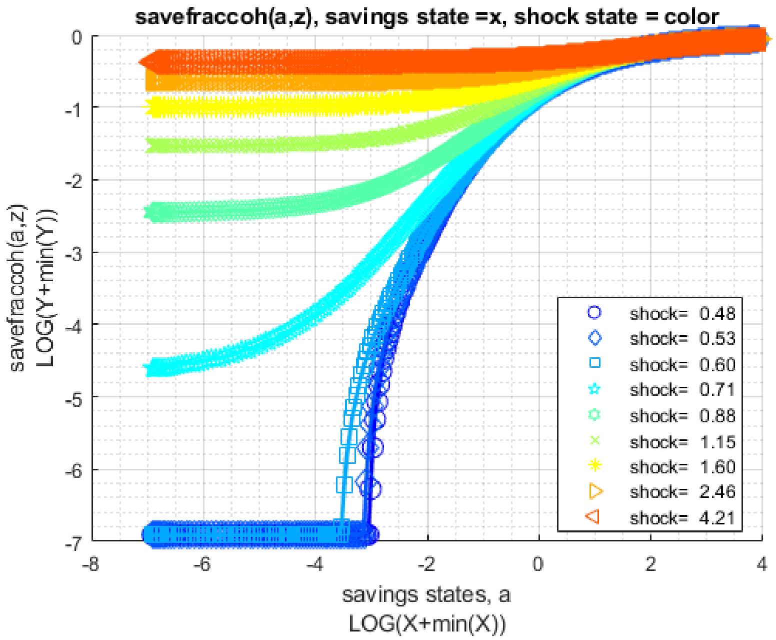
```
Elapsed time is 13.688997 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN    sum      mean       std      coefvari    min    ma
                   _    ___    ____    _____    ____    ____    ____    _______    ______    ________    ___    ____

    savefraccoh    1     1      2      6750     750      9      4640    0.68741    0.2821    0.41039      0     0.94

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
              c1         c2         c3          c4           c5          c6         c7         c8         c9   
            _______    _______    _______    _________    ________    _______    _______    _______    _______

    r1            0          0          0    0.0089972    0.085107    0.21343    0.37139    0.53578    0.68562
    r2            0          0          0    0.0089972    0.085107    0.21343    0.37139    0.53578    0.68562
    r3            0          0          0    0.0089972    0.085107    0.21343    0.37139    0.53578    0.68562
    r4            0          0          0    0.0089972    0.085107    0.21343    0.37139    0.53578    0.68562
    r5            0          0          0    0.0089972    0.085107    0.21343    0.37139    0.53578    0.68562
    r746    0.94105    0.94074    0.94041      0.93992     0.93943    0.93897    0.93848    0.93885    0.94083
    r747    0.94114    0.94083    0.94053      0.94004     0.93955    0.93909    0.93864    0.93909    0.94105
    r748    0.94126    0.94095    0.94062      0.94016     0.93964    0.93922    0.93879    0.93931    0.94126
    r749    0.94135    0.94105    0.94074      0.94025     0.93976    0.93934    0.93894    0.93955    0.94147
    r750    0.94144    0.94117    0.94083      0.94038     0.93989    0.93946    0.93915    0.93976    0.94169
```





Test FF_VFI_AZ_BISEC_VEC with Higher Uncertainty

Increase the standard deviation of the Shock.

```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
mp_support('ls_ffcmd') = {'savefraccoh'};
mp_support('ls_ffsna') = {};
mp_support('ls_ffgrh') = {};
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 750;
mp_params('it_z_n') = 9;
mp_params('fl_a_max') = 50;
mp_params('st_grid_type') = 'grid_powerspace';
```

Lower standard deviation of shock:

```
% Lower Risk Aversion
mp_params('fl_shk_std') = 0.10;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 14.533433 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean        std      coefvari    min    
                   _    ___    ____    _____    ____    ____    ______    _______    ______    ________    ___    _

    savefraccoh    1     1      2      6750     750      9      4025.4    0.59636    0.3153     0.5287      0     0.

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
           c1          c2          c3          c4          c5          c6           c7          c8          c9    
        _______     _______     _______     _______     _______     ________     ________    _______     _______
    r1              0          0          0          0          0    0.012568    0.063073    0.13604    0.22228
    r2              0          0          0          0          0    0.012568    0.063073    0.13604    0.22228
    r3              0          0          0          0          0    0.012598    0.063073    0.13604    0.22228
    r4              0          0          0          0          0    0.012598    0.063073    0.13604    0.22228
    r5              0          0          0          0          0    0.012598    0.063073    0.13604    0.22228
    r746      0.91276    0.91248    0.91196    0.91163    0.91111     0.91077     0.91025    0.90977    0.90913
    r747      0.91291     0.9126    0.91209    0.91178    0.91126     0.91093     0.91041    0.90992    0.90925
    r748      0.91303    0.91276    0.91224    0.91193    0.91138     0.91108     0.91056    0.91004     0.9094
    r749      0.91318    0.91288    0.91236    0.91206    0.91154      0.9112     0.91068    0.91019    0.90955
    r750      0.91331      0.913    0.91251    0.91221    0.91169     0.91135     0.91083    0.91035    0.90971
```

Higher shock standard deviation: low shock high asset save more, high shock more asset save less, high shock low asset save more:

```
% Higher Risk Aversion
mp_params('fl_shk_std') = 0.40;
ff_vfi_az_bisec_vec(mp_params, mp_support);
```

```
Elapsed time is 14.829381 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim    numel    rowN    colN     sum       mean        std       coefvari     min
                   _    ___    ____    _____    ____    ____    ______    _______    _______    ________    ___

    savefraccoh    1     1      2      6750     750      9      5105.2    0.75633    0.26373     0.3487      0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx
               c1         c2         c3         c4          c5          c6         c7         c8         c9
             _______    _______    _______    _______    ________    _______    _______    _______    _______

    r1              0          0          0          0    0.031641     0.2456    0.55455    0.80573    0.96207
    r2              0          0          0          0    0.031641     0.2456    0.55455    0.80573    0.96207
    r3              0          0          0          0    0.031641     0.2456    0.55455    0.80573    0.96207
    r4              0          0          0          0    0.031671     0.2456    0.55455    0.80573    0.96207
    r5              0          0          0          0    0.031671     0.2456    0.55455    0.80573    0.96207
    r746      0.93336    0.93287    0.93226    0.93149      0.9303     0.9289    0.92725    0.93293    0.97416
    r747      0.93342    0.93293    0.93232    0.93159      0.9304    0.92899    0.92737    0.93317    0.97419
    r748      0.93348    0.93299    0.93241    0.93165     0.93046    0.92905     0.9275    0.93339    0.97422
    r749      0.93354    0.93305    0.93247    0.93171     0.93055    0.92914    0.92762    0.93363    0.97425
    r750       0.9336    0.93311    0.93253    0.93177     0.93061    0.92924    0.92771    0.93384    0.97428
```