Reprinted from

HANDBOOK OF COMPUTATIONAL ECONOMICS

VOLUME I

Edited by

HANS M. AMMAN
University of Amsterdam

DAVID A. KENDRICK
University of Texas

and

JOHN RUST
University of Wisconsin

1996
ELSEVIER
AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • TOKYO

Chapter 3

COMPUTATIONAL METHODS FOR MACROECONOMETRIC MODELS

RAY C. FAIR

Yale University

Contents

1. Introduction 144
2. Notation 145
3. Two stage least squares 146
4. 3SLS and FIML 146
5. Two stage least absolute deviations 150
6. The Gauss–Seidel technique 152
7. Stochastic simulation 155
   7.1. Numerical procedures for drawing values 156
8. Optimal control 157
   8.1. Stochastic simulation option 158
9. Asymptotic distribution accuracy 159
10. Solution and FIML estimation of RE models 160
    10.1. Introduction 160
    10.2. The solution method 161
    10.3. Computational costs 164
    10.4. FIML estimation 165
    10.5. Stochastic simulation 166
    10.6. Conclusion 168

References 169

Handbook of Computational Economics, Volume I, Edited by H.M. Amman, D.A. Kendrick and J. Rust

1. Introduction

Advances in computer hardware in the last two decades have considerably lessened the computational burden of working with large scale macroeconometric models. Most methods for single equations are now computationally trivial, and many methods for complete models are now routine. In particular, the availability of fast, inexpensive computers has made stochastic simulation routine, and this has greatly expanded the ways in which models can be tested and analyzed.

This chapter discusses computational methods for the estimation and analysis of macroeconometric models. The focus is on methods that, while possibly computationally routine, are at least not trivial; computationally trivial methods are not discussed. Most of the methods discussed are methods for complete models. Nonlinear optimization algorithms, such as the DFP algorithm, are not discussed. The reader is assumed to be familiar with these algorithms.

Much of the material in this chapter is taken from Fair (1984) and (1994), where the methods are both discussed and applied, and the reader is referred to these two sources for the applications. To save space, no applications are presented in this chapter. It will help in what follows, however, to have an idea of the size of the model to which the methods have been applied. The latest version of this model, which will be called the "US" model, is presented in Fair (1994). It consists of 30 stochastic equations, 101 identities, and a little over 100 exogenous variables. The basic estimation period is 1954:1–1993:2, for a total of 158 observations. There are 166 unrestricted coefficients to estimate.

No computer times are reported in this chapter. Advances in hardware are so rapid that any times reported now would be out of date even by the time this book is published. Suffice it to say that none of the methods discussed in this chapter – with the possible exception of FIML estimation of models with rational expectations – are currently impractical in the sense of requiring days of personal computer time to run.

All the methods discussed in this chapter have been programmed into the Fair–Parke (1993) program, which is available for distribution. One advantage of this program is that once a model has been set up in the program, the methods can be carried out with a few simple commands. The only real work is setting up the model.

What I have tried to show in this chapter is that for the most part computational issues are no longer a problem in macroeconometric model building. Computational requirements should not be a major constraint in the advancement of the field in the future.

Finally, it should be stressed that this chapter is not meant to be a survey of the field. It is obviously beyond the scope of one paper to survey the literature on all the numerical methods that are used in macroeconometric modeling – methods for the various estimators, for deterministic simulation, for Monte Carlo/stochastic simulation, for optimal control, for nonlinear models, for rational expectations models, etc. There are not only an enormous number of methods, but many of them pertain to non

macroeconomic models as well. I have instead focused on those methods that I have found useful in my own work and that are in the Fair–Parke program.

2. Notation

The general model considered in this chapter can be dynamic, nonlinear, and simultaneous and can have autoregressive errors of any order. The model is written as:

$$f_i(y_t, x_t, \alpha_i) = u_{it}, \quad i = 1, \ldots, n, \quad t = 1, \ldots, T, \tag{1}$$

where y_t is an n-dimensional vector of endogenous variables, x_t is a vector of predetermined variables (including lagged endogenous variables), α_i is a vector of unknown coefficients, and u_{it} is the error term for equation i for observation t. It will be assumed that the first m equations are stochastic, with the remaining u_{it} $(i = m+1, \ldots, n)$ identically zero for all t.

The following notation is also used. u_i denotes the T-dimensional vector $(u_{i1}, \ldots, u_{iT})'$. G_i' denotes the $k_i \times T$ matrix whose tth column is $\partial f_i(y_t, x_t, \alpha_i)/\partial \alpha_i$, where k_i is the dimension of α_i. α denotes the vector of all the unknown coefficients in the model: $\alpha' = (\alpha_1', \ldots, \alpha_m')'$. The dimension of α is k, where $k = \sum_{i=1}^{m} k_i$. Finally, Z_i denotes a $T \times K_i$ matrix of predetermined variables that are to be used as first stage regressors for the 2SLS technique.

The following additional notation is needed when discussing the FIML and 3SLS estimators. J_t denotes the $n \times n$ Jacobian whose ij element is $\partial f_i/\partial y_{jt}, (i, j = 1, \ldots, n)$. u denotes the $m \cdot T$-dimensional vector $(u_{11}, \ldots, u_{1T}, \ldots, u_{m1}, \ldots, u_{mT})'$. G' denotes the $k \times m \cdot T$ matrix

$$G' = \begin{bmatrix} G_1' & 0 & \cdots & 0 \\ 0 & G_2' & & \\ \vdots & & \ddots & \\ 0 & & & G_m' \end{bmatrix}. \tag{2}$$

Finally, u_t denotes the m-dimensional vector $(u_{1t}, \ldots, u_{mt})$, and Σ denotes the $m \times m$ covariance matrix of u_t.

Each equation in (1) is assumed to have been transformed to eliminate any autoregressive properties of its error term. If the error term in the untransformed version, say w_{it} in equation i, follows a rth order autoregressive process, $w_{it} = \rho_{1i} w_{it-1} + \cdots + \rho_{ri} w_{it-r} + u_{it}$, where u_{it} is *iid*, then equation i is assumed to have been transformed into one with u_{it} on the right hand side. The autoregressive coefficients $\rho_{1i}, \ldots, \rho_{ri}$ are incorporated into the α_i coefficient vector, and the additional lagged values that are involved in the transformation are incorporated into the x_t vector. This transformation makes the equation nonlinear in coefficients if it were not

otherwise, but this adds no further complications to the model because it is already allowed to be nonlinear. It does result in the "loss" of the first r observations, but this has no effect on the asymptotic properties of the estimators. u_{it} in (1) can thus be assumed to be *iid* even though the original error term may follow an autoregressive process.

3. Two stage least squares

Probably the most widely used estimation technique for single equations that produces consistent estimates is two stage least squares (2SLS). Although the computation of 2SLS estimates is trivial, it will be useful for reference purposes to present the estimator. The 2SLS estimate of α_i (denoted $\hat{\alpha}_i$) is obtained by minimizing

$$S_i = u_i' Z_i \left(Z_i' Z_i\right)^{-1} Z_i' u_i = u_i' D_i u_i \tag{3}$$

with respect to α_i. Z_i can differ from equation to equation. An estimate of the covariance matrix of $\hat{\alpha}_i$ (denoted $\widehat{V}_{2ii}$) is

$$\widehat{V}_{2ii} = \hat{\sigma}_{ii} \left(\widehat{G}_i' D_i \widehat{G}_i\right)^{-1} \tag{4}$$

where $\widehat{G}_i$ is G_i evaluated at $\hat{\alpha}_i$, $\hat{\sigma}_{ii} = T^{-1} \sum_{t=1}^{T} \hat{u}_{it}^2$, and $\hat{u}_{it} = f_i(y_t, x_t, \hat{\alpha}_i)$.

The 2SLS estimate of the $k \times k$ covariance matrix of all the coefficient estimates in the model (denoted $\widehat{V}_2$) is

$$\widehat{V}_2 = \begin{bmatrix} \widehat{V}_{211} & \cdots & \widehat{V}_{21m} \\ \vdots & & \vdots \\ \widehat{V}_{2m1} & \cdots & \widehat{V}_{2mm} \end{bmatrix} \tag{5}$$

where

$$\widehat{V}_{2ij} = \hat{\sigma}_{ij} \left(\widehat{G}_i' D_i \widehat{G}_i'\right)^{-1} \left(\widehat{G}_i' D_i D_j \widehat{G}_j'\right) \left(\widehat{G}_j' D_j \widehat{G}_j'\right)^{-1} \tag{6}$$

and $\hat{\sigma}_{ij} = T^{-1} \sum_{t=1}^{T} \hat{u}_{it} \hat{u}_{jt}$.

4. 3SLS and FIML

3SLS estimates of α are obtained by minimizing

$$S = u' \left[\widehat{\Sigma}^{-1} \otimes Z(Z'Z)^{-1} Z'\right] u = u' D u \tag{7}$$

with respect to α, where $\widehat{\Sigma}$ is a consistent estimate of Σ and Z is a $T \times K$ matrix of predetermined variables. An estimate of the covariance matrix of the coefficient estimates (denoted $\widehat{V}_3$) is

$$\widehat{V}_3 = \left(\widehat{G}' D \widehat{G}\right)^{-1} \tag{8}$$

where $\widehat{G}$ is G evaluated at the 3SLS estimate of α. Σ is usually estimated from the 2SLS estimated residuals.

Under the assumption that u_t is independently and identically distributed as multivariate normal $N(0, \Sigma)$, FIML estimates of α are obtained by maximizing

$$L = -\frac{T}{2} \log |\Sigma| + \sum_{t=1}^{T} \log |J_t| \tag{9}$$

with respect to α. An estimate of the covariance matrix of the FIML estimates (denoted $\widehat{V}_4$) is

$$\widehat{V}_4 = -\left(\frac{\partial^2 L}{\partial \alpha \partial \alpha'}\right)^{-1} \tag{10}$$

where the derivatives are evaluated at the optimum.

The FIML computational problem can be separated into two main parts; the first is to find a fast way of computing L in (9) for a given value of α, and the second is to find an algorithm capable of maximizing L.

The main cost of computing L is computing the Jacobian term. Two savings can be made here. One is to exploit the sparseness of the Jacobian. The number of nonzero elements in J_t is usually much less than n^2. For the US model, for example, n is 131 (so $n^2 = 17,161$), whereas the number of nonzero elements is only 556. Considerable time can be saved by using sparse matrix routines to calculate the determinant of J_t.

The second saving is based on an approximation. Consider approximating $\sum_{t=1}^{T} \log |J_t|$ by simply the average of the first and last terms in the summation multiplied by T: $(T/2)(\log |J_1| + \log |J_T|)$. Let S_0 denote the true summation, and let S_1 denote the approximation. It turns out in many applications that $S_0 - S_1$ does not change very much as the coefficients change from their starting values (say 2SLS estimates) to the values that maximize the likelihood function. In other words, $S_0 - S_1$ is nearly a constant. This means that S_1 can be used instead of S_0 in computing L, and thus considerable computer time is saved since the determinant of the Jacobian only needs to be computed twice rather than T times for each evaluation of L. As noted above, T is 158 for the US model. Using S_1 in place of S_0 means, of course, that the coefficient values that maximize the likelihood function are not the exact FIML estimates. If one is concerned about the accuracy of the approximation, one

can switch from S_1 to S_0 after finding the maximum using S_1. If the approximation is good, one should see little further change in the coefficients; otherwise additional iterations using the algorithm will be needed to find the true maximum.

The choice of algorithm turns out to be crucial in maximizing L for large nonlinear models. My experience is that general purpose algorithms like DFP do not work for large problems, and in fact the only algorithm that does seem to work is the Parke (1982) algorithm, which is a special purpose algorithm designed for FIML and 3SLS estimation. The algorithm exploits two key features of models. The first is that the mean of a particular equation's estimated residuals is approximately zero for FIML and 3SLS estimates. For OLS this must be true, and empirically it turns out that it is approximately true for other estimators. The second feature is that the correlation of coefficient estimates within an equation is usually much greater than the correlation of coefficients across equations.

The problem with algorithms like DFP that require first derivatives is that the computed gradients do not appear to be good guides regarding the directions to move. Gradients are computed by perturbing one coefficient at a time. When a coefficient is changed without the constant term in the equation also being changed to preserve the mean of the residuals, a large change in L may result (and thus a large computed derivative), and this can be misleading. The Parke algorithm avoids this problem by spending most of its time perturbing two coefficients at once, namely a given coefficient and the constant term in the equation in which the coefficient appears. The constant term is perturbed to keep the mean of the residuals unchanged. (The algorithm does not, of course, do this all the time, since the means of the residuals must also be estimated.) To take advantage of the generally larger correlation within an equation than between equations, the Parke algorithm spends more time searching within equations than between them. General purpose algorithms do not do this, since they have no knowledge of the structure of the problem.

If only a few coefficients are changed before a new value of L is computed, considerable savings can be made by taking advantage of this fact. If, for example, the coefficients that are changed are not in the Jacobian, the Jacobian term does not have to be recomputed. If only a few equations are affected by the change in coefficients, only a few rows and columns in the Σ matrix have to be recomputed. Since the Parke algorithm spends much of its time perturbing two coefficients at a time, it is particularly suited for these savings.

The estimated covariance matrix for the FIML coefficient estimates, $\hat{V}_4$ in (10), is difficult to compute. It is not part of the output of the Parke algorithm, and thus extra work is involved in computing it once the algorithm has found the optimum. My experience is that simply trying to compute the second derivatives of L numerically does not result in a positive definite matrix. Although the true second derivative matrices at the optimum are undoubtedly positive definite, they seem to be nearly singular. If this is true, small errors in the numerical approximations to the second derivatives may be sufficient to make the matrix not positive definite.

Fortunately, there is an approach to computing $\widehat{V}_4$ that does work, which is derived from Parke (1982). Parke's results suggest that the inadequate numerical approximations may be due to the fact that the means of the right hand side variables in the estimated equations are not zero. If so, the problem can be solved by subtracting the means from the right hand side variables before taking numerical derivatives. Let β denote the coefficient vector that pertains to the model after the means have been subtracted, and let α denote the original coefficient vector. The relationship between α and β is

$$\alpha = M \cdot \beta \tag{11}$$

where M is a $k \times k$ matrix that is composed of the identity matrix plus additional nonzero elements that represent the means adjustments. Unless there are constraints across equations, M is block diagonal. Assume, for example, that the first equation of the model is

$$y_{1t} = \beta_1 + \beta_2(y_{2t} - m_2) + \beta_3(y_{3t} - m_3) + u_{1t}, \quad t = 1, \ldots, T, \tag{12}$$

where m_2 and m_3 are the sample means of y_{2t} and y_{3t} respectively. This equation can be written

$$\begin{aligned} y_{1t} &= \beta_1 - \beta_2 m_2 - \beta_3 m_3 + \beta_2 y_{2t} + \beta_3 y_{3t} + u_{1t} \\ &= \alpha_1 + \alpha_2 y_{2t} + \alpha_3 y_{3t} + u_{1t}, \quad t = 1, \ldots, T. \end{aligned} \tag{13}$$

In this case the part of (11) that corresponds to the first equation is

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} 1 & -m_2 & -m_3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}. \tag{14}$$

Parke found that the covariance matrix of β could easily be computed numerically. Let $\widehat{V}_4(\beta)$ denote this matrix:

$$\widehat{V}_4(\beta) = -\left(\frac{\partial^2 L(M \cdot \beta)}{\partial \beta \partial \beta'} \right)^{-1}. \tag{15}$$

Given $\widehat{V}_4(\beta)$, the covariance matrix of α is simply

$$\widehat{V}_4 = M \cdot \widehat{V}_4(\beta) \cdot M'. \tag{16}$$

$\widehat{V}_4$ can thus be obtained by first computing the covariance matrix of the coefficients of the transformed model (that it, the model in which the right hand side variables

have zero means) and then using (16) to get the covariance matrix of the original coefficients.

Using the Parke algorithm and the various savings discussed above, I have found it feasible to obtain FIML estimates of the US model. In fact, the main work in using the FIML estimator is not the computational burden but making sure that no errors have been made in taking the derivatives for the Jacobian.

Consider now the 3SLS estimation problem, which is to minimize (7). The only cost saving to note for this problem is that the D matrix, which is $m \cdot T \times m \cdot T$, need not be calculated from scratch each time (7) is computed if only a few coefficients are changed. In other words, pieces of D can be saved and used many times before needing to be recomputed. This saves considerable time because D is large. For example, for $T = 158$ and $m = 30$, D is 4740×4740. I have also found the 3SLS estimates easy to compute using the Parke algorithm.

5. Two stage least absolute deviations

A single equation estimator for which there are some computational issues is two stage least absolute deviations (2SLAD). This estimator is as follows. It is assumed for this estimator that the model in (1) can be written:

$$y_{it} = h_i(y_t, x_t, \alpha_i) + u_{it}, \quad i = 1, \ldots, n, \quad t = 1, \ldots, T, \tag{17}$$

where in the ith equation y_{it} appears only on the left-hand side. Let $\hat{y}_i = D_i y_i$ and $\hat{h}_i = D_i h_i$, where, as above, $D_i = Z_i(Z_i'Z_i)^{-1}Z_i'$, where Z_i is a matrix of first stage regressors. There are two ways of looking at the 2SLAD estimator. One is that it minimizes

$$\sum_{t=1}^{T} |\hat{y}_{it} - \hat{h}_{it}| \tag{18}$$

and the other is that it minimizes

$$\sum_{t=1}^{T} |y_{it} - \hat{h}_{it}|. \tag{19}$$

Amemiya (1982) has proposed minimizing

$$\sum_{t=1}^{T} |qy_{it} + (1-q)\hat{y}_{it} - \hat{h}_{it}| \tag{20}$$

where q is chosen ahead of time by the investigator. The estimator that is based on minimizing (20) will be called the 2SLAD estimator.

The 2SLAD computational problem is to minimize

$$\sum_{t=1}^{T} |v_{it}| \tag{21}$$

with respect to α_i, where $v_{it} = qy_{it} + (1-q)\hat{y}_{it} - \hat{h}_{it}$. This computational problem is not particularly easy, especially when v_{it} is a nonlinear function of α_i. For example, I have had no success in minimizing (21) using Powell's (1964) no derivative algorithm.

Because standard algorithms do not work, other approaches must be tried. I have found the following approach to work well, which uses the fact that

$$\sum_{t-1}^{T} |v_{it}| = \sum_{t=1}^{T} \frac{v_{it}^2}{|v_{it}|} = \sum_{t=1}^{T} \frac{v_{it}^2}{w_{it}} \tag{22}$$

where $w_{it} = |v_{it}|$. For a given set of values of w_{it} $(t = 1, \ldots, T)$, minimizing (22) is simply a weighted least squares problem. If v_{it} is a linear function of α_i, closed form expressions exist for $\hat{\alpha}_i$; otherwise a nonlinear optimization algorithm can be used. This suggests the following iterative procedure:

1. Pick an initial set of values of w_{it}. These can be the absolute values of the 2SLS estimated residuals.
2. Given these values, minimize (22).
3. Given the estimate of α_i from step 2, compute new values of v_{it} and thus new values of w_{it}.
4. With the new weights, go back to step 2 and minimize (22) again. Keep repeating steps 2 and 3 until successive estimates of α_i are within some prescribed tolerance level. If on any step some value of w_{it} is smaller than some small preassigned number (say ϵ), the value of w_{it} should be set equal to ϵ.

In the case in which the equation to be estimated is linear in coefficients, the closed form expression for $\hat{\alpha}_i$ for a given set of values of w_{it} is

$$\hat{\alpha}_i = \left(\hat{X}_i^{*\prime} \hat{X}_i^{*}\right)^{-1} \hat{X}_i^{*\prime} \hat{y}_i^{*}, \tag{23}$$

$\hat{X}_i^{*}$ is the same as $\hat{X}_i$, where $\hat{X}_i = D_i X_i$, except that each element in row t of $\hat{X}_i$ is divided by $\sqrt{w_{it}}$. The vector $\hat{y}_i^{*}$ equals $qy_i + (1-q)\hat{y}_i$ except that row t is divided by $\sqrt{w_{it}}$. ($\hat{y}_i$ equals $D_i y_i$.)

The accuracy of the estimates using this approach is a function of ϵ: the smaller is ϵ, the greater is the accuracy. If v_{it} is a linear function of α_i, the estimates will

never be exact because the true estimates correspond to k_i values of w_{it} being exactly equal to zero, where k_i is the number of elements of α_i. One might think that this would be a serious problem in practice, but I have not found it to be. I typically use a value of ϵ of 0.0000001 and a percentage stopping criterion for successive coefficient estimates of 0.001, and with these numbers it is seldom the case that any weight is less than ϵ. (I typically use a value of q of 0.5.) The method also works well when the equation is nonlinear in coefficients, such as when the error term is assumed to follow an autoregressive process and the autoregressive coefficients are estimated along with the structural coefficients.

6. The Gauss–Seidel technique

Turn now from estimation to solution. Most macroeconometric models are solved using the Gauss–Seidel technique. It is a remarkably simple technique and in most cases works remarkably well. Very early on it became the method of choice for solution purposes. Because the technique is so widely used, it is important to understand what it does. The technique is easiest to describe by means of an example.

Assume that the model in (1) consists of three equations, and let x_{it} denote the vector of predetermined variables in equation i. The model is as follows:

$$f_1(y_{1t}, y_{2t}, y_{3t}, x_{1t}, \alpha_1) = u_{1t}, \tag{24}$$

$$f_2(y_{1t}, y_{2t}, y_{3t}, x_{2t}, \alpha_2) = u_{2t}, \tag{25}$$

$$f_3(y_{1t}, y_{2t}, y_{3t}, x_{3t}, \alpha_3) = u_{3t} \tag{26}$$

where y_{1t}, y_{2t}, and y_{3t} are scalars. The technique requires that the equations be rewritten with each endogenous variable on the left hand side of one equation. This is usually quite easy for macroeconometric models, since most equations have an obvious left hand side variable. If, say, the left hand side variable for (25) is $\log(y_{2t}/y_{3t})$, then y_{2t} can be written on the left hand side by taking exponents and multiplying the resulting expression by y_{3t}. The technique does not require that each endogenous variable be isolated on the left hand side; the left hand side variable can also appear on the right hand side. It is almost always possible in macroeconometric work, however, to isolate the variable, and this will be assumed in the following example.

The model (24)–(26) will be written:

$$y_{1t} = g_1(y_{2t}, y_{3t}, x_{1t}, \alpha_1, u_{1t}), \tag{24}'$$

$$y_{2t} = g_2(y_{1t}, y_{3t}, x_{2t}, \alpha_2, u_{2t}), \tag{25}'$$

$$y_{3t} = g_3(y_{1t}, y_{2t}, x_{3t}, \alpha_3, u_{3t}). \tag{26}'$$

In order to solve the model, values of the coefficients and the error terms are needed. Given these values and given values of the predetermined variables, the solution proceeds as follows. Initial values of the endogenous variables are guessed. These are usually either the actual values or predicted values from the previous period. Given these values, (24)′–(26)′ can be solved for a new set of values. This requires one "pass" through the model: each equation is solved once. One pass through the model is also called an "iteration". Given this new set of values, the model can be solved again to get another set, and so on. Convergence is reached if for each endogenous variable the values on successive iterations are within some prescribed tolerance level.

There are two main options that can be used when passing through the model. One is to use the values from the previous iteration for all the computations for the current iteration, and the other is to use, whenever possible, the values from the current iteration in solving the remaining equations. Following the second option in the example just given would mean using the current solution for y_{1t} in the solution of y_{2t} and y_{3t} and using the current solutions for y_{1t} and y_{2t} in the solution of y_{3t}. In most cases convergence is somewhat faster using the second option. If the second option is used, the order of the equations obviously matters in terms of the likely speed of convergence. The first option is sometimes called the Jacobi technique rather than the Gauss–Seidel technique, but for present purposes both options will be referred to as the Gauss–Seidel technique.

There is no guarantee that the Gauss–Seidel technique will converge. It is easy to construct examples in which it does not, and I have seen many examples in practice where it did not. The advantage of the technique, however, is that it can usually be made to converge (assuming an actual solution exists) with sufficient damping. By "damping" is meant the following. Let $\hat{y}_{1t}^{(n-1)}$ denote the solution value of y_{1t} for iteration $n-1$ (or the initial value if n is 1), and let $\hat{\hat{y}}_{1t}^{(n)}$ denote the value computed by solving (24)′ on iteration n. Instead of using $\hat{\hat{y}}_{1t}^{(n)}$ as the solution value for iteration n, one can instead adjust $\hat{y}_{1t}^{(n-1)}$ only partway toward $\hat{\hat{y}}_{1t}^{(n)}$:

$$\hat{y}_{1t}^{(n)} = \hat{y}_{1t}^{(n-1)} + \lambda\left(\hat{\hat{y}}_{1t}^{(n)} - \hat{y}_{1t}^{(n-1)}\right), \quad 0 < \lambda \leqslant 1. \tag{27}$$

If λ is 1, there is no damping, but otherwise there is. Damping can be done for any or all of the endogenous variables, and different values of λ can be used for different variables.

My experience is that one can usually make λ small enough to achieve convergence. The cost of damping is, of course, slow convergence. In some cases I have seen values as low as 0.05 needed. In the vast majority of the problems that I have solved, however,

no damping at all was needed. Two other ways in which one can deal with problems of convergence are to try different starting values and to reorder the equations. This involves, however, more work than merely running the problem with lower values of λ, and I have generally not found it necessary to experiment with starting values and the order of the equations.

Note that nothing is changed in the foregoing discussion if, say, y_{1t} is also on the right hand side of (24)′. One still passes through the model in the same way. This generally means, however, that it takes longer to converge, and more damping may be required than if y_{1t} is only on the left hand side; thus it is better to isolate variables on the left hand side whenever possible.

The question of what to use for a stopping rule is not as easy as it might sound. The stopping rule can either be in absolute or percentage terms. In absolute terms it is

$$\left|\hat{y}_{it}^{(n)} - \hat{y}_{it}^{(n-1)}\right| < \epsilon_i \tag{28}$$

and in percentage terms it is

$$\left|\frac{\hat{y}_{it}^{(n)} - \hat{y}_{it}^{(n-1)}}{\hat{y}_{it}^{(n-1)}}\right| < \epsilon_i \tag{29}$$

where ϵ_i is the tolerance criterion for variable i. (If damping is used, $\hat{y}_{it}^{(n)}$ in (28) and (29) should be replaced with $\hat{\hat{y}}_{it}^{(n)}$.)

The problem comes in choosing the values for ϵ_i. It is inconvenient to have to choose different values of the tolerance criterion for different variables, and one would like to use just one value throughout. This is not, however, a sensible procedure if the units of the variables differ and if the absolute criterion is used. Setting the tolerance criterion small enough for the required accuracy of the variable with the smallest units is likely to lead to an excess number of iterations, since a large number of iterations are likely to be needed to satisfy the criterion for the variables with the largest units. Setting the criterion greater than this value, on the other hand, runs the risk of not achieving the desired accuracy for some variables. This problem is lessened if the percentage criterion is used, but in this case one must be concerned with variables that can be zero or close to zero.

My experience is that the number of iterations needed for convergence is quite sensitive to the stopping rule. It does not seem to be the case, for example, that once one has converged for most variables, one or two additional iterations increase the accuracy for the remaining variables very much. There is no real answer to this problem. One must do some initial experimentation to decide how many different values of ϵ_i are needed and whether to use the absolute or percentage criterion for a given variable.

7. Stochastic simulation

As noted in the Introduction, computer hardware advances have now made stochastic simulation routine, and this has greatly expanded the ways in which models can be tested and analyzed. Many applications of stochastic simulation are contained in Fair (1994), but these will not be discussed here. The following discussion will focus on computational aspects only. The notation in Section 2 will continue to be used.

Stochastic simulation requires that an assumption be made about the distribution of u_t. It is usually assumed that u_t is independently and identically distributed as multivariate normal $N(0, \Sigma)$, although other assumptions can clearly be used. Alternative assumptions simply change the way the error terms are drawn. Stochastic simulation also requires that consistent estimates of α_i be available for all i. Given these estimates, denoted $\hat{\alpha}_i$, consistent estimates of u_{it}, denoted $\hat{u}_{it}$, can be computed as $f_i(y_t, x_t, \hat{\alpha}_i)$. The covariance matrix Σ can then be estimated as $(1/T)\widehat{U}\widehat{U}'$, where $\widehat{U}$ is the $m \times T$ matrix of the values of $\hat{u}_{it}$.

Let u_t^* denote a particular draw of the m error terms for period t from the $N(0, \widehat{\Sigma})$ distribution. Given u_t^* and given $\hat{\alpha}_i$ for all i, one can solve the model for period t (using, say, the Gauss–Seidel technique). This is merely a deterministic simulation for the given values of the error terms and coefficients. Call this simulation a "repetition". Another repetition can be made by drawing a new set of values of u_t^* and solving again. This can be done as many times as desired. From each repetition one obtains a prediction of each endogenous variable. Let y_{it}^j denote the value on the jth repetition of variable i for period t. For J repetitions, the stochastic simulation estimate of the expected value of variable i for period t, denoted $\tilde{\mu}_{it}$, is

$$\tilde{\mu}_{it} = \frac{1}{J}\sum_{j=1}^{J} y_{it}^j. \tag{30}$$

Let

$$\sigma_{it}^{2j} = (y_{it}^j - \tilde{\mu}_{it})^2. \tag{31}$$

The stochastic simulation estimate of the variance of variable i for period t, denoted $\tilde{\sigma}_{it}^2$, is then

$$\tilde{\sigma}_{it}^2 = \frac{1}{J}\sum_{j=1}^{J} \sigma_{it}^{2j}. \tag{32}$$

Given the data from the repetitions, it is also possible to compute the variances of the stochastic simulation estimates and then to examine the precision of the estimates. The variance of $\tilde{\mu}_{it}$ is simply $\tilde{\sigma}_{it}^2/J$. The variance of $\tilde{\sigma}_{it}^2$, denoted $\text{var}(\tilde{\sigma}_{it}^2)$, is

$$\text{var}(\tilde{\sigma}_{it}^2) = \left(\frac{1}{J}\right)^2 \sum_{j=1}^{J} (\sigma_{it}^{2j} - \tilde{\sigma}_{it}^2)^2. \tag{33}$$

In many applications, one is interested in predicted values more than one period ahead, i.e., in predicted values from dynamic simulations. The above discussion can be easily modified to incorporate this case. One simply draws values for u_t for each period of the simulation. Each repetition is one dynamic simulation over the period of interest. For, say, an eight quarter period, each repetition yields eight predicted values, one per quarter, for each endogenous variable.

It is also possible to draw coefficients for the repetitions. Let $\hat{\alpha}$ denote, say, the 2SLS estimate of all the coefficients in the model, and let $\widehat{V}$ denote the estimate of the $k \times k$ covariance matrix of $\hat{\alpha}$. Given $\widehat{V}$ and given the normality assumption, an estimate of the distribution of the coefficient estimates is $N(\hat{\alpha}, \widehat{V})$. When coefficients are drawn, each repetition consists of a draw of the coefficient vector from $N(\hat{\alpha}, \widehat{V})$ and draws of the error terms as above.

7.1. Numerical procedures for drawing values

A standard way of drawing values of α^* from the $N(\hat{\alpha}, \widehat{V})$ distribution is to 1) factor numerically $\widehat{V}$ into PP', 2) draw k values of a standard normal random variable with mean 0 and variance 1, and 3) compute α^* as $\hat{\alpha} + Pe$, where e is the $k \times 1$ vector of the standard normal draws. Since $Eee' = I$, then $E(\alpha^* - \hat{\alpha})(\alpha^* - \hat{\alpha})' = EPee'P' = \widehat{V}$, which is as desired for the distribution of α^*. A similar procedure can be used to draw values of u_t^* from the $N(0, \widehat{\Sigma})$ distribution: $\widehat{\Sigma}$ is factored into PP', and u_t^* is computed as Pe, where e is a $m \times 1$ vector of standard normal draws.

An alternative procedure for drawing values of the error terms, derived from McCarthy (1972), has also been used in practice. For this procedure one begins with the $m \times T$ matrix of estimated error terms, $\widehat{U}$. T standard normal random variables are then drawn, and u_t^* is computed as $T^{-1/2}\widehat{U}e$, where e is a $T \times 1$ vector of the standard normal draws. It is easy to show that the covariance matrix of u_t^* is $\widehat{\Sigma}$, where, as earlier, $\widehat{\Sigma}$ is $(1/T)\widehat{U}\widehat{U}'$.

An alternative procedure is also available for drawing values of the coefficients. Given the estimation period (say, 1 through T) and given $\widehat{\Sigma}$, one can draw T values of u_t^* $(t = 1, \ldots, T)$. One can then add these errors to the model and solve the model over the estimation period (static simulation, using the original values of the coefficient estimates). The predicted values of the endogenous variables from this solution can

be taken to be a new data base, from which a new set of coefficients can be estimated. This set can then be taken to be one draw of the coefficients. This procedure is more expensive than drawing from the $N(\hat{\alpha}, \hat{V})$ distribution, since reestimation is required for each draw, but it has the advantage of not being based on a fixed estimate of the distribution of the coefficient estimates. It is, of course, based on a fixed value of $\hat{\Sigma}$ and a fixed set of original coefficient estimates.

8. Optimal control

There is a large literature on the use of optimal control techniques in macroeconometrics,[1] and it is beyond the scope of this chapter to review this literature. Instead, I will simply discuss one technique that I have found to be very useful in solving optimal control problems for large models. This technique is presented and applied in Fair (1974).

The first step in setting up an optimal control problem is to postulate an objective function. Assume that the period of interest is $t = 1, \ldots, T$. A general specification of the objective function is

$$W = h(y_1, \ldots, y_T, x_1, \ldots, x_T) \tag{34}$$

where W, a scalar, is the value of the objective function corresponding to values of the endogenous and exogenous variables for $t = 1, \ldots, T$. In most applications the objective function is assumed to be additive across time, which means that (34) can be written

$$W = \sum_{t=1}^{T} h_t(y_t, x_t) \tag{35}$$

where $h_t(y_t, x_t)$ is the value of the objective function for period t. The model can be taken to be the model in (1).

Let z_t be a k-dimensional vector of control variables, where z_t is a subset of x_t, and let z be the $k \cdot T$-dimensional vector of all the control values: $z = (z_1, \ldots, z_T)$. Consider first the deterministic case where the error terms in (1) are all set to zero. For each value of z one can compute a value of W by first solving the model (1) for $y_1, \ldots, y_T$ and then using these values along with the values for $x_1, \ldots, x_T$ to compute W in (34). Stated this way, the optimal control problem is choosing variables (the elements of z) to maximize an *unconstrained* nonlinear function. By substitution, the constrained maximization problem of maximizing W in (34) subject to the model is

[1] See Chow (1981) for a discussion of many of the techniques in this area.

transformed into the problem of maximizing an unconstrained function of the control variables:

$$W = \Phi(z) \tag{36}$$

where Φ stands for the mapping $z \rightarrow y_1, \ldots, y_T, x_1, \ldots, x_T \rightarrow W$. For nonlinear models it is generally not possible to express y_t explicitly in terms of x_t, which means that it is generally not possible to write W in (36) explicitly as a function of $x_1, \ldots, x_T$. Nevertheless, given values for $x_1, \ldots, x_T$, values of W can be obtained numerically for different values of z.

Given this setup, the problem can be turned over to a nonlinear maximization algorithm like DFP. For each iteration, the derivatives of Φ with respect to the elements of z, which are needed by the algorithm, can be computed numerically. Each iteration will thus require kT function evaluations for the derivatives plus a few more for the line search. Each function evaluation requires one solution (dynamic simulation) of the model for T periods plus the computation of W in (34) after $y_1, \ldots, y_T$ are determined.

There is one important cost saving feature regarding the method that should be noted. Assume that there are two control variables and that the length of the period is 30. The number of unknowns is thus 60, and therefore 60 function evaluations will have to be done per iteration to get the numerical derivatives. In perturbing the control values to get the derivatives, one should start from the end of the control period and work backward. When the control values for period 30 are perturbed, the solution of the model for periods 1 through 29 remains unchanged from the base solution, so these calculations can be skipped. The model only needs to be resolved for period 30. Similarly, when the control values for period 29 are perturbed, the model only needs to be resolved for periods 29 and 30, and so on. This cuts the cost of computing the derivatives roughly in half.

My experience is that algorithms like DFP are quite good at solving optimal control problems set up in the above way. See, for example, Fair (1974) for the use of the DFP algorithm to solve quite large problems.

8.1. Stochastic simulation option

Consider now the stochastic case, where the error terms in (1) are not zero. It is possible to convert this case into the deterministic case by simply setting the error terms to their expected values (usually zero). The problem can then be solved as above. In the nonlinear case this does not lead to the exact answer because the values of W that are computed numerically in the process of solving the problem are not the expected values. In order to compute the expected values correctly, stochastic simulation has to be done. In this case each function evaluation (i.e., each evaluation of he expected value of W for a given value of z) consists of the following.

1. A set of values of the error terms in (1) is drawn from an estimated distribution.
2. Given the values of the error terms, the model is solved for $y_1, \ldots, y_T$ and the value of W corresponding to this solution is computed from (34). Let $\widetilde{W}^j$ denote this value.
3. Steps 1 and 2 are repeated J times, where J is the number of repetitions.
4. Given the J values of $\widetilde{W}^j$ $(j = 1, \ldots, J)$, the expected value of W is the mean of these values:

$$\overline{W} = \frac{1}{J} \sum_{j=1}^{J} \widetilde{W}^j. \tag{37}$$

This procedure increases the cost of solving control problems by roughly a factor of J, and it is probably not worth the cost for most applications. The bias in predicting the endogenous variables that results from using deterministic rather than stochastic simulation is usually small, and thus the bias in computing the expected value of W using deterministic simulation is likely to be small.

9. Asymptotic distribution accuracy

Computer advances have also increased the ability to compute "exact" distributions of the estimators that are used for macroeconometric models. These distributions can then be compared to the asymptotic approximations of the distributions that are generally used for hypothesis testing to see how accurate the asymptotic approximations are. If some variables are not stationary, the asymptotic approximations may not be very good. In fact, much of the recent literature in time series econometrics has been concerned with the consequences of non stationary variables. As will be seen, computing the exact distributions requires stochastic simulation and reestimation.

The procedure for examining asymptotic distribution accuracy is as follows. Take an estimator, say 2SLS, 3SLS, or FIML, and estimate the model. Take these coefficient estimates, denoted $\hat{\alpha}$, as the base values, and compute $\widehat{\Sigma}$ using these estimates. From the $N(0, \widehat{\Sigma})$ distribution (assuming the normality assumption is used), draw a vector of the m error terms for each of the T observations. Given these error terms and $\hat{\alpha}$, solve the model for the entire period 1 through T. This is a dynamic simulation of the model over the entire estimation period. The lagged endogenous variable values in (1) are updated in the solution process. Also, the matrices of first stage regressors, Z_{it}, are updated to incorporate the new lagged endogenous variable values if the matrices are used in the estimation, as for 2SLS. The predicted values from this solution form a new data set. Given this data set, estimate the model by the technique in question, and record the set of estimates. This is one repetition. Repeat the draws, solution, and estimation for many repetitions, and record each set of estimates. (Remember that the

draws of the errors are always from the $N(0, \hat{\Sigma})$ distribution and that the coefficient vector used in the solution is always $\hat{\alpha}$.)

If J repetitions are done, one has J values of each coefficient estimate, which are likely to be a good approximation of the exact distribution. For ease of exposition, this distribution of the J values will be called the "exact distribution", although it is only an approximation because Σ is estimated rather than known. The asymptotic distribution can then be compared to this exact distribution to see how close the two distributions are.

Once an exact distribution has been computed, there are a number of ways to examine the closeness of the asymptotic distribution to it. For the application in Fair (1994) the median of the exact distribution was first compared to the coefficient estimate to examine the bias of the estimate. Given the median from the exact distribution and given the estimated standard error of the coefficient estimate from the asymptotic distribution, one can compute the value above which, say, 20 percent of the coefficient estimates should lie if the asymptotic distribution is correct. For 20 percent, this value is the median plus 0.84 times the estimated asymptotic standard error. One can then compute the actual percent of the coefficient estimates from the exact distribution that lie above this value and compare this percent to 20 percent. For the work in Fair (1994), this comparison was made for 20, 10, and 5 percent values and for both left and right tails.

The results that I have obtained so far show that the exact and asymptotic distributions are generally quite similar regarding their tail properties. If this conclusion holds up upon further work, it has important consequences. It means that the unit root problems that have received so much attention lately may not be of much concern to macro model builders. While the existence of unit roots can in theory cause the asymptotic approximations that are relied on in macroeconometrics to be way off, in practice they seem fairly accurate.

10. Solution and FIML estimation of RE models

10.1. Introduction

The rest of this chapter is concerned with the estimation and solution of models with rational expectations. As will be discussed, these models have severe computational requirements.

The single equation estimation of equations with rational expectations can be carried out using Hansen's (1982) method of moments estimator, and there are no serious computational requirements here. It is also possible, however, to use FIML to estimate models with rational expectations, and here there are serious computational issues. Methods for the solution and FIML estimation of these models are presented in Fair and Taylor (1983). The basic solution method, called the "extended path" (EP) method,

has come to be widely used for deterministic simulations of rational expectations models,[2] but probably because of the expense, the full information estimation method has not been tried by others. This earlier work discussed a "less expensive" method for obtaining full information estimates, but the preliminary results using the method were mixed. Since this earlier work, however, more experimenting with the less expensive method has been done, and it seems much more promising than was originally thought. This work is reported in Fair and Taylor (1990), and the following discussion is taken from this paper.

The first part of this section discusses the new results using the less expensive method that have been obtained and argues that full information estimation now seems feasible for rational expectations models. In the process of doing this some errors in the earlier work regarding the treatment of models with rational expectations and autoregressive errors are corrected. The second part discusses methods for stochastic simulation of rational expectations models, something that was only briefly touched on in the earlier work.

10.2. The solution method

The notation for the model used here differs somewhat from the notation used in Eq. (1). The lagged values of the endogenous variables are written out explicitly, and x_t is now a vector of only exogenous variables. The model is written as

$$f_i(y_t, y_{t-1}, \ldots, y_{t-p}, E_{t-1}y_t, E_{t-1}y_{t+1}, \ldots, E_{t-1}y_{t+h}, x_t, \alpha_i) = u_{it}, \tag{38}$$

$$u_{it} = \rho_i u_{it-1} + \epsilon_{it}, \quad i = 1, \ldots, n, \tag{39}$$

where y_t is an n-dimensional vector of endogenous variables, x_t is a vector of exogenous variables, E_{t-1} is the conditional expectations operator based on the model and on information through period $t-1$, α_i is a vector of parameters, ρ_i is the serial correlation coefficient for the error term u_{it}, and ϵ_{it} is an error term that may be correlated across equations but not across time. The function f_i may be nonlinear in variables, parameters, and expectations. The following is a brief review of the solution method for this model. In what follows i is always meant to run from 1 through n.

[2]For example, the extended path method has been programmed as part of the TROLL computer package and is routinely used to solve large scale rational expectations models at the IMF, the Federal Reserve, the Canadian Financial Ministry, and other government agencies. It has also been used for simulation studies such as DeLong and Summers (1986) and King (1988). Other solution methods for rational expectations models are summarized in Taylor and Uhlig (1990). These other methods do not yet appear practical for medium size models and up. See also Fisher (1992) for a discussion of the EP method and various extensions and alternatives and for an extensive bibliography of work in this field.

Case 1. $\rho_i = 0$. Consider solving the model for period s. It is assumed that estimates of α_i are available, that current and expected future values of the exogenous variables are available, and that the current and future values of the error terms have been set to their expected values (which will always be taken to be zero here). If the expectations $E_{s-1}y_s$, $E_{s-1}y_{s+1}, \ldots, E_{s-1}y_{s+h}$ were known, (38) could be solved in the usual ways (usually by the Gauss–Seidel technique). The model would be simultaneous, but future predicted values would not affect current predicted values. The EP method iterates over solution *paths*. Values of the expectations through period $s+h+k+h$ are first guessed, where k is a fairly large number relative to h.[3] Given these guesses, the model can be solved for periods s through $s+h+k$ in the usual ways. This solution provides new values for the expectations through period $s+h+k$ – the new expectations values are the solution values. Given these new values, the model can be solved again for periods s through $s+h+k$, which provides new expectations values, and so on. This process stops (if it does) when the solution values on one iteration are within a prescribed tolerance criterion of the solution values on the previous iteration for all periods s through $s+h+k$.

So far the guessed values of the expectations for periods $s+h+k+1$ through $s+h+k+h$ (the h periods beyond the last period solved) have not been changed. If the solution values for periods s through $s+h$ depend in a nontrivial way on these guesses, then overall convergence has not been achieved. To check for this, the entire process above is repeated for k one larger. If increasing k by one has a trivial effect (based on a tolerance criterion) on the solution values for s through $s+h$, then overall convergence has been achieved; otherwise k must continue to be increased until the criterion is met. In practice what is usually done is to experiment to find the value of k that is large enough to make it likely that further increases are unnecessary for any experiment that might be run and then do no further checking using larger values of k.

The expected future values of the exogenous variables (which are needed for the solution) can either be assumed to be the actual values (if available and known by agents) or be projected from an assumed stochastic process. If the expected future values of the exogenous variables are not the actual values, one extra step is needed at the end of the overall solution. In the above process the expected values of the exogenous variables would be used for all the solutions, the expected values of the exogenous variables being chosen ahead of time. This yields values for $E_{s-1}y_s$, $E_{s-1}y_{s+1}, \ldots, E_{s-1}y_{t+h}$. Given these values, (38) is then solved for period s using the *actual* value of x_s, which yields the final solution value $\hat{y}_s$. To the extent that the expected value of x_s differs from the actual value, $E_{s-1}y_s$ will differ from $\hat{y}_s$.

[3]Guessed values are usually taken to be the actual values if the solution is within the period for which data exist. Otherwise, the last observed value of a variable can be used for the future values or the variable can be extrapolated in some simple way. Sometimes information on the steady state solution (if there is one) can be used to help form the guesses.

Two points about this method should be mentioned. First, no general convergence proofs are available. If convergence is a problem, one can sometimes "damp" the solution values to obtain convergence. In practice convergence is usually not a problem. There may, of course, be more than one set of solution values, and so there is no guarantee that the particular set found is unique. If there is more than one set, the set that the method finds may depend on the guesses used for the expectations for the h periods beyond $s + h + k$.

Second, the method relies on the certainty equivalence assumption even though the model is nonlinear. Since expectations of functions are treated as functions of the expectations in future periods in Eq. (38), the solution is only approximate unless f_i is linear. This assumption is like the linear quadratic approximation to rational expectations models that has been proposed, for example, by Kydland and Prescott (1982). Although the certainty equivalence assumption is widely used, including in the engineering literature, it is, of course, not always a good approximation.

Case 2. $\rho_i \neq 0$ and data before $s-1$ available. The existence of serial correlation complicates the problem considerably. The error terms for period $t-1$ ($u_{it-1}, i = 1, \ldots, n$) depend on expectations that were formed at the end of period $t-2$, and so a new viewpoint date is introduced. This case is discussed in Section 2.2 in Fair and Taylor (1983), but an error was made in the treatment of the second viewpoint date. The following method replaces the method in Section 2.2 of this paper.

Consider again solving for period s. If the values of u_{is-1} were known, one could solve the model as above. The only difference is that the value of an error term like u_{is+r-1} would be $\rho_i^r u_{is-1}$ instead of zero. The overall solution method first uses the EP method to solve for period $s-j$, where $j > 0$, based on the assumption that $u_{is-j-1} = 0$. Once the expectations are solved for, (38) is used to solve for u_{is-j}. The actual values of y_{s-j} and x_{s-j} are used for this purpose (although the solution values are used for the expectations) because these are structural errors being estimated, not reduced form errors. Given the values for u_{is-j}, the model is solved for period $s-j+1$ using the EP method, where an error term like u_{is-j+r} is computed as $\rho_i^r u_{is-j}$. Once the expectations are solved for, (38) is used to solve for u_{is-j+1}, which can be used in the solution for period $s-j+2$, and so on through the solution for period s.

The solution for period s is based on the assumption that the error terms for period $s-j-1$ are zero. To see if the solution values for period s are sensitive to this assumption, the entire process is repeated with j increased by 1. If going back one more period has effects on the solution values for period s that are within a prescribed tolerance criterion, then overall convergence has been achieved; otherwise j must continue to be increased. Again, in practice one usually finds a value of j that is large enough to make it likely that further increases are unnecessary for any experiment that might be run and then do no further checking using larger values of j.

It should be noted that once period s is solved for, period $s+1$ can be solved for without going back again. From the solution for period s, the values of u_{is} can

be computed, which can then be used in the solution for period $s + 1$ using the EP method.

Case 3. $\rho_i \neq 0$ and data before period $s - 1$ not available. This case is based on the assumption that $\epsilon_{is-1} = 0$ when solving for period s. This type of an assumption is usually made when estimating multiple equation models with moving average residuals. The solution problem is to find the values of u_{is-1} that are consistent with this assumption. The overall method begins by guessing values for u_{is-2}. Given these values, the model can be solved for period $s - 1$ using the EP method and the fact that $u_{is+r-2} = \rho_i^r u_{is-2}$. From the solution values for the expectations, (38) and (39) can be used to solve for ϵ_{is-1}.[4] If the absolute values of these errors are within a prescribed tolerance criterion, convergence has been achieved. Otherwise, the new guess for u_{is-2} is computed as the old guess plus ϵ_{is-1}/ρ_i. The model is solved again for period $s - 1$ using the new guess and the EP method, and so on until convergence is reached.

At the point of convergence u_{is-1} can be computed as $\rho_i u_{is-2}$, where u_{is-2} is the estimated value on the last iteration (the value consistent with ϵ_{is-1} being within a prescribed tolerance criterion of zero). Given the values of u_{is-1}, one can solve for period s using the EP method, and the solution is finished.

10.3. Computational costs

The easiest way to think about the computational costs of the solution method is to consider how many times the equations of a model must be "passed" through. Let N_1 be the number of passes through the model that it takes to solve the model for one period, given the expectations. N_1 is usually some number less than 10 when the Gauss–Seidel technique is used. The EP method requires solving the model for $h + k + 1$ periods. Let N_2 be the number of iterations it takes to achieve convergence over these periods. Then the total number of passes for convergence is $N_2 N_1 (h+k+1)$. If, say, h is 5, k is 30, N_2 is 15, and N_1 is 5, then the total number of passes needed to solve the model for one period is 11,250, which compares to only 5 when there are no expectations. If k is increased by one to check for overall convergence, the total number of passes is slightly more than doubled, although, as noted above, this check is not always done.

For Case 2 above the number of passes is increased by roughly a factor of j if overall convergence is not checked. Checking for overall convergence slightly more than doubles the number of passes. j is usually a number between 5 and 10. If q is the number of iterations it takes to achieve convergence for Case 3 above, the number

[4]These are again estimates of the structural error terms, not the reduced form error terms. Step (iii) on p. 1176 in Fair and Taylor (1983) is in error in this respect. The errors computed in step (iii) should be the structural error terms.

of passes is increased by a factor of $q + 1$. In practice q seems to be between about 5 and 10. Note for both Cases 2 and 3 that the number of passes is increased relative to the non serial correlation case only for the solution for the first period (period s). If period $s + 1$ is to be solved for, no additional passes are needed over those for the regular case.

10.4. FIML estimation

Assume that the estimation period is 1 through T. The objective function that FIML maximizes (assuming normality) is presented in Eq. (9) above. In the present notation, the ij element of Σ is $(1/T)\sum_{t=1}^{T} \epsilon_{it}\epsilon_{jt}$. Since the expectations have viewpoint date $t - 1$, they are predetermined from the point of view of taking derivatives for the Jacobian, and so no additional problems are involved for the Jacobian in the rational expectations case. In what follows α will be used to denote the vector of all the coefficients in the model. In the serial correlation case α also includes the ρ_i coefficients.

In the standard case computing Σ for a give value of α is fairly inexpensive. One simply solves (38) and (39) for the ϵ_{it} error terms given the data and the value of α. This is only one pass through the model since it is the structural error terms that are being computed. In the rational expectations case, however, computing the error terms requires knowing the values of the expectations, which themselves depend on α. Therefore, to compute Σ for a given value of α one has to solve for the expectations for each of the T periods. If, say, 11,250 passes through the model are needed to solve the model for one period and if T is 100, then 1,125,000 passes are needed for one evaluation of Σ and thus one evaluation of L. In a 25 coefficient problem discussed in Fair and Taylor (1990), the Parke algorithm required 2,817 evaluations of L to converge, which would be over 3 trillion passes if done this way.[5]

It should be clear that the straightforward combination of the EP solution method and FIML estimation procedures is not likely to be computationally feasible for most applications. There is, however, a way of cutting the number of times the model has to be solved over the estimation period to roughly the number of estimated coefficients. The trick is to compute numerical derivatives of the expectations with respect to the parameters and use these derivatives to compute Σ (and thus L) each time the algorithm requires a value of L for a given value of α.

Consider the derivative of $E_{t-1}y_{t+r}$ with respect to the first element of α. One can first solve the model for a given value of α and then solve it again for the first element of α changed by a certain percent, both solutions using the EP method. The computed derivative is then the difference in the two solution values of $E_{t-1}y_{t+r}$ divided by the change in the first element of α. To compute all the derivatives requires $K + 1$

[5]Note that these solutions of the error term ϵ_{it} are only approximations when f_i is nonlinear. Hence, the method gives an approximation of the likelihood function.

solutions of the model over the T number of observations, where K is the dimension of α.[6] One solution is for the base values, and the K solutions are for the K changes in α, one coefficient change per solution. From these $K+1$ solutions, $K \cdot T \cdot (h+1)$ derivatives are computed and stored for each expectations variable, one derivative for each length ahead for each period for each coefficient.[7] Once these derivatives are computed, they can be used in the computation of Σ for a given change in α, and no further solutions of the model are needed. In other words, when the maximization algorithm changes α and wants the corresponding value of L, the derivatives are first used to compute the expectations, which are then used in the computation of Σ. Since one has (from the derivatives) an estimate of how the expectations change when α changes, one does not have to solve the model any more to get the expectations.

Assuming that the solution method in Case 3 above is used for the FIML estimates, derivatives of u_{it-1} with respect to the coefficients are also needed when the errors are serially correlated. These derivatives can also be computed from the $K+1$ solutions, and so no extra solutions are needed in the serial correlation case.

Once the $K+1$ solutions of the model have been done and the maximization algorithm has found what it considers to be the optimum, the model can be solved again for the T periods using the optimal coefficient values and then L computed. This value of L will in general differ from the value of L computed using the derivatives for the same coefficient values, since the derivatives are only approximations. At this point the new solution values (not computed using the derivatives) can be used as new base values and the problem turned over to the maximization algorithm again. This is the second "iteration" of the overall process. Once the maximization algorithm has found the new optimum, new base values can be computed, a new iteration performed, and so on. Convergence is achieved when the coefficient estimates from one iteration to the next are within a prescribed tolerance criterion of each other. This procedure can be modified by recomputing the derivatives at the end of each iteration. This may improve convergence, but it obviously adds considerably to the expense. At a minimum, one might want to recompute the derivatives at the end of overall convergence and then do one more iteration. If the coefficients change substantially on this iteration, then overall convergence has not in fact been achieved.

10.5. Stochastic simulation

For models with rational expectations one must state very carefully what is meant by a stochastic simulation of the model and what stochastic simulation is to be used for.

[6]In the notation presented in Section 2 k rather than K is used to denote the dimension of α. K is used in this section since k has already been used in the description of the EP method.

[7]Derivatives computed this way are "one sided". "Two sided" derivatives would require an extra K solutions, where each coefficient would be both increased and decreased by the given percentage. My experience is that two sided derivatives are generally unnecessary.

In the present case stochastic simulation is *not* used to improve on the accuracy of the solutions of the expected values. The expected values are computed exactly as described above – using the EP method. This way of solving for the expected values can be interpreted as assuming that agents at the beginning of period s form their expectations of the endogenous variables for periods s and beyond by 1) forming expectations of the exogenous variables for periods s and beyond, 2) setting the error terms equal to their expected values (say zero) for periods s and beyond, 3) using the existing set of coefficient estimates of the model, and then 4) solving the model for periods s and beyond. These solution values are the agents' expectations.

For present purposes stochastic simulation begins once the expected values have been solved for. Given the expected values for periods s through $s + h$, stochastic simulation is performed for period s. The problem is now no different from the problem for a standard model because the expectations are predetermined. Assume that the errors are distributed $N(0, \widehat{\Sigma})$, where $\widehat{\Sigma}$ is the FIML estimate of Σ from Section 10.4. From this distribution one can draw a vector of error terms for period s. Given these draws (and the expectations), the model can be solved for period s in the usual ways. This is one repetition. Another repetition can be done using a new draw of the vector of error terms, and so on. The means and variances of the forecast values can be computed using Eqs (30) and (32).

One can also use this approach to analyze the effects of uncertainty in the coefficients by assuming that the coefficients are distributed $N(\hat{\alpha}, \widehat{V}_4)$, where $\hat{\alpha}$ is the FIML estimate of α and $\widehat{V}_4$ is the estimated covariance matrix of $\hat{\alpha}$. In this case each draw also involves the vector of coefficients.

If u_{it} is serially correlated as in (39), then an estimate of u_{is-1} is needed for the solution for period s. This estimate is, however, available from the solution of the model to get the expectations (see Case 2 in Section 10.2), and so no further work is needed. The estimate of u_{is-1} is simply taken as predetermined for all the repetitions, and u_{is} is computed as $\rho_i u_{is-1}$ plus the draw for ϵ_{is}. (Note that the ϵ errors are drawn, not the u errors.)

Stochastic simulation is quite inexpensive if only results for period s are needed because the model only needs to be solved once using the EP method. Once the expectations are obtained, each repetition merely requires solving the model for period s. If, on the other hand, results for more than one period are needed and the simulation is dynamic, the EP method must be used p times for each repetition, where p is the length of the period.

Consider the multi period problem. As above, the expectations with viewpoint date $s - 1$ can be solved for and then a vector of error terms and a vector of coefficients drawn to compute the predicted value of y_{is}. This is the first step.

Now go to period $s + 1$. An agent's expectation of, say, y_{is+2} is different with viewpoint date s than with viewpoint date $s - 1$. In particular, the value of y_{is} is in general different from what the agent at the end of period $s - 1$ expected it to be

(because of the error terms that were drawn for period s).[8] A new set of expectations must thus be computed with viewpoint date s. Agents are assumed to use the original set of coefficients (not the set that was drawn) and to set the values of the error terms for periods $s+1$ and beyond equal to zero. Then given the solution value of y_{is} and the actual value of x_s, agents are assumed to solve the model for their expectations for periods $s+1$ and beyond. This requires a second use of the EP method. Given these expectations, a vector of error terms for period $s+1$ is drawn and the model is solved for period $s+1$. If equation i has a serially correlated error, then u_{is+1} is equal to $\rho_i^2 u_{is-1}$ plus the draw for ϵ_{is+1}. Now go to period $s+2$ and repeat the process, where another use of the EP method is needed to compute the new expectations. The process is repeated through the end of the period of interest. At the end, this is one repetition. The overall process is then repeated for the second repetition, and so on. Note that only one coefficient draw is used per repetition, i.e., per dynamic simulation. After J repetitions one can compute means and variances just as above, where there are now means and variances for each period ahead of the prediction. Also note that agents are always assumed to use the original set of coefficients and to set the current and future error terms to zero. They do not perform stochastic simulation themselves.

Stochastic simulation has also been used to evaluate alternative international monetary systems using the multicountry models in Carloyzi and Taylor (1985) and Taylor (1988). For this work values of ϵ_{it} were drawn, but not values of the coefficients. The vector of coefficients α was taken to be fixed.

It seems that stochastic simulation as defined above is computationally feasible for models with rational expectations. Stochastic simulation is in fact likely to be cheaper than even FIML estimation using the derivatives. If, for example, the FIML estimation period is 100 observations and there are 25 coefficients to estimate, FIML estimation requires that the model be solved 2600 times using the EP method to get the derivatives. For a stochastic simulation of 8 periods and 100 repetitions, on the other hand, the model has to be solved using the EP method only 800 times.

10.6. Conclusion

The results reported in Fair and Taylor (1990) using the methods discussed in this section are encouraging regarding the use of models with rational expectations. FIML estimation seems computationally feasible using the procedure of computing derivatives for the expectations, and stochastic simulation is feasible when done in the manner described above. FIML estimation is particularly important because it takes into account all the nonlinear restrictions implied by the rational expectations hypothesis. It is hoped that the methods discussed in this section will open the way for many more tests of models with rational expectations.

[8] It may also be that the actual value of x_s differs from what the agent expected it to be at the end of $s-1$.

References

Amemiya, T. (1982) 'The two stage least absolute deviations estimators', *Econometrica*, 50:689–711.

Carloyzi, N. and Taylor, J.B. (1985) 'International capital mobility in the coordination of monetary policy rules', in: J. Bandhari, ed., *Exchange rate policy under uncertainty*. Cambridge, MA: MIT Press.

Chow, G.C. (1981) *Econometric analysis by control methods*. New York: Wiley.

DeLong, J.B. and Summers, L.H. (1986) 'Is increased price flexibility stabilizing?', *American Economic Review*, 78:1031–1043.

Fair, R.C. (1974) 'On the solution of optimal control problems as maximization problems', *Annals of Economic and Social Measurement*, 3:135–154.

Fair, R.C. (1984) *Specification, estimation, and analysis of macroeconometric models*. Cambridge, MA: Harvard Univ. Press.

Fair, R.C. (1994) *Testing macroeconometric models*. Cambridge, MA: Harvard Univ. Press.

Fair, R.C. and Parke, W.R. (1993) 'The Fair–Parke program for the estimation and analysis of nonlinear econometric models', mimeo.

Fair, R.C. and Taylor, J.B. (1983) 'Solution and maximum likelihood estimation of dynamic rational expectations models', *Econometrica*, 51:1169–1185.

Fair, R.C. and Taylor, J.B. (1990) 'Full information estimation and stochastic simulation of models with rational expectations', *Journal of Applied Econometrics*, 5:381–392.

Fisher, P. (1992) *Rational expectations in macroeconomic models*. London: Kluwer Academic Publishers.

Hansen, L.P. (1982) 'Large sample properties of generalized method of moments estimators', *Econometrica*, 50:1029–1054.

King, S.R. (1988) 'Is increased price flexibility stabilizing?', *American Economic Review*, 78:267–272.

Kydland, F.E. and Prescott, E.C. (1982) 'Time to build and aggregate fluctuations', *Econometrica*, 50:1342–1370.

McCarthy, M.D. (1972) 'Some notes on the generation of pseudo-structural errors for use in stochastic simulation studies', in: B.G. Hickman, ed., *Econometric models of cyclical behavior*. New York: Columbia Univ. Press, pp. 185–191.

Parke, W.R. (1982) 'An algorithm for FIML and 3SLS estimation of large nonlinear models', *Econometrica*, 50:81–96.

Powell, M.J.D. (1964) 'An efficient method for finding the minimum of a function of several variables without calculating derivatives', *Computer Journal*, 7:155–162.

Taylor, J.B. (1988) 'Policy analysis with a multicountry model', NBER working paper.

Taylor, J.B. and Uhlig, H. (1990) 'Solving nonlinear stochastic growth models: A comparison of alternative solution methods', *Journal of Business and Economic Statistics*, 8:1–21.