UNIVERSITÄT AUGSBURG
Universitätsbibliothek

OPEN ACCESS
OPUS AUGSBURG

Computation of Business Cycle Models: A Comparison of Numerical Methods

Alfred Maußner, Burkhard Heer

Angaben zur Veröffentlichung / Publication details:

Maußner, Alfred, and Burkhard Heer. 2008. "Computation of Business Cycle Models: A Comparison of Numerical Methods." *Macroeconomic Dynamics* 12 (5): 641–63. https://doi.org/10.1017/S1365100508070363.

Nutzungsbedingungen / Terms of use:

licgercopyright

Dieses Dokument wird unter folgenden Bedingungen zur Verfügung gestellt: / This document is made available under these conditions:
Deutsches Urheberrecht
Weitere Informationen finden Sie unter: / For more information see:
https://www.uni-augsburg.de/de/organisation/bibliothek/publizieren-zitieren-archivieren/publiz/

Macroeconomic Dynamics, **12**, 2008, 641–663. Printed in the United States of America.
doi:10.1017/S1365100508070363

COMPUTATION OF BUSINESS CYCLE MODELS: A COMPARISON OF NUMERICAL METHODS

BURKHARD HEER
Free University of Bolzano-Bozen
and
CESifo

ALFRED MAUßNER
University of Augsburg

We compare the numerical methods that are most widely applied in the computation of the standard business cycle model with flexible labor. The numerical techniques imply economically insignificant differences with regard to business cycle summary statistics. Furthermore, these results are robust with regard to the choice of the functional form of the utility function and the model's parameterization. In addition, the extended path approach, albeit time-consuming, and the Galerkin projection are found to be the most accurate methods, given that we have not used function approximations beyond the second degree.

Keywords: Log-Linearization, Perturbation Methods, Projection Methods, Extended Path, Value Function Iteration, Parameterized Expectations, Genetic Search

1. INTRODUCTION

The dominant paradigm of modern business cycle theory is the stochastic growth model with flexible labor. The stochastic growth model, however, is difficult to compute, as multiplicative elements such as the production function interact with additive elements such as depreciation or investment. As a consequence, only special cases (with log utility and full depreciation) can be solved analytically. For this reason, the comparison of different computational techniques that approximate the solution numerically is important. Previous work by Taylor and Uhlig (1990) has focused on the study of the stochastic growth model with inelastic labor supply, whereas Christiano and Fisher (2000) compare different numerical techniques for the solution of the stochastic growth model with binding constraints on nonnegative investment.

Part of this work was completed while Burkhard Heer was visiting Georgetown University and Stanford University. Burkhard Heer gratefully acknowledges support from the German Science Foundation (DFG) during this period. Address correspondence to: Alfred Maußner, Universität Augsburg, D-86159 Augsburg, Germany; e-mail: alfred.maussner@wiwi.uni-augsburg.de.

© 2008 Cambridge University Press 1365-1005/08

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

In the present paper, we evaluate the numerical techniques that are most widely applied in recent research on nonlinear rational expectations general equilibrium models. We analyze the properties of standard methods with regard to their accuracy and appropriateness for business cycle research. As is vital for the study of business cycles, we introduce flexible labor supply in the stochastic growth model. Furthermore, we study the sensitivity of the usual set of second moments of the variables that are important for the business cycle researcher, that is, output, employment, investment, consumption, and wages, with regard to the computational method, varying both the functional form of the utility function and the parameterization of the model. In particular, we apply parameter values in the range that are typically observed across countries.

We compare six different computation methods for two different parameter sets. The first set of parameters is chosen with regard to U.S. postwar data. The second set of parameters characterizes the German economy prior to unification. The solution methods are (1) log-linearization (LL), (2) second-order approximation (SO), (3) parameterized expectations (PE), (4) Galerkin projection (GA), (5) extended deterministic path (EP), and (6) value function iteration (VI). For the researcher who is only interested in the business cycle statistics, the log-linearization method is found to be the most convenient and appropriate for the standard business cycle model. Even if the researcher is interested in a more accurate solution, the log-linear solution might provide a good initial value for the parameters of more complex methods. To get comparable results, we do not use function approximation methods that involve higher-order terms than quadratic ones.[1] In terms of accuracy, however, we find that method (5) provides the best results and requires no more computation time than value function iteration.

The paper is organized as follows. In Section 2, the model is presented. In Section 3, we briefly review the methods most relevant for the computation of modern business cycle models, and Section 4 presents summary statistics for the various methods. Section 5 concludes.

2. THE STANDARD BUSINESS CYCLE MODEL

We consider a decentralized economy with households and firms. It is well known that the sequence of competitive equilibrium prices and quantities in this economy can be obtained from the solution of a stochastic planning problem. The fictitious planer maximizes

$$E_0 \sum_{t=0}^{\infty} \beta^t u(C_t, N_t), \tag{1}$$

subject to the resource constraint

$$K_{t+1} \leq Z_t (A_t N_t)^{\alpha} K_t^{1-\alpha} + (1-\delta) K_t - C_t \tag{2}$$

with respect to consumption C_0 and labor supply N_0 in period $t = 0$. The initial capital stock K_0 is given and the current level of total factor productivity Z_0 is

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

known. The evolution of this variable is governed by the AR(1)-process

$$\ln Z_t = \rho \ln Z_{t-1} + \epsilon_t, \tag{3}$$

where ϵ is a serially uncorrelated, normally distributed random variable with mean 0 and variance σ^2. Labor-augmenting technical progress A_t evolves according to the deterministic law

$$A_{t+1} = aA_t, \ a \geq 1. \tag{4}$$

Capital depreciates at the rate $\delta \in [0, 1]$, and utility obtained from consumption and leisure in period t, $u(C_t, N_t)$, is discounted by $\beta \in (0, 1)$. We consider four specifications of u that are commonly applied in business cycle models:

$$u(C_t, N_t) = \begin{cases} \frac{1}{1-\eta}\left[C_t^{1-\eta}(1-N_t)^{\theta(1-\eta)} - 1\right] & \text{I} \\ \frac{1}{1-\eta}\left[\left(C_t - \frac{\theta}{1+\nu}A_t N_t^{1+\nu}\right)^{1-\eta} - 1\right] & \text{II} \\ \ln C_t - \theta n_t & \text{III} \\ \ln C_t + \frac{\theta(1-N_t)^{1-\gamma}}{1-\gamma} & \text{IV.} \end{cases} \tag{5}$$

The functions I, III, and IV meet the requirements of King et al. (1988), p. 292, which allow for a balanced growth path in the presence of exogenous labor augmenting technical progress.[2] If one uses specification II, a balanced growth path exists only if one is willing to assume that the disutility of work is proportional to A_t.[3] We use the common parameter θ to ensure that the fraction of working hours per worker out of total hours available equals the respective empirical magnitude.

3. COMPUTATION OF SOLUTIONS

The solution of the model satisfies the following set of conditions, where lower case variables are scaled by A_t, that is, $x_t \equiv X_t/A_t$, except for the shadow price of wealth, Λ_t, where $\lambda_t := \Lambda_t A_t^{\eta}$:

$$\frac{\partial u(c_t, N_t)}{\partial c_t} = \lambda_t, \tag{6a}$$

$$\frac{\partial u(c_t, N_t)}{\partial N_t} = \alpha\lambda_t Z_t N_t^{\alpha-1} k_t^{1-\alpha}, \tag{6b}$$

$$ak_{t+1} = Z_t N_t^{\alpha} k_t^{1-\alpha} + (1-\delta)k_t - c_t, \tag{6c}$$

$$\lambda_t = \beta a^{-\eta} E_t \lambda_{t+1}(1 - \delta + (1-\alpha)Z_{t+1} N_{t+1}^{\alpha} k_{t+1}^{-\alpha}). \tag{6d}$$

Due to the recursive structure of the model, the solution can be represented by policy functions $x = g^x(k_t, Z_t)$, $x \in \{c_t, N_t, k_{t+1}\}$, that relate consumption c_t, hours N_t, and the next period capital stock k_{t+1} to the current state (k_t, Z_t) of the system.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

We solve this model with six different methods: (1) log-linearization and (2) second-order approximation, both belonging to the class of perturbation methods that use local information to compute the parameters of g^x, (3) parameterized expectations and (4) Galerkin projection, which are well-known examples of projection or weighted residuals methods, (5) deterministic extended path, and (6) value function iteration.[4]

The *log-linear solution method* and *second-order approximation* of the policy function use local information to determine the parameters of the policy functions. Both methods are special cases of perturbation methods [see Judd (1998), Chapter 13 and 14] that rest on the implicit function theorem and on Taylor's theorem. Let

$$\tilde{x} = g^x(\tilde{k}_t, \tilde{z}_t, \sigma), \quad \tilde{x} \in \{\tilde{k}_{t+1}, \tilde{c}_t, \tilde{N}_t, \tilde{\lambda}_t\},$$

denote the policy function for the natural logarithm of the stock of next-period capital, consumption, working hours, and marginal utility, respectively. The system of first-order conditions (6) can be written as

$$0 = u_c[g^c(\tilde{k}_t, \tilde{Z}_t, \sigma), g^N(\tilde{k}_t, \tilde{Z}_t, \sigma)] - g^\lambda(\tilde{k}_t, \tilde{Z}_t, \sigma), \tag{7a}$$

$$0 = u_N[g^c(\tilde{k}_t, \tilde{Z}_t, \sigma), g^N(\tilde{k}_t, \tilde{Z}_t, \sigma)] - \alpha e^{g^\lambda(\tilde{k}_t, \tilde{Z}_t, \sigma) + \tilde{Z}_t + (\alpha-1)g^N(\tilde{k}_t, \tilde{Z}_t, \sigma) + (1-\alpha)\tilde{k}_t}, \tag{7b}$$

$$0 = a e^{g^k(\tilde{k}_t, \tilde{Z}_t, \sigma)} - e^{\tilde{Z}_t + \alpha g^N(\tilde{k}_t, \tilde{Z}_t, \sigma) + (1-\alpha)\tilde{k}_t} - (1-\delta)e^{\tilde{k}_t} + e^{g^c(\tilde{k}_t, \tilde{Z}_t, \sigma)}, \tag{7c}$$

$$0 = e^{g^\lambda(\tilde{k}_t, \tilde{Z}_t, \sigma)} - \beta a^{-\eta} E_t e^{g^\lambda(g^k(\tilde{k}_t, \tilde{Z}_t, \sigma), \rho\tilde{Z}_t + \sigma v_{t+1}, \sigma)}[1 - \delta + (1-\alpha)e^{w_{t+1}}], \tag{7d}$$

$$w_{t+1} := \rho\tilde{Z}_t + \sigma v_{t+1} + \alpha g^N[g^k(\tilde{k}_t, \tilde{Z}_t, \sigma), \rho\tilde{Z}_t + \sigma v_{t+1}, \sigma] - \alpha g^k(\tilde{k}_t, \tilde{Z}_t, \sigma), \tag{7e}$$

$$v_t \sim N(0, 1). \tag{7f}$$

At $\sigma = 0$, the solution of this system of equations provides the stationary values of the log of capital, consumption, and working hours, $\tilde{k}^*$, $\tilde{c}^*$, and $\tilde{N}^*$, respectively. The coefficients of the log-linear approximation of the policy functions,

$$\tilde{x} = \tilde{x}^* + g^x_k(\tilde{k}_t - \tilde{k}^*) + g^x_Z(\tilde{Z}_t - \tilde{Z}^*) + g^x_\sigma\sigma,$$

can be found by differentiating (7a) through (7d) with respect to $\tilde{k}_t$, $\tilde{Z}_t$, and σ at the point $\sigma = 0$. This delivers a system of equations in the twelve unknown coefficients of the four policy functions. As Schmitt-Grohé and Uribe (2004) have shown, g^x_k and g^x_Z are independent of σ and $g^x_\sigma = 0$. Also, it is easy to show that the same coefficients are obtained by first log-linearizing the system (6) at the stationary equilibrium and then solving the resulting stochastic linear system of difference equations. Procedures to solve this system are proposed, among others, by Blanchard and Kahn (1980), Uhlig (1999), and King and Watson (2002). We

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

employ the solution proposed by King and Watson (2002), which rests on the numerically very accurate and stable Schur factorization.

Differentiating the system (7a) twice at $\sigma = 0$ delivers systems of linear equations that can be solved for the coefficients of the quadratic part in the second-order approximation of g^x:

$$\tilde{x} = \tilde{x}^* + g^x_k(\tilde{k}_t - \tilde{k}^*) + g^x_Z(\tilde{Z}_t - \tilde{Z}^*) + g^x_\sigma \sigma + \frac{1}{2}\left[\tilde{k}_t - \tilde{k}^* \;\; \tilde{Z}_t - \tilde{Z}^* \;\; \sigma\right] \begin{bmatrix} g^x_{kk} & g^x_{kZ} & g^x_{k\sigma} \\ g^x_{Zk} & g^x_{ZZ} & g^x_{Z\sigma} \\ g^x_{\sigma k} & g^x_{\sigma Z} & g^x_{\sigma\sigma} \end{bmatrix} \begin{bmatrix} \tilde{k}_t - \tilde{k}^* \\ \tilde{Z}_t - \tilde{Z}^* \\ \sigma \end{bmatrix}. \tag{8}$$

Schmitt-Grohé and Uribe (2004) show that even the coefficients g^x_{kk}, g^x_{ZZ}, and $g^x_{kZ} = g^x_{Zk}$ are independent of σ. Furthermore, they prove that $g^x_{k\sigma} = g^x_{\sigma k} = 0$ and $g^x_{Z\sigma} = g^x_{\sigma Z} = 0$. In general, it is very cumbersome and prone to failure to derive the second-order partial derivatives analytically. Fortunately, there exists software such as the MATLAB toolbox Symbolic Math or Mathematica that is able to execute symbolic differentiation. For our simple model it is not that difficult to employ paper and pencil to find the formulas from which our Fortran program computes the g^x_{ij}.

The *parameterized expectations approach* approximates the rhs of (6d) by a polynomial in $(\ln Z, \ln k)$.[5] We use a simple, complete, exponential polynomial of second degree,[6]

$$\psi(\gamma, \ln Z, \ln k) := \exp(\gamma_1 + \gamma_2 \ln Z + \gamma_3 \ln k + \gamma_4 (\ln z)^2 + \gamma_5 (\ln k)^2 + \gamma_6 \ln z \ln k). \tag{9}$$

The vector of parameters γ is determined as solution to a nonlinear set of equations. This system depends itself on a long series of points $\{Z_t\}_{t=0}^T$, $\{k_t\}_{t=0}^T$, $\{\lambda_t\}_{t=0}^T$, $\{N_t\}_{t=0}^T$, and $\{c_t\}_{t=0}^T$ obtained from iterations over

$$\lambda_t = \psi(\gamma, \ln Z_t, \ln k_t), \tag{10a}$$

$$\frac{\partial u(c_t, N_t)}{\partial c_t} = \lambda_t, \tag{10b}$$

$$\frac{\partial u(c_t, N_t)}{\partial N_t} = \lambda_t \alpha Z_t N_t^{\alpha-1} k_t^{1-\alpha}, \tag{10c}$$

$$a k_{t+1} = Z_t N_t^\alpha k_t^{1-\alpha} + (1-\delta) k_t - c_t, \tag{10d}$$

where Z_t is obtained from (3) using a random number generator that provides pseudo-normally distributed innovations ϵ_t. Given λ_t, equations (10b) and (10c) can be reduced to an equation that determines N_t, given (Z_t, k_t). Given N_t, it is easy to solve for c_t. For initial values of the parameters and the simulated time series, we can compute the least squares of the residuals $R(\ln Z_t, \ln k_t) = c_{t+1}^{-\eta} - \psi(\gamma, \ln Z_t, \ln k_t)$. The parameter vector $\gamma = (\gamma_1, \ldots, \gamma_6)$ is the solution

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

of the following system of nonlinear equations:

$$0 = \frac{-2}{T}\sum_{t=0}^{T-1} R(\ln Z_t, \ln k_t)\frac{\partial \psi(\gamma)}{\partial \gamma_i}, \quad i = 1, 2 \ldots, 6. \tag{11}$$

The crucial step in applying this algorithm is to find acceptable starting values for the nonlinear equations solver. One way to tackle this problem is to obtain linear approximations of the policy functions for consumption and labor supply, to use these to compute time series for λ_t, k_t, and Z_t, and to regress λ_t on $\psi(\ln k_t, \ln Z_t)$ by way of nonlinear least squares.[7] If this works, the parameterized expectations algorithm is a fast way to find the final solution. However, in most cases it does not work. For the German parameter set, we were not able to find admissible starting values for all four specifications of the utility function. For the U.S. parameter set, utility function II posed the same problem. In these cases, we used a genetic search algorithm.[8] To reduce computation time, this algorithm operated over short time series of 5,000 points. When a solution was found, we used it as a starting value for a system of nonlinear equations based on the final number of periods $T = 100,000$. This large number of points guarantees a good sample of the underlying stationary distribution implicitly defined by the system of stochastic difference equations (6), so that the final solution is very accurate.

As in the parameterized expectations approach, the *Galerkin projection method* rests on the approximation of the rhs of (6d). Different from the previous method, however, we use a product base Chebyshev exponential polynomial in $(\ln Z, \ln k)$ as the approximating function,

$$\psi(\ln k_t, \ln Z_t) := \exp\left\{\sum_{i=0}^{p_1}\sum_{j=0}^{p_2}\gamma_{ij} T_i\left[X(\ln Z_t)\right] T_j\left[X(\ln k_t)\right]\right\}, \tag{12}$$

where $T_i(\cdot)$ is the ith degree Chebyshev polynomial and $X(x)$ denotes the linear transformation of $[\underline{x}, \overline{x}]$, $x \in \{\ln Z, \ln k\}$, to $[-1, 1]$, the domain of Chebyshev polynomials. Given $\psi(\cdot)$ and $(\ln k, \ln Z)$, we are able to compute for each innovation ϵ the rhs of (6d): First, we solve (10) for $k' := k_{t+1}$. Letting $z' := \rho \ln Z_t + \epsilon$ and $N' := N_{t+1}$, we use $(\ln k', z')$ and solve (10) again to get

$$g(\epsilon, \ln Z, \ln k) := \psi(\ln k', z')[1 - \delta + (1 - \alpha)e^{z'}(N')^{\alpha}(k')^{-\alpha}].$$

The residual function $R(\gamma, \ln k, \ln Z)$ that is obtained from (6d) by using $\psi(\cdot)$ instead of the true but unknown solution is now given by

$$R(\gamma, \ln k, \ln Z)$$
$$:= \psi(\ln k, \ln Z) - \beta a^{-1}\int_{\ln \underline{Z}}^{\ln \overline{Z}} g(\epsilon, \ln k, \ln Z)(2\pi\sigma^2)^{-1/2} e^{-\epsilon^2/2\sigma^2} d\epsilon.$$

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

The Galerkin projection method chooses γ to solve[9]

$$\int_{\ln \underline{k}}^{\ln \overline{k}} \int_{\ln \underline{Z}}^{\ln \overline{Z}} R(\gamma, \ln k, \ln Z) T_i[X(\ln Z)] T_j[X(\ln k)] d \ln k \; d \ln Z = 0,$$
$$i = 1, 2, \ldots, p_1, \quad j = 1, 2, \ldots, p_2. \tag{13}$$

We use Gauss–Chebyshev quadrature with 50 nodes in each dimension to compute this double integral. The critical step is the choice of the interval $[\ln \underline{k}, \ln \overline{k}]$ for the capital stock so that the algorithm always stays within this interval. We solve this problem in the following way, which, to the best of our knowledge, has not been emphasized in the previous literature on projection methods: We simply use a larger interval that embeds the latter but integrate over the smaller only. More exactly, the conditional expectation is approximated over the interval $[\ln k_{\min}, \ln k_{\max}]$, whereas the Galerkin integral in (13) is computed over $[\ln \underline{k}, \ln \overline{k}] \subset [\ln k_{\min}, \ln k_{\max}]$. Although our program does not find a solution for $\ln \underline{k} = k_{\min}$ and $\ln \overline{k} = \ln k_{\max}$, it converges for $\ln \underline{k} = 1.3 \cdot k_{\min}$ and $\ln \overline{k} = \ln k_{\max}/1.3$. The basic reason for this behavior of the algorithm is that for initial values of γ in the parameterized function, $\ln k_{t+1}$ might happen to fall outside the interval $[\ln k_{\min}, \ln k_{\max}]$ if we chose $[\ln \underline{k}, \ln \overline{k}]$ too wide. In this case, however, we get highly inaccurate solutions for the policy functions. Note, that the accuracy of the solution depends on p_1 and p_2 for given integration bounds in (13) and not on the size of $[\ln k_{\min}, \ln k_{\max}]$. To get comparable results we set $p_1 = p_2 = 3$ so that at most quadratic terms in $\ln k$ and $\ln Z$ appear in the policy functions.

As in the case of the parameterized expectations method, we use the linear approximations of the solution for λ_t to initialize the nonlinear equations solver. This worked in all but two cases. For utility functions I and III of the German parameter set, the linear policy function did not provide a feasible initial parameter vector. Fortunately, in both cases the solution from utility function IV provided admissible starting values.

The *extended deterministic path method* assumes that after a shock in period t no further shock occurs and computes the dynamics for the next T periods. Therefore, it has to solve a set of $2T - 1$ equations in the unknowns $N_{t+s}, k_{t+s+1}, s = 0, 1, \ldots, \mathrm{T} - 1$ obtained from (6a) to (6d), assuming that k_{t+T} is equal to the respective stationary solution of the deterministic counterpart of the model. From this solution only N_t and k_{t+1} are retained. Then another shock is drawn and the respective systems of equations is solved for N_{t+1} and k_{t+2} and so forth. The accuracy of the solution depends upon T. We found that $T = 150$ gives a very accurate solution, yet at the cost of a long computation time.

The *value function iteration method* iterates on

$$v^{s+1}(k_t, Z_t) := \max_{k_{t+1}, N_t} u[Z_t N_t^{\alpha} k_t^{1-\alpha} + (1 - \delta) k_t - a k_{t+1}, N_t]$$
$$+ \beta a^{1-\eta} E_t v^s(k_{t+1}, Z_{t+1}), \tag{14}$$

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

where $v^s(\cdot)$ denotes the value function at iteration s. We use a discrete version of our model to perform these iterations: We approximate the continuous AR(1)-process (3) by a finite Markov chain $\mathscr{Z} = \{z_1, z_2, \ldots, z_m\}$ of m elements and transition matrix $P = (p_{ij})$, where p_{ij} denotes the probability to move from state z_i to state z_j (see Tauchen [1986]). The difference between $\ln z_m$ and $\ln z_1$ is μ times the size of the unconditional standard deviation of (3). We choose μ so that even in simulations with a large number of realizations the productivity shock remains in $[\ln z_1, \ln z_m]$. The capital stock can take values from a grid $\mathscr{K} = \{k_1, k_2, \ldots, k_n\}$ of n elements. The upper (lower) bound k_n (k_1) equals the value that the deterministic counterpart of the model would approach if Z_t would equal z_m (z_1) for all $t = 0, 1, \ldots$. Given the grid $\mathscr{G} = \mathscr{K} \times \mathscr{Z}$, the value function is an $n \times m$-matrix $V = (v_{ij})$, where v_{ij} is the maximum expected lifetime utility obtained from a sequence of optimal choices starting at $(k_i, z_j) \in \mathscr{G}$ at time $t = 0$. We determine the elements of this matrix by iterating over the discrete counterpart of (14):

$$v_{ij}^{s+1} := \max_{k_h \in \mathscr{K}, N} u(z_j N^{\alpha} k_i^{1-\alpha} + (1-\delta)k_i - a k_h, N) + \beta a^{1-\eta} \sum_{l=1}^{m} p_{jl} v_{hl}^{s}. \quad \textbf{(15)}$$

As initial elements of V, we use the value obtained from the balanced growth path of a model with productivity level $z_j \in \mathscr{Z}$ for all $t = 0, 1, \ldots$.

Given a triple (z_j, k_i, k_h), we must solve the first-order condition for optimal labor supply. The respective implicit equations in N_t have unique solutions. We easily computed them with a modified Newton method that takes care of the restriction $N \in [0, 1]$.

The approximate solution of (15) is an $n \times m$ matrix H with the typical element h_{ij} being a pointer to the index h of the capital stock $k_h \in \mathscr{K}$ that is optimal given the current capital stock $k_i \in \mathscr{K}$ and the current productivity level $z_j \in \mathscr{Z}$. Simulations of the model use bilinear interpolation over $\mathscr{G}$ and the associated policy function H to preserve the continuous nature of the productivity shock (3). A more detailed description of our optimization procedure, together with a justification of our approach vis-à-vis others, is presented in the Appendix.[10]

Obviously, the accuracy of the solution as well as the computation time increase with both n and m. With respect to the second moments of simulated time series, we found that n =5,000 and $m = 9$ are a good compromise between speed and accuracy. In terms of Euler equation residuals (see below), more accurate solutions require n =20,000 and $m = 19$.

4. RESULTS

We evaluate the methods with regard to (1) computation time, (2) the usual set of second moments computed from simulated, HP-filtered time series of employment, output, investment, consumption, and the real wage, (3) the risk-free rate of return, and (4) accuracy. We consider the risk-free rate of return in view of the results of Christiano and Fisher (2000), who find that asset price statistics seem to be more

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 1. Model calibration

German calibration		U.S. calibration	
Production	Preferences	Production	Preferences
$a = 1.005$	$\beta = 0.994$	$a = 1.0055$	$\beta = 0.99$
$\alpha = 0.73$	$\eta = 1.00$	$\alpha = 0.64$	$\eta = 1.00$
$\delta = 0.011$	$\nu = 5.00$	$\delta = 0.025$	$\nu = 3.33$
$\rho = 0.90$	$\gamma = 33.5$	$\rho = 0.95$	$\gamma = 7.00$
$\sigma = 0.0072$	$N = 0.13$	$\sigma = 0.00712$	$N = 0.33$

sensitive to the accuracy of the solution. Accuracy is measured using the residual of the Euler equation (6d).

As we are aiming to assess the suitability of the different methods for business cycle models more generally, we analyze two different sets of calibration parameters. The first set reflects parameters commonly applied in the business cycle study of the postwar U.S. economy. The second set of parameters is calibrated with the help of German postwar data prior to unification in 1989.[11] Importantly, these two economies are characterized by different institutional settings. In particular, the German capital market is less competitive, as many banks are state-owned or subsidized by the state. As a consequence, capital depreciates less rapidly in Germany, as capital utilization is lower. Furthermore, capital's share in output, $1 - \alpha$, is lower in Germany (0.27) than in the United States (0.36). One possible reason may be the presence of unions. Second, labor markets are more rigid in Germany and the social security system is more generous. As a consequence, average labor supply is lower in Germany as well.

For the U.S. economy, we use the set of parameters displayed in Table 1. Except for the rate of per capita output growth, they are in accordance with Hansen (1985). The average quarterly growth rate of the U.S. economy exceeds that of the German economy, $a' = 1.0050$, and amounts to $a = 1.0055$ during 1960–2002 on average. The estimates of the Frisch intertemporal labor supply elasticity $\eta_{n,w}$ implied by microeconometric studies and the implied values of γ and ν vary considerably. MaCurdy (1981) and Altonji (1986) both use PSID data to estimate values of 0.23 and 0.28, respectively, whereas Killingsworth (1983) finds a U.S. labor supply elasticity equal to $\eta_{n,w} = 0.4$.[12] We will use the conservative estimate $\eta_{n,w} = 0.3$ and, accordingly, apply the values $\nu = 3.33$ and $\gamma = 7.0$ in utilities III and IV, respectively.[13] For Germany, we use the same set of parameters as in Heer and Maußner (2005).[14] In addition, we use the value $\eta_{n,w} = 0.2$, following Heer and Trede (2003) implying $\nu = 5.0$ and $\gamma = 33.5$.

4.1. Computation Time

The computation time of the solution for the six methods is presented in Table 2. The algorithms can be ordered with respect to running time as follows:

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 2. Computation time

Method	Time
Log-linear approximation (LL)	0.42 s
Second order approximation (SO)	0.52 s
Parameterized expectations (PE)	
—initial value from LL	1 m 21 s
—two steps with search (UII, US)	1 m 10 s
—two steps with search (UII,GE)	23 m 11 s
Galerkin projection (GA)	
—initial value from LL	3 m 44 s
—initial value from utility IV (UI,GE)	2 m 4 s
Extended deterministic path (EP)	2 h 27 m
Value function iteration (VI)	
—n =50,00, $m = 9$	21 m 51 s
—n =20,000, $m = 9$	2 h 47 m
—n =20,000, $m = 19$	4 h 41 m

Notes: If not mentioned otherwise, the results are based on the solution for utility function I and the U.S. parameter set. The program run on a 3.2-Pentium IV personal computer. The program is written in Fortran 95 and compiled with the Compaq Digital Fortran developer environment. Computation time comprises the time needed to compute the solution and to compute the summary statistics from 500 simulations. The stochastic search routine rests on a population of 50 candidate solution vectors and iterates over 100 generations. h = hours, m = minutes, s = seconds.

(1) Log-linear approximation is by far the fastest method. (2) If one disregards the time needed to derive the analytical expressions for the second derivatives by way of paper and pencil, the second-order approximation is almost as fast as log-linearization. The further ranks depend on our measurement of computation time and our strategy to find acceptable starting values for the non-linear equations solver. If the algorithm converges using the initial values obtained from the log-linear solution, (3) the parameterized expectations approach (PE) is much faster than (4) the Galerkin projection method (GA), and both need less time than (5) the extended path (EP) method and (6) the value function iteration method (VI). In the case of the GA it was not necessary to use stochastic search. The search for acceptable starting values for the non-linear equations solver can considerably increase the computation time of the PE solution, namely up to about 24 minutes.[15] The ranking between value function iteration and the extended path method depends upon the desired accuracy. For the accuracy displayed in Table 3 ($n = 20{,}000$ and $m = 19$), EP clearly outperforms VI. Finally, we wish to mention that the solution in the case of utility function II, where an analytical solution for N given k and Z is available, requires substantially less time. For instance, the extended path method finds the solution in about 41 minutes, and value function iteration requires 2 hours and 26 minutes.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 3. Euler equation residuals

Utility function	EP	GA	LL	PE	SO	VI
			U.S. calibration			
Utility I	0.00000	0.00000	0.00109	0.00038	0.00006	0.00038
Utility II	0.00000	0.00000	0.00024	0.00013	0.00005	0.00072
Utility III	0.00000	0.00002	0.00159	0.00041	0.00010	0.00023
Utility IV	0.00000	0.00000	0.00077	0.00033	0.00005	0.00061
			German calibration			
Utility I	0.00000	0.00002	0.00113	0.00020	0.00008	0.00026
Utility II	0.00000	0.00000	0.00036	0.00016	0.00005	0.00091
Utility III	0.00000	0.00003	0.00148	0.00018	0.00014	0.00019
Utility IV	0.00000	0.00000	0.00051	0.00024	0.00005	0.00072

Notes: EP: extended path, GA: Galerkin projection, LL: log-linear approximation, PE: parameterized expectations SO: second-order approximation, VI: value function iteration. Each entry in this table is the maximum absolute value of the Euler equation residuals computed over an equally spaced grid of 400 knots in the interval $[0.8k^*, 1.2k^*] \times [0.95, 1.05]$. Following Christiano and Fisher (2000), we computed the Euler residual as $\tilde{c}/c - 1$, where c is optimal consumption given the policy function and $\tilde{c}$ is the amount of consumption that is necessary to equate the lhs of the Euler equation for capital to its rhs. The conditional expectation on the rhs of the Euler equation for capital is computed using Gauss–Hermite quadrature with four points. We used $n = 20,000$ and $m = 19$ to compute the VI solution.

4.2. Summary Statistics

Our simulation results with regard to the usual business cycle statistics are displayed in Table 4 for the utility functions I–IV. The first column presents the variable; the next three columns display the standard deviation of the variable, the variable's correlation with output, and its autocorrelation, respectively. Except for only a few cases, e.g., the volatility of German investment computed for utility I, the six different methods imply statistics that deviate by less than 0.01. Indeed, in most cases, the minimum and the maximum values obtained from the various methods coincide. Therefore, it is save to conclude that the solution method has no influence on the business cycle statistics of the simulations.

4.3. The Risk-Free Rate of Return

Table 5 provides a measure of the risk-free rate of return, $r_t^f := (\lambda_t / (\beta a^{-\eta} E_t \lambda_{t+1})) - 1$. Because the time series averages are very close to the stationary value $(a^\eta/\beta) - 1$, we do not display the actual time series averages of r_t^f but its percentage deviations from the stationary rate. The averages are computed from time series of 100,000 points. All simulations use the same sequence of random numbers. In all cases, the time series average is slightly smaller than the stationary value. There are small differences between the different utility functions and parameterizations, but no significant differences between the different methods.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 4. Summary business cycle statistics

	s_x			s_{xy}			r_x		
Variable	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
	Utility function I								
	U.S. calibration								
Output	1.26	1.26	1.26	1.00	1.00	1.00	0.66	0.66	0.66
Investment	3.66	3.67	3.67	0.99	0.99	0.99	0.65	0.65	0.65
Consumption	0.43	0.43	0.43	0.90	0.91	0.91	0.76	0.76	0.76
Hours	0.60	0.60	0.60	0.98	0.98	0.98	0.65	0.65	0.65
Real wage	0.69	0.69	0.69	0.98	0.98	0.98	0.69	0.69	0.69
	German calibration								
Output	1.92	1.92	1.92	1.00	1.00	1.00	0.63	0.63	0.64
Investment	8.73	8.74	8.75	1.00	1.00	1.00	0.63	0.63	0.63
Consumption	0.33	0.33	0.33	0.81	0.82	0.82	0.77	0.77	0.77
Hours	1.44	1.45	1.45	0.99	0.99	0.99	0.63	0.63	0.63
Real wage	0.51	0.51	0.52	0.94	0.95	0.95	0.69	0.69	0.69
	Utility function II								
	U.S. calibration								
Output	1.03	1.03	1.03	1.00	1.00	1.00	0.67	0.67	0.67
Investment	2.59	2.59	2.59	0.99	0.99	0.99	0.66	0.66	0.66
Consumption	0.47	0.47	0.48	0.97	0.97	0.97	0.71	0.71	0.71
Hours	0.24	0.24	0.24	1.00	1.00	1.00	0.67	0.67	0.67
Real wage	0.79	0.79	0.79	1.00	1.00	1.00	0.67	0.67	0.67
	German calibration								
Output	0.99	0.99	0.99	1.00	1.00	1.00	0.64	0.64	0.64
Investment	3.89	3.89	3.89	1.00	1.00	1.00	0.63	0.63	0.64
Consumption	0.30	0.30	0.31	0.96	0.96	0.96	0.68	0.68	0.68
Hours	0.17	0.17	0.17	1.00	1.00	1.00	0.64	0.64	0.64
Real wage	0.83	0.83	0.83	1.00	1.00	1.00	0.64	0.64	0.64
	Utility function III								
	U.S. calibration								
Output	1.66	1.66	1.66	1.00	1.00	1.00	0.66	0.66	0.66
Investment	4.94	4.95	4.96	0.99	0.99	0.99	0.65	0.65	0.65
Consumption	0.52	0.52	0.52	0.88	0.89	0.89	0.77	0.77	0.77
Hours	1.22	1.22	1.23	0.98	0.98	0.98	0.64	0.64	0.64
Real wage	0.52	0.52	0.52	0.88	0.89	0.89	0.77	0.77	0.77

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 4. Continued

	s_x			s_{xy}			r_x		
Variable	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
				German calibration					
Output	2.39	2.39	2.39	1.00	1.00	1.00	0.63	0.63	0.63
Investment	10.98	10.99	11.01	1.00	1.00	1.00	0.63	0.63	0.63
Consumption	0.39	0.39	0.39	0.80	0.80	0.80	0.78	0.78	0.78
Hours	2.09	2.09	2.10	0.99	0.99	0.99	0.63	0.63	0.63
Real wage	0.39	0.39	0.39	0.80	0.80	0.80	0.78	0.78	0.78
	Utility function IV								
				U.S. calibration					
Output	0.98	0.98	0.98	1.00	1.00	1.00	0.67	0.67	0.67
Investment	2.76	2.76	2.76	0.99	0.99	0.99	0.65	0.65	0.65
Consumption	0.35	0.35	0.35	0.92	0.92	0.92	0.75	0.75	0.75
Hours	0.15	0.15	0.15	0.98	0.98	0.98	0.65	0.65	0.65
Real wage	0.83	0.83	0.83	1.00	1.00	1.00	0.67	0.67	0.67
				German calibration					
Output	0.97	0.97	0.97	1.00	1.00	1.00	0.64	0.64	0.64
Investment	4.26	4.26	4.26	1.00	1.00	1.00	0.63	0.63	0.63
Consumption	0.20	0.20	0.20	0.86	0.87	0.87	0.74	0.75	0.75
Hours	0.13	0.13	0.13	0.99	0.99	0.99	0.63	0.63	0.63
Real wage	0.84	0.84	0.84	1.00	1.00	1.00	0.64	0.64	0.64

Notes: s_x is the standard deviation of variable x listed in the first column. s_{xy} is the correlation of x with output. s_x denotes the first-order autocorrelation of x. All moments are averages over 500 simulations. The length of the respective time series was 60 periods in each simulation. All time series are HP-filtered with weight $\lambda = 1600$. The same 500×60 random numbers were used in all simulations. The columns labeled "Min", "Mean", and "Max" display the minimum, the mean, and the maximum from the six different methods, respectively.

4.4. Accuracy

The residual e of the Euler equation for capital is computed as $e = (\tilde{c}/c) - 1$, where c is consumption. $\tilde{c}$ is the amount of consumption that is necessary to equate the lhs of equation (6d) to our approximation of $\beta a^{-\eta} E_t u_c(c', 1-n')(1-\delta+(1-\alpha)Z'(N')^{\alpha}(k')^{-\alpha})$, where the prime denotes next-period values of consumption, hours, and capital, respectively. We use four-point Gauss–Hermite quadrature to compute the conditional expectation on the rhs of equation (6d) given the policy function delivered by the respective solution method. We compute the residuals for 400 equally spaced pairs of $(k, z) \in [0.8k^*, 1.2k^*] \times [0.95, 1.05]$. We choose this subset because it is the domain of simulated time series, even for very long ones with 100,000 points.

Table 3 displays the maximum absolute value of the 400 residuals.[16] Obviously, the extended path method (EP) provides the most accurate results. The Galerkin projection method (GA) is almost as precise as EP, followed by second-order approximation (SO). The Euler equation residuals from SO are even 1.4 to 18 times

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 5. Risk-free rate of return

Utility function	EP	GA	LL	PE	SO	VI
			U.S. calibration			
Utility I	−0.027	−0.027	−0.027	−0.027	−0.027	−0.027
Utility II	−0.018	−0.018	−0.018	−0.017	−0.018	−0.018
Utility III	−0.038	−0.038	−0.038	−0.038	−0.038	−0.038
Utility IV	−0.019	−0.019	−0.019	−0.019	−0.019	−0.018
			German calibration			
Utility I	−0.023	−0.023	−0.022	−0.023	−0.023	−0.024
Utility II	−0.009	−0.009	−0.009	−0.009	−0.009	−0.004
Utility III	−0.030	−0.030	−0.030	−0.030	−0.030	−0.032
Utility IV	−0.010	−0.010	−0.010	−0.010	−0.010	−0.012

Notes: EP: extended path, GA: Galerkin projection, LL: log-linear approximation, PE: parameterized expectations SO: second-order approximation, VI: value function iteration. The risk-free rate of return is the average $\bar{r}^f$ of $r_t^f := \lambda_t/(\beta a^{-\eta} E_t \lambda_{t+1}) - 1$ from a time series with 100,000 points. The conditional expectation in the definition of the risk-free rate is computed using Gauss–Hermite quadrature with four points. The risk-free rate of return in the stationary equilibrium is given by $r^f = a^\eta/\beta - 1 = 0.0156566$ for the U.S. calibration and by $r^f = a^\eta/\beta - 1 = 0.0110664$ for the German calibration. The entries in the table are $100 \times ((\bar{r}^f/r^f) - 1)$. The same sequence of random numbers has been applied to all computations.

smaller than those obtained from the value function iteration (VI), even though we used $20{,}000 \times 19$ in the latter case. Similarly, EP, Galerkin projection, and second-order approximation clearly outperform log-linearization in terms of accuracy by one or two digits at least. Notice that the method of parameterized expectations is also less accurate than second-order approximation, and, in particular, Galerkin projection. The better performance of the GA vis-à-vis the PE does not depend upon our choice of the respective polynomials. We computed the PE solution using the Chebyshev polynomial (12). The results, displayed in Table A.1 in the Appendix, show that for the same degrees p_1 and p_2, the GA solution always provides smaller Euler residuals than the PE solution.

Figures 1 and 2 provide a second view of the different degrees of accuracy. They plot the policy function for consumption and working hours in the case of utility function I and the US parameter set at $Z \equiv 1$. The capital stock, consumption, and hours are measured relative to their respective stationary values. The poor performance of the log-linear solution can be seen from the upper left panel of Figure 1. Consumption is only close to the solution obtained from the extended path method near the stationary stock of capital. The same is true for working hours (see the upper left panel in Figure 2). For the remaining four methods the respective policy function for consumption is visually indiscernible from the EP solution. Figure 2 also shows that the PE policy function for hours deviates from the EP solution for small and large values of the stock of capital k_t. Because the parameterized expectations approach relies on Monte Carlo integration, it delivers inaccurate results for extreme values of k_t that are unlikely to occur.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

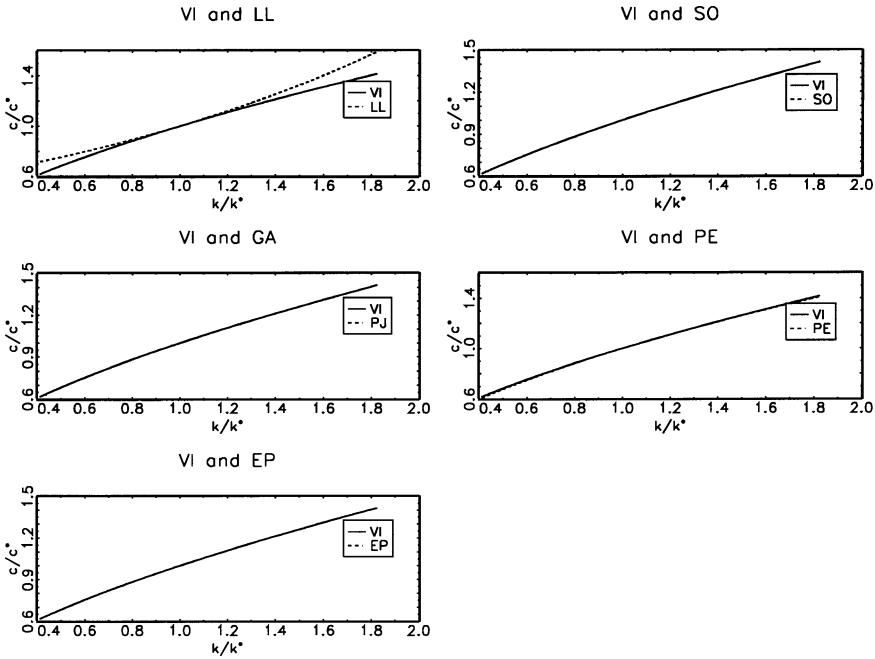

FIGURE 1. Policy function for consumption.

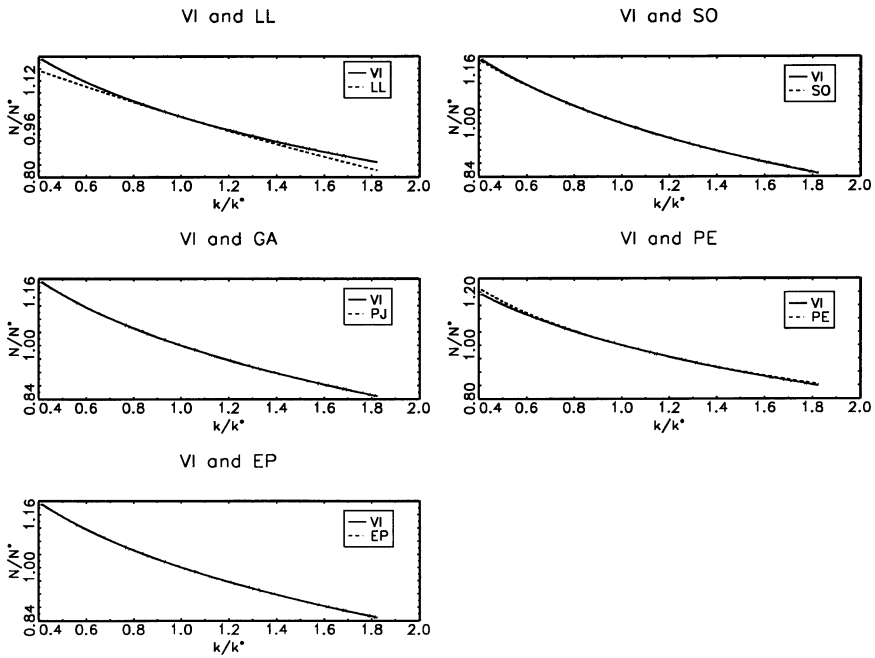

FIGURE 2. Policy function for hours.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 6. Summary statistics: Utility I, U.S. calibration, $\sigma = 0.0712$

	s_x			s_{xy}			r_x		
Variable	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
Output	12.63	13.45	13.71	1.00	1.00	1.00	0.66	0.66	0.66
Investment	36.66	39.35	40.17	0.99	0.99	0.99	0.64	0.65	0.65
Consumption	4.26	4.44	4.50	0.90	0.90	0.91	0.76	0.76	0.76
Hours	5.98	5.99	6.00	0.96	0.96	0.98	0.65	0.65	0.65
Real wage	6.86	7.24	7.36	0.98	0.98	0.98	0.69	0.69	0.69

Notes: See Table 4.

4.5. Sensitivity with Respect to σ

The log-linear approximations of the policy functions are independent of the standard deviation of the productivity shock σ. The coefficient on σ^2 in the second-order approximation (8) reflects only information about the stationary solution. The remaining four methods incorporate the effect of σ on the policy function via the larger range of realizations of the shock and the associated larger intervals within which the model's variables reside. To check the sensitivity of our results with respect to the size of σ, we also consider the solutions of our model for utility function I and the U.S. parameter set for $\sigma = 0.0712$, which amounts to 10 times the size of the standard deviation of the log of the U.S.–Solow residual commonly applied (see Table 1). In order to get results that are comparable to those in Table 4 we do not change the parameters of the methods that determine their respective accuracy. For instance, we use the same degree of the polynomials that approximate the rhs of the Euler equation for capital (6d) and the same number of grid-points in the value function iteration.

Table 6 reveals that our previous conclusion with respect to the sensitivity of the quantity allocations still holds: the business cycle summary statistics are hardly affected by the choice of the method. In particular, the correlation of the variables with output, s_{xy}, and the autocorrelation of variables, r_x, do not depend on the method. Only the volatilities of output, consumption, investment, and real wages are affected. However, the deviations are quantitatively small and only average around 1% between the different methods. In fact, in our 500 simulations of the economy for each method, we observe much larger deviations of the statistics on average. Notice further that the relative ordering and magnitudes of the volatilities are hardly affected. Thus, even with an unrealistically volatile productivity shock, the researcher does not arrive at different conclusions about the nature of the business cycle if he/she arbitrarily uses any of the six methods.

Table 7 shows that the ranking of the methods with respect to running time also remains unchanged. Value function iteration is still the most time-consuming method, due to the large number of grid-points. Because we were unable to find a solution with the parameterized expectations method when we used the solution

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE 7. Selected statistics $\sigma = 0.0712$, utility I, U.S. calibration

Method	Run time	Euler equation residual	Risk-free rate of return
LA	0.48 s	0.1841209	−2.310
SO	0.58 s	0.1787492	−3.175
PE	44 m 7.19 s	0.0053931	−2.753
GA	3 m 4.95 s	0.0055251	−2.691
EP	3 h 37m 44.81	0.0125591	−3.523
VI	5 h 33 m 19.09 s	0.0096766	−2.950

Notes: See Table 2 for the abbreviation of methods, Table 3 for the definition of the Euler equation residual, and Table 5 for the definition and computation of the risk-free rate of return. See also Table 2 for the definition of run time. The Euler equation residuals are the maximum absolute value from a grid of 2,500 equally spaced points over the square $[0.6k^*, 2k^*] \times [0.7, 1.5]$. About 90% of the points of a long time series with 100,000 elements belong to this interval.

from the log-linear method to initialize the parameter vector, we had to use our genetic search algorithm instead. Therefore, the PE method is slow and the running time amounts to 44 minutes. The Galerkin projection algorithm, on the other hand, converges from the log-linear solution within about 3 minutes.

The high volatility of the productivity shock entails a large support of the stationary distribution of (k_t, Z_t). For this reason, we computed Euler equation residuals on 2,500 grid-points over the square $[0.6k^*, 2k^*] \times [0.7, 1.5]$. Because we have not taken any measures to increase the accuracy of our methods it comes as no surprise that all Euler equation residuals are about a factor of 10 higher than those reported in Table 3. Still, EP and GA are the most accurate methods, where GA now outperforms EP. Second-order approximation, being a local method, is now dominated by the PE. This method also provides smaller errors than value function iteration.

The deviation of the mean risk-free rate of return from its stationary solution value is displayed in column four of Table 7. As in our benchmark calibration for the U.S. economy, there are no remarkable differences over the various methods.

5. CONCLUSION

This paper has shown that several numerical methods can be applied to study the standard business cycle model. Using either log-linearization, second-order approximation, parameterized expectations, Galerkin projection, deterministic extended path, or value function iteration basically results in the same values for the second moments of the variables that the business cycle researcher is most interested in, that is, output, employment, investment, consumption, and wages. Log-linearization, of course, is very easy to implement and by far the fastest method. Furthermore, the solution from this method can often be successfully applied as an initial value for more sophisticated nonlinear methods such as parameterized expectations or Galerkin projection, where the computation of a good

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

initial value with genetic search algorithm or homotopy methods may become very time-consuming, as may be the case in more complex multidimensional state-space applications. Our results, therefore, suggest that the researcher may benefit from using log-linearization methods in the first place and, possibly, also should use nonlinear methods such as extended path or projection methods to improve the accuracy of the computation in more nonlinear problems that may arise, for example, in the presence of binding constraints, e.g., a nonnegativity constraint on investment or a constraint on the maximum number of working hours, more curved utility functions, or models with sizable shocks.

Our work has also emphasized an important detail in the application of projection methods for the approximation of polynomial solution functions in more complex dynamic models. In our example of the stochastic growth model with flexible labor supply, standard projection methods failed to converge even for good initial values that were computed with the help of the log-linear solution. We find that the basic reason for this observation is the poor approximation of functions with Chebyshev polynomials outside the approximation interval. As a solution to this problem, we suggest the use of a wider interval for the approximation of the function than for the integration over the residual function.

NOTES

1. In related, but independent research, Aruoba et al. (2006) also compare value function iteration with higher-order projection methods and fifth-order perturbation methods.

2. Function I is the standard functional type. Function III models indivisible labor as in Hansen (1985) and is able to explain the fact that total hours and the number of employed workers is much more variable than individual working hours. Function IV, finally, is used by Castañeda et al. (2003) in their work on heterogeneous-agent economies. For this type of utility function, working hours vary less with individual productivity and are in better accordance with empirical observations.

3. This functional form is suggested by Greenwood et al. (1998) and has the attractive feature that there is no wealth effect on the labor supply decision. Hence, richer households do not supply less labor ceteris paribus.

4. Except for the second-order approximation, a detailed description of these methods is provided by Heer and Maußner (2005). The Fortran program that performs all the computations referred to in the text can be downloaded from Alfred Maußner's homepage, http://www.wiwi.uni-augsburg.de/vwl/maussner/. It allows the user to supply his or her own parameter set and has a lot of options for the computation of solutions and test statistics.

5. The method of parameterized expectations can be interpreted as a special case of the more general class of projections methods, as pointed out by Judd (1996). Furthermore, he emphasizes that the approach was originally developed by Williams and Wright (1982) and Wright and Williams (1984, 1991).

6. See, for example, den Haan and Marcet (1994).

7. Because the error term in $\lambda_t - \psi(\ln k_t, \ln Z_t)$ is additive, it is inappropriate to regress $\ln \lambda_t$ on $\ln k_t$ and $\ln Z_t$ by means of ordinary least squares.

8. In particular, we applied two different specifications of the genetic search algorithm. In our first specification, we follow Duffy and McNelis (2001). Yet, differently from their paper, our fitness criterion is the minimal absolute value of the rhs of (11). Second, we use a different selection scheme and employ a larger set of crossover operators than Duffy and McNelis (2001). In particular, we use stochastic universal sampling, as in Mitchell (1996). The genetic search algorithms are described in more detail in Heer and Maußner (2005). Neither of the two algorithms is

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

found to dominate the other in terms of speed across all calibrations and parameterizations in our model.

9. For a more detailed description of the projection methods, see Judd (1992, 1998) and McGrattan (1999).

10. Our method exploits the monotonicity of the policy function and the concavity of the value function. We find this algorithm to be much more accurate for given computation time than Howard's algorithm or a variant that uses interpolation between grid points. See Ljungqvist and Sargent (2000), Chapters 2 and 3, on Howard's improvement algorithm.

11. A detailed description of this calibration is provided in Chapter 1 of Heer and Maußner (2005).

12. Domeij and Floden (2006) argue that these estimates are biased downward due to the omission of borrowing constraints.

13. Greenwood et al. (1988) even apply a value $\eta_{n,w} = 1.7$ corresponding to $\nu = 0.6$ in their study, while Castañeda et al. (2003) use $\gamma := 5.5$.

14. In particular, we did not find any compelling evidence that the intertemporal elasticity of substitution is different between the United States and Germany.

15. Computation time can exceed 1 hour if the search process is applied to the final number of periods $T = 100{,}000$. We, however, used a two-step procedure: find admissible starting values for a small number of periods $T = 5{,}000$ and use these to compute the solution for the final number of periods $T = 100{,}000$.

16. See Figure A.1 in the Appendix for a plot of these residuals.

17. Note that N is a function of k_h.

18. It seems that the literature has not paid ample attention to this point. See, for example, the description of value function iteration on p. 41 of Ljungqvist and Sargent (2000).

19. k^* is the stationary stock of capital obtained from the deterministic version of the model. We compute the Euler equation residual using Gauss–Hermite quadrature with four nodes and interpolate bilinearly between the points of $\mathscr{G}$. Following Christiano and Fisher (2000), we compute the Euler residual as $\tilde{c}/c - 1$, where c is optimal consumption given the policy function. $\tilde{c}$ is the amount of consumption that is necessary to equate the lhs of the Euler equation for capital to its rhs.

REFERENCES

Altonji, J. G. (1986) Intertemporal substitution in labor supply: Evidence from micro data, *Journal of Political Economics* 94, S117–S215.

Aruoba, S. B., J. Fernández-Villaverde, and J. F. Rubrio-Ramírez (2006) Comparing solution methods for dynamic equilibrium economies, *Journal of Economic Dynamics and Control* 30, 2477–2508.

Blanchard, O. J. and Ch. M. Kahn (1980) The solution of linear difference models under rational expectations, *Econometrica* 48, 1305–1311.

Castañeda, A., J. Díaz-Giminénez, and J.-V. Ríos-Rull (2003) Accounting for the U.S. earnings and wealth inequality, *Journal of Political Economy* 111, 818–857.

Christiano, L. J. and J. D. M. Fisher (2000) Algorithms for solving dynamic models with occasionally binding constraints, *Journal of Economic Dynamics and Control* 24, 1179–1232.

Den Haan, W. J. and A. Marcet (1994) Accuracy in simulations, *Review of Economic Studies* 61, 3–17.

Domeij, D. and M. Floden (2006) The labor supply elasticity and borrowing constraints: Why estimates are biased, *Review of Economic Dynamics* 9, 242–262.

Duffy, J. and P. D. McNelis (2001) Approximating and simulating the stochastic growth model: Parameterized expectations, neural networks, and the genetic algorithm, *Journal of Economic Dynamics and Control* 25, 1273–1303.

Greenwood, J., Z. Hercowitz, and G. W. Huffman (1988) Investment, capacity utilization, and the real business cycle, *American Economic Review* 78, 402–417.

Hansen, G. (1985) Indivisible labor and the business cycle, *Journal of Monetary Economics* 16, 309–327.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

Heer, B. and A. Maußner (2005) *Dynamic General Equilibrium Modelling: Computational Methods and Applications*. Berlin: Springer-Verlag.
Heer, B. and M. Trede (2003) Efficiency and distribution effects of a revenue-neutral income tax reform, *Journal of Macroeconomics* 25, 87–107.
Judd, K. L. (1992) Projection methods for aggregate growth models, *Journal of Economic Theory* 58, 410–452.
Judd, K. L. (1996) Approximation, perturbation, and projection methods in economic analysis. In H. M. Amman, D. A. Kendrick, and J. Rust (eds.), *Handbook of Computational Economics*, vol. I, pp. 509–586. Amsterdam: North-Holland.
Judd, K. L. (1998) *Numerical Methods in Economics*. Cambridge, MA: MIT Press.
Killingsworth, M. R. (1983) *Labor Supply*. Cambridge, UK: Cambridge University Press.
King, R. G., Ch. I. Plosser, and S. Rebelo (1988) Production, growth and business cycles I: The basic neoclassical model, *Journal of Monetary Economics* 21, 195–232.
King, R. G. and M. W. Watson (2002) System reduction and solution algorithms for singular linear difference systems under rational expectations, *Computational Economics* 20, 57–86.
Kremer, J. (2001) *Arbeitslosigkeit, Lohndifferenzierung und wirtschaftliche Entwicklung*. Cologne: Josef Eul Verlag.
Ljungqvist, L. and T. J. Sargent (2000) *Recursive Macroeconomic Theory*. Cambridge, MA/London: MIT Press.
MaCurdy, T. E. (1981) An empirical model of labor supply in a life-cycle setting, *Journal of Political Economy* 89, 1059–1085.
McGrattan, E. R. (1999) Application of weighted residual methods to dynamic economic models. In R. Marimon and A. Scott (eds.), *Computational Methods for the Study of Dynamic Economies*, pp. 114–142. Oxford New York: Oxford University Press.
Mitchell, M. (1996) *An Introduction to Genetic Algorithms*. Cambridge, MA: Harvard University Press.
Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery (1992) *Numerical Recipes in Fortran* 77. Cambridge, UK: Cambridge University Press.
Schmitt-Grohé, S. and M. Uribe (2004) Solving dynamic general equilibrium models using a second-order approximation to the policy function, *Journal of Economic Dynamics and Control* 28, 755–775.
Sefton, J. A. (2000) A solution method for consumption decisions in a dynamic stochastic general equilibrium model, *Journal of Economic Dynamics and Control* 24, 1097–1119.
Tauchen, G. (1986) Finite state markov-chain approximations to univariate and vector autoregressions, *Economics Letters* 20, 177–181.
Taylor, J. B. and H. Uhlig (1990) Solving nonlinear stochastic growth models: A comparison of alternative solution methods, *Journal of Business and Economic Statistics* 8, 1–17.
Uhlig, H. (1999) A toolkit for analysing nonlinear dynamic stochastic models easily. In Ramon Marimon and Andrew Scott (eds.), *Computational Methods for the Study of Dynamic Economies*, pp. 30–61. Oxford: Oxford University Press.
Williams, J. C. and B. D. Wright (1982) The economic role of commodity storage, *Economic Journal* 92, 596–614.
Wright, B. D. and J. C. Williams (1984) The welfare effects of the introduction of storage, *Quarterly Journal of Economics* 99, 169–182.
Wright, B. D. and J. C. Williams (1991) *Storage and Commodity Markets*. Cambridge, UK: Cambridge University Press.

APPENDIX

A.1. VALUE FUNCTION ITERATION

Researchers who use value function iteration on a two-dimensional grid over the stock of capital and the realizations of the productivity shock face a trade-off between accuracy

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

of the solution and computation speed. One way to alleviate this trade-off is interpolation between grid points (see, e.g., Sefton [2000]). This section motivates our procedure and provides the reasons that we think that it is superior to interpolation.

Consider the two-dimensional grid $\mathscr{G} := \mathscr{K} \times \mathscr{Z}$, over the stock of capital k, $\mathscr{K} = \{k_1, k_2, \ldots, k_n\}$, and the productivity shock z, $\mathscr{Z} = \{z_1, z_2, \ldots, z_m\}$. Let $P = (p_{jl})$ denote the transition matrix between states z_j and z_l that is associated with the Markov chain approximation of the productivity shock on $\mathscr{Z}$, as suggested by Tauchen (1986). The $(n \times m)$-matrix $V^s = (v_{ij}^s)$ stores the value function at iteration step s. Given the current state $(k_i, z_j) \in \mathscr{G}$, this allows us to write[17]

$$v_{ij}^{s+1} = \max_{k_h \in \mathscr{K},\, N} \quad \phi(k_h) := u(z_j N^{\alpha} k_i^{1-\alpha} + (1-\delta)k_i - k_h, N) + \beta a^{-\eta} \sum_{l=1}^{m} p_{jl} v_{hl}^s. \quad \textbf{(A.1)}$$

To find the index of the maximal element k_h, we do not need to evaluate the expression on the rhs of this equation for all n elements of $\mathscr{K}$.[18] Because the policy function is monotonically increasing in k_i, we can start with the index of k found optimal for k_{i-1}. Furthermore, because $\phi(k_h)$ is strictly concave, we are able to use a binary search algorithm that locates the maximal element after at most $\log_2 n$ evaluations of $\phi(k_h)$ [see Kremer, (2001, p. 166) for a proof]. This is a considerable reduction of function evaluations. For instance, for n =1,000 elements, we are done after 13 calls to the routine that returns $\phi(k_h)$ at the most.

If we do not want to use the Markov chain approximation of the productivity shock, equation (A.1) becomes

$$v_{ij}^{s+1} = \max_{k_h \in \mathscr{K},\, N} u(z_j N^{\alpha} k_i^{1-\alpha} + (1-\delta)k_i - k_h, N) + \beta a^{-\eta} \int_{-\infty}^{\infty} v(k_h, \rho z_j + \epsilon)(2\pi\sigma^2)^{-1/2} e^{-(1/2\sigma^2)\epsilon^2} d\epsilon. \quad \textbf{(A.2)}$$

Gauss–Hermite quadrature is an efficient and accurate method for approximating the integral on the rhs of this equation. In our program we use four nodes. For this method to be applicable, we must interpolate the value function between the points of $\mathscr{Z}$. We employ a cubic spline. For this reason, each time we call the routine that evaluates the right-hand side of the max-operator we have to solve an m-dimensional linear tri-diagonal system of equations [see, e.g., Press et al. (1992, pp. 107–110)]. Not only does the interpolation step increase the computational time considerably, but also so does the Gauss–Hermite integration decrease accuracy. Because the interval $[z_1, z_m]$ must contain the nodes of the Gauss–Hermite quadrature, we have to pick a rather large interval. With four nodes, its size is roughly 21 times the unconditional standard deviation of the AR(1)-process (3). Even for simulations with as many as 100,000 realizations, the productivity shock stays in a much smaller interval of about nine times the size of the unconditional standard deviation of (3). Because the bounds of the grid for k depend on z_1 and z_m, we must also increase the interval $[k_1, k_n]$ to ensure that the policy function does not hit the boundary of $\mathscr{K}$. Given the same number of grid points as used in our preferred method, the larger bounds introduce greater imprecision and the interpolation slows down the program. For instance, with $n = 10,000$, $m = 31$, and $z_m - z_1 = 9\sigma/\sqrt{1-\varrho^2}$), our method needs about 4 hours and 18 minutes to compute the policy function on a 3.2 GH Pentium 4 desktop computer. The accuracy of the solution as given by the maximum absolute value of the Euler equation residual computed over 400 equally spaced points in $[0.8k^*, 1.2k^*] \times [0.95, 1.05]$ amounts to 0.0002418.[19] When we use the interpolation method with Gauss–Hermite quadrature, we must decrease

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

TABLE A.1. Selected statistics GA versus PE

U.S. calibration, $p_1 = 2$, $p_2 = 3$				
	Euler equation residual		Risk free rate of return	
Utility	GA	PE	GA	PE
Utility I	0.00004	0.00038	−0.028	−0.028
Utility II	0.00003	0.00013	−0.018	−0.018
Utility III	0.00008	0.00068	−0.039	−0.039
Utility IV	0.00004	0.00027	−0.020	−0.020
U.S. calibration, $p_1 = 3$, $p_2 = 3$				
Utility I	0.00000	0.00022	−0.027	−0.028
Utility II	0.00000	0.00014	−0.018	−0.018
Utility III	0.00002	0.00024	−0.038	−0.039
Utility IV	0.00000	0.00022	−0.019	−0.020

Notes: See Table 2 for the abbreviation of methods, Table 3 for the definition of the Euler equation residual, and Table 5 for the definition and computation of the risk-free rate of return.

the number of grid points from $m = 31$ to $m = 15$ so that the program requires about the same time to compute the policy function (4 hours and 55 minutes). The respective Euler equation residual is 0.0009473. Thus, our method is faster and much more accurate.

A.2. FURTHER RESULTS

Table A.1 presents Euler equation residuals and risk-free rates of return for the GA and PE solution when we use Chebyshev polynomials of degree p_1 and p_2 in both methods.

Figure A.1 displays plots of the Euler equation residuals for utility function I and the U.S. parameter set. Note that unlike the residuals obtained from an econometric regression, Euler equation residuals must neither sum to zero nor be scattered arbitrarily around zero. Euler equation residuals indicate the deviation of the approximate policy function from the true one. In most cases this deviation is systematic, as can be seen most easily in the case of the deterministic growth model with exogenous labor supply $N = 1$. In this model, the Euler equation is given by

$$\left[\frac{C(K')}{C(K)}\right]^{\eta} = \beta\left[1 - \delta + \alpha(K')^{\alpha-1}\right], \quad K' = K^{\alpha} + (1-\delta)K - C(K),$$

where $C(K)$ is the true policy function for consumption. The linear approximation of this function $\hat{C}(K)$ is tangent to $C(K)$ at the stationary solution K^* but lies everywhere else above $C(K)$. Thus $\hat{C}(K) \geq C(K)$ and, therefore, $\hat{K}' = K^{\alpha} + (1-\delta)K - \hat{C}(K) < K'$. Because the policy function is monotonically increasing in K, $C'(K') \leq \hat{C}(\hat{K}')$. Thus

$$\left[\frac{\hat{C}(\hat{K}')}{\hat{C}(K)}\right]^{\eta} - \beta\left[1 - \delta + \alpha(\hat{K}')^{\alpha-1}\right] \leq 0$$

and the Euler equation residuals are nonpositive.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363

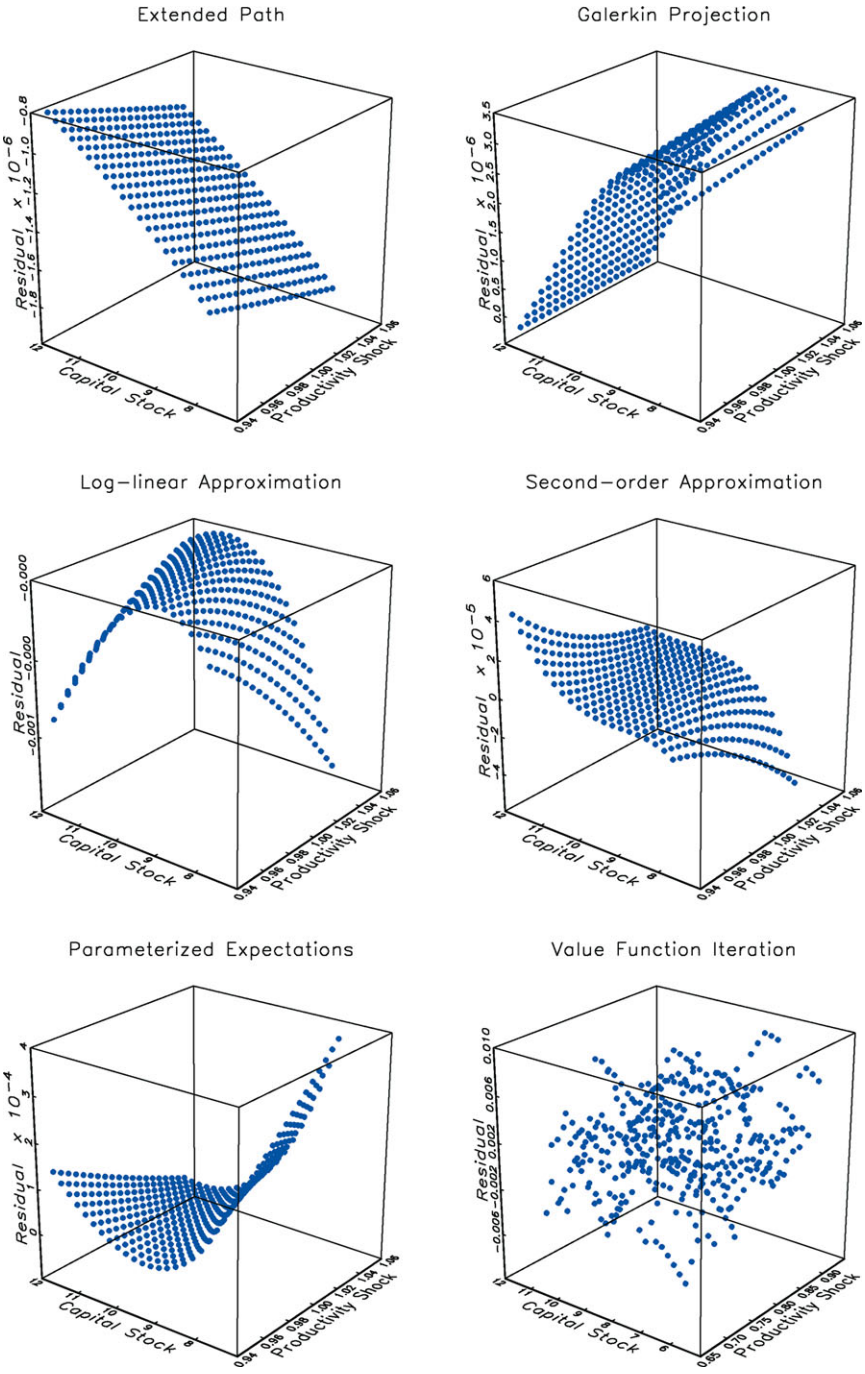

FIGURE A.1. Euler equation residuals.

Downloaded from https://www.cambridge.org/core. Universitaet Augsburg, on 22 Aug 2017 at 06:54:34, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S1365100508070363