

# A Review of Solution and Estimation Methods for Nonlinear DSGE Models with the Zero Lower Bound\*

Yasuo Hirose<sup>†</sup>      Takeki Sunakawa<sup>‡</sup>

December 23, 2018

## Abstract

We review solution and estimation methods for nonlinear dynamic stochastic general equilibrium models and their application, with a special focus on the zero lower bound on the nominal interest rate. In a fully nonlinear setting, both the solution and estimation methods implement iterative procedures, and their computational expense grows rapidly with the dimensionality of state variables and parameters. We describe how the procedures deal with the dimensionality problem.

*JEL Classification:* C32, C63, E32, E52

*Keywords:* Nonlinearity, DSGE models, Zero lower bound, Time iteration method, Parameterized expectations algorithm, Smolyak algorithm, Simulation-based grid, Particle filter, Central difference Kalman filter, Markov chain Monte Carlo, Sequential Monte Carlo

## 1 Introduction

Dynamic stochastic general equilibrium (DSGE) models have been one of the primary tools in macroeconomic analysis, with many economists reaching the stage of estimating DSGE models following the development of Bayesian estimation and evaluation techniques. During much of the same recent period, the zero lower bound (ZLB) on nominal interest rates has been a primary concern for policy makers, and much work has been devoted to incorporating nonlinearities arising from the bound into DSGE models. In this review, we appraise the solution and estimation methods for nonlinear DSGE models and related applications with a special focus on the treatment of the ZLB.

A DSGE model typically consists of optimality conditions for households and firms, policy reaction functions, and market clearing conditions. While these equilibrium conditions are inherently nonlinear, nonlinearities often remain small in many applications. For this reason, most of the existing work linearly approximates the equilibrium conditions. The rational expectations solution, also

---

\*We would like to thank Fumio Hayashi for inviting us to write this review and providing us with constructive suggestions. We have also benefited from Toni Braun's comments on the draft of this review. Hirose is supported by research grants from Japan Center for Economic Research and Nomura Foundation. Sunakawa is supported by JSPS Grant-in-Aid for Scientific Research (KAKENHI) for Young Scientists, Project Number 18K12743. All remaining errors are our own.

<sup>†</sup>Faculty of Economics, Keio University. Address: 2-15-45 Mita, Minato-ku, Tokyo 108-8345 Japan. Phone: +81-3-5418-6459. E-mail: [yhirose@econ.keio.ac.jp](mailto:yhirose@econ.keio.ac.jp)

<sup>‡</sup>Center for Social Systems Innovation, Kobe University. Address: 2-1 Rokkodai-cho Nada-ku, Kobe 657-8501 Japan. Phone: +81-78-803-6853. E-mail: [takeki.sunakawa@gmail.com](mailto:takeki.sunakawa@gmail.com)

referred to as the policy function, can then be obtained using a standard solution algorithm, such as those developed by [Blanchard and Kahn \(1980\)](#), [Klein \(2000\)](#), or [Sims \(2002\)](#). Moreover, with the assumption of the normality of exogenous shocks, a linear state-space representation enables us to exactly evaluate the likelihood function using the Kalman filter. Given a set of parameters, both the solution of a model and the evaluation of the likelihood can be done within seconds, even if the model is sufficiently large in terms of the numbers of variables and parameters. These efficient algorithms make it even easier to solve and estimate DSGE models in a linear setting.

However, linearized DSGE models are unable to deal with features that generate pronounced nonlinearities like the occasionally binding ZLB constraint. To take account of such nonlinearities in DSGE models, we need to rely on nonlinear solution and estimation techniques. It is especially a very challenging task to solve and estimate DSGE models in a fully nonlinear setting because of the heavy computational burden. This is because both the solution and estimation techniques are based on iterative procedures, and their computational expense grows rapidly with an increase in the dimensionality of state variables in a model and the parameters to be estimated. A primary objective of this review is to present recent developments in nonlinear solution and estimation techniques that deal with this dimensionality problem and their applications to DSGE models with the ZLB. In order for this review to be self-contained, we also provide the basic ideas and algorithms underpinning recently developed techniques.

Regarding nonlinear solution methods, some nonlinearities are captured by perturbation methods that locally approximate an equilibrium solution up to second or third order as in [Schmitt-Grohe and Uribe \(2004\)](#) or [Andreasen et al. \(2018\)](#). However, the ZLB makes a monetary policy reaction function for the nominal interest rate kinked at zero and this induces strong asymmetry in a model. In such a situation, the perturbation methods cannot well approximate an equilibrium solution. Thus, in what follows, we focus on fully nonlinear global solution methods as employed in the literature on DSGE models with the ZLB.

This is not the first review to assess nonlinear solution and estimation methods for DSGE models. Notably, [Fernández-Villaverde et al. \(2016\)](#) provide an extensive review of the solution and estimation methods for DSGE models. They present both perturbation and global solution methods and review a variety of frequentist estimation techniques and limited-information approaches as well as full-information Bayesian estimation techniques. We differentiate our review from theirs by focusing on the treatment of the ZLB and reviewing recent work that solves and estimates DSGE models with the ZLB.

The remainder of this review proceeds as follows. Section 2 presents standard methods for solving nonlinear DSGE models using a simple neoclassical growth model as an example. Section 3 provides an application of the nonlinear solution methods to a New Keynesian DSGE model incorporating the ZLB and extends the methods to deal with the curse of dimensionality. Section 4 explains the estimation methods for a general class of nonlinear DSGE models based on a full-information likelihood approach. Section 5 considers recent studies related to the estimation of fully nonlinear DSGE models with the ZLB. Section 6 is the conclusion.

## 2 Solution Methods for Nonlinear DSGE Models

In this and following sections, we discuss how to solve nonlinear DSGE models. We also provide numerical examples using a stochastic neoclassical model and a New Keynesian model.<sup>1</sup>

In this section, we use a stochastic neoclassical growth model as an example to demonstrate how to solve nonlinear DSGE models with the time iteration (TI) method, which solves a functional

---

<sup>1</sup>MATLAB codes are available at <https://github.com/tkksnk/NKZLB>.

equation called the Coleman operator (Coleman, 1990) to obtain a policy function by using an iterative method. The TI method is especially useful when the second welfare theorem does not hold and we need to work with the first-order necessary conditions for the equilibrium in a decentralized economy rather than the social planner's problem. This is typically the case in New Keynesian models, as we explain in the next section.

In terms of numerical methods, we use three different techniques to solve a nonlinear DSGE model by the standard TI method; *interpolation*, *optimization*, and *integration*. *Interpolation*: First, we approximate the state space of continuous variables (for example, capital and technology in the standard neoclassical growth model) by discretized grid points. We interpolate function values between grid points by fitting polynomials such as Chebyshev polynomials. Chebyshev polynomials are useful as they satisfy a property called orthogonality. *Optimization*: Then, in the standard problem, we need to solve a nonlinear optimization (i.e., root-finding) problem at each grid point in the model's state space. *Integration*: We also need to evaluate the integral with regard to the expectation on the next period's variables. We use Gaussian–Hermite quadrature, which is often used to solve nonlinear DSGE models.<sup>2</sup>

As is well known in the literature, by applying the nonstochastic parameterized expectations algorithm (PEA) originally proposed by Christiano and Fisher (2000), we can avoid costly nonlinear optimization in the standard TI method.<sup>3</sup> There are two distinct ways to apply nonstochastic PEAs. One is to fit polynomials to future variables, and the other is to fit polynomials to current variables. With the latter approach, we can further avoid costly numerical integration by utilizing the technique of *precomputation of integrals* (Judd et al., 2017) and reduce computational costs even more. Compared with the conventional method that requires nonlinear optimization and numerical integration, the methods considered in this review are more efficient in terms of computation time. Therefore, they are also useful for estimating structural parameters in nonlinear DSGE models.

## 2.1 Setup

Time is discrete and goes from zero to infinity  $t = 0, 1, \dots, \infty$ . The social planner maximizes the expected lifetime utility

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t u(c_t) \quad (1)$$

subject to the resource constraint

$$c_t + k_{t+1} \leq f(k_t, z_t) \quad (2)$$

where  $c_t$  is consumption,  $k_t$  is capital, and  $z_t$  is technology at time  $t$ .  $u(c)$  and  $f(k, z)$  are utility and production functions satisfying standard assumptions.  $\beta$  is a discount factor.  $\mathbb{E}_0$  is the expectation operator at time 0.  $z_t$  follows an AR(1) process

$$z_{t+1} = \rho z_t + \epsilon_{t+1}, \quad \epsilon_{t+1} \sim N(0, \sigma_{\epsilon}^2), \quad (3)$$

where  $\rho$  is a parameter for the persistence of  $z_t$ .  $\epsilon_{t+1}$  is i.i.d. and follows a normal distribution with a variance of  $\sigma_{\epsilon}^2$ .

---

<sup>2</sup>Tauchen (1986) suggested another method that uses a Markov chain to approximate the original first-order autoregressive, or AR(1) process.

<sup>3</sup>The endogenous grid point method (Carroll, 2006; Barillas and Fernández-Villaverde, 2007; Fella, 2014) can also avoid nonlinear optimization.

By choosing the sequence of  $c_t$  so as to maximize (1), we obtain the value function

$$V(k_0, z_0) = \max_{\{c_t\}} \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t u(c_t),$$

where  $k_0$  and  $z_0$  are given as the state variables. We can write the sequential problem recursively as

$$\begin{aligned} V(k_t, z_t) &= \max_{c_t, k_{t+1}} \{u(c_t) + \beta \mathbb{E}_t V(k_{t+1}, z_{t+1})\}, \\ &= \max_{c_t} \{u(c_t) + \beta \mathbb{E}_t V(f(k_t, z_t) - c_t, z_{t+1})\}. \end{aligned}$$

This is the Bellman equation. The maximizer of the right-hand side is the policy function, and is denoted by  $c_t = \sigma(k_t, z_t)$ .

The first-order necessary condition (called the Euler equation) for the maximization is given by

$$u_c(c_t) = \beta \mathbb{E}_t \{u_c(c_{t+1}) f_k(k_{t+1}, z_{t+1})\},$$

where  $u_c(c)$  denotes the differential of  $u(c)$  in terms of  $c$  and  $f_k(k, z)$  is the differential of  $f(k, z)$  in terms of  $k$ .

In what follows, we first briefly review how to solve the maximization problem by using the Bellman operator, which is defined on the Bellman equation. We then present the Coleman operator ([Coleman, 1990](#)), which is defined on the Euler equation, as an analogy to the Bellman operator. The purpose of the Bellman operator is to solve the Bellman equation for the value function, whereas the Coleman operator is helpful to solve the Euler equation for the policy function.

**Bellman operator** For the sake of exposition, hereafter we drop the time subscripts and replace  $x_{t+1}$  by  $x'$ . Then, the Bellman equation becomes

$$V(k, z) = \max_{c \in (0, f(k, z)]} \left\{ u(c) + \beta \int V(f(k, z) - c, z') p(z'|z) dz' \right\}, \quad (4)$$

where  $p(z'|z)$  is the probability density function of  $z'$  conditional on  $z$ . The maximizer of the right-hand side,  $c = \sigma(k, z)$ , satisfies

$$V(k, z) = u(\sigma(k, z)) + \beta \int V(f(k, z) - \sigma(k, z), z') p(z'|z) dz'. \quad (5)$$

Equation (4) or (5) is a functional equation to be solved for an unknown function  $V(k, z)$  (and associated function  $\sigma(k, z)$ ). The Bellman equation can be written as  $V = T(V)$ , where a mapping  $T$  is called the Bellman operator.  $V(k, z)$  is obtained as the solution of the functional equation.

The value function iteration method proceeds by constructing a sequence of value functions. The sequence is created by iterating on the following equation, starting from  $V^{(0)}$ , and continuing until  $V^{(i)}$  has converged. The new value function is obtained by

$$V^{(i)}(k, z) = \max_{c \in (0, f(k, z)]} \left\{ u(c) + \beta \int V^{(i-1)}(f(k, z) - \sigma(k, z), z') p(z'|z) dz' \right\},$$

where  $(k, z)$  are given as the state variables. Thus, the Bellman operator  $T$  can be interpreted as a mapping from the old value function  $V^{(i-1)}$  to the new value function  $V^{(i)}$ .

**Coleman operator** Now consider the first-order necessary condition for the same maximization problem

$$\begin{aligned} u_c(c) &= \beta \int u_c(c') f_k(k', z') p(z'|z) dz' \\ &= \beta \int u_c(\sigma(f(k, z) - c, z')) f_k(f(k, z) - c, z') p(z'|z) dz' \end{aligned} \quad (6)$$

Note that we have substituted  $c' = \sigma(k', z')$  and  $k' = f(k, z) - c$ . This is also a functional equation to be solved for an unknown function  $c = \sigma(k, z)$  and can be written as  $\sigma = K(\sigma)$ , where a mapping  $K$  is referred to as the Coleman operator in acknowledging the work of [Coleman \(1990\)](#).

Similarly to the value function iteration method, the TI method proceeds by constructing a sequence of policy functions. The sequence is created by iterating on the following equation, starting from  $\sigma^{(0)}$ , and continuing until  $\sigma^{(i)}$  has converged. The new policy function is obtained by solving

$$u_c(c) = \beta \int u_c\left(\sigma^{(i-1)}(f(k, z) - c), z'\right) f_k(f(k, z) - c, z') p(z'|z) dz'$$

for  $c = \sigma^{(i)}(k, z)$ , taking  $(k, z)$  as given. Thus, the Coleman operator  $K$  can be interpreted as a mapping from the old policy function  $\sigma^{(i-1)}$  to the new policy function  $\sigma^{(i)}$ . This is analogous to the Bellman operator in dynamic programming and the value function iteration method where we have a mapping from the old to the new value function.

Some studies prove the existence and uniqueness of the solution to the Coleman operator. [Coleman \(1991\)](#) proves the existence of the equilibrium as the fixed point of a functional equation in a stochastic neoclassical growth model with distortionary tax. [Greenwood and Huffman \(1995\)](#) extend it to several cases. [Sargent and Stachurski \(2018\)](#) discuss the property of the Coleman operator in a neoclassical growth model similar to that in the present review. They also prove that the TI converges at the same rate as the value function iteration for solving the social planner's problem. [Richter et al. \(2014\)](#) briefly review the theory of this kind of operator applied to DSGE models and related studies.

## 2.2 Time iteration (TI) method

The TI method is a way to solve the Coleman operator for the policy function by using an iterative method. It takes the following steps:

1. Make an initial guess for the policy function  $\sigma^{(0)}$ .
2. For  $i = 1, 2, \dots$  ( $i$  is an index for the number of iterations), taking the policy function previously obtained  $\sigma^{(i-1)}$  as given, solve

$$u_c(c) = \beta \int u_c\left(\sigma^{(i-1)}(f(k, z) - c, z')\right) f_k(f(k, z) - c, z') p(z'|z) dz' \quad (7)$$

for  $c$ .

3. Update the policy function by setting  $c = \sigma^{(i)}(k, z)$ .
4. Repeat steps 2–3 until  $\|\sigma^{(i)} - \sigma^{(i-1)}\|$  is small enough.

In steps 2–3, we have a mapping from the old policy function  $\sigma^{(i-1)}$  to the new policy function  $\sigma^{(i)}$ . We obtain the equilibrium policy function by solving (7) iteratively until the difference between the old and new policy functions is sufficiently small.

From a computational perspective, we cannot solve equation (7) for  $c$  for all the possible values of  $(k, z)$  as they are continuous state variables on a compact state space  $[k_1, k_{N_k}] \times [z_1, z_{N_z}]$ . Therefore, we discretize the state space of  $(k, z)$  by grid points:

$$k_j \in \{k_1, k_2, \dots, k_{N_k}\} \quad z_m \in \{z_1, z_2, \dots, z_{N_z}\}$$

where  $(j, m)$  is a pair of indices for grid points of the state variables. Then, after rewriting equation (7) using the grid points, we solve the residual function

$$\begin{aligned} R(c_{jm}^*; k_j, z_m, \sigma^{(i-1)}) &= -u_c(c_{jm}^*) \\ &\quad + \beta \int \left[ u_c \left( \sigma^{(i-1)}(f(k_j, z_m) - c_{jm}^*, z') \right) f_k(f(k_j, z_m) - c_{jm}^*, z') p(z'|z_m) \right] dz' \\ &= 0 \end{aligned} \tag{8}$$

for the optimal level of consumption  $c_{jm}^*$  at each grid point  $(k_j, z_m)$ .

In terms of numerical methods to solve the residual function (8) at each grid point, we use the following numerical techniques; *interpolation*, *optimization*, and *integration*. In this review, we especially focus on the technique of interpolation, as we revisit this technique in the context of sparse grid methods in Section 3.4. For more extensive and detailed explanations of the techniques covered in this section, see, for example, [Judd \(1998\)](#), [Miranda and Fackler \(2002\)](#), and [Heer and Maussner \(2009\)](#) among others.

**Interpolation** We interpolate the policy function to compute the value of  $c' = \sigma^{(i-1)}(k', z')$  in (8) off the grid points as we know the value of  $\sigma^{(i-1)}(k_j, z_m)$  only on the grid points. We fit a polynomial  $\hat{\sigma}^{(i-1)}(k, z; \boldsymbol{\theta})$ , where  $\boldsymbol{\theta}$  is a set of the polynomial's coefficients, to the values of  $\{\sigma^{(i-1)}(k_j, z_m)\}$  to evaluate the policy function off the grid points.

For example, consider a univariate function  $\sigma(x)$  and assume that we only know the values of the function at grid points of  $x \in \{x_1, \dots, x_N\}$ .<sup>4</sup> Suppose we interpolate functional values of  $\sigma(x)$  between the grid points using an  $N$ -order polynomial function<sup>5</sup>

$$\hat{\sigma}(x; \boldsymbol{\theta}) = \theta_0 + \theta_1 x + \theta_2 x^2 + \dots + \theta_{N-1} x^{N-1}.$$

We have  $N$  unknown coefficients in  $\boldsymbol{\theta} = [\theta_0, \theta_1, \dots, \theta_{N-1}]'$  and the values of  $\sigma(x)$  at  $N$  grid points. We then fit the polynomial to the values by ordinary least squares. However, it is well known that fitting ordinary polynomials is subject to a multicollinearity problem and the estimated coefficients can be unstable.

Instead, we use a different type of polynomials known as Chebyshev polynomials. Taking the

<sup>4</sup>In general, the number of grid points,  $\tilde{N}$ , can be larger than the degree of polynomials,  $N$ . We assume  $\tilde{N} = N$  so that the residual function is exactly satisfied only on the grid points. This is called *collocation*. There are other types of projection methods, such as the Galerkin method ([McGrattan, 1996](#)), but in the present paper, we focus on the collocation method.

<sup>5</sup>Using a polynomial is a global approximation as a single function covers all the state space. We can instead consider local approximations using multiple piecewise functions to cover the state space, such as linear interpolation and cubic spline interpolation. For detailed explanation of linear interpolation, cubic spline interpolation, and Chebyshev polynomials and their application to economic models, see Chapter 6 in [Miranda and Fackler \(2002\)](#).

basis functions  $T(x) : [-1, 1] \rightarrow [-1, 1]$  as given, we have an univariate polynomial

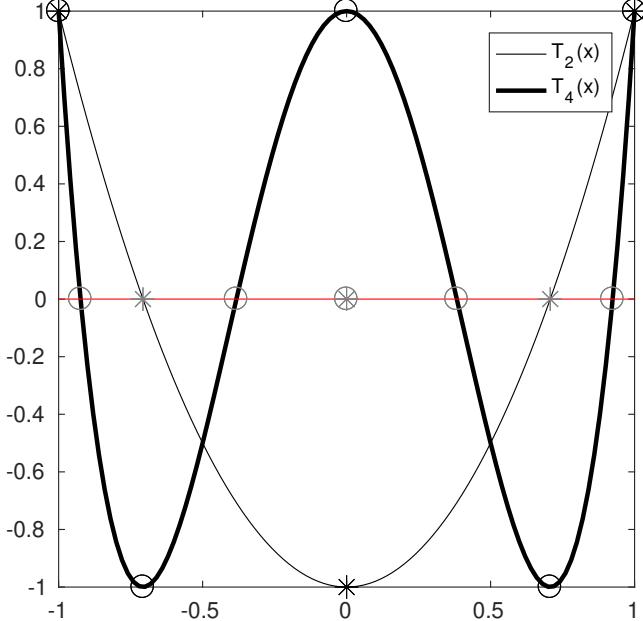
$$\hat{\sigma}(x; \boldsymbol{\theta}) = \theta_0 + \theta_1 T_1(x) + \theta_2 T_2(x) + \cdots + \theta_{N-1} T_{N-1}(x).$$

Chebyshev polynomials or Chebyshev basis functions are of the form

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_2(x) &= 2x^2 - 1, \\ &\vdots \\ T_{N-1}(x) &= 2xT_{N-2}(x) - T_{N-3}(x), \end{aligned}$$

where  $x \in [-1, 1]$ . Figure 1 shows the shape of the second- and fourth-order polynomials  $T_2(x)$  and  $T_4(x)$ .

Figure 1: Chebyshev polynomials



Notes: Gray stars are Chebyshev zeros of  $T_2(x)$  and black stars are Chebyshev extrema of  $T_2(x)$ . Gray circles are Chebyshev zeros of  $T_4(x)$  and black circles are Chebyshev extrema of  $T_4(x)$ .

The polynomial is evaluated at collocation points  $x_j$  for  $j = 0, 1, \dots, N - 1$  at which  $\hat{\sigma}(x_j; \boldsymbol{\theta}) = \sigma(x_j)$  holds. There are two types of collocation points. (i) Chebyshev zeros:  $x_0 = 0$  and  $x_j = \cos\left(\frac{(2j-1)\pi}{2(N-1)}\right)$  for  $j = 1, \dots, N - 1$ , and (ii) Chebyshev extrema:  $x_j = \cos\left(\frac{j\pi}{N-1}\right)$  for  $j = 0, 1, \dots, N - 1$ . That is, Chebyshev zeros are the solutions of  $T(x) = 0$  (and  $x_0 = 0$ ) and Chebyshev extrema are the solutions of  $|T(x)| = 1$ . For example, if the second-order Chebyshev polynomial  $T_2(x)$  is used as a basis function (with  $N = 3$ ), the collocation points of Chebyshev zeros are given by a vector  $[0, -\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$  and the collocation points of Chebyshev extrema are given by a vector  $[0, -1, 1]$ .

In Figure 1, we can see that the collocation points (i.e., the values on the x-axis) of the Chebyshev extrema with  $N = 5$  (shown by black circles at  $(x, y) = (-1, 1), (-0.7071, -1), (0, 1), (0.7071, -1)$ )

$(1), (1, 1))$  nest those with  $N = 3$  (shown by black stars at  $(x, y) = (-1, 1), (0, -1), (1, 1)$ ). This property is useful when we construct multidimensional sparse grid points based on unidimensional grid points as discussed in Section 3.4.

Once we have the collocation points  $\{x_j\}$  and the function values  $\{\sigma(x_j)\}$  evaluated at  $x_j$  for  $j = 0, 1, \dots, N - 1$ , we fit  $\hat{\sigma}(x_j; \boldsymbol{\theta})$  to the associated values to obtain  $\boldsymbol{\theta}$ :

$$\begin{bmatrix} \sigma(x_0) \\ \sigma(x_1) \\ \vdots \\ \sigma(x_{N-1}) \end{bmatrix} = \begin{bmatrix} 1 & T_1(x_0) & T_2(x_0) & \cdots & T_{N-1}(x_0) \\ 1 & T_1(x_1) & T_2(x_1) & \cdots & T_{N-1}(x_1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & T_1(x_{N-1}) & T_2(x_{N-1}) & \cdots & T_{N-1}(x_{N-1}) \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_{N-1} \end{bmatrix},$$

or

$$\sigma(\mathbf{x}) = T(\mathbf{x})\boldsymbol{\theta},$$

where  $\mathbf{x} = [x_0, \dots, x_{N-1}]$ . Then, given that  $T(\mathbf{x})$  is nonsingular, we have  $\boldsymbol{\theta} = T(\mathbf{x})^{-1}\sigma(\mathbf{x})$ .

Note that the inverse of  $T(\mathbf{x})$  can be ill-conditioned (i.e.,  $T(\mathbf{x})$  is nearly singular) and expensive to calculate. The Chebyshev basis functions and collocation points are more likely nonsingular as they satisfy the orthogonality property. That is, each column of  $T(\mathbf{x})$  is uncorrelated with each other. Therefore, the estimated coefficients (for example, by ordinary least squares) become more stable. Also,  $T(\mathbf{x})$  is fixed once we choose the basis function and grid points. In solving DSGE models, we can precompute the inverse of  $T(\mathbf{x})$  at the initialization stage, before the main part of the TI to conserve computation time.

So far, we have assumed that the values of  $x_j$  are restricted to  $[-1, 1]$ . For a general function that takes a value of  $k_j \in [k_1, k_N]$  at each grid point, we have to transform  $k_j$  to  $x_j \in [-1, 1]$  by applying<sup>6</sup>

$$x_j = \varphi(k_j) = \frac{2(k_j - k_1)}{k_N - k_1} - 1,$$

Then we have a univariate polynomial

$$\hat{\sigma}(k; \boldsymbol{\theta}) = \theta_0 + \theta_1 T_1(\varphi(k)) + \theta_2 T_2(\varphi(k)) + \cdots + \theta_{N-1} T_{N-1}(\varphi(k)).$$

It is easy to extend the above analysis to higher dimensions. Now we discuss the case of two-dimensional second-order polynomials. Consider a bivariate function  $\sigma(x, y)$  and assume that we only know the values of the function at grid points of  $x \in \{x_1, \dots, x_{N_x}\}$  and  $y \in \{y_1, \dots, y_{N_y}\}$  where  $N_l$ ,  $l \in \{x, y\}$  are the numbers of grid points for each dimension. Suppose  $N_x = N_y = 3$ . ( $N_x$  and  $N_y$  can be different numbers.) We interpolate the functional values of  $\sigma(x, y)$  between the grid points by fitting the following two-dimensional second-order polynomial

$$\begin{aligned} \hat{\sigma}(x, y; \boldsymbol{\theta}) = & \theta_{0,0} + \theta_{1,0} T_1(x) + \theta_{2,0} T_2(x) + \theta_{0,1} T_1(y) + \theta_{0,2} T_2(y) \\ & + \theta_{1,1} T_1(x) T_1(y) + \theta_{1,2} T_1(x) T_2(y) \\ & + \theta_{2,1} T_2(x) T_1(y) + \theta_{2,2} T_2(x) T_2(y) \end{aligned}$$

This is a tensor product of basis functions for each dimension  $T(x) = [1, T_1(x), T_2(x)]$  and  $T(y) = [1, T_1(y), T_2(y)]$ . There are 9 coefficients, so we need at least  $N = N_x N_y = 9$  collocation points. We

---

<sup>6</sup> The inverse of the mapping is given by  $k_j = \varphi^{-1}(x_j) = k_1 + 0.5(1 + x_j)(k_N - k_1)$ .

employ a tensor product of Chebyshev extrema for each dimension  $[0, -1, 1]$  as collocation points:

$$(x, y) \in \{(0, 0), (-1, 0), (1, 0), (0, -1), (0, 1), \\ (-1, -1), (1, -1), (-1, 1), (1, 1)\}.$$

Once we have the collocation points  $\{x_j, y_k\}$  and the function values  $\{\sigma(x_j, y_k)\}$  evaluated at  $(x_j, y_k)$  for  $j = 0, 1, \dots, N_x - 1$  and  $k = 0, 1, \dots, N_y - 1$ , we fit  $\hat{\sigma}(x, y; \boldsymbol{\theta})$  to the data to obtain  $\boldsymbol{\theta}$ :

$$\begin{bmatrix} \sigma(x_0, y_0) \\ \sigma(x_1, y_0) \\ \sigma(x_2, y_0) \\ \sigma(x_0, y_1) \\ \vdots \\ \sigma(x_2, y_2) \end{bmatrix} = \begin{bmatrix} 1 & T_1(x_0) & T_2(x_0) & T_1(y_0) & T_2(y_0) & T_1(x_0)T_1(y_0) & \cdots & T_2(x_0)T_2(y_0) \\ 1 & T_1(x_1) & T_2(x_1) & T_1(y_0) & T_2(y_0) & T_1(x_1)T_1(y_0) & \cdots & T_2(x_1)T_2(y_0) \\ 1 & T_1(x_2) & T_2(x_2) & T_1(y_0) & T_2(y_0) & T_1(x_2)T_1(y_0) & \cdots & T_2(x_2)T_2(y_0) \\ 1 & T_1(x_0) & T_2(x_0) & T_1(y_1) & T_2(y_1) & T_1(x_0)T_1(y_1) & \cdots & T_2(x_0)T_2(y_1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & T_1(x_2) & T_2(x_2) & T_1(y_2) & T_2(y_2) & T_1(x_2)T_1(y_2) & \cdots & T_2(x_2)T_2(y_2) \end{bmatrix} \begin{bmatrix} \theta_{0,0} \\ \theta_{1,0} \\ \theta_{2,0} \\ \theta_{0,1} \\ \vdots \\ \theta_{2,2} \end{bmatrix}$$

or  $\sigma(\mathbf{x}, \mathbf{y}) = T(\mathbf{x}, \mathbf{y})\boldsymbol{\theta}$  where  $\mathbf{x} = [x_0, \dots, x_{N_x-1}]$ ,  $\mathbf{y} = [y_0, \dots, y_{N_y-1}]$ , and  $T(\mathbf{x}, \mathbf{y}) = T(\mathbf{x}) \otimes T(\mathbf{y})$ . Then, given that  $T(\mathbf{x}, \mathbf{y})$  is nonsingular, we have  $\boldsymbol{\theta} = T(\mathbf{x}, \mathbf{y})^{-1}\sigma(\mathbf{x}, \mathbf{y})$ .

For dimensions higher than three, we merely need to compute the tensor products of the basis functions and collocation points for each dimension. However, the number of elements in tensor products exponentially increases in the number of dimensions of the state space. For example, if we have three grid points for each dimension, the number of grid points for each of four, five, and 10 state variables are  $3^4 = 81$ ,  $3^5 = 243$ , and  $3^{10} = 59049$ , respectively. This is known as the curse of dimensionality. Sections 3.4 and 3.5 illustrate how to deal with this problem.

**Optimization** We solve an optimization (root-finding) problem in (8). For example, a variant of Newton's method is available in MATLAB with the command `fsolve`. Various other methods are also implemented in modern programming languages; alternatively, we can write a nonlinear optimization routine ourselves. However, nonlinear optimization can be time consuming when the number of grid points and/or nonlinear equations to be solved is large.

**Integration** We also calculate integrals in (8) with regard to  $z'$ . Gaussian–Hermite quadrature is often used in solving nonlinear DSGE models. In short, the quadrature nodes  $\{x_i\}_{i=1}^n$  and quadrature weights  $\{w_i\}_{i=1}^n$  approximate  $\int f(x)w(x)dx \approx \sum w_i f(x_i)$ , where  $n$  is the number of the quadrature nodes and weights for each variable.  $\{x_i, w_i\}_{i=1}^n$  are chosen so that an  $n$ -point Gaussian–Hermite quadrature is order  $2n - 1$  exact. That is, it exactly computes the integral of any polynomial of order  $2n - 1$  or less, up to rounding errors if any.<sup>7</sup> Note that the number of the quadrature nodes and weights is exponentially increasing in the number of exogenous state variables.

However, numerical integration can also be costly as we calculate the weighted average of the next period's values evaluated at each quadrature point. The following subsection explains how to avoid such costly nonlinear optimization and numerical integration.

## 2.3 Nonstochastic PEAs

Marcet (1988) originally develops the PEA, which uses a stochastic approach based on Monte Carlo simulations to solve for the coefficients of polynomials that approximate the next period's

---

<sup>7</sup>See, for example, Chapter 5.2 in [Miranda and Fackler \(2002\)](#).

expectation.<sup>8</sup> Christiano and Fisher (2000) point out that PEAs can be applied to nonstochastic grid points such as Chebyshev collocation points. They use Chebyshev polynomials to approximate the expectation functions and solve for the coefficients of polynomials by a projection method (Judd, 1992).<sup>9</sup> They refer to such a nonstochastic approach as Chebyshev PEA. In more recent work, Gust et al. (2017) apply the nonstochastic Chebyshev PEA by fitting polynomials to future variables to solve a nonlinear New Keynesian model with occasionally binding ZLB constraints.<sup>10</sup>

By using nonstochastic PEAs, we can avoid costly nonlinear optimization. There are two distinct ways to apply these: the first is to fit polynomials to future variables, and the second is to fit polynomials to current variables. With the latter approach, we can also avoid costly numerical integration, as explained below.

### 2.3.1 Fitting polynomials to future variables

We replace steps 2–3 presented in Section 2.2. To start, we define a new function for the expectation term of future variables known as the expectation function<sup>11</sup>

$$e(k, z) \equiv \beta \int u_c(c') f_k(k', z') p(z'|z) dz'.$$

Then, in step 2, taking the values of the expectation function  $e^{(i-1)}(k_j, z_m)$  at each grid point as given, we immediately obtain

$$c_{jm}^* = u_c^{-1}\left(e^{(i-1)}(k_j, z_m)\right),$$

and the policy function  $c = \sigma^{(i)}(k, z)$ , where  $u_c^{-1}$  is a monotonic transformation. Note that we do not solve the nonlinear equation (8) using optimization routines. Instead, we use analytical mapping between  $e^{(i-1)}(k, z)$  and  $\sigma^{(i)}(k, z)$ .

In step 3, having the policy functions at hand, we also update

$$\begin{aligned} e^{(i)}(k_j, z_m) &= \beta \int u_c\left(\sigma^{(i)}(k', z')\right) f_k(k', z') p(z'|z_m) dz' \\ &\approx \beta \int u_c\left(u_c^{-1}\left(\hat{e}^{(i-1)}(k', z'; \boldsymbol{\theta})\right)\right) f_k(k', z') p(z'|z_m) dz' \end{aligned}$$

where

$$k' = f(k_j, z_m) - c_{jm}^*$$

is obtained in the previous step. Note that we interpolate the values of  $\sigma^{(i)}(k', z')$  (or equivalently  $e^{(i-1)}(k', z')$ ) by using a polynomial  $\hat{e}^{(i-1)}(k', z'; \boldsymbol{\theta})$  parameterized with a vector  $\boldsymbol{\theta}$ . That is,  $c' = \sigma^{(i)}(k', z') \approx u_c^{-1}(\hat{e}^{(i-1)}(k', z'; \boldsymbol{\theta}))$  holds, where  $\boldsymbol{\theta}$  is obtained by fitting the polynomial to the data of future variables at each grid point  $\{e^{(i-1)}(k_j, z_m)\}$ . We compute an integral of the composite function with regard to  $z'$ , and then apply a numerical integration technique with Gaussian–Hermite

<sup>8</sup>Judd et al. (2011) and Maliar and Maliar (2015) further develop this approach to make the computation more robust and efficient. We discuss this application in Section 3.5.

<sup>9</sup>den Haan (2018) also mentions nonstochastic PEAs in his lecture notes.

<sup>10</sup>They also use Smolyak's method for sparse grid points to mitigate the curse of dimensionality, as do we in Section 3.4.

<sup>11</sup>Note that with this definition we implicitly assume the relationship between the current and future variables (i.e.,  $c' = \sigma(k', z')$  and  $k' = f(k, z) - \sigma(k, z)$ ).

quadrature.

### 2.3.2 Fitting polynomials to current variables

As an alternative to defining the expectation function for future variables, we can define the following function for current variables<sup>12</sup>

$$v(k, z) \equiv \beta u_c(c) f_k(k, z).$$

Note that the expectation term is obtained as the integral of the expectation function evaluated at the next period's state variables. Then, in step 2, taking the expectation function  $v^{(i-1)}(k, z)$  and the values of  $\sigma^{(i-1)}(k_j, z_m)$  at each grid point as given, we have

$$\begin{aligned} c_{jm}^* &= u_c^{-1} \left( \int v^{(i-1)}(k', z') p(z'|z_m) dz' \right), \\ &\approx u_c^{-1} \left( \int \hat{v}^{(i-1)}(k', z'; \boldsymbol{\theta}) p(z'|z_m) dz' \right) \end{aligned}$$

where

$$k' = f(k_j, z_m) - \sigma^{(i-1)}(k_j, z_m).$$

Note that we interpolate the values of  $v^{(i-1)}(k', z')$  by a polynomial  $\hat{v}^{(i-1)}(k', z'; \boldsymbol{\theta})$ , where  $\boldsymbol{\theta}$  is obtained by fitting the polynomial to the values of current variables at each grid point  $\{v^{(i-1)}(k_j, z_m)\}$ . We can avoid nonlinear optimization by a successive approximation  $k' = f(k_j, z_m) - \sigma^{(i-1)}(k_j, z_m)$ .<sup>13</sup> An integral of the polynomial  $\hat{v}^{(i-1)}(k', z'; \boldsymbol{\theta})$  with regard to  $z'$  can be computed either numerically or analytically.

In step 3, we also update

$$\begin{aligned} v^{(i)}(k_j, z_m) &= \beta u_c(c_{jm}^*) f_k(k_j, z_m), \\ \sigma^{(i)}(k_j, z_m) &= c_{jm}^*, \end{aligned}$$

where  $c_{jm}^*$  is obtained in the previous step.

## 2.4 Precomputation of integrals

In each case of the nonstochastic PEAs considered in the previous subsection, we compute integrals with regard to  $z'$ . Only when we fit polynomials to current variables can we utilize the precomputation technique of integrals developed by Judd et al. (2017) to avoid computing numerical integration. For example, we fit a second-order polynomial (without cross terms) to the values of  $v(k_j, z_m) = u_c(\sigma(k_j, z_m)) f_k(k_j, z_m)$  at each grid point  $(k_j, z_m)$ :<sup>14</sup>

$$\hat{v}(k, z; \boldsymbol{\theta}) = \theta_{0,0} + \theta_{1,0}k + \theta_{2,0}k^2 + \theta_{0,1}z + \theta_{0,2}z^2.$$

---

<sup>12</sup>Note that with this definition we implicitly assume the relationship  $c = \sigma(k, z)$ .

<sup>13</sup>This is also known as fixed-point iteration. Alternatively, we can solve the nonlinear equation  $c = u_c^{-1} \left( \int \hat{v}^{(i-1)}(f(k_j, z_m) - c, z'; \boldsymbol{\theta}) p(z'|z_m) dz' \right)$  for  $c$ .

<sup>14</sup>For the sake of exposition, an ordinary polynomial is used. A Chebyshev polynomial  $\theta_{0,0} + \theta_{1,0}T_1(\varphi(k)) + \theta_{2,0}T_2(\varphi(k)) + \theta_{0,1}T_1(\varphi(z)) + \theta_{0,2}T_2(\varphi(z))$  can be used instead in a straightforward way.

Then, we can analytically obtain the integral as

$$\begin{aligned}
\int \hat{v}(k', z'; \boldsymbol{\theta}) p(z'|z) dz' &= \theta_{0,0} + \theta_{1,0} k' + \theta_{2,0} (k')^2 + \int (\theta_{0,1} z' + \theta_{0,2} (z')^2) p(z'|z) dz' \\
&= \theta_{0,0} + \theta_{1,0} k' + \theta_{2,0} (k')^2 + \int (\theta_{0,1}(\rho z + \epsilon') + \theta_{0,2}(\rho z + \epsilon')^2) p(\epsilon') d\epsilon' \\
&= \theta_{0,0} + \theta_{1,0} k' + \theta_{2,0} (k')^2 + \theta_{0,1} \rho z + \theta_{0,2} \rho^2 z^2 + \theta_{0,2} \sigma_\epsilon^2.
\end{aligned}$$

As pointed out by [Judd et al. \(2017\)](#), the precomputation technique can be used only when we express the function to be integrated by a simple parameterized form. This is precisely the case when we fit polynomials to current variables. By contrast, when we fit polynomials to future variables, we need to compute the integral of

$$\int u_c(u_c^{-1}(\hat{e}(k', z'; \boldsymbol{\theta}))) f_k(k', z') p(z'|z) dz',$$

which is a composite nonlinear function. It is difficult or impossible to calculate the integrals of such a function analytically.

## 2.5 Numerical examples

For numerical illustration, we employ the following functional forms:  $u(c) = \frac{c^{1-\tau}-1}{1-\tau}$  and  $f(k, z) = zk^\alpha$ . The parameter values are set as  $\beta = 0.99$ ,  $\alpha = 1/3$ ,  $\delta = 0.025$ ,  $\rho_z = 0.95$ , and  $\sigma_\epsilon = 0.008$ . We consider three different values for the risk aversion parameter; i.e.,  $\tau = \{1, 2, 5\}$ , to investigate how accuracy and computation time can change as the nonlinearity of the model increases. We use the second- and fourth-order Chebyshev polynomials for  $k$  and  $z$  for interpolation in each solution algorithm. The number of grid points are  $3^2 = 9$  and  $5^2 = 25$  for each polynomial case. The number of Gaussian–Hermite quadrature is set to three.

We evaluate the accuracy of approximation using the Euler equation error, which is calculated as

$$\mathcal{E}(k, z) = 1 - \beta \int \left\{ \left( \frac{\sigma_c(k', z')}{\sigma_c(k, z)} \right)^{-\tau} (1 - \delta + \alpha z' k'^{\alpha-1}) \right\} \mu(z'|z) dz' \quad (9)$$

where  $k' = f(k, z) - \sigma_c(k, z)$ . The Euler equation errors are calculated based on the series simulated from the approximated solutions. Each stochastic simulation is done for 10,500 periods with the first 500 periods discarded. The same sequence of random variables for  $z$  is used throughout all the simulations.

[Table 1](#) compares the performance of the different methods (TI, future PEA, and current PEA) with the different orders of polynomials (second-order and fourth-order) and values of the risk aversion parameter ( $\tau = \{1, 2, 5\}$ ) in terms of accuracy measured by the Euler equation errors given by equation (9) and the computation time (in seconds).

Table 1: Accuracy and speed of TI, future PEA, and current PEA: Stochastic neoclassical growth model

Polynomial, $\tau$	TI			future PEA			current PEA		
	$L_1$	$L_\infty$	CPU	$L_1$	$L_\infty$	CPU	$L_1$	$L_\infty$	CPU
2nd, $\tau = 1.0$	-5.12	-4.60	4.03	-4.23	-3.69	0.04	-3.13	-2.44	0.02
4th, $\tau = 1.0$	-7.08	-6.72	9.76	-5.92	-5.59	0.09	-3.13	-2.44	0.04
2nd, $\tau = 2.0$	-4.82	-4.35	0.86	-3.99	-3.53	0.03	-2.95	-2.26	0.01
4th, $\tau = 2.0$	-6.76	-6.45	2.88	-5.63	-5.36	0.11	-2.96	-2.27	0.04
2nd, $\tau = 5.0$	-4.48	-3.87	0.62	-3.57	-2.88	0.05	-2.67	-1.99	0.02
4th, $\tau = 5.0$	-6.43	-5.38	1.91	-5.10	-3.90	0.15	-2.69	-2.00	0.05

Notes:  $L_1$  is the average and  $L_\infty$  is the maximum of the Euler equation errors in absolute terms (9) (log 10 units) based on a 10,000-period stochastic simulation. CPU is the elapsed time for computing equilibrium (in seconds). Computation is done in MATLAB R2016b using a laptop with Xeon E3-1505Mv5 (2.8 GHz) and 16 GB memory without any parallelization.

To start, the method of PEA collocation with fitting polynomials to future variables (future PEA) is about 10–100 times faster than the standard TI method. Also, PEA collocation with fitting polynomials to current variables (current PEA) is about 2–4 times faster than future PEA. This is because future PEA avoids only nonlinear optimization, whereas current PEA avoids both numerical optimization and numerical integration. Using the proposed methods provides a substantial gain in computational time (current or future PEA), and both are potentially useful for structural estimation. However, there is a trade-off between accuracy and computation speed. Specifically, TI is the most accurate in terms of the averages and maximums of the Euler equation errors, while future PEA is better than current PEA in terms of errors.

The higher-order polynomial reduces the Euler equation errors and increases computation time in all of the methods (except for current PEA). We also find that the larger the risk aversion parameter  $\tau$ , the larger the averages and maximums of the Euler equation errors in all of the methods. This is not surprising, as the model becomes more nonlinear with an increase in  $\tau$ .

### 3 Applications of Nonlinear Solution Methods to DSGE Models with the ZLB

In this section, we show an application of the solution methods in the previous section to the standard New Keynesian model, which has long been a workhorse for monetary policy analysis. The model economy consists of final- and intermediate-good producing firms, households, and monetary and fiscal authorities. Prices are sticky because of Rotemberg-type (1982) adjustment costs. The details of the model, including the optimization problems for households and firms, are provided in the Appendix A.1. As in the previous section, we solve the model in a fully nonlinear and stochastic setting using three different methods: standard TI, nonstochastic PEAs with fitting current variables, and nonstochastic PEAs with fitting future variables.

Moreover, we incorporate the ZLB on nominal interest rates into the New Keynesian model. This extension is not straightforward as the ZLB is occasionally binding depending on the state of the economy. Specifically, we need to deal with kinks in the policy function from the nonnegativity constraint on the nominal interest rate. This is a nontrivial task when we use global approximation techniques such as Chebyshev polynomials that presume smooth policy functions. To deal with

this issue, we adapt an index function approach to the solution methods as in Aruoba et al. (2018); Gust et al. (2017); Hirose and Sunakawa (2015, 2017); Nakata (2017).

While we employ a small-scale New Keynesian model as an example, one might wish to solve a larger-scale DSGE model with a richer dynamic structure using nonlinear methods. The curse of dimensionality would then become an issue as there would be a large number of state variables. We demonstrate how to avoid the curse of dimensionality by using sparse grid point methods such as Smolyak's (1963) method as adapted by Judd et al. (2014) and the simulation-based grid method as in Maliar and Maliar (2015).

### 3.1 Setup

The equilibrium conditions of the New Keynesian model are given by

$$c_t^{-\tau} = \beta \bar{\gamma}^{-1} R_t \mathbb{E}_t \left[ \frac{c_{t+1}^{-\tau}}{z_{t+1} \pi_{t+1}} \right], \quad (10)$$

$$\begin{aligned} 0 &= \left( (1 - \nu^{-1}) + \nu^{-1} c_t^\tau - \phi (\pi_t - \bar{\pi}) \left[ \pi_t - \frac{1}{2\nu} (\pi_t - \bar{\pi}) \right] \right) c_t^{-\tau} y_t \\ &\quad + \beta \phi \mathbb{E}_t [c_{t+1}^{-\tau} y_{t+1} (\pi_{t+1} - \bar{\pi}) \pi_{t+1}], \end{aligned} \quad (11)$$

$$c_t + \frac{\phi}{2} (\pi_t - \bar{\pi})^2 y_t = g_t^{-1} y_t, \quad (12)$$

$$R_t^* = \left( \bar{r} \bar{\pi} \left( \frac{\pi}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_t}{y_t^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{t-1}^{*\rho_R} e^{\epsilon_{R,t}}, \quad (13)$$

$$R_t = \max \{ R_t^*, 1 \}. \quad (14)$$

We have four equations: the consumption Euler equation (10), the New Keynesian Phillips curve (11), the resource constraint (12), and a Taylor-type monetary policy rule (13).  $c_t$  is consumption,  $\pi_t$  is the inflation rate,  $y_t$  is output, and  $R_t^*$  is the notional interest rate that the central bank wishes to set.  $R_t$  is the actual nominal interest rate bounded below by one when we consider the ZLB. Government expenditure  $g_t$  is exogenous and the natural level of output is given by  $y_t^* = (1-\nu)^{1/\tau} g_t$ , which is obtained by setting  $\phi = 0$  in the equilibrium conditions. The description of the parameters ( $\beta, \tau, \nu, \phi, \bar{r}, \bar{\pi}, \psi_1, \psi_2, \rho_R$ ) is given in Table 3 that appears in Section 3.6. Total factor productivity  $A_t$  has a deterministic trend  $\bar{\gamma}$  and a shock to the trend  $z_t$  such as  $\ln \gamma_t \equiv \ln(A_t/A_{t-1}) = \ln \bar{\gamma} + \ln z_t$ . The exogenous shocks  $\{z_t, g_t\}$  follow AR(1) processes

$$\begin{aligned} \ln z_t &= \rho_z \ln z_{t-1} + \epsilon_{z,t}, \\ \ln g_t &= (1 - \rho_g) \ln \bar{g} + \rho_g \ln g_{t-1} + \epsilon_{g,t}, \end{aligned}$$

where  $\rho_g$  and  $\rho_z$  are the parameters for persistence of the shocks. The disturbance terms  $\{\epsilon_{z,t}, \epsilon_{g,t}, \epsilon_{R,t}\}$  are serially uncorrelated and independent from each other. The three disturbances are normally distributed with means zero and standard deviations  $\sigma_z$ ,  $\sigma_g$ , and  $\sigma_R$ , respectively.

It is well known that the standard linearized New Keynesian model without the ZLB has a unique equilibrium when monetary policy is active; that is, the central bank adjusts the nominal interest rate more than one for one to inflation. Otherwise, it has multiple equilibria, such that there are an infinite number of equilibrium trajectories that converge to the steady state. Moreover, a number of studies (Boneva et al., 2016; Cochrane, 2017, 2018; Christiano et al., 2018) document that even where monetary policy is active, there are multiple equilibria once the ZLB is considered

in a nonlinear setting. However, it would be a formidable task to compute all possible equilibria, so we focus on that equilibrium whose linear counterpart is unique, as considered in most of the literature reviewed here.<sup>15</sup>

**Coleman operator** We define a functional operator on the equilibrium conditions. The solution to the functional operator takes the form

$$\begin{aligned} c &= \sigma_c(R_{-1}^*, s), & \pi &= \sigma_\pi(R_{-1}^*, s), \\ R^* &= \sigma_{R^*}(R_{-1}^*, s), & y &= \sigma_y(R_{-1}^*, s), \end{aligned}$$

where  $s = (z, g, \epsilon_R)$ . We have four equilibrium conditions and four endogenous variables to be solved. Note that the ZLB on the nominal interest rate is imposed by  $R = \max\{R^*, 1\}$ . The mapping  $\sigma = K(\sigma)$ , where  $\sigma = [\sigma_c, \sigma_\pi, \sigma_y, \sigma_{R^*}]'$ , solves

$$\begin{aligned} 0 &= -c^{-\tau} + \beta\bar{\gamma}^{-1}R \int \left[ \frac{\sigma_c(R^*, s')^{-\tau}}{z'\sigma_\pi(R^*, s')} \right] p(s'|s) ds', \\ 0 &= \left( (1 - \nu^{-1}) + \nu^{-1}c^\tau - \phi(\pi - \bar{\pi}) \left[ \pi - \frac{1}{2\nu}(\pi - \bar{\pi}) \right] \right) c^{-\tau}y \\ &\quad + \beta\phi \int [\sigma_c(R^*, s')^{-\tau} \sigma_y(R^*, s') (\sigma_\pi(R^*, s') - \bar{\pi}) \sigma_\pi(R^*, s')] p(s'|s) ds', \\ c &+ \frac{\phi}{2}(\pi - \bar{\pi})^2 y = g^{-1}y, \\ R^* &= \left( r\bar{\pi} \left( \frac{\pi}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{-1}^{*\rho_R} e^{\epsilon_R}, \\ R &= \max\{R^*, 1\} \end{aligned}$$

for  $\sigma$ , where  $z' = z^{\rho_z}e^{\epsilon'_z}$ ,  $g'/\bar{g} = (g/\bar{g})^{\rho_g}e^{\epsilon'_g}$ , and  $p(s'|s)$  is the probability density function of  $s'$  conditional on  $s$ . Note that we substitute  $c' = \sigma_c(k', z')$ ,  $\pi' = \sigma_\pi(k', z')$ , and  $y' = \sigma_y(k', z')$ . The operator  $K$  takes its argument as  $\sigma$  and returns a vector of new functions  $K\sigma$  that solves the relevant equations for  $(c, \pi, y, R^*)$  (and  $R$  when the ZLB matters).

### 3.2 Time iteration (TI) method

The TI method involves the following steps:

1. Make an initial guess of the policy function  $\sigma^{(0)}$ .
2. Taking as given the policy function previously obtained  $\sigma^{(i-1)}$  solve the relevant equations for  $(c, \pi, R^*, y)$ .
3. Update the policy function by setting  $c = \sigma_c^{(i)}(R_{-1}^*, s)$ ,  $\pi = \sigma_\pi^{(i)}(R_{-1}^*, s)$ ,  $R^* = \sigma_{R^*}^{(i)}(R_{-1}^*, s)$ , and  $y = \sigma_y^{(i)}(R_{-1}^*, s)$ .
4. Repeat steps 2–3 until  $\|\sigma^{(i)} - \sigma^{(i-1)}\|$  is sufficiently small.

---

<sup>15</sup>Feng et al. (2014) propose a numerical algorithm for computing the set of all Markov equilibria in dynamic competitive market economies in which the welfare theorems may not hold. Their solution method is based on those developed by Abreu et al. (1990) and Judd et al. (2003), which have been applied to computing the set of monetary competitive equilibria by Chang (1998) and Orlik and Presno (2013).

In the standard TI method, we use the techniques of *interpolation*, *optimization*, and *integration*, as per the stochastic neoclassical growth model in Section 2. Namely, we discretize the state space of  $(R_{-1}^*, z, g, \epsilon_R)$  by grid points and interpolate the policy functions off the grid. We solve the multidimensional root-finding problem at each grid point. We calculate the integrals in the consumption Euler equation (10) and the nonlinear New Keynesian Phillips curve (11).

Specifically, in step 2, taking policy functions  $\sigma_x^{(i-1)}(R_{-1}^*, s)$  for  $x \in \{c, \pi, y\}$  as given, we solve

$$\begin{aligned} c_{jm} &= \beta \bar{\gamma}^{-1} R_{jm}^* \int \left[ \frac{\hat{\sigma}_c^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta})^{-\tau}}{z' \hat{\sigma}_\pi^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta})} \right] p(s'|s) ds', \\ 0 &= \left( (1 - \nu^{-1}) + \nu^{-1} c_{jm}^\tau - \phi(\pi_{jm} - \bar{\pi}) \left[ \pi_{jm} - \frac{1}{2\nu} (\pi_{jm} - \bar{\pi}) \right] \right) c_{jm}^{-\tau} y_{jm} \\ &\quad + \beta \phi \int \left[ \hat{\sigma}_c^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta})^{-\tau} \hat{\sigma}_y^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta}) \left( \hat{\sigma}_\pi^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta}) - \bar{\pi} \right) \hat{\sigma}_\pi^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta}) \right] p(s'|s) ds', \\ y_{jm} &= \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm} - \bar{\pi})^2 \right]^{-1} c_{jm}, \\ R_{jm}^* &= \left( \bar{r} \bar{\pi} \left( \frac{\pi_{jm}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^{*\rho_R} e^{\epsilon_{R,m}}. \end{aligned}$$

for  $(c_{jm}, \pi_{jm}, y_{jm}, R_{jm}^*)$ . Note that we interpolate the values of  $\hat{\sigma}_x^{(i-1)}(R^*, s'; \boldsymbol{\theta})$  for  $x \in \{c, \pi, y\}$  by four-dimensional Chebyshev polynomials. We have additional endogenous variables to be solved for; therefore, nonlinear optimization is even more costly than the stochastic neoclassical growth model. Furthermore, we compute numerical integrals with regard to  $s'$ , and the number of Gaussian quadrature points increases exponentially with increases in the number of exogenous shocks. For example, if we have three quadrature points for each shock, we have  $3^3 = 27$  quadrature points in total. To avoid costly nonlinear optimization and numerical integration, we can also apply the nonstochastic PEAs as before. Details are deferred to the Appendix A.2.

### 3.3 Index function approach to deal with the ZLB

The ZLB gives rise to kinks in the policy functions given the nonnegativity constraint on the nominal interest rate. Thus, it is not appropriate to directly apply global solution methods that presume smooth policy functions, such as Chebyshev polynomials.<sup>16</sup>

To deal with this issue, this subsection presents an index function approach as in Aruoba et al. (2018); Gust et al. (2017); Hirose and Sunakawa (2015, 2017); Nakata (2017). For each  $x \in \{c, \pi, y, R^*\}$ , let  $\sigma_{x,\text{NZLB}}(R_{-1}^*, s)$  be the policy function assuming that the ZLB does not bind and let  $\sigma_{x,\text{ZLB}}(R_{-1}^*, s)$  be the policy function assuming that the ZLB binds. Taking a pair of policy functions  $(\sigma_{x,\text{NZLB}}(R_{-1}^*, s), \sigma_{x,\text{ZLB}}(R_{-1}^*, s))$  as given, we use an index function to obtain

$$\sigma_x(R_{-1}^*, s) = \mathbb{I}_{(R^* < 1)} \sigma_{x,\text{ZLB}}(R_{-1}^*, s) + (1 - \mathbb{I}_{(R^* < 1)}) \sigma_{x,\text{NZLB}}(R_{-1}^*, s),$$

where the index function depends on the value of the current notional interest rate  $R^*$  and is defined

---

<sup>16</sup>Instead, we can use local approximation such as piecewise linear or cubic spline interpolation. In this case, as we approximate each interval between the consecutive grid points by separated polynomials, the effect of a kink from one interval to the other is limited.

as

$$\mathbb{I}_{(R^* < 1)} = \begin{cases} 1 & \text{when } R^* = \sigma_{R^*, \text{NZLB}}(R_{-1}^*, s) < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Then, the problem becomes finding a pair of tuples,  $(c_{jm\text{NZLB}}, \pi_{jm\text{NZLB}}, y_{jm\text{NZLB}}, R_{jm\text{NZLB}}^*)$  and  $(c_{jm\text{ZLB}}, \pi_{jm\text{ZLB}}, y_{jm\text{ZLB}}, R_{jm\text{ZLB}}^*)$ . Specifically, taking policy functions  $\sigma_x^{(i-1)}(R_{-1}^*, s)$  for  $x \in \{c, \pi, y\}$  as given, we solve

$$\begin{aligned} c_{jm\text{NZLB}} &= \beta \bar{\gamma}^{-1} R_{jm\text{NZLB}}^* \int \left[ \frac{\hat{\sigma}_c^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta})^{-\tau}}{z' \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta})} \right] p(s'|s) ds', \\ 0 &= \left( (1 - \nu^{-1}) + \nu^{-1} c_{jm\text{NZLB}}^\tau - \phi(\pi_{jm\text{NZLB}} - \bar{\pi}) \left[ \pi_{jm\text{NZLB}} - \frac{1}{2\nu} (\pi_{jm\text{NZLB}} - \bar{\pi}) \right] \right) c_{jm\text{NZLB}}^{-\tau} y_{jm\text{NZLB}} \\ &\quad + \beta \phi \int \left[ \hat{\sigma}_c^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta})^{-\tau} \hat{\sigma}_y^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta}) \right. \\ &\quad \times \left. \left( \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta}) - \bar{\pi} \right) \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{NZLB}}^*, s'; \boldsymbol{\theta}) \right] p(s'|s) ds', \\ y_{jm\text{NZLB}} &= \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm\text{NZLB}} - \bar{\pi})^2 \right]^{-1} c_{jm\text{NZLB}}, \\ R_{jm\text{NZLB}}^* &= \left( \bar{r} \bar{\pi} \left( \frac{\pi_{jm\text{NZLB}}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm\text{NZLB}}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^{*\rho_R} e^{\epsilon_{R,m}}. \end{aligned}$$

for  $(c_{jm\text{NZLB}}, \pi_{jm\text{NZLB}}, y_{jm\text{NZLB}}, R_{jm\text{NZLB}}^*)$ , and

$$\begin{aligned} c_{jm\text{ZLB}} &= \beta \bar{\gamma}^{-1} \int \left[ \frac{\hat{\sigma}_c^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta})^{-\tau}}{z' \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta})} \right] p(s'|s) ds', \\ 0 &= \left( (1 - \nu^{-1}) + \nu^{-1} c_{jm\text{ZLB}}^\tau - \phi(\pi_{jm\text{ZLB}} - \bar{\pi}) \left[ \pi_{jm\text{ZLB}} - \frac{1}{2\nu} (\pi_{jm\text{ZLB}} - \bar{\pi}) \right] \right) c_{jm\text{ZLB}}^{-\tau} y_{jm\text{ZLB}} \\ &\quad + \beta \phi \int \left[ \hat{\sigma}_c^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta})^{-\tau} \hat{\sigma}_y^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta}) \left( \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta}) - \bar{\pi} \right) \hat{\sigma}_\pi^{(i-1)}(R_{jm\text{ZLB}}^*, s'; \boldsymbol{\theta}) \right] p(s'|s) ds', \\ y_{jm\text{ZLB}} &= \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm\text{ZLB}} - \bar{\pi})^2 \right]^{-1} c_{jm\text{ZLB}}, \\ R_{jm\text{ZLB}}^* &= \left( \bar{r} \bar{\pi} \left( \frac{\pi_{jm\text{ZLB}}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm\text{ZLB}}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^{*\rho_R} e^{\epsilon_{R,m}}. \end{aligned}$$

for  $(c_{jm\text{ZLB}}, \pi_{jm\text{ZLB}}, y_{jm\text{ZLB}}, R_{jm\text{ZLB}}^*)$ .

To avoid costly nonlinear optimization and numerical integration, [Gust et al. \(2017\)](#) apply the nonstochastic Chebyshev PEA by fitting polynomials to future variables to solve a nonlinear New Keynesian model with the occasionally binding ZLB constraint. We cannot directly apply the nonstochastic Chebyshev PEA by fitting polynomials to current variables and the precomputation technique of [Judd et al. \(2017\)](#) when we deal with the occasionally binding constraint, because the expectation function is approximated by a piecewise polynomial based on two smooth Chebyshev polynomials in which we assume the constraint always or never binds. [Sunakawa \(2018\)](#) shows how to apply the nonstochastic PEA and precomputation technique with the index function approach by fitting polynomials to current variables.

### 3.4 Smolyak's method

The curse of dimensionality can be an issue when we solve large-scale DSGE models with nonlinear methods as they have a large number of state variables.

For example, the New Keynesian model that we have considered has four state variables. Assuming that we have only three grid points (with a second-order polynomial) for each dimension, we can then likely solve the model with four state variables and  $3^4 = 81$  grid points within a certain amount of computation time; however, we cannot as readily solve the model with many, say 10 state variables and  $3^{10} = 59,049$  grid points within a reasonable computation time. This is because the number of grid points increases exponentially with an increase in the model's dimensions, known as the curse of dimensionality. Moreover, even if we can solve the model within some amount of time, shortening computation time is important for estimating parameters in nonlinear models. This is the reason we introduce Smolyak's (1963) method with sparse grid points to handle models with high dimensions. Applications of Smolyak's method to New Keynesian models are found in Fernández-Villaverde et al. (2015); Gust et al. (2017); Hirose and Sunakawa (2015, 2017).<sup>17</sup>

Consider a bivariate function  $\sigma(x, y)$  and assume that we only know the values of the function at grid points of  $x \in \{x_1, \dots, x_{N_x}\}$  and  $y \in \{y_1, \dots, y_{N_y}\}$ . We interpolate functional values of  $\sigma(x, y)$  between the grid points by fitting the following polynomial

$$\hat{\sigma}(x, y; \boldsymbol{\theta}) = \theta_{0,0} + \theta_{1,0}T_1(x) + \theta_{2,0}T_2(x) + \theta_{0,1}T_1(y) + \theta_{0,2}T_2(y).$$

We have just eliminated all the cross terms in the example in Section 2.2 (the reason is below). There are 5 coefficients, so we need 5 collocation points. We select a subset of the tensor product of Chebyshev extrema as collocation points

$$(x, y) \in \{(0, 0), (-1, 0), (1, 0), (0, -1), (0, 1)\}.$$

Let  $(j(i), k(i))$  be a pair of functions that map the index of the collocation points  $i = 1, \dots, N$  to the index of the grid points we choose for each dimension. In this example, we choose  $(j(i), k(i)) \in \{(1, 1), (2, 1), (3, 1), (1, 2), (1, 3)\}$ . Once we have the collocation points  $\{x_{j(i)}, y_{k(i)}\}$  and the function values  $\{\sigma(x_{j(i)}, y_{k(i)})\}$  evaluated at  $(x_{j(i)}, y_{k(i)})$  for  $i = 1, 2, \dots, N$ , we fit  $\hat{\sigma}(x, y; \boldsymbol{\theta})$  to these values to obtain  $\boldsymbol{\theta}$ :

$$\begin{bmatrix} \sigma(x_1, y_1) \\ \sigma(x_2, y_1) \\ \sigma(x_3, y_1) \\ \sigma(x_1, y_2) \\ \sigma(x_1, y_3) \end{bmatrix} = \begin{bmatrix} 1 & T_1(x_1) & T_2(x_1) & T_1(y_1) & T_2(y_1) \\ 1 & T_1(x_2) & T_2(x_2) & T_1(y_1) & T_2(y_1) \\ 1 & T_1(x_3) & T_2(x_3) & T_1(y_1) & T_2(y_1) \\ 1 & T_1(x_1) & T_2(x_1) & T_1(y_2) & T_2(y_2) \\ 1 & T_1(x_1) & T_2(x_1) & T_1(y_3) & T_2(y_3) \end{bmatrix} \begin{bmatrix} \theta_{0,0} \\ \theta_{1,0} \\ \theta_{2,0} \\ \theta_{0,1} \\ \theta_{0,2} \end{bmatrix},$$

or

$$\sigma(\mathbf{x}, \mathbf{y}) = T(\mathbf{x}, \mathbf{y})\boldsymbol{\theta}.$$

Then, given that  $T(\mathbf{x}, \mathbf{y})$  is nonsingular, we have  $\boldsymbol{\theta} = T(\mathbf{x}, \mathbf{y})^{-1}\sigma(\mathbf{x}, \mathbf{y})$ .

**Unidimensional disjoint sets** How should we choose the basis functions and collocation points above? In the above example, we use tensor products of *unidimensional disjoint sets* of Chebyshev extrema and basis functions (Judd et al., 2014). We start with the following sets of unidimensional

---

<sup>17</sup>Krueger and Kubler (2004) first introduce Smolyak sparse grid points into macroeconomics by solving heterogeneous overlapping generations models with aggregate uncertainty.

grid points

$$\begin{aligned}
\mathcal{S}_1 &= \{0\}, \\
\mathcal{S}_2 &= \{0, -1, 1\}, \\
\mathcal{S}_3 &= \{0, -1, 1, \frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}, \\
\mathcal{S}_4 &= \{0, -1, 1, \frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{-\sqrt{2+\sqrt{2}}}{2}, \frac{-\sqrt{2-\sqrt{2}}}{2}, \frac{\sqrt{2+\sqrt{2}}}{2}, \frac{\sqrt{2-\sqrt{2}}}{2}\}, \\
&\vdots
\end{aligned}$$

These are sets of Chebyshev extrema at degrees  $N_i = 1, 3, 5, 9$  and satisfy the property of  $\mathcal{S}_i \subset \mathcal{S}_{i+1}$ . Then we have disjoint sets of the unidimensional grid points by  $\mathcal{A}_i = \mathcal{S}_i \setminus \mathcal{S}_{i-1}$ . That is,

$$\begin{aligned}
\mathcal{A}_1 &= \{0\}, \\
\mathcal{A}_2 &= \{-1, 1\}, \\
\mathcal{A}_3 &= \{\frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}, \\
\mathcal{A}_4 &= \{\frac{-\sqrt{2+\sqrt{2}}}{2}, \frac{-\sqrt{2-\sqrt{2}}}{2}, \frac{\sqrt{2+\sqrt{2}}}{2}, \frac{\sqrt{2-\sqrt{2}}}{2}\}, \\
&\vdots
\end{aligned}$$

By taking the tensor products of the unidimensional disjoint sets, we have

$$\begin{aligned}
\mathcal{A}_{1,1} &= \mathcal{A}_1 \otimes \mathcal{A}_1 = \{(0, 0)\}, \\
\mathcal{A}_{1,2} &= \mathcal{A}_1 \otimes \mathcal{A}_2 = \{(0, -1), (0, 1)\}, \\
\mathcal{A}_{2,1} &= \mathcal{A}_2 \otimes \mathcal{A}_1 = \{(-1, 0), (1, 0)\}, \\
\mathcal{A}_{2,2} &= \mathcal{A}_2 \otimes \mathcal{A}_2 = \{(-1, -1), (-1, 1), (1, -1), (1, 1)\}, \\
&\vdots
\end{aligned}$$

Similarly, we have disjoint sets of the basis functions

$$\begin{aligned}
\mathcal{F}_1(x) &= \{1\}, \\
\mathcal{F}_2(x) &= \{T_1(x), T_2(x)\}, \\
\mathcal{F}_3(x) &= \{T_3(x), T_4(x)\}, \\
\mathcal{F}_4(x) &= \{T_5(x), T_6(x), T_7(x), T_8(x)\} \\
&\vdots
\end{aligned}$$

and the tensor products of the unidimensional disjoint sets

$$\begin{aligned}
\mathcal{F}_{1,1}(x, y) &= \mathcal{F}_1(x) \otimes \mathcal{F}_1(y) = \{1\} \\
\mathcal{F}_{1,2}(x, y) &= \mathcal{F}_1(x) \otimes \mathcal{F}_2(y) = \{T_1(x), T_2(x)\} \\
\mathcal{F}_{2,1}(x, y) &= \mathcal{F}_2(x) \otimes \mathcal{F}_1(y) = \{T_1(y), T_2(y)\} \\
\mathcal{F}_{2,2}(x, y) &= \mathcal{F}_2(x) \otimes \mathcal{F}_2(y) = \{T_1(x)T_1(y), T_1(x)T_2(y), T_2(x)T_1(y), T_2(x)T_2(y)\} \\
&\vdots
\end{aligned}$$

Smolyak's algorithm chooses a subset of the tensor products as grid points and basis functions based on the following rule:  $d \leq i_x + i_y \leq d + \mu$  where  $i_l$ ,  $l \in \{x, y\}$  is an index for disjoint sets for each dimension,  $d$  is the number of dimensions, and  $\mu$  is known as the approximation level. For example, if  $d = 2$  and  $\mu = 1$ ,  $(i_x, i_y) \in \{(1, 1), (1, 2), (2, 1)\}$  and we construct grid points as  $\mathcal{A}_{1,1} \cup \mathcal{A}_{1,2} \cup \mathcal{A}_{2,1} = \{(0, 0), (0, -1), (0, 1), (-1, 0), (1, 0)\}$  and basis functions as  $\mathcal{F}_{1,1}(x, y) \cup \mathcal{F}_{1,2}(x, y) \cup \mathcal{F}_{2,1}(x, y) = \{1, T_1(x), T_2(x), T_1(y), T_2(y)\}$ .<sup>18</sup>

Having the grid points and basis functions at hand, we can construct a polynomial. Judd et al. (2014) provide a general framework called Lagrange interpolation. Let  $f : [-1, 1]^d \rightarrow \mathbb{R}$  a smooth function defined on a normalized  $d$  dimensional hypercube and let  $\hat{f}(\cdot; \boldsymbol{\theta})$  be a polynomial function of the form

$$\hat{f}(\mathbf{x}; \boldsymbol{\theta}) = \sum_{i=1}^N \theta_i \Psi_i(\mathbf{x})$$

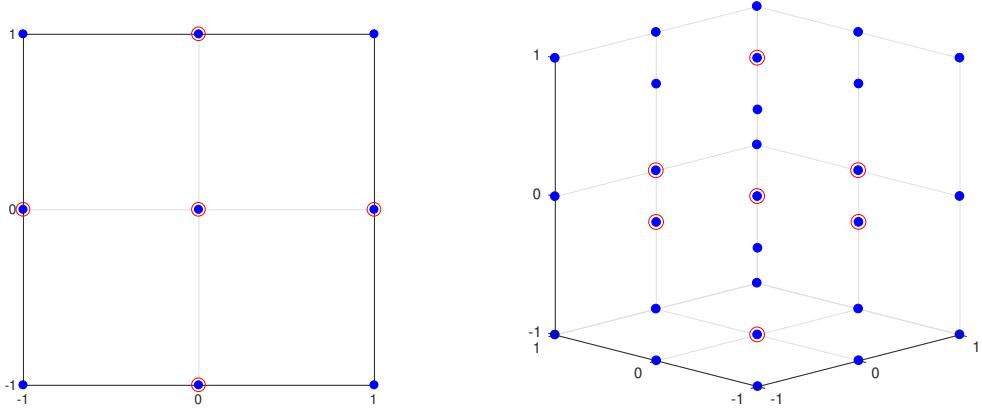
In this framework, we need to choose (i)  $N$  grid points and (ii)  $N$  basis functions, and (iii) to compute  $\boldsymbol{\theta}$  that makes  $f(x_i)$  coincide with  $\hat{f}(x_i; \boldsymbol{\theta})$  in all the grid points.

It is easy to extend the above algorithm to higher dimensions. Figure 2 depicts the cases of two and three dimensions (i.e.,  $d = 2$  and 3) with second-order polynomials and  $N_l = 3$  for  $l = 1, \dots, d$ . The total number of the grid points with the Smolyak's method is  $1 + 2d$ , whereas it is  $3^d$  with the standard method. As shown in Table 2, for example, there are just 21 grid points for the 10 state variables using the Smolyak's method, whereas there would be 59,049 grid points using the standard method. It is also straightforward to extend the algorithm to higher-order polynomials such as  $N_l = 5, 9, \dots$ . See Judd et al. (2014) for a more detailed discussion.

---

<sup>18</sup>By exploiting the structure of disjoint sets as in Judd et al. (2014), we have no duplication between the sets of tensor products and therefore compute the Smolyak basis functions and grid points more efficiently.

Figure 2: Sparse grid points



$$\begin{aligned} \text{Left : } (x, y) &\in \{(0, 0), (1, 0), (-1, 0), (0, 1), (0, -1)\} \\ \text{Right : } (x, y, z) &\in \{(0, 0, 0), (1, 0, 0), (-1, 0, 0), \\ &\quad (0, 1, 0), (0, -1, 0), (0, 0, 1), (0, 0, -1)\} \end{aligned}$$

Notes: Red circles show the sparse grid points chosen by Smolyak's algorithm, whereas blue dots show the tensor product of Chebyshev extrema with  $N_l = 3$  for each dimension.

Table 2: Number of grid points with  $N_l = 3$

$d$	$1 + 2d$	$3^d$
2	5	9
3	7	27
4	9	81
5	11	243
$\vdots$	$\vdots$	$\vdots$
10	21	59,049
20	41	3,486,784,401

### 3.5 Simulation-based method

Another widely used method to deal with the curse of dimensionality is the simulation-based method recently developed by Judd et al. (2011) and Maliar and Maliar (2015; hereafter MM). The simulation-based method stems from the original work of Marcer's (1988) PEA. Applications to New Keynesian models are found in Maliar and Maliar (2015); Lepetuyk et al. (2017); Aruoba et al. (2018); Hills et al. (2018).

We solve for the policy functions on simulation-based grid points based on the ergodic distribution of the state variables. The key to the simulation-based method is to construct a small number of grid points that represent the ergodic distribution of the state variables in DSGE models. The ergodic distribution in DSGE models is usually only a part of a state-space hypercube. Therefore, if we use nonstochastic and nonadaptive grid point methods with a state-space hypercube, the state space may include the region rarely visited in equilibrium dynamics. For example, Judd

et al. (2011) demonstrates that the ratio of the volume of a dimensional sphere with a diameter of one (which presumably represents the ergodic distribution of the state variables) to that of a dimensional hypercube with a width of one (which represents the rectangle state space we typically assume) is only 31%; the ratio rapidly decreases in the number of dimensions.

This subsection presents an  $\epsilon$ -distinguishable set (EDS) method developed by MM.<sup>19</sup> In short, to construct an EDS grid from the ergodic set, the procedure takes the following two steps (See MM for details):

1. Select points within an essentially ergodic set (called Algorithm  $\mathcal{A}^\eta$  in MM)
2. Construct a uniformly spaced set of points that covers the essentially ergodic set (called Algorithm  $P^\epsilon$  in MM)

In step 1, the essentially ergodic set is defined by a threshold parameter  $\eta$  and the density function  $g(x)$  of the original ergodic set from the simulated data. We select only some points in the ergodic set so that  $g(x) \geq \eta$  holds to exclude points unlikely visited in an equilibrium.<sup>20</sup> Having the essentially ergodic set at hand, in step 2, we construct uniformly spaced grid points that cover the essentially ergodic set. That is, two points  $(x_i, x_j)$  in the essential ergodic set are  $\epsilon$ -distinguishable when the distance between two points  $D(x_i, x_j)$  is greater than a parameter  $\epsilon$ , that is,  $D(x_i, x_j) > \epsilon$  holds.<sup>21</sup> In practice, we set the number of grid points  $M$  by adjusting the parameter  $\epsilon$ .

We now merge the simulation-based sparse grid and the TI method. First, we initialize the EDS grid using the simulated data from an approximated solution such as that obtained by linear approximation. Second, we solve for the policy function on the EDS grid using the TI method. Third, using the policy function, we update the EDS grid with the simulated data. We repeat these processes until the EDS grid converges. Specifically, we have the following steps:

0. Initialization
  - (a) Choose initial values  $(R_{-1}^*, s_0)$  and simulation length,  $T$ .
  - (b) Draw a sequence of  $\{s_t\}_{t=1}^T$  and fix the sequence throughout the iterations.
  - (c) Choose approximating policy functions  $\hat{\sigma}(R_{-1}^*, s; \boldsymbol{\theta})$  and make an initial guess of  $\boldsymbol{\theta}$ . We use second-order polynomials with cross terms for  $\hat{\sigma}$ .
1. Construction of an EDS grid
  - (a) Given  $(R_{-1}^*, s_0)$  and  $\{s_t\}_{t=1}^T$ , use  $\hat{\sigma}(R_{-1}^*, s; \boldsymbol{\theta})$  to simulate  $\{R_{t-1}^*\}_{t=1}^T$ .
  - (b) Construct an EDS grid  $\Gamma \equiv \{R_{-1,m}^*, s_m\}_{m=1}^M$  from the simulated sequence of  $\{R_{t-1}^*, s_t\}_{t=1}^T$ .
2. Computation of a solution on the EDS grid,  $\hat{\sigma}(R_{-1}^*, s; \boldsymbol{\theta})$ , using the TI method.
3. Repeat steps 2–3 until convergence of the EDS grid.

---

<sup>19</sup>MM also consider locally-adaptive EDS and cluster grid methods.

<sup>20</sup>We also sample from the simulated data with some interval to eliminate the effect of serial correlation.

<sup>21</sup>We take an  $L_2$  norm as the distance. Also, the distance function should be unit-free, whereas variables in economic models have different measures and are correlated with each other. Therefore we need to normalize and orthogonalize the simulated data.

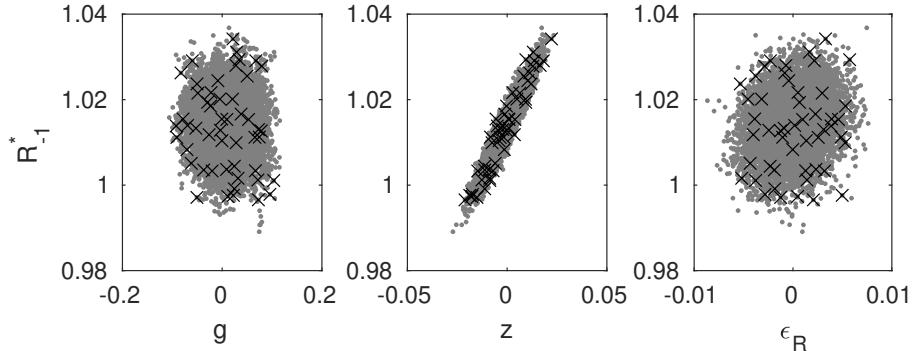
Regarding convergence, MM suggest the following criterion. Let  $\Gamma \equiv \{x_m\}_{m=1}^M$  be a set of EDS grid points. We minimize the maximum distance between a point from the old EDS grid  $x' \in \Gamma' \equiv \{x'_m\}_{m=1}^{M'}$  and a point from the new EDS grid  $x'' \in \Gamma'' \equiv \{x''_m\}_{m=1}^{M''}$ . That is,  $\sup_{x' \in \Gamma'} \inf_{x'' \in \Gamma''} D(x', x'') < 2\epsilon$  where  $\epsilon$  is used in the second step in the two-step procedure.<sup>22</sup> Usually the EDS grid converges after a few iterations.

Figure 3 illustrates the EDS grid and the ergodic distribution of each of the state variables in the New Keynesian model with and without the ZLB. In each panel, the x-axis is the exogenous state  $x \in \{g, z, \epsilon_R\}$  and the y-axis is the endogenous state  $R_{-1}^*$ . As the theory suggests, the EDS grid uniformly covers the ergodic distribution. In addition, the number of EDS grid points is small (as we choose the number by controlling  $\epsilon$ ) so that the TI method is quick. This is quite important as we repeat the TI until the EDS grid converges.

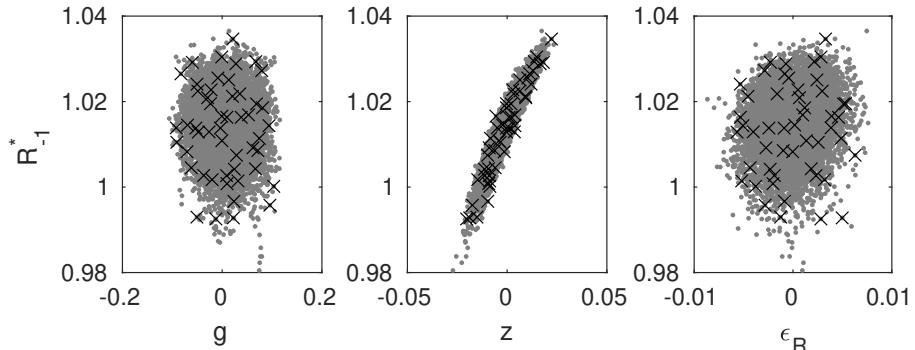
The advantage of the simulation-based method over Smolyak's method is its flexibility in locating grid points. As pointed out by MM, it is not necessary that the EDS grid is an efficient way to choose the grid points for approximating the policy function. MM argue that a locally adaptive EDS grid is useful to obtain a more precise approximation to the policy function (i.e., smaller residual equation errors). Aruoba et al. (2018) merge the EDS grid with the observed data points. This augmented grid covers some regions in the state space that the endogenous dynamics of the model scarcely explain and improves the fit of the model to the data.

Figure 3: Simulated grid points

a. Without ZLB



b. With ZLB



<sup>22</sup> $2\epsilon$  is the maximum distance in theory between grid points within a set of EDS grids. See Proposition 3 in Appendix A.3 in MM.

Table 3: Parameter values of New Keynesian model

Parameter	Value	
$\nu$	Inverse of demand elasticity	1/6
$\bar{g}$	Steady-state government expenditure	1.25
$\gamma$	Steady-state technology growth	1.0052
$\beta$	Discount factor	0.9990
$\bar{\pi}$	Steady-state inflation	1.0083
$\tau$	CRRA parameter	2.83
$\phi$	Price adjustment cost	17.85
$\psi_1$	Interest rate elasticity to inflation	1.80
$\psi_2$	Interest rate elasticity to output gap	0.63
$\rho_r$	Interest rate smoothing	0.77
$\rho_g$	Persistence of government shock	0.98
$\rho_z$	Persistence of technology growth shock	0.88
$\sigma_r$	Std. dev. of monetary policy shock	0.0022
$\sigma_g$	Std. dev. of government shock	0.0071
$\sigma_z$	Std. dev. of technology growth shock	0.0031

### 3.6 Numerical examples

For numerical illustration, the New Keynesian model presented in Section (3.1) is parameterized according to the values in Table 3. These values except for  $(\nu, \bar{g})$  are taken from the parameter estimates of the log-linearized version of the model in [Herbst and Schorfheide \(2015\)](#).<sup>23</sup> We use the second- and fourth-order Chebyshev polynomials for  $R$  and  $s = (z, g, \epsilon_r)$  for interpolation in each solution algorithm. The number of grid points for each polynomial case are  $3^4 = 81$  and  $5^4 = 625$ . We also use the Smolyak algorithm, in which the number of grid points are 9 and 41 (see [Judd et al., 2014; Gust et al., 2017](#)). The number of the Gaussian–Hermite quadrature is set to  $3^3 = 27$ .

The Euler equation errors are given by

$$\mathcal{E}_c(R_{-1}, s) = 1 - \beta \bar{\gamma}^{-1} R \int \left\{ \left( \frac{\sigma_c(R, s')}{\sigma_c(R_{-1}, s)} \right)^{-\tau} \frac{1}{z' \sigma_\pi(R, s')} \right\} p(s'|s) ds' \quad (15)$$

$$\begin{aligned} \mathcal{E}_\pi(R_{-1}, s) = & (1 - \nu^{-1}) + \nu^{-1} \sigma_c(R_{-1}, s)^{-\tau} \\ & - \phi (\sigma_\pi(R_{-1}, s) - \bar{\pi}) \left[ \sigma_\pi(R_{-1}, s) - \frac{1}{2\nu} (\sigma_\pi(R_{-1}, s) - \bar{\pi}) \right] \\ & + \beta \phi \int \left\{ \left( \frac{\sigma_c(R, s')}{\sigma_c(R_{-1}, s)} \right)^{-\tau} \frac{y'}{y} (\sigma_\pi(R, s') - \bar{\pi}) \sigma_\pi(R, s') \right\} p(s'|s) ds', \end{aligned} \quad (16)$$

<sup>23</sup>The parameter for price adjustment cost  $\phi$  is obtained by setting the values of the elasticity of demand  $\nu^{-1} = 6$  and the slope of the New Keynesian Phillips curve (which is a composite function of the parameters)  $\kappa = \frac{\tau(\nu^{-1}-1)}{\bar{\pi}^2 \phi} = 0.78$ . The steady-state government expenditure shock is given by  $\bar{g} = (1 - g_y)^{-1}$ , where  $g_y$  is the ratio of government expenditure to output and set at 0.2.

where

$$y = \left( g^{-1} - \frac{\phi}{2} (\sigma_\pi(R_{-1}, s) - \bar{\pi})^2 \right)^{-1} \sigma_c(R_{-1}, s), \quad (17)$$

$$R = \left( \bar{r}\bar{\pi} \left( \frac{\sigma_\pi(R_{-1}, s)}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{-1}^{\rho_R} e^{\epsilon_R}. \quad (18)$$

Note that the last two equations of the static equilibrium conditions (17)-(18) hold with equality.<sup>24</sup> The Euler equation errors are calculated based on the series simulated from the approximated solutions. Each stochastic simulation is done for 10,500 periods, and the first 500 are discarded. The same sequence of random variables for  $s$  is used throughout all simulations.

Table 4 compares the performance of the different methods (TI, future PEA, and current PEA) and polynomials (second-order Chebyshev, second-order Smolyak, and fourth-order Smolyak) in terms of accuracy as measured by residual function errors in equation (15) and (16), the simulated moments of output growth, inflation, and interest rates, and the computation time (in seconds). Table 4.a provides the results in the case without the ZLB. First, we can see that future PEA is faster than TI and that current PEA is faster than the future PEA and TI. This is because future PEA avoids only nonlinear optimization, while current PEA avoids both numerical optimization and numerical integration, as in the previous example using the stochastic neoclassical growth model. In addition, the use of Smolyak polynomials makes the computation even faster by making the number of grid points smaller. Second, TI, future PEA, and current PEA yield similar second-order simulated moments. However, the inflation rate and the policy rate are a bit more volatile in future PEA. Finally, TI, future PEA, and current PEA are also comparable in terms of accuracy. The use of second-order Smolyak polynomials makes the errors worse compared with second-order Chebyshev polynomials, but the fourth-order Smolyak polynomials are comparable with the second-order Chebyshev polynomials. Note that the computation time for the methods with fourth-order Smolyak polynomials is still about 5–6 times shorter than that for the methods with second-order Chebyshev polynomials.

Table 4.b details the results in the case with the ZLB. We observe the same pattern in terms of comparison of the computation time, the simulated moments, and accuracy of the different methods and polynomials as in the case without the ZLB. It takes longer to solve the model with the ZLB using the index function approach. The overall accuracy is also a bit poorer compared with the case without the ZLB but still reasonable. The probability of binding the ZLB is similar for the different methods, but the methods with second-order Smolyak polynomials have a higher probability of binding the ZLB. If we use fourth-order polynomials, we can solve the model at an acceptable accuracy within less than a second by using current PEA.

Table 5 compares the performance of TI and future PEA based on the simulation-based method for the different number of grid points  $M = 25, 50$ , and  $100$ . This choice does not matter for accuracy and simulated moments, and an increase in  $M$  merely makes the computation more costly. Future PEA is faster than TI as before, but the deterioration of accuracy is more severe in the case with the ZLB. Further, the inflation and policy rates are a little more volatile compared with the nonstochastic methods in the case with the ZLB.

---

<sup>24</sup> $y'$  in (16) is calculated by substituting  $R$  implied by (18) into  $R_{-1}^*$  in (17).

Table 4: Accuracy and speed of TI, future PEA, current PEA: New Keynesian model

a. Without ZLB

Polynomial				TI				
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	CPU
2nd	-4.15	-2.45	-3.47	-1.79	0.76	1.93	2.36	318.07
2nd, Smolyak	-3.45	-2.35	-2.34	-1.31	0.76	1.97	2.42	3.71
4th, Smolyak	-5.09	-3.73	-3.72	-2.57	0.76	1.99	2.43	61.94
Polynomial				future PEA				
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	CPU
2nd	-4.86	-2.76	-3.52	-2.04	0.76	2.01	2.46	20.90
2nd, Smolyak	-3.32	-2.66	-2.21	-1.63	0.76	2.13	2.59	0.29
4th, Smolyak	-5.03	-3.69	-3.71	-2.69	0.76	2.00	2.44	4.06
Polynomial				current PEA				
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	CPU
2nd	-4.32	-2.46	-3.30	-1.75	0.76	1.92	2.35	1.39
2nd, Smolyak	-3.36	-2.36	-2.21	-1.34	0.76	1.98	2.43	0.04
4th, Smolyak	-4.93	-3.75	-3.56	-2.53	0.76	1.99	2.43	0.32

b. With ZLB

Polynomial				TI					
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	$\text{Pr}_{(R^*<1)}$	CPU
2nd	-3.73	-2.62	-2.07	-1.42	0.76	2.05	2.53	1.53	1127.3
2nd, Smolyak	-3.40	-2.38	-2.06	-1.09	0.76	2.02	2.50	1.79	12.98
4th, Smolyak	-3.97	-3.14	-2.07	-1.73	0.76	2.04	2.50	1.40	270.65
Polynomial				future PEA					
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	$\text{Pr}_{(R^*<1)}$	CPU
2nd	-3.73	-2.67	-2.08	-1.44	0.76	2.11	2.59	1.71	82.28
2nd, Smolyak	-3.26	-2.66	-1.92	-1.48	0.76	2.19	2.68	3.42	0.96
4th, Smolyak	-4.04	-3.54	-2.13	-1.49	0.76	2.03	2.48	1.18	14.66
Polynomial				current PEA					
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	$\text{Pr}_{(R^*<1)}$	CPU
2nd	-4.05	-2.71	-2.12	-1.53	0.76	2.01	2.45	1.03	4.05
2nd, Smolyak	-3.35	-2.42	-1.97	-1.25	0.76	2.01	2.47	1.92	0.19
4th, Smolyak	-4.17	-2.95	-2.12	-1.44	0.76	2.03	2.47	1.07	0.99

Notes:  $L_{1,c}$ ,  $L_{1,\pi}$ ,  $L_{\infty,c}$ , and  $L_{\infty,\pi}$  are the averages and maximums of the Euler equation errors in absolute terms (15)–(16) (log 10 units) based on a 10,000-period stochastic simulation, respectively. CPU is the elapsed time for computing the equilibrium (in seconds).  $\sigma_{\Delta y}$ ,  $\sigma_\pi$ , and  $\sigma_R$  are the standard deviations of output growth, inflation, and the policy rate, respectively.  $\text{Pr}_{(R^*<1)}$  is the probability of the ZLB binding. Computation is done in MATLAB R2016b using a laptop with Xeon E3-1505Mv5 (2.8 GHz) and 16 GB memory without any parallelization.

Table 5: Accuracy and speed of TI, future PEA, current PEA: New Keynesian model (Simulation-based method)

a. Without ZLB

M	TI							
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	CPU
25	-5.08	-4.09	-3.94	-2.84	0.76	1.99	2.43	7.48
50	-5.02	-4.20	-3.77	-2.75	0.76	2.00	2.44	18.73
100	-5.20	-4.25	-3.82	-2.81	0.76	2.00	2.44	28.61
M	future PEA							
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$	CPU
25	-5.07	-3.74	-3.90	-2.69	0.76	1.99	2.43	1.59
50	-4.94	-3.73	-3.95	-2.62	0.76	2.00	2.44	2.32
100	-5.08	-3.76	-3.91	-2.67	0.76	2.00	2.44	3.99

b. With ZLB

M	TI							PrZLB	CPU
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$		
25	-3.39	-2.93	-2.21	-1.66	0.77	2.14	2.63	1.21	44.41
50	-3.00	-2.56	-1.83	-1.48	0.77	2.21	2.71	1.92	65.37
100	-3.28	-2.80	-2.08	-1.69	0.76	2.10	2.57	1.25	106.17
M	future PEA							PrZLB	CPU
	$L_{1,c}$	$L_{1,\pi}$	$L_{\infty,c}$	$L_{\infty,\pi}$	$\sigma_{\Delta y}$	$\sigma_\pi$	$\sigma_R$		
25	-3.02	-2.50	-1.94	-1.38	0.77	2.18	2.68	1.31	3.37
50	-3.19	-2.58	-2.00	-1.34	0.77	2.15	2.63	1.46	5.08
100	-3.25	-2.68	-2.06	-1.44	0.76	2.11	2.58	1.31	7.93

Notes:  $L_{1,c}$ ,  $L_{1,\pi}$ ,  $L_{\infty,c}$ , and  $L_{\infty,\pi}$  are the averages and maximums of the Euler equation errors in absolute terms (15)–(16) (log 10 units) based on a 10,000-period stochastic simulation, respectively. CPU is the elapsed time for computing the equilibrium (in seconds).  $\sigma_{\Delta y}$ ,  $\sigma_\pi$ , and  $\sigma_R$  are the standard deviations of output growth, inflation, and the policy rate, respectively.  $\text{Pr}_{(R^* < 1)}$  is the probability of the ZLB binding. Computation is done in MATLAB R2016b using a laptop with Xeon E3-1505Mv5 (2.8 GHz) and 16 GB memory without any parallelization.

## 4 Estimation Methods for Nonlinear DSGE Models

While the preceding sections have demonstrated how to solve a nonlinear DSGE model given a set of parameters, this section explains how to estimate the parameters of a model solved with nonlinear techniques. According to the previous sections, the solution, i.e., the policy function, of a DSGE model takes a general nonlinear form

$$s_t = \Phi(s_{t-1}, \varepsilon_t; \theta), \quad \varepsilon_t \sim F_\varepsilon(\cdot; \theta), \quad (19)$$

where  $s_t$  denotes a vector of endogenous variables,  $\varepsilon_t$  is a vector of exogenous disturbances to fundamental shocks, and  $\theta$  is a set of model parameters. The function  $\Phi(s_{t-1}, \varepsilon_t; \theta)$  is obtained numerically by nonlinear solution methods, whereas  $F_\varepsilon(\cdot; \theta)$  represents a generic probability distribution function for  $\varepsilon_t$ . The latent states (model variables)  $s_t$  are related to a set of observed

variables  $y_t$  by the observation equations

$$y_t = \Psi(s_t; \theta) + u_t, \quad u_t \sim F_u(\cdot; \theta), \quad (20)$$

where the function  $\Psi(s_t; \theta)$  links a subset of  $s_t$  to the data and  $u_t$  is a vector of measurement errors with a generic probability distribution function  $F_u(\cdot; \theta)$ . The state-space representation given by the state transition equations (19) and the observation equations (20) provides a joint density

$$p(Y_{1:T}, S_{1:T} | \theta) = \prod_{t=1}^T p(y_t, s_t | Y_{1:t-1}, S_{1:t-1}, \theta)$$

for the latent states and observations given the parameters, thereby allowing us to evaluate the likelihood function

$$p(Y_{1:T} | \theta) = \prod_{t=1}^T p(y_t | Y_{1:t-1}, \theta)$$

by integrating out the latent states, where  $Y_{1:t} = \{y_1, y_2, \dots, y_t\}$  and  $S_{1:t} = \{s_1, s_2, \dots, s_t\}$ .

If both  $\Phi(s_{t-1}, \varepsilon_t; \theta)$  and  $\Psi(s_t; \theta)$  are linear functions and both  $F_\varepsilon(\cdot; \theta)$  and  $F_u(\cdot; \theta)$  are normal distributions, then the Kalman filter can exactly evaluate the likelihood function  $p(Y_{1:T} | \theta)$  in a very efficient manner.<sup>25</sup> However, the Kalman filter is not applicable when the solution of a DSGE model is of nonlinear form or when the shocks are nonnormal. A challenging task in such a case is to approximate the likelihood function associated with a nonlinear state-space model with possibly nonnormal shocks. To this end, we subsequently present two methods that approximate the likelihood function of nonlinear DSGE models: a particle filter and the central difference Kalman filter (CDKF).

Once we approximate and evaluate the likelihood function  $p(Y_{1:T} | \theta)$ , we can, in principle, estimate the parameters of a DSGE model using a (quasi-)maximum likelihood approach. However, it turns out to be very difficult to obtain plausible parameter estimates that fall into such bounded domains as economic theory or empirical observation would suggest. To deal with this problem, many researchers now rely on the Bayesian approach to estimate DSGE models. In the Bayesian approach, a prior distribution  $p(\theta)$  is combined with the likelihood function  $p(Y_{1:T} | \theta)$  to form a posterior distribution  $p(\theta | Y_{1:T})$ , based on Bayes' theorem

$$p(\theta | Y_{1:T}) = \frac{p(Y_{1:T} | \theta)p(\theta)}{p(Y_{1:T})},$$

where  $p(Y_{1:T}) = \int p(Y_{1:T} | \theta)p(\theta) d\theta$ , and inferences on parameters are made based on the posterior distribution. An issue here is that the posterior distribution of parameters cannot be derived analytically, even in a very simple DSGE model. Although this issue is inherent not only in the estimation of nonlinear DSGE models, but also in linear forms, this section presents two important algorithms to generate parameters drawn from posterior distributions: the Markov chain Monte Carlo (MCMC) and the sequential Monte Carlo (SMC).

## 4.1 Particle filter

For nonlinear state-space models, the Kalman filter is not available because the distribution of  $y_t | Y_{1:t-1}$  is nonnormal. The idea of a particle filter is that the distribution of  $y_t | Y_{1:t-1}$  can be

---

<sup>25</sup>For a textbook treatment of the Kalman filter, see [Hamilton \(1994\)](#).

approximated by propagating particles representing  $s_t$  according to the state-space representation given by (19) and (20). While there are many particle filters, we focus on the *bootstrap particle filter* as in Gordon et al. (1993), because it is the most basic filter and widely used in the estimation of nonlinear DSGE models. The exposition below follows from Herbst and Schorfheide (2015).

The bootstrap particle filter proceeds with the following steps.

**Initialization:**

1. Draw the initial particles  $\{s_0^{(j)}\}_{j=1}^M$  from the distribution

$$s_0^{(j)} \sim N(\bar{s}_0, P_0),$$

where  $M$  denotes the number of particles, and  $\bar{s}_0$  and  $P_0$  are set equal to the unconditional first and second moments of  $s_t$  implied by the nonlinear solution (19).

2. Initialize particle weights  $W_0^{(j)} = 1$ , for  $j = 1, \dots, M$ .

**Recursion:** For  $t = 1, \dots, T$ :

1. Draw shocks in period  $t$  from their distribution functions

$$\varepsilon_t^{(j)} \sim F_\varepsilon(\cdot; \theta),$$

and propagate particles  $\{s_{t-1}^{(j)}\}$  using the nonlinear solution (19),

$$\tilde{s}_t^{(j)} = \Phi(s_{t-1}^{(j)}, \varepsilon_t^{(j)}; \theta).$$

2. Define the incremental weights as

$$\tilde{w}_t^{(j)} = p(y_t | \tilde{s}_t^{(j)}, \theta),$$

for  $j = 1, \dots, M$ . If the measurement errors  $u_t \sim N(0, \Sigma_u)$ , the observation equations (20) allow us to evaluate the incremental weights by

$$\tilde{w}_t^{(j)} = (2\pi)^{-\frac{n}{2}} |\Sigma_u|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (y_t - \Psi(\tilde{s}_t^{(j)}; \theta))' \Sigma_u^{-1} (y_t - \Psi(\tilde{s}_t^{(j)}; \theta)) \right\},$$

where  $n$  denotes the number of observables. Then, the predictive density  $p(y_t | Y_{1:t-1}, \theta)$  can be approximated by

$$\hat{p}(y_t | Y_{1:t-1}, \theta) = \frac{1}{M} \sum_{j=1}^M \tilde{w}_t^{(j)} W_{t-1}^{(j)}.$$

3. Define the normalized weights as

$$\tilde{W}_t^{(j)} = \frac{\tilde{w}_t^{(j)} W_{t-1}^{(j)}}{\frac{1}{M} \sum_{j=1}^M \tilde{w}_t^{(j)} W_{t-1}^{(j)}},$$

for  $j = 1, \dots, M$ , and then an approximation of  $\hat{\mathbb{E}}[s_t|Y_{1:t}, \theta]$  is given by

$$\hat{\mathbb{E}}[s_t|Y_{1:t}, \theta] = \frac{1}{M} \sum_{j=1}^M \tilde{s}_t^{(j)} \tilde{W}_t^{(j)}.$$

4. If resampling is needed, draw particles  $\{s_t^{(j)}\}_{j=1}^M$  from a multinomial distribution characterized by support points and weights  $\{\tilde{s}_t^{(j)}, \tilde{W}_t^{(j)}\}$  and set  $W_t^{(j)} = 1$ , for  $j = 1, \dots, M$ . If resampling is not needed, let  $s_t^{(j)} = \tilde{s}_t^{(j)}$  and  $W_t^{(j)} = \tilde{W}_t^{(j)}$  for  $j = 1, \dots, M$ . (We discuss the resampling step in more detail below.)
5. Repeat steps 1–4 for the next  $t$ .

The resampling in step 4 is necessary to avoid the degeneracy of the distribution of particle weights, i.e., a situation in which all but a few of the weights are near zero. Given that resampling is done with replacement, a particle with a large weight is likely to be drawn many times, whereas particles with small weights are not likely to be drawn at all. Thus, the resampling step effectively deals with the degeneracy problem by eliminating particles with very small weights. The judgment of whether resampling is needed is made based on the following effective sample size (ESS) of the particle weights

$$ESS_t = \frac{M}{\frac{1}{M} \sum_{j=1}^M (\tilde{W}_t^{(j)})^2}.$$

The ESS takes values between 1 and  $M$ , and resampling is done when it falls below some threshold, typically  $M/2$ .

**Likelihood approximation:** The recursion steps above generate particle weights  $\{\tilde{w}_t^{(j)}, W_{t-1}^{(j)}\}_{j=1}^M$  for each  $t$ . The approximation of the log-likelihood function is given by

$$\ln \hat{p}(Y_{1:T}|\theta) = \sum_{t=1}^T \ln \left( \frac{1}{M} \sum_{j=1}^M \tilde{w}_t^{(j)} W_{t-1}^{(j)} \right).$$

## 4.2 Central difference Kalman filter (CDKF)

Likelihood approximation using a particle filter results in a huge computational cost because many iterations are required for all of the particles and time periods. However, if a nonlinear DSGE model is approximated by a second- or third-order perturbation method, the CDKF, proposed by [Andreasen \(2013\)](#), can approximate the likelihood function more efficiently. [Andreasen \(2013\)](#) shows that a quasi-maximum likelihood estimator based on the CDKF can be consistent and asymptotically normal for DSGE models solved up to the third order. The subsequent exposition follows from [Andreasen \(2013\)](#).

We begin by considering a general filtering problem. The objective of filtering is to estimate the state variables  $s_t$  as data  $y_t$  become available and thereby evaluate the likelihood function. Specifically, filtering recursively computes the mean of the state variables and its covariance matrix based on a two-step procedure of prediction and updating. Suppose that the state-space representation is given by (19) and (20). To simplify notation, ex-ante and ex-post estimates are respectively denoted by a bar and a hat; for instance,  $\bar{s}_{t+1} = \mathbb{E}[s_{t+1}|Y_{1:t}]$  and  $\hat{s}_{t+1} = \mathbb{E}[s_{t+1}|Y_{1:t+1}]$ . Then, the

prediction of  $s_{t+1}$  and  $y_{t+1}$  are given by

$$\bar{s}_{t+1} = \mathbb{E}[\Phi(s_t, \varepsilon_{t+1}; \theta) | Y_{1:t}], \quad (21)$$

$$\bar{y}_{t+1} = \mathbb{E}[\Psi(s_{t+1}; \theta) | Y_{1:t}], \quad (22)$$

and their conditional error covariance matrices are denoted by

$$\bar{P}_{ss}(t+1) = \mathbb{E}[(s_{t+1} - \bar{s}_{t+1})(s_{t+1} - \bar{s}_{t+1})' | Y_{1:t}], \quad (23)$$

$$\bar{P}_{yy}(t+1) = \mathbb{E}[(y_{t+1} - \bar{y}_{t+1})(y_{t+1} - \bar{y}_{t+1})' | Y_{1:t}]. \quad (24)$$

The updating rule is

$$\hat{s}_{t+1} = \bar{s}_{t+1} + K_{t+1}(y_{t+1} - \bar{y}_{t+1}), \quad (25)$$

where  $K_{t+1}$  is called the Kalman gain and determined such that the conditional forecast errors for  $\hat{s}_{t+1}$  are minimized. It is known that  $K_{t+1}$  is given by

$$K_{t+1} = P_{sy}(t+1)\bar{P}_{yy}(t+1)^{-1}, \quad (26)$$

where

$$P_{sy}(t+1) = \mathbb{E}[(s_{t+1} - \bar{s}_{t+1})(y_{t+1} - \bar{y}_{t+1})' | Y_{1:t}]. \quad (27)$$

The ex-post conditional error covariance matrix for the state variables can be expressed as

$$\hat{P}_{ss}(t+1) = \bar{P}_{ss}(t+1) - K_{t+1}\bar{P}_{yy}(t+1)K'_{t+1} \quad (28)$$

If  $\Phi(\cdot)$  and  $\Psi(\cdot)$  are linear functions, the filtering equations (21)–(28) exactly evaluate the first and second moments of  $s_{t+1}$  and  $y_{t+1}$ , and this leads to the Kalman filter. The idea behind the CDKF is to approximate the filtering equations up to second order using multivariate Stirling interpolations. Let  $\Sigma_\varepsilon$  and  $\Sigma_u$  denote the variance covariance matrices of the fundamental shocks  $\varepsilon_t$  and the measurement errors  $u_t$ , and let  $S_\varepsilon(t) = [s_{\varepsilon,1}, s_{\varepsilon,2}, \dots, s_{\varepsilon,n_\varepsilon}]$ ,  $S_u(t) = [s_{u,1}, s_{u,2}, \dots, s_{u,n_u}]$ ,  $\bar{S}_s(t) = [\bar{s}_{s,1}, \bar{s}_{s,2}, \dots, \bar{s}_{s,n_s}]$ , and  $\hat{S}_s(t) = [\hat{s}_{s,1}, \hat{s}_{s,2}, \dots, \hat{s}_{s,n_s}]$  be upper triangular Cholesky factorizations of  $\Sigma_\varepsilon(t)$ ,  $\Sigma_u(t)$ ,  $\bar{P}_{ss}(t)$ , and  $\hat{P}_{ss}(t)$ , respectively, with lowercase letters denoting their corresponding vector elements, where  $n_\varepsilon$ ,  $n_u$ , and  $n_s$ , are the dimensions of  $\varepsilon_t$ ,  $u_t$ , and  $s_t$ . We define matrices of the form:

$$S_{ss}^{(1)}(t) = \left\{ \frac{1}{2h} (\Phi_i(\hat{s}_t + h\hat{s}_{s,j}, \bar{\varepsilon}_{t+1}; \theta) - \Phi_i(\hat{s}_t - h\hat{s}_{s,j}, \bar{\varepsilon}_{t+1}; \theta)) \right\}, \text{ for } i, j = 1, \dots, n_s,$$

$$S_{s\varepsilon}^{(1)}(t) = \left\{ \frac{1}{2h} (\Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1} + hs_{\varepsilon,j}; \theta) - \Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1} - hs_{\varepsilon,j}; \theta)) \right\}, \text{ for } i = 1, \dots, n_s \text{ and } j = 1, \dots, n_\varepsilon,$$

$$S_{ys}^{(1)}(t) = \left\{ \frac{1}{2h} (\Psi_i(\bar{s}_t + h\bar{s}_{s,j}; \theta) - \Psi_i(\bar{s}_t - h\bar{s}_{s,j}; \theta)) \right\}, \text{ for } i = 1, \dots, n_y \text{ and } j = 1, \dots, n_s,$$

$$S_{ss}^{(2)}(t) = \left\{ \frac{\sqrt{h^2 - 1}}{2h} (\Phi_i(\hat{s}_t + h\hat{s}_{s,j}, \bar{\varepsilon}_{t+1}; \theta) + \Phi_i(\hat{s}_t - h\hat{s}_{s,j}, \bar{\varepsilon}_{t+1}; \theta) - 2\Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1}; \theta)) \right\},$$

for  $i, j = 1, \dots, n_s$ ,

$$S_{s\varepsilon}^{(2)}(t) = \left\{ \frac{\sqrt{h^2 - 1}}{2h} (\Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1} + hs_{\varepsilon,j}; \theta) + \Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1} - hs_{\varepsilon,j}; \theta) - 2\Phi_i(\hat{s}_t, \bar{\varepsilon}_{t+1}; \theta)) \right\},$$

for  $i = 1, \dots, n_s$  and  $j = 1, \dots, n_\varepsilon$ ,

$$S_{ys}^{(2)}(t) = \left\{ \frac{\sqrt{h^2 - 1}}{2h} (\Psi_i(\bar{s}_t + h\bar{s}_{s,j}; \theta) + \Psi_i(\bar{s}_t - h\bar{s}_{s,j}; \theta) - 2\Psi_i(\bar{s}_t; \theta)) \right\},$$

for  $i = 1, \dots, n_y$  and  $j = 1, \dots, n_s$ ,

where  $n_y$  is the dimension of  $y_t$  and the superscripts (1) and (2) represent the first- and second-order effects of the general nonlinear functions, respectively. According to Nøgaard et al. (2000), the step size  $h^2 \geq 1$  is selected based on the distributions to which the interpolation applies. In the present setting, it is optimal to set  $h^2$  equal to the kurtosis of the distributions.<sup>26</sup>

The second-order approximation of equation (21) by the multivariate Stirling interpolation gives

$$\begin{aligned} \bar{s}_{t+1} &= \frac{h^2 - n_s - n_\varepsilon}{h^2} \Phi(\hat{s}_t, \bar{\varepsilon}_{t+1}; \theta) + \frac{1}{2h^2} \sum_{p=1}^{n_s} (\Phi(\hat{s}_t + h\hat{s}_{s,p}, \bar{\varepsilon}_{t+1}; \theta) + \Phi(\hat{s}_t - h\hat{s}_{s,p}, \bar{\varepsilon}_{t+1}; \theta)) \\ &\quad + \frac{1}{2h^2} \sum_{p=1}^{n_\varepsilon} (\Phi(\hat{s}_t, \bar{\varepsilon}_{t+1} + hs_{\varepsilon,p}; \theta) + \Phi(\hat{s}_t, \bar{\varepsilon}_{t+1} - hs_{\varepsilon,p}; \theta)). \end{aligned} \quad (29)$$

A lower triangular Cholesky factor  $\bar{S}_s(t+1)$  of the ex-ante covariance matrix of  $\bar{s}_{t+1}$  is obtained by

$$\bar{S}_s(t+1) = \Upsilon \left( \begin{bmatrix} S_{ss}^{(1)}(t) & S_{s\varepsilon}^{(1)}(t) & S_{ss}^{(2)}(t) & S_{s\varepsilon}^{(2)}(t) \end{bmatrix} \right), \quad (30)$$

where  $\Upsilon$  denotes the Householder transformation. Then, the approximation of (23) is expressed as

$$\begin{aligned} \bar{P}_{ss}(t+1) &= \bar{S}_s(t+1)\bar{S}_s(t+1)' \\ &= S_{ss}^{(1)}(t)S_{ss}^{(1)}(t)' + S_{ss}^{(2)}(t)S_{ss}^{(2)}(t)' + S_{s\varepsilon}^{(1)}(t)S_{s\varepsilon}^{(1)}(t)' + S_{s\varepsilon}^{(2)}(t)S_{s\varepsilon}^{(2)}(t)'. \end{aligned}$$

The ex-ante estimator of the measurement vector is given by

$$\bar{y}_{t+1} = \frac{h^2 - n_s}{h^2} \Psi(\bar{s}_{t+1}; \theta) + \frac{1}{2h^2} \sum_{p=1}^{n_s} (\Psi(\bar{s}_{t+1} + h\bar{s}_{s,p}; \theta) + \Psi(\bar{s}_{t+1} - h\bar{s}_{s,p}; \theta)) \quad (31)$$

which corresponds to the second-order approximation of equation (22). The covariance matrix for this estimator is calculated from

$$\bar{S}_y(t+1) = \Upsilon \left( \begin{bmatrix} S_{ys}^{(1)}(t+1) & S_u(t+1) & S_{ys}^{(2)}(t+1) \end{bmatrix} \right), \quad (32)$$

and thus, the approximation of (24) is expressed as

$$\begin{aligned} \bar{P}_{yy}(t+1) &= \bar{S}_y(t+1)\bar{S}_y(t+1)' \\ &= S_{ys}^{(1)}(t+1)S_{ys}^{(1)}(t+1)' + S_u(t+1)S_u(t+1)' + S_{ys}^{(2)}(t+1)S_{ys}^{(2)}(t+1)'. \end{aligned}$$

---

<sup>26</sup>For the normal distribution, the optimal value for  $h^2$  is 3.

As  $P_{sy}(t+1) = \bar{S}_s(t+1)S_{ys}^{(1)}(t+1)'$ , we can rewrite the expression (26) for the Kalman gain as

$$K_{t+1} = \bar{S}_s(t+1)S_{ys}^{(1)}(t+1)' [\bar{S}_y(t+1)\bar{S}_y(t+1)']^{-1}. \quad (33)$$

From (28), straightforward algebra yields

$$\hat{P}_{ss}(t+1) = \hat{S}_s(t+1)\hat{S}_s(t+1)',$$

where

$$\hat{S}_s(t+1) = \Upsilon \left( \begin{bmatrix} \bar{S}_s(t+1) - K_{t+1}S_{ys}^{(1)}(t+1) & K_{t+1}S_u(t+1) & K_{t+1}S_{ys}^{(2)}(t+1) \end{bmatrix} \right). \quad (34)$$

Therefore, the algorithm for the CDKF is summarized as follows.

**Initialization:** Set  $\hat{s}_0$  and  $\hat{S}_0$  based on the unconditional first and second moments of  $s_t$  implied by the nonlinear solution (19).

**Prediction step:** Using (29), (30), (31), and (32), calculate  $\bar{s}_{t+1}$ ,  $\bar{S}_s(t+1)$ ,  $\bar{y}_{t+1}$ , and  $\bar{S}_y(t+1)$ , respectively.

**Updating step:** Using (33), (25), and (34), calculate  $K_{t+1}$ ,  $\hat{s}_{t+1}$ , and  $\hat{S}_s(t+1)$ , respectively.

**Likelihood approximation:** Assuming that the conditional density  $y_{t+1}|Y_{1:t}$  is approximated by a normal distribution, a quasi-log-likelihood function based on the CDKF is given by

$$\begin{aligned} \ln \hat{p}(Y_{1:T}|\theta) &= -\frac{n_y T}{2} \ln(2\pi) - \frac{1}{2} \sum_{t=1}^T \ln(|\bar{P}_{yy}(t;\theta)|) \\ &\quad - \frac{1}{2} \sum_{t=1}^T (y_t - \bar{y}_t(\theta))' \bar{P}_{yy}(t;\theta)^{-1} (y_t - \bar{y}_t(\theta)). \end{aligned}$$

### 4.3 Markov chain Monte Carlo (MCMC) algorithm

In the Bayesian approach, a posterior distribution  $p(\theta|Y_{1:T})$ , which characterizes the joint distribution of parameters and data, is obtained by combining the likelihood function  $p(Y_{1:T}|\theta)$  and a prior distribution  $p(\theta)$ , based on Bayes' theorem

$$p(\theta|Y_{1:T}) = \frac{p(Y_{1:T}|\theta)p(\theta)}{p(Y_{1:T})},$$

where  $p(Y_{1:T}) = \int p(Y_{1:T}|\theta)p(\theta) d\theta$ , and inferences are made based on the posterior distribution. Because it is generally impossible to calculate the posterior distribution analytically, we rely on sampling techniques to generate draws from the posterior distribution. In particular, MCMC algorithms are widely used. MCMC algorithms produce a sequence of serially correlated parameter draws  $\theta^{(i)}$ ,  $i = 1, \dots, N$  with the property that the random variables  $\theta^{(i)}$  converge in distribution to the target posterior distribution as  $N \rightarrow \infty$ .

In the class of MCMC algorithms, the random-walk Metropolis–Hastings (RWMH) algorithm is the most extensively used to estimate DSGE models, irrespective of whether the models are linear or nonlinear. In what follows, we present a procedure of the RWMH algorithm.

**Initialization:**

1. Use a numerical optimization routine to find the mode  $\theta^*$  of the posterior density  $p(\theta|Y_{1:T}) \propto p(Y_{1:T}|\theta)p(\theta)$ , and compute the inverse Hessian  $\Sigma^*$  at the mode, i.e.,

$$\Sigma^* = - \left[ \frac{\partial^2 p(Y_{1:T}|\theta)p(\theta)}{\partial \theta \partial \theta'} \Big|_{\theta=\theta^*} \right]^{-1}.$$

2. Initialize  $\theta^{(0)} = \theta^*$  and  $i = 1$ .

**Sampling:** For  $i = 1$  to  $N$ :

1. Draw a candidate parameter vector  $\vartheta$  from a proposal distribution  $N(\theta^{(i-1)}, c^2 \Sigma^*)$ , where  $c$  is a scale parameter that controls the variances in the proposal distribution.
2. The candidate draw of parameters is accepted ( $\theta^{(i)} = \vartheta$ ) with probability  $\min(r, 1)$  and rejected ( $\theta^{(i)} = \theta^{(i-1)}$ ) otherwise, where

$$r = \frac{p(Y_{1:T}|\vartheta)p(\vartheta)}{p(Y_{1:T}|\theta^{(i-1)})p(\theta^{(i-1)})}.$$

To generate draws efficiently, it is recommended that the scale parameter  $c$ , which controls the variances in the proposal distribution, should be adjusted so that the average acceptance rate of candidate draws is around 25 percent.<sup>27</sup>

It is known that the distribution of the draws  $\{\theta^{(i)}\}_{i=1}^N$  converges to the true posterior distribution as  $N \rightarrow \infty$ . From a practical perspective, however, the initial draws do not necessarily reflect the posterior distribution and a large number of draws are required before the resulting distribution converges. For this reason, it is common to discard a substantial part of the initial draws, say, the first  $N_0$  draws. Thus, the remaining posterior draws  $\{\theta^{(N_0+1)}, \dots, \theta^{(N)}\}$  are regarded to constitute the posterior distribution. Based on these draws, the mean estimates of parameters are given by

$$\hat{\mathbb{E}}[\theta|Y_{1:T}] = \frac{1}{N-N_0} \sum_{i=N_0+1}^N \theta^{(i)}.$$

Moreover, we can calculate the transformation of parameters  $\theta^{(i)}$ , denoted by  $g(\theta^{(i)})$ ; e.g., as higher-order moments, impulse response functions, and variance decompositions, and evaluate

$$\hat{\mathbb{E}}[g(\theta)|Y_{1:T}] = \frac{1}{N-N_0} \sum_{i=N_0+1}^N g(\theta^{(i)})$$

and the distribution of  $g(\theta)|Y_{1:T}$ .

The overall fit of an estimated DSGE model, denoted by  $\mathcal{M}$ , is evaluated by the marginal data density:

$$p(Y_{1:T}|\mathcal{M}) = \int p(Y_{1:T}|\theta, \mathcal{M}) p(\theta|\mathcal{M}) d\theta.$$

---

<sup>27</sup>Roberts et al. (1997) show that the asymptotically optimal acceptance rate in the RWMH algorithm is 0.234 under general conditions.

Based on the posterior parameter draws  $\{\theta^{(N_0+1)}, \dots, \theta^{(N)}\}$ , we can approximate the marginal data density by

$$\hat{p}(Y_{1:T}|\mathcal{M}) = \left[ \frac{1}{N - N_0} \sum_{i=N_0+1}^N \frac{f(\theta^{(i)})}{p(Y_{1:T}|\theta, \mathcal{M}) p(\theta^{(i)}|\mathcal{M})} \right]^{-1},$$

where  $f(\theta)$  is a function with the property that  $\int f(\theta)d\theta = 1$ . [Geweke \(1999\)](#) proposes an efficient choice of  $f(\theta)$ ; that is, a truncated normal approximation of the posterior distribution for  $\theta$  generated from a posterior sampler. The approximation based on his choice of  $f(\theta)$  is called the modified harmonic mean estimator.

#### 4.4 Sequential Monte Carlo (SMC) algorithm

In general, the likelihood function for a DSGE model is a very complicated function of its structural parameters, and hence, the resulting posterior distribution can exhibit an irregular shape, such as discontinuity and multimodality. In such a case, the widely-used RWMH algorithm can get stuck near a local mode and fail to find the entire posterior distribution. To deal with this issue, [Herbst and Schorfheide \(2014, 2015\)](#) propose the use of the SMC algorithm to approximate the posterior distribution.

The idea of the SMC algorithm is as follows. Let  $\phi_n, n = 0, 1, \dots, N_\phi$  be a sequence that increases from zero to one, for instance,  $\phi_n = n/N_\phi$ . We define a sequence of tempered posteriors as

$$\pi_n(\theta) = \frac{[p(Y_{1:T}|\theta)]^{\phi_n} p(\theta)}{\int [p(Y_{1:T}|\theta)]^{\phi_n} p(\theta) d\theta}, \quad n = 0, \dots, N_\phi.$$

The tempered posterior is initially, i.e., when  $n = 0$ , just a prior density  $p(\theta)$ . Then, the information on data is gradually added through the tempered likelihood term  $[p(Y_{1:T}|\theta)]^{\phi_n}$ . The tempered posterior eventually approaches to the posterior distribution as  $\phi_n \rightarrow 1$ . Thus, the SMC algorithm can overcome the aforementioned issue by constructing an approximation to the posterior gradually through tempering the likelihood function. To approximate the tempered posteriors and the resulting posterior distribution, particles representing  $\theta$  are propagated as in the particle filter described in Section 4.1.

Another advantage of the SMC algorithm over the RWMH algorithm is that we do not need to compute the mode of the posterior distribution and the Hessian at the mode. In the RWMH algorithm, sampling starts at the posterior mode, which we need to find using a numerical optimization routine, with the variance of the proposal distribution being set proportional to the inverse Hessian at the mode. In many applications, however, it is often tiresome to find the posterior mode, and the numerical approximation of the Hessian may be inaccurate. By contrast, the SMC algorithm starts with a prior distribution, from which we can easily sample.

A generic SMC algorithm is presented below following the exposition in [Herbst and Schorfheide \(2015\)](#).

**Initialization ( $\phi_0 = 0$ ):**

1. Draw the initial particles  $\{\theta_0^{(i)}\}_{i=1}^N$  from the prior distribution  $\theta_0^{(i)} \sim p(\theta)$ , where  $N$  denotes the number of particles.
2. Initialize particle weights  $W_0^{(i)} = 1$ , for  $i = 1, \dots, N$ .

**Recursion:** For  $n = 1, \dots, N_\phi$ :

1. For  $i = 1, \dots, N$ , Calculate the incremental weights

$$\tilde{w}_n^{(i)} = [p(Y_{1:T} | \theta_{n-1}^{(i)})]^{\phi_n - \phi_{n-1}}$$

and the normalized weights

$$\tilde{W}_n^{(i)} = \frac{\tilde{w}_n^{(i)} W_{n-1}^{(i)}}{\frac{1}{N} \sum_{i=1}^N \tilde{w}_n^{(i)} W_{n-1}^{(i)}}$$

2. If resampling is needed, draw particles  $\{\hat{\theta}_n^{(i)}\}_{i=1}^N$  from a multinomial distribution characterized by support points and weights  $\{\theta_{n-1}^{(i)}, \tilde{W}_n^{(i)}\}$  and set  $W_n^{(i)} = 1$ , for  $i = 1, \dots, N$ . If resampling is not needed, let  $\hat{\theta}_n^{(i)} = \theta_{n-1}^{(i)}$  and  $W_n^{(i)} = \tilde{W}_n^{(i)}$  for  $i = 1, \dots, N$ .
3. Propagate the particles by running  $N_{MH}$  steps of the RWMH algorithm with a proposal distribution  $N(\hat{\theta}_n^{(i)}, \hat{c}_n^2 \Sigma(\hat{\theta}_n^{(i)}))$ , where  $\Sigma(\hat{\theta}_n^{(i)})$  is the covariance matrix of the particles  $\{\hat{\theta}_n^{(i)}\}_{i=1}^N$  and  $\hat{c}_n$  controls the variances in the proposal distribution.  $\hat{c}_n$  is adjusted to ensure that the acceptance rate in the MH step is around 25 percent, according to

$$\hat{c}_n = \hat{c}_{n-1} \left[ 0.95 + 0.10 \frac{e^{16(\hat{a}_{n-1} - 0.25)}}{1 + e^{16(\hat{a}_{n-1} - 0.25)}} \right],$$

where  $\hat{a}_{n-1}$  is the average acceptance rate over the particles in stage  $n - 1$ . The resulting propagated particles are denoted by  $\{\theta_n^{(i)}\}_{i=1}^N$ .

Regarding the choice of a tempering schedule, [Herbst and Schorfheide \(2014\)](#) propose to specify

$$\phi_n = \left( \frac{n}{N_\phi} \right)^\lambda$$

and set  $\lambda = 2$ . With this choice, the information of the likelihood function is fed to the tempered posterior slowly in the beginning of the algorithm so that the possibility of the degeneracy problem is reduced, and then the pace of feeding the likelihood information increases after the tempered posterior distribution becomes relatively stable.

As in the particle filter presented in Section 4.1, the resampling in step 2 is necessary to avoid the degeneracy of the distribution of particle weights, and resampling is done when the ESS of the particle weights  $ESS_n = N / (1/N \sum_{i=1}^N (\tilde{W}_n^{(i)})^2)$  falls below a threshold; typically,  $N/2$ .

It is known that under some regularity conditions the distribution of the particles  $\{\theta_{N_\phi}^{(i)}\}_{i=1}^N$  obtained in the final stage ( $n = N_\phi$ ,  $\phi_{N_\phi} = 1$ ) converges to the true posterior distribution as  $N \rightarrow \infty$ . Thus, based on the particles in the final stage, the mean estimates of parameters are given by

$$\hat{\mathbb{E}}[\theta | Y_{1:T}] = \frac{1}{N} \sum_{i=1}^N \theta_{N_\phi}^{(i)}.$$

Moreover, we can evaluate the transformation of parameters  $\theta_{N_\phi}^{(i)}$ , denoted by  $g(\theta_{N_\phi}^{(i)})$ , as

$$\hat{\mathbb{E}}[g(\theta) | Y_{1:T}] = \frac{1}{N} \sum_{i=1}^N g(\theta_{N_\phi}^{(i)})$$

and the distribution of  $g(\theta) | Y_{1:T}$ .

The SMC algorithm delivers an approximation of the marginal data density as a byproduct. It can be verified that

$$\begin{aligned} \frac{1}{N} \sum_{i=1}^N \tilde{w}_n^{(i)} W_{n-1}^{(i)} &\xrightarrow{a.s.} \int [p(Y_{1:T}|\theta)]^{\phi_n - \phi_{n-1}} \frac{[p(Y_{1:T}|\theta)]^{\phi_{n-1}} p(\theta)}{\int [p(Y_{1:T}|\theta)]^{\phi_{n-1}} p(\theta) d\theta} d\theta \\ &= \frac{\int [p(Y_{1:T}|\theta)]^{\phi_n} p(\theta) d\theta}{\int [p(Y_{1:T}|\theta)]^{\phi_{n-1}} p(\theta) d\theta}. \end{aligned}$$

Therefore, the approximation of the marginal data density is given by

$$\hat{p}(Y_{1:T}|M) = \prod_{n=1}^{N_\phi} \left( \frac{1}{N} \sum_{i=1}^N \tilde{w}_n^{(i)} W_{n-1}^{(i)} \right).$$

## 5 Applications of Nonlinear Estimation Methods to DSGE Models with the ZLB

### 5.1 Necessity for nonlinear estimation

As the ZLB on nominal interest rates has become a primary concern for policy makers, much work has been devoted to understanding how the economy works and how policy should be conducted in the presence of this bound from a theoretical perspective.<sup>28</sup> However, empirical studies that estimate DSGE models including the ZLB remain limited because of the computational burden in the treatment of nonlinearity arising from the bound. As reviewed, to evaluate the likelihood function in a nonlinear setting, we need to rely on nonlinear solution and estimation methods, both of which require iterative procedures, and their computational expense grows rapidly with an increase in the dimensionality of problems. For this reason, most practitioners continue to estimate linearized DSGE models without explicitly considering the ZLB. However, as shown by Fernández-Villaverde and Rubio-Ramírez (2005) and Fernández-Villaverde et al. (2006), the level of likelihood and parameter estimates based on a linearized model can be significantly different from those based on its nonlinear counterpart. The following four studies demonstrate the necessity for the use of a nonlinear estimation procedure in estimating DSGE models in the presence of the ZLB.

Hirose and Inoue (2016) examine how and to what extent the estimates of structural parameters can be biased in an estimated DSGE model where the existence of the ZLB is omitted in the estimation process. They construct artificial time series of output, inflation, and the nominal interest rate simulated from a standard New Keynesian DSGE model that incorporates an occasionally binding constraint on the nominal interest rate. More specifically, they consider a quasi-linear model in which the ZLB is imposed but all the equilibrium conditions are linearized, to focus on parameter biases only resulting from omitting the ZLB and exclude possible biases from ignoring other nonlinearities in the original equilibrium conditions.<sup>29</sup> The parameters calibrated in this data

---

<sup>28</sup>See, for instance, Eggertsson and Woodford (2003), Jung et al. (2005), Adam and Billi (2006, 2007), Christiano et al. (2011), Aruoba et al. (2017), Bodenstein et al. (2013), Boneva et al. (2016), Erceg and Lindé (2014), Fernández-Villaverde et al. (2015), Gavin et al. (2015), Nakata (2016, 2017), and Ngo (2014).

<sup>29</sup>The model is solved using the algorithm employed by Erceg and Lindé (2014) and Bodenstein et al. (2013). In the algorithm, if the model-implied nominal interest rate falls below zero, a sequence of contractionary monetary policy shocks is added in both the current and anticipated periods so that the contemporaneous and expected interest rates at the lower bound are zero.

generating process (DGP) are regarded as true values. Then, using the simulated data, a Monte Carlo experiment is conducted, where the model is estimated without imposing the ZLB. They assess the parameter biases from neglecting the ZLB by comparing the estimated parameters with their true values. According to the results in their baseline experiment, no significant biases are detected in the estimated parameters, whereas the parameter estimates associated with the monetary policy rule are slightly biased. However, with the increased probability of hitting the ZLB or the longer duration of ZLB spells, the parameter biases become large and substantially affect the dynamic properties of the model. Moreover, they demonstrate that the estimation of the model omitting the ZLB leads to biased estimates of the structural shocks, even though the estimated parameters are virtually unbiased. These findings caution researchers against the common practice of estimating linearized DSGE models without considering the ZLB.

[Hirose and Sunakawa \(2015\)](#) extend Hirose and Inoue's [\(2016\)](#) analysis by employing a fully nonlinear DSGE model including the ZLB as a DGP and examine whether parameter estimates can be biased in the linearized version of the model. While [Hirose and Inoue \(2016\)](#) point to parameter biases only resulting from omitting the ZLB, [Hirose and Sunakawa \(2015\)](#) are able to investigate biases caused by missing nonlinearities associated with both the ZLB and the other equilibrium conditions. Their main results are summarized as follows. In the baseline experiment, while the estimates of structural parameters related to preferences and nominal rigidities are not biased, significant biases are detected in the estimates of monetary policy parameters and the steady-state inflation and real interest rates. They show that these biases arise for the most part from missing the ZLB rather than linearizing the equilibrium conditions. They also demonstrate that the biases in parameter estimates increase if the true parameter values in the DGP are altered so that the frequency of binding at the ZLB and the average duration of ZLB spells increase, which is in line with the result of [Hirose and Inoue \(2016\)](#). Given that parameters are fixed at the true values, the variance–covariance matrix and impulse response functions of observed variables implied by the linearized model are substantially different from those implied by its nonlinear counterpart. However, they find that the linearized model with the biased estimates of parameters can successfully replicate the unconditional variances and covariances of observables implied by the true nonlinear model, while some of the estimated impulse responses are noticeably different from the true ones.

[Richter and Throckmorton \(2016a\)](#) compare the estimates of three versions of a small-scale New Keynesian DSGE model: (1) a fully nonlinear model with the ZLB; (2) a constrained linear model where the ZLB is imposed in the filter but not in the solution; and (3) a linear model without the ZLB. Each model is estimated with US data over the period from 1986Q1 to 2015Q4 using a particle filter and the RWMH algorithm. They find that the posterior distributions of the parameters are not much different across the three models. The marginal data densities are also very similar between the nonlinear and constrained linear models, whereas the linear model without the ZLB delivers a smaller likelihood than the other two models. However, they demonstrate that the nonlinear model fits better to the data in periods when the ZLB is or is likely to be binding, and that it can replicate higher output volatility and negative skewness in output and inflation as observed in the data during the ZLB period. By contrast, the linear models with and without the ZLB give rise to large contractionary monetary policy shocks to explain the data. They also consider the prediction from a quasi-linear model where the ZLB is imposed but all the equilibrium conditions are linearized. They show that the quasi-linear model does a better job than the linear model, but that it still fails to generate the high volatilities of output and inflation during the ZLB period.

[Atkinson et al. \(2018\)](#) conduct a similar analysis to [Hirose and Sunakawa \(2015\)](#), but focus on comparing the accuracy of linear and nonlinear estimation methods for a DSGE model with the

ZLB. They solve a fully nonlinear small-scale New Keynesian model that incorporates the ZLB using a policy function iteration algorithm and simulate artificial time series of output, inflation, and the nominal interest rate from the nonlinear model. Using the simulated data, they estimate a linearized version of the model with the Kalman filter and the original nonlinear model with an unscented Kalman filter and a particle filter. They also investigate how the variances of measurement errors in observation equations and the probability that the ZLB binds in the data affects the accuracy of each estimation method. Their results indicate that the nonlinear model with the particle filter is more accurate and attains a higher marginal data density than the linear model with the Kalman filter and the nonlinear model with the unscented Kalman filter, as long as the variances of the measurement errors are not too large (less than 5 percent of the variances in the data) and the probability of being at the ZLB is sufficiently high (more than 15 percent of quarters in the sample).

## 5.2 Two-step approach

Although researchers recognize the importance of taking account of nonlinearity in the estimation of DSGE models with the ZLB, it is very time consuming to actually conduct the estimation of model parameters in a fully nonlinear setting. This is because we need to run procedures for nonlinear solution methods and a particle filter to evaluate the likelihood function repeatedly for each draw of parameters. Indeed, most applications require repeating the likelihood evaluation more than 100,000 times. To avoid such a huge computational cost, several studies take a two-step approach to conduct empirical analysis using nonlinear DSGE models. In this approach, a linearized version of a model is first estimated with the Kalman filter. Next, given the estimated parameters, the model is solved in a fully nonlinear setting to simulate the model or to apply a nonlinear filter to estimate unobserved state variables in the model.

[Aruoba et al. \(2018\)](#) consider Markov switching between the targeted-inflation and deflation steady state in a small-scale New Keynesian DSGE model with the ZLB and estimate whether the US and Japan have been in either the targeted-inflation or deflation regime.<sup>30</sup> To this end, they first estimate a linearized version of the model omitting the ZLB using the data ranging from 1984Q1 to 2007Q4 for the US and from 1981Q1 to 1994Q4 for Japan, when each economy was not subject to the ZLB and consistent with the targeted-inflation regime. The parameters regarding the steady-state inflation and the transition probabilities between the two regimes are calibrated so that the unconditional probability of being in each regime and long-run inflation are reasonable enough. Then, given the estimated and calibrated parameters, they solve the fully nonlinear model incorporated with Markov switching and the ZLB and apply a particle filter for a full sample up to 2015Q2 to extract the estimates of latent states including the regimes which each economy either remained in or switched to. For both countries, they analyze six specifications that differ in terms of the monetary policy rules and the observed variables in the estimation. According to their results, five out of the six specifications suggest that the US economy remained in the targeted-inflation regime throughout the ZLB period, whereas four of the specifications indicate that Japan's economy since the late 1990s has spent most of the time in the deflation regime.

[Hirose and Sunakawa \(2017\)](#) estimate the natural interest rate in the US using a nonlinear New Keynesian DSGE model with the ZLB and examine how nonlinearities affect the estimates

---

<sup>30</sup>In the standard New Keynesian framework, a central bank follows an active monetary policy rule; that is, the nominal interest rate is adjusted more than one for one when inflation deviates from a given target, and the economy fluctuates around the steady state where actual inflation coincides with the targeted inflation. In addition to such a target-inflation steady state, [Benhabib et al. \(2001\)](#) argue that the combination of an active monetary policy rule and the ZLB on the nominal interest rate gives rise to another long-run equilibrium, called a deflation steady state, where the inflation rate is negative and the nominal interest rate is very close to zero.

of the natural rate and its driving forces.<sup>31</sup> To parameterize the model, they estimate a linearized version of the model using US data prior to the date when the nominal interest rate was bounded at zero (1983Q1–2007Q4). Next, given the estimated parameters, they solve the model in a fully nonlinear setting with the ZLB and apply a particle filter for a full sample (1983Q1–2016Q3) to extract the sequence of the natural interest rate. Their main results are as follows. Comparing the estimated natural interest rate based on the nonlinear model with that based on the linear counterpart, they find that the former is higher than the latter to a substantial degree, particularly in the periods when the nominal interest rate is close to or bounded at zero. This difference is ascribed to a contractionary effect arising from the ZLB, which is considered only in the nonlinear model. Although such a contractionary effect lowers expected output and inflation, actual output and inflation are pegged to the corresponding observables in the filtering process. Then, positive shocks to aggregate demand must be identified as being larger to satisfy the optimality conditions of households and firms. As a consequence, the estimated natural rate increases in the nonlinear setting. Although other nonlinearities, including price and wage dispersion, potentially affect the identification of shocks and the estimates of the natural rate, they demonstrate that these effects are relatively minor.

### 5.3 Fully-nonlinear estimation

While the two-step approach is practically useful for conducting empirical analysis using nonlinear DSGE models, it is unavoidably subject to the issues that are raised by the papers addressed in Section 5.1. To deal with these issues, several studies have challenged themselves to estimate the parameters of DSGE models in a fully nonlinear setting. An early example is [Fernández-Villaverde and Rubio-Ramírez \(2005\)](#), which estimates a neoclassical growth model in a nonlinear form.<sup>32</sup> In this subsection, we review [Gust et al. \(2017\)](#); [Plante et al. \(2018\)](#); [Iiboshi et al. \(2018\)](#), who successfully estimate nonlinear DSGE models incorporated with the ZLB using a particle filter.<sup>33</sup>

[Table 6](#) compares the estimation procedures used in these three studies. [Gust et al. \(2017\)](#) estimate a larger model with a far larger number of particles and replications for likelihood approximation and parameter sampling than the other two studies. While [Gust et al. \(2017\)](#) and [Plante et al. \(2018\)](#) employ the RWMH algorithm to generate posterior distributions of parameters, [Iiboshi et al. \(2018\)](#) use the SMC algorithm with relatively small numbers of parameter particles and tempering stages. [Iiboshi et al. \(2018\)](#) emphasize that the SMC algorithm delivers a more reliable posterior inference than the RWMH algorithm and that large measurement errors are no longer required in evaluating the likelihood function using a particle filter.

It is worth mentioning that all of these studies utilize parallelization for the solution of their nonlinear models and the evaluation of the particle filter. The extensive use of parallelization makes their fully-nonlinear estimation feasible.<sup>34</sup> [Gust et al. \(2017\)](#) use the High Performance Computing Cluster with 336 core processors maintained at the US Federal Reserve Board. They code in Fortran and use MPICH for parallel computing. [Plante et al. \(2018\)](#) perform computations

<sup>31</sup>In the paper, the natural interest rate is defined as the real interest rate that would prevail if prices and wages were flexible without any markup shocks.

<sup>32</sup>Another strand of the literature solves and estimates DSGE models approximated with higher-order perturbation methods. For instance, [Fernández-Villaverde and Rubio-Ramírez \(2007\)](#) estimate a real business cycle model with stochastic volatility, using a second-order perturbation method, and [Binsbergen et al. \(2012\)](#) estimate a model in which households have Epstein–Zin recursive preferences with a third-order perturbation.

<sup>33</sup>[Richter and Throckmorton \(2016b\)](#) also estimate nonlinear DSGE models similar to the one estimated in [Plante et al. \(2018\)](#), but with different assumptions for firms' price setting.

<sup>34</sup>[Fernández-Villaverde and Zarruk-Valencia \(2018\)](#) provide a practical guide to parallelization in economics.

Table 6: Comparison of estimation procedures

	Gust et al. (2017)	Plante et al. (2018)	Iiboshi et al. (2018)
Model size	Medium	Small	Small
Observables	5	3	3
Particles ( $M$ )	1,500,000	40,000	40,000
Sampling algorithm	RWMH	RWMH	SMC
Replications ( $N, N_\phi$ )	200,000	100,000	1,200 (particles) $\times 10$ (stages)

on the Hopper computing cluster at Auburn University. Their Fortran codes are parallelized using Open MPI. Iiboshi et al. (2018) run their codes using a workstation with 32 core processors.

Gust et al. (2017) estimate a fully nonlinear version of a medium-scale DSGE model incorporated with the ZLB for the US economy. The model is similar to those in Christiano et al. (2005) and Smets and Wouters (2007) and features habit persistence in consumption, investment adjustment cost, the variable capital utilization rate, and sticky wages and prices that are indexed to lagged inflation. In addition to shocks to total factor productivity, government spending, and monetary policy, two financial shocks are considered in the model. One is a shock to the marginal efficiency of investment as in Justiniano et al. (2011), who argue that this shock can be interpreted as a disturbance to the financial sector's ability to transform savings to investment. The other is a risk premium shock as in Smets and Wouters (2007). This shock affects the spread between the risk-free rate and the return on risky assets and alters consumption and investment through the Euler equation. Five quarterly time series are used for estimation: the growth rates of output, consumption, and investment, the inflation rate, and the nominal interest rate, ranging from 1983Q1 to 2014Q1. According to the estimated model, the two financial shocks were the primary sources of the economy being constrained at the ZLB during and after the Great Recession. Moreover, they demonstrate that about 30 percent of the sharp drop in output during the Great Recession was explained by the ZLB, and that in the absence of the ZLB, output would have recovered to its prerecession level about a year earlier. They also emphasize the importance of estimating the model in the fully nonlinear setting by comparing their baseline estimates with those obtained from the linearized version of the model. They show that the linearized version is likely to overestimate the role of exogenous shocks because of the lack of endogenous feedback mechanisms arising from the existence of the ZLB.

Plante et al. (2018) investigate the relationship between economic uncertainty from the ZLB and real economic activity in the US by estimating a vector autoregression (VAR) model and a nonlinear New Keynesian model with the ZLB. They first estimate a time-varying parameter VAR model with stochastic volatility and find that a strong negative correlation between output growth and its predicted volatility emerged in the late 2008 when the Fed became constrained by the ZLB. Then, they estimate a fully nonlinear small-scale New Keynesian model incorporated with the ZLB using quarterly data on the output growth rate, the inflation rate, and the nominal interest rate from 1986Q1 to 2014Q2. Given the posterior mean estimates of parameters, they apply a particle filter to extract the sequence of their endogenous uncertainty measure given by

$$\sigma_{\hat{y}^{gdp},t} = \sqrt{\mathbb{E}_t \left[ \left( \hat{y}_{t+1}^{gdp} - \mathbb{E}_t \hat{y}_{t+1}^{gdp} \right)^2 \right]},$$

where  $\hat{y}_{t+1}^{gdp}$  and  $\mathbb{E}_t \hat{y}_{t+1}^{gdp}$  denote the actual and expected real GDP growth rate implied by the model, respectively. The filtered sequence of the real GDP growth uncertainty indicates that uncertainty was almost constant when the ZLB constraint did not bind, but increased to a considerable degree as the nominal interest rate was close to and bounded at zero. Because the sharp drop in output coincides with the period immediately before and after the ZLB period, the uncertainty measure and output growth exhibit a strong negative correlation as implied by the VAR and various uncertainty measures. To understand the mechanism, they demonstrate that their uncertainty measure substantially increases in response to a shock to the discount factor when the ZLB binds, but changes little when the nominal interest rate is away from the ZLB.

[Iiboshi et al. \(2018\)](#) estimate a small-scale New Keynesian model with the ZLB in a fully nonlinear setting for the Japanese economy. The Bank of Japan has conducted a virtually zero interest rate policy for a very extended period (since 1999 with the exception of August 2000–March 2001 and July 2006–December 2008), and hence, estimating a linearized version of the model omitting the ZLB would lead to much larger biases in parameter estimates than in the case for the US economy. They use quarterly data on the output growth rate, the inflation rate, and the nominal interest rate from 1983Q2 to 2016Q2.<sup>35</sup> In estimating the model, they consider two specifications for a monetary policy inertia regarding which previous interest rate the central bank refers to in its monetary policy rule: (1) the actual nominal interest rate, which takes a value greater than or equal to zero, or (2) the notional interest rate suggested by a Taylor-type monetary policy rule, which can be lower than zero. Bayesian model comparison supports the latter specification, reflecting the Bank of Japan’s commitment to keep the nominal interest nearly at zero, even when economic conditions improve temporarily. Given the estimated parameters, they apply a particle filter to estimate the sequence of the natural interest rate and show that it was frequently negative after the mid-1990s. Surprisingly, the estimated natural interest rate is very similar to that based on the model without the ZLB. This is because the effect of biased parameter estimates from neglecting the ZLB on the estimate of the natural interest rate is canceled out by the effect of biased estimates of the sequences of shocks.

## 6 Concluding Remarks

In this review, we presented how nonlinear DSGE models can be solved and estimated using numerical approximation techniques, particularly when the model economy is subject to the occasionally binding ZLB constraint.

Empirical studies that estimate nonlinear DSGE models remain limited because of the huge computational cost of solving these models and evaluating the likelihood function. While such computational costs can be partly ameliorated using efficient algorithms and parallelization, as discussed in this review, more investment in both hardware and software may be needed. In this regard, some research groups have started using a high performance computing cluster or supercomputers. Faster programming languages are also available.<sup>36</sup> We trust that future methodological progress will enable many more researchers to solve and estimate nonlinear DSGE models.

---

<sup>35</sup>In a robustness analysis, they replace the data on output growth with that for the output gap constructed by the Bank of Japan.

<sup>36</sup>[Aruoba and Fernández-Villaverde \(2015\)](#) compare the execution time of the codes that solve a stochastic neoclassical growth using C++, Fortran, Java, Julia, Python, MATLAB, Mathematica, and R.

## References

- Abreu, D., D. Pearce, and E. Stacchetti (1990) “Toward a Theory of Discounted Repeated Games with Imperfect Monitoring”, *Econometrica*, Vol. 58, No. 5, pp. 1041–1063.
- Adam, K. and R. Billi (2006) “Optimal Monetary Policy under Commitment with a Zero Bound on Nominal Interest Rates”, *Journal of Money, Credit and Banking*, Vol. 38, No. 7, pp. 1877–1905.
- (2007) “Discretionary Monetary Policy and the Zero Lower Bound on Nominal Interest Rates”, *Journal of Monetary Economics*, Vol. 54, No. 3, pp. 728–752.
- Andreasen, M. M. (2013) “Non-linear DSGE Models and the Central Difference Kalman Filter”, *Journal of Applied Econometrics*, Vol. 28, No. 6, pp. 929–955.
- Andreasen, M. M., J. Fernández-Villaverde, and J. F. Rubio-Ramírez (2018) “The Pruned State-Space System for Non-Linear DSGE Models: Theory and Empirical Applications”, *Review of Economic Studies*, Vol. 85, No. 1, pp. 1–49.
- Aruoba, S. B., L. Bocola, and F. Schorfheide (2017) “Assessing DSGE Model Nonlinearities”, *Journal of Economic Dynamics and Control*, Vol. 83, No. C, pp. 34–54.
- Aruoba, S. B., P. Cuba-Borda, and F. Schorfheide (2018) “Macroeconomic Dynamics Near the ZLB: A Tale of Two Countries”, *Review of Economic Studies*, Vol. 85, No. 1, pp. 87–118.
- Aruoba, S. B. and J. Fernández-Villaverde (2015) “A Comparison of Programming Languages in Macroeconomics”, *Journal of Economic Dynamics and Control*, Vol. 58, No. C, pp. 265–273.
- Atkinson, T., A. Richter, and N. A. Throckmorton (2018) “The Accuracy of Linear and Nonlinear Estimation in the Presence of the Zero Lower Bound”, Federal Reserve Bank of Dallas Working Paper 1804.
- Barillas, F. and J. Fernández-Villaverde (2007) “A Generalization of the Endogenous Grid Method”, *Journal of Economic Dynamics and Control*, Vol. 31, No. 8, pp. 2698–2712.
- Benhabib, J., S. Schmitt-Grohé, and M. Uribe (2001) “The Perils of Taylor Rules”, *Journal of Economic Theory*, Vol. 96, No. 1-2, pp. 40–69.
- van Binsbergen, J., J. Fernández-Villaverde, R. Koijen, and J. F. Rubio-Ramírez (2012) “The Term Structure of Interest Rates in a DSGE Model with Recursive Preferences”, *Journal of Monetary Economics*, Vol. 59, No. 7, pp. 634–648.
- Blanchard, O. J. and C. M. Kahn (1980) “The Solution of Linear Difference Models under Rational Expectations”, *Econometrica*, Vol. 48, No. 5, pp. 1305–1311.
- Bodenstein, M., L. Guerrieri, and C. J. Gust (2013) “Oil Shocks and the Zero Bound on Nominal Interest Rates”, *Journal of International Money and Finance*, Vol. 32, No. C, pp. 941–967.
- Boneva, L. M., R. A. Braun, and Y. Waki (2016) “Some Unpleasant Properties of Log-linearized Solutions When the Nominal Rate is Zero”, *Journal of Monetary Economics*, Vol. 84, pp. 216–232.
- Carroll, C. (2006) “The Method of Endogenous Gridpoints for Solving Dynamic Stochastic Optimization Problems”, *Economics Letters*, Vol. 91, No. 3, pp. 312–320.

- Chang, R. (1998) “Credible Monetary Policy in an Infinite Horizon Model: Recursive Approaches”, *Journal of Economic Theory*, Vol. 81, No. 2, pp. 431–461.
- Christiano, L., M. Eichenbaum, and S. Rebelo (2011) “When Is the Government Spending Multiplier Large?”, *Journal of Political Economy*, Vol. 119, No. 1, pp. 78–121.
- Christiano, L., M. S. Eichenbaum, and B. K. Johannsen (2018) “Does the New Keynesian Model Have a Uniqueness Problem?”, NBER Working Papers No. 24612.
- Christiano, L. and J. Fisher (2000) “Algorithms for Solving Dynamic Models with Occasionally Binding Constraints”, *Journal of Economic Dynamics and Control*, Vol. 24, No. 8, pp. 1179–1232.
- Christiano, L. J., M. Eichenbaum, and C. L. Evans (2005) “Nominal Rigidities and the Dynamic Effects of a Shock to Monetary Policy”, *Journal of Political Economy*, Vol. 113, No. 1, pp. 1–45.
- Cochrane, J. H. (2017) “The New-Keynesian Liquidity Trap”, *Journal of Monetary Economics*, Vol. 92, No. C, pp. 47–63.
- (2018) “Michelson-Morley, Fisher, and Occam: The Radical Implications of Stable Quiet Inflation at the Zero Bound”, *NBER Macroeconomics Annual*, Vol. 32, pp. 113–226.
- Coleman, W. J. (1990) “Solving the Stochastic Growth Model by Policy-Function Iteration”, *Journal of Business & Economic Statistics*, Vol. 8, No. 1, pp. 27–29.
- (1991) “Equilibrium in a Production Economy with an Income Tax”, *Econometrica*, Vol. 59, No. 4, pp. 1091–1104.
- Eggertsson, G. B. and M. Woodford (2003) “The Zero Bound on Interest Rates and Optimal Monetary Policy”, *Brookings Papers on Economic Activity*, Vol. 34, No. 1, pp. 139–235.
- Erceg, C. and J. Lindé (2014) “Is There a Fiscal Free Lunch in a Liquidity Trap?”, *Journal of the European Economic Association*, Vol. 12, No. 1, pp. 73–107.
- Fella, G. (2014) “A Generalized Endogenous Grid Method for Non-Smooth and Non-Concave Problems”, *Review of Economic Dynamics*, Vol. 17, No. 2, pp. 329–344.
- Feng, Z., J. Miao, A. Peralta-Alva, and M. S. Santos (2014) “Numerical Simulation of Nonoptimal Dynamic Equilibrium Models”, *International Economic Review*, Vol. 55, pp. 83–110.
- Fernández-Villaverde, J., G. Gordon, P. Guerrón-Quintana, and J. F. Rubio-Ramírez (2015) “Nonlinear Adventures at the Zero Lower Bound”, *Journal of Economic Dynamics and Control*, Vol. 57, No. C, pp. 182–204.
- Fernández-Villaverde, J. and J. F. Rubio-Ramírez (2005) “Estimating Dynamic Equilibrium Economies: Linear Versus Nonlinear Likelihood”, *Journal of Applied Econometrics*, Vol. 20, No. 7, pp. 891–910.
- (2007) “Estimating Macroeconomic Models: A Likelihood Approach”, *Review of Economic Studies*, Vol. 74, No. 4, pp. 1059–1087.
- Fernández-Villaverde, J., J. F. Rubio-Ramírez, and M. S. Santos (2006) “Convergence Properties of the Likelihood of Computed Dynamic Models”, *Econometrica*, Vol. 74, No. 1, pp. 93–119.

- Fernández-Villaverde, J., J. F. Rubio-Ramírez, and F. Schorfheide (2016) “Solution and Estimation Methods for DSGE Models”, in Taylor, J. B. and H. Uhlig, eds., *Handbook of Macroeconomics*, Vol. 2, Amsterdam: Elsevier, Chap. 9, pp. 527–724.
- Fernández-Villaverde, J. and D. Zarruk-Valencia (2018) “A Practical Guide to Parallelization in Economics”, NBER Working Paper No. 24561.
- Gavin, W. T., B. Keen, A. Richter, and N. Throckmorton (2015) “The Zero Lower Bound, the Dual Mandate, and Unconventional Dynamics”, *Journal of Economic Dynamics and Control*, Vol. 55, No. C, pp. 14–38.
- Geweke, J. (1999) “Using Simulation Methods for Bayesian Econometric Models: Inference, Development, and Communication”, *Econometric Reviews*, Vol. 18, No. 1, pp. 1–73.
- Gordon, N., D. Salmond, and A. Smith (1993) “Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation”, *IEE Proceedings-F*, Vol. 140, pp. 107–113.
- Greenwood, J. and G. W. Huffman (1995) “On the Existence of Nonoptimal Equilibria in Dynamic Stochastic Economies”, *Journal of Economic Theory*, Vol. 65, No. 2, pp. 611–623.
- Gust, C., E. Herbst, D. López-Salido, and M. E. Smith (2017) “The Empirical Implications of the Interest-Rate Lower Bound”, *American Economic Review*, Vol. 107, No. 7, pp. 1971–2006.
- den Haan, W. J. (2018) “Parameterized Expectations”, Unpublished lecture note.
- Hamilton, J. D. (1994) *Time Series Analysis*, Princeton: Princeton University Press.
- Heer, B. and A. Maussner (2009) *Dynamic General Equilibrium Modeling: Computational Methods and Applications*, New York: Springer, 2nd edition.
- Herbst, E. P. and F. Schorfheide (2014) “Sequential Monte Carlo Sampling for DSGE Models”, *Journal of Applied Econometrics*, Vol. 29, No. 7, pp. 1073–1098.
- (2015) *Bayesian Estimation of DSGE Models*, Princeton: Princeton University Press.
- Hills, T. S., T. Nakata, and T. Sunakawa (2018) “A Promised Value Approach to Optimal Monetary Policy”, Mimeo.
- Hirose, Y. and A. Inoue (2016) “The Zero Lower Bound and Parameter Bias in an Estimated DSGE Model”, *Journal of Applied Econometrics*, Vol. 31, No. 4, pp. 630–651.
- Hirose, Y. and T. Sunakawa (2015) “Parameter Bias in an Estimated DSGE Model: Does Nonlinearity Matter?”, CAMA Working Paper 46/2015.
- (2017) “The Natural Rate of Interest in a Nonlinear DSGE Model”, CAMA Working Paper 38/2017.
- Iiboshi, H., M. Shintani, and K. Ueda (2018) “Estimating a Nonlinear New Keynesian Model with a Zero Lower Bound for Japan”, Tokyo Center for Economic Research Working Paper E-120.
- Judd, K. L. (1992) “Projection Methods for Solving Aggregate Growth Models”, *Journal of Economic Theory*, Vol. 58, No. 2, pp. 410–452.
- (1998) *Numerical Methods in Economics*, Cambridge: MIT Press.

- Judd, K. L., L. Maliar, and S. Maliar (2011) “Numerically Stable and Accurate Stochastic Simulation Approaches for Solving Dynamic Economic Models”, *Quantitative Economics*, Vol. 2, No. 2, pp. 173–210.
- Judd, K. L., L. Maliar, S. Maliar, and I. Tsener (2017) “How to Solve Dynamic Stochastic Models Computing Expectations Just Once”, *Quantitative Economics*, Vol. 8, No. 3, pp. 851–893.
- Judd, K. L., L. Maliar, S. Maliar, and R. Valero (2014) “Smolyak Method for Solving Dynamic Economic Models: Lagrange Interpolation, Anisotropic Grid and Adaptive Domain”, *Journal of Economic Dynamics and Control*, Vol. 44, No. C, pp. 92–123.
- Judd, K. L., S. Yeltekin, and J. Conklin (2003) “Computing Supergame Equilibria”, *Econometrica*, Vol. 71, No. 4, pp. 1239–1254.
- Jung, T., Y. Teranishi, and T. Watanabe (2005) “Optimal Monetary Policy at the Zero-Interest-Rate Bound”, *Journal of Money, Credit and Banking*, Vol. 37, No. 5, pp. 813–835.
- Justiniano, A., G. Primiceri, and A. Tambalotti (2011) “Investment Shocks and the Relative Price of Investment”, *Review of Economic Dynamics*, Vol. 14, No. 1, pp. 101–121.
- Klein, P. (2000) “Using the Generalized Schur Form to Solve a Multivariate Linear Rational Expectations Model”, *Journal of Economic Dynamics and Control*, Vol. 24, No. 10, pp. 1405–1423.
- Krueger, D. and F. Kubler (2004) “Computing Equilibrium in OLG Models with Stochastic Production”, *Journal of Economic Dynamics and Control*, Vol. 28, No. 7, pp. 1411–1436.
- Lepetuyk, V., L. Maliar, and S. Maliar (2017) “Should Central Banks Worry about Nonlinearities of Their Large-Scale Macroeconomic Models?”, Bank of Canada staff paper 2017-21.
- Maliar, L. and S. Maliar (2015) “Merging Simulation and Projection Approaches to Solve High-Dimensional Problems with an Application to a New Keynesian Model”, *Quantitative Economics*, Vol. 6, No. 1, pp. 1–47.
- Marcat, A. (1988) “Solving Non-Linear Models by Parameterizing Expectations”, Carnegie Mellon University, Graduate School of Industrial Administration.
- McGrattan, E. (1996) “Solving the Stochastic Growth Model with a Finite Element Method”, *Journal of Economic Dynamics and Control*, Vol. 20, No. 1-3, pp. 19–42.
- Miranda, M. J. and P. L. Fackler (2002) *Applied Computational Economics and Finance*, Cambridge: MIT Press.
- Nakata, T. (2016) “Optimal Fiscal and Monetary Policy with Occasionally Binding Zero Bound Constraints”, *Journal of Economic Dynamics and Control*, Vol. 73, No. C, pp. 220–240.
- (2017) “Uncertainty at the Zero Lower Bound”, *American Economic Journal: Macroeconomics*, Vol. 9, No. 3, pp. 186–221.
- Ngo, V. P. (2014) “Optimal Discretionary Monetary Policy in a Micro-Founded Model with a Zero Lower Bound on Nominal Interest Rate”, *Journal of Economic Dynamics and Control*, Vol. 45, No. C, pp. 44–65.
- Nøgaard, M., N. K. Poulsen, and O. Ravn (2000) “Advances in Derivative-Free State Estimation for Nonlinear Systems”, *Automatica*, Vol. 36, No. 11, pp. 1627–1638.

- Orlik, A. and I. Presno (2013) “On Credible Monetary Policies with Model Uncertainty”, Mimeo.
- Plante, M., A. W. Richter, and N. A. Throckmorton (2018) “The Zero Lower Bound and Endogenous Uncertainty”, *Economic Journal*, Vol. 128, No. 611, pp. 1730–1757.
- Richter, A., N. Throckmorton, and T. Walker (2014) “Accuracy, Speed and Robustness of Policy Function Iteration”, *Computational Economics*, Vol. 44, No. 4, pp. 445–476.
- Richter, A. W. and N. A. Throckmorton (2016a) “Are Nonlinear Methods Necessary at the Zero Lower Bound?”, Federal Reserve Bank of Dallas Working Paper 1606.
- (2016b) “Is Rotemberg Pricing Justified by Macro Data?”, *Economics Letters*, Vol. 149, pp. 44–48.
- Roberts, G. O., A. Gelman, and W. R. Gilks (1997) “Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms”, *Annals of Applied Probability*, Vol. 7, No. 1, pp. 110–120.
- Rotemberg, J. (1982) “Sticky Prices in the United States”, *Journal of Political Economy*, Vol. 90, No. 6, pp. 1187–1211.
- Sargent, T. J. and J. Stachurski (2018) *Lectures in Quantitative Economics*, <https://lectures.quantecon.org/>.
- Schmitt-Grohe, 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*, Vol. 28, No. 4, pp. 755–775.
- Sims, C. A. (2002) “Solving Linear Rational Expectations Models”, *Computational Economics*, Vol. 20, No. 1-2, pp. 1–20.
- Smets, F. and R. Wouters (2007) “Shocks and Frictions in US Business Cycles: A Bayesian DSGE Approach”, *American Economic Review*, Vol. 97, No. 3, pp. 586–606.
- Smolyak, S. (1963) “Quadrature and Interpolation Formulas for Tensor Products of Certain Classes of Functions”, *Dokl. Akad. Nauk*, Vol. 148, No. 5, pp. 1042–1045.
- Sunakawa, T. (2018) “Applying Precomputation of Integrals to Nonlinear DSGE Models with Occasionally Binding Constraints”, Working Paper.
- Tauchen, G. (1986) “Finite State Markov-Chain Approximations to Univariate and Vector Autoregressions”, *Economics Letters*, Vol. 20, No. 2, pp. 177–181.

# A Appendix

## A.1 A New Keynesian model

The model economy consists of final- and intermediate-good producing firms, households, and monetary and fiscal authorities. Prices are sticky due to Rotemberg-type (1982) adjustment costs.

### A.1.1 Final-good firm

The final good firm uses intermediate inputs  $Y_t(j)$  for all  $j \in [0, 1]$  to produce aggregate output

$$Y_t = \left( \int_0^1 Y_t(j)^{1-\nu} dj \right)^{\frac{1}{1-\nu}},$$

where  $\nu$  is the inverse of demand elasticity. Profit maximization (or cost minimization) gives the demand curve:

$$Y_t(j) = \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu} Y_t, \quad (35)$$

$P_t(j)$  is the price for the intermediate input  $Y_t(j)$  for each  $j$  and  $P_t$  is the aggregate price level.

### A.1.2 Intermediate-good firms

Intermediate-good firms produce  $Y_t(j)$  in monopolistic competition and set prices facing quadratic price adjustment costs. The production function is given by

$$Y_t(j) = A_t N_t(j) \quad (36)$$

where  $A_t$  is the total factor productivity, which has a deterministic trend  $\bar{\gamma}$  and a shock to the trend  $z_t$  such as  $\ln \gamma_t \equiv \ln(A_t/A_{t-1}) = \ln \bar{\gamma} + \ln z_t$ .  $z_t$  follows an AR(1) process of  $\ln z_t = \rho_z \ln z_{t-1} + \epsilon_{z,t}$ , where  $\epsilon_{z,t} \sim N(0, \sigma_z^2)$ .  $N_t(j)$  is labor input. Firm  $j$  maximizes the present value of future profits

$$\mathbb{E}_t \left[ \sum_{s=0}^{\infty} \beta^s Q_{t+s|t} \left( \frac{P_{t+s}(j)}{P_{t+s}} Y_{t+s}(j) - W_{t+s} N_{t+s}(j) - AC_{t+s}(j) \right) \right]$$

subject to the production function (36) and demand curve (35).  $\beta^s Q_{t+s|t} = \beta^s (C_{t+s}/C_t)^{-\tau} (A_t/A_{t+s})^{1-\tau}$  is the stochastic discount factor from the household's Euler equation where  $\beta$  is a discount factor,  $\tau$  is a parameter for risk aversion, and  $C_t$  is aggregate consumption. The adjustment cost is of the form  $AC_t(j) = \frac{\phi}{2} \left( \frac{P_t(j)}{P_{t-1}(j)} - \pi \right)^2 Y_t(j)$  where  $\phi$  is a scale parameter and  $\pi$  is the steady-state gross inflation rate.  $W_t$  is the wage rate. The optimality condition is given by

$$\begin{aligned} & (1 - \nu^{-1}) \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu} \frac{Y_t}{P_t} + \nu^{-1} \frac{W_t}{A_t} \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu-1} \frac{Y_t}{P_t} \\ & - \frac{\partial AC_t(j)}{\partial P_t(j)} - \beta \mathbb{E}_t Q_{t+1|t} \frac{\partial AC_{t+1}(j)}{\partial P_t(j)} = 0. \end{aligned} \quad (37)$$

Note that

$$\begin{aligned}\frac{\partial AC_t(j)}{\partial P_t(j)} &= \phi \left( \frac{P_t(j)}{P_{t-1}(j)} - \pi \right) \frac{1}{P_{t-1}(j)} \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu} Y_t \\ &\quad - \nu^{-1} \frac{\phi}{2} \left( \frac{P_t(j)}{P_{t-1}(j)} - \pi \right)^2 \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu-1} \frac{Y_t}{P_t}, \\ \frac{\partial AC_{t+1}(j)}{\partial P_t(j)} &= -\phi \left( \frac{P_{t+1}(j)}{P_t(j)} - \pi \right) \frac{P_{t+1}(j)}{P_t(j)^2} \left( \frac{P_t(j)}{P_t} \right)^{-1/\nu} Y_t.\end{aligned}$$

### A.1.3 Household

The representative household maximizes lifetime utility

$$\mathbb{E}_t \left[ \sum_{s=0}^{\infty} \beta^s \left( \frac{(C_{t+s}/A_{t+s})^{1-\tau} - 1}{1-\tau} + \chi \ln \left( \frac{M_{t+s}}{P_{t+s}} \right) - \chi_H H_{t+s} \right) \right]$$

subject to the budget constraint

$$P_t C_t + B_t + M_t + T_t = P_t W_t H_t + R_{t-1} B_{t-1} + M_{t-1} + P_t D_t + P_t S C_t.$$

$\chi$  and  $\chi_H$  are the weights for money utility and labor disutility, respectively.  $M_t$  is money holding and  $B_t$  is bond holding at the end of period  $t$ .  $H_t$  is the aggregate labor supply.  $T_t$  is the lump sum tax.  $R_t$  is the gross nominal interest rate.  $D_t$  is the dividend from intermediate-good producing firms.  $S C_t$  is the net cash inflow from trading a full set of state-contingent securities. The optimality conditions are given by

$$1 = \beta \mathbb{E}_t \left[ \left( \frac{C_{t+1}/C_t}{A_{t+1}/A_t} \right)^{-\tau} \frac{A_t}{A_{t+1}} \frac{R_t}{\pi_{t+1}} \right], \quad (38)$$

$$\frac{W_t}{A_t} = \chi_H \left( \frac{C_t}{A_t} \right)^\tau, \quad (39)$$

where  $\pi_t = P_t/P_{t-1}$  is the gross inflation rate.

### A.1.4 Closing the model

The monetary policy rule is of the form

$$R_t = \left( r \pi^* \left( \frac{\pi}{\pi^*} \right)^{\psi_1} \left( \frac{Y_t}{Y_t^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{t-1}^{\rho_R} e^{\epsilon_{R,t}} \quad (40)$$

where  $r$  is the steady-state real interest rate,  $\pi^*$  is the target of inflation, and  $\epsilon_{R,t} \sim N(0, \sigma_R^2)$  is the monetary policy shock.  $Y_t^*$  is the natural level of output and given by the allocation in a flexible-price economy with setting  $\phi = 0$ .  $\psi_1$  and  $\psi_2$  are parameters for the degree of response to inflation or output for each.  $\rho_R$  is a parameter for the policy inertia. Government expenditure is a fraction of output,  $G_t = (1 - g_t^{-1}) Y_t$ .  $\ln g_t$  follows an AR(1) process

$$\ln g_t = (1 - \rho_g) \ln g + \rho_g \ln g_{t-1} + \epsilon_{g,t},$$

where  $\epsilon_{g,t} \sim N(0, \sigma_g^2)$  is the disturbance to government expenditure. We consider the symmetric equilibrium in which all the intermediate goods firms make identical decisions, and thus  $j$  can be omitted. The market-clearing conditions are given by

$$\begin{aligned} C_t + G_t + AC_t &= Y_t, \\ H_t &= N_t. \end{aligned} \tag{41}$$

We have the following first-order necessary conditions from equations (37)-(41):

$$\begin{aligned} 0 &= (1 - \nu^{-1})Y_t + \nu^{-1}\frac{W_t}{A_t}Y_t - \phi(\pi_t - \pi)\pi_t Y_t + \nu^{-1}\frac{\phi}{2}(\pi_t - \pi)^2 Y_t \\ &\quad + \beta\mathbb{E}_t \left[ \left( \frac{C_{t+1}/C_t}{A_{t+1}/A_t} \right)^{-\tau} \frac{A_t}{A_{t+1}}\phi(\pi_{t+1} - \pi)\pi_{t+1}Y_{t+1} \right], \\ 1 &= \beta\mathbb{E}_t \left[ \left( \frac{C_{t+1}/C_t}{A_{t+1}/A_t} \right)^{-\tau} \frac{A_t}{A_{t+1}}\frac{R_t}{\pi_{t+1}} \right], \\ \frac{W_t}{A_t} &= \chi_H \left( \frac{C_t}{A_t} \right)^\tau, \\ R_t &= \left( r\pi^* \left( \frac{\pi}{\pi^*} \right)^{\psi_1} \left( \frac{Y_t}{Y_t^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{t-1}^{\rho_R} e^{\epsilon_{R,t}}, \\ C_t + \frac{\phi}{2}(\pi_t - \pi)^2 Y_t &= g_t^{-1}Y_t. \end{aligned}$$

Let  $y_t = Y_t/A_t$ ,  $y_t^* = Y_t^*/A_t$ , and  $c_t = C_t/A_t$ , then we have

$$\begin{aligned} 0 &= (1 - \nu^{-1})y_t + \nu^{-1}c_t^\tau y_t - \phi(\pi_t - \pi)\pi_t y_t + \nu^{-1}\frac{\phi}{2}(\pi_t - \pi)^2 y_t \\ &\quad + \beta\mathbb{E}_t [(c_{t+1}/c_t)^{-\tau} \phi(\pi_{t+1} - \pi)\pi_{t+1}y_{t+1}], \\ 1 &= \beta R_t \mathbb{E}_t \left[ (c_{t+1}/c_t)^{-\tau} \frac{1}{\gamma_{t+1}\pi_{t+1}} \right], \\ R_t &= \left( r\pi^* \left( \frac{\pi}{\pi^*} \right)^{\psi_1} \left( \frac{y_t}{y_t^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{t-1}^{\rho_R} e^{\epsilon_{R,t}}, \\ c_t + \frac{\phi}{2}(\pi_t - \pi)^2 y_t &= g_t^{-1}y_t. \end{aligned}$$

The steady-state values are given by

$$\begin{aligned} c_{ss} &= (1 - \nu)^{1/\tau}, & \pi_{ss} &= \bar{\pi}, \\ R_{ss} &= \bar{r}\bar{\pi} = \beta^{-1}\gamma\bar{\pi}, & y_{ss} &= g_{ss}. \end{aligned}$$

## A.2 Nonstochastic PEAs in the New Keynesian model

To solve the New Keynesian model presented in Section 3.1 by nonstochastic PEAs, we introduce auxiliary functions for the expectation terms (called the expectation functions) in the consumption Euler equation (10) and the New Keynesian Phillips curve (11). There are two distinct ways to apply nonstochastic PEAs. One is to fit polynomials to future variables, and the other is to fit polynomials to current variables. When we fit polynomials to future variables, we assume to know

the values of these functions at each grid point. When we fit polynomials to current variables, we interpolate and integrate the auxiliary functions to obtain the values of these functions in the future. Then, taking as given the expectation terms, we obtain the values of endogenous variables at each grid point analytically without nonlinear optimization routines. By fitting polynomials to current variables, the expectation terms to be integrated are simple polynomials; therefore, we can also apply the precomputation technique.

### A.2.1 Fitting polynomials to future variables

As in [Gust et al. \(2017\)](#), we define auxiliary functions for the expectation terms in the consumption Euler equation (10) and the New Keynesian Phillips curve (11) as follows<sup>37</sup>

$$e_c(R_{-1}^*, s) \equiv \beta \bar{\gamma}^{-1} R \int \left[ \frac{(c')^{-\tau}}{z' \pi'} \right] p(s'|s) ds',$$

$$e_\pi(R_{-1}^*, s) \equiv \beta \phi \int \left[ \frac{(c')^{-\tau} y'}{y} (\pi' - \bar{\pi}) \pi' \right] p(s'|s) ds'.$$

The modified TI with future PEA takes the following steps:

1. Make an initial guess for the expectation functions  $e^{(0)} \equiv (e_c^{(0)}, e_\pi^{(0)})$ .
2. Taking as given the expectation functions previously obtained  $e^{(i-1)}$ , solve the relevant equations for  $(c, \pi, y, R^*)$  and obtain the policy functions  $\sigma^{(i)} \equiv (\sigma_c^{(i)}, \sigma_\pi^{(i)}, \sigma_y^{(i)})$ .
3. Update the expectation functions to obtain  $e^{(i)}$  by using the policy function  $\sigma^{(i)}$ .
4. Repeat steps 2–3 until  $\|e^{(i)} - e^{(i-1)}\|$  is sufficiently small.

In step 2, taking as given the values of  $e_x^{(i-1)}(R_{j,-1}, s_m)$  for  $x \in \{c, \pi\}$  at each grid point indexed by  $(j, m)$ , we solve

$$c_{jm} = e_c^{(i-1)}(R_{j,-1}^*, s_m)^{-1/\tau},$$

$$0 = (1 - \nu^{-1}) - \phi(\pi_{jm} - \bar{\pi}) \left[ \pi_{jm} - \frac{1}{2\nu} (\pi_{jm} - \bar{\pi}) \right] + e_c^{(i-1)}(R_{j,-1}^*, s_m)^{-1} \left( \nu^{-1} + e_\pi^{(i-1)}(R_{j,-1}^*, s_m) \right),$$

$$y_{jm} = \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm} - \bar{\pi})^2 \right]^{-1} c_{jm},$$

$$R_{jm}^* = \left( \bar{r} \bar{\pi} \left( \frac{\pi_{jm}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^* e^{\epsilon_{R,m}},$$

for  $(c_{jm}, \pi_{jm}, y_{jm}, R_{jm}^*)$ . We analytically obtain  $c_{jm}$  immediately and  $\pi_{jm}$  as a solution to the second-order polynomial.<sup>38</sup> We then have intermediate policy functions  $\sigma_c^{(i)}(R_{-1}^*, s)$ ,  $\sigma_\pi^{(i)}(R_{-1}^*, s)$ , and  $\sigma_y^{(i)}(R_{-1}^*, s)$ . Note that we do not solve the nonlinear equations by using optimization routines. Instead, we utilize a mapping between  $(e_c^{(i-1)}, e_\pi^{(i-1)})$  and  $(\sigma_c^{(i)}, \sigma_\pi^{(i)}, \sigma_y^{(i)})$ .

<sup>37</sup>We include  $R$  in  $e_c(R_{-1}^*, s)$  and  $y$  in  $e_\pi(R_{-1}^*, s)$  so that we can avoid solving a root-finding problem.

<sup>38</sup>In particular, we solve the following second-order polynomial

$$\alpha_0 - 2\alpha_1 \pi_{jm} + \alpha_2 \pi_{jm}^2 = 0$$

We also update

$$e_c^{(i)}(R_{j,-1}^*, s_m) = \beta\bar{\gamma}^{-1} R_{jm} \int \left[ \frac{\hat{\sigma}_c^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta})^{-\tau}}{z' \hat{\sigma}_\pi^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta})} \right] p(s'|s_m) ds',$$

$$e_\pi^{(i)}(R_{j,-1}^*, s_m) = \beta\phi \int \left[ \hat{\sigma}_c^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta})^{-\tau} \frac{\hat{\sigma}_y^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta})}{y_{jm}} \left( \hat{\sigma}_\pi^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta}) - \bar{\pi} \right) \hat{\sigma}_\pi^{(i)}(R_{jm}^*, s'; \boldsymbol{\theta}) \right] p(s'|s_m) ds'.$$

where  $R_{jm}^*$  and  $y_{jm}$  are obtained from the previous step. Note that we interpolate the values of  $\hat{\sigma}_x^{(i)}(R^*, s'; \boldsymbol{\theta})$  for  $x \in \{c, \pi, y\}$  (or equivalently  $\hat{e}_x^{(i-1)}(R^*, s'; \boldsymbol{\theta})$  for  $x \in \{c, \pi\}$ ) by Chebyshev polynomials. We also compute integrals numerically with regard to  $s'$ .

### A.2.2 Fitting polynomials to current variables

Alternatively, we define auxiliary expectation functions as follows

$$v_c(R_{-1}^*, s) \equiv \beta\bar{\gamma}^{-1} \left[ \frac{c^{-\tau}}{z\pi} \right],$$

$$v_\pi(R_{-1}^*, s) \equiv \beta\phi [c^{-\tau} y (\pi - \bar{\pi}) \pi].$$

The modified TI with current PEA takes the following steps:

1. Make an initial guess for the expectation and policy functions  $\{v^{(0)}, \sigma^{(0)}\}$ .
2. Given the expectation and policy functions previously obtained  $\{v^{(i-1)}, \sigma^{(i-1)}\}$ , solve the relevant equations for  $(c, \pi, y, R^*)$ .
3. Update the expectation and policy functions.
4. Repeat steps 2–3 until both  $\|v^{(i)} - v^{(i-1)}\|$  and  $\|\sigma^{(i)} - \sigma^{(i-1)}\|$  are small enough.

for  $\pi_{jm}$ , where

$$\alpha_0 = \frac{\phi\bar{\pi}^2}{2\nu} + (1 - \nu^{-1}) + e_c^{(i-1)}(R_{j,-1}^*, s_m)^{-1} \left( \nu^{-1} + e_\pi^{(i-1)}(R_{j,-1}^*, s_m) \right),$$

$$\alpha_1 = \phi\bar{\pi}(\nu^{-1} - 1)/2,$$

$$\alpha_2 = \phi \left( \frac{1}{2\nu} - 1 \right).$$

We select the root  $\pi_{jm} = \alpha_1/\alpha_2 - \sqrt{(\alpha_1/\alpha_2)^2 - \alpha_0}$  of the polynomial and ignore the other root.

Specifically, in step 2, taking the expectation function  $v^{(i-1)}(R^*, s')$  and the values of  $\sigma^{(i-1)}(R_{j,-1}^*, s_m)$  at each grid point as given, we solve

$$\begin{aligned} c_{jm} &= \left\{ R_{jm}^* \int \hat{v}_c^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta}) p(s'|s_m) ds' \right\}^{-1/\tau}, \\ 0 &= \left( (1 - \nu^{-1}) + \nu^{-1} c_{jm}^\tau - \phi(\pi_{jm} - \bar{\pi}) \left[ \pi_{jm} - \frac{1}{2\nu} (\pi_{jm} - \bar{\pi}) \right] \right) c_{jm}^{-\tau} y_{jm} \\ &\quad + \beta \phi \left\{ \int \hat{v}_\pi^{(i-1)}(R_{jm}^*, s'; \boldsymbol{\theta}) p(s'|s_m) ds' \right\}, \\ y_{jm} &= \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm} - \bar{\pi})^2 \right]^{-1} c_{jm}, \\ R_{jm}^* &= \left( \bar{r}\bar{\pi} \left( \frac{\pi_{jm}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^{*\rho_R} e^{\epsilon_{R,m}}, \end{aligned}$$

Note that we interpolate the values of  $\hat{v}_x^{(i-1)}(R^*, s'; \boldsymbol{\theta})$  for  $x \in \{c, \pi\}$  by the Chebyshev polynomials. We use a successive approximation so that we know the values of  $R_{jm}^* = \sigma_{R^*}^{(i-1)}(R_{j,-1}^*, s_m)$  and  $y_{jm} = \sigma_y^{(i-1)}(R_{j,-1}^*, s_m)$ .<sup>39</sup> Then, we analytically obtain  $c_{jm}$  immediately and  $\pi_{jm}$  as a solution to the second-order polynomial.<sup>40</sup> In addition, to compute integrals of  $\hat{v}_x^{(i-1)}(R^*, s'; \boldsymbol{\theta})$ , we can apply the precomputation technique as  $\hat{v}_x^{(i-1)}(R^*, s'; \boldsymbol{\theta})$  has a simple parameterized form.

In step 3, we update the expectation and policy functions:

$$\begin{aligned} \sigma_y^{(i)}(R_{j,-1}^*, s_m) &= y_{jm} = \left[ g_m^{-1} - \frac{\phi}{2} (\pi_{jm} - \bar{\pi})^2 \right]^{-1} c_{jm}, \\ \sigma_{R^*}^{(i)}(R_{j,-1}^*, s_m) &= R_{jm}^* = \left( \bar{r}\bar{\pi} \left( \frac{\pi_{jm}}{\bar{\pi}} \right)^{\psi_1} \left( \frac{y_{jm}}{y^*} \right)^{\psi_2} \right)^{1-\rho_R} R_{j,-1}^{*\rho_R} e^{\epsilon_{R,m}}, \\ v_c^{(i)}(R_{j,-1}^*, s_m) &= \beta \bar{\gamma}^{-1} \left[ \frac{c_{jm}^{-\tau}}{z_m \pi_{jm}} \right], \\ v_\pi^{(i)}(R_{j,-1}^*, s_m) &= \beta \phi \left[ c_{jm}^{-\tau} y_{jm} (\pi_{jm} - \bar{\pi}) \pi_{jm} \right]. \end{aligned}$$

---

<sup>39</sup>This is also known as fixed-point iteration.

<sup>40</sup>We solve the following second-order polynomial

$$\alpha_0 - 2\alpha_1 \pi_{jm} + \alpha_2 \pi_{jm}^2 = 0$$

for  $\pi_{jm}$ , where

$$\begin{aligned} \alpha_0 &= \frac{\phi \bar{\pi}^2}{2\nu} + (1 - \nu^{-1}) + c_{jm}^\tau \left( \nu^{-1} + \frac{\int v_\pi^{(i-1)}(\sigma_{R^*}^{(i-1)}(R_{j,-1}^*, s_m), s') p(s'|s_m) ds'}{\sigma_y^{(i-1)}(R_{j,-1}^*, s_m)} \right), \\ \alpha_1 &= \phi \bar{\pi} (\nu^{-1} - 1)/2, \\ \alpha_2 &= \phi \left( \frac{1}{2\nu} - 1 \right). \end{aligned}$$

We select the root  $\pi_{jm} = \alpha_1/\alpha_2 - \sqrt{(\alpha_1/\alpha_2)^2 - \alpha_0}$  of the polynomial and ignore the other root.