Cepstral Coefficients, Covariance Lags, and Pole-Zero Models for Finite Data Strings

Christopher I. Byrnes, *Fellow, IEEE*, Per Enqvist, and Anders Lindquist, *Fellow, IEEE*

***Abstract*—One of the most widely used methods of spectral estimation in signal and speech processing is linear predictive coding (LPC). LPC has some attractive features, which account for its popularity, including the properties that the resulting modeling filter i) matches a finite window of $n + 1$ covariance lags, ii) is rational of degree at most n, and iii) has stable zeros and poles. The only limiting factor of this methodology is that the modeling filter is "all-pole," i.e., an autoregressive (AR) model.**

In this paper, we present a systematic description of all autoregressive moving-average (ARMA) models of processes that have properties i)–iii) in the context of cepstral analysis and homomorphic filtering. Indeed, we show that each such ARMA model determines and is completely determined by its finite windows of cepstral coefficients and covariance lags. This characterization has an intuitively appealing interpretation of a characterization by using measures of the transient and the steady-state behaviors of the signal, respectively. More precisely, we show that these nth-order windows form local coordinates for all ARMA models of degree n and that the pole-zero model can be determined from the windows as the unique minimum of a convex objective function. We refine this optimization method by first noting that the maximum entropy design of an LPC filter is obtained by maximizing the zeroth cepstral coefficient, subject to the constraint i). More generally, we modify this scheme to a more well-posed optimization problem where the covariance data enters as a constraint and the linear weights of the cepstral coefficients are "positive"—in a sense that a certain pseudo-polynomial is positive—rather succinctly generalizing the maximum entropy method. This new problem is a homomorphic filter generalization of the maximum entropy method, providing a procedure for the design of any stable, minimum-phase modeling filter of degree less or equal to n that interpolates the given covariance window.

We conclude the paper by presenting an algorithm for realizing these filters in a lattice-ladder form, given the covariance window and the moving average part of the model. While we also show how to determine the moving average part using cepstral smoothing, one can make use of any good *a priori* estimate for the system zeros to initialize the algorithm. Indeed, we conclude the paper with an example of this method, incorporating an example from the literature on ARMA modeling.

***Index Terms*—Autoregressive moving average processes, cepstral analysis, covariance analysis, identification, maximum entropy methods, optimization methods, spectral analysis, speech analysis.**

Manuscript received July 6, 1998; revised December 4, 2000. This work was supported in part by grants from the Air Force Office of Scientific Research, the Swedish Research Council for Engineering Sciences, the Göran Gustafsson Foundation, and Southwestern Bell. The associate editor coordinating the review of this paper and approving it for publication was Prof. Arnab K. Shaw.

C. I. Byrnes is with the Department of Systems Science and Mathematics, Washington University, St. Louis, MO 63130 USA.

P. Enqvist and A. Lindquist are with the Division of Optimization and Systems Theory, Department of Mathematics, Royal Institute of Technology, Stockholm, Sweden (e-mail: alq@math.kth.se)

Publisher Item Identifier S 1053-587X(01)02242-5.

I. Introduction

A PURELY nondeterministic (zero-mean) Gaussian stationary process is fully characterized by its infinite sequence of covariance lags, which are in fact the Fourier coefficients of its spectral density. In particular, if this density is rational, an infinite sample of such a process determines a unique autoregressive moving-average (ARMA) model for the process, leading to an explicit description of a spectral density and to a modeling filter that can regenerate the entire process in the sense that it shapes white noise into a process with the same covariance sequence. These equivalent models of the process play a fundamental role in spectral estimation [11], system identification [2], [23], [35], [39], speech processing [13], [24], [26], [32], [33], and several other applications in signal processing and systems and control [8]–[10], [16], [19], [20]. However, one never really has an infinite sample of a time series but rather a finite window of data, leading to a far more complicated set of modeling issues. However, from this data, one can estimate a window of approximate covariances, having a rather appealing and useful interpretation as moments that approximate the true covariances [22].

The basic inverse problem with which we begin, then, is that of finding a spectral density, positive on the unit circle, matching a finite covariance sequence that is positive definite in an appropriate sense [19], [31]. An autoregressive (AR) solution to this problem is provided by the linear predictive coding (LPC) filter [31]. The LPC filter can be realized by an all-pole lattice filter architecture, whose gains can be easily computed using standard algorithms. Nonetheless, the need for ARMA modeling, incorporating both poles and zeros, has long been understood in spectral estimation and signal processing [18], [21], [37], [38]. For example, in speech processing, Atal has pointed out that the perceived differences between real speech and the best synthetic speech obtainable using an LPC filter are at least partially due to the all-pole model restriction [26, p. 271], which limits its power spectral density from matching the "nulls," or "notches," in the periodogram of the data. Indeed, it is widely appreciated in the speech processing literature that regeneration of certain features of human speech, for example, sounds involving fricatives or nasals, requires the design of filters having zeros (see, e.g., [5, p. 1726], [26, pp. 271–272], and [33, pp. 105, 76–78]).

On the other hand, we are interested in those ARMA models that match a given window of covariances. To this end, we note that an alternative approach to pole-zero modeling is offered by cepstral analysis and homomorphic filtering [29], where pole-zero models are determined from the Fourier coefficients of the logarithm of the spectral density: the so-called cepstral

coefficients. As it turns out, the information contained in a finite window of cepstral coefficients is complementary to the information contained in a finite window of covariances so that by combining cepstral analysis with covariance methods, one can actually obtain a solution to the problem of covariance extension. Indeed, one of the main results obtained in this paper is that any pole-zero model of fixed degree determines, and is uniquely determined by, a prescribed window of cepstral coefficients and a prescribed window of covariances. This result has a number of amplifications and new consequences for pole-zero modeling of observed data.

We begin in Section II by setting notation and reviewing the derivation of linear predictive coding filters from a covariance window or, what is equivalent, a window of PARCOR coefficients. We then examine the LPC filter in terms of cepstral analysis, obtaining an interesting interpretation of its maximum entropy filter aspects in terms of maximization of cepstral gain, which is a problem that we later generalize in a substantial way.

In Section III, we present our first main result, *viz.*, that the finite cepstral and covariance windows can be interpreted as coordinates for stable, minimum-phase pole-zero models of fixed degree. In particular, we give formulae for the cepstral coefficients in terms of differences of Newton sums of the poles and of the zeros, generalizing the usual formulae in terms of Newton sums of the poles for LPC filters. We then show how a modeling filter of degree n arises as the unique minimum of an optimization scheme involving cepstral and covariance functionals together with a generalized entropy integral. After illustrating this minimization scheme in a simple example from the literature, we modify this scheme to a more well-posed optimization problem, where the covariance data enters as a constraint, and the linear weights of the cepstral coefficients are "positive"—in a sense that a certain pseudo-polynomial is positive, rather succinctly generalizing the maximum entropy method. This new problem is a homomorphic filter generalization of the maximum entropy method, leading to the design of all stable, minimum-phase modeling filters of degree n that interpolate the given covariance window. We conclude Section III with an illustration taken from speech synthesis.

ARMA processes can be realized as lattice-ladder filters, enhancing the lattice description of the AR model given by the LPC filter. In Section IV, we also show how the generalized maximum entropy method leads to a convex optimization scheme that uniquely determines the parameters in the lattice-ladder filter, given the window of covariance lags and the choice of positive pseudo-polynomial in the generalized maximum entropy problem. It is also noted that by spectral factorization, the choice of positive pseudo-polynomial corresponds to a choice of stable zeros of the numerator polynomial for the modeling filter. Thus, this homomorphic filtering generalization of maximum entropy methods gives a new derivation, based on cepstral analysis, of the recent resolution of the rational covariance extension problem. Briefly, in the early 1980s, Georgiou [16] proved the remarkable result that for a given covariance window, ARMA filters exist for *any* choice of stable zeros (inside the unit disc). Georgiou also conjectured the uniqueness of such modeling filters, which was an issue finally resolved in the positive in [8].

Since these filters can be realized in lattice-ladder form, and since this provides a design method for deriving modeling filters matching a covariance window but having arbitrary stable zeros (or "notches" in the power spectrum of the ARMA model), these filters are referred to as "lattice-ladder notch" (LLN) filters. Thus, the class of LLN filters coincide with the class of linear modeling filters, of degree at most n, which shape white noise into a process with the observed covariance data. This is illustrated using refinements of the spectral estimates developed in Section III for a frame of unvoiced speech.

Finally, we wish to emphasize that the algorithm presented here provides a new computational scheme for realizing these filters, given the covariance window and the moving average part of the model. While we also show how to determine the moving average part using cepstral smoothing, any *a priori* estimate of the zero polynomial can be used as an initial condition in our algorithm. In particular, one can make use of any ARMA modeling estimate for the system zeros to initialize an enhancement of the modeling filter as well as to obtain better covariance matching. Indeed, we conclude the paper with an example of this method, incorporating the ARMA modeling techniques of [35] to obtain an initial estimate of the system zeros.

II. Preliminaries

A fundamental problem in systems and signals is to determine a model for a stationary random process $\{y(t)\}$ from a finite window of data. A linear model for the data would consist either of a state-space model having the process as an output or of a description of the z-transform of the system, which, of course, can be determined by the filter zeros, poles, and the high frequency gain. Among the popular approaches to this problem are an approach based on matching the covariance lags—notably linear predictive coding, which is an approach based on cepstral methods and homomorphic filtering, and approaches based on variants of system identification methods for autoregressive (AR) or autoregressive moving-average (ARMA) models.

A. Analysis Based on Infinite Data

Of course, if one had an infinite data record, the problem would be much easier. For example, it is well known that the spectral density $\Phi(z)$ of a (purely nondeterministic) stationary random process $\{y(t)\}$ is given by the Fourier expansion

$$\Phi\left(e^{j\theta}\right) = \sum_{k=-\infty}^{\infty} r_k e^{jk\theta} \tag{2.1}$$

on the unit circle, where the covariance lags

$$r_k = E\{y(t+k)y(t)\} \qquad k = 0, 1, 2, \cdots \tag{2.2}$$

satisfy

$$r_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{jk\theta}\, \Phi(e^{j\theta})\, d\theta. \tag{2.3}$$

Consequently, the unique rational, stable, minimum-phase function $W(z)$ satisfying

$$W(z)W(z^{-1}) = \Phi(z) \tag{2.4}$$

is the transfer function of a *modeling filter*

$$\text{input} \xrightarrow{u} \boxed{W(z)} \xrightarrow{y} \text{output} \tag{2.5}$$

which shapes white noise into a random process with the covariance lags given by (2.3). Here, the rational transfer function

$$W(z) = \frac{B(z)}{A(z)} \tag{2.6}$$

is *minimum-phase*, i.e., the denominator polynomial $A(z)$ and the numerator polynomial $B(z)$ both have their roots strictly inside the unit circle; such a polynomial will be called *stable*.

In particular, given an (infinite) string of observed data

$$y_0, y_1, y_2, y_3, \cdots \tag{2.7}$$

satisfying an appropriate ergodicity property, the covariance sequence $r_0, r_1, r_2, r_3, \cdots$ can be determined as

$$r_k = \lim_{T\to\infty} \frac{1}{T+1} \sum_{t=0}^{T} y_{t+k} y_t \tag{2.8}$$

which defines a unique spectral density and, hence, a unique modeling filter.

Cepstral analysis also provides a means of determining the poles, zeros, and high frequency gain of the modeling filter. The starting point is the Fourier analysis of the (real) *cepstrum*

$$\log \Phi(e^{j\theta}) = \sum_{k=-\infty}^{\infty} c_k e^{jk\theta}. \tag{2.9}$$

The Fourier coefficients

$$c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{jk\theta} \log \Phi(e^{j\theta})\, d\theta \tag{2.10}$$

are known as the *cepstral coefficients*. We note that $c_{-k} = c_k$ for all k. Computing the logarithmic derivative of Φ, it can be seen that the series with Fourier coefficients kc_k has its poles at the poles and zeros of the modeling filter [29]. A second property of the cepstrum is key to homomorphic filtering, *viz.*, if the process $\{y(t)\}$ is the response to a signal $\{u(t)\}$ driving the modeling filter $W(z)$, then the cepstrum of $\{y(t)\}$ is the superposition of the cepstrum of the impulse response of the modeling filter and the cepstrum of the input. In particular, if the impulse response of the filter is slowly varying compared with the input, a finite window of cepstral coefficients will capture most of the information about the filter. Of course, there is a one-to-one correspondence between the *infinite* sequences $r_0, r_1, r_2, \cdots$ and $c_0, c_1, c_2, \cdots$.

B. LPC Filters

In practice, however, only a finite string of observed data

$$y_0, y_1, y_2, \cdots, y_N \tag{2.11}$$

is available, and both of the approaches outlined above need considerable refinement. If N is sufficiently large, the estimate

$$\frac{1}{N+1} \sum_{t=0}^{N-k} y_{t+k} y_t \tag{2.12}$$

for example, is a good approximation of r_k, but now, only a finite covariance sequence

$$r_0, r_1, r_2, \cdots, r_n \tag{2.13}$$

where $n \ll N$, can be produced. However, at least using the covariance estimates (2.12), the Toeplitz matrix

$$T_n = \begin{bmatrix} r_0 & r_1 & \cdots & r_n \\ r_1 & r_0 & \cdots & r_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ r_n & r_{n-1} & \cdots & r_0 \end{bmatrix} \tag{2.14}$$

is positive definite, as required.

Indeed, in spectral estimation [11], identification [2], [23], [39], speech processing [13], [24], [26], [32], [33], and several other applications in signal processing and systems and control [8]–[10], [16], [19], [20], one is typically faced with the inverse problem of finding a spectral density, positive on the unit circle and of degree at most n, given only a sequence (2.13) for which the Toeplitz matrix (2.14) is positive definite. Finding any such spectral density is the same as finding a modeling filter, of degree at most n, which shapes white noise into a process with the observed window of covariance data. An autoregressive solution to this problem is provided by the linear predictive coding (LPC) filter [31], which, as is well-known, can be realized by a lattice filter, containing unit delays z^{-1}, summing junctions, and gains, as illustrated in Fig. 1. The gains $\gamma_0, \gamma_1, \gamma_2, \cdots, \gamma_{n-1}$ and

$$\rho_n := r_0 \prod_{k=0}^{n-1} (1 - \gamma_k^2)$$

are recursively defined from the covariance lags $r_0, r_1, r_2, \cdots, r_n$ via the *Levinson algorithm*

$$\gamma_t = \frac{1}{\rho_t} \sum_{k=0}^{t} \varphi_{t,\,t-k} r_{k+1} \tag{2.15}$$

$$\varphi_{t+1,\,j} = \varphi_{tj} - \gamma_t \varphi_{t,\,t+1-j}; \quad \varphi_{00} = 1 \tag{2.16}$$

$$\rho_{t+1} = \rho_t (1 - \gamma_t^2); \quad \rho_0 = r_0 \tag{2.17}$$

where $\varphi_{t0} = 1$ and $\varphi_{t,\,t+1} = 0$; see, e.g., [26] and [31]. The gains $\gamma_0, \gamma_1, \gamma_2, \cdots, \gamma_{n-1}$ are called *reflection coefficients* or *Schur parameters* and have the property that $|\gamma_k| < 1$ for $k = 0, 1, \cdots, n$.

The reflection coefficients can be determined either directly from the data (2.13) by some version of the Burg's algorithm [31, p. 175] or from the covariance estimates (2.12) via (2.15). The filter has the transfer function

$$W(z) = \frac{\sqrt{\rho_n} z^n}{\varphi_n(z)} \tag{2.18}$$

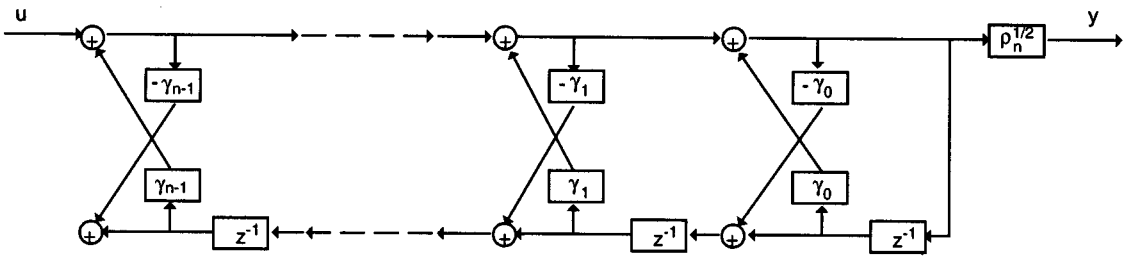

Fig. 1. Lattice filter representation of an LPC filter.

where $\varphi_n(z)$ is the nth Szegö polynomial

$$\varphi_n(z) = z^n + \varphi_{n1}z^{n-1} + \cdots + \varphi_{nn} \tag{2.19}$$

with the coefficients $\{\varphi_{nj}\}$ being defined by (2.15), leading to the AR model

$$y(t) + \varphi_{n1}y(t-1) + \cdots + \varphi_{nn}y(t-n) = \sqrt{\rho_n}u(t) \tag{2.20}$$

where $\{u(t)\}$ is normalized white noise, i.e., $E\{e(t)e(s)\} = \delta_{ts}$.

C. Cepstral Maximization and LPC Filters

Returning briefly to the case of infinite data, any modeling filter gives an infinite sequence of covariance lags from which one can generate an infinite sequence of Schur parameters satisfying

$$|\gamma_k| < 1, \qquad k = 0, 1, 2, \cdots \tag{2.21}$$

via Levinson's algorithm. In this case, the square of the filter gain is given by

$$W(\infty) = \rho_\infty := r_0 \prod_{k=0}^{\infty}(1-\gamma_k^2). \tag{2.22}$$

(See, e.g., [8].) Every choice of Schur parameters satisfying (2.21) corresponds to a not necessarily rational filter that shapes white noise into a process with the given covariance lags. The problem of determining which Schur sequences are rational of degree at most n is challenging and unsolved [9], [15], [19]. However, it is known that the choice

$$\gamma_n = \gamma_{n+1} = \cdots = 0 \tag{2.23}$$

always leads to an LPC filter.

In fact, the LPC filter is the filter obtained by maximizing the zeroth-order cepstral coefficient c_0 once the correlation coefficients $r_0, r_1, \cdots, r_n$ have been fixed. To see this well-known fact, note that the cepstral gain c_0 is the logarithm of the modeling filter gain (2.22), i.e.,

$$c_0 = -2\log a_0 = \log \rho_\infty. \tag{2.24}$$

Again, appealing to (2.22), this yields

$$c_0 = \log r_0 + \log \prod_{k=0}^{n-1}(1-\gamma_k^2) + \log \prod_{k=n}^{\infty}(1-\gamma_k^2). \tag{2.25}$$

Therefore, if the correlation coefficients $r_0, r_1, \cdots, r_n$ are given, the first two terms in (2.25) are fixed, and all possible modeling filters having this window of covariance lags are obtained by choosing $\gamma_n, \gamma_{n+1}, \cdots$ in the last term appropriately. Obviously, the entropy gain c_0 is maximized if these free reflection coefficients are chosen to be all zero as in (2.23), which is the LPC solution. On the other hand, by definition, the cepstral gain c_0 is given by

$$c_0 = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log \Phi(e^{j\theta})\, d\theta \tag{2.26}$$

which gives a derivation interpreting a LPC filter as a *maximum entropy filter*.

III. Homomorphic Filtering and Generalizations of Linear Predictive Filtering

In the previous section, we noted that the maximum entropy design of LPC filters can be interpreted as a problem of maximizing a very special piece of the cepstral window, subject to constraints on a given window of covariance data. Indeed, maximizing the zeroth cepstral coefficient yields the unique all-pole, or AR, modeling filter that matches the given covariance data. The point of this paper is a generalization of this observation. That is, by blending the information in a covariance window with a window of cepstral coefficients, rather than with just the zeroth cepstral coefficient, it should be possible to develop a parameterization of the ARMA, or pole-zero, model that generates these windows. Based on such a generalization of LPC filter design, one could also ask whether, given the possible windows of cepstral coefficients of covariance lags, we can parameterize each of the corresponding modeling filters as the solution of some parameterized family of optimization problems. Indeed, it turns out that each modeling filter—with *a priori* constrained covariance lags—is the maximum of an optimization problem for some (positive, in a suitable sense) linear combination of cepstral coefficients, generalizing maximum entropy filtering in a homomorphic filtering context.

A. Cepstral and Covariance Windows as Local Coordinates for Pole-Zero Models

On the real number line $\mathbb{R}$, there are many choices of coordinates. A smooth function g is said to be a local coordinate near $x = 0$ if every smooth function f can be expressed, near 0, as $f(x) = h(g(x))$ for some smooth function h. For example, $g(x) = \sin x$ is a local coordinate near $x = 0$, and $g(x) = x^2$ is not. In general, in $\mathbb{R}^N$, N smooth real-valued

functions $g_1, g_2, \cdots, g_N$ are local coordinates near x_0 if every smooth real-valued function f can be expressed, near x_0, as

$$f(x) = h(g_1(x), g_2(x), \cdots, g_N(x))$$

for some smooth function h. In particular, we are interested in whether the coefficients of pole and zero polynomials are functions of cepstral coefficients and covariance lags.

The methods we now describe for pole-zero models for finite data strings retain some of the most important features of LPC design. We require that the resulting modeling filter be rational of degree at most n, have stable zeros and poles, and match the finite window of covariance lags. As in [21], we will begin by also incorporating the superposition property of homomorphic filtering, *viz.*, we will initially require the resulting modeling filter to also match a finite window of cepstral coefficients.

Our first new result is that there is a one-to-one correspondence between the $2n + 1$ coefficients $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ and the $2n + 1$ coefficients $a_0, a_1, \cdots, a_n, b_1, b_2, \cdots, b_n$ of the denominator and numerator polynomials

$$A(z) = a_0 z^n + a_1 z^{n-1} + \cdots + a_n \qquad (a_0 > 0) \tag{3.1}$$

$$B(z) = z^n + b_1 z^{n-1} + \cdots + b_n \tag{3.2}$$

of the corresponding modeling filter (2.6), provided W has *exactly* degree n.

Theorem 3.1: Each modeling filter (2.6) of degree n determines and is uniquely determined by its window $r_0, r_1, \cdots, r_n$ of covariance lags and its window $c_1, c_2, \cdots, c_n$ of cepstral coefficients.

It is, of course, clear that $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ is determined by a modeling filter (2.6) of degree n.

For example, for any (stable) polynomials (3.2) and (3.1), the coefficients $r_0, r_1, \cdots, r_n$ in the expansion

$$\frac{|B(e^{j\theta})|^2}{|A(e^{j\theta})|^2} = r_0 + 2\sum_{k=1}^{\infty} r_k \cos(k\theta) \tag{3.3}$$

can be determined from (2.6) via (2.3) and (2.4), using the inverse Levinson algorithm [31] in the following way. We begin by determining the coefficients $g_0, g_1, \cdots, g_{2n}$ in the expansion

$$\frac{1}{|A(e^{j\theta})|^2} = g_0 + 2\sum_{k=1}^{\infty} g_k \cos(k\theta)$$

corresponding to a LPC filter. This is done by first applying the inverse Levinson algorithm [31, pp. 47, 165] to $A(z)$ for computing the reflection coefficients and then the inverse Schur algorithm [31, p. 166] for computing $g_0, g_1, \cdots, g_n$, after which, the recursion

$$g_{n+k} = -\sum_{i=0}^{k-1} \frac{a_i}{a_0} g_{n-i+1}$$

yields $g_{n+1}, g_{n+2}, \cdots, g_{2n}$. Finally, the coefficients $r_0, r_1, \cdots, r_n$ are obtained from

$$r_j = p_0 g_j + \sum_{i=1}^{m} p_i \left(g_{|i-j|} + g_{i+j}\right) \tag{3.4}$$

where $p_0, p_1, \cdots, p_n$ are the coefficients in the pseudo-polynomial

$$P(z) = p_0 + \tfrac{1}{2} p_1 (z + z^{-1}) + \cdots + \tfrac{1}{2} p_n (z^n + z^{-n}) \tag{3.5}$$

where

$$P(z) = B(z)B(z^{-1}). \tag{3.6}$$

Consequently, the covariance coefficients $r_0, r_1, \cdots, r_n$ can be computed using just recursive algorithms and ordinary arithmetic operations.

For the sake of completeness, we also give the explicit formulae for the cepstral coefficients $c_1, c_2, \cdots, c_n$ in terms of the poles and zeros of (2.6), generalizing the well-known formulae for the case of LPC filters in the literature [25].

$$c_0 = -2\log a_0$$

$$c_k = \frac{1}{k}\{s_k(A) - s_k(B)\} \qquad k = 1, 2, 3, \cdots \tag{3.7}$$

where

$$s_k(A) = p_1^k + p_2^k + \cdots + p_n^k \tag{3.8}$$

$$s_k(B) = z_1^k + z_2^k + \cdots + z_n^k \tag{3.9}$$

and where $p_1, p_2, \cdots, p_n$ are the roots of $A(z)$, and $z_1, z_2, \cdots, z_n$ are the roots of $B(z)$. Moreover, by Newton's identities [14], we have the recursion formulae for the cepstral coefficients, generalizing those known in the literature for the case of LPC filters:

$$s_k(A) = -k\frac{a_k}{a_0} - \sum_{j=1}^{k-1} \frac{a_{k-j}}{a_0} s_j(A) \tag{3.10}$$

$$s_k(B) = -kb_k - \sum_{j=1}^{k-1} b_{k-j} s_j(B) \tag{3.11}$$

where for $k > n$, we set $a_k = 0$ and $b_k = 0$.

Conversely, it is much more nontrivial, and certainly new, that the modeling filter can be recovered from the observed covariance and cepstral windows. One of the key points of this observation, then, is that $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ form local coordinates for the space of pole-zero filters of degree n, with respect to which one can use calculus. In fact, the proof of the converse uses a minimization argument in the coefficients of A and B in a coordinate system adapted to take advantage of tools, such as convexity, in optimization. More explicitly, the pseudo-polynomial P that we constructed from B above lies in the space $\mathcal{D}$ of all pseudo-polynomials (3.5) of degree n that take non-negative values on the unit circle. $\mathcal{D}$ is a closed, convex set with interior consisting of $\mathcal{D}_+$, which are those pseudo-polynomials that take positive values on the unit circle. Since B is a Schur polynomial, i.e., a monic polynomial with all its roots

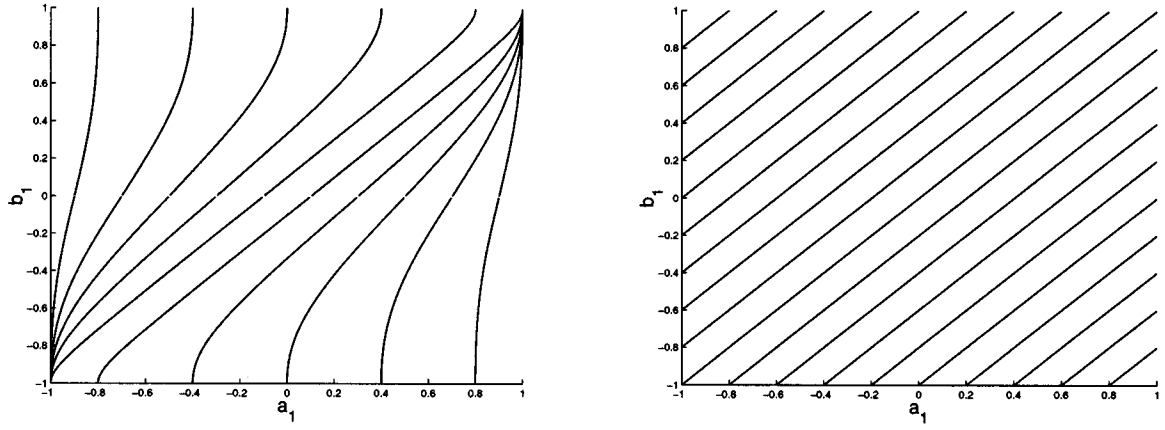

Fig. 2. Level sets for (left) r_1 and (right) c_1 in the case $n = 1$.

in the open unit disc, P actually lies in $\mathcal{D}_+$. Since the space of Schur polynomials is not convex for $n \geq 3$ [6], we will also convert A to a pseudo-polynomial

$$Q(z) = q_0 + \tfrac{1}{2}\, q_1(z + z^{-1}) + \cdots + \tfrac{1}{2}\, q_n(z^n + z^{-n}) \quad (3.12)$$

defined via

$$Q(z) = A(z)A(z^{-1}). \quad (3.13)$$

The polynomials A and B can, of course, be recovered from P and Q using spectral factorization.

In this context, given $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$, we propose to minimize the convex functional

$$\begin{aligned} J(p, q) = {} & r_0q_0 + r_1q_1 + \cdots + r_nq_n \\ & - c_1p_1 - c_2p_2 - \cdots - c_np_n \\ & + \frac{1}{2\pi} \int_{-\pi}^{\pi} P(e^{j\theta}) \log \frac{P(e^{j\theta})}{Q(e^{j\theta})}\, d\theta \end{aligned} \quad (3.14)$$

with the pseudo polynomials (3.5) and (3.12), which are given by (3.13) and (3.6) respectively, ranging over the closed convex region in $\mathbb{R}^{2n+1}$ of variables $p_1, p_2, \cdots, p_n, q_0, q_1, \cdots, q_n$ such that the pseudo-polynomials (3.5) and (3.12) are non-negative on the unit circle.

While J is nonlinear, it always has a minimum since it is a convex function defined on a closed convex set. Moreover, since we assume that the data $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ is generated by a pair (2.6) of Schur polynomials, the corresponding pseudo-polynomials lie in the open set $\mathcal{D}_+$. Therefore, to establish uniqueness of the pair (2.6), it is sufficient to show that J has a unique minimum in $\mathcal{D}_+$, which we can do by using the first derivative test (since J is convex). Such a minimum therefore must satisfy

$$\frac{\partial J}{\partial q_k} = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{jk\theta}\, \frac{P(e^{j\theta})}{Q(e^{j\theta})}\, d\theta - r_k = 0 \quad (3.15)$$

$$\frac{\partial J}{\partial p_k} = c_k - \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{jk\theta} \log \frac{P(e^{j\theta})}{Q(e^{j\theta})}\, d\theta = 0. \quad (3.16)$$

In particular, it follows from (2.3) and (2.10) that any minimum of J must define a modeling filter that matches the covariance and the cepstral window. Moreover, when A and B are coprime polynomials, i.e., when the filter is of degree precisely n, the modeling filter is uniquely determined by the spectral density

$$\Phi(z) = \frac{P(z)}{Q(z)}.$$

Now, $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ can be estimated from data; see, for example, (2.12) for the covariance lags and [30] for the cepstral coefficients. A modeling filter (2.6) obtained in this way by minimizing (3.14) will be referred to as a *Cepstral-covariance matching (CCM) filter*.

This result gives a method for regenerating a modeling filter of degree n from its covariance and cepstral windows of length n. In Figs. 2 and 3, we illustrate this result in the case $n = 1$ by showing the level sets for the cepstral coefficient c_1 and the covariance lag r_1, where we have set $r_0 = 1$.

We note that the level sets coincide when $c_1 = 0$ and $r_1 = 0$, i.e., when A and B have a common factor (and therefore coincide). This holds for all n *mutatis mutandis*.

Remark 3.2: In Figs. 2 and 3, the maximum entropy filters lie on the horizontal line $b_1 = 0$. One can see that when restricted to this (or any horizontal) line, either c_1 or r_1 is a coordinate function and hence—for maximum entropy filters—one can express c_1 or r_1 as a function of r_1 or c_1, respectively. While this result holds for arbitrary n, when restricted to (the n-dimensional submanifold of) LPC filters, for arbitrary ARMA models, our result asserts that rather than being dependent variables on one another, the windows $r_0, r_1, \cdots, r_n$ and of $c_1, c_2, \cdots, c_n$ are complementary sets of partial coordinates, which together are uniquely defined by, and uniquely define, stable, minimum phase filters of degree n.

In general, it has been long appreciated that autoregressive, moving-average (ARMA) alternatives to LPC filter design would be desirable in signal processing. Early work in this direction was developed in [21] in which the LPC method was used to first find a candidate pole polynomial from which a zero polynomial was found using Shanks' method [26]. Inspired by this work, Stieglitz [38] developed a method to simultaneously estimate the poles and zeros of the modeling filter noting, however, that the algorithm could lead to unstable pole polynomials, which would cause divergence. In general, algorithms producing pole-zero models or, equivalently, the parameters in an ARMA model, are known to have convergence problems

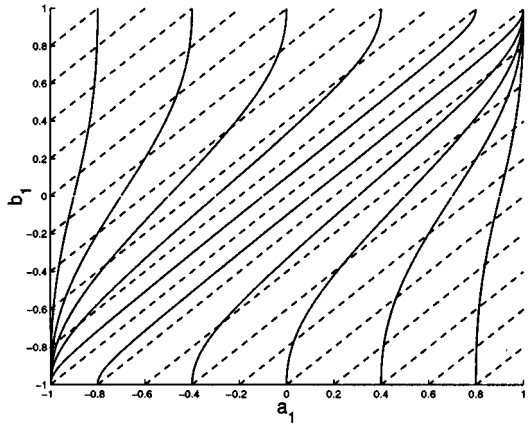

Fig. 3. Transversality of the level sets for $n = 1$ when $A \neq B$.

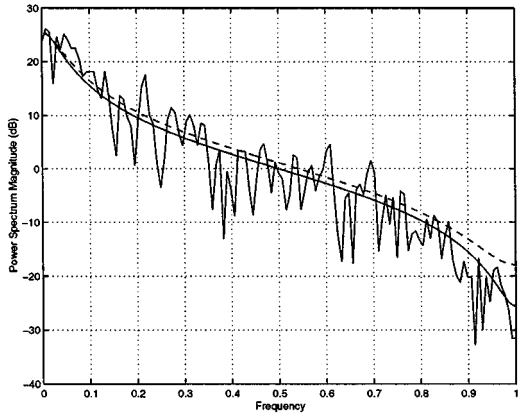

Fig. 4. First-order CCM spectral envelope (dashed line) for [36, Ex. 9.6].

and several proposed schemes [18, ch. 10] do not guarantee that the numerator or denominator in a resulting modeling filter will be stable. This is in part because these optimization schemes are nonlinear but nonconvex, as can be seen in detail in, for example, [36, p. 333, (9.47), p. 334, (9.48)].

The example analyzed in [36, p. 340, Example 9.6], which is a one-dimensional (1-D) ARMA process with a pole and zero near the unit circle, illustrates the fact that global convergence of the associated algorithms may fail depending on the choice of certain design parameters (e.g., forgetting factors) that need to be set in the standard algorithms—in sharp contrast to the convex minimization scheme presented here. In Fig. 4, we depict the periodogram of the system response to white noise and the corresponding true spectrum as a solid curve, whereas the dashed curve represents the spectral envelope of the corresponding CCM filter. This compares quite favorably with the simulations in [36] for various choices of forgetting factors.

If $n = 1$, it follows that the cepstral coefficient must satisfy $-2 \leq c_1 \leq 2$ while the positive definiteness of the associated Toeplitz matrix (2.14) constrains the covariance lags to satisfy $r_0^2 - r_1^2 > 0$. Normalizing $r_0 = 1$, Fig. 5 illustrates the possible values of an attainable cepstral coefficient, covariance coefficient pair (c_1, r_1) as the shaded subregion within the larger region dictated by the constraints on c_1 and r_1 separately. One also sees that the coordinates (c_1, r_1) are singular at $(c_1, r_1) = (0, 0)$, where A and B have a common factor.

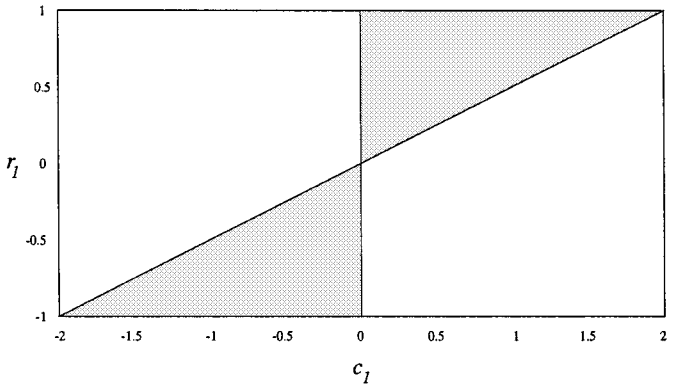

Fig. 5. Feasible (c_1, r_1) pairs for $n = 1$.

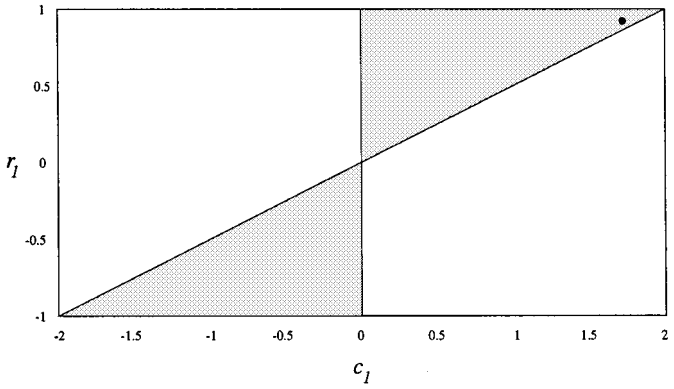

Fig. 6. Showing feasibility for cepstral/covariance data for [36, Ex. 9.6].

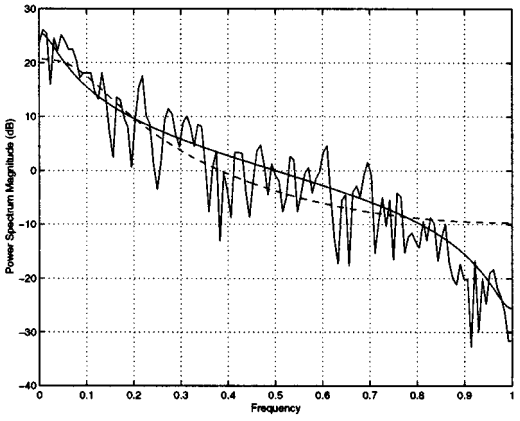

Fig. 7. Second-order LPC spectral envelope (dashed line) for [36, Ex. 9.6].

Fig. 6 shows that the problem data for [36, p. 340, Ex. 9.6], which is depicted by a dot, does lie in the feasible region of cepstral/covariance pairs.

We also note that for this example, the first covariance coefficient r_1, with r_0 normalized to be 1, lies in the feasibility region $1 > r_1^2$ for LPC filtering. Moreover, estimating r_2 for [36, Ex. 9.6] renders the corresponding Toeplitz matrix (2.14) positive definite. The spectral envelope, for the second-order LPC filter fitting this pair of data, is illustrated in Fig. 7 with a dashed curve.

As another illustration, let us consider an example in [35]. Consider a data string (2.11) with $N = 512$ obtained by passing white noise through a fourth-order filter with poles at

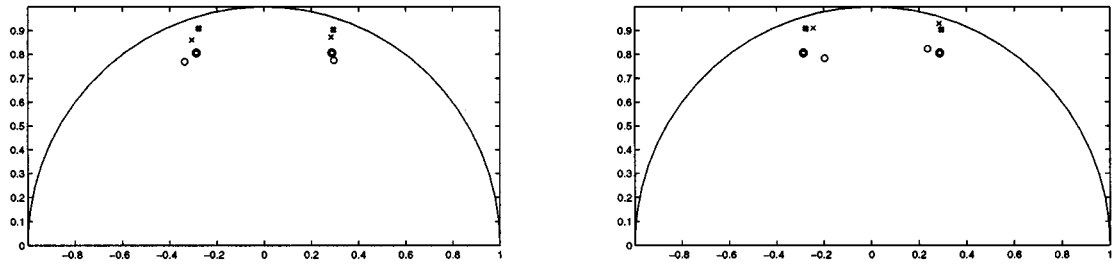

Fig. 8. Poles and zeros for CCM filtering (left) and [35] i) (right) versus true poles and zeros (bold).

TABLE I
CEPSTRAL AND COVARIANCE ERRORS

	1 time	10 times	100 times	[35](i)	[35](ii)	[35](iii)
cepstral error	.1404	.0300	.0098	.0561	.1005	.1549
covariance error	.0789	.0334	.0128	.0765	.0677	.3348

$0.95e^{\pm j0.4\pi}$, $0.95e^{\pm j107\pi/180}$ and zeros at $0.85e^{\pm j70.4\pi/180}$, $0.85e^{\pm j109.6\pi/180}$. The performance of three methods are compared in [35]. The first and second are ARMA algorithms that see $(r_0, r_1, \cdots, r_8)$ and $(r_0, r_1, \cdots, r_9)$, respectively, and estimate the AR and MA parts separately [27], [34]. The third ARMA algorithm estimates the model components simultaneously [1]. The corresponding estimates in [35] are determined from the pole and zero averages computed there from 100 Monte Carlo runs and are shown in Table I. The three algorithms in [35] are denoted here as i), ii), and iii). For comparative purposes, algorithms 32 i) and 32 iii) are closest to what we will employ here since the covariance lag records have the same length.

For the sake of illustration, in this paper, we describe two sets of simulations with this example, returning to the second in Section IV. For this simulation, we made one, ten, or 100 Monte Carlo runs and take the average of the filter coefficients $a_0, a_1, \cdots, a_n, b_1, b_2, \cdots, b_n$ thus obtained from the underlying model data. Based on these averages, the corresponding values of $r_0, r_1, \cdots, r_n, c_1, c_2, \cdots, c_n$ are computed via (3.4) and (3.7), and the ℓ_2 norms of the errors in the vectors of cepstral and covariance lags are computed. The results are shown in Table I. For comparison, the corresponding errors in [35], which are determined from the pole and zero averages computed there from 100 Monte Carlo runs, are shown in Table I.

We note that the average covariance errors incurred in the CCM algorithm and the algorithm [35] i) are of the same order of magnitude for 100 Monte Carlo runs. There is, however, a greater difference in the cepstral coefficients, presumably since [35] i) did not use any measure of cepstral distance as a penalty function. In Fig. 8, we depict the estimated poles and zeros (as $\circ$ and $\times$, respectively) versus the true poles and zeros (in bold) for both the CCM algorithm and the algorithm [35] i). The results compare favorably, as one should expect from the closeness of the cepstral and covariance approximations.

These examples are in harmony with our experience, *viz.*, such simulations work reasonably well when the cepstral and covariance data are generated by a modeling filter of degree n. However, the problem of optimizing a modeling filter from candidate cepstral and covariance data requires, of course, some knowledge of the inverse problem of determining those windows that can arise from a modeling filter of degree n. We have already seen that this problem is considerably complicated by the fact that there is a nontrivial coupling between attainable joint windows, even in the case $n = 1$.

As we show in the next subsection, there is a rather appealing alternative optimization problem for which the parameters can be chosen from an *a priori* given set and provides a direct generalization of the maximum entropy design for LPC filters.

B. Cepstral Maximization and a Generalization of LPC Design

In contrast to the cepstral window, for which we have only necessary conditions, the constraints on the covariance lags for higher n are given by the positive definiteness of the Toeplitz matrix (2.14).

This suggests a generalization of the maximum entropy approach in a homomorphic filtering context. More explicitly, we consider maximizing a linear combination of the window of cepstral coefficients

$$p_0c_0 + p_1c_1 + \cdots + p_nc_n \tag{3.17}$$

subject to the interpolation condition

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{jk\theta}\Phi(e^{j\theta})\,d\theta = r_k, \qquad k = 0, 1, \cdots, n. \tag{3.18}$$

For the maximal entropy filter, one chooses $p_0 > 0$ and $p_1 = \cdots = p_n = 0$. The positivity of p_0 reflects the fact that in any maximization (or minimization) problem, there needs to be some fixing of sign definiteness. In the more general case, we ask that the associated pseudo-polynomial (3.5) be positive on the unit circle, i.e., that P lie in $\mathcal{D}_+$. This maximization problem leads to a rather neat solution, directly generalizing linear predictive code filters.

Theorem 3.3: The problem to maximize (3.17) subject to (3.18) has a finite solution only if the pseudo-polynomial (3.5) belongs to $\mathcal{D}$. If $P \in \mathcal{D}_+$, there is a unique solution Φ, and this solution has the form

$$\Phi(z) = \frac{P(z)}{Q(z)} \tag{3.19}$$

where $Q \in \mathcal{D}_+$. The corresponding modeling filter is obtained from the stable, minimum-phase spectral factor

$$W(z) = \frac{B(z)}{A(z)}$$

of $\Phi(z)$. In particular, $A(z)$ and $B(z)$ are the unique stable polynomial factors (3.13) and (3.6) of $Q(z)$ and $P(z)$, respectively.

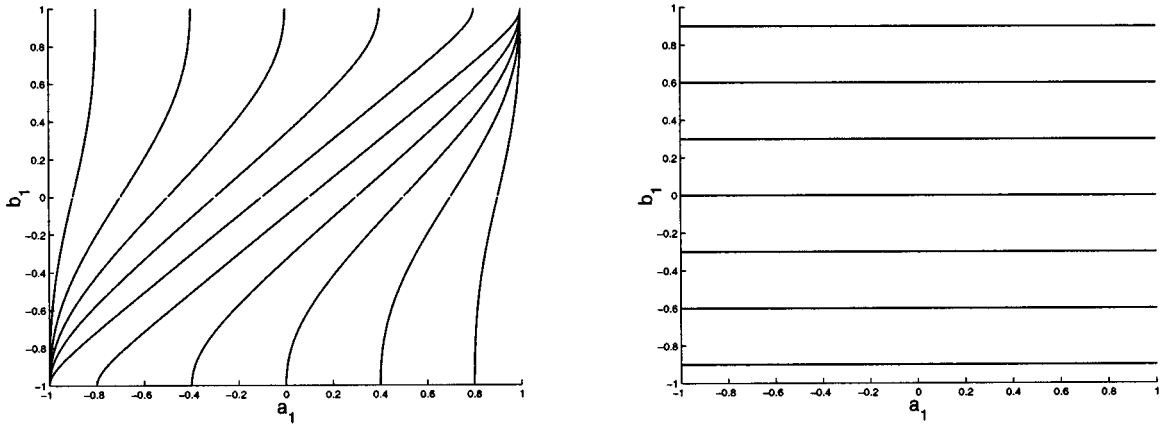

Fig. 9. Level sets for r_1 (left) and p_1 (right) for $n = 1$.

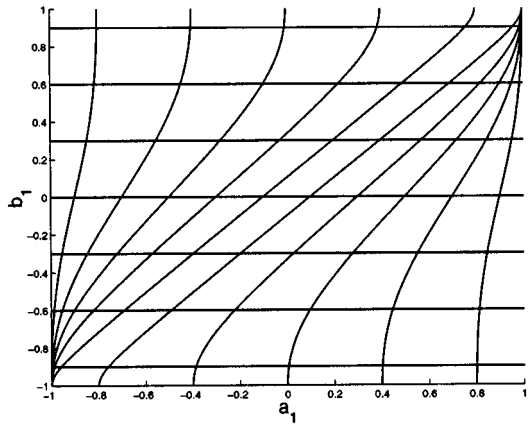

Fig. 10. Complementarity of the level sets for r_1 and p_1 for $n = 1$.

Thus, in contrast to the nontrivial coupling between attainable cepstral and covariance pairs, the feasible covariance windows and the set of positive pseudo-polynomials are independent quantities. Indeed, one should expect that these sets of quantities would form complementary sets of coordinates for the space of modeling filters, and this is a fact we illustrate in Figs. 9 and 10 for the case $n = 1$, where we have normalized P so that $p_0 = 1$.

These figures are in sharp contrast to Figs. 2 and 3 in a fundamental way. In contrast to the level sets for cepstral coefficient, covariance coefficient pairs, each covariance level set meets each level set of the linear combination vector p in one and only one point. In particular, the set of feasible pairs is uncoupled, being determined separately by the positive definiteness of the corresponding Toeplitz matrix and the corresponding pseudo-polynomial. This overcomes the limitations of cepstral-covariance minimization in a very effective manner. For example, choosing a cepstral coefficient, covariance coefficient pair (c_1, r_1) outside the attainable region in Fig. 5 and running the minimization algorithm always yields an ARMA system with zeros on the unit circle because while the stationarity condition (3.15)—ensuring covariance matching—will always be satisfied, the cepstral matching condition (3.16) may fail. Theorem 3.3 asserts that this strong form of transversality of the level sets holds for all n.

We now briefly outline the proof of Theorem 3.3. In view of (2.10) and (3.5), the cost function (3.17) can be written as

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} P(e^{j\theta})\log\Phi(e^{j\theta})\,d\theta. \tag{3.20}$$

Therefore, the optimization problem of Theorem 3.3 is reduced to finding the spectral density

$$\Phi(e^{j\theta}) = f_0 + 2\sum_{k=1}^{\infty} f_k \cos k\theta$$

that maximizes the generalized entropy gain (3.20) subject to the covariance matching condition (3.18).

More precisely, consider the infinite-dimensional convex optimization problem to maximize (3.20) subject to the $n+1$ constraints (3.18) over all sequences $f = (f_0, f_1, f_2, \cdots)$ such that $\Phi(e^{j\theta}) > 0$ for all $\theta \in [0, 2\pi]$. In order to solve this (primal) problem, we must find the saddle point of the Lagrangian

$$L(f, q) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left[f_0 + 2\sum_{k=1}^{\infty} f_k\cos k\theta\right] P(e^{j\theta})\,d\theta + \sum_{k=0}^{n} q_k(r_k - f_k) \tag{3.21}$$

as, for example, in [28]. It is not hard to see that the Lagrangian has a finite maximum only if both P and Q belong to $\mathcal{D}$. Any feasible maximum will occur in a stationary point

$$\frac{\partial L}{\partial f_k} = 0, \qquad k = 0, 1, 2, \cdots.$$

This stationarity condition can be seen to be equivalent to (3.19), and hence , we must have P and Q in $\mathcal{D}_+$ for $\Phi(e^{j\theta}) > 0$ to hold for all $\theta \in [0, 2\pi]$.

Forming the function $q \mapsto \max L(f, q)$ then leads to a finite-dimensional dual problem, namely, the problem to minimize the function

$$J_P(q_0, q_1, \cdots, q_n) = r_0q_0 + r_1q_1 + \cdots + r_nq_n - \frac{1}{2\pi}\int_{-\pi}^{\pi} P(e^{j\theta})\log Q(e^{j\theta})\,d\theta \tag{3.22}$$

in the $n+1$ variables $q_0, q_1, \cdots, q_n$, which are the coefficients of Q, over all choices of variables $(q_0, q_1, \cdots, q_n)$ such that

$$Q(e^{j\theta}) > 0 \quad \text{for all } \theta \in [-\pi, \pi]. \tag{3.23}$$

The functional (3.22) was introduced in [10], where it was shown that it has a unique minimum in $\mathcal{D}_+$.

It is readily seen that the gradient of (3.22) is given by

$$\frac{\partial J_P}{\partial q_k}(q_0, q_1, \cdots, q_n) = r_k - f_k \tag{3.24}$$

where

$$f_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{jk\theta}\frac{P(e^{j\theta})}{Q(e^{j\theta})}\,d\theta \quad \text{for } k = 0, 1, 2, \cdots, n \tag{3.25}$$

are the covariance lags corresponding to a process with spectral density

$$\frac{P(e^{j\theta})}{Q(e^{j\theta})} = f_0 + 2\sum_{k=1}^{\infty} f_k \cos(k\theta). \tag{3.26}$$

Since the gradient is zero at the minimizing point $(q_0, q_1, \cdots, q_n)$, we have, at this point

$$r_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{jk\theta}\frac{P(e^{j\theta})}{Q(e^{j\theta})}\,d\theta, \qquad k = 0, 1, 2, \cdots, n$$

which, as seen from (2.3), are precisely the $n+1$ first covariance lags where Φ is given by (3.19).

Example 3.4: We now illustrate the use of this generalized maximization problem in the design of filters for signals arising in speech analysis. In speech synthesis, conventional vocoders identify phonemes into voiced and unvoiced speech. A single phoneme evolves on the order of magnitude of 100 ms. The speech pattern is usually divided into frames of a few tens of milliseconds, where it is regarded to be stationary. On each such 20–30 ms frame of speech, the signal is sampled to yield the speech data $y_0, y_1, y_2, \cdots, y_N$, where N is typically on the order of 200 to 300. To illustrate Theorem 3.3, we will compare the performance of three filters with respect to a frame of speech acquired during the formation of the voiced nasal [ng]. We have sampled the phonemes at a rate of 8000 samples/s and retained 250 sample points for each frame. Thus, each frame represents a time history of speech over a period of roughly 30 ms. From this data, the first $n+1$ covariance lags $r_0, r_1, r_2, \cdots, r_n$ have been estimated, For the sake of exposition, we begin with $n = 6$ and two choices of a sixth-degree positive pseudo-polynomial. Using ergodic estimates (2.8), we obtain $(r_0, r_1, r_2, \cdots, r_6) = (0.7468, 0.6487, 0.4335, 0.1885, -0.0040, -0.0990, -0.1265)$. For the sake of comparison, we then illustrate the case of a 12th-order LPC filter, for which we also need the estimates $(r_7, r_8, \cdots, r_{12}) = (-0.1105, -0.0700, -0.0085, 0.0706, 0.1559, 0.2160)$.

Fig. 11 shows a periodogram determined from the frame of the voiced nasal [ng] together with the spectral envelope of a sixth-order optimal filter with $p_0 = 1$ and $p_1 = \cdots = p_6 = 0$ designed from this frame. Fig. 12 shows the same periodogram together with the spectral envelope of a sixth-order optimal filter with $(p_0, p_1, \cdots, p_6) = (1, 0.0478, -0.4831, -0.4440, -0.7453, 0.5203, 0.4486)$ designed from the same frame of the voiced nasal [ng].

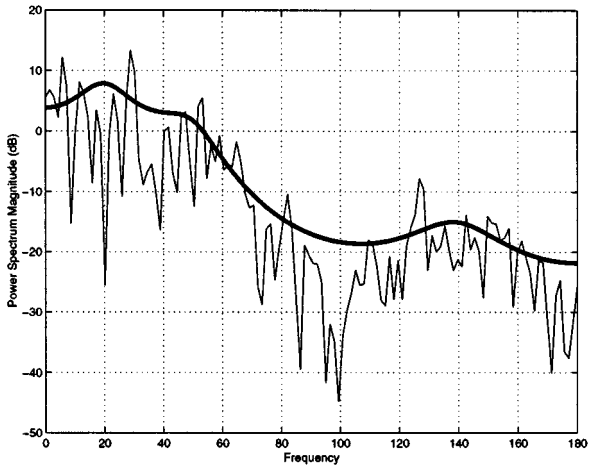

Fig. 11. Sixth-order filter with $p_0 = 1$ and $p_1 = p_2 = \cdots = p_6 = 0$.

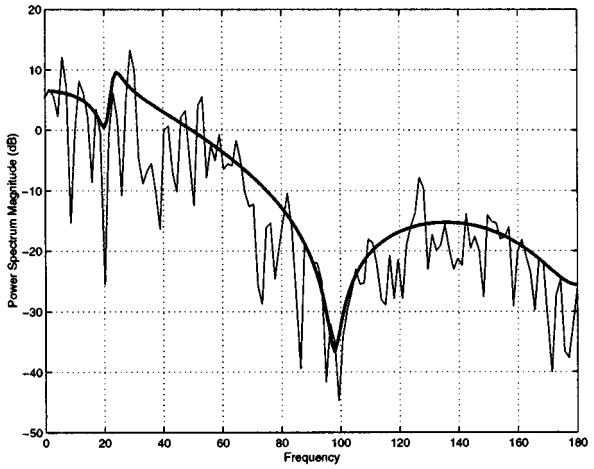

Fig. 12. Sixth-order filter with $(p_0, p_1, \cdots, p_6) = (1, 0.0478, -0.4831, -0.4440, -0.7453, 0.5203, 0.4486)$.

Following our derivation of the generalized maximum entropy optimization criterion, it is clear that the spectral envelope depicted in Fig. 11 is that of the LPC filter determined by the covariance window of length 6. While this filter does correspond to the default choice $p_0 = 1$ and $p_1 = \cdots = p_6 = 0$ in our current design scheme, it is also fair to note that in general, this filter only makes use of the data $r_0, r_1, r_2, \cdots, r_6$, whereas the filter of Fig. 12 makes use of the additional data string $p_1, \cdots, p_6$. Therefore, it is better to compare the performance of the second filter with an LPC filter of order 12 obtained from the data string $r_0, r_1, r_2, \cdots, r_{12}$. Fig. 13 shows a periodogram determined from the frame of the voiced nasal [ng] together with the spectral envelope of a 12th-order optimal filter with $p_0 = 1$ and $p_1 = \cdots = p_{12} = 0$ designed from the same frame of the voiced nasal [ng].

While the spectral envelope derived from a generalized maximum entropy design compares favorably with the spectral envelopes of an LPC filter with twice the order (but, of course, an equal amount of data points), a key issue is how to choose the coefficients p_i. We note from Fig. 9 that for $n = 1$, to set p_1 constant is to set b_1 constant. More generally, from spectral

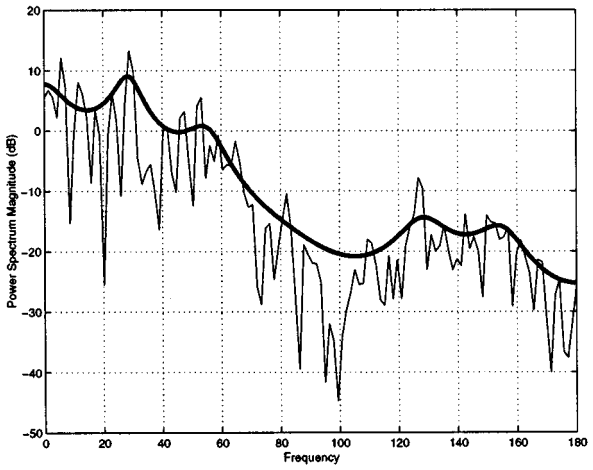

Fig. 13. Twelfth-order filter with $p_0 = 1$ and $p_1 = p_2 = \cdots = p_{12} = 0$.

factorization, it follows that to fix the positive pseudo-polynomial P constant is to set the numerator polynomial B constant. Tuning the zeros of a modeling (or shaping) filter has long been a desired goal in the ARMA modeling of signals and systems. One approach to the identification of zeros (and poles) from data has been cepstral analysis, particularly the use of cepstral windowing and smoothing. In the next section, we will describe methods for using cepstral analysis to estimate the zeros, as well as to compute the pole polynomial A from the estimates of the zeros and the covariance window—leading to a realization algorithm for representing this particular ARMA model in a lattice-ladder architecture, as depicted in Fig. 14.

We conclude this section by illustrating that the realization issues here need to take into account the fact that we are developing models from a finite, not an infinite, data string.

Remark 3.5: At first blush, given the numerator polynomial $B(z)$, it might seem possible to develop an ARMA model for a finite covariance window by first passing the observed signal through a zero filter as a prefilter

$$\text{input} \xrightarrow{y} \boxed{W_0(z)} \xrightarrow{v} \text{output} \tag{3.27}$$

with transfer function

$$W_0(z) = \frac{z^n}{B(z)}$$

and then to derive an LPC "all pole" filter from the filtered observations in order to generate an ARMA model. To this end, suppose for the moment that the output process v is stationary having partial covariance sequence $\tilde{r}_0, \tilde{r}_1, \cdots, \tilde{r}_n$, and let $\tilde{\gamma}_0, \tilde{\gamma}_1, \cdots, \tilde{\gamma}_{n-1}, \tilde{\rho}_0, \tilde{\rho}_1, \cdots, \tilde{\rho}_n$ and the Szegö polynomials $\tilde{\varphi}_0(z), \tilde{\varphi}_1(z), \cdots, \tilde{\varphi}_n(z)$ be the corresponding output from the Levinson algorithm (2.15). Moreover, for $k = 0, 1, \cdots, n$, let $B_k(z)$ be a polynomial of degree k generated by the recursion

$$B_{k-1}(z) = z^{-1}[B_k(z) - B_k(0)z^k\tilde{\varphi}_k(z^{-1})],\ B_n(z) = B(z). \tag{3.28}$$

Then, using formulas in [3, pp. 117–118], the pole-zero model obtained in this way can be realized by the lattice-ladder filter depicted in Fig. 14, where the gains

$$\alpha_0, \alpha_1, \cdots, \alpha_{n-1}, \beta_0, \beta_1, \cdots, \beta_n \tag{3.29}$$

are given by

$$\alpha_k = \tilde{\gamma}_k, \qquad k = 0, 1, \cdots, n-1$$

$$\beta_k = -\frac{\tilde{\rho}_k}{\sqrt{\tilde{\rho}_n}} B_k(0), \qquad k = 0, 1, \cdots, n.$$

This approach, however, turns out to be naive in that it does not solve the problem stated above. In fact, if

$$W_0(z) = h_0 + h_1 z^{-1} + h_2 z^{-2} + \cdots$$

is the Laurent expansion of the proper rational function W_0 about infinity

$$\tilde{r}_k := E\{v(k)v(0)\} = \sum_{\ell=0}^{\infty}\sum_{m=0}^{\infty} h_\ell h_m r_{\ell+k-m}$$

and consequently, the partial covariance sequence $\tilde{r}_0, \tilde{r}_1, \cdots, \tilde{r}_n$ will depend on the infinite sequence $r_0, r_1, r_2, \cdots$ rather than on the partial covariance sequence (2.13). Therefore, this construction of an ARMA model will not reproduce the covariance data of the original process. Moreover, to obtain a stationary output process $\{v(t)\}$ in (3.27), we need to let the system come to steady state, which will require many steps if the roots of $B(z)$ are close to the unit circle.

IV. Realization Algorithms for Lattice-Ladder Notch (LLN) Filters

The desirability of being able to assign zeros to modeling (or shaping) filters has been widely cited, for example, in speech processing [5, p. 1726], [26, pp. 271–272], [33, p. 105, 76–78]. With its ability to guarantee stability of the numerator and denominator of the ARMA model, the generalized maximum entropy scheme we presented in the previous section also gives an independent solution of the long standing problem of realizing a covariance window with a guaranteed stable and minimum phase modeling filter. This is the problem of covariance extension by a positive real rational function of bounded degree.

In [15] and [16], Georgiou proved that, given a covariance window for which the Toeplitz matrix (2.14) is positive definite, for any choice of a Schur polynomial B of degree n, there exists a Schur polynomial A of degree n for which the modeling filter W determines a spectral density Φ matching the window of covariance lags. He also conjectured uniqueness of A and, hence, W and Φ, which would give a well-posed solution of the covariance matching problem with guaranteed stability of the numerator and denominator of the (unique) modeling filter. Georgiou's conjecture was finally established in [8] using geometric methods and, later in [10], using a convex minimization argument that turns out to be equivalent to the dual optimization problem used in the above proof (after one identifies the linear coefficients in the cepstral maximization problem with the

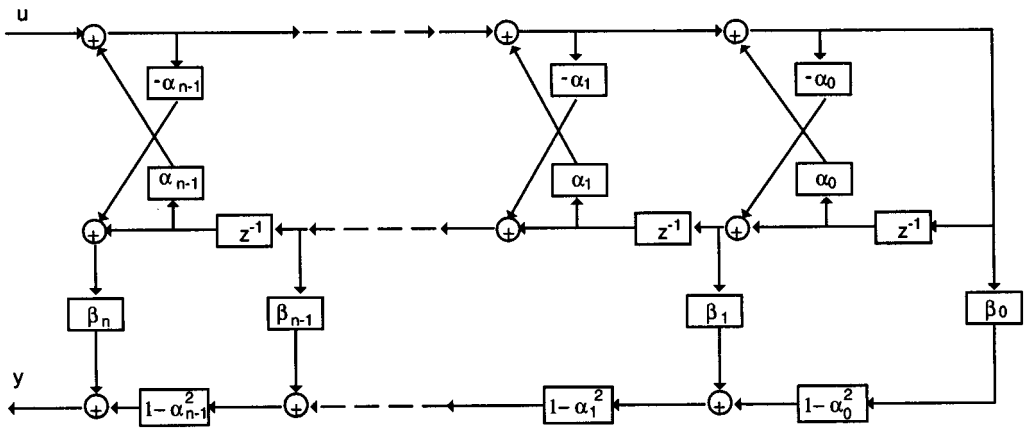

Fig. 14. Lattice-ladder filter.

coefficients of the defining pseudo-polynomial P). The primal problem of maximization of the cepstral coefficients presented here is, however, new and has the results of [8] as a corollary, which we now state for the sake of completeness.

Corollary 4.1 (Moving-Average Assignability Theorem): Let $r_0, r_1, \cdots, r_n$ be a given positive partial covariance sequence. Then, given any stable polynomial (3.2), there exists a unique stable polynomial (3.1) such that

$$W(z) = \frac{B(z)}{A(z)} \tag{4.1}$$

is a minimum-phase spectral factor of a spectral density $\Phi(z)$ satisfying

$$\Phi(e^{j\theta}) = f_0 + 2\sum_{k=1}^{\infty} f_k \cos k\theta;$$
$$f_k = r_k \qquad \text{for } k = 0, 1, \cdots, n.$$

In particular, all nth-order ARMA models for the given partial covariance data are in one–one correspondence with self-conjugate sets of n points (counted with multiplicity) lying in the open unit disc, i.e., with all possible zero structures of modeling filters. Moreover, the modeling filter $W(z)$ depends analytically on the covariance data and the choice of zero polynomial $B(z)$.

Taking this intuitive parameterization of all ARMA modeling filters that match the given covariance data as the starting point, the purpose of this section is to describe a computationally effective algorithm for realizing filters satisfying the same interpolation properties as the LPC filter but allowing for the coefficients of the positive pseudo-polynomial P—or, equivalently, the zeros of the modeling filter—to be set arbitrarily or to be determined from data using, for example, cepstral methods. In this section, we will also develop an algorithm for computing the pole polynomial, and we will also illustrate how to update the parameters of a lattice-ladder filter realization of such modeling filters, enhancing the lattice realization of LPC filtering.

In this language, we require the filter to meet the following specifications.

i) For the covariance coefficients $r_0, r_1, r_2, \cdots, r_n$ extracted from the finite data record, the (stable) transfer function satisfies

$$\left|W(e^{j\theta})\right|^2 = f_0 + 2f_1\cos\theta + 2f_2\cos 2\theta + \cdots \tag{4.2}$$

where

$$f_k = r_k \qquad \text{for } k = 0, 1, \cdots, n. \tag{4.3}$$

ii) The filter has prescribed zeros $\zeta_1, \zeta_2, \cdots, \zeta_n$ inside the unit disc determined by a prescribed positive pseudopolynomial P.

Here, $P(z) = B(z)B(z^{-1})$, and the transmission zeros are determined as the (self-conjugate) roots of the numerator polynomial

$$B(z) := (z-\zeta_1)(z-\zeta_2)\cdots(z-\zeta_n)$$
$$= z^n + b_1 z^{n-1} + \cdots + b_n. \tag{4.4}$$

By Theorem 4.1, there is exactly one filter (4.1) that satisfies these specifications.

In automatic control, filters having prescribed transmission zeros are referred to as *notch filters*. Notch filters have been used as dynamic compensators to attenuate stable harmonic fluctuations by achieving desired pole-zero cancellations. Although the desirability of notches in the power spectrum of the filters presented in this paper plays a quite different role for signal processing and speech synthesis than the role played by notch filters in automatic control, the analogy suggests that we refer to the filters satisfying specifications i) and ii) as *LLN filters*. This acronym also reflects the lattice-ladder architecture that can be used to implement these filters, as illustrated in Fig. 14.

We observe that the lattice-ladder filter representation is an enhancement of the lattice filter representation depicted in Fig. 1, where the difference is the incorporation of the spec parameters denoted by β, which allow for the incorporation of the prescribed zeros into the filter design. In fact, the lattice filter representation of an all-pole filter can be designed from the lattice-ladder filter architecture by setting the parameter

specifications $\beta_0 = r_n^{-1/2}$, $\beta_1 = \beta_2 = \cdots = \beta_n = 0$ and $\alpha_k = \gamma_k$ for $k = 0, 1, \cdots, n-1$.

In this section, we will describe an effective computational algorithm for finding the parameters [see (3.29)] for an LLN filter, given a fixed positive pseudo-polynomial and covariance window. We begin by discussing various methods for choosing the positive pseudo-polynomial from covariance and cepstral data, presenting a form of cepstral smoothing as the preferred method when the positive pseudo-polynomial is not *a priori* available.

A. Selecting the Positive Pseudo-Polynomial

As mentioned above, LPC design has some attractive features that account for its popularity. These include the properties that the resulting modeling filter is rational of degree at most n, has stable zeros and poles, and matches the finite window of covariance lags. The major disadvantage is that the zeros are in the default setting $z = 0$. The LLN filter allows for placing the zeros arbitrarily while retaining the features of LPC filtering mentioned above. The basic idea is that even an approximate choice of zeros is better than just placing them at the origin. Therefore, except for certain applications where zeros are part of the specifications, when this is a measured variable, for example, through an enrollment session such as occurs in speaker verification, we are left with the task of estimating the positive pseudo-polynomial P or, equivalently, the numerator polynomial B.

A straightforward way to determine P would be to estimate the phase and the moduli of the zeros from the notches in an observed spectrum, as represented by a periodogram or as computed using fast Fourier transforms (FFTs). This is depicted in Fig. 15, where a periodogram is used. The depth of the notches determines the closeness to the unit circle.

Alternatively, B could be determined from any of the ARMA (or MA) procedures described in [26, pp. 271–275] or [18, ch. 10], including Prony's method with constant term. These methods are by themselves less than satisfactory in producing synthetic speech because they do not match the finite window of covariance lags and may not yield stable minimum-phase models. However, the estimates of the zeros need not be perfect since our procedure produces corresponding poles so that the overall zero-pole model matches the finite window of covariance lags and is stable minimum phase.

With all this in mind, we now proceed to describe the method for zero estimation that we propose and that we have used in the simulations in this paper. It has several features in common with the procedures described in [38], but it always yields a stable numerator polynomial B. The spectrum is estimated using a smoothed periodogram obtained by cepstral smoothing. Explicitly, the cepstral parameters are calculated from the data (2.7) using an inverse discrete Fourier transform on the logarithm of the periodogram, after which, the cepstral coefficients are windowed and inversely transformed [30, pp. 494–495]. Using this procedure, we obtain a smooth estimate

$$\hat{\Phi}(e^{j\theta_k}), \qquad k = 1, \cdots, N. \tag{4.5}$$

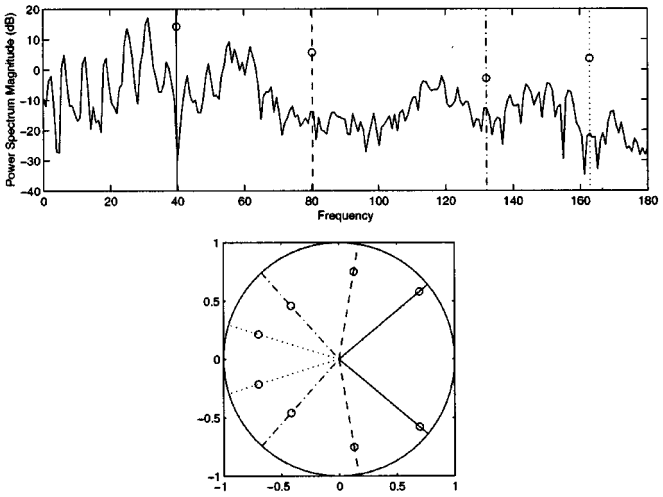

Fig. 15. Selecting the zeros from a periodogram.

Then, in view of (2.4), (2.6), (3.13), and (3.6), the basic spectral estimation problem could be formulated in the following way. Given the estimates (4.5), find pseudo-polynomials P and Q of the form (3.5) and (3.12) such that

$$\max_k \left| Q(e^{j\theta_k})\hat{\Phi}(e^{j\theta_k}) - P(e^{j\theta_k}) \right|$$

is minimized. This leads to a standard linear programming problem in the $2n+2$ variables δ, $p_1, \cdots, p_n$, $q_0, q_1, \cdots, q_n$, namely, to find δ, P, Q that minimizes δ subject to the $4N$ constraints that

$$\begin{aligned} Q(e^{j\theta_k})\hat{\Phi}(e^{j\theta_k}) - P(e^{j\theta_k}) - \delta &\le 0 \\ -Q(e^{j\theta_k})\hat{\Phi}(e^{j\theta_k}) + P(e^{j\theta_k}) - \delta &\le 0 \\ P(e^{j\theta_k}) &\ge \varepsilon \\ Q(e^{j\theta_k}) &\ge \varepsilon \end{aligned}$$

hold for $k = 1, 2, \cdots, N$. Here, the design parameter $\varepsilon > 0$ must be chosen large enough to ensure that P and Q are positive on the unit circle. Given the solution to this linear programming problem, A and B can be obtained via spectral factorization (3.13) and (3.6) of P and Q.

Note that this procedure in general only provides a good estimate of the positive pseudo-polynomial P, which is precisely what we need. However, the estimate of Q is good enough to serve as an initial condition for the optimization algorithm, which we will present next.

B. Algorithm

The minimization of (3.22) given the constraints (3.23) is a convex optimization problem for which there are many standard algorithms and software that determine the minimizing $(q_0, q_1, \cdots, q_n)$ recursively. Most generic codes for convex optimization will compute the gradient (first derivative) and/or Hessian (second derivative) for use in a recursive algorithm, such as those defined, for example, by Newton's method. However, for the specific problem of minimizing J_P, both the gradient and the Hessian can be computed directly, without computing the values of the function (3.22), using the computation

of the covariances of an associated process and using Fourier transforms. (On the other hand, the values of J_P can also be computed in this way. These may be useful in deciding the step size.) While the covariance data are well known to be computable using just recursive algorithms employing only ordinary arithmetic operations, the fact that the computation of the Fourier coefficients can be computed using just recursive algorithms and ordinary arithmetic operations is quite unexpected. For this reason, a direct application of Newton's method gives an efficient and easily implementable algorithm.

More precisely, the gradient is given by (3.24), where the covariances $f_0, f_1, \cdots, f_n$ can be determined, via ordinary arithmetic operations, by first performing the factorization (3.13) and then applying the procedure to determine covariance lags described in Section III-A to $A(z)$. To implement Newton's method, we also need the Hessian of (3.22), i.e., the matrix function of second derivatives of (3.22), i.e.,

$$\begin{aligned} &H_{ij}(q_0, q_1, \cdots, q_n) \\ &\quad := \frac{\partial^2 J_P}{\partial q_i \partial q_j}(q_0, q_1, \cdots, q_n) \\ &\quad = \tfrac{1}{2}(h_{i+j} + h_{i-j}) \qquad i, j = 0, 1, 2, \cdots, n \end{aligned} \tag{4.6}$$

where

$$h_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{jk\theta}\,\frac{P(e^{j\theta})}{Q(e^{j\theta})^2}\,d\theta, \quad \text{for } k = 0, 1, 2, \cdots, 2n \tag{4.7}$$

and $h_{-k} = h_k$. Consequently, $h_0, h_1, \cdots, h_{2n}$ are the $2n+1$ first Fourier coefficients of the spectral representation

$$\frac{P(e^{j\theta})}{Q(e^{j\theta})^2} = h_0 + 2\sum_{k=1}^{\infty} h_k \cos(k\theta). \tag{4.8}$$

Therefore, in the same way as above, the procedure described in Section III-A to compute covariance lags can be used to compute $h_0, h_1, \cdots, h_{2n}$. Since the Hessian is the sum of a Toeplitz matrix and a Hankel matrix, the search direction at the point $q_0, q_1, \cdots, q_n$, i.e.,

$$d := H^{-1}(f - r) \tag{4.9}$$

(where r and f are the $n+1$ vectors with components $r_0, r_1, \cdots, r_n$ and $f_0, f_1, \cdots, f_n$, respectively) can then be determined directly or via a fast algorithm [17].

In fact, Newton's method amounts to recursively updating the vector q with components $q_0, q_1, \cdots, q_n$ according to the rule

$$q^{k+1} = q^k + \lambda_k\, d^k \tag{4.10}$$

where d^k is the search direction (4.9) at the point q^k; see, e.g., [28, pp. 94–95]. Here, the step size λ_k is chosen so that q^{k+1} satisfies the constraints (3.23), which could be tested, for example, in a preselected number of points on the interval $[-\pi, \pi]$ and/or through the positivity test performed in conjunction with the factorization (3.13). An efficient alternative test is given in [7]. For initial point q^0, we may use the Q obtained by the linear programming procedure in Section IV-A, or q^0 can be obtained via

$$\begin{aligned} Q^0(e^{j\theta}) &= q_0^0 + q_1^0\cos\theta + q_2^0\cos 2\theta + \cdots + q_n^0\cos n\theta \\ &= \left|A^0(e^{j\theta})\right|^2 \end{aligned}$$

where $A^0(z)$ is an arbitrary stable polynomial. For $A^0(z)$, we could choose the polynomial $A(z)$ obtained by the LPC (maximum entropy) procedure.

We can determine the gain parameters (3.29) from the LLN filter polynomials A and B in the following way. For $k = n, n-1, \cdots, 1$, solve the recursions

$$\begin{cases} a_{k-1,\,j} = a_{kj} + \alpha_{k-1}a_{k,\,k-j}, & a_{nj} = a_j \\ \alpha_{k-1} = -\dfrac{a_{kk}}{a_{k0}} & \\ b_{k-1,\,j} = b_{kj} - \beta_k a_{k,\,k-j}, & b_{nj} = b_j \\ \beta_k = \dfrac{b_{kk}}{a_{k0}} & \end{cases} \tag{4.11}$$

for $j = 0, 1, \cdots, k$, and set $\beta_0 = b_{00}/a_{00}$. This is a well-known procedure; see, e.g., [3], [4].

Given r and $B(z)$, the recursive selection method for determining the LLN gains is described in the flow chart of Fig. 16. Starting with the initial pole polynomial, in each step, we first determine the vector f with components $f_0, f_1, \cdots, f_n$ via (3.26) in the manner described above, taking $Q(e^{j\theta}) = |A(e^{j\theta})|^2$. Next, we test whether the current approximation $\hat{r}$ of the partial covariance sequence c is within the tolerance ϵ of c. If it is not, we continue the recursive algorithm by updating $A(z)$. If it is, we terminate the recursive steps and determine the filter parameters (3.29) via the recursions (4.11). The updating is performed by taking a Newton step (4.10) as described above, computed from the present $A(z)$ by setting $Q(e^{j\theta}) := |A(e^{j\theta})|^2$. The updated $A(z)$ polynomial is obtained by factoring the pseudo polynomial $Q^{k+1}(z)$ corresponding to the updated point q^{k+1}, thereby checking that the positivity condition (3.23) is also fulfilled.

C. Examples

In this example, we continue the analysis of the frame of speech acquired during the formation of the voiced nasal [ng]. Using the algorithm presented in this section, we developed an LLN filter of order ten, corresponding to $p_0, p_1, \cdots, p_{10}$ estimated as

$$\begin{aligned} &8.5239, -12.7547, 3.6721, 4.4813, -8.4441, 7.3920 \\ &-3.3536, 0.1054, 0.9298, -0.6658, 0.1987 \end{aligned}$$

and matching the covariance lags $(r_0, r_1, \cdots, r_{10})$ given by

$$\begin{aligned} &(0.7468, 0.6487, 0.4335, 0.1885, -0.0040, -0.0990 \\ &-0.1265, -0.1105, -0.0700, -0.0085, 0.0706). \end{aligned}$$

To the left in Fig. 17, we show a periodogram determined from the frame of the voiced nasal [ng] together with the spectral envelope of the corresponding tenth-order LLN filter. To the

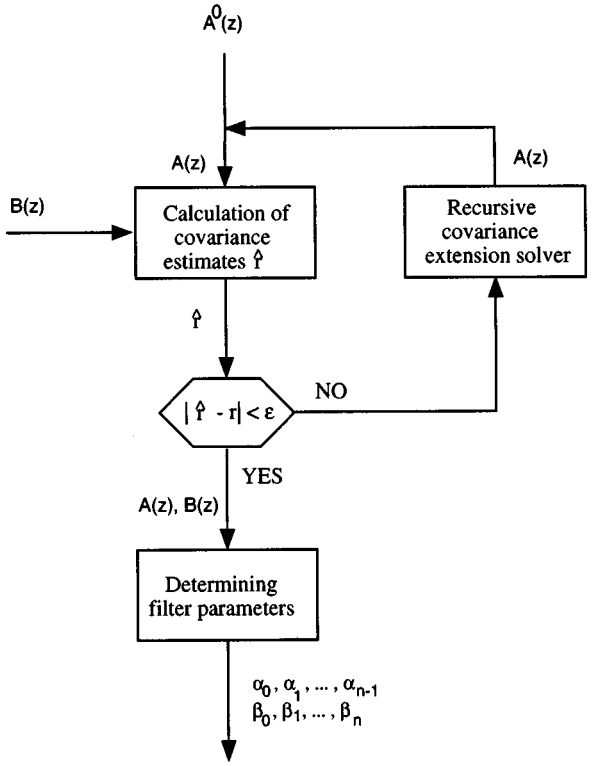

Fig. 16. Determining the LLN parameters.

right in Fig. 17, we compare the performance of this tenth-order LLN filter with an LPC filter of order 20, which was obtained from the data string $r_0, r_1, r_2, \cdots, r_{20}$, where the additional data $(r_{11}, r_{12}, \cdots, r_{20})$ is given by

$$(0.1559, 0.2160, 0.2192, 0.1591, 0.0413, -0.0968 \\ -0.2131, -0.2811, -0.2772, -0.2103).$$

Unlike the sixth-order LLN filter presented in the previous section (Fig. 12), the 20th-order LPC filter uses eight more parameters but still cannot incorporate the notch occurring at roughly $\theta = \pi/2$. This series of simulations suggests that, at least for certain signals, it is better to use extra parameters to fit zeros than to fit additional poles, in harmony with the literature on speech synthesis [5], [26], [33].

As a final illustration, we return to the example in [35] studied in Section III in the context of designing LLN filters. More explicitly, we wish to illustrate the option, mentioned above, of using good *a priori* estimates of the modeling filter zeros as an initial condition for the LLN algorithm. One source of such *a priori* estimates is, of course, zero estimates obtained from an ARMA estimation scheme, where the estimated zeros can then be used to produce an enhanced ARMA model *matching the covariance window*. Here, as an initial condition for the LLN algorithm, we have used two sources for estimates of the zeros: the zero estimates obtained from the CCM algorithm and the zero estimates obtained in [35] i) as described above (see Fig. 18).

We then compare the resulting sequence of covariance lags and the pole locations derived from this combination of CCM/LLN with the locations obtained with CCM and from the combination of [35] i)/LLN with the locations obtained in [35] i).

It is interesting to note the tradeoff in cepstral matching for the CCM filter—which was designed using a cepstral penalty—for improved covariance matching for the CCM/LLN filter; while in the case of the [35] i)/LLN filter, both cepstral matching and covariance matching were improved, as shown in Table II. These results both confirm the fact that the use of good *a priori* information or estimation of the zero polynomial will be reflected in the quality of the pole estimates for this method and that existing ARMA schemes may in fact be used in conjunction with LLN techniques to enhance the performance of both algorithms with respect to covariance matching.

V. Conclusions

The methods for pole-zero modeling that we described in this paper retain some of the most important features of LPC design, namely, that the resulting modeling filter is rational of degree at most n, have stable zeros and poles, and match the finite window of covariance lags. To start, we required the resulting modeling filter to also match a finite window of cepstral coefficients. Generalizing the Newton sum formulae for LPC filters to the case of pole-zero models, we show that each modeling filter of degree n determines, and is uniquely determined by, its nth-order windows of cepstral and covariance coefficients. This characterization has an intuitively appealing interpretation of a characterization in terms of measures of the transient and the steady-state behaviors of the signal, respectively. We show that this follows from a convex minimization algorithm that yields a model with the required matching properties, provided the filter has degree n.

Generalizing the maximum-entropy construction of LPC filters, we modified this scheme to a more well-posed optimization problem, where the covariance data enters as a constraint, and the linear weights of the cepstral coefficients are "positive"—in a sense that a certain pseudo-polynomial is positive. This new problem is a homomorphic filter generalization of the maximum entropy method, leading to the design of all stable, minimum-phase modeling filter of degree n that interpolate the given covariance window. This was illustrated in the context of developing a spectral envelope for a frame of speech extracted from an unvoiced sample in the case of both the new cepstral maximization method for a system of order six and for the classical construction of an LPC (all-pole) model of order 12.

In the last section, it was first observed that by spectral factorization, each choice of positive pseudo-polynomial determines, and is determined by, a choice of stable zeros for the modeling filter, giving an alternative derivation of the recent solution to the rational covariance extension problem in terms of the assignability of the moving average part. This parameterization of all modeling filters in terms of the modeling zeros and the covariance window is another manifestation of our earlier description of modeling filters using measures of the transient and the steady-state behaviors of the signal. Indeed, the choice of zeros and covariance window determines, and is uniquely determined, by a choice of zeros and poles of the modeling filter.

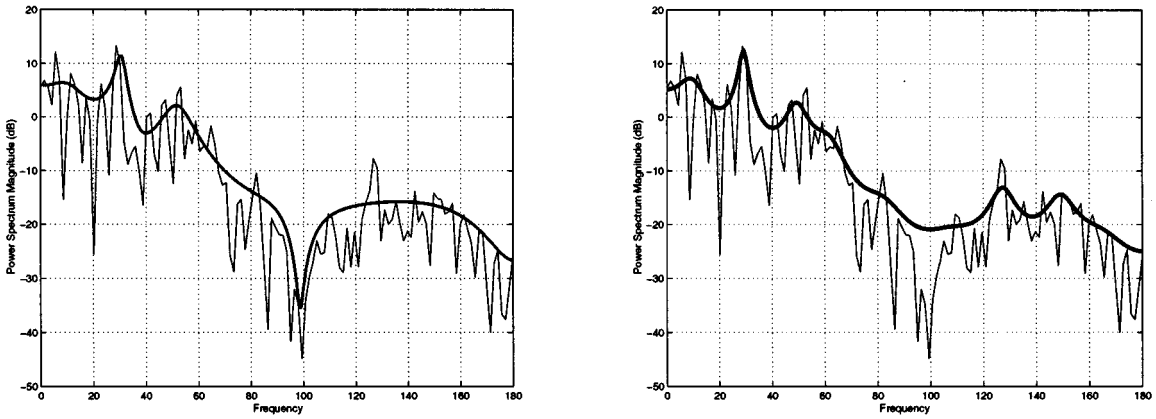

Fig. 17. Periodogram with a tenth-order LLN (left) and 20th-order LPC spectral envelope (right).

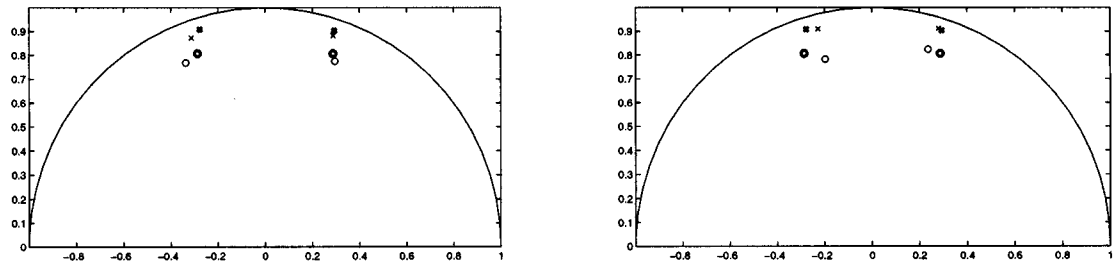

Fig. 18. Poles and zeros for using CCM zeros (left) and zeros from [35] i) (right) versus true ones (bold).

TABLE II
COMPARING LLN USING ZEROS FROM CCM AND FROM [35] WITH THE CORRESPONDING RESULT IN [35]

	LLN with CCM zeros	LLN with zeros from [35](i)	CCM	[35](i)
cepstral error	.0270	.0395	.0098	.0561
covariance error	.0099	.0191	.0428	.0765

Since these filters can be realized in lattice-ladder form, and since this provides a design method for deriving modeling filters matching a covariance window but having arbitrary stable zeros (or "notches" in the power spectrum of the ARMA model), these filters are referred to in this paper as "lattice-ladder notch" filters, or LLN filters. An algorithm is presented for realizing LLN filters in lattice-ladder form, given the covariance window and the moving average part of the model. This is illustrated using refinements of the spectral estimates developed in Section III for a frame of unvoiced speech.

Finally, we illustrate the fact that while we also show how to determine the moving average part using cepstral smoothing, one can make use of any *a priori* (e.g., ARMA modeling) estimate for the system zeros to initialize an enhancement of the modeling filter as well as to obtain better covariance matching. Indeed, we concluded the paper with an example of this method, incorporating an ARMA modeling technique from the literature to obtain an initial estimate of the system zeros.

REFERENCES

[1] G. Alengrin and J. Zerubia, "A method to estimate the parameters of an ARMA model," *IEEE Trans. Automat. Contr.*, vol. AC-32, Dec. 1987.
[2] M. Aoki, *State Space Modeling of Time Series*. New York: Springer-Verlag, 1987.
[3] K. J. Åström, *Introduction to Stochastic Realization Theory*. New York: Academic, 1970.
[4] ——, "Evaluation of quadratic loss functions for linear systems," in *Fundamentals of Discrete-Time Systems: A Tribute to Professor Eliahu I. Jury*, M. Jamshidi, M. Mansour, and B. D. O. Anderson, Eds. Albuquerque, NM: IITSI, 1993, pp. 45–56.
[5] C. G. Bell, H. Fujisaki, J. M. Heinz, K. N. Stevens, and A. S. House, "Reduction of speech spectra by analysis-by-synthesis techniques," *J. Acoust. Soc. Amer.*, vol. 33, pp. 1725–1736, 1961.
[6] C. I. Byrnes and A. Lindquist, "On the geometry of the Kimura–Georgiou parameterization of modeling filter," *Int. J. Contr.*, no. 50, pp. 2301–2312, 1989.
[7] ——, "Rational solutions of the covariance extension problem," in *Linear Circuits, Systems and Signal Processing: Theory and Application*, C. I. Byrnes, C. F. Martin, and R. E. Saeks, Eds. Amsterdam, The Netherlands: North-Holland, 1988.
[8] C. I. Byrnes, A. Lindquist, S. V. Gusev, and A. V. Matveev, "A complete parameterization of all positive rational extensions of a covariance sequence," *IEEE Trans. Automat. Contr.*, vol. 40, pp. 1841–1857, 1995.
[9] C. I. Byrnes and A. Lindquist, "On the partial stochastic realization problem," *IEEE Trans. Automat. Contr.*, vol. 42, pp. 1049–1069, 1997.
[10] C. I. Byrnes, S. V. Gusev, and A. Lindquist, "A convex optimization approach to the rational covariance extension problem," *SIAM J. Contr. Optim.*, vol. 37, pp. 211–229, 1999.
[11] J. A. Cadzow, "Spectral estimation: An overdetermined rational model equation approach," *Proc. IEEE*, vol. 70, pp. 907–939, 1982.
[12] A. Dahlén, A. Lindquist, and J. Mari, "Experimental evidence showing that stochastic subspace identification methods may fail," *Syst. Contr. Lett.*, vol. 34, pp. 303–312, 1998.
[13] P. Delsarte, Y. Genin, Y. Kamp, and P. van Dooren, "Speech modeling and the trigonometric moment problem," *Philips J. Res.*, vol. 37, pp. 277–292, 1982.
[14] P. A. Fuhrmann, *A Polynomial Approach to Linear Algebra*. New York: Springer-Verlag, 1996.

[15] T. T. Georgiou, "Partial realization of covariance sequences," Ph.D. dissertation, Univ. Florida, Gainesville, 1983.
[16] ——, "Realization of power spectra from partial covariance sequences," *IEEE Trans. Acoust., Speech Signal Processing*, vol. ASSP-35, pp. 438–449, 1987.
[17] G. Heinig, P. Jankowski, and K. Rost, "Fast inversion algorithms of Toeplitz-plus-Hankel matrices," *Numer. Math.*, vol. 52, pp. 665–682, 1988.
[18] L. B. Jackson, *Digital Filters and Signal Processing*. Boston, MA: Kluwer.
[19] R. E. Kalman, "Realization of covariance sequences," in *Proc. Toeplitz Memorial Conf.*, Tel Aviv, Israel, 1981.
[20] H. Kimura, "Positive partial realization of covariance sequences," in *Modeling, Identification and Robust Control*, C. I. Byrnes and A. Lindquist, Eds. Amsterdam, The Netherlands: North-Holland, 1987, pp. 499–513.
[21] G. E. Kopec, A. V. Oppenheim, and J. M. Tribolet, "Speech processing by homomorphic prediction," *IEEE Trans. Acoust., Speech Signal Processing*, vol. ASSP-25, pp. 40–49, 1977.
[22] H. J. Laundau, Ed., *Moments in Mathematics*. Providence, RI.: Amer. Math. Soc., 1987.
[23] A. Lindquist and G. Picci, "Canonical correlation analysis, approximate covariance extension, and identification of stationary time series," *Automatica*, vol. 32, pp. 709–733, May 1996.
[24] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, pp. 561–580, 1975.
[25] R. J. Mammone and X. Zhang, "Robust speech processing as an inverse problem," in *The Digital Signal Processing Handbook*, V. K. Madisetti and D. B. Williams, Eds. New York: IEEE Press, 1998, ch. 27.
[26] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. Berlin, Germany: Springer-Verlag, 1976.
[27] J. M. F. Moura and M. I. Ribeiro, "Parametric spectral estimation for ARMA processes," in *Proc. 3rd Workshop Spectrum Estimation Modeling*, Boston, MA, Nov. 1986.
[28] M. Minoux, *Mathematical Programming: Theory and Algorithms*. New York: Wiley, 1986.
[29] A. V. Oppenheim, G. E. Kopec, and J. M. Tribolet, "Signal analysis by Homomorphic prediction," *IEEE Trans. Acoust., Speech Signal Processing*, vol. ASSP-24, pp. 327–332, 1976.
[30] A. V. Oppenheim and R. W. Shafer, *Digital Signal Processing*. London, U.K.: Prentice-Hall, 1975.
[31] B. Porat, *Digital Processing of Random Signals*. London, U.K.: Prentice-Hall, 1994.
[32] L. R. Rabiner, B. S. Atal, and J. L. Flanagan, "Current methods of digital speech processing," in *Selected Topics in Signal Processing*, S. Haykin, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1989, pp. 112–132.
[33] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ.: Prentice-Hall, 1978.
[34] M. I. Ribeiro and J. M. F. Moura, "LD^2-ARMA identification algorithm," *IEEE Trans. Signal Processing*, vol. 39, pp. 1822–1835, Aug. 1991.
[35] M. I. Ribeiro, J. Zerubia, J. M. F. Moura, and G. Alengrin, "Comparison of two ARMA estimators," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 4, 1989, pp. 2186–2189.
[36] T. Söderström and P. Stoica, *System Identification*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[37] P. Stoica and R. Moses, *Introduction to Spectral Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1997.
[38] K. Stieglitz, "On the simultaneous estimation of poles and zeros in speech analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 229–234, 1977.
[39] P. van Overschee and B. De Moor, "Subspace algorithms for stochastic identification problem," *Automatica*, vol. 29, pp. 649–660, 1993.
[40] G. Wilson, "Factorization of the covariance generating function of a pure moving average process," *SIAM J. Numer. Anal.*, vol. 6, pp. 1–7, 1969.

Christopher I. Byrnes (M'80–SM'85–F'89) received the B.S. degree from Manhattan College, Bronx, NY, in 1971 and the M.S. and Ph.D. degrees from the University of Massachusetts, Amherst, in 1973 and 1975, respectively.

He was an Instructor with the Department of Mathematics, University of Utah, Salt Lake City, from 1975 to 1978, when he was appointed Assistant Professor with the Department of Mathematics and the Division Applied Sciences, Harvard University, Cambridge, MA. From 1982 to 1985, he was an Associate Professor of Applied Mathematics with the Gordon McKay Endowment at Harvard. From 1984 to 1989, he was with Arizona State University, Tempe. He is currently Dean of the School of Engineering and Applied Science and Professor with the Department of Systems Science and Mathematics, Washington University, St. Louis, MO. His research interests include adaptive control, distributed parameter systems, and linear and nonlinear control.

Per Enqvist received the M.S. degree in engineering physics from the Royal Institute of Technology, Stockholm, Sweden, in 1994. He is presently pursuing the Ph.D. degree with the Division of Optimization and Systems Theory, the Royal Institute of Technology, Stockholm, Sweden. His main research interest is stochastic realization theory, especially time series analysis and interpolation problems.

Anders Lindquist (M'77–SM'86–F'89) received the Ph.D. degree from the Royal Institute of Technology (KTH), Stockholm, Sweden, where, in 1972, he was appointed a Docent of Optimization and Systems Theory.

From 1972 to 1974, he held visiting positions at the University of Florida, Gainesville, Brown University, Providence, RI, and the State University of New York, Albany. In 1974, he became an Associate Professor and, in 1980, a Professor at the University of Kentucky, Lexington, where he remained until 1983. He is now a Professor at KTH, where, in 1982, he was appointed to the Chair of Optimization and Systems Theory. Presently, he is the Chairman of the Mathematics Department at KTH. He has been an Affiliate Professor of Optimization and Systems Theory at Washington University, St. Louis, MO, since 1989. He is the author of almost 100 papers in the area of systems and control and signal processing, especially stochastic control, filtering, stochastic systems theory, realization theory, identification, robust control, interpolation theory, and applications of nonlinear dynamics and global analysis in estimation and control. He has served on many editorial boards of journals and book series.

Dr. Lindquist is a member of the Royal Swedish Academy of Engineering Sciences, a foreign member of the Russian Academy of Natural Sciences, an honorary member the Hungarian Operations Research Society, and an Advisory Board member of the Institute for Mathematics of the Life Sciences at Texas Tech University, Lubbock.