STATE-SPACE MODELING OF LARGE DOMAIN WAVE PROPAGATION SYSTEMS BY CY-PARTITIONED MATRICES

Richard S. Darling, Minh Q. Phan, and Stephen A. Ketcham

The $C_y$-partitioned method to produce a reduced-order, discrete-time, state-space model for large domain wave propagation is described. The method is suitable when the output dimension is orders of magnitude higher than the number of discrete-time samples specifying the time duration of interest. The model is characterized by a relatively small dynamic component relating the inputs to a relatively small number of anchored outputs, and a large static component relating these anchored outputs to all remaining outputs. The dynamic-static partition permits identification of these models in the time domain which is not possible otherwise due to the extremely large computational power and memory requirements.

INTRODUCTION

In the area of short-duration large-domain dynamic signal propagation, highly accurate reduced-order models represent an enabling technology, Refs. [1]-[4]. Such models can be used for rapid prediction of the dynamic responses without resorting to the time-consuming simulation that is typically carried out on a high-performance supercomputer with hundreds of nodes. Significant savings in computational resources can be achieved by this strategy, reducing what normally takes hours on a supercomputer to minutes on a laptop. By taking advantage of the knowledge of the dynamics of the environment represented by these reduced-order models, it is possible to address output prediction, source signal recovery and localization applications in highly complex environments with multiple reflections. Another example is the modeling of shock propagation throughout a spacecraft structure for damage assessment. To date, identification of these reduced-order models is performed in the frequency domain which involves the use of specialized input signals (e.g., a frequency-weighted Gaussian pulse) to minimize identification errors associated with the inverse Fast Fourier Transform (FFT) technique.

Of interest is the capability to identify these models in the time domain as well. Time-domain techniques allow the use of more general input signals both with HPC simulation data or actual field measurements where specialized input signals are not available, Refs. [5]-[13]. Existing time-domain system identification methods, however, cannot be applied directly due to the extreme requirements caused by the large number of outputs that are needed to describe the wave propagation. In this paper, we exploit a particular feature of the problem to make time-domain identification feasible again. Although the true system order and the number of outputs are may be very high, the time dimension of the desired predictive model is many orders of magnitude smaller. For example, a problem might have one source and a millions output locations, but the time duration of interest is

$^*$Signature Physics Branch, Cold Regions Research and Engineering Laboratory, Hanover, NH 03755, USA.
$^†$Thayer School of Engineering, Dartmouth College, Hanover, NH 03755, USA
$^‡$Signature Physics Branch, Cold Regions Research and Engineering Laboratory, Hanover, NH 03755, USA.
only 512 samples. Due to the complexity of the environment, although the dynamics from a source to each of the million output locations can all be different, it is not necessary to identify each of these 1 million dynamical relationships independently. Instead, it is only necessary to identify the dynamical relationships from the source to a set of outputs, called anchored outputs, from which the dynamical responses at all remaining output locations can be inferred by a static relationship. Returning to the previous example, if the interested time duration is 512 time steps, then the minimum number of anchored outputs is at most 512. The responses of the remaining output locations (1 million minus 512) can be inferred from the anchored output responses by a static relationship.

We develop here a model structure consisting of two portions: a dynamic portion that relates the inputs to a relatively small number of anchored outputs, and a static portion relating these anchored outputs to all remaining outputs. The dynamic portion of the model is relatively small in dimensions and can be easily identified. The static portion can be large in dimensions, but being static, it can be identified far more easily than a dynamic model. Furthermore, the computational requirement for the identification of the static portion of the model turns out to be quite minimal, thus making the entire time-domain identification of these models feasible.

In the following sections, we first describe three versions of the $C_y$-partitioned models, referred to as basic, extended, and generalized. These versions are related to the rank of the output influence matrix (the $C$-matrix in a typical $A, B, C$ state-space representation). Next, we describe a mechanism to reduce the dimensions of these models further including a technique where the two steps of model identification and model order reduction can be performed simultaneously from input-output measurements. Finally, numerical results are provided to illustrate how these models can be identified in practice.

**MATHEMATICAL FORMULATION**

For simplicity, the formulation starts out with the assumption that a state-space model of the system is available, and various versions of the $C_y$-partitioned models can be generated. Later, we show that these $C_y$-partitioned models can be directly derived from input-output measurements. Consider a $r$-input, $q$-output, $n$-dimensional state-space model of the form,

$$
x(k+1) = Ax(k) + Bu(k) \tag{1}
$$

$$
y(k) = Cx(k) \tag{2}
$$

where $q >> n$. In a typical wave propagation problem in large domain that we are addressing, typical values of $q$ are in the millions and $n$ in the hundreds. For finite-time problems, it is known that $n$ is bounded above by $pr$ or $pq$, whichever is less, where $p$ is the number of samples in the finite-duration of interest, Ref. [14].

**The Basic Case: Rank($C$) = $n$, $q_1 = n$**

The simplest possible case occurs when $C$ is full rank, i.e., the rank of $C$ is equal to the state dimension $n$ of the model. Let $C$ be partitioned into two parts, $C_1$ and $C_2$, where $C_1$ is square and full rank, and $C_2$ consists of the remaining $q - n$ rows of $C$. Correspondingly, let $y$ be partitioned into two groups, $y_1$ and $y_2$, where $y_1$ contains the $q_1 = n$ outputs associated with $C_1$, and $y_2$ contains the remaining $q - n$ outputs associated with $C_2$,

$$
y(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} x(k) = \begin{bmatrix} C_1 x(k) \\ C_2 x(k) \end{bmatrix} \tag{3}
$$
Pre-multiplying (1) and the $y_2(k)$ portion in (3) with $C_1$ and inserting the identity $C_1^{-1}C_1 = I$ in appropriate locations produces

$$C_1x(k + 1) = C_1AC_1^{-1}C_1x(k) + C_1Bu(k)$$  \hspace{1cm} (4)

$$y_2(k) = C_2C_1^{-1}C_1x(k)$$  \hspace{1cm} (5)

Recognizing that $y_1(k) = C_1x(k)$, we arrive at the simplest $Cy$-partitioned model,

$$y_1(k + 1) = \bar{A}_1y_1(k) + \bar{B}_1u(k)$$  \hspace{1cm} (6)

$$y_2(k) = \bar{C}_1y_1(k)$$  \hspace{1cm} (7)

where $\bar{A}_1 = C_1AC_1^{-1}$, $\bar{B}_1 = C_1B$, $\bar{C}_1 = C_2C_1^{-1}$. The model consists of two portions: a dynamic portion that relates the outputs in $y_1$ directly to the inputs in $u$, and a static portion that relates the outputs in $y_1$ to all remaining outputs in $y_2$. For this reason, the outputs in $y_1$ are referred to as “anchored” outputs. In this case, the number of anchored outputs is equal to the number of states, $q_1 = n$, hence the dynamic portion of the $Cy$-partitioned model $A_1$, $B_1$ has the same state dimension as that of the original model $A$, $B$. Note that the anchored outputs themselves serve as the states for the dynamic portion of the model. As later shown, this is a major simplification feature from the point of view of system identification. To bring out the implication of this model in real-world applications, consider a single-input, 1-million-output model describing the wave propagation from a single source to the output space consisting of 1 million output locations in an arbitrarily complex environment. Let the time duration of interest be 512 samples long. Instead of identifying 1 million dynamic maps relating the input source to every output locations, where each input-output map can be different from another, the same system can be represented by a 512-state model relating the input source to 512 anchored outputs, and the dynamic responses of the remaining outputs (1 million minus 512) can be determined from the dynamic responses of the 512 anchored outputs by a simple static map. In other words, once this static map is known, it is possible to determine the dynamic responses of all remaining output locations from the dynamic responses of the 512 anchored output locations. This fact is true regardless of the dimension of the output space and the complexity of the environment.

The Extended Case: $\text{Rank}(C) = n$, $q_1 > n$

Although in theory any $n$ independent outputs can be used as anchored outputs, one might wish to determine the “most independent” outputs to be used as anchored outputs. For a system with a small number of outputs, the problem might be posed mathematically in a certain manner to be solved. For problems of our size, it is not practical to do so due to the huge number of outputs. A simpler approach that has been found to work in practice is to use a sufficiently large number of outputs as anchored outputs, and these outputs can be selected either randomly or geometrically. In other words, the number of outputs treated as anchored outputs will be larger than the minimum. In the following we will provide a simple derivation to show that redundant anchored outputs are allowed. The derivation parallels that given in (4)-(7), except that the identity $C_1^+C_1 = I$ is used instead of $C_1^{-1}C_1 = I$, where $C_1^+$ denotes a left-inverse of $C_1$. This left inverse exists because $C_1$ has full column rank,

$$C_1x(k + 1) = C_1AC_1^+C_1x(k) + C_1Bu(k)$$  \hspace{1cm} (8)

$$y_2(k) = C_2C_1^+C_1x(k)$$  \hspace{1cm} (9)
The $C_y$-partitioned model in this case is simply

\begin{align}
y_1(k+1) &= \bar{A}_2 y_1(k) + \bar{B}_2 u(k) \\
y_2(k) &= \bar{C}_2 y_1(k)
\end{align}

where $\bar{A}_2 = C_1 A C_1^+$, $\bar{B}_2 = C_1 B$, $\bar{C}_1 = C_2 C_1^+$. The most important difference between the extended versus basic $C_y$-partitioned models is that the dimension of the extended model is as large as the number of outputs selected to be anchored outputs, whereas the basic model has minimum dimension. For example, if 2000 outputs are selected as anchored outputs for a system whose true minimum state dimension is 100, then the dimension of the extended model is 2000. In actual applications, it was found that relatively accurate models can be found by this approach rather than trying to select the best 100 anchored outputs. Because the extended model is over-parameterized, model reduction is a question that naturally arises with this approach. We will address this issue in a later section of the paper.

**The Generalized Case: $\text{Rank}(C) < n$**

So far, in both the basic and extended models, actual outputs are used as anchored outputs. We now consider the possibility of using a set of "virtual" outputs as anchored outputs, and determine whether such a $C_y$-partitioned model exists. To this end, perform a singular value decomposition of the $C$ matrix of the original state-space model,

\[ C = U S V^T = U_1 S_1 V_1^T \]

where $S_1$ contains only the non-zero singular values of $C$, which would be the case when $\text{Rank}(C) < n$, and $U_1, V_1$ contain only the left and right singular vectors associated with these non-zero singular values, then

\[ y(k) = C x(k) = U_1 S_1 V_1^T x(k) \]

Let the state $s(k)$ be defined as

\[ s(k) = V_1^T x(k) \]

It follows that

\[ y(k) = U_1 S_1 s(k), \quad s(k) = S_1^{-1} U_1^T y(k) \]

The original $A, B, C$ model can now be interpreted as a generalized $C_y$-partitioned model that has a dynamic portion relating the input $u(k)$ to a minimum number of virtual anchored outputs $s(k)$,

\begin{align}
x(k+1) &= A x(k) + Bu(k) \\
s(k) &= V_1^T x(k)
\end{align}

and a static portion relating these virtual anchored outputs $s(k)$ to all remaining outputs $y(k)$,

\[ y(k) = U_1 S_1 s(k) \]

The number of virtual anchored outputs is minimum as it corresponds to the rank of $C$. If there are "small" singular values, they could be neglected as well. In that case, the dimension of the model can be reduced further at the expense of model accuracy.
The original \( A, B, C \) model can be put in finite-time superstable form \( A_s, B_s, C_s \), where \( A_s \) and \( B_s \) are known matrices of 0’s and 1’s, and the \( C_s \) matrices contain the \( p \) Markov parameters, \( h_1 = CB, h_2 = CAB, \ldots, h_p = CAP^{-1}B \), for a \( p \)-time step duration of interest, Ref. [14]

\[
A_s = \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
I_{r \times r} & 0 & \cdots & 0 & 0 \\
0 & I_{r \times r} & \ddots & \vdots & \vdots \\
\vdots & \ddots & \ddots & 0 & 0 \\
0 & \cdots & 0 & I_{r \times r} & 0
\end{bmatrix}_{pr \times pr}
\]

\[
B_s = \begin{bmatrix}
I_{r \times r} \\
0 \\
0 \\
\vdots \\
0
\end{bmatrix}_{pr \times r}
\]

\[
C_s = \begin{bmatrix}
h_1 & h_2 & h_3 & \cdots & h_p
\end{bmatrix}_{q \times pr}
\]

The derivation of the generalized \( C_y \)-partitioned model can proceed from \( A_s, B_s, C_s \) instead from the original \( A, B, C \). The singular value decomposition is now performed on \( C_s \) instead of \( C \), and the corresponding generalized \( C_y \)-partitioned model can be derived.

\( C_y \)-partitioned models are particularly suitable for problems where the number outputs \( q \) is orders of magnitudes larger than the minimum state dimension \( n \). Ref. [15] develops another class of models referred to as hybrid models. Hybrid models combine the benefits of the \( C_y \)-partitioned models with the convenience of the superstable representation for modeling the dynamics of finite-time processes. Hyprid models are particularly suitable for problems where the number of outputs \( q \) is orders of magnitudes larger than the number of time steps \( p \) specifying the finite-time duration of interest.

**Identification of \( C_y \)-Partitioned Models from Input-Output Data**

So far the \( C_y \)-partitioned models are derived from a known \( A, B, C \) model. We now address the question of how they can be identified from input-output data. Because the true (minimum) state dimension of the system is not known beforehand, one option is to identify the extended version of the \( C_y \)-partitioned model followed by model reduction. To this end, select a sufficiently large number of outputs, \( q_1 > n \), and treat them as anchored outputs. Let \( Y_1 \) denote a matrix of these anchored output measurements, \( Y_2 \) a matrix of the remaining output measurements, and \( V_1 \) a matrix of anchored output measurements and input data,

\[
Y_1 = \begin{bmatrix}
y_1(1) & y_1(2) & \cdots & y_1(N)
\end{bmatrix}
\]

\[
Y_2 = \begin{bmatrix}
y_2(1) & y_2(2) & \cdots & y_2(N)
\end{bmatrix}
\]

\[
V_1 = \begin{bmatrix}
y_1(0) & y_1(0) & \cdots & y_1(N - 1) \\
u(0) & u(1) & \cdots & u(N - 1)
\end{bmatrix}
\]

so that the \( C_y \)-partitioned model can be written for all available time steps as

\[
Y_1 = \begin{bmatrix}
\bar{A}_2 & \bar{B}_2
\end{bmatrix} V_1 \quad Y_2 = \bar{C}_2 Y_1
\]

The model matrices can be solved from

\[
\begin{bmatrix}
\bar{A}_2 & \bar{B}_2
\end{bmatrix} = Y_1 V_1^+ \quad \bar{C}_2 = Y_2 Y_1^+
\]
where the $+$ sign denotes pseudo-inverse which can be computed via the singular value decomposition. Other than matrix multiplication, the identification involves only the pseudo-inverses, $V_1^+$ and $Y_1^+$, both have small dimensions. In a typical application with $r$ inputs and a large number of outputs where the finite duration of interest is $p$ time steps, the finite-time superstable model dimension is $pr$. The minimum number of anchored outputs is therefore also bounded above by $pr$. Typically $q_1$ is selected to be a few times larger than $pr$ as the number of candidate anchored outputs. Obviously, the dimension of the identified model is higher than necessary, and model reduction is called for. The step is explained in the next section.

Model Reduction of Identified $Cy$-partitioned Model

In the identification of the extended $Cy$-partitioned model, we seek to satisfy:

$$
\begin{bmatrix}
  y_1(1) & y_1(2) & \cdots & y_1(N)
\end{bmatrix} = \begin{bmatrix} \bar{A}_2 & \bar{B}_2 \end{bmatrix} V_1
$$

Let the singular value decomposition of $V_1$ be

$$
V_1 = \bar{U}\bar{S}\bar{V}^T = \bar{U}_1\bar{S}_1\bar{V}_1^T
$$

where only the non-zero singular values are kept in $\bar{S}_1$. The pseudo-inverse $V_1^+$ is

$$
V_1^+ = \bar{V}_1\bar{S}_1^{-1}\bar{U}_1^T
$$

The row space of $V_1$ is spanned by the rows of $\bar{V}_1^T$ whose row dimension is now minimum. The columns of $\bar{V}_1^T$ can be treated as the reduced-dimension state vector $z(k)$,

$$
\bar{V}_1^T = \begin{bmatrix} z(1) & z(2) & \cdots & z(N) \end{bmatrix}
$$

Thus,

$$
\begin{bmatrix}
  y_1(1) & y_1(2) & \cdots & y_1(N)
\end{bmatrix} = \begin{bmatrix} \bar{A}_2 & \bar{B}_2 \end{bmatrix} \bar{U}_1\bar{S}_1 \begin{bmatrix} z(1) & z(2) & \cdots & z(N) \end{bmatrix}
$$

Examination of () reveals that a transformation matrix that reduces $y_1(k)$ to $z(k)$ by $y_1(k) = Tz(k)$ where $T$ can be derived as follows,

$$
T = \begin{bmatrix} \bar{A}_2 & \bar{B}_2 \end{bmatrix} \bar{U}_1\bar{S}_1 = Y_1V_1^+\bar{U}_1\bar{S}_1 = Y_1\bar{V}_1\bar{S}_1^{-1}\bar{U}_1^T\bar{U}_1\bar{S}_1 = Y_1\bar{V}_1
$$

This transformation matrix $T$ can be used to reduce the dimension of the identified $Cy$-partitioned model as follows,

$$
z(k + 1) = \bar{A}_R z(k) + \bar{B}_R u(k)
$$

$$
y(k) = \bar{C}_R z(k)
$$

where

$$
\bar{A}_R = T^+ \bar{A}_2 T, \quad \bar{B}_R = T^+ \bar{B}_2, \quad \bar{C}_R = \begin{bmatrix} T \\ \bar{C}_2 T \end{bmatrix}
$$
Direct Identification of Reduced-Order $C_{Y}$-partitioned Model

In the previous section, the model reduction step is performed after the $C_{Y}$-partitioned model is identified. Now we will show that the two steps can be combined into one so that a reduced-order $C_{Y}$-partitioned can be identified directly from input-output data. Recall the $z(k)$ are the columns of $V_{1}^{T}$, where $V_{1}$ is a matrix of the right singular vectors associated with the non-zero singular values of $V_{1}$ defined in (). The state-space model matrices $\bar{A}_{R}$, $\bar{B}_{R}$, $\bar{C}_{R}$ can be solved from

\[
\begin{bmatrix}
  z(1) & z(2) & \cdots & z(N)
\end{bmatrix} =
\begin{bmatrix}
  \bar{A}_{R} & \bar{B}_{R}
\end{bmatrix}
\begin{bmatrix}
  z(0) & z(1) & \cdots & z(N-1)
  u(0) & u(1) & \cdots & u(N-1)
\end{bmatrix}
\] (34)

\[
\begin{bmatrix}
  y(1) & y(2) & \cdots & y(N)
\end{bmatrix} =
\bar{C}_{R}
\begin{bmatrix}
  z(1) & z(2) & \cdots & z(N)
\end{bmatrix}
\] (35)

Define the following matrices

\[Z = \begin{bmatrix}
  z(1) & z(2) & \cdots & z(N)
\end{bmatrix}
\] (36)

\[Y = \begin{bmatrix}
  y(1) & y(2) & \cdots & y(N)
\end{bmatrix}
\] (37)

\[W = \begin{bmatrix}
  z(0) & z(1) & \cdots & z(N-1)
  u(0) & u(1) & \cdots & u(N-1)
\end{bmatrix}
\] (38)

The reduced-order $C_{Y}$-partitioned matrices can be solved directly from

\[
\begin{bmatrix}
  \bar{A}_{R} & \bar{B}_{R}
\end{bmatrix} = ZW^{+}, \quad \bar{C}_{R} = YZ^{+}
\] (39)

where again, the $+$ denotes the pseudo-inverse.

ILLUSTRATIVE EXAMPLES

We generated several seismic cases for High Performance Computer (HPC) finite-difference, time domain (FDTD) simulation, representing different input signals applied to a single generic geophysical model. The model represents a volume 120.0 m by 156.8 m to a depth of 51.65 m. Multiple subsurface layers and the surface to air interface were included. The model incorporates multiple cavities in the topology representing unroofed structural excavations extending to varying depths. The level at which outputs were obtained underlies the varying topographic surface at a constant offset of 1 m. Although this is not a single plane, we label it the ‘$xy$-plane’. The 443,120 output locations in the $xy$-plane were 0.2 m apart in both the $x$ and $y$ directions. At each of the output locations the FDTD simulation produced time histories of $x$, $y$, and $z$-direction velocities, normal and shear stresses, normal and shear strains, octahedral strains, strain energy, and other linear strain combinations relevant to fiber response studies. The input signal is a normal force in the $z$-direction located at a single point, in this study at $x=60.0$ m, $y=78.4$ m, $z=-0.25$ m. The non-zero portion of the time history of the input force differed among the cases. Figures 1 and 2 show the input signals for the two cases considered here. The first of these is the filtered pulse input, used to generate input-output data for creation of the $C_{Y}$-partitioned model. The second is a filtered random signal for validation of the model. In each of the cases the input force shown was extended with zeros to create a signal with a total duration of 256 time steps of 0.00151 s each. Note that the internal time step size in the HPC computations was significantly smaller (approximately 0.00002 s, as constrained by numerical stability). The model includes a perfectly absorbing boundary that eliminates edge effects. The HPC generated input-output histories are considered to be the truth for our example cases. The specific output parameter considered in this illustration is the $y$-direction
Figure 1. Filtered pulse input used in HPC FDTD simulation to produce a 256 time-step input-output history for creation of a reduced-order $\mathcal{C}_Y$-partitioned model.

Figure 2. Filtered random input used in the HPC FDTD simulation to produce a 'true' input-output history for validation of the reduced-order $\mathcal{C}_Y$-partitioned model.
normal strain ($\epsilon_{yy}$) as it varies over the model xy-plane during the computational time period of 0.3866 s.

The input-output history of the filtered pulse case (Figure 1) was used to produce a reduced-order $C_y$-partitioned model, using the procedure described above. Four thousand candidate anchor output locations were randomly distributed over the $xy$-plane (approximately 1 percent of the 443,120 total output locations). The singular values in the pseudo-inverse step for production of the $C_y$-partitioned system matrices (as defined in Equation (24)) were progressively reduced until a stable system was obtained. A stable system was obtained when 71 singular values were retained, defining a reduced-order $C_y$-partitioned system of order 71. The resulting $\bar{A}_R$ is 71 x 71, $\bar{B}_R$ is 71 x 1, and $\bar{C}_R$ is 443,120 x 71. To validate the $C_y$-partitioned model, the filtered random input signal shown in Figure 2 was applied, and the Matlab `dlsim` function was used to evolve all 443,120 outputs over the 0.3866 s input time period. Figure 3 presents a succession of images of wave-fronts from the filtered random input interacting with the topology, as computed by the HPC. Figure 4 presents similar images as computed using the $C_y$-partitioned model on a typical laptop computer of average

![Figure 3](image-url)
performance (a MacBook Pro, 2.2 GHz Intel i7, 8 GB 1333 MHz DDR3).

Figure 4. Snapshots of the time-wise evolution of \( \epsilon_{yy} \) in the \( xy \)-plane as computed by the \( Cy \)-partition model running on a laptop computer in response to the filtered random input of Figure 2. The rectangular areas outlined in red define depressions in the topology representing generic structure foundations. The 2-dimensional output surface (the \( xy \)-plane) underlies the topology at an offset of 1 m. The output array consists of 580 points in the \( x \)-direction and 764 points in the \( y \)-direction, totaling 443,120 points, spaced at 0.2 m intervals. These results are essentially identical to the ‘truth’ results shown in Figure 3.

Figures 5 and 6 show respectively the integrated \( y \)-direction normal strain output distribution as computed by the HPC, and as computed from the reduced-order \( Cy \)-partitioned model by the laptop computer at the end of 256 timesteps, given the imposed filtered random input signal. The agreement between the reduced-order \( Cy \)-partitioned model and the HPC truth is seen to be excellent, in fact, effectively exact. The computational requirement for the 443,120 outputs over 256 time steps for the reduced-order \( Cy \)-partitioned model are on the order of 1 minute on the typical laptop computer.

Figures 7 and 8 compare time histories at 10 specific locations, identified in Figure 10. Figure 9 plots the difference between the true output history of Figure 7 and the \( Cy \)-partitioned model output history of Figure 8 and shows that this is typically -30 dB, or roughly 0.1%. 

10
Figure 5. Map of the summed squared $y$-direction normal strain $\epsilon_{yy}$ in the 443,120 output locations of the $xy$-plane over the 256 time step duration of the filtered random input shown in Figure 2, as computed by the HPC FDTD simulation. This is considered to be the truth model for this system.
Figure 6. Map of the summed squared $y$-direction normal strain $\epsilon_{yy}$ in the $xy$-plane over the 256 time step duration of the filtered random input shown in Figure 2, as computed by laptop computer using the order-71 reduced-order $C_{yy}$-partitioned model. This result and that of Figure 5 can be seen to be essentially pixel-for-pixel identical.
Figure 7. Time history of the truth model $\gamma$-direction normal strain $\epsilon_{yy}$ at 10 specific locations as computed by the HPC FDTD simulation, under the filtered random input of Figure 2.
Figure 8. Time history of $y$-direction normal strain at the same 10 specific locations, as computed by the laptop computer using the $C_y$-partitioned model, under the filtered random input of Figure 1.
<table>
<thead>
<tr>
<th>ch#</th>
<th>RMS power (dB rel 1 με)</th>
<th>( \mu_\mathit{F,yy} )</th>
<th>p2p</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.05</td>
<td>0.1</td>
<td>0.15</td>
</tr>
<tr>
<td>-122</td>
<td>5.2e-06</td>
<td>4.7e-06</td>
<td>4.5e-06</td>
</tr>
</tbody>
</table>

Figure 9. Error defined as the relative magnitude of the difference between \( \epsilon_{yy} \) as computed by the truth model and as computed by the laptop computer using the \( C_{yy} \)-partitioned model. This difference is typically -30 dB relative to the output level, corresponding to an error of 0.1%.
Figure 10. Key to the output locations plotted in Figures 7-9. Although the output signal amplitude is small in these areas, these locations visually evidenced the most extreme error levels.
CONCLUSION

A time-domain method for the identification of systems with a very large number of outputs has been presented. The method deals with the specific situation where the number of outputs is orders of magnitudes larger than the minimum state dimension of the system. For example, a system might have a million output nodes and a few inputs to model the vibrations propagating through a large structure, or an acoustic or seismic source propagating through a complex terrain. This is a regime in which existing time-domain system identification methods cannot be applied without proper specialization to handle the extreme computational power and memory storage requirements. The essence of the $Cy$-partitioned models is the realization that in spite of the complexity of the dynamics, the identification problem is not one that treats the dynamical relationship from each input to every output as independent. Instead, there is only a finite number of independent relationships between the input to a set of anchored outputs, and the dynamic responses of all remaining outputs can be inferred from the dynamic responses of the anchored outputs via a static relationship. This is a mathematical fact, which itself is not surprising, but its implication for the identification problem is significant.

Taking advantage of this mathematical feature, various $Cy$-partitioned models and techniques to identify them from input-output data are developed. A simple procedure to reduce the dimensions of the identified $Cy$-partitioned models is also presented, including a technique where the reduced-order models can be identified directly from input-output measurements. The method was applied to find low-dimensional seismic propagation models that produce extremely accurate results when compared to those obtained from complex HPC simulation.

ACKNOWLEDGMENTS

This study was in support of the Enhanced Linear Sensors for Persistent Intelligence, Surveillance, and Reconnaissance project performed at the U.S. Army Engineer Research and Development Center. Computational support was from the U.S. Department of Defense High Performance Computing Modernization Program. Support to Thayer School at Dartmouth was from a Small Business Technology Transfer (STTR) grant by the U.S. Department of the Army.

REFERENCES


1, 88–95 (1992).
Model Identification Class of Algorithms. International Journal of Control. 56, No. 5, 1187–1210
Large Domain Wave Propagation. Modeling, Simulation, and Optimization of Complex Process. Bock,
Propagation. (in preparation).