Economics Letters 120 (2013) 262–266

Contents lists available at SciVerse ScienceDirect

Economics Letters

journal homepage: www.elsevier.com/locate/ecolet

Envelope condition method versus endogenous grid method for solving dynamic programming problems

Lilia Maliar *, Serguei Maliar

Hoover Institution, Stanford University, USA
University of Alicante, Spain

HIGHLIGHTS

- We introduce the envelope condition method (ECM) for solving dynamic programming problems.
- ECM simplifies rootfinding and is faster than conventional value function iteration.
- ECM is similar in accuracy and speed to Carroll's (2005) endogenous grid method (EGM).
- We introduce accurate EGM and ECM that approximate derivatives of value function.
- Codes are available.

ARTICLE INFO

Article history:
Received 5 January 2013
Received in revised form
14 April 2013
Accepted 19 April 2013
Available online 24 April 2013

JEL classification:
C6
C61
C63
C68

Keywords:
Numerical dynamic programming
Value function iteration
Endogenous grid
Envelope condition
Curse of dimensionality
Large scale

ABSTRACT

We introduce an envelope condition method (ECM) for solving dynamic programming problems. The ECM method is simple to implement, dominates conventional value function iteration and is comparable in accuracy and cost to Carroll's (2005) endogenous grid method. Codes are available.

1. Introduction

Dynamic programming methods are an important tool in economics; see Judd (1998), Santos (1999), Rust (2008) and Stachursky (2009) for reviews. Conventional value function iteration (VFI) goes backward: we guess a value function in period $t+1$, and we solve for a value function in period t using the Bellman equation. Conventional VFI is expensive: it requires us to find a root to a non-linear equation in all grid points, which involves interpolating value function off the grid and approximating conditional expectation in a large number of candidate solution points; see Aruoba et al. (2006) for examples assessing the cost of VFI.

Carroll (2005) introduces an endogenous grid method (EGM) that simplifies rootfinding under time iteration. The idea is to construct a grid on future endogenous state variables instead of current endogenous state variables, which are treated as unknowns. In a typical economic model, it is easier to solve for current endogenous state variables given the future state variables than to solve for future endogenous state variables given the current state variables. This is why EGM dominates conventional VFI.

In this paper, we have two contributions. First, we introduce *an envelope condition method (ECM)*, another solution method that

* Correspondence to: T24, Hoover Institution, 434 Galvez Mall, Stanford University, Stanford, CA 94305-6010, USA. Tel.: +1 650 725 3416.
E-mail address: maliarl@stanford.edu (L. Maliar).

0165-1765/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.econlet.2013.04.031

simplifies rootfinding in dynamic programming problems. ECM does not perform conventional backward iteration on the Bellman equation but iterates forward. Also, to construct policy functions, ECM uses the envelope condition instead of the first-order conditions used by conventional VFI and EGM. We find that systems of equations produced by ECM are typically easier to solve than those produced by conventional VFI. In this sense, ECM is similar to EGM.

Second, we introduce versions of the EGM and ECM methods that approximate derivatives of value function instead of value function itself. We find that these versions produce far more accurate solutions than do similar methods that approximate value function itself.

We compare the EGM and ECM methods using both analytical arguments and numerical examples. We find that EGM and ECM are nearly identical in terms of accuracy and speed in our test problem, the neoclassical growth model with elastic labor supply. Codes are available at http://www.stanford.edu/~maliarl.

2. The model

We study the standard neoclassical growth model with elastic labor supply.

2.1. Bellman equation

We solve for value function V that satisfies the Bellman equation,

$$V(k,a)=\max_{k',c,\ell}\left\{u(c,\ell)+\beta E\left[V\left(k',a'\right)\right]\right\} \tag{1}$$

$$\text{s.t. } k'=(1-\delta)k+af(k,\ell)-c, \tag{2}$$

$$\ln a'=\rho\ln a+\epsilon',\quad \epsilon'\sim\mathcal{N}\left(0,\sigma^2\right), \tag{3}$$

where k, c, ℓ and a are capital, consumption, labor and productivity level, respectively; $\beta\in(0,1)$; $\delta\in(0,1]$; $\rho\in(-1,1)$; $\sigma\geq 0$; the utility and production functions, u and f, respectively, are strictly increasing, continuously differentiable and concave; the primes on variables denote next-period values, and $E\left[V\left(k',a'\right)\right]$ is an expectation conditional on state (k,a).

2.2. Optimality conditions

We divide the optimality conditions in two blocks. The first block identifies policy functions that correspond to a given value function V, and the second block identifies a value function that corresponds to given policy functions.

2.2.1. Block 1: identifying policy functions given a value function

If a solution to Bellman equation (1)–(3) is interior, the optimal quantities satisfy first-order conditions (FOCs) with respect to labor and consumption and the envelope condition, which, respectively, are

$$u_\ell(c,\ell)=-u_c(c,\ell)af_\ell(k,\ell), \tag{4}$$

$$u_c(c,\ell)=\beta E\left[V_k\left(k',a'\right)\right], \tag{5}$$

$$V_k(k,a)=u_c(c,\ell)\left[1-\delta+af_k(k,\ell)\right], \tag{6}$$

as well as budget constraint (2). Here, $F_x(\ldots,x,\ldots)$ denotes a first-order partial derivative of function $F(\ldots,x,\ldots)$ with respect to variable x.

2.2.2. Block 2: identifying a value function given policy functions

In the optimum, value function V and its derivative V_k satisfy

$$V(k,a)=u(c,\ell)+\beta E\left[V\left(k',a'\right)\right], \tag{7}$$

$$V_k(k,a)=\beta\left[1-\delta+af_k(k,\ell)\right]E\left[V_k\left(k',a'\right)\right]. \tag{8}$$

Condition (7) is Bellman equation (1) evaluated under the optimal policy functions (which makes a maximization sign unnecessary), and condition (8) follows by combining (5) and (6).

2.3. Discussion

Envelope condition (6) is central to our analysis.[1] Observe that we have two conditions that describe the relation between V_k and the policy functions: one is FOC (5) and the other is envelope condition (6). Conventional VFI and EGM of Carroll (2005) approximate policy functions using FOC (5), namely, they solve the system (2), (4) and (5). In contrast, our ECM method will approximate policy functions using envelope condition (6), namely, it will solve the system (2), (4) and (6). In Sections 3 and 4, we show that the system of equations built on envelope condition (6) is easier to solve than the system of equations built on conventional FOC (5), in which case ECM is a preferred choice.

Furthermore, the envelope condition provides a basis for condition (8). This condition allows us to approximate V_k without finding V. Under our construction, all methods described in the paper can approximate a solution by iterating on either (7) or (8) or both, whereas the previous literature including conventional VFI and EGM of Carroll (2005) iterate only on Bellman equation (7). In Section 5, we show that the iteration on (8) leads to far more accurate solutions than iteration on (7).

3. The model with inelastic labor supply

We first consider a model with inelastic labor supply under the following assumptions

$$u(c,\ell)=\frac{c^{1-\gamma}-1}{1-\gamma}\quad\text{and}\quad f(k,\ell)=k^\alpha, \tag{9}$$

where $\gamma>0$ and $\alpha\in(0,1)$. In this case, we have $\ell=1$, and FOC (4) is absent.

3.1. Conventional VFI

The conventional VFI method makes a guess on the future value function $V\left(k',a'\right)$ and identifies policy functions using budget constraint (2) and FOC (5). By substituting c from (2) into (5) under the assumptions (9), we obtain

$$\beta E\left[V_k\left(k',a'\right)\right]=\left[k'-(1-\delta)k-ak^\alpha\right]^{-\gamma}. \tag{10}$$

We must solve (10) for k' in each grid point (k,a). Finding a solution to (10) is expensive. For example, if we parameterize V with a polynomial function, then solving (10) includes interpolation of V_k to new values $\left(k',a'\right)$, as well as approximation of conditional expectation $E\left[V_k\left(k',a'\right)\right]$. We must explore many different candidate values of $\left(k',a'\right)$ until we find a solution to (10).

3.2. Endogenous grid method

The EGM of Carroll (2005) also makes a guess on the future value function $V\left(k',a'\right)$ and identifies policy functions using budget constraint (2) and FOC (5). The difference is that EGM treats the future endogenous state variable as fixed, and it treats the current endogenous state variable as unknown. Since the values for k' are fixed, it is possible to compute up-front $E\left[V\left(k',a'\right)\right]\equiv W\left(k',a\right)$ and $E\left[V_k\left(k',a'\right)\right]\equiv W_k\left(k',a\right)$.

[1] Typically, the envelope condition is used to derive the Euler equation (namely, (6) is updated to get $V_k\left(k',a'\right)$ and the result is substituted into (5) to eliminate the unknown derivative of the value function). In the present paper, we do not derive the Euler equation but concentrate on the envelope condition in the form (6).

Consider again the system (2), (4) and (5) under assumptions (9). Now, we find c directly from (5), $c = \left[\beta W_k\left(k', a\right)\right]^{-1/\gamma}$, and we are left to solve for k that satisfies budget constraint (2) given $\left(k', a\right)$

$$(1-\delta)\,k + ak^{\alpha} = \left[\beta W_k\left(k', a\right)\right]^{-1/\gamma} + k'. \tag{11}$$

Observe that (11) is easier to solve numerically than (10) because it does not involve either interpolation or approximation of conditional expectation.

Carroll's (2005) change of variables. Still, Eq. (11) must be solved numerically. However, Carroll (2005) finds a clever change of variables that makes unnecessary solving (11) on each iteration. He introduces a new variable $Y \equiv (1-\delta)\,k + ak^{\alpha} = c + k'$, which allows us to rewrite Bellman equation (1) as

$$V\left(Y, a\right) = \max_{k'}\left\{\frac{c^{1-\gamma}-1}{1-\gamma} + \beta W\left(Y', a\right)\right\}, \tag{12}$$

where $W\left(Y', a\right) = E\left[V\left(Y', a'\right)\right]$. The FOC of this problem is

$$c^{-\gamma} = \beta E\left[V_Y\left(Y', a'\right)\left(1-\delta+\alpha a'\left(k'\right)^{\alpha-1}\right)\right]. \tag{13}$$

Since we know that $Y' \equiv (1-\delta)\,k' + a'\left(k'\right)^{\alpha}$, we can find the expectation in the right side of (13) and hence, we can compute c and $Y = k' + c$. Therefore, we can iterate on Bellman equation (12) without using a solver. Once V is computed, we find k that corresponds to $Y = (1-\delta)\,k + ak^{\alpha}$ using a numerical solver (just once).

3.3. Envelope condition method

Like the conventional VFI, our ECM method also operates on an exogenous grid however makes a guess on the current value function $V\left(k, a\right)$ (or its derivative $V_k(k, a)$) instead of the future value function. This enables us to solve for c using envelope condition (6) instead of FOC (5). Under assumptions (9), c can be derived explicitly from (6),

$$c = \left(\frac{V_k\left(k, a\right)}{1-\delta+\alpha a(k)^{\alpha-1}}\right)^{-1/\gamma}. \tag{14}$$

We can next compute k' directly from budget constraint (2). In this example, ECM is simpler than Carroll's (2005) EGM as all policy functions can be constructed analytically and a solver must never be used (not even once).

3.4. Discussion

Four combinations are possible from two alternative conditions for V_k (FOC (5) and envelope condition (6)) and two alternative grids (exogenous and endogenous). So far, we have distinguished two competitive methods: one is EGM of Carroll (2005) (FOC (5) and endogenous grid) and the other is our ECM (envelope condition (6) and exogenous grid). The conventional VFI (FOC (5) and exogenous grid) is not competitive. Therefore, we are left to explore the remaining combination (envelope condition (6) and endogenous grid). Combining (2) and (14) yields

$$(1-\delta)\,k + ak^{\alpha} = \left(\frac{V_k\left(k, a\right)}{1-\delta+\alpha a(k)^{\alpha-1}}\right)^{-1/\gamma} + k'. \tag{15}$$

We must solve (15) for k given $\left(k', a\right)$. This involves evaluations of $V_k\left(k, a\right)$ for many candidate solution points (k, a), which are costly. We conclude that the combination of the envelope condition and endogenous grid does not lead to a competitive method. Our results are suggestive for other applications.

4. The model with elastic labor supply

We now consider the model with elastic labor supply under the following assumptions:

$$u\left(c, \ell\right) = \frac{c^{1-\gamma}-1}{1-\gamma} + B\frac{(1-\ell)^{1-\mu}-1}{1-\mu} \quad \text{and} \qquad f\left(k, \ell\right) = k^{\alpha}\ell^{1-\alpha}, \tag{16}$$

where $\gamma > 0$, $\mu > 0$ and $\alpha \in (0, 1)$. We restrict attention to EGM and ECM that we found to be competitive.

4.1. Endogenous grid method

Under EGM, we must solve Eqs. (2), (4) and (5) for (c, ℓ, k) given $\left(k', a\right)$. As in the model with inelastic labor supply, we compute $E\left[V\left(k', a'\right)\right] \equiv W\left(k', a\right)$, $E\left[V_k\left(k', a'\right)\right] \equiv W_k\left(k', a\right)$ given V, and we find $c = \left[\beta W_k\left(k', a\right)\right]^{-1/\gamma}$ using (4). Under (16), we can express k from (4) and substitute it into (2) to get

$$k' = (1-\delta)\left(\frac{B\,(1-\ell)^{-\mu}}{\beta W_k\left(k', a\right) a\,(1-\alpha)}\right)^{1/\alpha}\ell + \frac{B\,(1-\ell)^{-\mu}\,\ell}{\beta W_k\left(k', a\right)(1-\alpha)} - \left[\beta W_k\left(k', a\right)\right]^{-1/\gamma}. \tag{17}$$

Eq. (17) must be solved numerically for one unknown ℓ. This equation is relatively cheap as it does not involve either interpolation or approximation of expectations.

4.2. Envelope condition method

Under ECM, we must solve Eqs. (2), (4) and (6) for $\left(c, \ell, k'\right)$ given (k, a). By substituting $c^{-\gamma}$ from (4) into envelope condition (6), we obtain

$$V_k\left(k, a\right) = \frac{B\,(1-\ell)^{-\mu}}{a\,(1-\alpha)\,k^{\alpha}\ell^{-\alpha}}\left[1-\delta+a\alpha k^{\alpha-1}\ell^{1-\alpha}\right]. \tag{18}$$

We must solve Eq. (18) for ℓ. Like (17), Eq. (18) does not involve either interpolation or approximation of expectations.

4.3. Discussion

Under our implementation, the rootfinding problems under EGM and ECM are comparable in their complexity. In both cases, we must find a solution to a non-linear equation in each grid point. Such an equation is relatively cheap to solve as it does not involve either interpolation or approximation of expectations.

In the model with elastic labor supply, Carroll's (2005) change of variables does not avoid rootfinding. The variable $Y' = a'f\left(k', \ell'\right) + k'$ depends on future labor ℓ', and $E\left[V\left(Y', a'\right)\right]$ cannot be computed without specifying labor policy functions. Barillas and Fernández-Villaverde (2007) propose a way of extending EGM to the model with elastic labor supply. Namely, they fix a policy function for labor $\ell = L\left(k', a\right)$, construct the grid of $\left(Y', a\right)$, solve the model on that grid holding L fixed and use the solution to reevaluate L; and they iterate on these steps until L converges.

Our implementation of EGM for the model with elastic labor supply differs from that in Barillas and Fernández-Villaverde (2007). First, we use future endogenous state variables for constructing grid points but we do not use Carroll's (2005) change of variables. Second, to deal with rootfinding, we use a numerical solver while Barillas and Fernández-Villaverde (2007) iterate on a state contingent policy function for labor $L\left(k', a\right)$.

5. Numerical analysis

We compare the performance of EGM and ECM in the context of the model with elastic labor supply.

Table 1
Accuracy and speed of EGM-VF and ECM-VF in the model with elastic labor supply.[a]

Polynomial degree	EGM-VF			ECM-VF		
	L_1	L_∞	CPU	L_1	L_∞	CPU
1st	–	–	–	–	–	–
2nd	−3.28	−2.81	8.3	−3.34	−2.75	5.8
3rd	−4.31	−3.99	8.9	−4.38	−3.87	7.2
4th	−5.32	−4.96	7.3	−5.45	−4.86	5.8
5th	−6.37	−5.85	6.5	−6.57	−5.72	4.7

[a] Notes: L_1 and L_∞ are, respectively, the average and maximum of absolute residuals across optimality condition and test points (in log10 units) on a stochastic simulation of 10,000 observations; CPU is the time necessary for computing a solution (in s).

Table 2
Accuracy and speed of EGM-DVF and ECM-DVF in the model with elastic labor supply.[a]

Polynomial degree	EGM-DVF			ECM-DVF		
	L_1	L_∞	CPU	L_1	L_∞	CPU
1st	−3.03	−2.87	8.1	−3.08	−2.92	7.2
2nd	−4.13	−3.82	7.2	−4.18	−3.91	6.5
3rd	−5.06	−4.77	7.3	−5.20	−4.87	6.7
4th	−6.09	−5.64	7.4	−6.29	−5.72	6.8
5th	−7.12	−6.26	7.6	−7.36	−6.32	6.9

[a] Notes: L_1 and L_∞ are, respectively, the average and maximum of absolute residuals across optimality condition and test points (in log10 units) on a stochastic simulation of 10,000 observations; CPU is the time necessary for computing a solution (in s).

5.1. Methodology

We calibrate the model (1)–(3) under (16) such that in the steady state, the capital–output ratio is $\pi_k = 10$, the consumption–output ratio is $\pi_c = 3/4$, the steady state labor is $\ell = 1/3$ and $\alpha = 1/3$; this implies $\beta = 0.99$, $\delta = 0.025$ and $B = (1-\alpha)\,\pi_k^{(1-\gamma)\alpha/(1-\alpha)}\pi_c^{-\gamma}\,(1-\ell)^{\mu}\,\ell^{-\gamma}$. In the benchmark case, we use $(\gamma,\mu) = (2, 2)$. The parameters in (3) are $\rho = 0.95$ and $\sigma = 0.01$. Our design of EGM and ECM is similar. As a solution domain, we use a rectangular, uniformly spaced grid of 10×10 points for capital and productivity within an ergodic range. We use a 3-node Gauss–Hermite quadrature rule for approximating integrals. We parameterize value function with complete ordinary polynomials of degrees up to 5. To solve for the polynomial coefficients, we use fixed-point iteration. To solve non-linear equations (17) and (18), we use a solver *csolve* written by Christopher Sims. We use MATLAB software, version 7.6.0.324 (R2008a) and a desktop computer ASUS with Intel(R) Core(TM)2 Quad CPU Q9400 (2.66 GHz), 6 GB RAM. A detailed description of the algorithms is provided in the Appendix.

5.2. Results for the model with elastic labor supply

We first solve for V by iterating on Bellman equation (7); we refer to the corresponding methods as EGM-VF and ECM-VF. The results are shown in Table 1. The performance of EGM-VF and ECM-VF is very similar. EGM-VF produces slightly smaller maximum residuals, while ECM-VF produces slightly smaller average residuals. EGM-VF is somewhat slower than ECM-VF.

We next solve for V_k by iterating on (8); we call these methods EGM-DVF and ECM-DVF. The results are provided in Table 2. Again, EGM-DVF and ECM-DVF perform very similarly. Both methods deliver accuracy levels that are about an order of magnitude higher than those of EGM-VF and ECM-VF. Overall, we attain accuracy levels that are comparable to the best accuracy attained in the related literature.

Iterating on (8) produces more accurate solutions than iterating on (7) because the object that is relevant for accuracy is V_k and not V (namely, V_k identifies the model's variables from (2)–(6)). Approximating a supplementary object V and computing its derivative V_k involves an accuracy loss compared to the case when we focus on the relevant object V_k directly. For example, if we approximate V with a polynomial, we effectively approximate V_k with a polynomial which is one degree lower, i.e., we "lose" one polynomial degree.

We finally implement versions of EGM and ECM which approximate V jointly with V_k by iterating on both (7) and (8); we call them EGM-VF&DVF and ECM-VF&DVF. We specifically fit a polynomial approximation for V on the grid using a constrained linear least-squares that imposes a linear restriction on the coefficients of a polynomial that approximates V_k. This procedure is similar in spirit to a Hermite interpolation method described in Cai and Judd (2012). In our simple example, approximating V jointly with V_k leads to the same results as those obtained approximating V alone. However, in more complex models in which value function has many endogenous arguments, fitting both V and V_k on the grid may improve accuracy of solutions because it imposes consistency on cross derivatives of V.

6. Conclusion

The conventional VFI is expensive. Carroll (2005) introduces the EGM method that reduces the cost of value iteration dramatically. In this paper, we propose the ECM method that can compete with Carroll's (2005) method. In our simple application, EGM and ECM perform similarly. But in more complex applications, one method may lead to a more simple system of equations and thus, be preferable to the other. One application in which ECM can be a useful choice is models of sovereign default; see, e.g., Villemot (2012).

In this paper, we build ECM and EGM using tensor product grids. However, ECM and EGM can be implemented using non-product techniques that are tractable in high dimensional applications; see Maliar and Maliar (2005) for a numerical method that solves for a value function on simulated series, and see Judd et al. (2011, 2012) for effective non-product grid constructions, low-cost monomial integration formulas and numerically stable fitting methods. In particular, Maliar and Maliar (2012) show versions of ECM that solve dynamic programming problems with up to 16 state variables.

Acknowledgments

Lilia Maliar and Serguei Maliar acknowledge support from the Hoover Institution and Department of Economics at Stanford University, University of Alicante, Ivie, MCI and FEDER funds under the projects SEJ-2007-62656 and ECO2012-36719. We thank Kenneth L. Judd for many valuable comments.

Appendix

We first provide a description of 3 versions of the ECM used.

ECM-VF, ECM-DVF, ECM-VF&DVF

Step 0. *Initialization*.
a. Choose an approximating function $\widehat{V}(\cdot; b) \approx V$.
b. Choose integration nodes, ϵ_j, and weights, ω_j, $j = 1, \ldots, J$.
c. Construct a grid $\Gamma = \{k_m, a_m\}_{m=1,\ldots,M}$
d. Make an initial guess on $b^{(1)}$.

Step 1. *Computation of a solution for* V.

At iteration i, for $m = 1, \ldots, M$,

a. Solve for ℓ_m that satisfies

$$B(1-\ell_m)^{-\mu}\left(1-\delta+a\alpha k_m^{\alpha-1}\ell_m^{1-\alpha}\right) = \widehat{V}_k\left(k_m, a_m; b^{(i)}\right) a(1-\alpha) k_m^{\alpha}\ell_m^{-\alpha};$$

b. Compute $c_m \equiv \left[\frac{\widehat{V}_k\left(k_m, a_m; b^{(i)}\right)}{1-\delta+a\alpha k_m^{\alpha-1}\ell_m^{1-\alpha}}\right]^{-1/\gamma}$;

c. Compute $k'_m = (1-\delta)k_m + a_m k_m^{\alpha}\ell_m^{1-\alpha} - c_m$;

ECM-VF. Find value function on the grid

$$v_m \equiv \frac{c_m^{1-\gamma}-1}{1-\gamma} + B\frac{(1-\ell_m)^{1-\mu}-1}{1-\mu} + \beta\sum_{j=1}^{J}\omega_j\widehat{V}\left(k'_m, a_m^{\rho}\exp\left(\epsilon_j\right); b^{(i)}\right);$$

ECM-DVF. Find the derivative of value function on the grid

$$d_m = \beta\left[1-\delta+\alpha a k_m^{\alpha-1}\ell_m^{1-\alpha}\right]\sum_{j=1}^{J}\omega_j \widehat{V}_k\left(k'_m, a_m^{\rho}\exp\left(\epsilon_j\right); b^{(i)}\right).$$

Step 2. *Computation of* b *that fits the value function on the grid*.

a. Run a regression to find $\widehat{b}$

ECM-VF. $\widehat{b} = \arg\min_b \sum_{m=1}^{M}\left\| v_m - \widehat{V}(k_m, a_m; b)\right\|$.

ECM-DVF. $\widehat{b} = \arg\min_b \sum_{m=1}^{M}\left\| d_m - \widehat{V}_k(k_m, a_m; b)\right\|$.

ECM-VF&DVF. $\widehat{b} = \arg\min_b \sum_{m=1}^{M}\left\| v_m - \widehat{V}(k_m, a_m; b)\right\|$ s.t. $d_m = \widehat{V}_k(k_m, a_m; b)$.

b. Use damping to compute $b^{(i+1)} = (1-\xi)b^{(i)} + \xi\widehat{b}$.

c. Check for convergence: end Step 2 if

$$\frac{1}{M}\sum_{m=1}^{M}\left|\frac{(k'_m)^{(i+1)}-(k'_m)^{(i)}}{(k'_m)^{(i)}}\right| < \varpi.$$

We now provide a description of 3 different versions of EGM used (steps that are identical under ECM and EGM are omitted).

EGM-VF, EGM-DVF, EGM-VF&DVF

…c. Construct a grid $\Gamma = \left\{k'_m, a_m\right\}_{m=1,\ldots,M}$

…At iteration i, for $m = 1, \ldots, M$,

a. Find $c_m = \left(\beta\sum_{j=1}^{J}\omega_j\widehat{V}_k\left(k'_m, a_m^{\rho}\exp\left(\epsilon_j\right); b^{(i)}\right)\right)^{-1/\gamma}$

b. Solve for ℓ_m that satisfies

$$k'_m = (1-\delta)\left(\frac{B(1-\ell_m)^{-\mu}}{c_m^{-\gamma}a(1-\alpha)}\right)^{1/\alpha}\ell_m + \frac{B(1-\ell_m)^{-\mu}\ell_m}{c_m^{-\gamma}(1-\alpha)} - c_m;$$

c. Compute $k_m \equiv \left(\frac{B(1-\ell_m)^{-\mu}}{c_m^{-\gamma}a(1-\alpha)\ell_m^{-\alpha}}\right)^{1/\alpha}$;…

References

Aruoba, S., Fernández-Villaverde, J., Rubio-Ramírez, J., 2006. Comparing solution methods for dynamic equilibrium economies. Journal of Economic Dynamics and Control 30, 2477–2508.

Barillas, F., Fernández-Villaverde, J., 2007. A generalization of the endogenous grid method. Journal of Economic Dynamics and Control 31, 2698–2712. Elsevier.

Cai, Y., Judd, K.L., 2012. Dynamic programming with shape-preserving rational spline Hermite interpolation. Economics Letters 117, 161–164.

Carroll, K., 2005. The method of endogenous grid points for solving dynamic stochastic optimal problems. Economics Letters 91, 312–320.

Judd, K., 1998. Numerical Methods in Economics. MIT Press, Cambridge, MA.

Judd, K., Maliar, L., Maliar, S., 2011. Numerically stable and accurate stochastic simulation approaches for solving dynamic models. Quantitative Economics 2, 173–210.

Judd, K., Maliar, L., Maliar, S., 2012. Merging simulation and projection approaches to solve high-dimensional problems. NBER Working Paper 18501.

Maliar, L., Maliar, S., 2005. Solving nonlinear stochastic growth models: iterating on value function by simulations. Economics Letters 87, 135–140.

Maliar, L., Maliar, S., 2012. Solving the multi-country real business cycle model using an envelope-condition method. Manuscript.

Rust, J., 2008. Dynamic programming. In: Durlauf, S., Blume, L. (Eds.), The New Palgrave Dictionary of Economics. Palgrave Macmillan.

Santos, M., 1999. Numerical solution of dynamic economic models. In: Taylor, J., Woodford, M. (Eds.), Handbook of Macroeconomics. Elsevier Science, Amsterdam, pp. 312–382.

Stachursky, J., 2009. Economic Dynamics: Theory and Computation. MIT Press, Cambridge.

Villemot, S., 2012. Accelerating the resolution of sovereign debt models using an endogenous grid method. Dynare Working Paper 17. http://www.dynare.org/wp.