SECOND ORDER ACCURATE SOLUTION OF DISCRETE TIME DYNAMIC EQUILIBRIUM MODELS

CHRISTOPHER A. SIMS

I. Introduction

It is now widely understood how to obtain first-order accurate approximations to the solution to a dynamic, stochastic general equilibrium model (DSGE model). Such solutions are fairly easy to construct and useful for a wide variety of purposes. They are likely to be accurate enough to be a basis for fitting the models to data, for example.

However, for some purposes first-order accuracy is not enough. This is true in particular for comparing welfare across policies that do not have first-order effects on the model's deterministic steady state, for example. It is also true for attempts to study asset pricing in the context of DSGE models. It is possible to make separate arguments or assumptions that allow use of first-order approximations in these contexts[1], but the usual reliance on local approximation being generally locally accurate does not apply to these contexts.

It is therefore of some interest to have an algorithm available that will produce second-order accurate approximations to the solutions to DSGE's from a straightforward second-order expansion of the model's equilibrium equations. Matlab code that does this is available at `eco-072399b.princeton.edu/yftp/gensys2/`, where the current version of this paper will also be found.

II. The General Form of the Model

We suppose a model that takes the form

$$\underset{n\times 1}{K}\left(\underset{n\times 1}{w_t}, w_{t-1}, \underset{m\times 1}{\varepsilon_t}\right) + \Pi \underset{p\times 1}{\eta_t} = 0\,, \tag{1}$$

Date: December 30, 2000: Preliminary version.

Discussions with Ernst Schaumburg, who has work of his own on using second order expansions to study dynamics, sharpened my understanding of the problem.

[1]Woodford (1999) is an example of making the necessary auxiliary assumptions explicit.

where $E_t\eta_{t+1} = 0$ and $E_t\varepsilon_{t+1} = 0$. The disturbances ε_t are exogenously given, while η_t is determined as a function of ε when the model is solved.[2]

We assume that the solution will imply that w_t remains always on a stable manifold, defined by $H(w_t) = 0$ and satisfying

$$\{ \underset{n_u \times n}{H}(w_t) = 0 \quad \text{and} \quad K(w_{t+1}, w_t, \varepsilon_{t+1}) + \eta_{t+1} = 0\} \;\Rightarrow\; H(w_{t+1}) = 0\,, \tag{2}$$

when η_{t+1} depends on ε_{t+1} and w_t appropriately. We consider expansion of the system about a point $\bar{w}$ in the stable manifold, i.e. satisfying $H(\bar{w}) = 0$. The "stability" of this manifold means either that $w_t \to \bar{w}$ along it, or that in the neighborhood of $\bar{w}$ w_t diverges from $\bar{w}$ at a rate no faster than some known bound. It is this condition that allows us to solve for η.

We also assume that the nonlinear system (1) is formulated in such a way that its first-order expansion characterizes the first-order behavior of the solution. That is, we assume that solving the first-order expansion of (1) about $\bar{w}$

$$K_1 dw_t = -K_2 dw_{t-1} - K_3\varepsilon_t + \Pi\eta_t \tag{3}$$

as a linear system results in a unique stable saddle path in the neighborhood of the deterministic steady state. If so, this saddle path characterizes the first-order behavior of the system. We assume further that $H'(\bar{w})$ is of full row rank, so that the first-order character of the saddle path is determined by the first-order expansion of H.[3]

The system (1) has the second-order Taylor expansion about $\bar{w}$

$$\begin{aligned} K_{1ij}dw_{jt} = &-K_{2ij}dw_{j,t-1} - K_{3ij}\varepsilon_{jt} + \Pi_{ij}\eta_{jt} \\ &- \tfrac{1}{2}(K_{11ijk}dw_{jt}dw_{kt} + 2K_{12ijk}dw_{jt}dw_{k,t-1} + 2K_{13ijk}dw_{jt}\varepsilon_{kt} \\ &\qquad + K_{22ijk}dw_{j,t-1}dw_{k,t-1} + 2K_{23ijk}dw_{j,t-1}\varepsilon_{kt} + K_{33ijk}\varepsilon_{jt}\varepsilon_{kt})\,, \end{aligned} \tag{4}$$

where we have resorted to tensor notation. That is, we are using the notation that

$$A_{ijk}B_{mnjq} = C_{ikmnq} \quad \Leftrightarrow \quad c_{ikmnq} = \sum_j a_{ijk}b_{mnjq}\,. \tag{5}$$

where a, b, c in this expression refer to individual elements of multidimensional arrays, while A, B, C refer to the arrays themselves. As special case, for example, ordinary matrix multiplication is $AB = A_{ij}B_{jk}$ and the usual matrix expression $A'BA$ becomes $A_{ji}B_{jk}A_{km}$. Note that we are distinguishing the array K_{mij} of first derivatives from

[2]This form is more general than it might seem. See Sims (to appear) for examples showing how models with explicit expectations operators, including lagged expectations, can be cast into this form.

[3]This assumption on H is not restrictive. So long as there is a continuous, differentiable saddle manifold, we will be able to choose an H satisfying this condition.

the array K_{mnijk} of second derivatives only by the number of indexing subscripts the two arrays have.

III. Solution Method

The solution we are looking for can be written in the form

$$w_t = F^*(w_{t-1}, \varepsilon_t)\,. \tag{6}$$

Because we know the saddle manifold characterized by H exists and that $D_w H(\bar{w})$ has full row rank n_u, we can use H to express n_u linear combinations of w's in terms of the remaining $n_s = n - n_u$. Let the n_s linear combinations of w's chosen as "explanatory" variables in this relation be

$$y_t = \underset{n_s \times n}{\Phi}\, w_t\,. \tag{7}$$

Then the solution (6) can be expressed equivalently, in a neighborhood of $\bar{w}$, as

$$y_t = \Phi F^*(w_{t-1}, \varepsilon_t) = F(y_{t-1}, \varepsilon_t) \tag{8}$$

$$\underset{n_u \times 1}{x_t} = h(y_t)\,, \tag{9}$$

where (9) is just the solved version of the $H = 0$ equation that characterizes the stable manifold. Here of course x, like y, is a linear combination of w's.

The solution method for linear rational expectations systems described in Sims (to appear) begins by applying linear transformations to the list of variables and to the equation system to produce an upper triangular block recursive system. In the transformed system, the unstable roots of the system are all associated with the lower right block, η_t does not appear in the upper set of equations in the system,[4] and the upper part of the equation system is normalized to have the identity as the coefficient matrix on current values of the upper part of the transformed variable matrix. In other words, we can use the methods described in the earlier paper[5] to transform (4) to

$$\begin{aligned} dy_{it} =& G_{1ij}dx_{jt} + G_{2ij}dv_{t-1} + G_{3ij}\varepsilon_{jt} + \tfrac{1}{2}\Big(G_{11ijk}dv_{jt}dv_{kt} + 2G_{12ijk}dv_{jt}dv_{k,t-1} \\ &+ 2G_{13ijk}dv_{jt}\varepsilon_{kt} + G_{22ijk}dv_{j,t-1}dv_{k,t-1} + 2G_{23ijk}dv_{j,t-1}\varepsilon_{kt} + G_{33ijk}\varepsilon_{jt}\varepsilon_{kt}\Big) \end{aligned} \tag{10}$$

$$\begin{aligned} J_{1ij}dx_{jt} =& J_{2ij}dx_{j,t-1} + J_{3ij}\varepsilon_{jt} + \Pi^*\eta_t + \tfrac{1}{2}\Big(J_{11ijk}dv_{jt}dv_{kt} + 2J_{12ijk}dv_{jt}dv_{k,t-1} \\ &+ 2J_{13ijk}dv_{jt}\varepsilon_{kt} + J_{22ijk}dv_{j,t-1}dv_{k,t-1} + 2J_{23ijk}dv_{j,t-1}\varepsilon_{kt} + J_{33ijk}\varepsilon_{jt}\varepsilon_{kt}\Big)\,, \end{aligned} \tag{11}$$

[4] It may not be possible in fact to eliminate η_t from the upper part of the system. When it is not, the solution is not unique. The programs signal the non-uniqueness and deliver one solution, in which the η's are set to zero in the upper block of this system.

[5] and implemented in the Matlab function `gensys.m`

where $v_t = [y_t; x_t]$, i.e. the y and x vectors stacked up.

Now the y and x introduced above may seem to have no connection to the y and x in terms of which we wrote the solution (8)-(9). But that solution has second-order expansion

$$\begin{aligned} dy_{it} &= F_{1ij} dy_{j,t-1} + F_{2ij}\varepsilon_{jt} + F_{3i}\sigma^2 \\ &\quad + \tfrac{1}{2}\Big(F_{11ijk} dy_{j,t-1} dy_{k,t-1} + 2F_{12ijk} dy_{j,t-1}\varepsilon_{kt} + F_{33ijk}\varepsilon_{jt}\varepsilon_{kt}\Big) \end{aligned} \tag{12}$$

$$dx_{it} = \tfrac{1}{2} M_{11ijk} dy_{jt} dy_{kt} + M_2\sigma^2 \,. \tag{13}$$

Of course if x were chosen as an arbitrary linear combination of w's, there would in general be a first-order term in dy_t on the right-hand side of (13). However, we can always move such terms to the left-hand side and then redefine x to include them. We will now proceed to show that the dy and dx in (12)-(13) are indeed those in (10)-(11), and that indeed we can construct the coefficient matrices in the former from knowledge of the coefficient matrices in the latter.

The terms F_3 and M_2 in (12)-(13) deserve discussion. Because of the implicit appearance of expectations operators in our system (via the $E_t\eta_{t+1} = 0$ condition), the solution depends not only on an actual sequence $\{\varepsilon_t,\, t = 0, \ldots, \infty\}$ of disturbances, but also on probability-weighted integrals over possible, but unrealized, values of disturbances. When we assert 1st or 2nd order accuracy, we are making not simply the usual claim that accuracy increases as all elements of the observed ε_t sequence shrink toward zero, but a claim about what happens as the actual ε_t sequence shrinks to zero and the probability distribution over possible ε's both shrink at the same rate. The integrals over the distribution of ε that enter the solution produce second-order effects on constant terms, which is why F_3 and M_2 are needed.[6]

Observe that dx_t in (10)-(11) must be zero to first order, because otherwise there would be an explosive component in the first order part of the solution, contradicting the stability assumption. Therefore, F_1 is just the square matrix consisting of the first columns of G_2 in (10). Clearly also $F_2 = G_3$. Therefore we have a complete first-order solution for dy and dx in hand:

$$dy_t \doteq F_1 dy_{t-1} + F_2\varepsilon_t \tag{14}$$

$$dx_t \doteq 0 \,. \tag{15}$$

We find the second order terms in the following steps. First shift (11) forward in time by one (so that the left-hand side is dx_{t+1}) and substitute the right-hand side of (13), shifted forward in time by 1, for the dx_{t+1} on the left. Then substitute

[6] If we proceeded to higher-order expansions, we would find e.g. third order terms in which the coefficients of linear terms vary with the variance of ε, and so on.

the right-hand-side of (14), shifted forward by 1, for all occurrences of dy_{t+1} in the resulting system. Finally apply the E_t operator to the result. In doing this, we are dropping all the second order terms in the solution for dy and dx when these terms themselves occur in second order terms. This makes sense because cross products involving terms higher than first order are third order or higher, and thus do not contribute to the second order expansion. Note that this means that, since dx is zero to first order, in (10)-(11) all the second-order terms in dv can be written in terms of dy alone. We will abuse notation by using the same G and J labels for the smaller second-order coefficient matrices that apply to dy alone that we use in (10)-(11) for the second order terms involving the full v vector. In this way we arrive at

$$\begin{aligned} J_{1ij}\Big(\tfrac{1}{2}\big(M_{11jk\ell}F_{1kr}dy_{rt}F_{1\ell s}dy_{st} + M_{11jk\ell}F_{2kr}F_{2\ell s}\Omega_{rs}\sigma^2\big) + M_{2j}\sigma^2\Big) \\ = J_{2ij}\Big(\tfrac{1}{2}M_{11jk\ell}dy_{kt}dy_{\ell t} + M_{2j}\sigma^2\Big) + \tfrac{1}{2}\Big(J_{11ijk}\big(F_{1jr}F_{1ks}dy_{rt}dy_{st} \\ + F_{2jr}F_{2ks}\Omega_{rs}\sigma^2\big) + 2J_{12ijk}F_{1jr}dy_{rt}dy_{kt} + 2J_{13ijk}F_{2jr}\Omega_{rk}\sigma^2 \\ + J_{22ijk}dy_{jt}dy_{kt} + J_{33ijk}\Omega_{jk}\sigma^2\Big)\,, \end{aligned} \tag{16}$$

Where we have set $\operatorname{Var}\varepsilon_t = \sigma^2\Omega$.

For this equation to hold for all dy and σ^2 values, we must match coefficients on common terms. Therefore, looking at the $dy_t \cdot dy_t$ terms, we conclude that

$$J_{1ij}M_{11jkt}F_{1kr}F_{1\ell s} = J_{2ij}M_{11jrs} + J_{11ijk}F_{1jr}F_{1ks} + 2J_{12ijs}F_{1jr} + J_{22ijk}\,. \tag{17}$$

This is a linear equation, and every element of it is known except for $M_{11\cdots}$. The transformations that produced the block-recursive system with ordered roots guarantee that $J_{2\cdot\cdot}$, an ordinary 2×2 matrix, has all its eigenvalues above the critical stability value. It is therefore invertible, and we can multiply (17) through on the left by J_2^{-1}, to get a system in the form

$$AM^*F_1 \otimes F_1 = M^* + B\,. \tag{18}$$

In this equation, M^* is the ordinary $n_s \times n_s^2$ matrix obtained by stacking up the second and third dimensions of $M_{11\cdots}$, $A = J_2 \backslash J_1$, and B is everything else in the equation that doesn't depend on M^*. If the dividing line we have specified between stable and unstable roots is $1+\delta$, then our construction of the block-recursive system has guaranteed that $J_2 \backslash J_1$ has all its eigenvalues $\leq 1/(1+\delta)$, while at the same time it is a condition on the solution that all the eigenvalues of F_1 be $< 1+\delta$. To guarantee that a second-order solution exists, we require that the product of the largest eigenvalue of $F_1 \otimes F_1$, which is the square of the largest eigenvalue of F_1, be less than the inverse of the largest eigenvalue of $A = J_2 \backslash J_1$. If $\delta = 0$ this condition is automatically satisfied. Otherwise, there is an extra condition that was not required for finding a solution to

the linear system: the smallest unstable root must exceed the square of the largest stable root.

Assuming this condition holds, (18) has the form of a discrete Lyapunov or Sylvester equation that is guaranteed to have a solution. Because of the special structure of $F1 \otimes F1$, it would be very inefficient to solve this system with standard packages (like Matlab's `lyap.m`), but it is easy to exploit the special structure with a doubling algorithm to obtain an efficient solution for M^*.

With $M_{11\cdots}$ in hand, it is easy to see from (16) that we can obtain a solution for M_2 by matching coefficients on σ^2. The only slightly demanding calculation is a required inversion of $J_2 - J_1$. But since $J_2 \backslash J_1$ has all its eigenvalues less than one, this $J_2 - J_1$ is guaranteed to be nonsingular.

The next step is to use (13) to substitute for the first-order term in dx_t on the right of (10) and (14)-(15) to substitute for all occurrences of dy_t and dx_t in second-order terms on the right in the resulting equation. This produces an equation with dy_t on the left, and first and second-order terms in dy_{t-1} and ε_t and terms in σ^2 on the right. With M_{11} and M_2 in hand, it turns out that it is only a matter of bookkeeping to read off the values of F_{12}, F_{22}, and F_3 by matching them to the collected coefficients in this equation.[7]

IV. Difficulties with Models of Risk Sharing via Asset Trading

The assumption that K's first-order expansion determines the behavior of the system about steady state is restrictive. In models with just enough assets to provide complete markets the quantities of assets held by agents are determinate for every specification with randomness, but all assets become equivalent when stochastic shocks entirely disappear. The K for such a model, when generated in the straightforward way from constraints and first-order conditions, has both K_1 and K_2 singular, with the same left eigenvectors corresponding to zero eigenvalues in both matrices. This means that (3), considered as a linear system, is incomplete, and can provide no determinate solution, even though the nonlinear system does in fact have a determinate solution.

For complete-market models, this situation is a minor inconvenience. One can usually impose a common market discount factor equal to ratios of marginal utilities or productivities across all agents and, without encountering singular-K problems,

[7] I realize it might be useful to display the expressions for these remaining coefficients explicitly, but it really is straightforward, albeit tedious, and I'm trying to get this note out quickly so people can understand what the computer code, already posted, is doing. For these final stages of the computation, it is not so clear that the tensor algebra expressions are much easier to grasp than the computer code itself, in any case.

solve the resulting model for real allocations. A second order accurate solution for these allocations (but not a solution that is only first-order accurate) can then be used to price any set of assets one might consider. If the asset list is complete, but not redundant, asset shares can be calculated by requiring that they implement the already computed complete markets real allocation.

It is worth noting that using the first-order solution to price assets has no firm justification. Asset prices of course depend on covariances of a market discount factor with asset yields, and covariances are second-order quantities. One can simulate a first-order solution to the stochastic model and obtain first-order accurate results for both the market discount factor and the asset yield. However, second-order effects on the market discount factor and the asset yield will also enter the pricing equation, and will in general be the same order of magnitude as the expected product of first-order effects. There is no guarantee, therefore, that using the first-order solution to price assets even ranks assets by yield properly, whether shocks are small or large. It may be possible to show in particular cases that the second order terms that are being ignored when pricing assets off a first-order accurate equilibrium solution are small relative to the second-order effects that are being accounted for. But this requires a separate argument. The usual notion that a local approximation to the solution will be accurate at least in some small neighborhood of the point around which the expansion is taken does not apply here.

For incomplete markets models with no endogenous borrowing — for example models in which agents trade only real capital and government-supplied bonds — the usual K functions also create no problems. There are serious difficulties, though, in considering models in which agents borrow and lend to share risk with an incomplete menu of assets. In such models the natural K is incomplete, and the shortcut of directly imposing equalized ratios of marginal productivities and utilities across dates is not available. There may be a general approach to deriving a K function for such models usable in the algorithm described here, but straightforward use of constraints and first-order conditions will not work.

References

Sims, C. A. (to appear): "Solving Linear Rational Expectations Models," *Computational Economics*.

Woodford, M. (1999): "Inflation Stabilization and Welfare," Discussion paper, Princeton University, http://www.princeton.edu/∼woodford/chapter6.pdf.

Department of Economics, Princeton University
E-mail address: sims@princeton.edu