

HC
103
D57
1988
no.5

The research program of the Center for Economic Studies produces a wide range of theoretical and empirical economic analyses which serve to improve the statistical programs of the U.S. Bureau of the Census. Many of these analyses take the form of research papers. The purpose of the Discussion Papers is to circulate intermediate and final results of this research among interested readers within and outside the Census Bureau. The opinions and conclusions expressed in the papers are those of the authors and do not necessarily represent those of the U.S. Bureau of the Census. All papers are screened to ensure that they do not disclose confidential information. Persons who wish to obtain copies of papers, submit comments about the papers, or obtain general information about the series should contact Peter Zadrozny, Editor, Discussion Papers, Center for Economic Studies, Room 3442, FOB 3, U.S. Bureau of the Census, Washington, DC 20233 (301-763-2490).

**ANALYTIC DERIVATIVES FOR ESTIMATION
OF LINEAR DYNAMIC MODELS**

by

Peter A. Zadrozny

CES 88-5 November 1988

3002378

**BUREAU OF THE CENSUS
LIBRARY**

ABSTRACT

This paper develops two algorithms. Algorithm 1 computes the exact, Gaussian, log-likelihood function, its exact, gradient vector, and an asymptotic approximation of its Hessian matrix, for discrete-time, linear, dynamic models in state-space form. Algorithm 2, derived from algorithm 1, computes the exact, sample, information matrix of this likelihood function. The computed quantities are analytic (not numerical approximations) and should, therefore, be useful for reliably, quickly, and accurately: (i) checking local identifiability of parameters by checking the rank of the information matrix; (ii) using the gradient vector and Hessian matrix to compute maximum likelihood estimates of parameters with Newton methods; and, (iii) computing asymptotic covariances (Cramer-Rao bounds) of the parameter estimates with the Hessian or the information matrix. The principal contribution of the paper is algorithm 2, which extends to multivariate models the univariate results of Porat and Friedlander (1986). By relying on the Kalman filter instead of the Levinson-Durbin filter used by Porat and Friedlander, algorithms 1 and 2 can automatically handle any pattern of missing or linearly aggregated data. Although algorithm 1 is well known, it is treated in detail in order to make the paper self contained.

1. Introduction.

This paper develops two algorithms. Algorithm 1 computes the exact, Gaussian, log-likelihood function, its exact, gradient vector, and an asymptotic approximation of its Hessian matrix, for discrete-time, linear, dynamic models in state-space form. Algorithm 2, derived from algorithm 1, computes the exact, sample, information matrix of this likelihood function. The computed quantities are analytic (not numerical approximations) and should, therefore, be useful for reliably, quickly, and accurately: (i) checking local identifiability of parameters by checking the rank of the information matrix ([10, pp. 1071-1073]); (ii) using the gradient vector and the Hessian matrix to compute maximum likelihood estimates of parameters with Newton methods ([27, pp. 442-450]); and, (iii) computing asymptotic covariances (Cramer-Rao bounds) of the parameter estimates with the Hessian or the information matrix ([40, pp. 68-86]). The principle contribution of the paper is algorithm 2, which extends to multivariate models the univariate results of Porat and Friedlander [37]. By relying on the Kalman filter instead of the Levinson-Durbin filter used by Porat and Friedlander, algorithms 1 and 2 can automatically handle any pattern of missing or linearly aggregated data. Although algorithm 1 is well known, it is treated in detail in order to make the paper self contained.

Box and Jenkins [11], Reinsel [38], and Zadrozny [46] stated recursive formulas for computing the gradient of the approximate, conditional, Gaussian, log-likelihood function of univariate and multivariate, ARMA (autoregressive moving-average)

and ARMAX (ARMA with exogenous variables) models. Akaike [1] stated formulas, requiring inverse Fourier transformation for their final resolution, for computing the gradient of a spectral approximation of the Gaussian, log-likelihood function of a multivariate ARMA model. Kashyap [26] described a Lagrange-multiplier method for computing the gradient of an asymptotic approximation of this log-likelihood when the model is multivariate ARMAX. Wilson and Kumar [45] essentially restated Kashyap's method for a state-space formulation of a model.

Recently, there has been a shift toward using exact likelihood functions. This has been prompted by falling costs of computing and the recognition that an approximate likelihood may lead to seriously biased parameter estimates. For example, Hillmer and Tiao [23] showed this to be the case for an ARMA model with seasonal MA roots close to the unit circle. In the same vein, the accuracy of Akaike's [1] approximation depends on the AR and MA roots being sufficiently outside the unit circle.

Ansley and Kohn [6] and Melard [33] stated algorithms, involving the Chandrasekhar form of the Kalman filter, for computing the gradient of the exact, Gaussian log-likelihood of a univariate, ARMA model. The Chandrasekhar form takes advantage of the observation vector being smaller than the state vector ([35]), but has the disadvantage of not being able to handle missing (or irregularly observed) data. In a related article, Ansley and Kohn [5] use the standard form of the Kalman filter, which can handle missing data. The likelihood and gradient parts of the present algorithm 1 are essentially identical to the

algorithm stated by Harvey and McKenzie [20]. More recently, Tuan [41] developed a quick gradient algorithm for univariate, ARMA models. His algorithm uses backward and forward representations of the model and produces, as an automatic by-product, a Lagrange-multiplier statistic for testing the goodness of fit of the model.

Akaike [1], Reinsel [38], and Zadrozny [46] also stated formulas for computing approximations of the Hessian which are asymptotically equivalent to each other, and to the present one. This equivalence and equation (5.7) below indicate that these Hessian matrices are positive semi-definite by construction and are expected to be positive definite when the parameters are locally identifiable. (We are concerned with minimizing -2 times the log-likelihood, so that the relevant Hessian should be positive definite, not negative definite.) A Hessian is useful for computing maximum likelihood estimates only if it is positive definite ([27, pp. 442-450]). The exact Hessian is generally positive definite only in a neighborhood of a local minimum and must, therefore, generally be modified to make it positive definite. For this reason, we only consider the approximate Hessian. It is straightforward to extend algorithm 1 to one which computes the exact Hessian, but this involves the additional disadvantage of a much greater computational burden. Algorithm 1 requires very little extra work to compute the approximate Hessian, beyond the work required to compute the log-likelihood and its gradient. In their articles [5, 6], Ansley and Kohn also stated algorithms for computing the exact, Hessian matrix, but

did not show how to modify it, if necessary, to make it positive definite. Algorithm 1 also bears some resemblance to the gradient and Hessian algorithm developed by Berndt et al. [9], which has been frequently cited in the econometrics literature.

There appears to be much less literature on the computation of information matrices in general, linear, dynamic models. Box and Jenkins [11, pp. 240-246] stated closed-form expressions for asymptotic information matrices in some special univariate, ARMA models. Zadrozny [46] stated infinite-series representations of asymptotic information matrices in multivariate, ARMAX models. Apparently only Porat and Friedlander [37] have developed results similar to present ones for computing the exact, sample, information matrix. They describe an analogous method, based on Levinson-Durbin filtering, for computing the sample, information matrix in univariate, stationary, linear, dynamic models with exogenous variables. As an example, they work out details for the univariate, ARMA model. Their results do not immediately extend to multivariate models. The present paper extends their work to multivariate models by replacing Levinson-Durbin filtering with Kalman filtering. In both algorithm 2 and in Porat and Friedlander's algorithm, the asymptotic, information matrix is readily obtained by iterating the sample, information matrix to convergence.

In section 2, we introduce the state-space form of a linear dynamic model. In section 3, we describe how to compute with the Kalman filter the exact, Gaussian log-likelihood for a model in state-space form. In sections 4 and 5, we extend the likelihood

algorithm to additionally compute the exact gradient and the approximate Hessian. At the ends of these sections, we also state chain rules which extend these results to models with differentiable equality restrictions on parameters. In section 6, we extend the results of sections 3 to 5 to algorithm 2, which computes the sample information matrix. Section 7 ends the paper with some remarks, including remarks about the possibilities of using so-called square-root versions of the Kalman filter.

2. State-Space Form of a Linear Dynamic Model.

We start this section by describing in general terms the state-space form of a time-varying, linear, dynamic model and end it by illustrating how the commonly used, time-invariant, ARMAX(p,q,r,) model (autoregressive, moving-average model with exogenous variables) can be put into this form. Throughout this section and the remainder of the paper, we shall follow conventional assumptions.

A state-space representation has three essential parts: (i) a state vector which comprises the relevant information for forecasting the process in question; (ii) a law of motion which tells how the state vector evolves over time; and, (iii) an observation (or measurement) equation which tells how observations of the process are made in terms of the state vector.

Let $u(t)$ be an $n \times 1$ vector process of interest which is generated for time periods $t = 1, \dots, N$. Let $x(t)$ be an $s \times 1$, state vector of a particular, state-space representation of the

generating process of $u(t)$. A process which can be represented in state-space form will have infinitely many state-space representations. The state law of motion of a process which is linear in variables has the form

$$(2.1) \quad x(t) = F(\phi, t)x(t-1) + G(\phi, t)e(t) + H(\phi, t)z(t),$$

where $e(t)$ is an $n \times 1$, unobservable, disturbance vector; $z(t)$ is an $h \times 1$, observable, nonstochastic vector of exogenous variables; and, the $s \times s$, $s \times n$, and $s \times h$ system matrices, $F(\phi, t)$, $G(\phi, t)$, and $H(\phi, t)$, are known, nonstochastic functions of a $p \times 1$ vector of underlying, structural parameters, ϕ , and of time, t . For simplicity, we shall suppress the ϕ and t arguments whenever they are not explicitly needed.

The disturbance vector, $e(t)$, is assumed to be serially uncorrelated and to be uncorrelated with all past values of $x(t)$. Because the mean of $u(t)$ can be accounted for with constant terms (setting some element of z to be identically equal to one), without loss of generality, $e(t)$ is assumed to have a zero mean. Finally, $x(1)$, $e(t)$, and, hence, $u(t)$, are assumed to have Gaussian (or normal) probability distributions. These distributional assumptions are denoted by $x(1) \sim N[\mu_x(\phi, 1), \Sigma_x(\phi, 1)]$ and $e(t) \sim \text{NIID}[0, \Sigma_e(\phi, t)]$.

The observation equation is constructed in two steps. Let $w(t)$ denote the $m \times 1$ vector of potential observations on $u(t)$, where $m \leq n$. By "potential" we mean that $w(t)$ is not yet adjusted for the fact some or all of its elements may be unobserved

(missing) in period t . The relation between $w(t)$ and $x(t)$ in a linear model is

$$(2.2) \quad w(t) = \Delta(\phi, t)x(t) + \zeta(t).$$

The $m \times s$ matrix $\Delta(\phi, t)$ does three things: (i) it picks out elements of $x(t)$ which correspond to $u(t)$; (ii) it forms linear combinations of elements of $x(t)$ to account for some elements of $u(t)$ being observed as linear (cross-sectional or temporal) aggregates; and, (iii) it represents the part of the model for $u(t)$ which cannot be captured by (2.1).

The $m \times 1$ vector $\zeta(t)$ is an optional vector of observation errors. Following the usual practice, we assume that $\zeta(t)$ is serially uncorrelated and that it is uncorrelated with all values of the state vector and of its disturbances. In other words, $\zeta(t) \sim \text{NIID}[0, \Sigma_{\zeta}(\phi, t)]$ and $E\zeta(\tau)e(t)^T = 0$ and $E\zeta(\tau)x(1)^T = 0$, for all τ , $t \geq 1$ (superscript T denotes transposition). In principle, Σ_{ζ} can be time varying even when the model is time invariant, although, in practice, Σ_{ζ} is usually taken to be time invariant, even when the model is assumed to be time varying. When Σ_{ζ} depends on parameters to be estimated, then, it is convenient to include these in ϕ , even though observation error covariances are not usually thought of as structural quantities.

When $\Delta(\phi, t)$ does not represent a part of the model, then, it does not depend on (ϕ, t) and is simply a selection matrix of 0's and 1's. This usual case is illustrated below with the ARMAX model. The time-varying, regression model ([25, pp. 391-397])

illustrates the case in which temporal variation in Δ represents a part of the model. Linear rational expectations models in which economic agents explicitly solve linear-quadratic dynamic optimization problems ([18], [19]) provide an example in which the dependence of Δ on ϕ represents a part of the model. Suppose for the moment that $z(t)$ in (2.1) is not an exogenous vector but is a control vector which an agent in such a model sets optimally. Then, the agent's optimal decision rule is of the form $z(t) = \Delta_1(\phi)x(t-1)$, where Δ_1 , depends on ϕ through the solution of a Riccati equation ([48]). Suppose further that $x(t)$ partitions as $x(t) = [x_1(t)^T, x_2(t)^T]^T$, where $x_1(t)$ is observable (by us as well as by the agents in the model) and $x_2(t)$ is unobservable (at least by us), and that $w(t) = [z(t+1)^T, x_1(t)^T]^T + [\zeta_1(t)^T, \zeta_2(t)^T]^T$. Then, $\Delta(\phi) = [\Delta_1(\phi)^T, \Delta_2^T]^T$, where $\Delta_2 = [I, 0]$ is the selection matrix which picks $x_1(t)$ out of $x(t)$. Examples in which Δ accounts for cross-sectional and temporal aggregation are, respectively, given by Ansley and Kohn [3] and Zadrozny [47].

To continue, let $y(t)$ denote the $m(t) \times 1$ vector of values of w for period t which are actually observed, where $m(t) \leq m$. Therefore, we have $y(t) = \Lambda(t)w(t)$, where $\Lambda(t)$ is the $m(t) \times m$ selection matrix which picks out the observed elements of $w(t)$. A frequently occurring example of missing data is the case in which different variables in a multivariate model are observed at different frequencies ([47], [49]). Upon combining (2.2) and $y(t) = \Lambda(t)w(t)$, we get the observation equation,

$$(2.3) \quad y(t) = D(\phi, t)x(t) + v(t),$$

where $D(\phi, t) = \Lambda(t)\Delta(\phi, t)$ and $v(t) = \Lambda(t)\zeta(t)$. Error $v(t)$ inherits $\zeta(t)$'s properties: $v(t) \sim \text{NIID}[0, \Sigma_v(\phi, t)]$, where $\Sigma_v(\phi, t) = \Lambda(t)\Sigma_\zeta(\phi, t)\Lambda(t)^T$, and $\text{Ev}(\tau)e(t)^T = 0$, for all τ and t . This approach for handling missing data originates with Jones [24] and was extended to multivariate cases by Ansley and Kohn [3].

In the case of the exogenous variables, z , we shall assume that there are no missing values. In the event that some values of z are missing from the sample, then, one can either include z in the model and treat it symmetrically with u , or one can interpolate the missing values of z . The latter possibility is discussed in Zadrozny [49], in the context of continuous-time models.

Let the coefficients of the state-space representation, (2.1) and (2.3), be collected in the vector $\theta = [\text{vec}(F)^T, \text{vec}(G)^T, \text{vec}(H)^T, \text{vec}(\Sigma_e)^T, \text{vec}(D)^T, \text{vec}(\Sigma_v)^T]^T$, where $\text{vec}(\cdot)$ vectorizes a matrix columnwise by putting column 2 below column 1, etc. A particular model is, therefore, characterized by a mapping, $\theta = \Psi(\phi, t)$, from (ϕ, t) to θ . We assume that the admissible values of ϕ lie in an open set, so that any equality restrictions are incorporated in Ψ , and that Ψ is differentiable at least once with respect to ϕ .

The estimation problem being considered here is the problem of estimating ϕ by maximizing with respect to ϕ the Gaussian likelihood of the observations on $y(t)$. Under general assumptions

on the model, in particular, on the restriction mapping Ψ , the estimates will be consistent, asymptotically efficient, and will have a known (usually Gaussian) asymptotic distribution. Within limits, these properties are preserved when the true, data generating process is not Gaussian, but the Gaussian likelihood is still being maximized ([17] and references therein). We shall not directly be concerned with assumptions which ensure these properties; rather, we shall only state assumptions on parameters, implicitly in terms of θ , which are sufficient to ensure that the computational formulas can be implemented.

It is computationally convenient to set up Ψ so that admissible values of ϕ lie in a Euclidian space, i.e., are unconstrained. When this is done, the algorithm for maximizing the likelihood does not have to worry about going off, hitting, or crossing a boundary which defines a restriction on ϕ . Let α denote a $k \times 1$ vector of initially specified, structural parameters which the model specification maps into θ , as $\theta = \Gamma(\alpha, t)$. The mapping Γ is supposed to account for restrictions on θ in terms of α , including constant values of elements of θ . However, in an initial specification, α is also likely to be subject to equality restrictions ($\beta_i(\alpha_1, \dots, \alpha_k) = 0$) and strict inequality restrictions ($\beta_i(\alpha_1, \dots, \alpha_k) < 0$) which are not incorporated in Γ . Such restrictions may often be incorporated into a mapping from structural parameters to θ , by reparameterizing α to ϕ with an elementary, smooth, monotonic transformation. For example, suppose that α is a scalar which is restricted by $c_1 < \alpha < c_2$. Then, the reparameterization $\alpha = \Pi(\phi)$

$= (c_1 + c_2 e^\phi)/(1 + e^\phi)$, for $-\infty < \phi < +\infty$, imposes $c_1 < \alpha < c_2$. Thus, in this case and in general, Ψ is the composite mapping $\Psi(\phi, t) = \Gamma(\Pi(\phi), t)$. Nonstrict, inequality restrictions ($\beta_i(\alpha_1, \dots, \alpha_k) \leq 0$) cannot be handled in this way; we shall not further be concerned with such restrictions, which involve further, substantial difficulties.

We now illustrate one way of putting the time-invariant, ARMAX(p,q,r) model into state-space form. Let $u(t)$ be an $n \times 1$ vector which is generated by the ARMAX(p,q,r) process

$$\begin{aligned}
 (2.4) \quad u(t) &= A_1 u(t-1) + \dots + A_p u(t-p) \\
 &\quad + B_0 e(t) + \dots + B_q e(t-q) \\
 &\quad + C_0 z(t) + \dots + C_r z(t-r),
 \end{aligned}$$

where $e(t)$ is $n \times 1$ and \sim NIID $[0, \Sigma_e]$. (There is some redundancy (identification problem) involving B_0 and Σ_e , which can be resolved, e.g., by setting $B_0 = I_n$ or $\Sigma_e = I_n$, the $n \times n$ identity matrix; which normalizations are feasible or convenient depends on the prior restrictions in terms of ϕ .) The model is time invariant when the system matrices, A_i , B_i , C_i , and Σ_e , are time invariant.

Following Ansley and Kohn [3], let $x(t) = [x_1(t)^T, \dots, x_k(t)^T]^T$ be the $s \times 1$ state vector, where the $x_i(t)$ are $n \times 1$ and $k = \max(p, q + 1, r + 1)$, so that $s = nk$. Now, setting $w(t) = x_1(t) = u(t)$, one gets the state law of motion

$$(2.5) \quad x(t) = Fx(t-1) + Ge(t) + Hz(t),$$

$$F = \begin{bmatrix} A_1 & I & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & I \\ A_k & 0 & \dots & \dots & 0 \end{bmatrix}, \quad G = \begin{bmatrix} B_0 \\ \vdots \\ B_{k-1} \end{bmatrix}, \quad H = \begin{bmatrix} C_0 \\ \vdots \\ C_{k-1} \end{bmatrix},$$

where $A_i = 0$ for $i > p$, $B_i = 0$ for $i > q$, $C_i = 0$ for $i > r$, and the zero and identity matrices in F are all $n \times n$. If all of the elements of $u(t)$ are observed directly (not as cross-sectional or temporal aggregates) and if there are no missing observations, then, in the observation equation $\Delta(\phi, t) = [I, 0, \dots, 0]$ and $\Lambda(t) = I$, where the zero and identity matrices are all $n \times n$.

3. Exact Gaussian Log-Likelihood Function.

First, we give some definitions. Let $Y(t) = \{y(1), \dots, y(t)\}$ and $Z(t) = \{z(1), \dots, z(t)\}$, for $t = 1, \dots, N$, so that $Y(N)$ and $Z(N)$ are the full samples of observations on $y(t)$ and $z(t)$. Let $L(t)$, for $t = 1, \dots, N$, denote the nonconstant part of -2 times the exact, Gaussian, log-likelihood function of $Y(t)$ conditional on $Z(t)$. $L(t)$ is directly a function of θ , which, in turn, is restricted by $\theta = \Psi(\phi, t)$; however, for simplicity and without loss, we mostly suppress explicit references to θ and ϕ in the notation. Let $x(t|t-1) = E[x(t)|Y(t-1), Z(t)]$ and $y(t|t-1) = E[y(t)|Y(t-1), Z(t)]$, with associated errors, $\tilde{x}(t) = x(t) - x(t|t-1)$ and $\xi(t) = y(t) - y(t|t-1)$, and error covariances, $V(t) = E[\tilde{x}(t)\tilde{x}(t)^T]$ and $M(t) = E[\xi(t)\xi(t)^T]$. (For present purposes, we could equivalently define $x(t|t-1)$ and $y(t|t-1)$ with $Z(N)$ instead of $Z(t)$ in the conditioning sets.) Following the usual practice, we call $\xi(t)$ the innovation of $y(t)$, even though an

innovation is usually defined to be a prediction error from predicting a variable solely with its current and past values; here $Z(t)$ is also used in the prediction of $y(t)$.

Given θ , $Y(N)$, and $Z(N)$, the algorithm computes $L(N)$ by iterating over $t = 1, \dots, N$, as follows. At the beginning of iteration t , $x(t|t-1)$, $V(t)$, and $L(t-1)$ are given from the previous iteration. Given the values of these quantities, $L(t-1)$ is updated with

$$(3.1) \quad M(t) = \Sigma_v(t) + D(t)V(t)D(t)^T,$$

$$(3.2) \quad \xi(t) = y(t) - D(t)x(t|t-1),$$

$$(3.3) \quad L(t) = L(t-1) + \ln|M(t)| + \xi(t)^T M(t)^{-1} \xi(t),$$

where $|\cdot|$ is the determinant; and, $x(t|t-1)$ and $V(t)$ are updated with

$$(3.4) \quad J(t) = F(t+1)V(t)D(t)^T M(t)^{-1},$$

$$(3.5) \quad x(t+1|t) = F(t+1)x(t|t-1) + H(t+1)z(t+1) + J(t)\xi(t),$$

$$(3.6) \quad V(t+1) = G(t+1)\Sigma_e(t+1)G(t+1)^T + F(t+1)V(t)F(t+1)^T \\ - J(t)M(t)J(t)^T.$$

This is the basic form of the Kalman filter; $J(t)$ is the Kalman gain matrix. In keeping with the notation $L(t)$, we have

written $F(\phi, t)$ as $F(t)$, etc., and shall continue to do so. Iterations (3.1) to (3.6) follow the statement of the basic Kalman filter given by Morf and Kailath [34] and Zadrozny [49] and differ from the statement of the basic filter given by Jones [24] and Ansley and Kohn [3] in that they do not also involve $x(t|t) = E[x(t)|Y(t), Z(t)]$ and its error covariance. These quantities are unnecessary for computing the likelihood function and are easily removed, so that, for purposes of computing the likelihood function, the two versions of the filter are essentially equivalent.

It is more efficient to compute with closely related iterations stated in terms of the normalized innovation vector. To state these iterations, we first note that $L(N)$ is computable if and only if $M(t)$ is positive definite over the sample, i.e., $M(t) > 0$, for $t = 1, \dots, N$. Further down in this section, we discuss restrictions on parameters which ensure that this condition holds. When $M(t) > 0$, then, it has a unique Cholesky (or square-root) decomposition, $M(t) = \Omega(t)\Omega(t)^T$, where $\Omega(t)$ is a unique, lower triangular matrix which has positive elements on its principal diagonal ([13, pp. 82-92]). $\Omega(t)$ is called the Cholesky factor (or square root) of $M(t)$.

Let $\eta(t)$ be the normalized, innovation vector defined by $\eta(t) = \Omega(t)^{-1}\xi(t)$, so that it has an identity covariance matrix. Then, equivalent to (3.1) to (3.6) are

$$(3.7) \quad \Omega(t)\Omega(t)^T = \Sigma_v(t) + D(t)V(t)D(t)^T,$$

$$(3.8) \quad \eta(t) = \Omega(t)^{-1}[y(t) - D(t)x(t|t-1)],$$

$$(3.9) \quad L(t) = L(t-1) + 2 \cdot \ln|\Omega(t)| + \eta(t)^T \eta(t),$$

$$(3.10) \quad K(t) = F(t+1)V(t)D(t)^T \Omega(t)^{-T},$$

$$(3.11) \quad x(t+1|t) = F(t+1)x(t|t-1) + H(t+1)z(t+1) + K(t)\eta(t),$$

$$(3.12) \quad V(t+1) = G(t+1)\Sigma_e(t+1)G(t+1)^T + F(t+1)V(t)F(t+1)^T \\ - K(t)K(t)^T,$$

where superscript $-T$ denotes inversion and transposition. By writing (3.7), we mean that $M(t)$ is first computed according to the right side of (3.7) and is, then, Cholesky factorized ([13, p. 86-89]).

To start the iterations, the initial values, $x(1|0)$ and $V(1)$, must be specified. Obviously, we also set $L(0) = 0$. We shall only explicitly treat the specification of $x(1|0)$ and $V(1)$ in the stationary case. We thus limit the discussion because, whereas there is general agreement about how to set $x(1|0)$ and $V(1)$ in the stationary case so as to obtain the exact likelihood function, this question is still not entirely settled in the nonstationary case. We shall, however, briefly discuss some of the methods which have been proposed for the nonstationary case.

There is general agreement that in the stationary case the exact likelihood is obtained when $x(1|0) = \mu_x$, the unconditional mean of $x(t)$, and $V(1) = \Sigma_x$, the unconditional covariance of

$x(t)$. A model is stationary when its law of motion and the structural part of its observation equation (e.g., $\Delta_1(\phi)$ in the example in section 2) are time invariant and when its state law of motion is asymptotically stable. The latter condition means that F has all eigenvalues inside the unit circle in the complex plane.

When constant terms are used to account for means of the data, it is appropriate in the stationary case to set $x(1|0) = \mu_x = 0$. In the stationary case, the unconditional covariance of $x(t)$, Σ_x , solves the (discrete-time, algebraic) Lyapunov equation,

$$(3.13) \quad \Sigma_x - F\Sigma_x F^T = G\Sigma_e G^T.$$

When F has all eigenvalues inside the unit circle and $G\Sigma_e G^T$ is positive (semi-) definite, then, (3.13) yields a unique, symmetric, and positive (semi-) definite value of Σ_x ([2, pp. 64-67]). Therefore, in the stationary case, the exact likelihood is obtained when $x(1|0) = 0$ and $V(1) = \Sigma_x$, where Σ_x solves (3.13).

There are various methods for solving (3.13). An obvious, but computationally inefficient, way to solve (3.13) is to apply the vectorization rule $\text{vec}(ABC) = [C^T \otimes A] \cdot \text{vec}(B)$ ([36, p. 954], [14, p. 25]) to it, to obtain

$$(3.14) \quad (I_{ss} - [F \otimes F])\text{vec}(\Sigma_x) = \text{vec}(G\Sigma_e G^T),$$

where I_{ss} is the $s^2 \times s^2$ identity matrix, s being the dimension

of $x(t)$. Equation (3.14) is in the standard form of a linear system, $Ab = c$, and can, therefore, be solved for $b = \text{vec}(\Sigma_x)$ by any standard method ([13, pp. 52-80]). Kohn and Ansley [28] discuss a more sophisticated version of this idea, which takes into account the symmetry of Σ_x and $G\Sigma_e G^T$, as well as, in ARMAX cases, the companion form of F . There are other, transformation and iterative, methods for solving (3.13) ([15], [44]).

When the model is nonstationary (time varying or asymptotically unstable), the unconditional mean and covariance, μ_x and Σ_x , are not well defined; in particular, when some eigenvalues of F are near the unit circle, (3.13) is ill conditioned. Therefore, in the nonstationary case, one cannot meaningfully set $x(1|0) = \mu_x$ and $V(1) = \Sigma_x$. The following methods have been suggested for setting these values in the nonstationary case.

First, $x(1|0)$ and $V(1)$ may be set according to prior information, although in economic applications little or nothing is generally known about $x(1)$, except to the extent that it comprises presample ($t < 1$) values of $y(t)$. As a way of imposing a diffuse, prior distribution on the unknown elements of $x(1)$, Harvey and Phillips [21], Ansley and Kohn [4], and others have suggested setting the prediction-error covariances of these state variables to infinity. When this is done, the values of $x(1|0)$ corresponding to the unknown variables do not affect the likelihood computations and can, thus, be set to any value. Ansley and Kohn [4], Kohn and Ansley [29], and Bell and Hillmer [7] showed that such a partially-diffuse prior is equivalently

and computationally more effectively implemented with a data-transformation method which is a generalization of differencing. Bergstrom [8] and Harvey and Stock [22] described methods for estimating $x(1|0)$ by generalized least squares and setting $V(1)$ to the covariance of the GLS estimator. Bergstrom suggested using conventional formulas; following Rosenberg [39], Harvey and Stock suggested using Kalman-filtering-like recursions. Fortunately, the problem of setting initial values is mitigated by the fact that when the model is stabilizable and detectable ([30, pp. 53-81]), conditions which generally hold, then, $L(N)$ is $O_p(1)$ in $x(1|0)$ and $V(1)$. This is true whether or not the model is stationary.

We now discuss restrictions on θ which ensure that $M(t) > 0$, for $t = 1, \dots, N$, i.e., that $L(N)$ is computable. In essence, $L(N)$ is computable if the model and the sampling scheme transmit enough variations from $e(t)$ and $\zeta(t)$ to $y(t)$, so that $y(t)$ has a nonsingular probability distribution. If $V(1) \geq 0$ and $\Sigma_v(t) + D(t)G(t)\Sigma_e(t)G(t)^T D(t)^T > 0$, for $t = 1, \dots, N$, then, a straightforward induction proof shows that $M(t) > 0$, for $t = 1, \dots, N$. $V(1) \geq 0$ holds automatically, by construction. Therefore, $L(N)$ is computable if $\Sigma_v(t) - \Lambda(t)\Sigma_\zeta(t)\Lambda(t)^T > 0$ or if $D(t)G(t)\Sigma_e(t)G(t)^T D(t)^T > 0$, for $t = 1, \dots, N$. In practice, $G(t)\Sigma_e(t)G(t)^T$ has much less than full rank, but $D(t)$ has full row rank and picks a full-rank part out of $G(t)\Sigma_e(t)G(t)^T$, so that $D(t)G(t)\Sigma_e(t)G(t)^T D(t)^T$ has full rank, for $t = 1, \dots, N$. For example, in the ARMAX model, (2.4) and (2.5), $D(t)G(t)\Sigma_e(t)G(t)^T D(t)^T = B_0 \Sigma_e B_0^T$, so that $L(N)$ is computable if

B_0 and Σ_e have full ranks. We shall assume, as is usually the case in practice, that $M(t)$ inherits full rank from $e(t)$, for $t = 1, \dots, N$.

We note that often the most convenient way to enforce symmetry and positive definiteness on $\Sigma_e(t)$ and $\Sigma_f(t)$, and, thereby, enforce $M(t) > 0$, is to reparameterize these covariances to their Choleksy factors. That is, reparameterize $\Sigma_e(t)$ to $Q(t)$, where $Q(t)$ is lower triangular and satisfies $Q(t)Q(t)^T = \Sigma_e(t)$, and impose $Q_{ii}(t) > 0$, for $i = 1, \dots, n$; and, reparameterize $\Sigma_f(t)$ to $R(t)$, where $R(t)$ is lower triangular and satisfies $R(t)R(t)^T = \Sigma_f(t)$, and impose $R_{ii}(t) > 0$, for $i = 1, \dots, m$.

We conclude this section by noting some computational efficiencies in the evaluation of (3.7) to (3.12). Because $\Omega(t)$ is lower triangular, it is best to view (3.8) as $\Omega(t)\eta(t) = y(t) - D(t)x(t|t-1)$ and to solve for $\eta(t)$ by forward elimination ([13, pp. 52-53]); similarly, it is best to view (3.10) as $\Omega(t)K(t)^T = D(t)V(t)F(t+1)^T$ and to solve for $K(t)^T$ by forward elimination. These solutions of (3.8) and (3.10) will be feasible if $\Omega_{ii}(t) > 0$, for $t = 1, \dots, N$ and $i = 1, \dots, m(t)$, which will be the case if $M(t) > 0$, for $t = 1, \dots, N$. Because $\Omega(t)$ is lower triangular, $\det[\Omega(t)] = \Omega_{11}(t) \cdots \Omega_{m(t)m(t)}(t)$, so that $\ln|\Omega(t)|$ is most effectively computed as $\sum_{i=1}^{m(t)} \ln[\Omega_{ii}(t)]$. By computing with (3.7) to (3.12) instead of with (3.1) to (3.6), one avoids having to compute the determinant or the inverse of $M(t)$. Structural information about patterns of zeroes and ones in $F(t)$, $G(t)$, $H(t)$, $\Sigma_e(t)$, $D(t)$, and $\Sigma_v(t)$ should, of course, also be exploited to avoid unnecessary multiplications with zeroes or ones. In ARMA

or ARMAX models especially, $F(t)$, $G(t)$, and $H(t)$ will have exploitable, sparse patterns.

4. Exact Gradient of the Log-Likelihood Function.

Following the advocacy of Neudecker [36] and Magnus and Neudecker [32], for algebraic and notational convenience, we shall derive and state results with differential forms. Differential forms are not directly useful for computations, but, as we now show, they are immediately convertible to computable, partial-derivative forms.

Let $A(\theta)$ be a representative, differentiable, $m \times n$, matrix function of the $p \times 1$, parameter vector $\theta = (\theta_1, \dots, \theta_p)^T$. The $m \times n$ matrices $\partial_k A = (\partial A_{ij} / \partial \theta_k)$, for $k = 1, \dots, p$, collect first-order derivatives of $A(\theta)$ in partial-derivative form, where A_{ij} is the (i, j) element of A . Let $dA_{ij} = \sum_{k=1}^p (\partial A_{ij} / \partial \theta_k) d\theta_k$, where $d\theta_k$ is an infinitesimal variation in θ_k . The $m \times n$ matrix $dA = (dA_{ij})$ is the differential form of first-order derivatives of $A(\theta)$. Except for the scalar factor $d\theta_k$, $\partial_k A$ is a special case of dA , so that to obtain partial-derivative forms from differential forms, we only have to everywhere replace d with ∂_k , for $k = 1, \dots, p$. Let $a = (a_1, \dots, a_{mn})^T = \text{vec}(A)$. The $mn \times p$ matrix $\nabla A = (\partial a_h / \partial \theta_k) = [\text{vec}(\partial_1 A), \dots, \text{vec}(\partial_p A)]$ is the gradient form of first-order derivatives of $A(\theta)$ (usually ∇A is called the gradient when A is a scalar and is called the Jacobian otherwise).

We shall use the following differentiation rules. Suppose $A(\theta)$ and $B(\theta)$ are representative, $m \times n$ and $n \times q$,

differentiable, matrix functions of θ . Componentwise application of the scalar product rule of differentiation yields the matrix product rule of differentiation, $d(AB) = dA \cdot B + A \cdot dB$ ([36, p. 955], [14, p. 79]). Setting $B = A^{-1}$ when A is square and invertible and using $dI = 0$ shows that $d(A^{-1}) = -A^{-1} \cdot dA \cdot A^{-1}$. To avoid confusion, it should be understood that d only operates on the matrix immediately following it, unless it is followed by parentheses: e.g., $dAB = d(A)B \neq d(AB)$. Finally, when A is a square matrix and $\det(A) > 0$, then, $d[\ln(\det(A))] = \text{tr}[A^{-1} \cdot dA]$ ([14, p. 79]).

Using these differentiation rules, the differentials of (3.7) to (3.12) with respect to θ are seen to be

$$(4.1) \quad d\Omega(t)\Omega(t)^T + \Omega(t)d\Omega(t)^T = d\Sigma_V(t) + dD(t)V(t)D(t)^T \\ + D(t)dV(t)D(t)^T + D(t)V(t)dD(t)^T,$$

$$(4.2) \quad d\eta(t) = -\Omega(t)^{-1}[d\Omega(t)\eta(t) + dD(t)x(t|t-1) \\ + D(t)dx(t|t-1)],$$

$$(4.3) \quad dL(t) = dL(t-1) + 2 \cdot \text{tr}[\Omega(t)^{-1}d\Omega(t)] + 2 \cdot \eta(t)^T d\eta(t),$$

$$(4.4) \quad dK(t) = dF(t+1)V(t)D(t)^T\Omega(t)^{-T} + F(t+1)dV(t)D(t)^T\Omega(t)^{-T} \\ + F(t+1)V(t)dD(t)^T\Omega(t)^{-T} - K(t)d\Omega(t)^T\Omega(t)^{-T}$$

$$(4.5) \quad dx(t+1|t) = dF(t+1)x(t|t-1) + F(t+1)dx(t|t-1) \\ + dH(t+1)z(t+1) + dK(t)\eta(t) + K(t)d\eta(t),$$

$$\begin{aligned}
(4.6) \quad dV(t+1) &= dG(t+1)\Sigma_e(t+1)G(t+1)^T + G(t+1)d\Sigma_e(t+1)G(t+1)^T \\
&+ G(t+1)\Sigma_e(t+1)dG(t+1)^T + dF(t+1)V(t)F(t+1)^T \\
&+ F(t+1)dV(t)F(t+1)^T + F(t+1)V(t)dF(t+1)^T \\
&- dK(t)K(t)^T - K(t)dK(t)^T.
\end{aligned}$$

In obtaining (4.1) to (4.6), $dy(t) = 0$ and $dz(t) = 0$, for $t = 1, \dots, N$, were used; these equalities hold because the given realizations of $y(t)$ and $z(t)$ are independent of variations in θ which are being considered. Note also that, because $\Omega(t)$ is lower triangular, in (4.3) it is expedient to compute $\text{tr}[\Omega(t)^{-1}d\Omega(t)]$ as $\sum_{i=1}^{m(t)} [\Omega_{ii}(t)^{-1}d\Omega_{ii}(t)]$.

To describe how to efficiently solve for $d\Omega(t)$, let the right side of (4.1) be denoted by $C(t)$. Let $\Omega_{ij}(t)$, $d\Omega_{ij}(t)$, and $C_{ij}(t)$, respectively, denote the (i,j) elements of $\Omega(t)$, $d\Omega(t)$, and $C(t)$. Because (4.1) is symmetric and $\Omega(t)$ and $d\Omega(t)$ are lower triangular, we only need to consider (i,j) for $i = 1, \dots, m(t)$ and $j = 1, \dots, i$. The lower triangularity of $\Omega(t)$ and $d\Omega(t)$ imply that, for $i \geq j$, the (i,j) element of (4.1) is $\sum_{k=1}^j [d\Omega_{ik}(t)\Omega_{jk}(t) + \Omega_{ik}(t)d\Omega_{jk}(t)] = C_{ij}(t)$. Therefore, for $j = 1, \dots, m(t)$,

$$(4.7) \quad d\Omega_{jj}(t) = \Omega_{jj}(t)^{-1} [C_{jj}(t)/2 - \sum_{k=1}^{j-1} d\Omega_{jk}(t)\Omega_{jk}(t)],$$

the summation over k being null when $j = 1$; and, for $i = j, \dots, m(t)$,

$$(4.8) \quad d\Omega_{ij}(t) = \Omega_{jj}(t)^{-1} \{ C_{ij}(t) - \Omega_{ij}(t) d\Omega_{jj}(t) \\ - \sum_{k=1}^{j-1} [d\Omega_{ik}(t) \Omega_{jk}(t) + \Omega_{ik}(t) d\Omega_{jk}(t)] \},$$

the summation over k being null when $j = 1$.

Therefore, given $\Omega(t)$ and $C(t)$, the elements of $d\Omega(t)$ are recursively computed with (4.7) and (4.8), in the order (1,1), ..., (m(t),1), (2,2), ..., (m(t),2), ..., (m(t),m(t)). The only condition required to execute the gradient recursions, (4.1) to (4.8), is that $\Omega(t)$ be nonsingular, i.e., $\Omega_{ii}(t) \neq 0$, for $i = 1, \dots, m(t)$. This condition, of course, holds when $M(t) > 0$, for $t = 1, \dots, N$, i.e., when $L(N)$ is computable.

The relevant starting values for (4.1) to (4.8) are $x(1|0)$, $dx(1|0)$, $V(1)$, $dV(1)$, $L(0)$, and $dL(0)$. The values of $x(1|0)$, $V(1)$, and $L(0)$ are chosen as before. Of course, $L(0) = 0$ implies $dL(0) = 0$. When means are accounted for with z , so that $x(1|0) = 0$, then, $dx(1|0) = 0$. When $V(1) = \Sigma_x$, where Σ_x solves (3.13), then, $dV(1) = d\Sigma_x$, where $d\Sigma_x$ solves

$$(4.9) \quad d\Sigma_x - F d\Sigma_x F^T = dF \Sigma_x F^T + F \Sigma_x dF^T + dG \Sigma_e G^T + G d\Sigma_e G^T + G \Sigma_e dG^T,$$

which is obtained by differentiating (3.13). Equation (4.9) has the same, Lyapunov form as (3.13) and can, therefore, be solved in the same manner.

We remind the reader that to make the relations just derived computable, d is replaced with ∂_k everywhere, for $k = 1, \dots, p$. The computations involve a double loop: for $t = 1, \dots, N$ and for $k = 1, \dots, p$. When $t = 1, \dots, N$ is the outer loop, the

observations only need to be traversed once; when $k = 1, \dots, p$ is the outer loop, the storage of intermediate partial derivative matrices is reduced. In practice, p and N will generally, respectively, be relatively small and relatively large, so that it will generally be preferable to let $t = 1, \dots, N$ be the outer loop. Also, because the gradient computations use quantities propagated in the likelihood computations, it is expedient to merge the likelihood and gradient computations, e.g., for each t to evaluate in the order (3.7), (4.7), (4.8), (3.8), (4.2), ..., (3.12), (4.6).

Again, we remind the reader to take advantage of all other, generally occurring sparsity. In this regard the partial-derivative forms of the constituent matrices of θ , namely $\partial_k F(t)$, $\partial_k G(t)$, $\partial_k H(t)$, $\partial_k \Sigma_e(t)$, $\partial_k D(t)$, and $\partial_k \Sigma_v(t)$, will be extremely sparse selection matrices: for each particular k , one of these matrices has one element equal to one and has all other elements equal to zero; the remaining of these matrices are equal to zero. For example, suppose that we are considering an ARMAX model in the form (2.4) and (2.5) and that θ_1 is the (1,1) element of the leading AR coefficient matrix, A_1 . Then, the (1,1) element of $\partial_1 F$ is 1, all of its other elements are 0, and $\partial_1 G$, $\partial_1 H$, $\partial_1 \Sigma_e$, $\partial_1 D$, and $\partial_1 \Sigma_v$ are 0. One should directly make the selections implied by these matrices and not multiply with them. By contrast, derived quantities like $\partial_k \eta(t)$ and $\partial_k K(t)$ will generally be full.

The gradient is extended as follows to models with differentiable restrictions on θ . Let the differentiable mapping $\theta = \Psi(\phi)$ describe restrictions on θ in terms of ϕ , as in section

2. Let us write the gradient of the log-likelihood parameterized in θ as $\nabla_{\theta} L(N)$ and the gradient of the restriction mapping as $\nabla_{\phi} \Psi$. Consider the composite, differentiable, matrix function $C(\theta) = B(A(\theta))$ formed with the differentiable, matrix functions $B(A)$ and $A(\theta)$. The gradient of $A(\theta)$ can be alternatively defined by $\text{vec}(dA) = \nabla A \cdot \text{vec}(d\theta) = \nabla A \cdot d\theta$. Then, the gradient of $C(\theta)$ is seen to be given by the chain rule $\nabla C = \nabla B \cdot \nabla A$. Applying this result to the restricted log-likelihood parameterized in ϕ shows that its gradient vector is given by

$$(4.10) \quad \nabla_{\phi} L(N) = \nabla_{\theta} L(N) \cdot \nabla_{\phi} \Psi.$$

5. Approximate Hessian of the Log-Likelihood Function.

We first define second-order matrix derivatives, which extend the first-order derivatives defined in the previous section. Let $A(\theta)$ be a representative, twice-differentiable, $m \times n$, matrix function of $\theta = (\theta_1, \dots, \theta_p)^T$. The $m \times n$ matrices $\partial_{kl}^2 A = \{\partial^2 A_{ij} / \partial \theta_k \partial \theta_l\}$, for k and $l = 1, \dots, p$, collect second-order derivatives of $A(\theta)$ in partial-derivative form, where A_{ij} is the (i, j) element of A . Let $d^2 A_{ij} = \sum_{k=1}^p \sum_{l=1}^p (\partial^2 A_{ij} / \partial \theta_k \partial \theta_l) d\theta_k d\theta_l$, where $\{d\theta_k \mid k = 1, \dots, p\}$ and $\{d\theta_l \mid l = 1, \dots, p\}$ are independent, infinitesimal sets of variations in θ . The $m \times n$ matrix $d^2 A = \{d^2 A_{ij}\}$ is the differential form of second-order derivatives of $A(\theta)$. Let $b = (b_1, \dots, b_{mnp})^T = \text{vec}(\nabla A)$. The $mnp \times p$ matrix $\nabla^2 A = \{\partial b_h / \partial \theta_k\} = [\text{vec}(\partial_1(\nabla A)), \dots, \text{vec}(\partial_p(\nabla A))]$ is the Hessian form of second-order derivatives of $A(\theta)$.

After differentiating (4.3), consolidating terms, and summing over $t = 1, \dots, N$, we get

$$\begin{aligned}
 (5.1) \quad d^2L(N) = & 2 \cdot \sum_{t=1}^N \{ -\text{tr}[\Omega(t)^{-1} d\Omega(t) \Omega(t)^{-1} d\Omega(t)] \\
 & + \text{tr}[\Omega(t)^{-1} d^2\Omega(t)] + d\eta(t)^T d\eta(t) \\
 & + 2 \cdot \eta(t)^T \Omega(t)^{-1} d\Omega(t) \Omega(t)^{-1} d\Omega(t) \eta(t) \\
 & + 2 \cdot \eta(t)^T \Omega(t)^{-1} d\Omega(t) \Omega(t)^{-1} dD(t) x(t|t-1) \\
 & + 2 \cdot \eta(t)^T \Omega(t)^{-1} d\Omega(t) \Omega(t)^{-1} D(t) dx(t|t-1) \\
 & - \eta(t)^T \Omega(t)^{-1} d^2\Omega(t) \eta(t) \\
 & - \eta(t)^T \Omega(t)^{-1} d^2D(t) x(t|t-1) \\
 & - 2 \cdot \eta(t)^T \Omega(t)^{-1} dD(t) dx(t|t-1) \\
 & - \eta(t)^T \Omega(t)^{-1} D(t) d^2x(t|t-1) \}.
 \end{aligned}$$

The recursions in sections 3 and 4 produce exact values of the log-likelihood and of its gradient for any admissible value of θ . By contrast, the approximate Hessian derived in this section accurately approximates the exact Hessian only to the extent that some additional conditions hold. One possible set of conditions is that: (i) the model being used correctly represents the true data generating process; (ii) the model is stationary; (iii) $\theta = \theta_0$, the true value of θ (or $\theta = \hat{\theta}$, a consistent estimate of θ_0); and, (iv) $N \rightarrow \infty$.

Under these additional conditions, $\eta(t)$ is uncorrelated with $Y(t-1)$ for all $t \geq 2$. Also, because $dx(t|t-1)$ and $d^2x(t|t-1)$ lie in the linear space spanned by $Y(t-1)$, $E\eta(t)dx(t|t-1)^T = 0$ and $E\eta(t)d^2x(t|t-1)^T = 0$. Then, following a line of argument similar to that given by Tunnicliffe-Wilson [43, pp. 78-79], it

can be shown that $\eta(t)$, $dx(t|t-1)$, and $d^2x(t|t-1)$ deviate from realizations of a stationary, Gaussian, generalized, linear process by $o_p(\gamma^t)$, where $0 < \gamma < 1$ is the maximal absolute eigenvalue of F . As a result, Hannan's Theorem 6 [16, p. 210] implies that, almost surely,

$$(5.2) \quad \lim_{N \rightarrow \infty} (1/N) \cdot \sum_{t=1}^N \eta(t) dx(t|t-1)^T = E\eta(t) dx(t|t-1)^T = 0,$$

$$(5.3) \quad \lim_{N \rightarrow \infty} (1/N) \cdot \sum_{t=1}^N \eta(t) d^2x(t|t-1)^T = E\eta(t) d^2x(t|t-1)^T = 0,$$

$$(5.4) \quad \lim_{N \rightarrow \infty} (1/N) \cdot \sum_{t=1}^N \eta(t) \eta(t)^T = E\eta(t) \eta(t)^T = I.$$

We assume that enough assumptions are in force so that $\Omega(t)$ and its derivatives converge to limiting values or to periodic cycles as $t \rightarrow \infty$. In the time invariant case, a sufficient condition for these quantities to converge to unique, limiting values is that the state-space form of the model, (2.1) and (2.3), is stabilizable and detectable ([30, pp. 459-467], [17]). Cases in which the model is time invariant and $D(t)$ varies because of missing data have apparently not been studied. Some numerical experiments suggest that when the model is time invariant, stabilizable, and detectable, and data are periodically missing (e.g., different variables are observed at different frequencies), then, $\Omega(t)$ and its derivatives converge to periodic cycles.

Given that $\Omega(t)$ and its derivatives converge to limiting values or to periodic cycles, equations (5.2) to (5.4) imply that

on the right side of (5.1): the sum of terms 1 and 4 differs from the negative of term 1 by $o_p(N)$; the sum of terms 2 and 7 is $o_p(N)$; and, terms 5, 6, 8, 9, and 10 are $o_p(N)$. Then, $d^2L(N)$ has the $o_p(N)$ approximation

$$(5.5) \quad d^2f(N) = 2 \cdot \sum_{t=1}^N \{ \text{tr}[\Omega(t)^{-1} d\Omega(t) \Omega(t)^{-1} d\Omega(t)] \\ + d\eta(t)^T d\eta(t) \},$$

which yields the approximate Hessian matrix $\nabla^2 f(N)$, with (j,k) element

$$(5.6) \quad \partial_{jk}^2 f(N) = 2 \cdot \sum_{t=1}^N \{ \text{tr}[\Omega(t)^{-1} \partial_j \Omega(t) \Omega(t)^{-1} \partial_k \Omega(t)] \\ + \partial_j \eta(t)^T \partial_k \eta(t) \},$$

where j and $k = 1, \dots, p$. To obtain $\nabla^2 f(N)$, in addition to $L(N)$ and $\nabla L(N)$, one only needs to additionally compute with (5.6). Because $\Omega(t)$, $\partial_j \Omega(t)$, and $\partial_k \Omega(t)$ are lower triangular, it is expedient to compute $\text{tr}[\Omega(t)^{-1} \partial_j \Omega(t) \Omega(t)^{-1} \partial_k \Omega(t)]$ as $\sum_{i=1}^m [\partial_j \Omega_{ii}(t) \partial_k \Omega_{ii}(t) / \Omega_{ii}(t)^2]$.

We now show that $\nabla^2 f(N)$ is positive semi-definite by construction, as desired. Let $A(\theta)$ be a twice-differentiable, $m \times n$, matrix function of θ . Analogous to $\text{vec}(dA) = \nabla A \cdot d\theta$, it can be shown that $\nabla^2 A$ satisfies $\text{vec}(d^2 A) = [d\theta^T \otimes I_{mn}] \cdot \nabla^2 A \cdot d\theta$. Accordingly, in (5.5) we make the substitutions $d^2 f(N) = d\theta^T \cdot \nabla^2 f(N) \cdot d\theta$, $d\eta(t) = \nabla \eta(t) \cdot d\theta$, and $\text{vec}(d\Omega(t)) = \nabla \Omega(t) \cdot d\theta$. Let A , B , and C be any matrices (not necessarily differentiable) conformable to the product ABC . Then, $\text{tr}(AB) = \text{vec}(A^T)^T \cdot \text{vec}(B)$

([36, p. 954], [14, p. 19]) and, as noted before, $\text{vec}(ABC) = [C^T \otimes A] \cdot \text{vec}(B)$.

We apply these rules to the right side of (5.5) to get its terms into quadratic forms in $d\theta$. We, then, cancel the common factors, $d\theta^T$ and $d\theta$, across both sides of the equality. The result is

$$(5.7) \quad \nabla^2 f(N) = 2 \cdot \sum_{t=1}^N \{ \nabla \Omega(t)^T [\Omega(t)^{-1} \otimes \Omega(t)^{-1}] \nabla \Omega(t) + \nabla \eta(t)^T \nabla \eta(t) \}.$$

Equation (5.7) is positive semi-definite by construction, in particular, because $\Omega(t) > 0$ implies $[\Omega(t)^{-1} \otimes \Omega(t)^{-1}] > 0$. When, in addition to $\Omega(t) > 0$, $[\nabla \Omega(t)^T, \nabla \eta(t)^T]^T$ has full column rank for at least one $t = 1, \dots, N$, then, $\nabla^2 f(N)$ will be positive definite; $[\nabla \Omega(t)^T, \nabla \eta(t)^T]^T$ is expected to have full column rank when θ is locally identifiable ([40, pp. 81-82]).

Analogous to the general, gradient, chain rule $\nabla C = \nabla B \cdot \nabla A$, for the composite mapping $C(x) = B(A(x))$, where B and C are $q \times r$ and x is $s \times 1$, we can verify the Hessian chain rule $\nabla^2 C = [\nabla A^T \otimes I_{qr}] \cdot \nabla^2 B \cdot \nabla A + [I_s \otimes \nabla B] \cdot \nabla^2 A$. Consider $A(x)$ to be the restriction function, $\theta = \Psi(\phi, t)$, and consider $B(A)$ to be the log-likelihood function parameterized in θ . Asymptotically, when θ is at θ_0 (or at a consistent estimate of it), then, $\nabla B = 0$. Thus, we get the chain rule extension of $\nabla_{\theta}^2 f(N)$ to $\nabla_{\phi}^2 f(N)$,

$$(5.8) \quad \nabla_{\phi}^2 f(N) = \nabla_{\phi} \Psi^T \cdot \nabla_{\theta}^2 f(N) \cdot \nabla_{\phi} \Psi.$$

6. Sample and Asymptotic Information Matrices.

The sample and asymptotic information matrices of the parameter vector θ are $I_\theta(N) = (1/2) \cdot E[\nabla L(N)^T \nabla L(N)]$ and $I_\theta(\infty) = (N/2) \cdot \lim_{N \rightarrow \infty} E[\nabla L(N)^T \nabla L(N)/N]$. Under regularity conditions ([40, pp. 37-38]) which are known to hold in the present case, it is easier to compute these with equivalent expressions obtained by replacing $\nabla L(N)^T \nabla L(N)$ with $\nabla^2 L(N)$. In fact, because, as in the previous section, we are presuming that the model being considered is correct and that $\theta = \theta_0$ (or $\theta = \hat{\theta}$, a consistent estimate of θ_0 , and $N \rightarrow \infty$), the expectation operator can be understood to be with respect to the true probability distribution. Therefore, the right-most equalities of (5.2) to (5.4) apply, so that $E[\nabla^2 L(N)] = E[\nabla^2 f(N)]$, for finite N and as $N \rightarrow \infty$. Accordingly, we are concerned with computing the sample and asymptotic, information matrices in terms of $I_\theta(N) = (1/2) \cdot E[\nabla^2 f(N)]$ and $I_\theta(\infty) = (N/2) \cdot \lim_{N \rightarrow \infty} E[\nabla^2 f(N)/N]$.

To compute $E[\partial_{jk}^2 f(N)]$ with (5.6), we need to develop a method for computing $E[\partial_j \eta(t)^T \partial_k \eta(t)]$. To do this, we use $dy(t) = 0$, $dv(t) = 0$, and $de(t) = 0$, for $t = 1, \dots, N$. Like $dy(t) = 0$, $dv(t) = 0$ and $de(t) = 0$ hold because the given realizations of $v(t)$ and $e(t)$ are independent of variations in θ which are being considered. Using $dy(t) = 0$ and $dv(t) = 0$, (2.3) implies that $dD(t)x(t) + D(t)dx(t) = 0$. Therefore, (3.8) and (4.2) may be combined as

$$(6.1) \quad \eta_j^*(t) = D_j^*(t) \tilde{x}_j^*(t) + E_j^*(t) v(t),$$

for $j = 1, \dots, p$, where $\eta_j^*(t)$ and $\tilde{x}_j^*(t)$ are defined by $\eta_j^*(t) = [\eta(t)^T, \partial_j \eta(t)^T]^T$ and $\tilde{x}_j^*(t) = [\tilde{x}(t)^T, \partial_j \tilde{x}(t)^T]^T$, so that the coefficient matrices $D_j^*(t)$ and $E_j^*(t)$ are given by

$$D_j^*(t) = \begin{bmatrix} \Omega(t)^{-1} D(t) & 0 \\ \Omega(t)^{-1} \partial_j D(t) & \Omega(t)^{-1} D(t) \end{bmatrix},$$

$$E_j^*(t) = \begin{bmatrix} \Omega(t)^{-1} \\ -\Omega(t)^{-1} \partial_j \Omega(t) \Omega(t)^{-1} \end{bmatrix}.$$

Because $v(t)$ is uncorrelated with $\tilde{x}(t)$ and $\partial_j \tilde{x}(t)$, (6.1) implies that

$$(6.2) \quad E[\eta_j^*(t) \eta_k^*(t)^T] = D_j^*(t) V_{jk}^*(t) D_k^*(t)^T + E_j^*(t) \Sigma_v(t) E_k^*(t)^T,$$

for j and $k = 1, \dots, p$, where $V_{jk}^*(t) = E[\tilde{x}_j^*(t) \tilde{x}_k^*(t)^T]$. Then, $E[\partial_j \eta(t) \partial_k \eta(t)^T]$ is given by the (2,2) (south-east) quadrant of (6.2).

To continue, we derive a recursion, corresponding to (3.12), for updating $V_{jk}^*(t)$ to $V_{jk}^*(t+1)$. We carry out the following steps: combine (2.1) and (3.11) into prediction-error form; use $de(t) = 0$ to differentiate (2.1); use the differential of (2.1) to put (4.5) into prediction-error form; combine the two prediction-error, differential forms into a single equation; and, eliminate $\eta_j^*(t)$ from this equation using (6.1). The result is

$$(6.3) \quad \tilde{x}_j^*(t+1) = \Phi_j^*(t+1)\tilde{x}_j^*(t) + G_j^*(t+1)e(t+1) - K_j^*(t)E_j^*(t)v(t),$$

for $j = 1, \dots, p$, where the as yet undefined coefficient matrices in this equation are given by

$$\Phi_j^*(t+1) = F_j^*(t+1) - K_j^*(t)D_j^*(t),$$

$$F_j^*(t) = \begin{bmatrix} F(t) & 0 \\ \partial_j F(t) & F(t) \end{bmatrix}, \quad G_j^*(t) = \begin{bmatrix} G(t) \\ \partial_j G(t) \end{bmatrix},$$

$$K_j^*(t) = \begin{bmatrix} K(t) & 0 \\ \partial_j K(t) & K(t) \end{bmatrix}.$$

Then, because $\tilde{x}_j^*(t)$, $e(t+1)$, and $v(t)$ are uncorrelated with each other, (6.3) implies that

$$(6.4) \quad V_{jk}^*(t+1) = \Phi_j^*(t+1)V_{jk}^*(t)\Phi_k^*(t+1)^T + G_j^*(t+1)\Sigma_e(t+1)G_k^*(t+1)^T \\ + K_j^*(t)E_j^*(t)\Sigma_v(t)E_k^*(t)^TK_k^*(t)^T.$$

A little algebra shows that the (1,1) quadrant of (6.4) is identical to (3.12); this must be the case because the (1,1) quadrant of (6.4) and (3.12) both update $V(t) = E[\tilde{x}(t)\tilde{x}(t)^T]$.

The starting value of $V_{jk}^*(t)$ is set in essentially the same way as the starting value of $V(t)$: in the stationary case, $V_{jk}^*(1)$ solves

$$(6.5) \quad V_{jk}^*(1) - F_j^* V_{jk}^*(1) F_k^{*T} = G_j^* \Sigma_e G_k^{*T}.$$

Like (4.9), (6.5) has the Lyapunov form of (3.13) and can, therefore, be solved in the same way.

$I_\theta(N) = (1/2) \cdot E[V^2 f(N)]$ is, thus, computed with (5.6) by appending the (2,2) quadrant of (6.2) and the (2,1) and (2,2) quadrants of (6.4) and (6.5) to the recursions of sections 3 and 4 which are needed to produce $\Omega(t)$ and $\partial_k \Omega(t)$. Again, we emphasize that lower triangularity and other sparsity of the relevant, coefficient sub-matrices in (6.2), (6.4), and (6.5) should be exploited in the computations; we shall not further explicate these computational efficiencies.

To compute $I_\theta(\infty)$ one continues in this fashion until $E[V^2 f(N)/N]$ has converged in some norm (e.g., the L_2 norm [13, pp. 11-16]). When $\Omega(t)$, $\partial_j \Omega(t)$, and $V_{jk}^*(t)$ converge to limiting values as $t \rightarrow \infty$, as will be the case when the model is time invariant, stabilizable, and detectable, and no data are missing, then, $I_\theta(\infty)$ can be more simply computed in terms of $f(N)$, the N -th term in (5.6): $I_\theta(\infty) = N \cdot \lim_{N \rightarrow \infty} f(N)$.

The Hessian, chain rule (5.8) implies the similar, information-matrix, chain rule,

$$(6.6) \quad I_\phi(N) = \nabla_\phi \Psi^T \cdot I_\theta(N) \cdot \nabla_\phi \Psi,$$

which is valid for finite N and in the limit as $N \rightarrow \infty$.

7. Concluding Remarks.

The cumulated effect of rounding or truncation errors inherent to finite-precision, computer arithmetic may cause, after a certain number of iterations, (3.7), (3.10), and (3.12) to produce a value of $M(t)$ which is not positive definite. There are so-called square-root filtering algorithms which avoid this problem by propagating square roots of $M(t)$ and $V(t)$ instead of the covariances themselves. Square-root filters have the following advantages, which come at the cost of greater computational complexity: (i) given a nonsingular value of $\Omega(t)$, $M(t) = \Omega(t)\Omega(t)^T$ is always positive definite, even after rounding or truncation; (ii) the effective stored precision of a covariance matrix is doubled when it is stored in terms of its square root; (iii) square-root algorithms are numerically more stable because they propagate with perfectly conditioned, orthogonal, transformation matrices ([13, pp. 24-29]). We did not develop gradient, Hessian, and information matrix algorithms from a square-root likelihood algorithm because differentiation destroys orthogonality and because the resulting algorithms would involve substantially more computations than the present ones. In any case, recursions (3.7), (3.10), and (3.12) can be replaced in algorithms 1 or 2 with a square-root analogue, e.g., the one described in the appendix. For further discussions comparing Kalman and square-root filtering, see [2, pp. 147-164], [34], [35], and [43].

The approximate Hessian, sample information, and asymptotic information matrices considered here are generally asymptotically equivalent. Nevertheless: (i) to save on computations, it seems

best to use the approximate Hessian in nonlinear-estimation iterations; (ii) although there is some controversy about this ([12]), it seems best to use the sample or asymptotic information matrices to compute covariances (Cramer-Rao bounds) of the estimated parameters, because these matrices better reflect asymptotic theory of inference ([40, pp. 68-86]) than the approximate Hessian matrix; and, (iii) although theory ([10, pp. 1071-1073]) indicates that local identifiability is checked by checking the rank of the sample information matrix, the approximate Hessian matrix may better detect underidentifiability caused by insufficient variation in the data. Generally, one will only be able to numerically determine the rank of $I_{\phi}(N)$ or $\nabla_{\phi}^2 f(N)$ at a representative scatter of values of ϕ . The rank of a matrix can be reliably calculated with the singular value decomposition ([13, pp. 16-20]). Present results should be especially useful when data are missing; the principal advantage here of the Kalman filter is its ability to automatically handle any pattern of missing data.

Appendix: A Householder-Transformation Square-Root Filter.

Let (3.7), (3.10), and (3.12) be replaced with

(A.1)

$$\begin{bmatrix} \Omega(t) & 0 & 0 \\ K(t) & W(t+1) & 0 \end{bmatrix} = \begin{bmatrix} \Lambda(t)R(t) & D(t)W(t) & 0 \\ 0 & F(t+1)W(t) & G(t+1)Q(t+1) \end{bmatrix} \cdot P(t),$$

where $Q(t)$, $R(t)$, and $W(t)$ are lower-triangular square roots of $\Sigma_e(t)$, $\Sigma_f(t)$, and $V(t)$, and $P(t)$ is an orthogonal matrix to be

specified. Viewed as $B(t) = A(t)P(t)$, (A.1) defines the $(m(t)+s) \times (m(t)+s+n)$ matrices $A(t)$ and $B(t)$. When $P(t)$ is any $(m(t)+s+n) \times (m(t)+s+n)$ orthogonal matrix which induces the indicated pattern of zeroes in $B(t)$, then, (3.7), (3.10), and (3.12) are equivalent to (A.1). This is immediately verified by multiplying out $B(t)B(t)^T = A(t)P(t)P(t)^T A(t)^T$ and using $P(t)P(t)^T = I$. Because $\Omega(t)$ and $W(t)$ are lower triangular, $B(t)$ is also lower triangular.

Let $A_{ij}(t)$ denote the (i,j) element of $A(t)$. For $j = 1, \dots, m(t) + s$, let the scalar $\alpha_j(t)$, the $(m(t)+s+n) \times 1$ vector $\nu_j(t)$, and the $(m(t)+s+n) \times (m(t)+s+n)$ matrix $P_j(t)$ be defined sequentially by

$$(A.2) \quad \alpha_j(t) = [A_{jj}(t)^2 + \dots + A_{j,m(t)+s+n}(t)^2]^{1/2},$$

$$(A.3) \quad \nu_j(t) = [0, \dots, 0, A_{jj}(t) + \text{sign}(A_{jj}(t)) \cdot \alpha_j(t), \\ A_{j,j+1}(t), \dots, A_{j,m(t)+s+n}(t)]^T,$$

$$(A.4) \quad P_j(t) = I - 2[\nu_j(t)\nu_j(t)^T]/[\nu_j(t)^T\nu_j(t)].$$

Then, $P(t)$ is given by

$$(A.5) \quad P(t) = P_1(t) \cdots P_j(t) \cdots P_{m(t)+s}(t).$$

For $j = 1, \dots, m(t) + s$, let $B_j(t) = A(t)P_1(t) \cdots P_j(t)$. Postmultiplication of $B_{j-1}(t)$ by the Householder transformation matrix $P_j(t)$ causes: (i) the first $j-1$ rows of $B_{j-1}(t)$ and $B_j(t)$

to be identical; (ii) the (j,j) element of $B_j(t)$ to be nonnegative; and, (iii) the elements $(j,j+1), \dots, (j,m(t)+s+n)$ of $B_j(t)$ to be zero ([13, pp. 38-43]). Therefore, $B(t) = B_{m(t)+s}(t)$ is a lower-triangular matrix, with nonnegative elements on its principal diagonal, as desired. In practice, $\nu_j(t)^T \nu_j(t) = 0$ causes no difficulties in the division in (A.4), because, when this is the case, $P_j(t) = I$ and (A.2) to (A.4) are replaced by $P_j(t) = I$. When the state-space coefficient matrices are especially sparse, it may be more efficient to instead construct $P(t)$ with a sequence of Givens transformations ([13, pp. 43-47]).

REFERENCES

1. H. Akaike, Maximum likelihood identification of Gaussian autoregressive moving-average models, Biometrika 60, 255-265 (1973).
2. B. D. O. Anderson and J. B. Moore, Optimal Filtering, Prentice-Hall, Englewood Cliffs, NJ (1979).
3. C. F. Ansley and R. Kohn, Exact likelihood of vector autoregressive moving-average process with missing or aggregated data, Biometrika 70, 275-278 (1983).
4. C. F. Ansley and R. Kohn, Estimation, filtering, and smoothing in state space models with incompletely specified initial conditions, Annals of Statistics 13, 1286-1316 (1985).
5. C. F. Ansley and R. Kohn, A structured state space approach to computing the likelihood of an ARIMA process and its derivatives, Journal of Statistical Computation and Simulation 21, 135-169 (1985).
6. C. F. Ansley and R. Kohn, Computing the likelihood and its derivatives for a Gaussian ARMA model, Journal of Statistical Computation and Simulation 22, 229-263 (1985).
7. W. R. Bell and S. Hillmer, Initializing the Kalman filter in the nonstationary case: with applications to signal

- extraction, Research Report, Statistical Research Division, U.S. Bureau of the Census, Washington, DC (1987).
8. A. R. Bergstrom, The estimation of parameters in nonstationary higher-order continuous-time dynamic models, Econometric Theory 1, 369-385 (1985).
 9. E. R. Berndt, B. H. Hall, R. E. Hall, and J. A. Hausman, Estimation and inference in nonlinear structural models, Annals of Economic and Social Measurement 3 & 4, 653-665 (1974).
 10. R. Bowden, The theory of parametric identification, Econometrica 41, 1069-1074 (1973).
 11. G. E. P. Box and G. M. Jenkins, Time Series Analysis: Forecasting and Control, rev. ed., Holden-Day, San Francisco, CA (1976).
 12. B. Efron and D. V. Hinkley, Assessing the accuracy of the maximum likelihood estimator: observed versus expected Fisher information, Biometrika 65, 457-487 (1978).
 13. G. H. Golub and C. F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, MD (1983).
 14. A. Graham, Kronecker Products and Matrix Calculus: with Applications, Ellis Horwood, Chichester, UK (1981).
 15. S. J. Hammarling, Numerical solution of the stable, non-negative definite Lyapunov equation, IMA Journal of Numerical Analysis 2, 303-323 (1982).
 16. E. J. Hannan, Multiple Time Series, Wiley, New York, NY (1970).
 17. E. J. Hannan, The statistical theory of linear systems, pp. 83-121 in Developments in Statistics, Vol. 2, ed. by P. R. Krishnaiah, Academic Press, Orlando, FL (1979).
 18. L. P. Hansen and T. J. Sargent, Formulating and estimating dynamic linear rational expectations models, Journal of Economic Dynamics and Control 2, 7-46 (1980).
 19. L. P. Hansen and T. J. Sargent, Linear rational expectations models for dynamically interrelated variables, pp. 127-156 in Rational Expectations and Econometric Practice, Vol. 1, ed. by Robert E. Lucas Jr. and Thomas J. Sargent, University of Minnesota Press, Minneapolis, MN (1981).
 20. A. C. Harvey and C. R. McKenzie, Missing observations in dynamic econometric models: a partial synthesis, pp. 108-133 in Time Series Analysis of Irregularly Observed Data, ed. by E. Parzen, Springer-Verlag, New York, NY (1984).

21. A. C. Harvey and G. D. A. Phillips, Maximum likelihood estimation of regression models with autoregressive moving-average disturbances, Biometrika 66, 49-58 (1979).
22. A. C. Harvey and J. H. Stock, The estimation of higher-order continuous-time autoregressive models, Econometric Theory 1, 97-112 (1985).
23. S. Hillmer and G. C. Tiao, Likelihood function of stationary multiple autoregressive moving-average models, Journal of the American Statistical Association 74, 652-660 (1979).
24. R. H. Jones, Maximum likelihood fitting of ARMA models to time series with missing observations, Technometrics 22, 389-395 (1980).
25. G. G. Judge, W. E. Griffiths, R. C. Hill, and T. C. Lee, The Theory and Practice of Econometrics, Wiley, New York, NY (1980).
26. R. L. Kashyap, Maximum likelihood identification of stochastic linear systems, IEEE Transactions on Automatic Control AC-15, 25-34 (1970).
27. W. J. Kennedy Jr. and J. E. Gentle, Statistical Computing, Marcel Dekker, New York, NY (1980).
28. R. Kohn and C. F. Ansley, A note on obtaining the theoretical autocovariances of an ARMA process, Journal of Statistical Computation and Simulation 15, 273-283 (1982).
29. R. Kohn and C. F. Ansley, Estimation, prediction, and interpolation for ARMA models with missing data, Journal of the American Statistical Association 81, 751-761 (1986).
30. H. Kwakernaak and R. Sivan, Linear Optimal Control Systems, Wiley-Interscience, New York, NY (1972).
31. J. R. Magnus and H. Neudecker, Symmetry, 0-1 matrices, and Jacobians, Econometric Theory 2, 157-189 (1986).
32. J. R. Magnus and H. Neudecker, Matrix Differential Calculus with Applications in Statistics and Econometrics, Wiley, New York, NY (1988).
33. G. Melard, Exact derivatives of the likelihood of ARMA processes, presented at the annual meeting of the American Statistical Association, Las Vegas, NV (1985).
34. M. Morf and T. J. Kailath, Square-root algorithms for least-squares estimation, IEEE Transactions on Automatic Control AC-20, 487-497 (1975).

35. M. Morf, G. S. Sidhu, and T. J. Kailath, Some new algorithms for recursive estimation in constant, linear, discrete-time systems, IEEE Transactions on Automatic Control AC-19, 315-323 (1974).
36. H. Neudecker, Some theorems on matrix differentiation with special reference to Kronecker matrix products, Journal of the American Statistical Association 64, 953-963 (1969).
37. B. Porat and B. Friedlander, Computation of the exact information matrix of Gaussian time series with stationary random components, IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-34, 118-130 (1986).
38. G. Reinsel, FIML estimation of the dynamic simultaneous equations model with ARMA disturbances, Journal of Econometrics 9, 263-281 (1979).
39. B. Rosenberg, Random coefficient models: the analysis of a cross section of time series by stochastically convergent parameter regression, Annals of Economic and Social Measurement 2, 399-428 (1973).
40. S. D. Silvey, Statistical Inference, Chapman and Hall, London, UK (1975).
41. P. D. Tuan, Exact maximum likelihood estimate and Lagrange multiplier test statistic for ARMA models, Journal of Time Series Analysis 8, 61-78 (1987).
42. G. Tunnicliffe-Wilson, The estimation of parameters in multivariate time series models, Journal of the Royal Statistical Society, Series B, 35, 76-85 (1973).
43. M. Verhaegen and P. Van Dooren, Numerical aspects of different Kalman filter implementations, IEEE Transactions on Automatic Control AC-31, 907-917 (1986).
44. E. L. Wachpress, Iterative solution of the Lyapunov matrix equation, Applied Mathematics Letters 1, 87-90 (1988).
45. D. A. Wilson and A. Kumar, Derivative computations for the log-likelihood function, IEEE Transactions on Automatic Control AC-27, 230-232 (1982).
46. P. A. Zadrozny, Recursive analytic derivative formulas for the conditional Gaussian likelihood of vector ARMAX models, Discussion Paper, Center for Economic Studies, U.S. Bureau of the Census, Washington, DC (1984).
47. P. A. Zadrozny, Interpolation of stock and flow data observed at different frequencies, pp. 160-164 in Proceedings of the Business and Economic Statistics Section, American Statistical Association, Alexandria, VA (1986).

48. P. A. Zadrozny, Analytic derivatives for estimation of discrete-time, linear-quadratic dynamic optimization models, Econometrica 56, 467-472 (1988).
49. P. A. Zadrozny, Gaussian likelihood of continuous-time ARMAX models when data are stocks and flows at different frequencies, Econometric Theory 4, 108-124 (1988).