STRIDE for Structural Identification Using Expectation Maximization: Iterative Output-Only Method for Modal Identification

Thomas J. Matarazzo, S.M.ASCE¹; and Shamim N. Pakzad, A.M.ASCE²

Abstract: This paper introduces structural identification using expectation maximization (STRIDE), a novel application of the expectation maximization (EM) algorithm and approach for output-only modal identification. The EM algorithm can be used to estimate the maximum likelihood parameters of a state-space model. In this context, the state-space model represents the equation of motion for a linear dynamic system. STRIDE is an iterative procedure that uses Kalman filtering and Rauch-Tung-Striebel (RTS) smoothing equations to produce estimates of the unobserved states; these calculations are based on the observed data and prior estimates of the state-space parameters. With this information, the conditional likelihood of the model is maximized and the state-space parameters are updated at each iteration. Once an iteration meets user-prescribed convergence criterion, the algorithm ends—yielding maximum likelihood estimates (MLE) for the state-space model parameters. The modal properties of the structure are then extracted from these MLE. The performance of STRIDE is compared in detail with eigenvalue realization algorithm-natural excitation technique (ERA-NExT) and eigenvalue realization algorithm-observer Kalman filter identification of output-only systems (ERA-OKID-OO) identification algorithms in the analyses of ambient vibration data from the Northampton Street Bridge and Golden Gate Bridge, both collected using a dense wireless sensor network. A computational comparison shows that STRIDE provides a successful identification at a significantly lower model order than ERA-NExT, ERA-OKID-OO, or auto-regressive (AR), simultaneously requiring fewer cumulative floating point operations than ERA-OKID-OO in both applications.

DOI: 10.1061/(ASCE)EM.1943-7889.0000951. © 2016 American Society of Civil Engineers.

Author keywords: Modal identification; Maximum likelihood; Structural health monitoring; Signal processing.

Introduction

This paper introduces the expectation maximization (EM) algorithm to be used for structural modal identification, together known as structural identification using expectation maximization (STRIDE). The goal of EM is to determine the maximum likelihood estimates (MLE) of parameters in a specified model by alternating between two general steps: the expectation step (E-step) and the maximization step (M-step). It was first proposed in Dempster et al. (1977) as a general iterative approach to calculate MLE when the observations under consideration represent incomplete data. In these cases, the complete data of interest are not observed but can be mapped indirectly from observed incomplete data. For the case of this paper and in the context of a state-space model, the incomplete data are the observations \( y_k \) and the complete data are the states \( x_k \).

The EM algorithm has remained popular in likelihood applications due to its ease of implementation and general form. Work by Harvey and Pierse (1984), Phillips and Harvey (1979), and Jones (1980) introduced the role of Kalman filter and smoothing equations (Kalman 1960) for forecasting and identification of autoregressive moving average (ARMA) parameters using maximum likelihood estimation. Shumway and Stoffer (1982) and Digalakis et al. (1993) used these Kalman filter and smoothing equations within the E-step of EM to form specific parameter-updating equations (M-step) for the stochastic state-space model. The equations that update model parameters and maximize the conditional log-likelihood of the model are obtained by equating appropriate partial derivatives to zero (to be shown in the M-step section).

While the EM algorithm has proven to be at least initially more attractive than gradient-based search methods or Newton-Raphson techniques because EM does not require partial derivative calculations or solutions to nonlinear equations, its drawbacks are well documented and are explained fully in Dempster et al. (1977) and Wu (1983). For brevity, some main points summarized in Wu (1983) are rehearsed here:

**EM pros:**
- The conditional likelihood converges monotonically from below to some maximum value of the sequence. In general, if this likelihood has several maxima and stationary points, the choice of starting point determines which type of point the EM sequence will converge to; and
- Computations are relatively easy to program, if not readily available.

**EM cons:**
- Initial parameter estimates are required. The convergence of an EM sequence to a local or global maxima or stationary point depends on this choice;
- The algorithm tends to converge slowly in later iterations; and
- Matrices of second partial derivatives are not calculated by default; thus standard errors of parameter estimates are not immediately available; however, Matarazzo and Pakzad (2015) have recently derived the necessary equations.

¹Dept. of Civil and Environmental Engineering, Lehigh Univ., 117 ATLSS Dr., IMBT Labs, Bethlehem, PA 18015 (corresponding author). E-mail: thomas.matarazzo@gmail.com

²Associate Professor, Dept. of Civil and Environmental Engineering, Lehigh Univ., 117 ATLSS Dr., IMBT Labs, Bethlehem, PA 18015. E-mail: pakzad@lehigh.edu

Note. This manuscript was submitted on April 21, 2014; approved on February 25, 2015; published online on January 4, 2016. Discussion period open until June 4, 2016; separate discussions must be submitted for individual papers. This paper is part of the Journal of Engineering Mechanics, © ASCE, ISSN 0733-9399/04015109(19)/$25.00.


J. Eng. Mech., 04015109
The original contributions of this paper include the following:

1. Use maximum likelihood time-domain state-space model parameters to estimate structural modal properties, i.e., frequencies, damping, and mode shapes, of two real-life bridge structures in ambient vibration conditions;

2. Divide the initial EM estimation of the six model parameters into two groups due to their separate nature. Incorporate a variety of existing output-only system identification (SID) algorithms for initial estimates of the state and observation matrices $A_0$ and $C_0$ and provide guidelines for initialization of the remaining four parameters, which govern stochastic phenomena [see the section on initial parameter estimates and Eqs. (26a)–(26d)];

3. Recommendations on the selection of model order and defined likelihood function convergence criterion;

4. Two STRIDE applications using ambient vibration data collected from wireless sensor networks in which accuracy and computational efforts of STRIDE are compared with those of eigenvalue realization algorithm-natural excitation technique (ERA-NExT), eigenvalue realization algorithm-observer Kalman filter identification of output-only systems (ERA-OKID-OO), and auto-regressive (AR);

5. A detailed account of floating point operations (FLOPs) required in STRIDE; and

6. A complete derivation of the six parameter update equations required by the M-step, including an explanation of a nontrivial assumption embedded in the state input covariance and observation error covariance update formulas.

Additionally, STRIDE has many benefits when compared with other SID algorithms:

1. STRIDE is capable of providing accurate modal estimates at significantly lower model orders;

2. When considering multiple runs of an SID algorithm at increasing model orders (which is often required for a comprehensive modal identification), STRIDE can accurately identify modal properties using a comparable, sometimes smaller, number of cumulative FLOPs when compared to ERA-NExT, ERA-OKID-OO [Chang and Pakzad (2013) have shown that several other output-only SID methods are even more computationally expensive than these two], or AR;

3. Each iteration of STRIDE requires the same number of FLOPs as the previous, while ERA-NExT, ERA-OKID-OO, or AR requires increasingly more FLOPs as the model order increases. Consequently, iterations of STRIDE affect cumulative FLOPs linearly and model orders of ERA-NExT or ERA-OKID-OO affect cumulative FLOPs cubically;

4. A model selection procedure is not necessary. Assuming the user has defined adequate convergence criterion, the final iteration of STRIDE is designed to produce the most accurate modal estimates of all the iterations;

5. STRIDE iterates on a single model order $p$ to provide more stable, accurate damping ratio estimates than the SID algorithms requiring model order selection; and

6. EM has been a popular method to determine MLE for a few decades. Accordingly, many useful modifications are available: acceleration of EM via Aitken’s method (Dempster et al. 1977), Louis’s method (Louis 1982), or others (Meng and Rubin 1991; Jamshidian and Jennrich 1997), computation of Fisher information and Hessian matrices (Louis 1982; Matarazzo and Pakzad 2015), penalized likelihood (Green 1990), robust EM method for input-output time-domain (Gibson and Ninness 2005), etc. Krishnan (2008) provides a comprehensive volume of the EM algorithm and detailed explanations of its extensions and modifications.

The remainder of this paper is organized as follows. The next section provides the stochastic state-space model and likelihood function for a linear dynamic system. The following section introduces STRIDE, providing all necessary equations, the convergence criterion, initial parameter choices, and model order selection. The next two sections provide real-world STRIDE applications using ambient vibration data collected at the Northampton Street Bridge and Golden Gate Bridge, respectively. The performance and efficiency of STRIDE is compared to those of common output-only SID techniques. The authors’ conclusions are given in the following section. The references follow. The Appendix provides a detailed account of the FLOPs required by STRIDE as well as derivations for the M-step formulas presented.

**Likelihood of a Linear Dynamic System**

This section provides the stochastic state-space model and its likelihood function to be used for output-only structural modal identification of a linear dynamic system. The statistical approach of determining MLE for modal identification is not a new concept. Guillaume et al. (1998, 1999), Vanlanduit et al. (2003), and Verbøven et al. (2002) used frequency-domain maximum likelihood (ML) approaches to estimate polynomial coefficients of the frequency response functions. In Guillaume et al. (1999), this method was also derived to work for output-only data and successfully identified frequencies and damping ratios along corresponding standard deviations for a car-like subframe example. In Verbøven et al. (2002), this same method provided a full modal analysis for a slab track, detecting changes in modal parameters (mode-tracking) due to controlled structural changes.

Recently, the EM framework in Shumway and Stoffer (1982) has been adapted for structural modal identification by Pridham and Wilson (2004) and Cara et al. (2012) as well as in this paper. Despite the shared approach, there is a crucial methodological difference among these, which, in fact, occurs prior to the implementation of EM, namely, in the initial estimation of the six model parameters in the stochastic state-space model. Pridham and Wilson (2004) presented expectation maximization-stochastic subspace identification (EM-SSI) as a combined subspace-EM formulation, relying on the stochastic subspace identification (SSI) method for estimation of all six parameters, then Shumway and Stoffer’s EM for refinement. In addition to supporting an initial SSI, Cara et al. (2012) developed a random initialization procedure for initializing all six parameters simultaneously. STRIDE, on the contrary, uses two independent approaches to initialize the six model parameters, separating initial estimates of the two system dynamic matrices from those of aleatory entities due to their different nature. STRIDE initialization is detailed in the subsection preceding the bridge applications.

While successful, the methods presented in the literature neither clearly demonstrate their suitability for comprehensive modal identification, i.e., frequency, damping, and mode shape estimation, of real, large-scale civil structures, nor their advantages over well-established SID. Throughout this paper, the performance and computational requirements of STRIDE are compared directly to those of such SID.

**Stochastic State-Space Formulation**

The state-space formulation provides a practical approach to modeling a linear time-invariant dynamic system of interest as well as the measurement process. The observation Eq. (1) describes how discrete-time field measurements relate to the states of the...
governing dynamic system and the state Eq. (2) describes how these states behave over time. More specifically, in a structural health monitoring (SHM) context, the linear time-invariant dynamic system is a structural system, the observations may be accelerometer data, for which the states are the displacement and velocity responses of the structure at the sensing locations. For mathematical notation throughout this paper, matrices and scalars are italic, vectors are bold, and time step indices are subscripts:

\[ \mathbf{x}_k = A \mathbf{x}_{k-1} + \mathbf{u}_k, \quad k = 2, 3, \ldots, K \]  
\[ \mathbf{y}_k = C \mathbf{x}_k + \mathbf{v}_k, \quad k = 1, 2, \ldots, K \]  

where

\[ \mathbf{x}_k = \begin{bmatrix} \mathbf{u}_k \\ \mathbf{\dot{u}}_k \end{bmatrix} \]  
\[ \mathbf{y}_k = \mathbf{\dot{u}}_k \]  

and \( \mathbf{u}_k \) = sampled displacement vector at time step \( k \), \( \mathbf{\dot{u}}_k \) = sampled velocity vector at time step \( k \), and \( \mathbf{\dot{u}}_k \) = sampled acceleration vector at time step \( k \).

The observed accelerations at the \( N \) sensing locations are represented by the \( N \times 1 \) vector \( \mathbf{y}_k \) at time step \( k \). For a state-space model of order \( p \), these observations are mapped to the \( pN \times 1 \) unobserved stochastic state vector \( \mathbf{x}_k \) using the \( N \times pN \) observation matrix \( C \). Note \( p \) represents the number of time lags included in the time-series model (Box et al. 2008; Brockwell and Davis 2002); it is an integer that relates the observation size to the state size. The theoretical formulation of the state-space model defines the minimum model order as 2. The unobserved states are a quantity of interest because their behavior over time dictates the \( pN \times pN \) state matrix \( A \). The \( N \times 1 \) observation error/noise terms \( \mathbf{v}_k \) are assumed to be zero-mean, uncorrelated Gaussian vectors with \( N \times N \) covariance matrix \( \mathbf{R} \); this is a common statistical assumption for this type of time-series model (Chang and Pakzad 2013; Digalakis et al. 1993; Jones 1980; Shumway and Stoffer 1982). The \( pN \times 1 \) vector of state input \( \mathbf{u}_k \) represents the dynamic loading of the structure. This dynamic loading (or system input) is not often measured in practice. In SHM, this is referred to as an output-only system; that is, only the output of the system is measured and the input is assumed to be a random process. Here, it is assumed that the state loading terms are zero-mean, uncorrelated Gaussian vectors with \( pN \times pN \) covariance matrix \( \mathbf{Q} \), a common statistical assumption for output-only identification (Chang and Pakzad 2013; Huang 2001; Peeters and De Roeck 1999). Finally, the initial state \( \mathbf{x}_1 \) is assumed to be a Gaussian vector with mean vector \( \mathbf{\mu} \) and \( pN \times pN \) covariance matrix \( \mathbf{V} \); the initial state and covariance matrix are described for the first time step \( (k=1) \):

\[ \mathbf{x}_1 \sim N(\mathbf{\mu}, \mathbf{V}) \]  
\[ \mathbf{u}_1 \sim N(0, \mathbf{Q}) \]  
\[ \mathbf{v}_1 \sim N(0, \mathbf{R}) \]  

**Log-Likelihood Function of a State-Space Model**

One approach to validate or gain insight on a given set of state-space model parameters is to investigate the likelihood of these parameters given the observed data. For convenience of reference, all six state-space parameters are collectively referred to as the superparameter \( \Psi \):

\[ \Psi = (\mathbf{\mu}, \mathbf{V}, A, Q, C, R) \]  

This superparameter is similar to \( \Theta \) found in Chapter 6 of Shumway and Stoffer (2011), except the observation matrix \( C \) is also included here. The log-likelihood function for the state-space model can be written directly in terms of the superparameter, the observations, and the states. The function will be denoted by \( \ln[L_{X,Y}(\Psi)] \)—the subscripts \( X \) and \( Y \) indicate that this is a complete-data likelihood function because it depends on the observations and the states:

\[ \ln(L_{X,Y}(\Psi)) = -\frac{pN}{2}\ln(2\pi) - \frac{1}{2} \ln |V| \]

\[ -\frac{1}{2} (\mathbf{x}_1 - \mathbf{\mu})^t \mathbf{V}^{-1} (\mathbf{x}_1 - \mathbf{\mu}) - \frac{KN}{2} \ln(2\pi) \]

\[ -\frac{K}{2} \ln |R| - \frac{1}{2} \sum_{k=1}^{K} (\mathbf{y}_k - C\mathbf{x}_k)^t \mathbf{R}^{-1} (\mathbf{y}_k - C\mathbf{x}_k) \]

\[ -\frac{K-1}{2} \ln(2\pi) - \frac{K-1}{2} \ln |Q| \]

\[ -\frac{1}{2} \sum_{k=2}^{K} (\mathbf{x}_k - A\mathbf{x}_{k-1})^t \mathbf{Q}^{-1} (\mathbf{x}_k - A\mathbf{x}_{k-1}) \]  

The overall goal is to solve for the superparameter that maximizes this likelihood Eq. (6), thus yielding MLE for the model. Eq. (6) implies that state measurements are available; however, this is not the case. Thus, the equation cannot be used directly. The whole purpose of the state-space Eqs. (1) and (2) is to separate the observations from the states because the states cannot be directly observed [this is referred to as incomplete data versus complete data to agree with the terminology of Dempster et al. (1977) and Shumway and Stoffer (2011)]. In short, this issue is remedied using the Kalman filter and RTS smoother equations in the E-step. The EM algorithm maximizes the *conditional expectation of the complete data likelihood* \( G(\Psi_{j+1} | \Psi_j) \) for each iteration \( j \):

\[ G(\Psi_{j+1} | \Psi_j) = E[\ln(L_{X,Y}(\Psi_{j+1})) | \mathbf{y}_1, \ldots, \mathbf{y}_K, \Psi_j] \]  

Eq. (7) introduces the conditional expectation of Eq. (6), which will be used in place of the actual log-likelihood function of Eq. (6). From a computational perspective, it is more convenient to utilize Eq. (7) in its innovations form, i.e., directly in terms of the prediction error \( \mathbf{\epsilon}_k \) and its covariance \( \Sigma_k \), which both depend on the superparameter \( \Psi \):

\[ \ln[L_Y(\Psi)] = -\frac{KN}{2} \ln(2\pi) - \frac{K}{2} \ln |\Sigma_k(\Psi)| \]

\[ -\frac{1}{2} \sum_{k=1}^{K} \mathbf{\epsilon}_k(\Psi)^t \Sigma_k(\Psi)^{-1} \mathbf{\epsilon}_k(\Psi) \]

Eqs. (7) and (8) are similar to those found in Chapter 6 of Shumway and Stoffer (2011) except that these presented here include the often ignored constant terms.

**STRIDE**

The purpose of this section is to introduce the STRIDE algorithm, given the stochastic state-space model with incomplete data. The iterative procedure of the general EM algorithm, which is primarily comprised of the E-step and the M-step, was introduced in Dempster et al. (1977). The following procedure is presented in a manner similar to Shumway and Stoffer (2011), where it is described to alternate between the Kalman filtering recursive Eqs. (10a)–(11f), RTS smoothing recursive Eqs. (12a)–(13c), and the multivariate normal maximum likelihood estimators.
1. Initialize: Select superparameter \( \Psi_0 \) for \( j = 0 \). Begin iterations \( j = 0, 1, \ldots \);
2. E-step: Use the Kalman filter (Kalman 1960) and Rauch-Tung-Striebel (RTS) smoother (Rauch et al. 1965) equations to estimate the states and covariances, namely, \( \hat{x}_j \) and \( \hat{V}_{j,k}, \hat{V}_{j,k+1} \) for all time steps. Next, use these results with the observations and superparameter from iteration \( j \) to compute the superparameter estimate for the next iteration in Eqs. (14)–(19);
3. Compute the conditional expectation of the likelihood function. From a computational perspective, the prediction error/innovations form in Eq. (8) is recommended;
4. M-step: Update the superparameter estimate \( \Psi_j \rightarrow \Psi_{j+1} \) in Eq. (20); and
5. Repeat Steps 2 through 4 until the convergence criterion of Eq. (24) is met. The final \( \Psi_{Mj} \) provides MLE for the system.

The goals for STRIDE are different from typical solutions to MLE problems. For the purposes of SID, not all of the MLE state-space parameters are necessary; only two matrices within the superparameter are needed for modal identification. The identification is based on the observation matrix \( C \) and the eigendecomposition of the state matrix \( A \); thus, only the MLE of these two matrices are required to estimate the natural frequencies, mode shapes, and damping ratios of the structural system. The general equations for the E-step and M-step can be found in the literature; however, all calculations in STRIDE are presented within this paper to clearly display the required computational efforts.

**E-Step—Recursive Kalman Filter and RTS Smoother Equations**

In this section, the Kalman filter and RTS smoother equations are provided, constituting the E-step of the EM algorithm. The goal of the E-step is to estimate the states and their covariances using the observed data and the superparameter of the current iteration \( j \); subsequently, Eq. (7) can be computed.

In the following equations, the estimated states and their covariances will be denoted with hats. Additionally, the estimated states, state covariances, and prediction error covariances will have multiple subscripts: \( \hat{x}_{a|\tau}, \hat{V}_{a|\tau} \), and \( \hat{V}_{a|\tau} \). The vertical bar within the subscripts is intended to be consistent with conditional probability notation, e.g., \( \alpha|\tau \) means alpha given tau. More specifically, the subscript left of the vertical bar \( (\alpha) \) for the estimated states indicates the time step at which a quantity is estimated. The subscript right of the vertical bar \( (\tau) \) specifies the time series information that is known (given) at the time of estimation, i.e., specifically the observations \( y_1, \ldots, y_\tau \). The state variance \( \hat{V}_{a|\alpha|\tau} \) and delayed-state filter covariance \( \hat{V}_{a|\alpha|\tau} \) each have two subscripts left of the vertical bar \( (\alpha) \) and \( (\beta) \), which indicate the state time steps under consideration. Consequently, the values of these subscripts dictate whether each equation represents smoothing, filtering, or forecasting/prediction, as they specify information available at the time of estimation. The filtered and smoothed state and state covariance estimates are defined as follows in Eqs. (9a)–(9c):

\[
\hat{x}_{j|k} = E[x_k | y_1, \ldots, y_k] \quad (9a)
\]

\[
\hat{V}_{j,k|k} = E[(x_k - \hat{x}_{j|k})(x_k - \hat{x}_{j|k})^T | y_1, \ldots, y_k] \quad (9b)
\]

\[
\hat{V}_{j,k-1|k} = E[(x_k - \hat{x}_{j|k})(x_{k-1} - \hat{x}_{j-1|k})^T | y_1, \ldots, y_k] \quad (9c)
\]

The following recursive equations originate from Kalman (1960) and have been presented in terms of the state-space model throughout the literature [e.g., Box et al. (2008) or Shumway and Stoffer (1982)]. Basically, the Kalman equations complete the state-space model data. The Kalman filter equations use the Kalman gain matrix \( K_k \) to compute minimum mean square error (MMSE) estimates of the states and state covariances from the observations and current superparameter. With these quantities available, the prediction error \( \hat{\varepsilon}_k \) is defined as an \( N \times 1 \) zero-mean Gaussian random vector with covariance \( \Sigma_k \). This prediction error and its covariance can be used to compute the innovations form of the conditional likelihood introduced in Eq. (8). The forward recursive equations are computed for \( k = 1, 2, \ldots, K \).

With prediction equations:

\[
\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} \quad (10a)
\]

\[
\hat{V}_{k,k|k-1} = A\hat{V}_{k-1,k-1|k-1}A^T + Q \quad (10b)
\]

and filter/update equations:

\[
\hat{\varepsilon}_k = y_k - CS_k \hat{x}_{j|k-1} \quad (11a)
\]

\[
\Sigma_k = C\hat{V}_{k,k|k-1}C^T + R \quad (11b)
\]

\[
K_k = \hat{V}_{k,k|k-1}C\Sigma_k^{-1} \quad (11c)
\]

\[
\hat{x}_{j|k} = \hat{x}_{j|k-1} + K_k\hat{\varepsilon}_k \quad (11d)
\]

\[
\hat{V}_{k,k|k} = (I - K_kC)\hat{V}_{k,k|k-1} \quad (11e)
\]

The fixed-interval smoothing, RTS smoother equations introduced in Rauch et al. (1965) [embedded within the EM framework of Digalakis et al. (1993) and Shumway and Stoffer (1982)] are used to produce MMSE estimates of the states and state covariances given the observations, current superparameter, and Kalman filter results, further reducing the estimated variances of the states. These smoothed estimates of the states and state covariances use the smoother gain matrix \( J_k \) to obtain the optimal linear combination of the forward and backward filter estimates (Crassidis and Junkins 2011). The smoother estimates will be denoted in the same manner as those from the Kalman filter. The backward recursive equations are computed from \( k = K, K - 1, \ldots, 2 \):

\[
J_{k-1} = \hat{V}_{k-1,k-1|k-1}A^T\hat{V}_{k,k|k-1}^{-1} \quad (12a)
\]

\[
\hat{x}_{k-1|k} = \hat{x}_{k-1|k-1} + J_{k-1}(\hat{x}_{k|k} - \hat{x}_{k-1|k}) \quad (12b)
\]

\[
\hat{V}_{k-1,k-1|k} = \hat{V}_{k-1,k-1|k-1} + J_{k-1}(\hat{V}_{k,k} - \hat{V}_{k,k|k-1})J_{k-1}^T \quad (12c)
\]

The delayed-state smoother covariance is computed in Eq. (13c); its initial value, shown as follows in Eq. (13a), is actually computed in the delayed-state filter Eq. (11f) during the final Kalman filter run \( (k = K) \):

\[
\hat{V}_{K,k-1|K} = (I - K_kC)\hat{V}_{K-1,k-1|K} \quad (13a)
\]

There are two popular methods for calculating the delayed-state smoother covariance for the remaining time steps: \( K - 1, K - 2, \ldots, 1 \). Shumway and Stoffer (1981) first derived Eq. (13b) based on the preceding delayed-state smoother covariance and the filtered state covariance Eq. (11e), as well as two smoother gains Eq. (12a). Shumway and Stoffer (2011) refer to the quantity in Eq. (13b) as the lag-one covariance smoother. Note that Eq. (13b)
requires at least temporary storage of the following step’s smoother gain $J_{k-2}$:

$$\hat{V}_{k-1,k-2|k} = \hat{V}_{k-1,k-1|k-1} J_{k-2}^T + J_{k-1} (\hat{V}_{k-1,k-1|k} - \Lambda \hat{V}_{k-1,k-1|k-1}) J_{k-2}^T$$

(13b)

Digalakis et al. (1993) derived Eq. (13c) which is a function of the delayed-state filter covariance Eq. (11f), the filtered state covariance Eq. (11e), and the smoothed state covariance Eq. (12c). Eq. (13c) requires less information from other smoothing steps and is preferable over Eq. (13b) for this reason:

$$\hat{V}_{k-1,k-2|k} = \hat{V}_{k-1,k-1|k-1} + \left( \hat{V}_{k-1,k-1|k} - \hat{V}_{k-1,k-1|k-1} \right) \hat{V}_{k-1,k-1|k-1}^{-1} \hat{V}_{k-1,k-2|k-1}$$

(13c)

**M-Step—Maximize Conditional Expectation of Likelihood Function**

Once the states $\hat{x}_{k|k}$ and their covariances $\hat{V}_{k,k|k}$ and $\hat{V}_{k-1,k-1}$ have been determined (predicted, filtered, then smoothed) for all time steps, state-space parameters can be calculated for the next iteration ($j + 1$). The parameter updating equations are obtained from taking partial derivatives of the conditional log-likelihood Eq. (7), setting these equal to zero (Ghahramani and Hinton 1996), and using the superparameter at the current iteration ($j$). A detailed derivation of these update equations is provided in the Appendix. The following equations represent the iteration transition, namely $j \rightarrow j + 1$:

$$A_{j+1} = \sum_{k=2}^{K} [\hat{x}_{k|k} \hat{x}_{k-1|k}^T]$$

$$+ V_{k-1,k|k} \left[ \sum_{k=2}^{K} (\hat{x}_{k-1|k} \hat{x}_{k-1|k}^T + \hat{V}_{k-1,k-1|k}) \right]^{-1}$$

$$C_{j+1} = \left( \sum_{k=1}^{K} y_{j,k} \hat{x}_{k|k}^T \right) \left( \sum_{k=1}^{K} (\hat{x}_{k|k} \hat{x}_{k|k}^T + \hat{V}_{k,k|k}) \right)^{-1}$$

$$Q_{j+1} = \frac{1}{K-1} \sum_{k=2}^{K} (\hat{x}_{k|k} \hat{x}_{k|k}^T + \hat{V}_{k,k|k})$$

$$- A_{j+1} \sum_{k=2}^{K} (\hat{x}_{k-1|k} \hat{x}_{k-1|k}^T + \hat{V}_{k-1,k-1|k})$$

$$R_{j+1} = \frac{1}{K} \sum_{k=1}^{K} y_{j,k} y_{j,k}^T - C_{j+1} \sum_{k=1}^{K} \hat{x}_{k|k} y_{j,k}^T$$

The updates for initial state and state covariance (smoothed estimates at $k = 1$) are determined directly from the filtered and smoothed states and state covariances:

$$\hat{\mu}_{j+1} = \hat{x}_{1|1}$$

$$\hat{V}_{j+1} = \hat{V}_{1,1|1}$$

Finally, to be consistent with the model assumptions, the user must ensure that the covariance matrices $Q_{j+1}$ and $R_{j+1}$ are diagonal matrices. The superparameter is now updated, ready for the next iteration, namely $j + 1$:

$$\Psi_{j+1} = (\hat{\mu}_{j+1}, \hat{V}_{j+1}, A_{j+1}, Q_{j+1}, C_{j+1}, R_{j+1})$$

**MLE and Corresponding Modal Properties**

In this section, the state-space parameters $A$ and $C$ are used to determine the structural modal properties: natural frequencies, mode shapes, and damping ratios. Once the MLE of the superparameter $\Psi_{ML}$ has been determined at the final iteration, the corresponding modal properties can be calculated from $A_{ML}$ and $C_{ML}$. Let $\Lambda_{ML}$ and $\Gamma_{ML}$ represent the matrices of eigenvalues and eigenvectors, respectively, of $A_{ML}$. The diagonal terms of the eigenvalue matrix dictate the $pn \times n$ maximum likelihood frequency estimators $F_{ML}$ and the $pn \times 1$ maximum likelihood damping ratio estimators $\zeta_{ML}$. The eigenvector and the observation matrices are used to compute the $pn \times pn$ maximum likelihood mode shape matrix estimator $\Phi_{ML}$:

$$F_{ML} = 2\pi \frac{\ln(\text{diag}(\Lambda_{ML}))}{\Delta t}$$

(21a)

$$\Phi_{ML} = C_{ML} \Gamma_{ML}$$

(21b)

$$\zeta_{ML} = -\cos\{\Delta n(\text{diag}(\Lambda_{ML}))\} \cdot 100\%$$

(21c)

In Eqs. (21a) and (21c), if an eigenvalue is complex, e.g., $z = a + b \cdot i$, where $i$ is the imaginary number, the natural logarithm is computed as $\ln(z) = \ln|z| + i \cdot \text{atan}(b/a)$ where $-\pi \leq \text{atan}(b/a) \leq \pi$. In (21c), $\Delta n(\Lambda_{ML})$ is the phase angle of an eigenvalue in radians. As the STRIDE iterations progress, the modal estimates stabilize and exhibit convergent behavior. Similar to the likelihood function values, estimates update rapidly in earlier iterations and more slowly at higher iterations.

**STRIDE Performance for a Four-Degree-of-Freedom Simply Supported Beam**

The challenge of verifying modal estimates in real structures has motivated the analysis of a system with known modal properties. The example originally presented in Chang and Pakzad (2013) is analyzed by STRIDE to illustrate accuracy of its ML modal estimates with respect to iterations and known modal properties. Through comparing the performances of several SID algorithms, Chang and Pakzad (2013) have shown that damping ratio biases vary by model order. The STRIDE frequency $F_n$ and damping $\zeta_n$ errors defined as follows in Eqs. (22a) and (22b) are shown in Fig. 1. For modes two through four, stable, accurate damping estimates are available after about 50 iterations. The damping for the first mode fell under 2% error after 100 iterations and stabilized near 300. Overall, the STRIDE damping estimates are more stable and accurate with respect to iterations than those of ERA-OKID-OO, ERA-NExT-AVG (modified ERA-NExT), N4SID (numerical algorithm for subspace state-space system identification), or AR are with respect to model order as presented in Fig. 3 of Chang and Pakzad (2013):

$$e_{F_n} = 1 - \frac{F_n_{ML}}{F_n_{exact}}$$

(22a)

$$e_{\zeta_n} = 1 - \frac{\zeta_n_{ML}}{\zeta_n_{exact}}$$

(22b)

**Convergence Criterion for STRIDE**

A good indicator of STRIDE convergence is the slope of the conditional likelihood function. The actual slope $\partial G/\partial j$ is estimated in Eq. (23), using the difference and average of the two most recent likelihood values (Murphy 2004; Press et al. 1992):
Fig. 1. STRIDE frequency and damping performance for the 4DOF beam structure presented in Chang and Pakzad (2013)

\[
\frac{\partial G}{\partial j} \approx 2\frac{G(\Psi_j) - G(\Psi_{j-1})}{|G(\Psi_j) + G(\Psi_{j-1})|} \quad (23)
\]

The MLE of the superparameter is determined once the conditional likelihood function reaches its maximum value. For implementation, the current superparameter (at iteration \( j \)) is assumed to be the MLE when the actual slope is near zero. The convergence criterion of STRIDE is user-defined by choosing a slope threshold \( \theta \); specifically, the convergence condition for STRIDE is defined in Eq. (24).

STRIDE has converged at iteration \( j \) if

\[
\frac{\partial G}{\partial j} \leq \theta \quad (24)
\]

An adequate slope threshold must be selected to ensure the conditional likelihood function has practically attained its maximum value. Based on application examples presented in the bridge sections, a slope threshold of \( \theta = 5 \times 10^{-4} \) provided accurate results and can be recommended as guideline for model orders \( p = 2 \) or 4. Identification is possible at a larger slope threshold (fewer iterations), but it may be beneficial to allow for more iterations when processing new data.

**Initial Parameter Estimates and Model Order Selection**

The EM algorithm requires an initial estimate of the superparameter to commence, because the E-step computes the conditional log-likelihood function in Eq. (6), i.e., the likelihood function given the observations and previous iteration’s superparameter. This section will discuss the selection of an initial super-parameter \( \Psi_0 \) and model order \( p \) for STRIDE. The initial superparameter is denoted with subscript 0 (for \( j = 0 \)):

\[\Psi_0 = (\bar{\mu}_0, \bar{V}_0, A_0, Q_0, C_0, R_0)\quad (25)\]

The goal of the initial superparameter is to provide an adequate starting point for EM without a priori knowledge of the structural system. In Eqs. (26a)–(26d), \( \bar{\mu}_0 \), \( \bar{V}_0 \), \( Q_0 \), and \( R_0 \) can be specified to represent the model assumptions described earlier:

\[\bar{\mu}_0 = 0_{pN \times 1} \quad (26a)\]

The state error/loading covariance matrix \( Q_0 \) is set to be the \( pN \times pN \) identity matrix. The initial state mean \( \bar{\mu}_0 \) is set to be a \( pN \times 1 \) vector of zeros. The initial state covariance matrix \( \bar{V}_0 \) can be assumed to have the same magnitude as the loading, therefore set to the \( pN \times pN \) identity matrix, but may also be determined using \( A_0 \) and \( Q_0 \) in the Lyapunov equation (Boots 1999; Lyapunov 1907). Finally, the observation error/noise covariance \( R_0 \) is assumed to be the \( N \times N \) identity matrix. The covariance matrices \( Q_0 \) and \( R_0 \) are set to be diagonal matrices.

The two system matrices, \( A_0 \) and \( C_0 \), are most crucial to the algorithm’s performance for the output-only case, because they directly affect the estimated modal properties. These matrices are estimated by using the observed data. For the purposes of this paper, another output-only SID algorithm is implemented to provide an initial estimate for these matrices. To guarantee compatibility with STRIDE, the chosen initial SID algorithm(s) must be capable of either directly or indirectly estimating \( A_0 \) and \( C_0 \) for the discrete-time domain stochastic state-space model given earlier, e.g., N4SID (Van Overschee and De Moor 1992), AR (He and De Roeck 1997), ERA-NExT (James et al. 1993), ERA-NExT-AVG (Chang and Pakzad 2012), ERA-OKID-OO (Chang and Pakzad 2013), SSI (Peeters and De Roeck 1999), or others. In subsequent sections, ERA-OKID-OO is selected to compute both \( A_0 \) and \( C_0 \) matrices.

For STRIDE, the model order choice is brief compared to other SID algorithms because, as shown subsequently, it can accurately estimate modal parameters at the minimum, and default, model order \( p = 2 \). A higher model order, e.g., \( p = 4 \), can be implemented to enhance the accuracy at a marginally higher computational cost. In general, a need for a higher model order may arise if few sensing locations are used in analysis, i.e., \( N \) is small, but many modal properties are desired. In this case, it is desirable to increase the size of the \( pN \times pN \) state matrix by using a larger \( p \), thus increasing the quantity of eigenvalues in \( \Lambda \), and estimated modes.
Northampton Street Bridge Application

Bridge, Sensor Setup, Data Collection, and Initial Parameter Estimates

The Northampton Street Bridge (NHB) is a steel cantilever bridge connecting Easton, Pennsylvania, and Phillipsburg, New Jersey, over the Delaware River. Its center span is 300 ft (91 m), its side spans are each 125 ft (38 m), and it carries three lanes of east/west car traffic and two sidewalks.

The NHB data consisted of 100,000 samples collected from 21 sensors and oversampled at 280 Hz to limit the effect of sensor noise. Eighteen sensors were placed on the south side of the bridge and three mirroring sensors were placed on the north side of the bridge as shown in Fig. 2; the sensor channels that measure acceleration perpendicular to the roadway of the bridge were considered in this analysis for the identification of vertical and torsional vibration modes.

$A_0$ and $C_0$ were calculated from ERA-OKID-OO ($p = 2$) using the measured data. Next, the same data were filtered and downsampled to consider modal properties up to 10 Hz and used in STRIDE ($p = 2$) with convergence threshold $\theta = 5 \times 10^{-4}$. The remaining state-space parameters were estimated in accordance with the previous section. STRIDE required 55 iterations to achieve the designated slope threshold.

ERA-NExT, ERA-OKID-OO, and AR were implemented for even model orders 2 through 100 [using structural modal identification toolsuite (SMIT) (Chang and Pakzad 2013)] producing stabilization diagrams (those of ERA-NExT and ERA-OKID-OO are shown in Figs. 3 and 4). These stabilization plots were used to select the model order, for each SID algorithm, that provided the most accurate modal estimates.

Results Comparison—Frequencies, Damping Ratios, and Mode Shapes

In this section, a detailed comparison of the modal identification results from STRIDE, ERA-NExT, and ERA-OKID-OO is presented. These two algorithms were chosen because they exhibit a useful balance of computational speed and accuracy (Chang and Pakzad 2013). Additionally, since ERA-OKID-OO was selected to provide $A_0$ and $C_0$ for STRIDE, a comparison was made to higher order ERA-OKID-OO models. STRIDE results are also briefly compared to those of AR with respect to computational FLOPs and overall performance (Figs. 7 and 8, and Table 2).

In general, modal estimates in latter STRIDE iterations are more accurate than the previous iterations. Consequently, assuming an adequate convergence criterion is selected, the user can effortlessly choose the results from the final iteration, and attain accurate modal estimates provided by the algorithm because the likelihood function is at its maximum value.

To obtain accurate modal estimates using the other SID methods, the user must review the stabilization diagram and select the model order providing the optimal estimates. There are some issues with this model selection procedure: the process can be time-consuming and it is often the case that a single model order does not represent the results of all computed model orders, i.e., it does not provide the best modal parameters.

The second point is illustrated in the NHB analysis. Using the stabilization diagram for ERA-NExT, Fig. 3, model order 36 was...
chosen to represent the method, despite missing mode 6 (Table 1). For example, in model order 80, mode 6 was identified but mode 7 was absent. Similarly for ERA-OKID-OO, model order 100 was chosen using Fig. 4 to represent the method, despite missing modes 7 and 10 (Table 1). At model order 44, this method identified modes 7 and 10, but modes 3 and 5 were not identified. Although all modes were identified throughout the range of model orders analyzed, there was no single model order (neither in ERA-NExT nor ERA-OKID-OO) that included modal estimates for all the modes. To obtain a comprehensive modal analysis using either method, the user must compile results from candidate model orders, which can be time-consuming.

Table 1. Summary of Identified Modes of NHB below 10 Hz

<table>
<thead>
<tr>
<th>Modal property</th>
<th>1 (V)</th>
<th>2 (T)</th>
<th>3 (V)</th>
<th>4 (V)</th>
<th>5 (V)</th>
<th>6 (V)</th>
<th>7 (T)</th>
<th>8 (V)</th>
<th>9 (T)</th>
<th>10 (T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>OO frequencies (Hz)</td>
<td>1.793</td>
<td>1.991</td>
<td>2.832</td>
<td>4.402</td>
<td>4.733</td>
<td>5.227</td>
<td>miss</td>
<td>8.960</td>
<td>9.383</td>
<td>miss</td>
</tr>
<tr>
<td>XT damping (%)</td>
<td>2.33</td>
<td>1.86</td>
<td>5.46</td>
<td>1.16</td>
<td>1.33</td>
<td>miss</td>
<td>9.59</td>
<td>0.825</td>
<td>1.23</td>
<td>1.75</td>
</tr>
<tr>
<td>OO damping (%)</td>
<td>2.56</td>
<td>0.89</td>
<td>5.75</td>
<td>1.17</td>
<td>1.05</td>
<td>1.23</td>
<td>miss</td>
<td>1.25</td>
<td>0.85</td>
<td>miss</td>
</tr>
<tr>
<td>ST damping (%)</td>
<td>5.52</td>
<td>2.92</td>
<td>5.87</td>
<td>2.87</td>
<td>3.30</td>
<td>4.56</td>
<td>5.58</td>
<td>0.92</td>
<td>1.96</td>
<td>3.61</td>
</tr>
<tr>
<td>ST and OO MAC</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
<td>0.91</td>
<td>0.84</td>
<td>miss \textsuperscript{OO}</td>
<td>0.89</td>
<td>0.30\textsuperscript{ST}</td>
<td>miss \textsuperscript{OO}</td>
</tr>
<tr>
<td>ST and XT MAC</td>
<td>0.98</td>
<td>0.98</td>
<td>0.99</td>
<td>0.97</td>
<td>0.92</td>
<td>miss \textsuperscript{XT}</td>
<td>0.95</td>
<td>0.96</td>
<td>0.25\textsuperscript{ST}</td>
<td>0.17\textsuperscript{ST}</td>
</tr>
<tr>
<td>OO and XT MAC</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>miss \textsuperscript{XT}</td>
<td>miss \textsuperscript{OO}</td>
<td>0.94</td>
<td>0.98</td>
<td>miss \textsuperscript{OO}</td>
</tr>
</tbody>
</table>

Note: Vertical and torsional modes are represented by V or T, respectively; ST, OO, and XT denote STRIDE, ERA-OKID-OO, and ERA-NExT, respectively; miss denotes a completely missed mode; MAC values are computed for all three shape combinations; Superscripts ST, OO, and XT denote respective mode shape misses.

Table 2. Summary of Computational Efforts and Performance for Modal Analysis of NHB Data Using Various SID Algorithms and STRIDE

<table>
<thead>
<tr>
<th>Method</th>
<th>Model order</th>
<th>Iterations</th>
<th>Cumulative estimated FLOPs</th>
<th>Identified modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERA-NExT</td>
<td>2 through 36</td>
<td>—</td>
<td>9.00 \times 10^{10}</td>
<td>9 out of 10 (missed 6)</td>
</tr>
<tr>
<td>ERA-OKID-OO</td>
<td>2 through 100</td>
<td>—</td>
<td>8.90 \times 10^{12}</td>
<td>8 out of 10 (missed 7 and 10)</td>
</tr>
<tr>
<td>AR</td>
<td>2 through 90</td>
<td>—</td>
<td>3.29 \times 10^{13}</td>
<td>6 out of 10 (missed 6, 7, 8, and 10)</td>
</tr>
<tr>
<td>STRIDE</td>
<td>2</td>
<td>55</td>
<td>6.80 \times 10^{11}</td>
<td>10 out of 10</td>
</tr>
</tbody>
</table>

For the NHB analysis, STRIDE’s initial parameter estimates were provided by ERA-OKID-OO ($p = 2$), which contained an inflated damping ratio. Similar in performance to the four-degree-of-freedom (4-DOF) beam example presented in Fig. 1, damping estimates began to stabilize after about 50 iterations. The STRIDE
NHB analysis achieved the slope threshold after 55 iterations; therefore, it is possible some damping estimates remained inflated as a result of the initial SID estimate.

**Computation Comparison**

With a slope threshold chosen, the corresponding number of iterations will vary mostly depending on number of samples $K$, number of sensors $N$, and model order $p$. With larger sample sizes and higher model orders, more iterations may be necessary to achieve a desired slope threshold, but, more importantly, each iteration requires more computations. Fig. 6 shows the number of iterations required to reach a variety of slope thresholds, at model orders $p = 2$ and 4 in the NHB analysis. In this figure, a rather sharp demand in iterations, and in turn computational efforts, is apparent near $\theta < 5 \times 10^{-4}$. Consequently, from a computational perspective, slope thresholds near $\theta = 5 \times 10^{-4}$ (or a bit larger) are efficient for the NHB data.

Fig. 7 shows the estimated FLOPs for ERA-NExT, ERA-OKID-OO, and STRIDE with the NHB data and Fig. 8 shows the cumulative estimated FLOPs. Fig. 7 illustrates how each iteration of STRIDE requires the same number of FLOPs as the previous, while ERA-NExT, ERA-OKID-OO, or AR requires increasing numbers.

**Fig. 5.** Six selected modes from identifications of NHB data; ERA-NExT ($p = 36$), ERA-OKID-OO ($p = 100$), and STRIDE ($p = 2$)

**Fig. 6.** Number of STRIDE iterations required for decreasing slope thresholds using NHB data ($N = 21$ sensors; $K = 7,143$ samples); model orders are coded with iteration counts at selected slopes

**Fig. 7.** Estimated FLOPs for NHB data ($N = 21$ sensors); ERA-OKID-OO, ERA-NExT, and AR are plotted against model order; STRIDE ($p = 2$) is plotted against iterations
of FLOPs as the model order increases. Consequently, iterations for STRIDE affect cumulative FLOPs linearly and model orders for ERA-NExT, ERA-OKID-OO, and AR affect cumulative FLOPs cubically. Table 2 summarizes the cumulative computational costs for these methods. ERA-NExT required the least cumulative FLOPs, STRIDE used just under one order of magnitude more cumulative FLOPs, while ERA-OKID-OO and AR required about two orders of magnitude more cumulative FLOPs.

Golden Gate Bridge Application

Bridge, Sensor Setup, Data Collection, and Initial Parameter Estimates

The Golden Gate Bridge (GGB) is a steel suspension bridge that connects San Francisco, California, and Marin County, California, over the opening of the Pacific Ocean. The total length of the bridge is 8,981 ft (2,737 m), with a center span of 4,200 ft (1,280 m, which was the largest in the world up to 1964). The bridge carries six lanes of north/south car traffic and two sidewalks.

The GGB data consisted of 80,000 samples oversampled from 10 sensors at 50 Hz; this is a subset of the data in Pakzad and Fenves (2009) and Pakzad et al. (2008). Data from seven sensors from the west side of the bridge and three mirroring sensors placed on the east side of the bridge were used for this study, as shown in Fig. 9; the sensor channels measuring acceleration perpendicular to the roadway of the bridge were considered to identify vertical and torsional modes.

Initial estimates for $A$ and $C$ matrices were calculated from ERA-OKID-OO ($p = 4$) using the measured data. The data were filtered and downsampled to consider frequency content up to 1 Hz and used in STRIDE ($p = 4$) with $\theta = 5 \times 10^{-4}$. The remaining state-space parameters were estimated in accordance with the guidelines presented earlier in this paper. STRIDE required 88 iterations to achieve the designated convergence slope threshold.

Identical to the NHB analysis, ERA-NExT, ERA-OKID-OO, and AR were implemented for even model orders 2 through 100 [via SMIT (Chang and Pakzad 2013)], producing stabilization diagrams (ERA-NExT and ERA-OKID-OO are shown in Figs. 10 and 11) which were used to select the model order, $p$.

Results Comparison—Frequencies, Mode Shapes, and Damping Ratios

In this section, a detailed comparison of the modal identification results from STRIDE, ERA-NExT, and ERA-OKID-OO is presented for GGB data. STRIDE results are also briefly compared to those of AR in terms of computational FLOPs and overall performance (Figs. 14 and 15, and Table 4). Using STRIDE, estimates for all structural modes were available as early as the 16th iteration ($\theta = 0.10$); however, as discussed previously, earlier iterations of STRIDE are subject to inflated damping ratios (due to the initial estimate from a low model order ERA-OKID-OO) and were not selected.

Similar to the NHB analysis, there was not a single model order for ERA-OKID-OO that included modal estimates for all considered modes. Using the ERA-OKID-OO stabilization diagram, Fig. 11, model order 78 was chosen to represent the method, despite missing two modes (7 and 12, Table 3). Additionally, in model order 66, modes 7 and 12 were identified but modes 3, 6, and 10 were missing. This inconsistency reinforces the fact that identification results in algorithms that require model order selection have varied results among model orders; in this case, a comprehensive modal analysis for ERA-OKID-OO requires a combination of modal estimates from various model orders, which is not automated and can be time-consuming.

Table 3 summarizes the modal identification results of the three methods. Fig. 12 compares the mode shapes for six selected modes.
Fig. 10. Stabilization diagram using ERA-NExT for GGB data ($N = 10$ sensors; $K = 10,000$ samples)

Fig. 11. Stabilization diagram using ERA-OKID-OO for GGB data ($N = 10$ sensors; $K = 10,000$ samples)

<table>
<thead>
<tr>
<th>Table 3. Summary of Identified Vertical/Torsional Modes of GGB below 1 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modal property</td>
</tr>
<tr>
<td>XT frequencies (Hz)</td>
</tr>
<tr>
<td>OO frequencies (Hz)</td>
</tr>
<tr>
<td>ST frequencies (Hz)</td>
</tr>
<tr>
<td>XT damping (%)</td>
</tr>
<tr>
<td>OO damping (%)</td>
</tr>
<tr>
<td>ST damping (%)</td>
</tr>
<tr>
<td>ST and OO MAC</td>
</tr>
<tr>
<td>ST and XT MAC</td>
</tr>
<tr>
<td>OO and XT MAC</td>
</tr>
</tbody>
</table>

Note: Vertical and torsional modes are represented by $V$ or $T$, respectively; ST, OO, and XT denote STRIDE, ERA-OKID-OO, and ERA-NExT, respectively; miss denotes a completely missed mode; MAC values are computed for all three shape combinations; Superscripts ST, OO, and XT denote respective mode shape misses.
STRIDE and ERA-NExT successfully identified all 14 frequencies; ERA-OKID-OO failed to identify modes 7 and 12. Similar to the NHB results, the frequency results from STRIDE were consistent with ERA-NExT and ERA-OKID-OO, with some variance between the damping ratios for modes 1, 2, 3, and 6, likely a result of the aforementioned performance of SID algorithms at low model orders; however it is important to reiterate the difficulties in verifying estimated damping properties of real structural systems.

**Computation Comparison**

Fig. 13 shows the number of iterations necessary to achieve a variety of slope thresholds at model orders $p = 2$ and $4$ for the GGB data. As in the NHB data, a sharp demand in iterations, and in turn computational efforts, is evident around $\theta < 5 \times 10^{-4}$. This also implies that slope thresholds in the proximity of $\theta = 5 \times 10^{-4}$ are computationally efficient for the GGB data.

Fig. 14 shows the estimated FLOPs for ERA-NExT, ERA-OKID-OO, and STRIDE with the GGB data and Fig. 15 shows the cumulative estimated FLOPs. Fig. 15 shows the cumulative

---

**Fig. 12.** Six selected modes from identifications of GGB data; ERA-NExT ($p = 94$), ERA-OKID-OO ($p = 78$), and STRIDE ($p = 4$)

**Fig. 13.** Number of STRIDE iterations required for decreasing slope thresholds using GGB data ($N = 10$ sensors; $K = 4,000$ samples); model orders are coded with iteration counts at selected slopes

**Fig. 14.** Estimated FLOPs for GGB data ($N = 10$ sensors); ERA-OKID-OO, ERA-NExT, and AR are plotted against model order; STRIDE ($p = 4$) is plotted against iterations
Cumulative FLOPs Comparison: GGB

Fig. 15. Cumulative estimated FLOPs for GGB data (N = 10 sensors); ERA-OKID-OO, ERA-NExT, and AR are plotted against model order; STRIDE (p = 4) is plotted against iterations.

Table 4. Summary of Computational Efforts and Performance for Modal Analysis of GGB Data Using Various SID Algorithms and STRIDE

<table>
<thead>
<tr>
<th>Method</th>
<th>Model order</th>
<th>Iterations</th>
<th>Cumulative estimated FLOPs</th>
<th>Identified modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERA-NExT</td>
<td>2 through 94</td>
<td>—</td>
<td>2.02 x 10^{11}</td>
<td>14 out of 14</td>
</tr>
<tr>
<td>ERA-OKID-OO</td>
<td>2 through 78</td>
<td>—</td>
<td>7.17 x 10^{11}</td>
<td>12 out of 14</td>
</tr>
<tr>
<td>AR</td>
<td>2 through 68</td>
<td>—</td>
<td>1.89 x 10^{11}</td>
<td>14 out of 14</td>
</tr>
<tr>
<td>STRIDE</td>
<td>4</td>
<td>88</td>
<td>4.60 x 10^{11}</td>
<td>14 out of 14</td>
</tr>
</tbody>
</table>

computational costs of each SID algorithm, accounting for the model order selection process, i.e., repeated analyses at increasing model orders, which is required to determine an appropriate model order. Simultaneously, the cumulative computational costs of STRIDE versus iteration number are superimposed. While the abscissa is different for STRIDE and the other SID, the total computation costs between methods are comparable. Table 4 summarizes these cumulative computational costs for each method: ERA-NExT required the least cumulative FLOPs, STRIDE required less than ERA-OKID-OO and AR, but all were within the same order of magnitude.

Conclusions

This paper introduced structural identification using expectation maximization (STRIDE), a novel application for the EM algorithm and approach for output-only modal identification. Most required calculations in STRIDE come directly from the EM algorithm in the stochastic state-space model (Box et al. 2008; Digalakis et al. 1993; Ghahramani and Hinton 1996; Murphy 2012; Shumway and Stoffer 2011) and have been provided in this paper. The computation of initial state-space parameters required for commencement of STRIDE was discussed. It was proposed that an alternative output-only SID method such as ERA-NExT or ERA-OKID-OO should be used to calculate initial A and C matrices, and all other state-space parameters can be set in accordance with the section on initial parameters.

The main benefits of STRIDE are its capability of providing an accurate, comprehensive modal analysis at model orders significantly lower than those of other SID, its iterations affect cumulative FLOPs linearly while model orders affect other SID algorithms cubically, it avoids an involved model order selection procedure (assuming an adequate convergence criterion is defined) by conveniently organizing its most accurate results at the latter iterations, and it has provided more stable, accurate damping ratio estimates than the SID algorithms which require model order selection. Finally, STRIDE is only the most basic implementation of the EM algorithm—there have been numerous modifications and extensions (Krishnan 2008) to this date that are designed to remedy its least attractive features, e.g., speeding up convergence (Meng and van Dyk 1997) or computing confidence bounds (Matarazzo and Pakzad 2015).

This paper used STRIDE to identify modal parameters of the Northampton Street Bridge (NHB) in Easton, Pennsylvania, and the Golden Gate Bridge (GGB) in San Francisco, California, using collected sensor data. Its performance was validated through providing results consistent with those of ERA-NExT, ERA-OKID-OO, and AR. For the NHB application, STRIDE identified 10 modes under 10 Hz at model order 2, performing better than ERA-NExT, ERA-OKID-OO, and AR. For the GGB application, STRIDE identified 14 modes under 1 Hz at model order 4, performing as well as ERA-NExT and AR, and better than ERA-OKID-OO. In both applications, STRIDE used fewer cumulative FLOPs than ERA-OKID-OO.

Appendix. Detailed FLOP Estimation of STRIDE and Derivation of M-Step Updates

Detailed FLOP Estimation of STRIDE

In this section, a detailed estimation of FLOPs for STRIDE is provided. The computational costs of STRIDE are measured by the total number of operations (FLOPs), using lecture notes (Vandenberghe 2014), the LAPACK user’s guide (Anderson et al. 1999), and the lightspeed MATLAB toolbox (Minka 2013) to estimate the cost of solutions to Ax = B and xA = B for x, likelihood evaluation, and eigenvalue-decomposition. In general, STRIDE can accurately identify modal properties using a comparable, sometimes smaller, number of cumulative FLOPs when compared to ERA-NExT, ERA-OKID-OO, or AR.

Kalman Filter (E-Step): Eqs. (10a)–(11f)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>pN x pN</td>
<td>None, given</td>
</tr>
<tr>
<td>C</td>
<td>N x pN</td>
<td>None, given</td>
</tr>
</tbody>
</table>

### Appendix (Continued)

#### Kalman Filter (E-Step): Eqs. (10a)–(11f)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathbf{x}_{pr} = A \cdot \mathbf{x}_f$</td>
<td>$pN \times 1$</td>
<td>$\begin{bmatrix} p^2 N^3 \ pN \cdot (pN - 1) \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{V}_{pr} = A \cdot \mathbf{V}_f \cdot A + \mathbf{Q}$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} 2p^2 N^3 \ 2p^2 N^2 \cdot (pN - 1) \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{e} = \mathbf{y} - C \cdot \mathbf{x}_{pr}$</td>
<td>$N \times 1$</td>
<td>$\begin{bmatrix} p^2 N^3 \ pN \cdot (pN - 1) \end{bmatrix} + \begin{bmatrix} [N] \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{S} = C \cdot \mathbf{V}_{pr} \cdot C^T$</td>
<td>$N \times N$</td>
<td>$\begin{bmatrix} p^2 N^3 \ pN^2 + [pN^2 + N^2] \cdot (pN - 1) \end{bmatrix}$</td>
</tr>
<tr>
<td>$K = \mathbf{V}_{pr} \cdot C^T / S$</td>
<td>$pN \times N$</td>
<td>$\begin{bmatrix} p^2 N^3 \ pN^2 \cdot (pN - 1) \end{bmatrix} + \begin{bmatrix} 2N^2 \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{x}<em>{new} = \mathbf{x}</em>{pr} + K \cdot \mathbf{e}$</td>
<td>$pN \times 1$</td>
<td>$\begin{bmatrix} pN^2 \ pN \cdot (pN - 1) \end{bmatrix} + \begin{bmatrix} [N] \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{V}<em>{new} = (I - K \cdot C) \cdot \mathbf{V}</em>{pr}$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} p^2 N^3 + 2p^3 N^3 \ p^2 N^2 \cdot (N + pN - 2) + [p^2 N^2] \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{VV}_{new} = (I - K \cdot C) \cdot A \cdot \mathbf{V}$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} p^2 N^3 + 2p^3 N^3 \ p^2 N^2 \cdot (N + pN - 3) + [p^2 N^2] \end{bmatrix}$</td>
</tr>
</tbody>
</table>

Note: Repeat $K$ times, for iterations $j = 1, \ldots, J$.

#### Innovations Likelihood Calculation: Eq. (8)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G(\mathbf{y}_{j+1}</td>
<td>\mathbf{y}_j)$</td>
<td>Scalar</td>
</tr>
</tbody>
</table>

Note: Repeat for iterations $j = 1, \ldots, J$.

#### RTS Smoother (E-Step): Eqs. (12a)–(12c), and (13c)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J = \mathbf{V}<em>f \cdot A^T / \mathbf{V}</em>{pr}$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} p^4 N^3 + 2p^3 N^2 \cdot (pN - 1) \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{x}<em>{sm} = \mathbf{x}<em>f + J \cdot (\mathbf{x}</em>{sa} - \mathbf{x}</em>{pr})$</td>
<td>$pN \times 1$</td>
<td>$\begin{bmatrix} pN \ p^3 N^2 \ pN \cdot (pN - 1) \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{V}<em>{sm} = \mathbf{V}<em>f + J \cdot (\mathbf{V}</em>{sa} - \mathbf{V}</em>{pr}) \cdot J^T$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} 2p^3 N^3 \end{bmatrix}$</td>
</tr>
<tr>
<td>$\mathbf{VV}_{sm} = \mathbf{VV}<em>f + [(\mathbf{V}</em>{sa} - \mathbf{V}_f) / \mathbf{V}_f] \cdot \mathbf{VV}_f$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} 2p^2 N^2 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

Note: Repeat $K$ times, for iterations $j = 1, \ldots, J$.

#### State Statistics (M-Step): within Eqs. (14)–(19)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta = \Delta + \mathbf{y} \cdot \mathbf{x}^T$</td>
<td>$N \times pN$</td>
<td>$\begin{bmatrix} pN^2 \end{bmatrix}$</td>
</tr>
<tr>
<td>$\gamma = \gamma + \mathbf{x} \cdot \mathbf{x}^T + \mathbf{V}$</td>
<td>$pN \times pN$</td>
<td>$\begin{bmatrix} p^2 N^2 \end{bmatrix}$</td>
</tr>
<tr>
<td>$\alpha = \alpha + \mathbf{y} \cdot \mathbf{y}^T$</td>
<td>$N \times N$</td>
<td>$\begin{bmatrix} [N] \end{bmatrix}$</td>
</tr>
</tbody>
</table>

Note: Repeat $K$ times, for iterations $j = 1, \ldots, J$. 


J. Eng. Mech., 04015109
### Derivation of M-Step Updates

This derivation supplements the M-step section, explicitly detailing STRIDE parameter updates, which are derived from the conditional log-likelihood function in Eq. (7). In the equations that follow, the conditional expectation of the log-likelihood function \( G(\Psi_{j+1} | \Psi_j) \) is condensed to \( G \) (a scalar) and the iteration \( j \) parameter subscript may be omitted in some instances. Eq. (27) is used as a shorthand notation for a recurring conditional expectation:

\[
E^{(K,j)}[\cdot] = E[\cdot | \Psi_j, \ldots, \Psi_K] \tag{27}
\]

In Eq. (28), the partial derivative of \( G \) with respect to the state matrix \( A \) is set to zero to determine \( A_{j+1} \). Symmetry of \( Q \) is invoked, resulting in Eq. (14):

\[
\frac{\partial G}{\partial A} = \frac{\partial}{\partial A} \left( E^{(K,j)} \left[ -\frac{1}{2} \sum_{k=2}^{K} (x_k - Ax_{k-1})^T Q^{-1} (x_k - Ax_{k-1}) \right] \right) = 0 \tag{28}
\]

### State Statistics (M-Step): within Eqs. (14)–(19)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta = \beta + x \cdot x_j^T + V )</td>
<td>( pN \times pN )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
</tbody>
</table>

Note: Repeat \( K - 1 \) times, for iterations \( j = 1, \ldots, J \).

### State Statistics (M-Step): within Eqs. (14)–(19)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \gamma_1 = \gamma - x_K \cdot x_K^T - V_K )</td>
<td>( pN \times pN )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
<tr>
<td>( \gamma_2 = \gamma - x_1 \cdot x_1^T - V_1 )</td>
<td>( pN \times pN )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
</tbody>
</table>

Note: Repeat for iterations \( j = 1, \ldots, J \).

### Update Parameters (M-Step): within Eqs. (14)–(19)

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A = \beta / \gamma_1 )</td>
<td>( pN \times pN )</td>
<td>((2p^3N^3)^{\text{substitution}})</td>
</tr>
<tr>
<td>( Q = (\gamma_2 - A : \beta) / K )</td>
<td>( pN \times pN )</td>
<td>((p^3N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
<tr>
<td>( C = \Delta / \gamma )</td>
<td>( N \times pN )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
<tr>
<td>( R = (\alpha - C \cdot \Delta^T) / K )</td>
<td>( N \times N )</td>
<td>((pN^3)^{\text{multiplication}} + (N^2(pN - 1))^ {\text{addition}} + (N^2)^{\text{addition}})</td>
</tr>
<tr>
<td>( \hat{\mu} = x_1 )</td>
<td>( pN \times 1 )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
<tr>
<td>( \hat{V} = V_1 - x_1 \cdot x_1^T )</td>
<td>( pN \times pN )</td>
<td>((p^2N^2)^{\text{multiplication}} + (p^2N^2)^{\text{addition}})</td>
</tr>
</tbody>
</table>

Note: Repeat for iterations \( j = 1, \ldots, J \).

### Eigen Decomposition of \( A \): for Eqs. (21(a)–(21(c))

<table>
<thead>
<tr>
<th>Description</th>
<th>Dimension</th>
<th>Number of operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{eig}(A) = \Gamma_{ML} \cdot K_{ML} )</td>
<td>Both are ( pN \times pN )</td>
<td>((263.3) \cdot (p^3N^3)^{\text{eigendecomposition}})</td>
</tr>
</tbody>
</table>
Note that parameter update for iteration $j + 1$ in Eq. (28) depends on statistics from iteration $j$. Next in Eq. (29), the partial derivative of $G$ with respect to the observation matrix $C$ is set to zero to determine $C_{j+1}$. Symmetry of $R$ is invoked, yielding Eq. (15):

$$\frac{\partial G}{\partial C} = \frac{\partial}{\partial C} \left( E^{(K,\hat{j})} \left[ -\frac{1}{2} \sum_{k=1}^{K} (y_k - Cx_k)^T R^{-1} (y_k - Cx_k) \right] \right) = R^{-1} \sum_{k=1}^{K} (y_k E^{(K,\hat{j})} [x_k^T] - CE^{(K,\hat{j})} [x_k x_k^T]) = 0$$

$$C_{j+1} = \sum_{k=1}^{K} y_k E^{(K,\hat{j})} [x_k^T] \left( \sum_{k=1}^{K} E^{(K,\hat{j})} [x_k x_k^T] \right)^{-1}$$

$$C_{j+1} = \left( \sum_{k=1}^{K} y_k x_k^T \right) \left( \sum_{k=1}^{K} [x_k x_k^T + \hat{V}_{k,j}[k]] \right)^{-1}$$  \hspace{1cm} (29)$$

In Eq. (30), the partial derivative of $G$ with respect to the inverse of the input covariance matrix $Q^{-1}$ is set to zero to determine $Q_{j+1}$:

$$\frac{\partial G}{\partial Q^{-1}} = \frac{\partial}{\partial Q^{-1}} \left( E^{(K,\hat{j})} \left[ -\frac{1}{2} \ln(Q) - \frac{1}{2} \sum_{k=2}^{K} (x_k - Ax_{k-1})^T Q^{-1} (x_k - Ax_{k-1}) \right] \right) = 0$$

$$= \frac{K-1}{2} - \frac{1}{2} \sum_{k=2}^{K} \left( E^{(K,\hat{j})} [x_k x_k^T] - E^{(K,\hat{j})} [x_k x_{k-1}^T] A^T - AE^{(K,\hat{j})} [x_{k-1} x_k^T] + AE^{(K,\hat{j})} [x_{k-1} x_{k-1}^T] A^T \right) = 0$$

$$Q_{j+1} = \frac{1}{K-1} \sum_{k=2}^{K} \left( E^{(K,\hat{j})} [x_k x_k^T] - E^{(K,\hat{j})} [x_k x_{k-1}^T] A^T - AE^{(K,\hat{j})} [x_{k-1} x_k^T] + AE^{(K,\hat{j})} [x_{k-1} x_{k-1}^T] A^T \right)$$  \hspace{1cm} (30)$$

An important assumption is required to obtain Eq. (16) from the last line of Eq. (30). The result simplifies the expression and expedites the algorithm updates. While Eq. (16) is used throughout the literature (Cara et al. 2012; Digalakis et al. 1993; Pridham and Wilson 2004; Shumway and Stoffer 1982), the following assumption has been absent despite its necessity. Assuming the parameters are updated in the order presented, a new state matrix $A_{j+1}$ is available when $Q_{j+1}$ is computed. By replacing or preupdateing the $A$ terms in Eq. (30) with the result in Eq. (28) so that $A = A_{j+1}$, (as opposed to $A = A_{j}$) an interesting outcome enables more efficient parameter updates.

First, observe the update for $Q_{j+1}$ as a function of $A_{j}$ in Eq. (31) below assuming $j \geq 2$ (integers). This is perhaps the more intuitive approach, as the conditional log-likelihood function $G$ is defined given parameters at iteration $j$:

$$Q_{j+1}(A_{j}) = \frac{1}{K-1} \sum_{k=2}^{K} \left( E^{(K,\hat{j})} [x_k x_k^T] - E^{(K,\hat{j})} [x_k x_{k-1}^T] (A_{j})^{-1} E^{(K,\hat{j})} [x_{k-1} x_k^T] + (A_{j}) E^{(K,\hat{j})} [x_{k-1} x_{k-1}^T] (A_{j})^{-1} \right)$$

$$= \frac{1}{K-1} \left( \sum_{k=2}^{K} E^{(K,\hat{j})} [x_k x_k^T] - \sum_{k=2}^{K} E^{(K,\hat{j})} [x_k x_{k-1}^T] \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_k^T] \right)$$

$$- \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_k x_{k-1}^T] \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_k^T] \right)$$

$$+ \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_k x_k^T] \right) \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T]$$  \hspace{1cm} (31)$$

No further simplifications can be made to this expression; more importantly, this computation requires storage of statistics from the two most recent iterations ($j$ and $j - 1$). Now, if $Q_{j+1}$ is a function of $A_{j+1}$ as shown in Eq. (32) below, the terms can be simplified so that all statistics are in terms of the state and state covariance estimates at iteration $j$:

$$Q_{j+1}(A_{j+1}) = \frac{1}{K-1} \sum_{k=2}^{K} \left( E^{(K,\hat{j})} [x_k x_k^T] - E^{(K,\hat{j})} [x_k x_{k-1}^T] (A_{j+1})^{-1} E^{(K,\hat{j})} [x_{k-1} x_k^T] + (A_{j+1}) E^{(K,\hat{j})} [x_{k-1} x_{k-1}^T] (A_{j+1})^{-1} \right)$$

$$= \frac{1}{K-1} \left( \sum_{k=2}^{K} E^{(K,\hat{j})} [x_k x_k^T] - \sum_{k=2}^{K} E^{(K,\hat{j})} [x_k x_{k-1}^T] \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_k^T] \right)$$

$$- \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_k x_{k-1}^T] \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_k^T] \right)$$

$$+ \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_k x_k^T] \right) \left( \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T] \right)^{-1} \sum_{k=2}^{K} E^{(K,\hat{j}+1)} [x_{k-1} x_{k-1}^T]$$  \hspace{1cm} (32)$$

The last line of Eq. (32) is Eq. (16); this assumption does not necessarily skip an iteration. However, the behavior implies something similar as it forces the update to only use the most recent information, i.e., from the current iteration $j$. Additionally, the calculation Eq. (32) is simpler than Eq. (31).

Consider iterations $j \geq 2$ and the following assumption for the parameter updates:

$$A_j \rightarrow A_{j+1}, \quad \sum_{k=2}^{K} E^{(K,j-1)} [x_{k-1} x_{k-1}^T] \rightarrow \sum_{k=2}^{K} E^{(K,j)} [x_{k-1} x_{k-1}^T] \quad \text{and} \quad Q_{j+1}(A_j) \rightarrow Q_{j+1}(A_{j+1})$$ (33)

That is, as EM progresses and $j$ becomes large, mean square statistics for consecutive iterations are very close. The authors have empirically validated the behavior of the $Q$ update in Eq. (33); more specifically, for large $j$ Eq. (31) converges to Eq. (32), i.e., $\{Q_{j+1}(A_j) - Q_{j+1}(A_{j+1})\} \rightarrow 0$.

In Eq. (34), the partial derivative of $G$ with respect to the inverse of the observation noise covariance matrix $R^{-1}$ is set to zero to determine $R_{j+1}$. Note, the updates of $R$ and $C$ have a similar relationship to that of $Q$ and $A$:

$$\frac{\partial G}{\partial R^{-1}} = \frac{\partial}{\partial R^{-1}} \left( E^{(K,j)} \left[ -\frac{K}{2} \ln(R) - \frac{1}{2} \sum_{k=1}^{K} (y_k - C x_k)^T R^{-1} (y_k - C x_k) \right] \right) = 0$$

$$= \frac{K}{2} R - \frac{1}{2} \sum_{k=1}^{K} (y_k y_k^T - y_k E^{(K,j)} [x_k^T] C^T - C E^{(K,j)} [x_k] y_k^T + C E^{(K,j)} [x_k x_k^T] C^T) = 0$$

$$R_{j+1} = \frac{1}{K} \sum_{k=1}^{K} (y_k y_k^T - y_k E^{(K,j)} [x_k^T] C^T - C E^{(K,j)} [x_k] y_k^T + C E^{(K,j)} [x_k x_k^T] C^T)$$ (34)

As before, assume the parameters are updated in the order presented so that a new observation matrix $C_{j+1}$ is available when $R_{j+1}$ is computed. Replace or preupdate the $C$ terms in Eq. (34) with the result in Eq. (29) using $C = C_{j+1}$ (as opposed to $C = C_j$). Eq. (35) shows the update for $R_{j+1}$ if $C_j$ were used (assuming $j \geq 2$; note the parallel between $R$ and $C$, and $Q$ and $A$):

$$R_{j+1}(C_j) = \frac{1}{K} \sum_{k=1}^{K} \left( y_k y_k^T - y_k E^{(K,j)} [x_k^T] \{ C_j^T \} - \{ C_j \} E^{(K,j)} [x_k] y_k^T + \{ C_j \} E^{(K,j)} [x_k x_k^T] \{ C_j^T \} \right)$$

$$= \frac{1}{K} \left( \sum_{k=1}^{K} y_k y_k^T - \sum_{k=1}^{K} y_k E^{(K,j)} [x_k^T] \left( \left( \sum_{k=1}^{K} E^{(K,j-1)} [x_k x_k^T] \right)^{-1} \sum_{k=1}^{K} E^{(K,j-1)} [x_k] y_k^T \right) \right)$$

$$- \left\{ \sum_{k=1}^{K} y_k E^{(K,j-1)} [x_k^T] \left( \sum_{k=1}^{K} E^{(K,j-1)} [x_k x_k^T] \right)^{-1} \right\} \sum_{k=1}^{K} E^{(K,j)} [x_k] y_k^T$$

$$+ \left\{ \sum_{k=1}^{K} y_k E^{(K,j-1)} [x_k^T] \left( \sum_{k=1}^{K} E^{(K,j-1)} [x_k x_k^T] \right)^{-1} \right\} E^{(K,j)} [x_k x_k^T] \left( \left( \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right)^{-1} \sum_{k=1}^{K} E^{(K,j)} [x_k] y_k^T \right)$$ (35)

As with $Q_{j+1}(A_j)$ in Eq. (31), this expression cannot be simplified further and also requires storage of statistics from two iterations ($j$ and $j-1$). If $R_{j+1}$ is a function of $C_{j+1}$ as in Eq. (36) below, the terms can be further reduced so that all required statistics are in terms of the state and state covariance estimates at the same iteration $j$:

$$R_{j+1}(C_{j+1}) = \frac{1}{K} \sum_{k=1}^{K} \left( y_k y_k^T - y_k E^{(K,j)} [x_k^T] \{ C_{j+1}^T \} - \{ C_{j+1} \} E^{(K,j)} [x_k] y_k^T + \{ C_{j+1} \} E^{(K,j)} [x_k x_k^T] \{ C_{j+1}^T \} \right)$$

$$= \frac{1}{K} \left( \sum_{k=1}^{K} y_k y_k^T - \sum_{k=1}^{K} y_k E^{(K,j)} [x_k^T] \left( \left( \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right)^{-1} \sum_{k=1}^{K} E^{(K,j)} [x_k] y_k^T \right) \right)$$

$$- \left\{ \sum_{k=1}^{K} y_k E^{(K,j)} [x_k^T] \left( \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right)^{-1} \right\} \sum_{k=1}^{K} E^{(K,j)} [x_k] y_k^T$$

$$+ \left\{ \sum_{k=1}^{K} y_k E^{(K,j)} [x_k^T] \left( \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right)^{-1} \right\} E^{(K,j)} [x_k x_k^T] \left( \left( \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right)^{-1} \sum_{k=1}^{K} E^{(K,j)} [x_k] y_k^T \right)$$

$$= \frac{1}{K} \left( \sum_{k=1}^{K} y_k y_k^T - C_{j+1} \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \right) = \frac{1}{K} \left( \sum_{k=1}^{K} y_k y_k^T - C_{j+1} \sum_{k=1}^{K} \tilde{x}_{k[K]} y_k^T \right)$$ (36)

The last line of Eq. (36) is Eq. (17); the calculation in Eq. (36) is simpler than Eq. (35). Consider iterations $j \geq 2$ and the following assumption for the parameter updates:

$$as C_j \rightarrow C_{j+1}, \quad \sum_{k=1}^{K} E^{(K,j-1)} [x_k x_k^T] \rightarrow \sum_{k=1}^{K} E^{(K,j)} [x_k x_k^T] \quad \text{and} \quad R_{j+1}(C_j) \rightarrow R_{j+1}(C_{j+1})$$ (37)

© ASCE

Like Eq. (33), as EM progresses and \( j \) becomes large, mean square state statistics for consecutive iterations are very close. The authors have empirically validated the nature of the \( R \) update in Eq. (37); for large \( j \), Eq. (35) converges to Eq. (36), i.e., \( R_{j+1}(C) - R_{j+1}(C_{j+1}) \rightarrow 0 \).

In Eq. (38) the partial derivative of \( G \) with respect to the initial state vector \( \mu \) is set to zero to determine \( \hat{\mu}_{j+1} \); note the result is identical to Eq. (18):

\[
\frac{\partial G}{\partial \mu} = \frac{\partial}{\partial \mu} \left( E(K,J) \left[ -\frac{1}{2} \ln(V) - \frac{1}{2} (x_1 - \hat{\mu})^T V^{-1} (x_1 - \hat{\mu}) \right] \right) = 0
\]

\[
= -2V^{-1}(E(K,J))[x_1] - \hat{\mu} = 0
\]

\[
\hat{\mu}_{j+1} = E(K,J)[x_1] = \bar{x}_{1|K}
\]  

(38)

In Eq. (39), the partial derivative of \( G \) with respect to the inverse of the initial state covariance matrix \( \bar{V}^{-1} \) is set to zero to determine \( \bar{V}_{j+1}^{-1} \); the result is identical to Eq. (19):

\[
\frac{\partial G}{\partial \bar{V}^{-1}} = \frac{\partial}{\partial \bar{V}^{-1}} \left( E(K,J) \left[ -\frac{1}{2} \ln(V) - \frac{1}{2} (x_1 - \hat{\mu})^T V^{-1} (x_1 - \hat{\mu}) \right] \right) = 0
\]

\[
= \frac{1}{2} \bar{V} - \frac{1}{2} E(K,J)[x_1 - \hat{\mu}]^T (x_1 - \hat{\mu}) = 0
\]

\[
\bar{V}_{j+1}^{-1} = E(K,J)[x_1 - \hat{\mu}]^T (x_1 - \hat{\mu}) = \bar{V}_{1,1|K}
\]  

(39)

Acknowledgments

Research funding is partially provided by the National Science Foundation through Grant No. CMMI-1351537 by the Hazard Mitigation and Structural Engineering program, and by a grant from the Commonwealth of Pennsylvania, Department of Community and Economic Development, through the Pennsylvania Infrastructure Technology Alliance (PTA).

References


