# LINEAR DYNAMICAL MODELS IN SPEECH SYNTHESIS

*Vassilis Tsiaras*[1], *Ranniery Maia*[2], *Vassilis Diakoloukas*[1], *Yannis Stylianou*[2], *Vassilis Digalakis*[1]

[1]Technical University of Crete, School of Electronic and Computer Engineering, Chania, Greece
[2]Toshiba Research Europe Limited, Cambridge Research Laboratory, Cambridge, UK

## ABSTRACT

Hidden Markov models (HMMs) are becoming the dominant approach for text-to-speech synthesis (TTS). HMMs provide an attractive acoustic modeling scheme which has been exhaustively investigated and developed for many years. Modern HMM-based speech synthesizers have approached the quality of the best state-of-the-art unit selection systems. However, we believe that statistical parametric speech synthesis has not reached its potential, since HMMs are limited by several assumptions which do not apply to the properties of speech. We, therefore, propose in this paper to use Linear Dynamical Models (LDMs) instead of HMMs. LDMs can better model the dynamics of speech and can produce a naturally smoother trajectory of the synthesized speech. We perform a series of experiments using different system configurations to check on the performance of LDMs for speech synthesis. We show that LDM-based synthesizers can outperform HMM-based ones in terms of cepstral distance and are a very promising acoustic modeling alternative for statistical parametric TTS.

***Index Terms***— Statistical parametric speech synthesis, Linear dynamical model, Kalman filter

## 1. INTRODUCTION

The dominant methods in statistical parametric speech synthesis are based on hidden Markov models (HMMs). Natural sounding speech has been synthesized with HMMs and the quality of the best HMM-based synthesis systems approaches the quality of the best unit selection synthesis systems [1]. However, although HMMs can be a relatively efficient modeling scheme for speech, they suffer from a number of limitations that have been pointed out in the literature[2, 3]. The HMM limitations derive from assumptions such as: a) conditional independence of observations given the state sequence and b) speech statistics of each state do not change dynamically. A simple mechanism for capturing time dependence is to augment the observation space with feature derivatives under the false frame-independence assumption. This mechanism was further improved by the trajectory HMM [3], which impose relationships between static and dynamic feature vector sequences. Alternatively, a variety of models have also been proposed to explicitly capture the dynamics of speech, e.g. [4, 2, 5, 6].

It is our belief that improving the HMM modeling scheme has reached its limits and we need to investigate novel acoustic models in order to make considerable progress in statistical speech synthesis. We therefore examine in this work an entirely different model, the Linear Dynamical Models (LDMs) (also known as Kalman filter models) for speech synthesis. LDMs are probabilistic, state space models, which explicitly model some of the dynamics of speech and introduce the continuity and context dependence needed for good quality synthesis. Temporal dynamics is modeled as a smooth, continuous motion in a hidden state space, which is then projected onto the observation space.

One of the attractive properties of these models is that they may readily be trained via the EM algorithm in a maximum likelihood framework [4]. Although LDMs have been used in speech recognition in the past there is only a single effort utilizing them in speech synthesis by Quillen [7]. However, the work in [7] is only a preliminary effort which we further extend. Specifically in this work, the spectral envelope of speech is modeled with LDMs and a number of design and algorithmic choices have been evaluated.

This paper is organized as follows. Section 2 gives an outline of LDMs; Section 3 shows how LDMs can be used for statistical parametric synthesis; in Section 4 we present some experiments; and in Section 6 we conclude this paper.

## 2. LINEAR DYNAMICAL MODELS

The LDMs are the simplest dynamical models with continuous state vectors. The state evolution process is a linear first-order Gauss-Markov random process while the observation process is a factor analyzer. The output of the process follows a time-varying multivariate Gaussian distribution. An LDM can be specified by the following equations:

$$\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \tag{1a}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{w}_k, \quad \boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{\mu}_k^s, \boldsymbol{Q}_k) \tag{1b}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k, \quad \boldsymbol{v}_k \sim \mathcal{N}(\mu_k^o, \boldsymbol{R}_k) \tag{1c}$$

where $\boldsymbol{F}_k$ is a $n \times n$ state transition matrix and $\boldsymbol{H}_k$ is a $m \times n$ observation matrix. The state $\boldsymbol{x}$ is an $n$-dimensional vector

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

which evolves according to linear difference equation (1b), with initial condition $\boldsymbol{x}_0$. The state cannot be observed directly. Instead, $m$-dimensional measurements $\boldsymbol{y}$ are available at discrete sampling times as described by (1c). The vectors $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are called state evolution noise and observation noise respectively and are independent of each other and temporally uncorrelated (i.e., $E[\boldsymbol{w}_k \boldsymbol{w}_l^\top] = \boldsymbol{Q}_k \delta_{kl}$ and $E[\boldsymbol{v}_k \boldsymbol{v}_l^\top] = \boldsymbol{R}_k \delta_{kl}$, where $\delta_{kl}$ is the Kronecker delta).

## 3. LDM APPLIED TO SPEECH SYNTHESIS

### 3.1. Parameter estimation

Equation (1) defines a time varying system and this formulation may introduce too many parameters. By assuming that within sub-phone segments the output distribution evolves in a linear, continuous fashion, the parameters can be set to the same values (i.e., $\boldsymbol{F}_k = \boldsymbol{F}$, $\boldsymbol{H}_k = \boldsymbol{H}$, ... for all samples of a segment). Non-linearities in the output distribution are incorporated at segment boundaries where the parameters change. To ensure that the state process remains continuous through such shifts, first and second order state statistics should be passed across segment boundaries. Algorithms 1 and 2 are high level descriptions of parameter estimation procedures. The symbol $\boldsymbol{\Sigma}_{k/k-1}$ refers to the covariance of the prediction error at time $k$, given $k-1$ observations, with $\boldsymbol{\Sigma}_{1/0}$ being a reasonable initial value. Both algorithms have as inputs the observations and the corresponding labels and they calculate the parameters of an LDM model for each label. In Algorithm 1 the forward and backward recursion are run in sub-phoneme level, while in Algorithm 2 these recursions are run in phoneme level. Note that the statistics are passed within phoneme bounds and the state is passed across a utterance. The equations of the forward and backward recursions (Kalman filter and smoother), of the sufficient statistics and of the parameters modification step can be found in [4]. These equations assume a single run of output observations. However, their extension to multiple output runs is simple and involves summing the sufficient statistics over the different runs.

**Algorithm 1:** EM Algorithm - Segment Level

```
foreach label do
    while not converged do
        // E-step
        zero the statistics variables
        foreach segment do
            Initialize x_0 and Σ_{1/0} = Σ_0
            run the forward and backward recursions
            accumulate the statistics
        // M-step
        Update the parameters using the statistics
```

**Algorithm 2:** EM Algorithm - Phoneme Level

```
while not converged do
    // E-step
    foreach label do
        zero the statistics variables
    foreach utterance do
        Initialize x_0
        foreach phoneme do
            Let L_1, L_2, L_3, L_4, L_5 be the
              sub-phoneme labels
            Initialize Σ_{1/0} = Σ_0 of label L_1
            run the forward and backward recursions
            accumulate the statistics per label
            x_0 = x_last
    // M-step
    foreach label do
        Update the parameters using the corresponding
        statistics
```

### 3.2. Initialization

Sensible initialization of parameter $\boldsymbol{F}$ of an LDM is crucial, since the EM solution highly depends on its initial estimate. When the dimension of the state space is equal to the dimension of the observation space, then we can assume that initially $\boldsymbol{x}_k \equiv \boldsymbol{y}_k$. Therefore, an initial estimate of $\boldsymbol{F}$ is given by $\boldsymbol{F} = \boldsymbol{\Gamma}_4 \boldsymbol{\Gamma}_3^{-1}$, where

$$\boldsymbol{\Gamma}_4 = \sum_{k=1}^{T-1} \boldsymbol{y}_{k+1} \boldsymbol{y}_k^\top - \frac{1}{T-1} \sum_{k=1}^{T-1} \boldsymbol{y}_{k+1} \sum_{k=1}^{T-1} \boldsymbol{y}_k^\top, \quad (2)$$

$$\boldsymbol{\Gamma}_3 = \sum_{k=1}^{T-1} \boldsymbol{y}_k \boldsymbol{y}_k^\top - \frac{1}{T-1} \sum_{k=1}^{T-1} \boldsymbol{y}_k \sum_{k=1}^{T-1} \boldsymbol{y}_k^\top, \quad (3)$$

and $T$ is the number of observations. This estimate applies to a single time series but can easily be extended to multiple experiments case. Parameter $\boldsymbol{H}$ was set to a random matrix. Another strategy, which is applicable when the state space is less than or equal to the observation space, is to use subspace identification methods [8] to estimate $\boldsymbol{F}$ and $\boldsymbol{H}$. When the dimension of the state space is greater than the dimension of the observation space then the initialization of $\boldsymbol{F}$ requires knowledge of the mapping between the two spaces.

### 3.3. Constraints

One constraint is enforced during training: The spectral radius, $\rho(\boldsymbol{F})$ of $\boldsymbol{F}$ is constrained to be less than or equal to one, i.e., $\rho(\boldsymbol{F}) \leq 1$. If $\rho(\boldsymbol{F}) > 1$ were allowed, the state evolution could give a model of exponential growth. Such behavior may not be apparent over small numbers of frames, while still introducing an element of numerical instability. In

this work, two methods of constraining $\rho(\boldsymbol{F})$ have been implemented. The first one employs the eigenvalue decomposition, replaces each eigenvalue with magnitude greater than one with an eigenvalue that has magnitude less than or equal to one and then reconstructs $\boldsymbol{F}$. The second method adds inequality constraints in the EM auxiliary function and is described in [7]. Most of the experiments performed in this work use the second method, although there is no noticeable difference in the quality of the synthesized speech if the first method is used instead.

### 3.4. Speech parameter generation algorithm

To synthesize speech, a file, that contains sub-phoneme segment labels and durations, is presented as input. The proposed model synthesizes an entire utterance at a time by stepping the following equations for each phoneme of the utterance according to the duration model:

$$\begin{aligned} \boldsymbol{x}_1 &= \rho\boldsymbol{\mu}_0 + (1-\rho)\boldsymbol{x}_{prev} \\ \boldsymbol{y}_k &= \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{\mu}_k^o \\ \boldsymbol{x}_{k+1} &= \boldsymbol{F}_k\boldsymbol{x}_k + \boldsymbol{\mu}_k^s \end{aligned} \tag{4}$$

where $0 \leq \rho \leq 1$, $\boldsymbol{x}_{prev}$ is the last state of the previous phoneme and $\boldsymbol{\mu}_0$ is the initial state of the current phoneme. In this work $\rho$ was set to 0.5. The state is passed across all iterations while the parameters $\boldsymbol{F}_k$, $\boldsymbol{H}_k$, $\boldsymbol{\mu}_k^o$ and $\boldsymbol{\mu}_k^s$ are chosen according to the current segment label. The derived features $\boldsymbol{y}_k$ are then used to synthesize a waveform.

## 4. EXPERIMENTS

### 4.1. Speech corpus and parameter extraction

The Edinburgh speech synthesis database release for the Hurricane Challenge [9] was used. The training was done using 2551 utterances sampled at 16 kHz, selected from the *herald* and *hvd* sets. Full context labels were created by using the Festival Speech Synthesis [10] frontend. From the training utterances, 40 mel-cepstral coefficients were extracted at every 5 ms using the complex cepstrum analysis method shown in [11]. Mel-cepstral coefficients were regarded as the minimum-phase cepstrum component of the complex cepstrum.

### 4.2. Speech segmentation

Mel-cepstral parameters were used to train a baseline synthesizer based on HMM using state-of-the-art configurations as follows. Each full context model was represented by a five-state left-to-right hidden semi-Markov model (HSMM) [12], and decision tree clustering using minimum description length criterion was used to cluster HSMM states [13]. In the end of the clustering, the 340,885 states were tied to produce 965 states, representing a reduction to 0.2%. The observation vectors were consisted of 40 mel-cepstral coefficients, delta and delta-delta.

The trained HSMMs were used to segment the database at the state level using the method described in [14]. Fig. 1 shows the histogram of the 4th mel-cepstral coefficient of the observations belonging to a given tree leaf. Note that the histogram follows approximately a Gaussian distribution. The terminal nodes of the decision trees and their respective observations were used to train the LDMs.

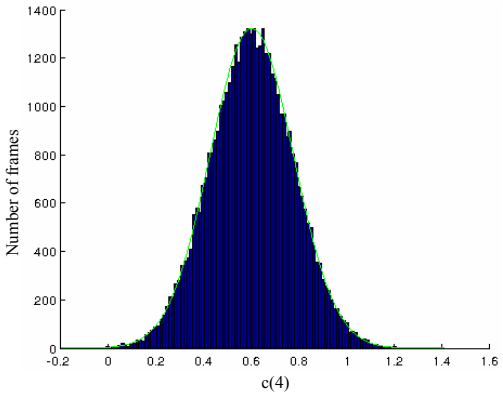


**Fig. 1**. Histogram of the 4th mel-cepstral coefficient taken from the observations belonging to a given terminal node of the decision trees for mel-cepstral coefficients.

### 4.3. Results

A large number of LDM training configurations were performed and the difference between natural and generated cepstra were measured. The differences in terms of LDM training regarded the following: a) the number of HSMM states per phoneme considered; b) the state space dimension; c) the algorithm that is used for the estimation of the parameters (Algorithm 1 or Algorithm 2); d) whether $\boldsymbol{\mu}^o$ is set to zero; e) whether $\boldsymbol{\mu}^s$ is set to zero; f) whether $\boldsymbol{Q}$ is diagonal; g) whether $\boldsymbol{R}$ is diagonal; h) whether the constraint, $\rho(\boldsymbol{F}) \leq 1$, was applied or not. The evaluation of the models was based on the mean value of the cepstral distance and the raw PESQ [15] score metrics, which were applied to 50 randomly selected utterances. The cepstral distance in dB between two sequences of mel-cepstral coefficients sets is given by

$$d\left(\boldsymbol{c}_1, \boldsymbol{c}_2\right) = \frac{10}{T \ln 10} \sum_{t=0}^{T-1} \sqrt{\sum_{i=1}^{m} \left[c_{t,1}(i) - c_{t,2}(i)\right]^2}, \tag{5}$$

where $c_{t,1}(i)$ and $c_{t,2}(i)$ are the $i$-th mel-cepstral coefficient for the $t$-frame of the natural and generated sequence of coefficients sets, respectively, with $T$ being the number of frames and $m$ the cepstrum order. Smaller distances correspond to better modeling. On the other hand the raw PESQ score measures the perceptual similarity between the original and the

**Table 1**. Cepstral distances and PESQ scores for each of LDM training methods, represented by models $M_1, \ldots, M_6$. $n$ is the state space dimension. The best measures are highlighted in boldface. Cepstral distances are in dB.

| Model | $M_1$ | $M_2$ | $M_3$ | $M_4$ | $M_5$ | $M_6$ |
|---|---|---|---|---|---|---|
| states | 5 | 5 | 5 | 5 | 5 | 5 |
| $n$ | 40 | 15 | 40 | 40 | 80 | 30 |
| Algorithm | 1 | 1 | 1 | 1 | 1 | 1 |
| $\boldsymbol{\mu}^s \neq 0$ | | | | | | |
| diag $\boldsymbol{Q}$ | ✓ | ✓ | | | | |
| diag $\boldsymbol{R}$ | ✓ | ✓ | | ✓ | | |
| Cep. dist. | **5.62** | **5.62** | 5.63 | 5.63 | 5.63 | 5.63 |
| PESQ | **2.75** | **2.75** | 2.74 | **2.75** | 2.74 | 2.74 |

**Table 2**. Cepstral distances and PESQ scores for each of LDM training methods, represented by models $M_7, \ldots, M_{12}$. $n$ is the state space dimension. The best measures are highlighted in boldface. Cepstral distances are in dB.

| Model | $M_7$ | $M_8$ | $M_9$ | $M_{10}$ | $M_{11}$ | $M_{12}$ |
|---|---|---|---|---|---|---|
| states | 3 | 5 | 5 | 5 | 5 | 5 |
| $n$ | 40 | 40 | 40 | 80 | 40 | 40 |
| Algorithm | 1 | 1 | 1 | 1 | 2 | 2 |
| $\boldsymbol{\mu}^s \neq 0$ | | | ✓ | ✓ | | ✓ |
| diag $\boldsymbol{Q}$ | | ✓ | | | | |
| diag $\boldsymbol{R}$ | | | | | | |
| Cep. dist. | **5.64** | 5.65 | 5.65 | 5.68 | 5.69 | 5.72 |
| PESQ | 2.67 | **2.73** | 2.72 | 2.68 | 2.71 | 2.62 |

synthesized waveform and higher scores correspond to better modeling. To calculate the PESQ scores, speech was synthesized from generated cepstrum, aligned durations, and natural $F_0$, by passing a simple excitation signal through the mel log spectrum approximation filter [16].

Tables 1 and 2 show the cepstral distances and PESQ scores for each training configuration. From the results of the experiments it is inferred that: a) Algorithm 1 is superior to Algorithm 2; b) 5 states per full context model seems to lead to better models than 3 states; c) the $\boldsymbol{\mu}^o$ parameter is necessary; d) the $\boldsymbol{\mu}^s$ parameter should be omitted (i.e., set to zero); e) one can safely assume that matrix $\boldsymbol{R}$ is diagonal. Therefore it is possible to use simpler, faster and numerically more stable inference algorithms such as the sequential Kalman filtering. As far as the state space dimension is concerned, the quality of the synthesized speech does not deteriorate if the state space dimension becomes smaller than the observation space dimension (up to $n = 10$). For state space dimension greater than the observation space dimension more research is needed before a definite conclusion can be drawn, because in this case the identifiability issue [17, 18] as well as the data insufficiency issue have to be addressed.

The mean cepstral distance and PESQ score between natural and mel-cepstral parameters generated from the HSMMs are 5.94 dB and 2.41, respectively. Figure 2 shows trajectories of 2-th mel-cepstral coefficient. The $c(2)$ of LDM (red line) is closer to true $c(2)$ (blue line) when compared with the $c(2)$ generated from HSMM (green line).

## 5. RELATION TO PRIOR WORK

The present results are generally consistent with findings from a recent study of Quillen [7] employing LDMs in speech synthesis. However, this work further extends the results of Quillen [7] by evaluating many modeling alternatives. It was shown that the five state segmentation of a phoneme gives

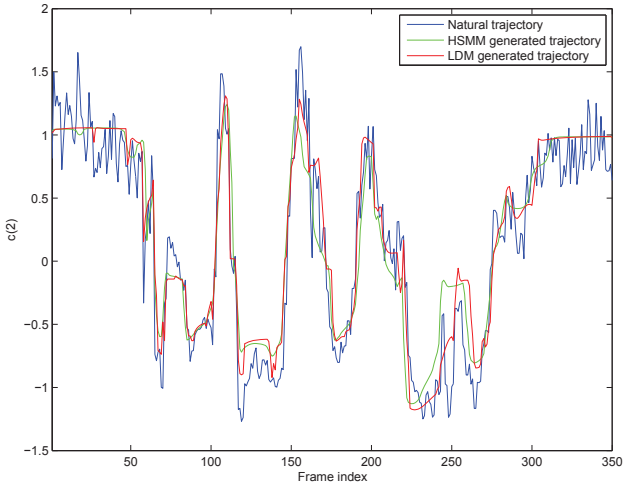


**Fig. 2**. Trajectories of the 2-th mel-cepstral coefficient.

better synthesized speech than the three state segmentation as it was expected. On the other hand, it was unexpected that Algorithm 1 performs better than Algorithm 2.

## 6. CONCLUSION AND FUTURE STEPS

The LDMs are promising acoustic models in the effort to produce natural speech with parametric statistical models. The preliminary results of this study suggest that LDM produced spectral parameters that are closer to their natural versions. According to informal listening, speech synthesized with LDM-generated cepstra, HSMM-aligned durations and natural $F_0$ sounds less muffled than speech synthesized from HSMM-generated cepstrum. In the future we plan to apply LDM to model $F_0$, duration and band-aperiodicity, and perform a formal evaluation of the synthesized speech.

## 7. REFERENCES


[1] Heiga Zen, Keiichi Tokuda, and Alan W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[2] Mari Ostendorf, Vassilios Digalakis, and Owen A. Kimball, "From HMMs to segment models: A unified view of stochastic modeling for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 360–378, 1995.

[3] Heiga Zen, Keiichi Tokuda, and Tadashi Kitamura, "Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences," *Computer Speech and Language*, vol. 21, no. 1, pp. 153–173, 2007.

[4] Vassilios Digalakis, J.R. Rohlicek, and Mary Ostendorf, "Ml estimation of a stochastic linear system with the em algorithm and its application to speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 4, pp. 431–442, 1993.

[5] Matt Shannon and William Byrne, "A formulation of the autoregressive HMM for speech synthesis," Tech. Rep. CUED/F-INFENG/TR.629, University of Cambridge, Department of Engineering, 2009.

[6] Carl Quillen, "Autoregressive HMM synthesis," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2012*, 2012, pp. 4021–4024.

[7] Carl Quillen, "Kalman filter based speech synthesis," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010*. 2010, pp. 4618–4621, IEEE.

[8] Peter Van Overschee and Bart De Moor, *Subspace identification for linear systems. Theory, implementation, applications*, Kluwer Academic publ, Boston, 1996.

[9] "Data release for the hurricane challenge," `http://www.cstr.ed.ac.uk/projects/hurricane/1/index.html`, Last visited: 4th November 2013.

[10] "The Festival Speech Synthesis System," `http://www.festvox.org/festival`, Last visited in October 2013.

[11] R. Maia, M.J.F. Gales, Y. Stylianou, and M. Akamine, "Minimum mean squared error based warped complex cepstrum analysis for statistical parametric speech synthesis," in *Proceedings of the Annual Conference of the International Speech Communication Assiciation, Interspeech 2013*, 2013, pp. 2336–2340.

[12] Heiga Zen, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura, "A hidden semi-Markov model based speech synthesis system," *IEICE Transactions on Information and Systems*, vol. E90-D, no. 5, pp. 825–834, May 2007.

[13] K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," in *Proceedings of the European Conference on Speech Communication and Technology, Eurospeech 1997*, 1997, pp. 99–102.

[14] Keiichiro Oura, Heiga Zen, Yoshihiko Nankaku, Akinobu Lee, and Keiichi Tokuda, "A fully consistent hidden semi-markov model-based speech recognition system," *IEICE - Transactions in Information and Systems*, vol. E91-D, no. 11, pp. 2693–2700, Nov. 2008.

[15] Antony W. Rix, John G. Beerends, Michael P. Hollier, and Andries P. Hekstra, "Perceptual evaluation of speech quality (PESQ) - a new method for speech quality assessment of telephone networks and codecs," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2001*. 2001, pp. 749–752, IEEE.

[16] Keiichi Tokuda, Takao Kobayashi, and Satoshi Imai, "Adaptive cepstral analysis of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 6, pp. 481–489, Nov. 1995.

[17] D. G. Luenberger, "Canonical forms for linear multivariate systems," *IEEE Transcations on Automatic Control*, vol. 12, no. 3, pp. 290–293, 1967.

[18] Edison T. S. Tse, Howard L. Weinert, John J. Anton, and Raman K. Mehra, *Model Structure Determination and Identifiability Problems in System Identification*, Office of Naval Research, 1973.