

*Comparing numerical solutions of
models with heterogeneous agents
(Models B): a grid-based Euler
equation algorithm*

**Lilia Maliar, Serguei Maliar
and Fernando Valli**

University of Alicante

Our algorithm is relatively stylized

- At the individual level: Euler equation method on a grid of prespecified points.
- At the aggregate level: Monte Carlo simulation and updating the ALM, as in Krusell and Smith (1998).
- Also, we propose a simple bisection and updating method for computing the ALM that in some applications can work faster than the pure updating, as in Krusell and Smith (1998).

Individual problem

- The Euler equation,

$$\begin{aligned} & C(k, \varepsilon, m, a)^{-\gamma} - h(k, \varepsilon, m, a) \\ &= \beta E \left\{ C(k', \varepsilon', m', a')^{-\gamma} \left[1 - \delta + \alpha a' \left(\frac{K'}{\bar{l}L'} \right)^{\alpha-1} \right] \right\}, \end{aligned}$$

where $C(k, \varepsilon, m, a)$ is the consumption function;

$h(k, \varepsilon, m, a)$ is the Lagrange multiplier;

m is a set of aggregate statistics.

- Kuhn-Tucker conditions are

$$h(k, \varepsilon, m, a) \geq 0,$$

$$A(k, \varepsilon, m, a) \geq 0, \quad h(k, \varepsilon, m, a) A(k, \varepsilon, m, a) = 0,$$

where $A(k, \varepsilon, m, a)$ is the asset function,

$$A(k, \varepsilon, m, a) \equiv k' = rk + w\varepsilon + (1 - \delta)k - C(k, \varepsilon, m, a).$$

$$A(s) \equiv rk + w\varepsilon + (1 - \delta)k - \left[h(s) + \beta E \left\{ \frac{1 - \delta + \alpha a' \left(\frac{K'}{iL'} \right)^{\alpha-1}}{[rk + w\varepsilon + (1 - \delta)A(s) - A(A(s))]^\gamma} \right\} \right]^{-1/\gamma},$$

where $(s) \equiv (k, \varepsilon, m, a)$ is the state space.

- *Step 1.* Choose a grid for the individual assets on the interval $[k_{\min}, k_{\max}]$. Fix initial asset function, $A(k, \varepsilon, m, a)$.
- *Step 2.* Set the Lagrange multiplier to zero $h(k, \varepsilon, m, a) = 0$. Substitute $A(k, \varepsilon, m, a)$ in the right side and compute $\tilde{A}(k, \varepsilon, m, a)$. For each grid point, in which $\tilde{A}(k, \varepsilon, m, a)$ does not belong to $[k_{\min}, k_{\max}]$, set $\tilde{A}(k, \varepsilon, m, a)$ at the corresponding boundary value.

- *Step 3.* Compute the asset function for next iteration by using updating:

$$\tilde{A}(k, \varepsilon, m, a) = \eta_A \tilde{A}(k, \varepsilon, m, a) + (1 - \eta_A) A(k, \varepsilon, m, a),$$

where $\eta_A \in (0, 1]$.

Iterate on *Steps 2 – 3* until $\tilde{A}(k, \varepsilon, m, a) = A(k, \varepsilon, m, a)$ with a given precision, 10^{-10} , according to the least-square norm.

By construction, the function $A(k, \varepsilon, m, a)$ satisfies:

- the Euler equation;
- the Kuhn-Tucker conditions

$$A(k, \varepsilon, m, a) \geq 0, \quad h(k, \varepsilon, m, a) A(k, \varepsilon, m, a) = 0;$$

- budget constraint.

We are left to check that the Lagrange multiplier is non-negative, i.e.,

$$h(k, \varepsilon, m, a) \geq 0,$$

Notice that under $\gamma > 0$, the term $[h(s) + \dots]^{-1/\gamma}$ in equation

$$A(s) \equiv rk + w\varepsilon + (1 - \delta)k - \left[h(s) + \beta E \left\{ \frac{1 - \delta + \alpha a' \left(\frac{K'}{L'} \right)^{\alpha-1}}{[rk + w\varepsilon + (1 - \delta)A(s) - A(A(s))]^\gamma} \right\} \right]^{-1/\gamma},$$

is decreasing in $h(s)$. Recall that, when the unconstrained solution (under $h(s) = 0$) violated the borrowing constraint, we set the asset holdings at the borrowing limit, i.e., we increase $A(s)$. To preserve the equality sign, we should decrease consumption $[h(s) + \dots]^{-1/\gamma}$, which requires to increase the multiplier $h(s)$.

The ALM: Krusell and Smith (1998)

Let $m = \{m_1, \dots, m_M\}$ be statistics characterizing the aggregate state and let the ALM be

$$m' = f(b, m),$$

where $b = (b_1, \dots, b_S)$ is a vector of coefficients.

We solve for the ALM by using updating:

- *Step I.* Fix some initial vector b . Generate and fix time series of length T for the aggregate shocks and idiosyncratic shocks. Fix the initial wealth distribution.
- *Step II.* Given b and ALM, compute a solution to the individual problem.

- *Step III.* Use ALM and the individual decision rules computed in *Step II* to simulate the economy T periods forward by explicitly solving for the asset holdings of each agent and by calculating the set of statistics m for each $t = 1, \dots, T$.
- *Step IV.* Use the time series for m calculated in *Step III* to re-estimate the ALM coefficients.
- *Step V.* Call the new ALM coefficients by \tilde{b} .
- *Step VI.* Compute the ALM for next iteration by using updating: $\tilde{b} = \eta_b \tilde{b} + (1 - \eta_b) b$, $\eta_b \in (0, 1]$.

Iterate on *Steps II – VI* until $\tilde{b} = b$.

Some computation issues

- Initial distribution of wealth matters. We solve the model twice: we first assume that all agents have initial wealth equal to the deterministic steady state, and we then use a distribution of wealth generated by the model as the initial one.
- To solve the individual problem, we are to choose the domain for assets $[k_{\min}, k_{\max}]$. By definition, $k_{\min} = 0$, and k_{\max} is to be sufficiently large not to be reached along simulations. We take $k_{\max} = 1000$; in our simulations, the assets never reached 500.
- To evaluate the asset function outside the grid, we tried both linear and cubic (spline) polynomial interpolation. We find that the cubic interpolation with few points is superior to the linear one.

Individual problem

- How many grid points to take and where to place them in the domain?
- The individual decision rules are non-linear near the liquidity constraints but are close to linear at large levels of capital.
- We need many grid points at low levels of capital and few grid points at high levels of capital. We use the following polynomial rule for placement of grid points

$$y(x) = \frac{x^m}{(0.5)^m} \text{ for } x \in [0, 0.5],$$

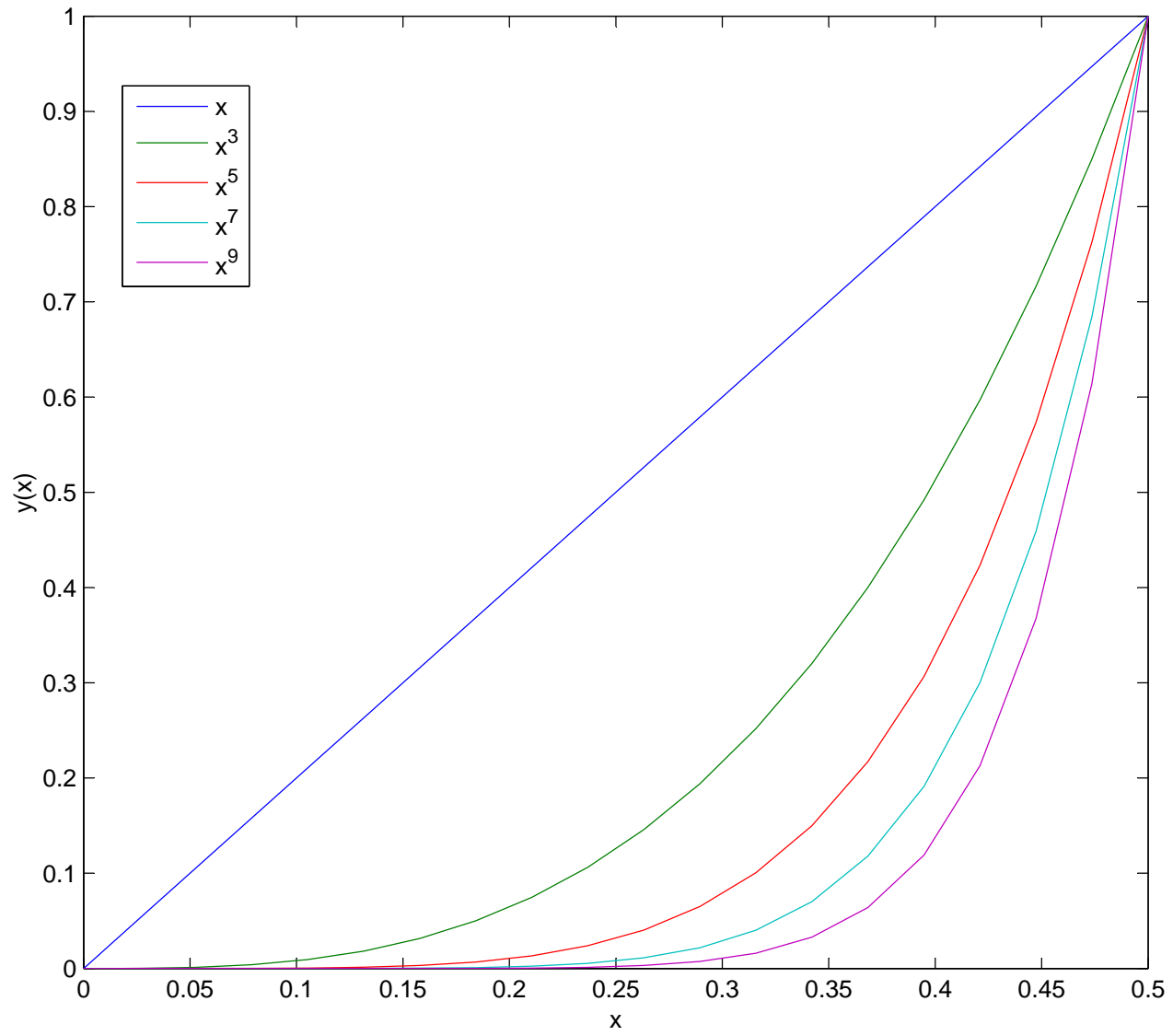
where m is a degree of the polynomial. Note that $y(0) = 0$ and $y(1) = 1$.

- By increasing m , we can increase (decrease) concentration of grid points in the beginning (the end) of the interval

TABLE 1
Grid-point placement depending on the polynomial degree

x	x^3	x^5	x^7	x^9
0	0	0	0	0
0,052632	0,00014579	4,0386e-007	1,1187e-009	3,099e-012
0,10526	0,0011664	1,2924e-005	1,432e-007	1,5867e-009
0,15789	0,0039364	9,8138e-005	2,4467e-006	6,0997e-008
0,21053	0,0093308	0,00041355	1,8329e-005	8,1238e-007
0,26316	0,018224	0,0012621	8,7401e-005	6,0527e-006
0,31579	0,031491	0,0031404	0,00031317	3,123e-005
0,36842	0,050007	0,0067877	0,00092132	0,00012505
0,42105	0,074646	0,013234	0,0023461	0,00041594
0,47368	0,10628	0,023848	0,0053508	0,0012006
0,52632	0,14579	0,040386	0,011187	0,003099
0,57895	0,19405	0,065042	0,021801	0,0073072
0,63158	0,25193	0,10049	0,040086	0,01599
0,68421	0,32031	0,14995	0,070199	0,032863
0,73684	0,40006	0,21721	0,11793	0,064028
0,78947	0,49205	0,30668	0,19115	0,11913
0,84211	0,59717	0,42348	0,30031	0,21296
0,89474	0,71629	0,57342	0,45906	0,3675
0,94737	0,85027	0,76312	0,68491	0,61471
1	1	1	1	1

Figure 1. Power grid functions.



- To analyze which m leads to the most accurate solution.
- we first compute an "accurate" solution by considering 100000 grid points uniformly placed in the interval $[k_{\min}, k_{\max}]$.
- we then compute the decision rules by considering 100 points placed under different degrees m , and we look at errors.
- the smallest maximum percentage error is achieved under the degree $m = 7$.
- we view the solution with 100-points as relatively accurate: the average error is about 0.0002% and the maximum error is about 0.09%.
- we also look at how the accuracy depends on the number of grid points and the level of wealth.

TABLE 2
Decision rule errors depending on polynomial degree and number of grid points

	Average error compared to the accurate solution	Maximum error compared to the accurate solution	Wealth under maximum error
100 grid point			
Polynomial degree			
1	0,0023444	0,34711	1.6
2	6,382e-006	0,010158	0.6
3	2,8005e-006	0,0037153	1.6
4	1,7269e-006	0,0019896	1.6
5	1,6978e-006	0,0018806	0.6
6	1,7102e-006	0,0016388	1.6
7	1,9658e-006	0,00086033	0.6
8	2,2776e-006	0,0012321	0.6
9	2,5752e-006	0,0011811	0.9
Polynomial of degree 7			
Number of grid points			
100	1,9658e-006	0,00086033	0.6
200	5,5139e-007	0,00078441	1.0
300	2,0893e-007	0,00021801	1.0
400	1,0151e-007	0,00015021	2.5

Model without aggregate uncertainty

- We solve for an interest rate in a model without aggregate uncertainty by simulation.
- We assume $N = 10000$ heterogeneous agents, we fix the length of time series $T = 1100$ and we discard the first $T_1 = 100$ periods in order to eliminate the effect of initial conditions.
- We again compare all solutions to an "accurate" solution by taking a grid of 5000 points distributed according to 7 degree polynomial.

TABLE 3
Errors in Model B II depending on the number of grid points and specific random draw

Number of the grid points	Average error compared to the accurate solution	Maximum error compared to the accurate solution	Percentage difference in capital stocks
100	4,2835e-005	0,0024234	1,4717e-009
200	1,6516e-005	0,00090135	1,4389e-009
300	1,039e-005	0,00044333	1,4359e-009
400	7,1918e-005	0,00039331	1,4717e-009

Random draw for shocks	Average difference compared to the benchmark solution	Maximum difference compared to the benchmark solution	Percentage difference in capital stocks compared to the benchmark solution
1	1,318e-011	3,4227e-011	1,0926e-011
2	9,7232e-011	1,9258e-010	5,0491e-011
3	2,0634e-010	4,0765e-010	1,0662e-010

- Our 100-point solution is still sufficiently accurate: the average error is about 0.005% and the maximum error is about 0.2%. The difference in the steady state levels of capital across the experiments is very small, $10^{-7}\%$.
- Does specific realization of shocks matter? No. The obtained differences in the decision rules and the steady state capital stock across four realizations are very small, $10^{-8}\%$.

Model with aggregate uncertainty

- We solved the models by characterizing the aggregate state space with the mean of the wealth distribution, and also, with both the mean and the variance of the wealth distribution.
- For the case of one moment (mean), the ALM are given by

$$\ln(K_{t+1}) = 0.1235 + 0.9657 \ln(K_t) \quad R^2 = 0.999934$$

with the low aggregate shock, and by

$$\ln(K_{t+1}) = 0.1385 + 0.9631 \ln(K_t) \quad R^2 = 0.999967$$

with the high aggregate shock.

- For the case of two moments (mean and variance), the ALM for aggregate (mean) capital are given by

$$\ln(K_{t+1}) = 0.1224 + 0.9661 \ln(K_t) \quad R^2 = 0.999935$$

with the low aggregate shock, and by

$$\ln(K_{t+1}) = 0.1398 + 0.9628 \ln(K_t) \quad R^2 = 0.999967$$

with the high aggregate shock; the ALM for the variance of capital, Σ , are given by

$$\ln(\Sigma_{t+1}) = 0.1745 - 0.0393 \ln(K_t) + 0.9958 \ln(\Sigma_t) \quad R^2 = 0.999699$$

with the low aggregate shock, and by

$$\ln(\Sigma_{t+1}) = 0.1705 - 0.0435 \ln(K_t) + 0.9958 \ln(\Sigma_t) \quad R^2 = 0.999920$$

with the high aggregate shock.

- **Results**

- We carried out all the required tests, and we calculated all the statistics for the model with one moment (mean).
- In the ALM-NO case, we simulate the time series of length $T = 1100$ for a panel of $N = 10000$ agents using the individual decision rules, and we average the individual quantities to compute the aggregate quantities.
- In the ALM-YES case, we use the ALM to compute the time series of aggregate capital, and we use the individual decision rules to compute the time series for one agent.
- Our program was written in Matlab.
- Simulations were carried out on Pentium 4 PC with 2.2GHz processor.

Figure 2. Individual decision rules.

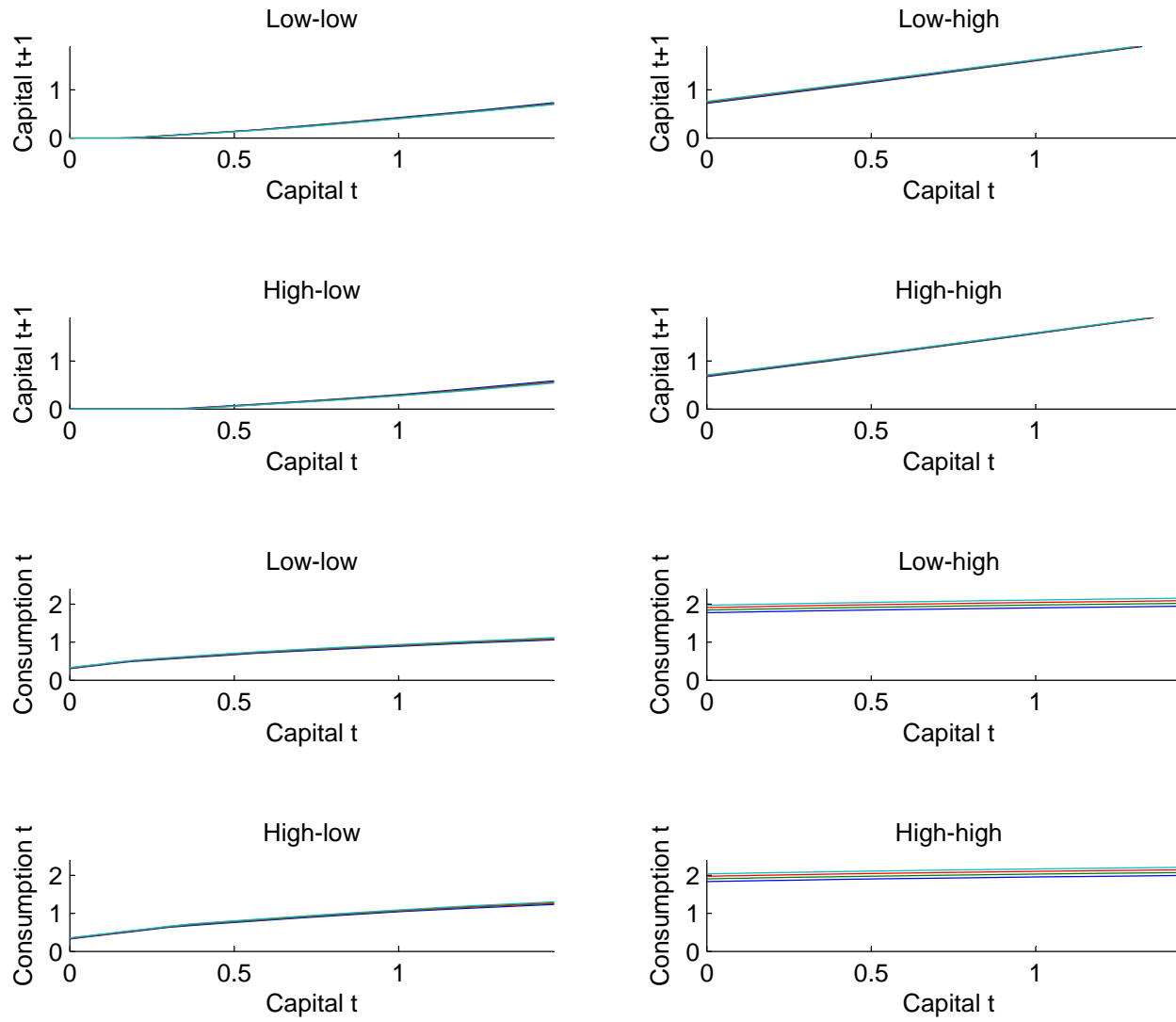


Figure 3. Properties of the solution.

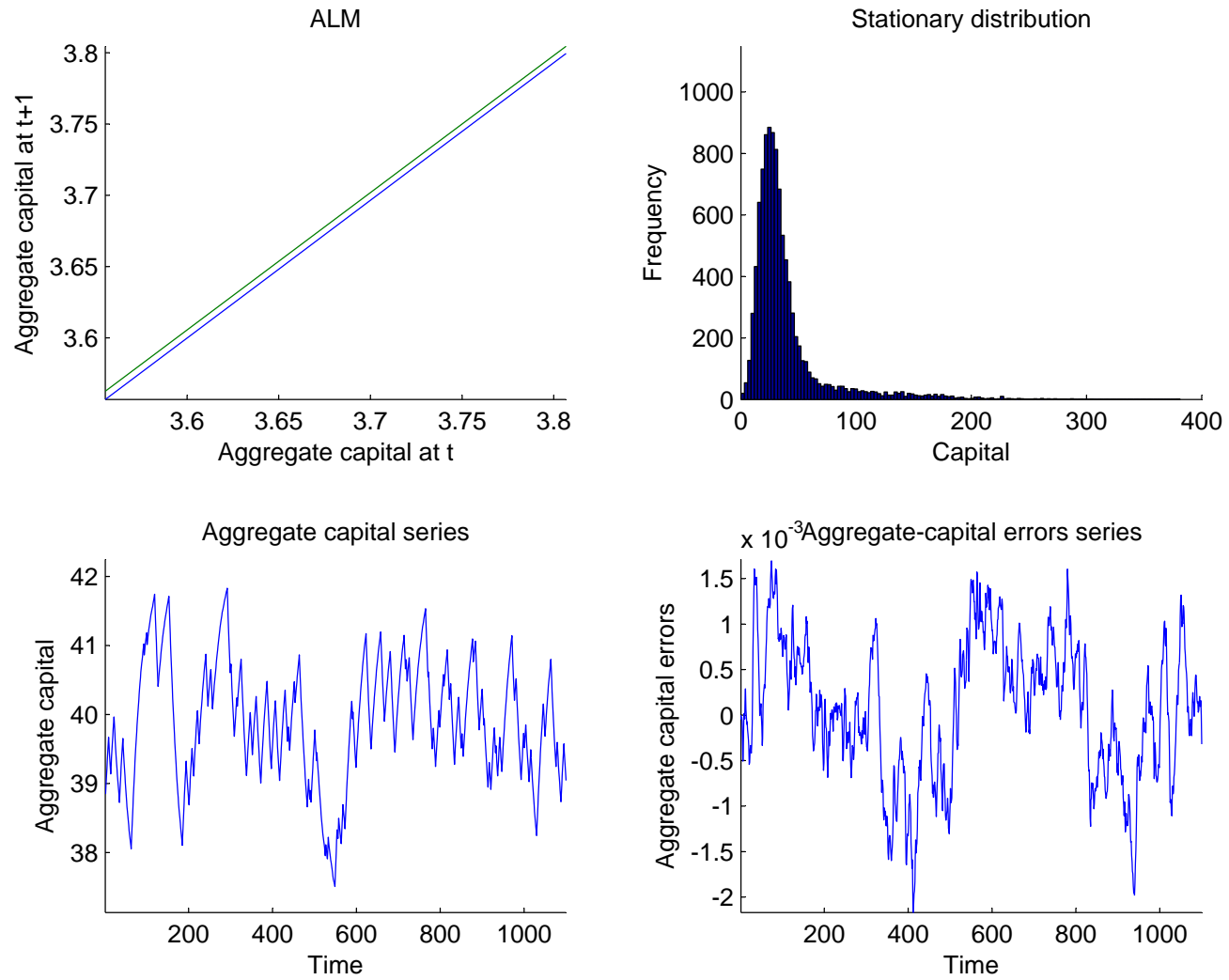


TABLE 4
Risk sharing statistics

	ALM-NO		ALM-YES
	Average across agents	Standard deviation	Statistic
Correlation of individual and aggregate consumption	0,4215	0,15806	0,44747
Correlation of individual consumption and aggregate income	0,28869	0,095484	0,30769
Correlation of individual consumption and aggregate wealth	0,4064	0,15831	0,42966
Correlation of individual consumption and individual income	0,54468	0,072166	0,57525
Correlation of individual consumption and individual wealth	0,91109	0,041365	0,93777
Standard deviation of individual consumption	0,11043		0,11478
Standard deviation of individual capital	7,3558		8,1248
Autocorrelation of individual consumption	0,98152	0,013052	0,98864
1 lag	0,96333	0,024117	0,97549
2 lags	0,94572	0,03352	0,96098
3 lags	0,99647	0,001932	0,99731
Autocorrelation of individual capital	0,98979	0,005551	0,99188
1 lag	0,98166	0,009914	0,98489
2 lags	0,4215	0,15806	0,44747
3 lags	0,28869	0,095484	0,30769

TABLE 5**Prices**

	ALM-NO		ALM-YES	
	Average	Standard deviation	Average	Standard deviation
Interest rate	0,034894	0,001055	0,034896	0,001056
Wage rate	2,384	0,018091	2,3839	0,018055
Autocorrelation of interest rate				
1 lag	0,7364		0,7381	
2 lags	0,51821		0,52035	
3 lags	0,3556		0,3564	
Autocorrelation of wage rate				
1 lag	0,97531		0,97527	
2 lags	0,9439		0,94363	
3 lags	0,9095		0,90893	

TABLE 6**Business cycle statistics**

	ALM-NO	ALM-YES
Average income	0,11889	0,12716
Average consumption	0,044095	0,042835
Average investment	0,10494	0,1049

TABLE 7**Cross correlation of income and consumption at leads and lags**

ALM-NO							
	y_t	y_{t-1}	y_{t-2}	y_{t-3}	c_t	c_{t-1}	
y_t	1	0,8019	0,63418	0,50755	0,68522	0,558	(
y_{t-1}		1	0,80161	0,63241	0,71034	0,68421	(
y_{t-2}			1	0,79998	0,71611	0,70939	(
y_{t-3}				1	0,71139	0,71581	(
c_t					1	0,98237	(
c_{t-1}						1	(
c_{t-2}]
c_{t-3}							

ALM-YES							
	y_t	y_{t-1}	y_{t-2}	y_{t-3}	c_t	c_{t-1}	
y_t	1	0,79687	0,62504	0,49286	0,6434	0,51944	(
y_{t-1}		1	0,79663	0,6233	0,67902	0,64235	(
y_{t-2}			1	0,79504	0,69336	0,67804	(
y_{t-3}				1	0,69515	0,69319	(
c_t					1	0,98567	(
c_{t-1}						1	(
c_{t-2}]
c_{t-3}							

TABLE 8
Cross correlation of income and investment at leads and lags

ALM-NO							
	y_t	y_{t-1}	y_{t-2}	y_{t-3}	i_t	i_{t-1}	
y_t	1	0,80156	0,6336	0,50684	0,943	0,75164	0,
y_{t-1}		1	0,80129	0,63188	0,68231	0,94286	0,
y_{t-2}			1	0,79971	0,4689	0,68194	0,
y_{t-3}				1	0,31172	0,4665	0,
i_t					1	0,74176	0,
i_{t-1}						1	0,
i_{t-2}							1
i_{t-3}							

ALM-YES							
	y_t	y_{t-1}	y_{t-2}	y_{t-3}	i_t	i_{t-1}	i_{t-2}
y_t	1	0,79687	0,62504	0,49286	0,94989	0,75376	0,
y_{t-1}		1	0,79663	0,6233	0,68903	0,94979	0,
y_{t-2}			1	0,79504	0,47481	0,68873	0,
y_{t-3}				1	0,3138	0,47246	0,
i_t					1	0,74228	0,
i_{t-1}						1	0,
i_{t-2}							1
i_{t-3}							

TABLE 9
Cross correlation of consumption and investment at leads and lags

ALM-NO							
	c_t	c_{t-1}	c_{t-2}	c_{t-3}	i_t	i_{t-1}	i_{t-2}
c_t	1	0,74176	0,52724	0,36911	0,943	0,68231	0,
c_{t-1}		1	0,74165	0,52525	0,75164	0,94286	0,
c_{t-2}			1	0,73967	0,58751	0,75133	0,
c_{t-3}				1	0,46475	0,5861	0,
i_t					1	0,80156	0,
i_{t-1}						1	0,
i_{t-2}							1
i_{t-3}							

ALM-YES							
	c_t	c_{t-1}	c_{t-2}	c_{t-3}	i_t	i_{t-1}	i_{t-2}
c_t	1	0,98567	0,96212	0,93277	0,37187	0,42059	0,
c_{t-1}		1	0,98566	0,96201	0,22741	0,37029	0,
c_{t-2}			1	0,9856	0,1161	0,22571	0,
c_{t-3}				1	0,03468	0,11408	0,
i_t					1	0,74228	0,
i_{t-1}						1	0,
i_{t-2}							1
i_{t-3}							

TABLE 10
Absolute percentage error

	Average error	Maximum error
Euler equation errors on a simulated time path (ALM-NO)	0,0021769	0,021991
Euler equation errors on a simulated time path (ALM-YES)	0,0086893	0,10129
Aggregate policy rule errors	0,00060847	0,0021909

TABLE 11
Den Haan and Marcet statistics

ALM-NO		ALM-YES	
Fraction of times below 2.5% critical value	0	Statistic for one agent	5,8149
Fraction of times above 97.5% critical value	43	Low critical value	1,2373
		High critical value	14,449

TABLE 12
Wealth distribution (ALM-NO)

	Unconditional	Conditional on low shock	Conditional on high shock
Fraction of times agent is at constraint	3,2062e-005	5,1954e-005	2,2478e-005
5 th percentile of wealth distribution	13,04	12,847	13,188
10 th percentile of wealth distribution	16,201	16,017	16,342
Average capital stock	39,934	39,715	40,103
Moment 2	0,86301	0,86523	0,86131
Moment 3	1,2616	1,2645	1,2595
Moment 4	1,7022	1,7061	1,6991
Moment 5	2,0751	2,0798	2,0714

The properties of the solution

- The statistics obtained under ALM-NO and ALM-YES are very close.
- Solutions are sufficiently accurate:
 - Maximum difference between the series for capital under ALM-NO and ALM-YES is about 0.2%.
- Our method does not have accumulation of errors over time since the budget constraint was imposed when calculating the solution.

Bisection&updating

- We propose a bisection and updating method that can be a fast alternative to Krusell and Smith's (1998) updating.
- Before we present this method, let us recall the standard one-dimensional bisection method for solving for the interest rate in the model without aggregate uncertainty, e.g., Aiyagari (1994).

One-dimensional section

- Suppose that we know two values, \underline{r} and \bar{r} , on the opposite sides of the steady state interest rate r^* , such that $\underline{r} < r^* < \bar{r}$.
- Define $r^{bis} = (\bar{r} + \underline{r}) / 2$ and compute the corresponding wealth distribution and the interest rate \tilde{r} . If $\tilde{r} > r^{bis}$, let $\bar{r} = r^{bis}$; otherwise, let $\underline{r} = r^{bis}$.

Iterate until $\tilde{r} = r^{bis} = r^*$.

The bisection method works fast and guarantees the convergence in 10-20 iterations. Unfortunately, the bisection method cannot be directly extended to more than one dimension because there is no easy way to make a bisection of vectors.

The ALM: bisection&updating

For multi-dimensional case, we propose a method that combines bisection and updating, namely, we perform bisection of one coefficient, and we update of the other coefficients. Let $b = (b_1, \dots, b_S)$, and let b_S be the bisection coefficient.

- *Step I.* Fix some initial vector b .
- ...
- *Step Va'.* Call the re-estimated ALM coefficients by $\tilde{b} = (\tilde{b}_1, \dots, \tilde{b}_S)$. If $\tilde{b}_S > b_S$, let $\bar{b} = \tilde{b}$ and $\underline{b} = b$; otherwise, let $\bar{b} = b$ and $\underline{b} = \tilde{b}$.
- *Step Vb'.* Let $b^{bis} = (\bar{b} + \underline{b}) / 2$. Perform *Steps II – IV* of the algorithm for b^{bis} , and get the new vector \tilde{b} . If $\tilde{b}_S > b^{bis}$ let $\bar{b} = b^{bis}$; otherwise let $\underline{b} = b^{bis}$.

- *Step VI.* Compute the ALM for next iteration by using updating: $\tilde{b} = \eta_b \tilde{b} + (1 - \eta_b) b$, $\eta_b \in (0, 1]$.

Iterate on *Steps II – VI* until $\tilde{b} = b$.

- Unlike in one-dimensional case, the completion of the bisection steps $Va' - Vb'$ does not guarantee finding the true solution because the convergence of b_S was achieved for some sequence of other coefficients (b_1, \dots, b_{S-1}) over which we had no control.
- Thus, at the end of the bisection step, we update \tilde{b} for next iteration and continue iterations until the convergence of all the coefficients, $\tilde{b} = b$.
- In some cases, our bisection and updating method can imply a substantial reduction of computational time compared to the updating method.

Bisection&updating versus updating

- To carry out the comparison, we consider Krusell and Smith's (1998) model. We assume the following ALM:

$$\ln(K_{t+1}) = b_1 + b_2 a_t + b_3 \ln(K_t).$$

This specification is more restrictive than the one in Krusell and Smith (1998) because it forces the coefficient on $\ln(K_t)$ to be the same with both high and low aggregate productivity shocks.

- We solve the model first by using Krusell and Smith's (1998) updating and then by using our bisection and updating method for the same set of parameters, the same realizations of the individual and aggregate shocks and the same initial guess for the coefficients, $b = (0, 0, 1)$.

Results of this experiments are summarized below:

Updating	Bisection and updating
$b_1 = 0.08799863$	$b_1 = 0.08800197$
$b_2 = 0.00532372$	$b_2 = 0.00532067$
$b_3 = 0.96321271$	$b_3 = 0.96321262$
$R^2 = 0.99994110$	$R^2 = 0.99994110$
Time = 34.24 min	Time = 13.44 min

- First, the one-equation parameterization leads to almost as accurate solution as one obtained under Krusell and Smith's (1998) state-contingent ALM. (Under their parameterization, we have $R^2 = 0.99994989$ and $R^2 = 0.99992851$ with the high and low states, respectively).

- Second, given the same parameterization and the same degree of accuracy, our bisection and updating method delivers a solution almost three times faster than the updating. (It took us more than an hour to solve the model by using the updating under Krusell and Smith's (1998) state-contingent ALM).
- We have discussed an application of our bisection and updating method only to a simple setting, however, we expect this to be also useful in more sophisticated settings.
- At least, we can use this method for fast solving a one-moment version of the model, so that the obtained solution is subsequently used as an initial guess for updating in several-moment versions of the model.

Conclusion

- We describe an algorithm for solving Krusell and Smith's (1998) model with continuum of agents and savings through capital.
- To solve the individual problem, we use an Euler equation algorithm iterating on a grid of prespecified points.
- To solve for the ALM, we summarize the economy's state by the first two moments of the wealth distribution and use Monte Carlo simulation.
- Our algorithm is relatively fast, especially, if the ALM is computed with our bisection and updating method: in our example, finding a solution to the benchmark Krusell and Smith's (1998) model requires around 13 minutes.

- Overall, our results are close to those in Young (2003), which is not surprising given that we characterize the aggregate state space with the same set of statistics as he does.
- It would be of interest to see the performance of our method in the context of Model C, where, as found by Young (2003) and Reiter (2003), no single aggregate statistics is sufficient for accurate forecasting of the future prices.