

# FF\_VFI\_AZ\_BISEC\_LOOP Dynamic Savings Problem Loop Continuous Choice

back to [Fan's Intro Math for Econ](#), [Matlab Examples](#), or [Dynamic Asset Repositories](#)

This is the example vignette for function: [ff\\_vfi\\_az\\_bisec\\_loop](#) 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 the looped code, it is slow for larger state-space problems. The code uses continuous choices, solved with bisection. The state-space is on a grid, but choice grids are in terms of percentage of resources to save and solved exactly.

## 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\_LOOP 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_loop(mp_params);
```

Elapsed time is 13.575906 seconds.

```
-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	max
ap	1	1	2	700	100	7	15835	22.621	13.367	0.59091	0	47.273

```
xxx TABLE:ap XXXXXXXXXXXXXXXXXXXXXXX
```

	c1	c2	c3	c4	c5	c6	c7
r1	0	0	0	0	0	0.38021	1.4609
r2	0.19477	0.18872	0.19731	0.24709	0.41492	0.79311	1.8893
r3	0.54595	0.54109	0.55664	0.62239	0.81173	1.2132	2.3195
r4	1.0101	1.0101	1.0101	1.0189	1.2217	1.6363	2.7464

r5	1.4388	1.4362	1.459	1.5151	1.6354	2.0602	3.1804
r96	43.225	43.246	43.3	43.422	43.632	44.155	45.413
r97	43.69	43.71	43.765	43.887	44.096	44.618	45.879
r98	44.154	44.174	44.228	44.352	44.559	45.083	46.344
r99	44.618	44.638	44.693	44.815	45.024	45.548	46.809
r100	45.08	45.101	45.156	45.28	45.487	46.012	47.273

## Test FF\_VFI\_AZ\_BISEC\_LOOP Speed Tests

Call the function with different a and z grid size, print out speed:

```
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_loop(mp_params, mp_support);
```

Elapsed time is 4.733351 seconds.

A grid 100, shock grid 7:

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 100;
mp_params('it_z_n') = 7;
ff_vfi_az_bisec_loop(mp_params, mp_support);
```

Elapsed time is 13.889250 seconds.

A grid 200, shock grid 9:

```
mp_params = containers.Map('KeyType','char', 'ValueType','any');
mp_params('it_a_n') = 200;
mp_params('it_z_n') = 9;
ff_vfi_az_bisec_loop(mp_params, mp_support);
```

Elapsed time is 38.195963 seconds.

## Test FF\_VFI\_AZ\_BISEC\_LOOP 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.

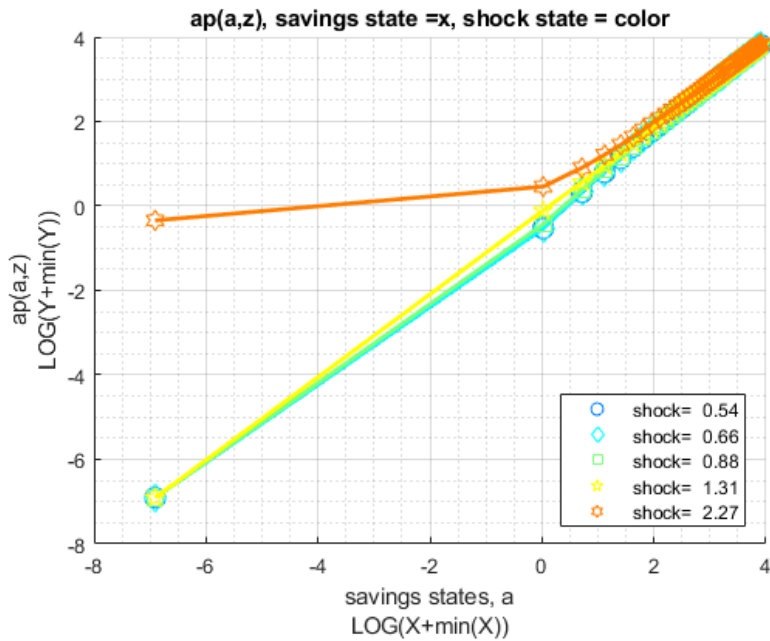
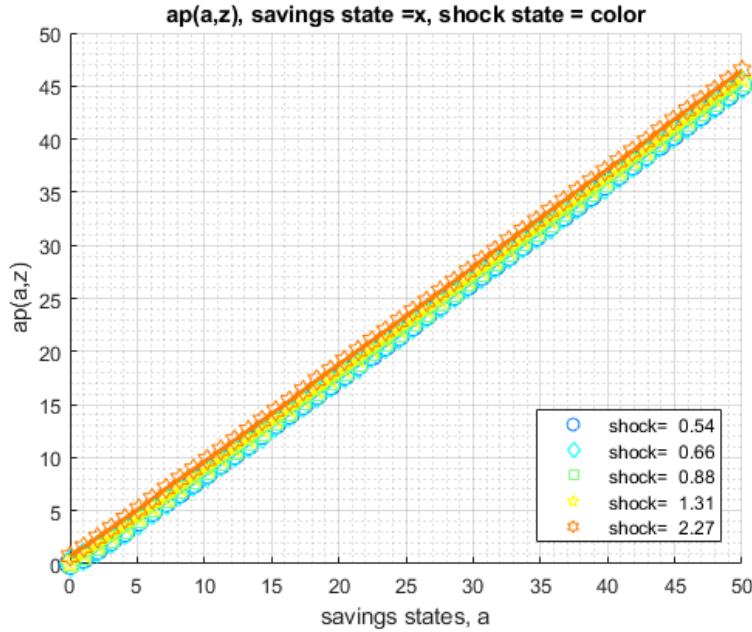
```
mp_support = containers.Map('KeyType','char', 'ValueType','any');
mp_support('bl_print_params') = false;
mp_support('bl_print_iterinfo') = false;
% ls_ffcmd: summary print which outcomes
mp_support('ls_ffcmd') = {};
% ls_ffsna: detail print which outcomes
mp_support('ls_ffsna') = {'ap'};
% ls_ffgrh: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

Elapsed time is 4.728102 seconds.

xxx ff\_vfi\_az\_vec, outcome=ap xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

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	0.70914
2	1.0204	0.58666	0.5889	0.64782	0.89696	1.5821
3	2.0408	1.3824	1.3926	1.4723	1.7501	2.4674
4	3.0612	2.2189	2.2357	2.3281	2.6228	3.3608
5	4.0816	3.0774	3.0995	3.2007	3.5069	4.2595
6	5.102	4.0815	4.0816	4.0833	4.3986	5.1595
7	6.1224	4.9896	5.0184	5.1021	5.2928	6.1223
8	7.1429	5.8564	5.8902	6.0116	6.1873	7.0951
9	8.1633	6.7406	6.7749	6.8956	7.1428	7.9989
10	9.1837	7.6346	7.6706	7.7922	8.1324	8.9028
11	10.204	8.5353	8.572	8.6944	9.0301	9.8113
12	11.224	9.4411	9.4787	9.6014	9.9343	10.726
13	12.245	10.35	10.389	10.512	10.845	11.644
14	13.265	11.263	11.302	11.427	11.761	12.565
15	14.286	12.245	12.245	12.344	12.68	13.488
16	15.306	13.224	13.264	13.265	13.6	14.413
17	16.327	14.141	14.182	14.286	14.523	15.339
18	17.347	15.052	15.097	15.228	15.446	16.327
19	18.367	15.971	16.014	16.147	16.37	17.311
20	19.388	16.889	16.933	17.065	17.347	18.233
21	20.408	17.811	17.856	17.989	18.34	19.157
22	21.429	18.735	18.779	18.912	19.265	20.083
23	22.449	19.66	19.705	19.838	20.187	21.01
24	23.469	20.586	20.632	20.765	21.113	21.938
25	24.49	21.513	21.559	21.693	22.041	22.869
26	25.51	22.449	22.488	22.622	22.97	23.8
27	26.531	23.469	23.47	23.551	23.9	24.732
28	27.551	24.418	24.464	24.49	24.831	25.665
29	28.571	25.348	25.394	25.51	25.763	26.599
30	29.592	26.276	26.325	26.46	26.696	27.551
31	30.612	27.203	27.252	27.392	27.629	28.572
32	31.633	28.135	28.182	28.321	28.571	29.514
33	32.653	29.067	29.115	29.251	29.592	30.446
34	33.673	29.998	30.047	30.185	30.542	31.379
35	34.694	30.931	30.979	31.118	31.476	32.312
36	35.714	31.866	31.913	32.05	32.407	33.246
37	36.735	32.799	32.848	32.985	33.34	34.182
38	37.755	33.733	33.782	33.921	34.276	35.118
39	38.776	34.694	34.718	34.856	35.211	36.054
40	39.796	35.714	35.714	35.792	36.145	36.99

41	40.816	36.651	36.7	36.735	37.082	37.929
42	41.837	37.587	37.636	37.755	38.02	38.865
43	42.857	38.523	38.573	38.712	38.957	39.805
44	43.878	39.457	39.508	39.648	39.894	40.816
45	44.898	40.391	40.441	40.585	40.832	41.788
46	45.918	41.328	41.379	41.519	41.836	42.727
47	46.939	42.266	42.316	42.455	42.816	43.665
48	47.959	43.204	43.254	43.393	43.754	44.601
49	48.98	44.14	44.191	44.332	44.693	45.539
50	50	45.077	45.128	45.269	45.631	46.478



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_loop(mp_params, mp_support);

```

Elapsed time is 18.110812 seconds.

XX

CONTAINER NAME: mp\_ffcmd ND Array (Matrix etc)

XX

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min
	—	—	—	—	—	—	—	—	—	—	—
ap	1	1	2	900	100	9	20493	22.77	13.386	0.5879	0
savefraccoh	2	2	2	900	100	9	701.94	0.77994	0.13136	0.16842	0

xxx TABLE:ap XXXXXXXXXXXXXXXXXXXXXXX

	c1	c2	c3	c4	c5	c6	c7	c8	c9
	—	—	—	—	—	—	—	—	—
r1	0	0	0	0	0	0	0.20716	0.89208	2.4759
r2	0.19971	0.19144	0.18896	0.2007	0.24755	0.38215	0.61592	1.3126	2.9098
r3	0.55145	0.54262	0.54255	0.5618	0.62321	0.77699	1.0303	1.7326	3.3477
r4	1.0101	1.0101	1.0101	1.0101	1.0198	1.1844	1.5151	2.1613	3.7899
r5	1.4445	1.436	1.4393	1.4657	1.5152	1.5944	1.9615	2.5895	4.2337
r96	43.226	43.233	43.257	43.313	43.424	43.584	43.951	44.764	46.479
r97	43.69	43.697	43.722	43.776	43.888	44.048	44.444	45.227	46.97
r98	44.155	44.161	44.186	44.241	44.352	44.512	44.933	45.692	47.461
r99	44.619	44.626	44.65	44.707	44.817	44.976	45.398	46.156	47.927
r100	45.081	45.088	45.114	45.169	45.28	45.454	45.861	46.621	48.391

xxx TABLE:savefraccoh XXXXXXXXXXXXXXXXXXXXXXX

	c1	c2	c3	c4	c5	c6	c7	c8	c9
	—	—	—	—	—	—	—	—	—
r1	0	0	0	0	0	0	0.10085	0.28368	0.45924
r2	0.17696	0.16018	0.14648	0.1404	0.15072	0.19253	0.23949	0.35842	0.49245
r3	0.33498	0.31679	0.30013	0.28853	0.2885	0.31047	0.33348	0.41451	0.52092
r4	0.46678	0.45284	0.43437	0.40981	0.38082	0.39214	0.42003	0.46007	0.54576
r5	0.53868	0.52254	0.50624	0.49144	0.47417	0.45067	0.47554	0.49651	0.56737
r96	0.86817	0.86713	0.86597	0.86469	0.86323	0.86054	0.85786	0.85551	0.85172
r97	0.86845	0.86744	0.86631	0.865	0.86356	0.86091	0.8588	0.8559	0.85264
r98	0.86875	0.86774	0.86662	0.86533	0.8639	0.86128	0.85966	0.8563	0.85352
r99	0.86903	0.86805	0.86692	0.86567	0.86424	0.86161	0.86002	0.8567	0.85395
r100	0.86927	0.86829	0.8672	0.86594	0.86454	0.86222	0.86036	0.85709	0.85435

## Test FF\_VFI\_AZ\_BISEC\_LOOP 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') = 50;
mp_params('it_z_n') = 5;
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_loop(mp_params, mp_support);
```

Elapsed time is 1.421696 seconds.

```
-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	max
savefraccoh	1	1	2	250	50	5	94.272	0.37709	0.25552	0.67761	0	0

```
xxx TABLE:savefraccoh XXXXXXXXXXXXXXXXXXXXX
```

	c1	c2	c3	c4	c5
r1	0	0	0	0	0.00014733
r2	0	0	0	0	0.00014733
r3	0	0	0	0	0.0011239
r4	0	0	0	0	0.0011544
r5	0	0	0	0	0.0039009
r46	0.67805	0.67137	0.66298	0.6526	0.64094
r47	0.67964	0.67329	0.66536	0.6555	0.64439
r48	0.6811	0.67506	0.66752	0.65818	0.6476
r49	0.68242	0.67671	0.6696	0.66075	0.65059
r50	0.68364	0.67826	0.67158	0.66319	0.65336

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

Elapsed time is 5.579251 seconds.

```
-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CONTAINER NAME: mp_ffcmd ND Array (Matrix etc)
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	max
savefraccoh	1	1	2	250	50	5	146.44	0.58575	0.29994	0.51206	0	0

```
xxx TABLE:savefraccoh XXXXXXXXXXXXXXXXXXXXX
```

	c1	c2	c3	c4	c5
r1	0	0	0	0.086968	0.3134
r2	0	0	0	0.086938	0.3135
r3	0	0	0	0.086877	0.31423
r4	0	0	0	0.091393	0.31621
r5	0	0	0.00036095	0.10012	0.31765
r46	0.87894	0.8773	0.87437	0.86796	0.86643
r47	0.88136	0.8798	0.87717	0.87083	0.86933
r48	0.88358	0.88215	0.87983	0.87348	0.87202
r49	0.88566	0.88432	0.8823	0.87595	0.87455
r50	0.88761	0.88633	0.88465	0.87827	0.87687

## Test FF\_VFI\_AZ\_BISEC\_LOOP 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') = 5;
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_loop(mp_params, mp_support);
```

Elapsed time is 9.991698 seconds.

-----

XX

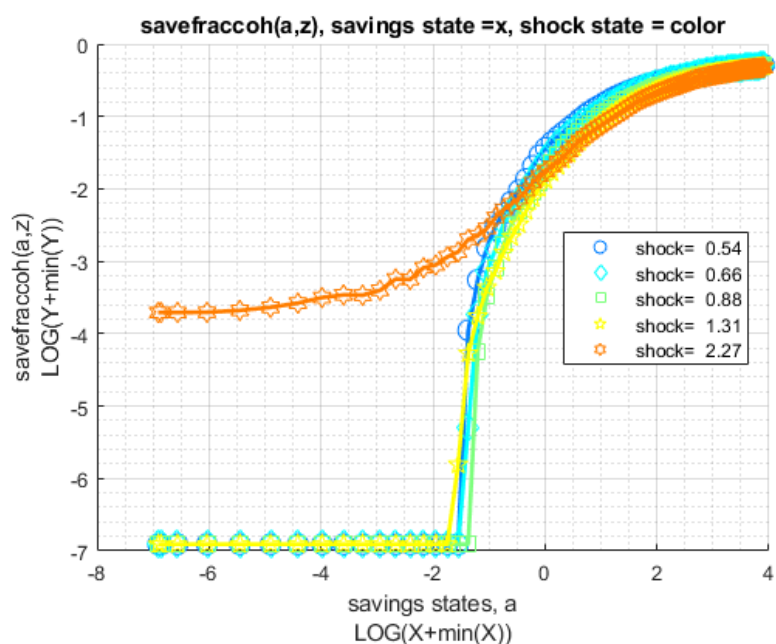
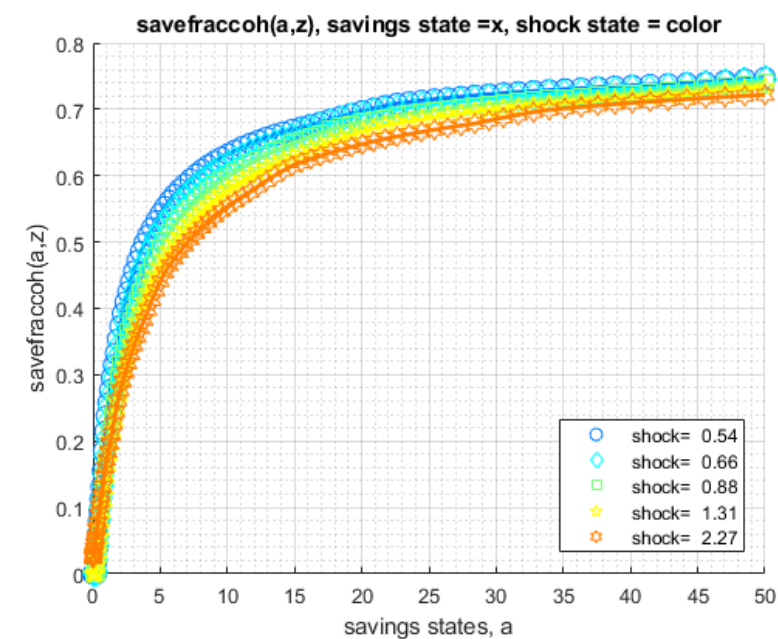
CONTAINER NAME: mp\_ffcmd ND Array (Matrix etc)

XX

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	
	—	—	—	—	—	—	—	—	—	—	—	—
savefraccoh	1	1	2	500	100	5	214.05	0.42811	0.27486	0.64204	0	0

xxx TABLE:savefraccoh XXXXXXXXXXXXXXXXXXXXXXX

	c1	c2	c3	c4	c5
	—	—	—	—	—
r1	0	0	0	0	0.023554
r2	0	0	0	0	0.023554
r3	0	0	0	0	0.023554
r4	0	0	0	0	0.023615
r5	0	0	0	0	0.024256
r96	0.7393	0.73551	0.73109	0.72261	0.71525
r97	0.74049	0.73805	0.73374	0.72544	0.71702
r98	0.7429	0.74052	0.73634	0.72825	0.7187
r99	0.74525	0.74296	0.73887	0.731	0.72032
r100	0.74757	0.74534	0.74113	0.73371	0.72187



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_loop(mp_params, mp_support);
```

Elapsed time is 8.955693 seconds.

XX

CONTAINER NAME: mp\_ffcmd ND Array (Matrix etc)

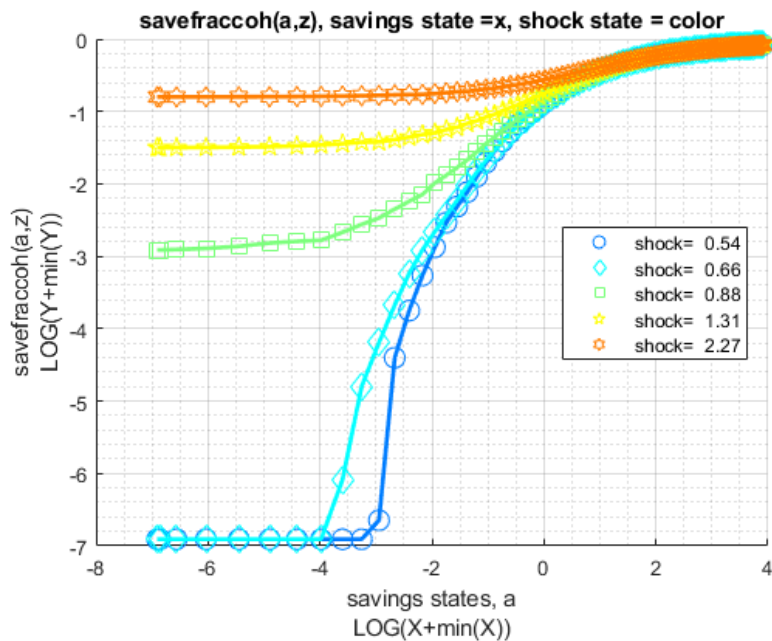
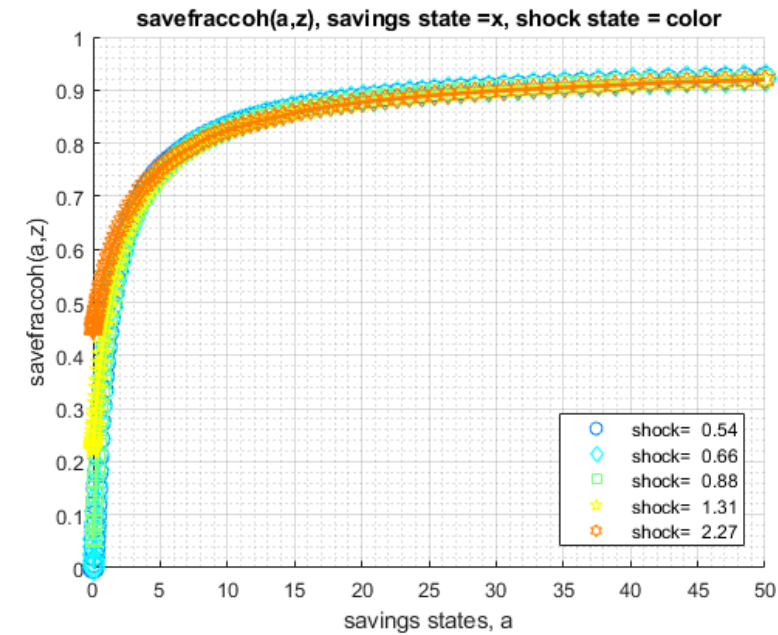
XX

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	max
savefraccoh	1	1	2	500	100	5	323.21	0.64642	0.28954	0.44792	0	0.75



xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5
r1	0	0	0.053308	0.22219	0.44872
r2	0	0	0.053338	0.22222	0.44875
r3	0	0	0.053644	0.2224	0.44881
r4	0	0	0.054498	0.22286	0.44899
r5	0	0	0.056115	0.22378	0.44936
r96	0.91981	0.91901	0.91813	0.91575	0.91523
r97	0.92081	0.92002	0.91914	0.91682	0.9163
r98	0.92176	0.921	0.92014	0.91785	0.9173
r99	0.92268	0.92194	0.92112	0.91883	0.91831
r100	0.92356	0.92286	0.92203	0.91981	0.91926



Test FF\_VFI\_AZ\_BISEC\_LOOP 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') = 100;
mp_params('it_z_n') = 5;
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_loop(mp_params, mp_support);
```

Elapsed time is 10.016136 seconds.

XX

CONTAINER NAME: mp\_ffcmd ND Array (Matrix etc)

XX

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	
	—	—	—	—	—	—	—	—	—	—	—	—
savefraccoh	1	1	2	500	100	5	266.09	0.53217	0.31606	0.59389	0	0

xxx TABLE:savefraccoh XXXXXXXXXXXXXXXXXXXXXXX

	c1	c2	c3	c4	c5
	—	—	—	—	—
r1	0	0	0	0	0.027887
r2	0	0	0	0	0.027887
r3	0	0	0	0	0.027887
r4	0	0	0	0	0.027857
r5	0	0	0	0	0.027826
r96	0.85941	0.85841	0.85734	0.85621	0.85218
r97	0.86076	0.85978	0.85871	0.85764	0.85368
r98	0.86204	0.86109	0.86008	0.85902	0.85511
r99	0.86329	0.86234	0.86137	0.86036	0.85688
r100	0.86448	0.86359	0.86262	0.86164	0.85862

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_loop(mp_params, mp_support);
```

Elapsed time is 10.186494 seconds.

XX

CONTAINER NAME: mp\_ffcmd ND Array (Matrix etc)

XX

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari	min	
	—	—	—	—	—	—	—	—	—	—	—	—
savefraccoh	1	1	2	500	100	5	324.66	0.64932	0.26596	0.40959	0	0
xxx TABLE:savefraccoh xxxxxxxxxxxxxxxxxxxxx												
	c1	c2	c3	c4	c5							
	—	—	—	—	—							
r1	0	0	0	0.26564	0.67546							
r2	0	0	0	0.26568	0.67549							
r3	0	0	0	0.26586	0.67549							
r4	0	0	0	0.26635	0.67552							
r5	0	0	0	0.26732	0.67558							
r96	0.88071	0.87922	0.87495	0.86723	0.85859							
r97	0.88178	0.88032	0.8762	0.86826	0.86085							
r98	0.88282	0.88139	0.87739	0.86927	0.86317							
r99	0.88383	0.88212	0.87855	0.87028	0.86668							
r100	0.88483	0.88279	0.87937	0.87128	0.87003							