FF_VFI_AZ_MZOOM_VEC Savings Vectorized Exact (VALUE) Examples

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

This is the example vignette for function: **ff_vfi_az_mzoom_vec** from the **MEconTools Package.** This function solves the dynamic programming problem for a (a,z) model. The state-space is on a grid, but choice grids are in terms of percentage of resources to save and solved exactly.

This is a **vectorized** code for **continuous** choices, solved with the **mzoom** algorithm. In contrast to the **bisection** based solution, this is slower, but this does not rely on first order conditions.

Links to Other Code:

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

- Common Choice and States Grid ***Loop***: **ff_vfi_az_loop**
- Common Choice and States Grid ***Vectorized***: **ff_vfi_az_vec**
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand, rely on FOC, ***Loop***: **ff_vfi_az_bisec_loop**
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand, rely on FOC ***Vectorized***: **ff_vfi_az_bisec_vec**
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand, VALUE comparison, ***Loop***: **ff_vfi_az_mzoom_loop**
- States Grid + Continuous Exact Savings as Share of Cash-on-Hand, VALUE comparison, ***Vectorized***: **ff_vfi_az_mzoom_vec**

Test FF_VFI_AZ_MZOOM_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_mzoom_vec(mp_params);
```

```
Elapsed time is 6.126702 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      9861.5    14.088    14.386     1.0212      0     50.115

xxx TABLE:ap xxxxxxxxxxxxxxxxx
             c1        c2        c3        c4        c5        c6        c7
```

r1	0	0	0	0.05343	0.25568	0.60598	1.1155
r2	0	0	0	0.053451	0.25571	0.60652	1.1161
r3	0	0	0	0.056468	0.25574	0.60897	1.1174
r4	0	0	0	0.061232	0.25995	0.61042	1.1238
r5	0	0	0	0.065929	0.2689	0.61091	1.1323
r96	43.387	43.517	43.7	43.922	44.221	44.657	45.225
r97	44.562	44.694	44.876	45.095	45.392	45.847	46.394
r98	45.758	45.89	46.071	46.287	46.583	47.037	47.596
r99	46.972	47.103	47.285	47.5	47.794	48.247	48.812
r100	48.183	48.337	48.518	48.732	49.025	49.478	50.115

Test FF_VFI_AZ_MZOOM_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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 1.996365 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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 337.171768 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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 1758.273287 seconds.
```

Test FF_VFI_AZ_MZOOM_VEC Control Outputs

Run the function first without any outputs, but only the timer.

```
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') = true;
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
```

```
mp_support('ls_ffcmd') = {};
ff_vfi_az_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 1.091918 seconds.
```

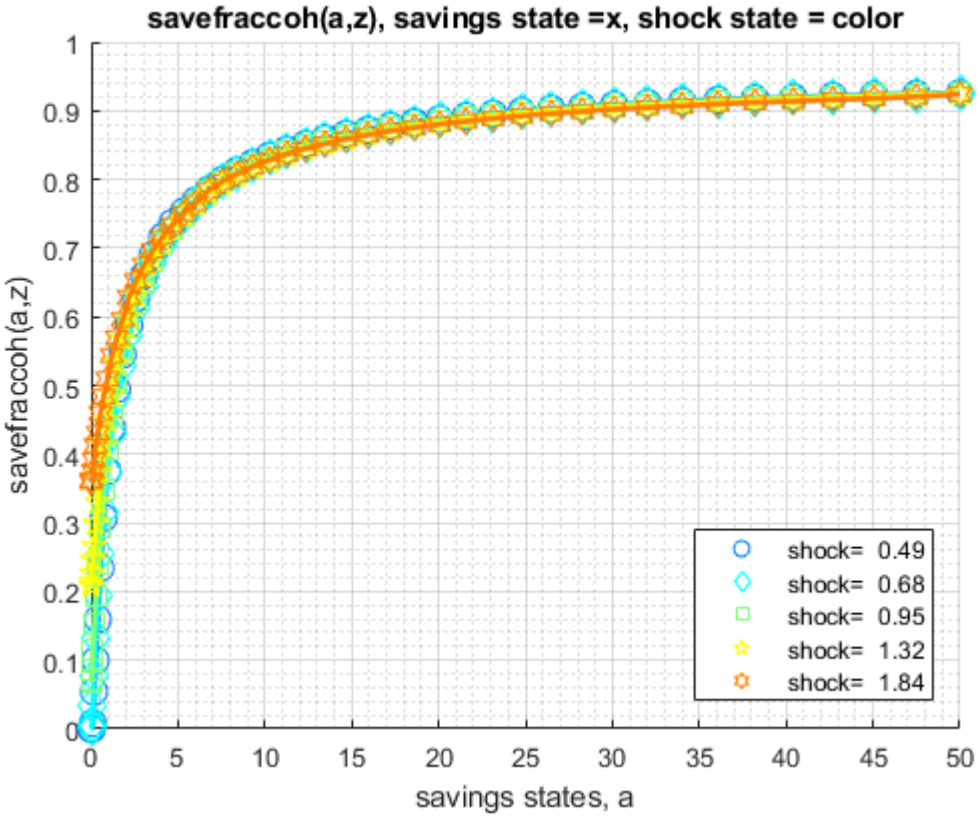
Run the function and show policy function for savings choice. For ls_ffcmd, ls_ffsna, ls_ffgrh, can include these: 'v', 'ap', 'c', 'y', 'coh', 'savefraccoh'. These are value, aprime savings choice, consumption, income, cash on hand, and savings fraction as cash-on-hand.

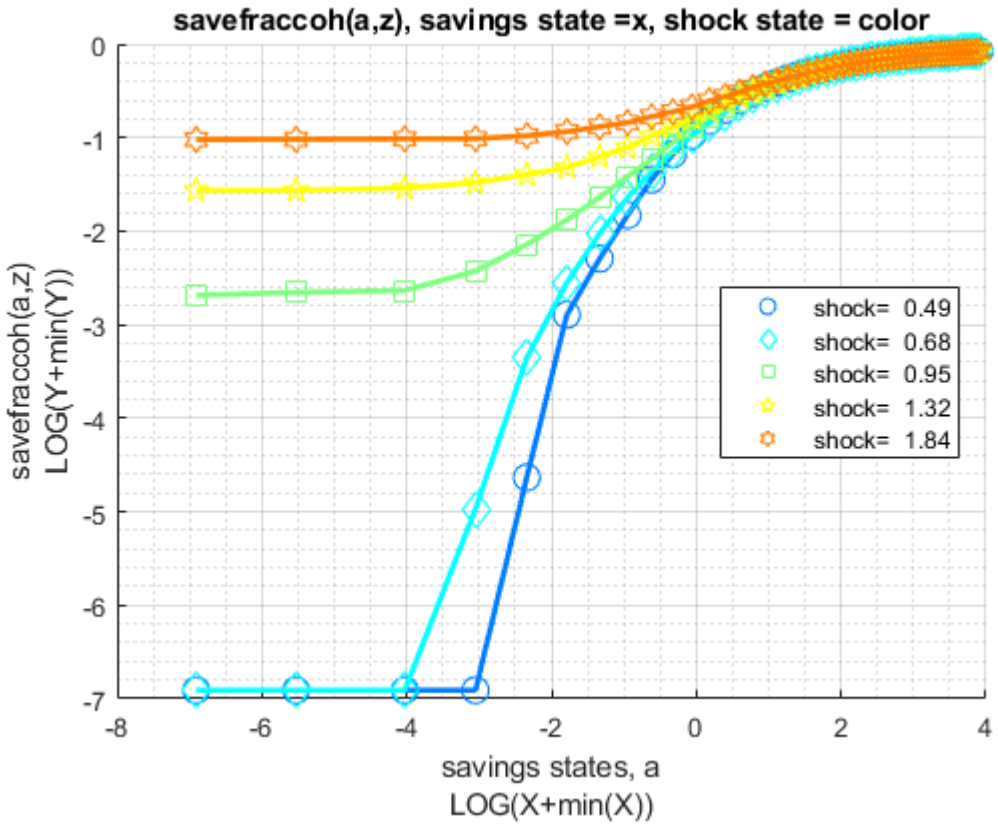
```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
% ls_ffcmd: summary print which outcomes
mp_support('ls_ffcmd') = {};
% ls_ffsna: detail print which outcomes
mp_support('ls_ffsna') = {'savefraccoh'};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

```
Elapsed time is 1.090424 seconds.
xxx  ff_vfi_az_vec, outcome=savefraccoh  xxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	a	mean_z_0_4858	mean_z_0_67798	mean_z_0_9462	mean_z_1_3205	mean_z_1_8429
1	0	0	0	0.067148	0.2084	0.35952
2	0.002975	0	0	0.069345	0.20826	0.36029
3	0.016829	0	0	0.070749	0.2136	0.36206
4	0.046375	0	0.0059631	0.08732	0.22641	0.36263
5	0.095198	0.008725	0.033935	0.11637	0.24674	0.3747
6	0.1663	0.054327	0.077152	0.15198	0.26635	0.39127
7	0.26234	0.099882	0.13131	0.1936	0.29922	0.41248
8	0.38568	0.15954	0.1928	0.24107	0.33005	0.43049
9	0.53852	0.23411	0.25482	0.29164	0.37407	0.4593
10	0.72291	0.30704	0.31604	0.34806	0.41148	0.48371
11	0.94076	0.37567	0.37487	0.40768	0.44925	0.50972
12	1.1939	0.43849	0.42939	0.4573	0.48691	0.54333
13	1.484	0.49491	0.48129	0.50332	0.53253	0.56934
14	1.8128	0.54486	0.53013	0.54642	0.56773	0.59615
15	2.1817	0.58868	0.57335	0.58545	0.60016	0.62817
16	2.5924	0.6271	0.61254	0.62056	0.63057	0.65247
17	3.0463	0.66058	0.6468	0.65237	0.65884	0.67518
18	3.5449	0.69019	0.67699	0.68069	0.68379	0.69636
19	4.0894	0.71615	0.70375	0.7058	0.70719	0.7159
20	4.6813	0.73661	0.72701	0.72843	0.72781	0.73341
21	5.3218	0.75302	0.7481	0.74821	0.74661	0.74981
22	6.0121	0.76912	0.76622	0.76622	0.76342	0.76534
23	6.7536	0.78503	0.78285	0.78223	0.77885	0.78383
24	7.5474	0.79943	0.79703	0.79623	0.79223	0.79677
25	8.3948	0.81264	0.81024	0.8093	0.80504	0.80784
26	9.2967	0.82384	0.82198	0.82064	0.81634	0.81874
27	10.254	0.83447	0.83225	0.83065	0.82653	0.82824
28	11.269	0.84345	0.84174	0.84025	0.83545	0.83703
29	12.342	0.85185	0.85017	0.84865	0.84417	0.84497
30	13.473	0.85962	0.85746	0.85642	0.85178	0.85185
31	14.665	0.86626	0.86466	0.86306	0.85873	0.85895
32	15.918	0.87226	0.87066	0.86959	0.86504	0.86466
33	17.233	0.87786	0.87626	0.87529	0.87146	0.87061
34	18.611	0.88332	0.88182	0.88026	0.87766	0.87546
35	20.053	0.888	0.88656	0.88507	0.88267	0.88026
36	21.56	0.89187	0.89087	0.88947	0.88825	0.88483

37	23.133	0.89587	0.89484	0.89347	0.89256	0.88867
38	24.773	0.8997	0.89827	0.89727	0.89587	0.89259
39	26.481	0.903	0.90147	0.90066	0.89964	0.89587
40	28.258	0.90601	0.90467	0.90376	0.90278	0.89907
41	30.104	0.90881	0.9077	0.90628	0.90547	0.90216
42	32.021	0.91137	0.91035	0.90908	0.90838	0.90467
43	34.01	0.91377	0.91275	0.91148	0.91068	0.90708
44	36.07	0.91595	0.91468	0.91388	0.91308	0.90983
45	38.204	0.91788	0.91708	0.91617	0.91531	0.91204
46	40.412	0.91948	0.91868	0.91788	0.91708	0.91388
47	42.695	0.92168	0.92085	0.91998	0.91915	0.91604
48	45.053	0.92331	0.92251	0.92171	0.92091	0.91788
49	47.488	0.92485	0.92408	0.92331	0.92254	0.9202
50	50	0.92588	0.92555	0.92485	0.92423	0.92402





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

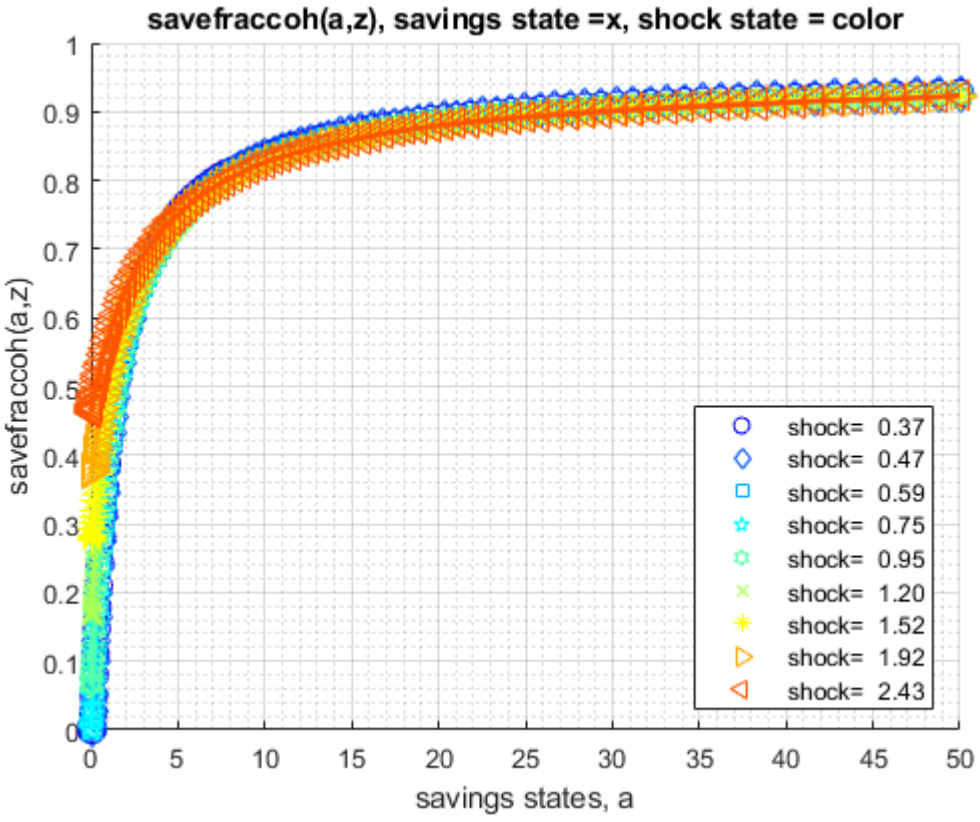
```
%mp_params
mp_params = containers.Map('KeyType','char', 'ValueType','any');
% mp_params('fl_crra') = 1.5;
% mp_params('fl_beta') = 0.94;
mp_params('it_a_n') = 100;
mp_params('it_z_n') = 9;
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = 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'};
% call function
ff_vfi_az_mzoom_vec(mp_params, mp_support);
```

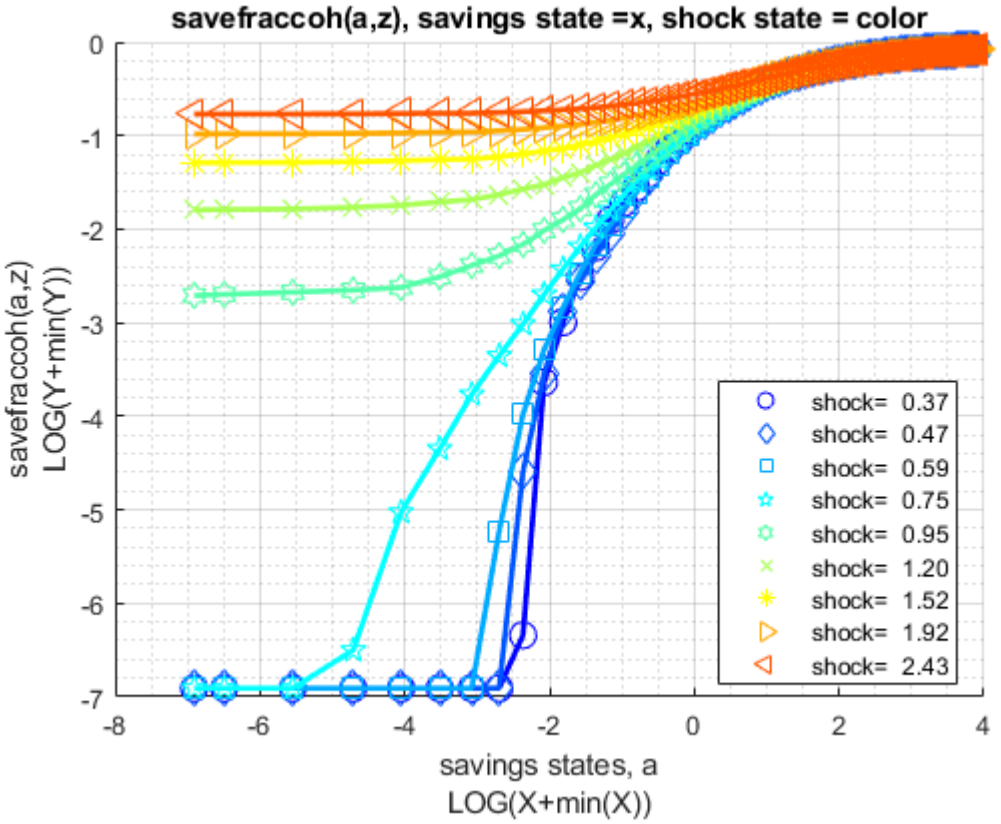
```
Elapsed time is 5.173849 seconds.
xxx  ff_vfi_az_vec, outcome=savefraccoh  xxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	a	mean_z_0_36853	mean_z_0_46648	mean_z_0_59047	mean_z_0_74742	mean_z_0_94608
1	0	0	0	0	0	0.065547
2	0.00051272	0	0	0	0	0.066347
3	0.0029004	0	0	0	0	0.067948
4	0.0079925	0	0	0	0.00050216	0.069549

5	0.016407	0	0	0	0.005563	0.071534
6	0.028662	0	0	0	0.011926	0.080274
7	0.045213	0	0	0	0.022095	0.090757
8	0.06647	0	0	0.0043625	0.033935	0.10076
9	0.092813	0.00076108	0.0091251	0.017748	0.047979	0.11397
10	0.12459	0.02539	0.027791	0.036336	0.066347	0.13237
11	0.16214	0.049062	0.054743	0.057497	0.087289	0.14878
12	0.20576	0.080353	0.076351	0.084213	0.11115	0.16729
13	0.25576	0.11036	0.10076	0.11357	0.13677	0.1944
14	0.31242	0.14798	0.12866	0.14076	0.16483	0.21731
15	0.37601	0.17839	0.16439	0.16895	0.194	0.24107
16	0.4468	0.2098	0.20032	0.1988	0.22401	0.26563
17	0.52503	0.24246	0.23721	0.23371	0.25482	0.29153
18	0.61095	0.28123	0.27422	0.26803	0.28577	0.31725
19	0.7048	0.31861	0.30964	0.30224	0.31644	0.34326
20	0.8068	0.35352	0.34406	0.33561	0.34646	0.37247
21	0.91719	0.38727	0.37774	0.36766	0.37639	0.40048
22	1.0362	0.42001	0.40688	0.39888	0.40495	0.42569
23	1.164	0.4501	0.43289	0.42881	0.43266	0.4501
24	1.3008	0.47851	0.45746	0.45719	0.45922	0.47371
25	1.4468	0.50572	0.48514	0.48371	0.48451	0.49652
26	1.6023	0.53093	0.51118	0.50952	0.50892	0.51852
27	1.7673	0.55214	0.53571	0.53333	0.53173	0.53973
28	1.9422	0.57052	0.55854	0.55614	0.55374	0.55981
29	2.127	0.58782	0.58031	0.57735	0.57415	0.57893
30	2.3221	0.60768	0.60016	0.59758	0.59375	0.59695
31	2.5275	0.62577	0.61947	0.61496	0.61226	0.61416
32	2.7434	0.64351	0.63697	0.63101	0.62956	0.63057
33	2.97	0.65976	0.65338	0.64537	0.64591	0.64617
34	3.2075	0.67458	0.66898	0.66058	0.66124	0.66058
35	3.456	0.68919	0.68379	0.67538	0.67538	0.67458
36	3.7158	0.7022	0.69739	0.68939	0.68928	0.68779
37	3.9869	0.7146	0.7098	0.7022	0.70205	0.70039
38	4.2696	0.72668	0.7218	0.7146	0.7138	0.71209
39	4.564	0.73741	0.73341	0.7262	0.7254	0.72317
40	4.8702	0.74798	0.74381	0.73711	0.73581	0.73341
41	5.1884	0.75768	0.75382	0.74727	0.74581	0.74348
42	5.5188	0.76679	0.7618	0.75684	0.75542	0.75281
43	5.8615	0.77502	0.76862	0.76542	0.76422	0.76165
44	6.2166	0.78303	0.77658	0.77422	0.77262	0.76996
45	6.5844	0.79063	0.78452	0.78223	0.78063	0.77742
46	6.9649	0.79783	0.79196	0.78983	0.78823	0.78529
47	7.3583	0.80499	0.79863	0.79695	0.79543	0.79223
48	7.7647	0.81024	0.80566	0.80343	0.80231	0.79863
49	8.1844	0.81504	0.81184	0.81003	0.80862	0.80504
50	8.6173	0.81984	0.81744	0.81584	0.81424	0.81104
51	9.0637	0.82544	0.82351	0.82144	0.82031	0.81664
52	9.5237	0.83065	0.82881	0.82664	0.82544	0.82224
53	9.9975	0.83545	0.83385	0.83217	0.83065	0.82744
54	10.485	0.84025	0.83863	0.83697	0.83545	0.83225
55	10.987	0.84494	0.84315	0.84155	0.84023	0.83703
56	11.502	0.84919	0.84705	0.84585	0.84425	0.84105
57	12.032	0.85319	0.85156	0.85002	0.84785	0.84562
58	12.577	0.85666	0.85506	0.85396	0.85174	0.84945
59	13.136	0.86064	0.85906	0.85746	0.85506	0.85338
60	13.709	0.86386	0.86226	0.86122	0.85826	0.85666
61	14.298	0.86706	0.86596	0.86461	0.86138	0.86042
62	14.901	0.87052	0.86906	0.86746	0.86464	0.86372
63	15.519	0.87306	0.87215	0.87066	0.86746	0.86682
64	16.152	0.87626	0.87466	0.87378	0.87066	0.86981
65	16.801	0.87866	0.87779	0.87626	0.8736	0.87226
66	17.465	0.88163	0.88026	0.87923	0.87626	0.87538
67	18.144	0.88409	0.88267	0.88179	0.87866	0.87786
68	18.839	0.88646	0.88507	0.88422	0.88107	0.88026
69	19.55	0.88867	0.88747	0.88653	0.88347	0.88267

70	20.277	0.89087	0.88947	0.88867	0.88587	0.88507
71	21.02	0.89267	0.89187	0.89087	0.88787	0.88736
72	21.778	0.89493	0.89347	0.89267	0.89027	0.88945
73	22.553	0.89667	0.89582	0.89487	0.89187	0.89107
74	23.345	0.89827	0.89747	0.89667	0.89422	0.89336
75	24.152	0.90034	0.89907	0.89827	0.89587	0.89507
76	24.977	0.90204	0.90111	0.89987	0.89747	0.89667
77	25.818	0.90361	0.90274	0.90147	0.89907	0.89827
78	26.675	0.90515	0.90387	0.90307	0.90067	0.89987
79	27.55	0.90628	0.90547	0.90467	0.90227	0.90147
80	28.441	0.90788	0.90708	0.90547	0.90387	0.90307
81	29.35	0.90908	0.9086	0.90708	0.90547	0.90467
82	30.276	0.91068	0.90988	0.90825	0.90697	0.90623
83	31.219	0.91195	0.91121	0.90908	0.90828	0.90758
84	32.179	0.91308	0.91228	0.91035	0.90958	0.90887
85	33.157	0.91388	0.91361	0.91148	0.91068	0.90988
86	34.153	0.91543	0.91468	0.91228	0.91198	0.9113
87	35.166	0.91628	0.91548	0.9138	0.91308	0.91228
88	36.198	0.91708	0.91688	0.91468	0.91388	0.91355
89	37.247	0.91851	0.91786	0.91548	0.91527	0.91463
90	38.314	0.91946	0.91868	0.91691	0.91628	0.91548
91	39.399	0.92028	0.91948	0.91788	0.91708	0.91628
92	40.503	0.92108	0.92028	0.91868	0.91788	0.91761
93	41.625	0.92188	0.92108	0.91948	0.91868	0.91851
94	42.765	0.92268	0.92188	0.92028	0.92001	0.9194
95	43.924	0.92348	0.92268	0.92108	0.92085	0.92026
96	45.102	0.92428	0.92348	0.92188	0.92168	0.92108
97	46.298	0.92508	0.92414	0.92268	0.92248	0.92188
98	47.513	0.92588	0.92469	0.92348	0.92325	0.92268
99	48.747	0.92668	0.92508	0.92428	0.92398	0.92347
100	50	0.92737	0.9258	0.92508	0.92428	0.9242





Test FF_VFI_AZ_MZOOM_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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 37.005214 seconds.
---------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                  i     idx     ndim     numel     rowN     colN      sum        mean        std       coefvari     min
```

```
                  i    idx    ndim    numel    rowN    colN      sum       mean       std       coefvari     min     ma
                  _    ___    ____    _____    ____    ____    ______    ______    _______    ________    ___     __

    savefraccoh   1      1       2     6750     750       9     3468.2    0.5138    0.27192     0.52924      0      0.

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

    r1          0          0          0          0          0          0          0    0.02073    0.065955
    r2          0          0          0          0          0          0          0    0.02073    0.065955
    r3          0          0          0          0          0          0          0    0.02073    0.065955
    r4          0          0          0          0          0          0          0    0.02073    0.065955
    r5          0          0          0          0          0          0          0    0.02073    0.065987
    r746   0.8008    0.79843     0.7959    0.79303    0.78983    0.78663    0.78303    0.77903     0.77502
    r747  0.80092    0.79855    0.79603    0.79303    0.79058    0.78713    0.78362    0.77953     0.77553
    r748  0.80102    0.79863    0.79615     0.7935    0.79063    0.78729    0.78378    0.77972     0.77568
    r749  0.80103    0.79863    0.79623    0.79369    0.79063    0.78743    0.78383    0.77983     0.77582
    r750  0.80103    0.79904    0.79623    0.79378    0.79063    0.78743    0.78383    0.77983     0.77582
```

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

```
Elapsed time is 159.606266 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     4667.7    0.6915    0.26685     0.3859       0      0.

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

    r1          0          0          0          0     0.0647    0.16668    0.27352    0.37327     0.4617
    r2          0          0          0          0     0.0647    0.16668    0.27352    0.37327     0.4617
    r3          0          0          0          0   0.064731    0.16668    0.27352    0.37327     0.4617
    r4          0          0          0          0   0.064731    0.16668    0.27355    0.37327     0.4617
    r5          0          0          0          0   0.064747    0.16671    0.27355    0.37327     0.4617
    r746  0.92657    0.92588    0.92508    0.92428    0.92348    0.92268    0.92235    0.92188     0.92188
    r747  0.92664    0.92588    0.92508    0.92428    0.92402    0.92318    0.92248    0.92188     0.92235
    r748  0.92668    0.92588    0.92508    0.92478    0.92411    0.92328     0.9226    0.92188      0.9226
    r749  0.92668    0.92588    0.92555    0.92488     0.9242     0.9234    0.92268    0.92254     0.92268
    r750  0.92668    0.92588    0.92565    0.92497    0.92427    0.92348    0.92268    0.92268     0.92268
```

Test FF_VFI_AZ_MZOOM_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') = 100;
```

```
mp_params('it_z_n') = 7;
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:

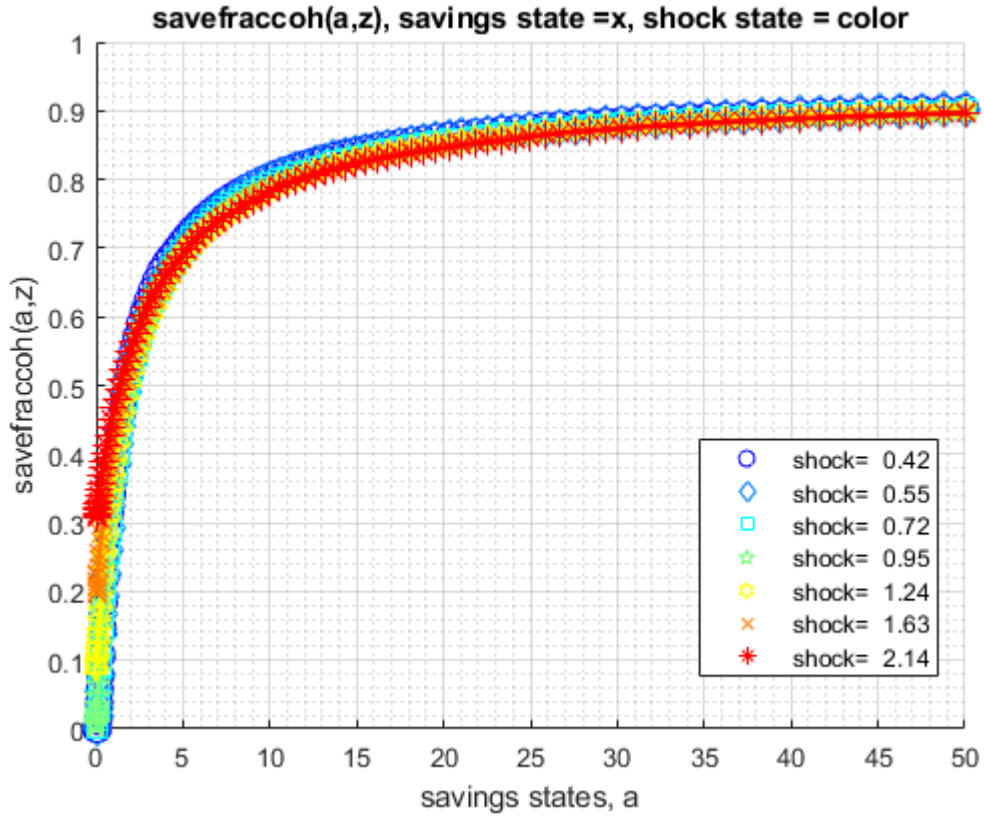
```
% Lower Risk Aversion
mp_params('fl_crra') = 0.5;
ff_vfi_az_mzoom_vec(mp_params, mp_support);
```

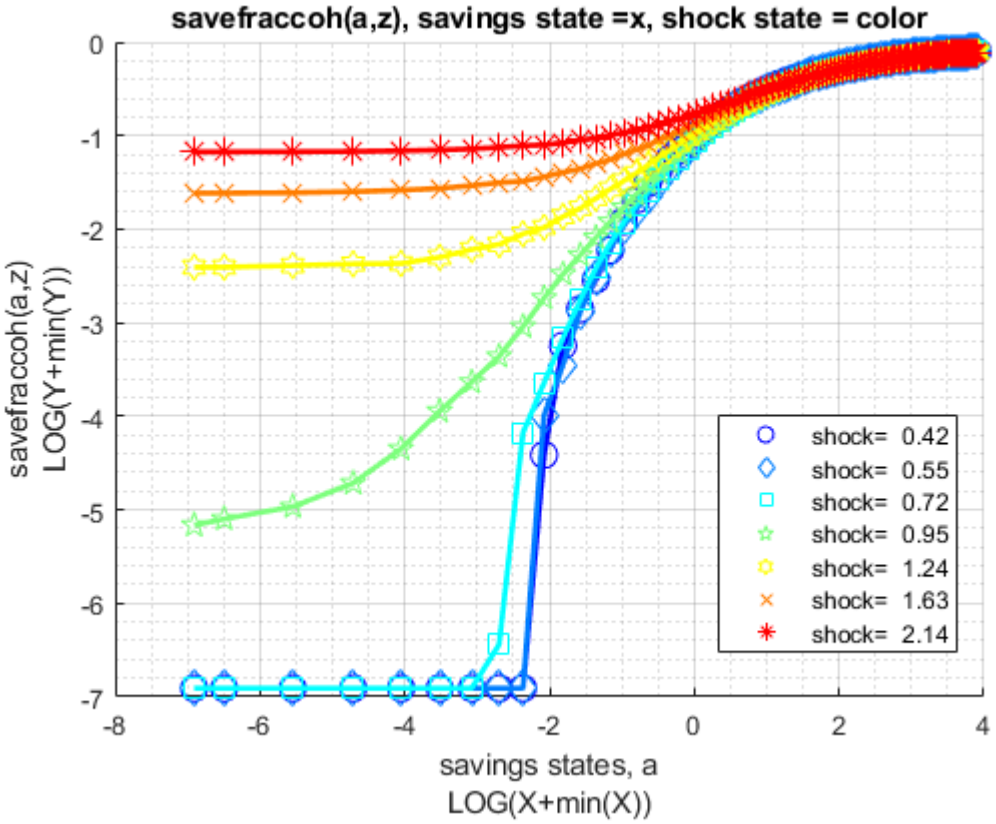
```
Elapsed time is 3.409484 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                  i    idx    ndim    numel    rowN    colN     sum      mean       std      coefvari    min
                  _    ___    ____    _____    ____    ____    ______    _______    _______    ________    ___

    savefraccoh   1     1       2       700     100      7      452.03    0.64575    0.28029    0.43406      0     0

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

    r1            0          0          0    0.0047077    0.089109      0.198    0.30781
    r2            0          0          0    0.0051079    0.089156      0.198    0.30793
    r3            0          0          0    0.0059631    0.090679     0.1988    0.30848
    r4            0          0          0    0.0079639    0.092358    0.20109    0.30964
    r5            0          0          0     0.011926    0.092758    0.20413    0.31171
    r96     0.90047    0.89907    0.89826      0.89727     0.89587    0.89347    0.89267
    r97     0.90127    0.89987    0.89907      0.89822     0.89727    0.89477    0.89394
    r98     0.90204    0.90067    0.89987      0.89907     0.89822    0.89573    0.89493
    r99     0.90278    0.90147    0.90067      0.89987     0.89907    0.89667    0.89587
    r100    0.90354    0.90227    0.90147      0.90067     0.89987    0.89801    0.89667
```





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

```
% Higher Risk Aversion
```

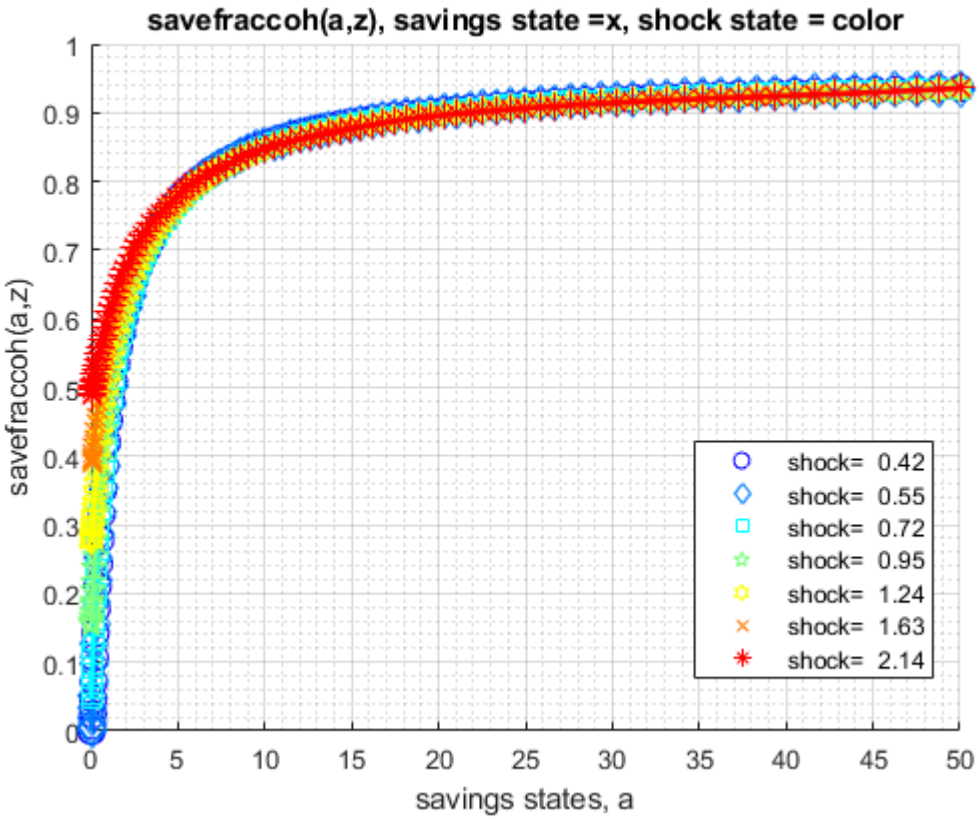
```
mp_params('fl_crra') = 5;
ff_vfi_az_mzoom_vec(mp_params, mp_support);
```

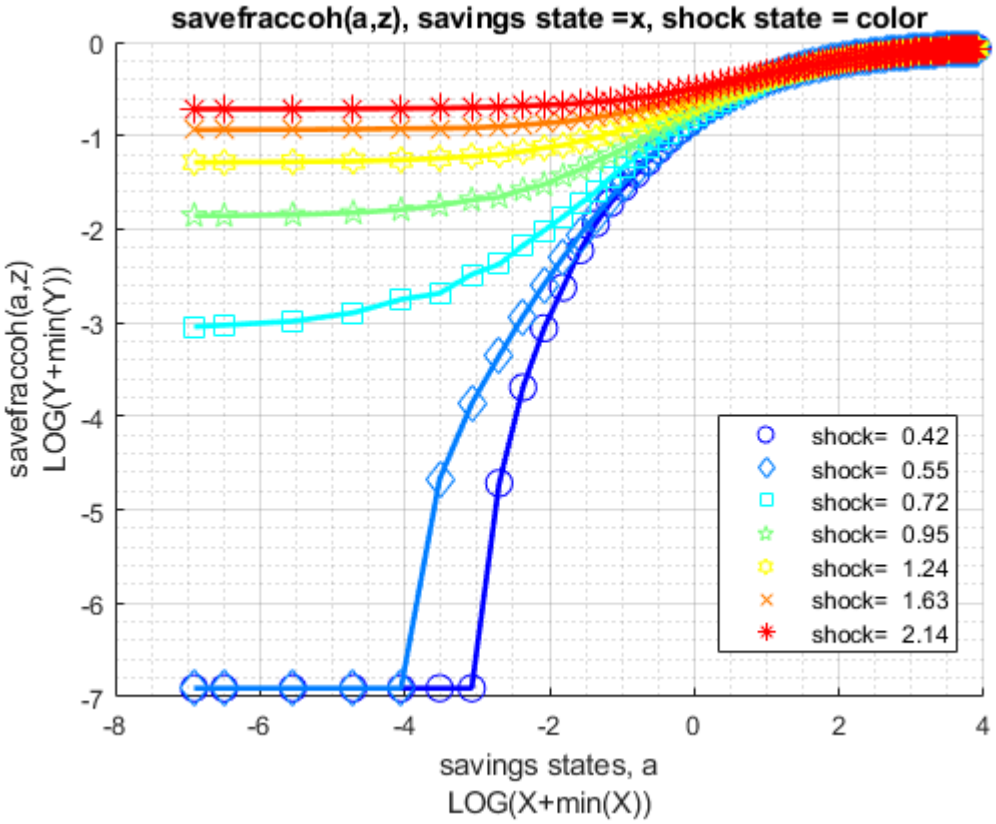
```
Elapsed time is 4.012888 seconds.
--------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	ma
savefraccoh	1	1	2	700	100	7	502.6	0.718	0.25437	0.35427	0	0.93

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5	c6	c7
r1	0	0	0.04674	0.15532	0.27563	0.39047	0.48771
r2	0	0	0.047493	0.15525	0.27563	0.39101	0.48771
r3	0	0	0.049541	0.15685	0.27693	0.39127	0.48834
r4	0	0	0.054343	0.16018	0.27883	0.39287	0.48923
r5	0	0	0.062848	0.16566	0.28272	0.39528	0.49071
r96	0.93269	0.93251	0.93189	0.93108	0.93014	0.92988	0.92968
r97	0.93349	0.93322	0.93269	0.93189	0.93107	0.93104	0.93108
r98	0.93429	0.93349	0.93347	0.93269	0.93189	0.93189	0.93269
r99	0.93507	0.93429	0.93424	0.93349	0.93331	0.93349	0.93429
r100	0.93575	0.93509	0.93507	0.93488	0.93491	0.93509	0.93587





Test FF_VFI_AZ_MZOOM_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') = 150;
mp_params('it_z_n') = 15;
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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 16.599473 seconds.
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                  i     idx     ndim     numel     rowN     colN      sum        mean         std       coefvari     min
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	
savefraccoh	1	1	2	2250	150	15	1507.2	0.66985	0.28667	0.42796	0	0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5	c11	c12	c13	c14	c15
r1	0	0	0	0	0	0.13838	0.18479	0.23021	0.27363	0.317
r2	0	0	0	0	0	0.13838	0.18479	0.23027	0.27363	0.317
r3	0	0	0	0	0	0.13894	0.18526	0.23041	0.27407	0.317
r4	0	0	0	0	0	0.13987	0.18606	0.23121	0.27443	0.318
r5	0	0	0	0	0	0.13998	0.18719	0.23201	0.27563	0.318
r146	0.92348	0.92348	0.92328	0.92268	0.92268	0.92085	0.92028	0.92028	0.91948	0.919
r147	0.9242	0.92398	0.92348	0.92348	0.92337	0.92108	0.92108	0.92097	0.92001	0.919
r148	0.92428	0.92428	0.92428	0.92408	0.92348	0.92188	0.92171	0.92108	0.92028	0.920
r149	0.92508	0.92497	0.92478	0.92428	0.92428	0.92241	0.92188	0.92188	0.92108	0.921
r150	0.92565	0.92508	0.92508	0.92507	0.92485	0.92268	0.92268	0.92254	0.92238	0.921

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_mzoom_vec(mp_params, mp_support);
```

```
Elapsed time is 16.323916 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	2250	150	15	1685.2	0.74898	0.22908	0.30585	0	0

xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5	c11	c12	c13	c14	c15
r1	0	0	0	0	0	0.52613	0.61256	0.68259	0.73901	0.784
r2	0	0	0	0	0	0.52613	0.61256	0.68259	0.73901	0.78
r3	0	0	0	0	0	0.52613	0.61256	0.68259	0.73901	0.78
r4	0	0	0	0	0	0.52682	0.61256	0.68259	0.73901	0.784
r5	0	0	0	0	0	0.52693	0.61309	0.68259	0.73901	0.784
r146	0.92948	0.92925	0.92828	0.92805	0.92737	0.92263	0.92348	0.92577	0.92901	0.933
r147	0.93017	0.92948	0.92868	0.92828	0.92748	0.92348	0.92428	0.92668	0.93002	0.934
r148	0.93028	0.93005	0.92948	0.92891	0.92827	0.92428	0.92587	0.92799	0.93101	0.935
r149	0.93091	0.93028	0.92948	0.92931	0.92828	0.92574	0.92668	0.92904	0.93189	0.935
r150	0.93108	0.93082	0.93027	0.92948	0.92868	0.92668	0.92814	0.93008	0.93269	0.936