Abstract—A multichannel characterization for autoregressive moving average (ARMA) spectrum estimation in subbands with applications to cognitive radio spectrum sensing is considered in this presentation. The fullband ARMA spectrum estimation can be realized in two-channels as a special form of this characterization. A complete orthogonalization of input multichannel data is accomplished using a modified form of sequential processing multichannel lattice stages. Matrix operations are avoided, only scalar operations are used, and a multichannel ARMA prediction filter with a highly modular and suitable structure for VLSI implementations is achieved. The filter is therefore considered as a good candidate for FPGA and DSP chip based spectrum sensing functions in software defined radio implementations. Lattice reflection coefficients for autoregressive and moving average parts are simultaneously computed. These coefficients are then converted to process parameters using a newly developed Levinson-Durbin type multichannel conversion algorithm, so that they can be utilized in the classification of the spectrum. Coarse and fine spectrum sensing can be accomplished using different prediction filter configurations. The computational complexity is given in terms of model order parameters. The performance of the proposed method is compared visually and statistically to that of the multitaper method.

I. INTRODUCTION

An important recent development in the design of wireless communication systems is the cognitive radio (CR), built on a software defined radio, functions as an intelligent system that is aware of its environment and uses the methodology of understanding-by-building to learn from the environment and adapt to statistical variations in the input stimuli in order to establish reliable communication by efficient utilization of the radio spectrum [1]. In CR systems, secondary users must have the capability to detect and opportunistically exploit under utilized spectral resources allocated to primary users. As a consequence of this fact, spectrum sensing functionality has emerged as an integral part of CR systems. Different approaches have been proposed to perform spectrum sensing function in CR systems such as matched filtering, the use of special signal features to detect and classify primary signal, energy detection, and spectral analysis by filter bank and multitaper methods [2]–[5].

Spectral analysis methods are preferable if the spectral properties of the signal to be detected are known, but the signal has otherwise no usable features that can be efficiently exploited. Accordingly, the desirable attributes of spectral analysis methods with application to spectrum sensing are their estimation accuracies and resolution characteristics. It has been reported in [1] that the multitaper method can produce an accurate estimate of the desired power spectrum by trading spectral resolution for improved spectral characteristics, that is, it can reduce variance of the spectral estimate without compromising the bias of the estimate. Additionally, the multitaper method outperformed the multiple signal classification (MUSIC) algorithm and the modified forward-backward linear prediction (MFBLP) algorithms in so far as the continuous parts of the spectrum are concerned, but did not perform well in dealing with the line spectral components. Filter bank methods were compared against the filter bank reinterpretation of the multitaper method with applications to spectrum sensing by Farhang-Boroujeny [4], and it was noted that, if a filter bank multicarrier technique is exploited as the physical layer of a CR network, then the same filter bank can also be used for channel sensing at virtually no cost.

On the contrary to nonparametric methods, parametric methods for spectrum sensing applications have not attracted much attention even though they deserve consideration particularly when they are implemented in subbands due to Rao and Pearlman’s findings presented in [6], which showed that the well-known autoregressive (AR) modeling was a promising method for spectrum estimation in subbands, and that $p$th-order prediction from subbands was superior to $p$th-order prediction in the fullband when $p$ was finite, and subband decomposition of a source resulted in a whitening of the composite subband spectrum. The equivalence of linear prediction and AR spectrum estimation was then exploited by Rao and Pearlman [6] to show that AR spectrum from subbands offers a gain over fullband AR spectrum estimation due to SNR and frequency spacing improvement as well as whitening effects. Furthermore, processes that are purely autoregressive are often transformed into ARMA processes by addition of measurement noise, and especially sinusoids in noise are known to obey the degenerate ARMA equation in fullband as well as in subbands [7]. Consequently, we think that developing new methods of subband spectrum estimation based on ARMA modeling is particularly relevant for spectrum sensing of multicarrier cognitive radio signals [8].

The concept of software radio on the other hand, as CR’s backbone, relies on the development of digital signal processing (DSP) technology, that is flexible, reconfigurable, and reprogrammable by software to adapt to an environment where there are multiple services, standards, and frequency bands [9]. As such, the infrastructure in a software radio system is generally required to use reconfigurable VLSI hardware components such as DSP chip sets, field programmable gate arrays (FPGAs), embedded processors, and even general pur-
pose processors [10]–[12].

Under these considerations, we propose a new subband spectrum estimation method based on ARMA modeling, which can also be viewed as a parametric filter bank method, and can meet the requirements for spectrum sensing in cognitive radios because of its highly modular, regular, and order recursive prediction filter structure, its suitability for reconfigurable VLSI implementations, and its spectrum estimation accuracy and resolution characteristics that improves with increasing number of subbands.

The proposed method relies on estimation of the driving noise in subbands, such that the subbands ARMA filtering problem is at first transformed into multichannel AR filtering problem by embedding subband ARMA processes into multichannel AR processes, and then a complete modified Gram-Schmidt orthogonalization of multichannel input signal is accomplished using a modified version of the sequential processing multichannel lattice stages (SPMLSs) [13]. Thus, the complete orthogonalization of multichannel input data and sequential nature of the modified SPMLSs make it possible to feed back the delayed forward prediction error signals to represent the unknown input noise signals of original ARMA processes, and lends itself to a novel ARMA prediction filter structure, which to the best of our knowledge does not exist in the literature and can be reconfigured as an equalizer during signal reception [14], and to the development of a new Levinson-Durbin type conversion algorithm for the modified SPMLSs in order to compute ARMA process parameters from lattice reflection coefficients.

A two-subband ARMA spectrum estimation problem is considered due to the ease of explanation and space limitations in developing the method. However, it is considered straightforward to apply the method to any number of subbands, and to AR spectrum estimation in subbands. The method is also appropriate for uniform as well as non-uniform filter bank realizations.

The paper is organized as follows. In Section 2, we present the development of the new multichannel ARMA lattice prediction filter using the modified SPMLSs. In Section 3, we develop the new Levinson-Durbin type multichannel conversion algorithm for the modified SPMLS, and relate lattice parameters to process parameters. Spectrum estimation expression in two subbands is given in Section 4. The computational complexity computations are treated in Section 5. Section 6 is concerned with the experimental results. Finally, Section 7 is about the discussions of results and conclusions. The following notations are used in this presentation. $(\bullet)^*$ represents the complex conjugate of $(\bullet)$, $(\bullet)^T$ and $(\bullet)^H$ stand for the transpose and the Hermitian transpose of $(\bullet)$ respectively. The variables $m$, $i$ and $n$ are global while all other variables are local. The variable $m$ represents the stage number while $n$ and $i$ are the time indexes related to data and coefficients respectively till we equate them in Section 3 to have a single time index.

II. ADAPTIVE MULTICHANNEL ARMA LATTICE PREDICTION FILTERING

A. Multichannel Prediction Problem

An illustration of the adaptive multichannel ARMA prediction filtering in subbands for two-subbands case is presented in figure 1. Therein, $y(n)$ represents the input fulldband signal while $y_1(n)$ and $y_2(n)$ stand for the input subband signals. In adaptive multichannel ARMA prediction filtering, the objective is to find an exponentially windowed, LS solution for the AR and MA coefficients of the $k$th forward prediction filter that minimizes each of the two cost functions,

$$J^k(i) = \sum_{n=0}^{i} \lambda^{-n} | f^k_{p_k}(n) |^2$$

at each time instant $i$, and $k = 1, 2$. The forward prediction error $f^k_{p_k}(n)$ in this expression is defined as,

$$f^k_{p_k}(n) = d^k(n) - \hat{d}^k_i(n),$$

and the $k$th forward prediction filter output, $\hat{d}^k_i(n)$, is an estimate of the $k$th desired signal, $d^k(n) = y_k(n)$, is given by

$$\hat{d}^k_i(n) = \sum_{j=1}^{p_k} \hat{a}^k_{1,j}(i)y_k(n-j) + \sum_{l=0}^{q_k} \hat{a}^k_{2,l}(i)\hat{u}_k(n-l).$$

Herein, $p_k$ and $q_k$ denote the order of the $(p_k, q_k)$ prediction error filter associated with the $k$th subband, and $\hat{u}_k(n)$ is the estimate of the $k$th ARMA process input signal. The estimated $k$th ARMA process input signal, $\hat{u}_k(n)$, is obtained by delaying and feeding back the $p_k$th order forward prediction error, $\hat{u}_k(n) = f^k_{p_k}(n-1)$. Hence, the input vector to the $k$th ARMA filter at time instant $n$, $\hat{y}_k(n)$, and the corresponding coefficient vector $\hat{a}^k(i)$, at time instant $i$, are defined as

$$\hat{y}_k(n) = [y_k(n-1), \ldots, y_k(n-p_k), \hat{u}_k(n), \hat{u}_k(n-1), \ldots, \hat{u}_k(n-q_k)]^T$$

and

$$\hat{a}^k_T(i) = [\hat{a}^k_{1,1}(i), \ldots, \hat{a}^k_{1,p_k}(i), \hat{a}^k_{2,0}(i), \hat{a}^k_{2,1}(i), \ldots, \hat{a}^k_{2,q_k}(i)]$$

respectively. Herein, $\hat{a}^k_{1,j}(i)$ and $\hat{a}^k_{2,l}(i)$ respectively represent the $j$th coefficient related to the AR and MA parts of the forward prediction filter for the $k$th subband at time instant $i$. It is assumed, without loss of generality, that $p_k \geq q_k$. $p_k = q_k$ case corresponds to the prediction filter for an ARMA($p_k$,$q_k$) process, while $p_k > q_k$ prediction filter is for a general ARMA($p_k$,$q_k$) process. Note that an ARMA backward prediction can be performed for the desired signal, $\hat{d}^k(n) = y_k(n-p_k)$, and the prediction filter in that case would use the reversed and conjugated forward prediction filter coefficients, which are defined in the backward prediction error coefficient vector as

$$\hat{c}^k_T(i) = [\hat{c}^k_{1,p_k}(i), \ldots, \hat{c}^k_{1,1}(i), \hat{c}^k_{2,q_k}(i), \ldots, \hat{c}^k_{2,1}(i), \hat{c}^k_{2,0}(i)]$$

(6)
where $\hat{c}_{k,j}(i)$ and $\tilde{c}_{k,j}(i)$ are respectively defined as the $j$th coefficient related to the AR and MA parts of the backward prediction filter for the $k$th subband at time instant $i$
.

Consequently, the main concern of the exponentially weighted LS problem under consideration is to find, at each time $i$, the $k$th optimal coefficient vector, $\hat{a}^k(i)$ that would minimize the cost function,

$$J^k(i) = \sum_{n=0}^{i} \lambda^{i-n} \left| d^k(n) - \hat{a}^k(i) \tilde{y}_k(n) \right|^2.$$  

(7)

The $k$th optimal coefficient vector related to the $k$th subband filter,

$$\hat{a}_{opt}^k(i) = R_k^{-1}(i)P_k(i)$$  

(8)

is found by differentiating $J^k(i)$ with respect to $\hat{a}^k(i)$, setting the derivative to zero, and solving for $\hat{a}^k(i)$, where

$$R_k(i) = \sum_{n=0}^{i} \lambda^{i-n} \tilde{y}_k(n) \tilde{y}_k^H(n)$$  

(9)

and

$$P_k(i) = \sum_{n=0}^{i} \lambda^{i-n} \tilde{y}_k(n) \ d_k^*(n).$$  

(10)

B. Sequential Lattice Orthogonalization

In order to find a modular, regular, and simple solution to the two subbands ARMA prediction problem, we would like to use a single multichannel lattice filter as depicted in figure 2, instead of using two separate transversal filters and solving two separate optimization problems as in figure 1. We would also like to avoid direct evaluations as in (8), and achieve good numerical properties. As the number of channels at different sections of the proposed multichannel lattice filter is different due to the sequential processing nature of SPMLSs, we carry out the exponentially weighted LS optimization problem by taking into consideration each of these sections separately, and therefore we assume that the filter is comprised of three cascaded filters, which are two-channel, three-channel, and four-channel lattice sections ; and we use a different index for each section while using $m$ to indicate a stage in the whole filter. We also assume $p_1 = p_2$ for the ease of explanation without loss of generality.

In order to sequentially solve the exponentially weighted LS optimization problem under consideration, we first organize the elements of input signal vectors $y_1(n) = [y_1(n), \ldots, y_1(n-\ell)]^T$, and $y_2(n) = [y_2(n), \ldots, y_2(n-\ell)]^T$ according to the natural ordering of SPMLSs as :

$$\tilde{y}_{\ell+1}(n) = \begin{bmatrix} y_1(n) \\ y_2(n) \\ y_1(n-1) \\ y_2(n-1) \\ \vdots \\ y_1(n-\ell) \\ y_2(n-\ell) \end{bmatrix},$$  

(11)

and input to two-channel stages for which the stage number $(m)$ has a range of values given by $0 < m \leq (p_1 - q_1)$. Accordingly, we redefine equations (9) and (10) using this new data vector as follows,

$$R_{\ell}(i) = \sum_{n=0}^{i} \lambda^{i-n} \tilde{y}_{\ell+1}(n) \tilde{y}_{\ell+1}^H(n)$$  

(12)

and

$$P_{\ell,k}(i) = \sum_{n=0}^{i} \lambda^{i-n} \tilde{y}_{\ell+1}(n) \ d_k^*(n)$$  

(13)

where $k = 1, 2$. The orthogonalization of data using SPMLSs corresponds to the transformation of (12) and (13) into

$$D_{\ell+1}^f(i) = \sum_{n=0}^{i} \lambda^{i-n} \Omega_{\ell}^f(i) \tilde{y}_{\ell+1}(n) \tilde{y}_{\ell+1}^H(n)$$  

(14)

and

$$Z_{\ell+1,k}^f(i) = \sum_{n=0}^{i} \lambda^{i-n} \Omega_{\ell}^f(i-1) \tilde{y}_{\ell+1}(n-1) \ d_k^*(n)$$  

(15)

respectively. Here, $\Omega_{\ell}^f(i)$ is the $2\ell \times 2\ell$ lower triangular transformation matrix for forward prediction, and is sequentially realized stage-by-stage using $2 \times 2$ lower triangular transformation matrices,

$$L_{\ell}^f(i) = \begin{bmatrix} 1 & 0 \\ \tilde{r}_{\ell}^f(i-1) & 1 \end{bmatrix}$$  

(16)

whose diagonal elements are all equal to unity at time instant $i$, and $\tilde{r}_{\ell}^f(i)$ is the reflection coefficient computed at the single circular cell in the triangular shaped self-orthogonalization processor of the $\ell$th two-channel SPMLS. Then, the forward lattice predictor coefficients are computed using

$$\Theta_{\ell,k}^f(i) = D_{\ell+1}^{-1}(i) Z_{\ell+1,k}^f(i)$$  

(17)

where $\Theta_{\ell,k}^f(i)$ represents the $k$th row of the $2 \times 2\ell$ lattice forward prediction reflection coefficient matrix $\Theta_{\ell}^f(i)$, and is also sequentially implemented stage-by-stage by means of $2 \times 2$ forward prediction reflection coefficient matrices,

$$\Delta_{\ell}^f(i) = \begin{bmatrix} \tilde{r}_{\ell+1,k}^f(i) & \tilde{r}_{\ell+2,k}^f(i) \\ \tilde{r}_{\ell+1,k}^f(i) & \tilde{r}_{\ell+2,k}^f(i) \end{bmatrix}$$  

(18)

in which $\tilde{r}_{\ell+1,k}^f(i)$ is the $j$th reflection coefficient related to the forward prediction of the $k$th channel signal, and it is computed at the $(k, j)$th single circular cell of the square shaped reference-orthogonalization processor related to forward prediction at the $\ell$th two-channel SPMLS. Note that the matrix inversion operation in equation (8) is now transformed into a simple scalar inversion operation in (17) due to the diagonal nature of $D_{\ell+1}^{-1}(i)$. The backward prediction counterpart of this optimization problem is similarly solved using $2 \times 2$ lower triangular transformation matrices, $L_{\ell}^b(i)$, and $2 \times 2$ lattice backward prediction reflection coefficient matrices, $\Delta_{\ell}^b(i)$.

After the processing of input signals by two-channel lattice stages, the delayed and fed back forward prediction error
\( \hat{u}_1(n) = f_{p_1}(n - 1) \) is incorporated at the \((p_1 - q_1 + 1)\)th stage, as the third channel. Accordingly, we expand the optimization problem by organizing the elements of the input data vectors \( y_1(n) = [y_1(n), \ldots, y_1(n - \nu)]^T, y_2(n) = [y_2(n), \ldots, y_2(n - \nu)]^T, \) and \( \hat{u}_1(n) = [\hat{u}_1(n), \ldots, \hat{u}_1(n - \nu)]^T \) as follows:

\[
\mathbf{Y}_{\alpha+1}(n) = \begin{bmatrix}
    y_1(n) \\
    y_2(n) \\
    \hat{u}_1(n) \\
    \vdots \\
    y_1(n - \alpha) \\
    y_2(n - \alpha) \\
    \hat{u}_1(n - \alpha)
\end{bmatrix}, \quad (19)
\]

and input to three-channel lattice section, where the stage number \((m)\) takes values in the range given by \((p_1 - q_1) < m \leq (p_2 - q_2)\). Subsequently, we solve the optimization problem in (17) once again with the new input vector, in which \( \mathbf{\Omega}_f(i) \) and \( \mathbf{\Theta}_f(i) \) are the \(3\alpha \times 3\alpha\) lower triangular matrix and the \(3 \times 3\) forward lattice prediction matrices respectively. \( \mathbf{\Omega}_f(i) \) is computed sequentially by means of \(3 \times 3\) lower triangular transformation matrices, \( \mathbf{L}_f(i) \), and \( \mathbf{\Theta}_f(i) \) is similarly realized stage-by-stage making use of \(3 \times 3\) forward prediction coefficient matrices, \( \mathbf{\Delta}_f(i) \), at time instant \(i\). Note that, since the delayed and fed back signal is considered to constitute a new channel in the multichannel sequential lattice filtering, we have three desired signals at this point, \( d^k(n) \), where \(k = 1, 2, 3\), one of which did not exist as a separate channel in the optimization problem stated in Subsection II.A, and this new desired signal, \( d^4(n) \), is related to the MA part of the first subband ARMA modeling.

Finally, the optimization problem is expanded one more time with the inclusion of the second delayed and fed back forward prediction error \( \hat{u}_2(n) = f_{p_2}(n - 1) \), and this time, the elements of input data vectors \( y_1(n) = [y_1(n), \ldots, y_1(n - \nu)]^T, y_2(n) = [y_2(n), \ldots, y_2(n - \nu)]^T, \) and \( \hat{u}_2(n) = [\hat{u}_2(n), \ldots, \hat{u}_2(n - \nu)]^T \) are organized similar to (11) and (19) with the stage number \((m)\) is in the range given by \((p_2 - q_2) < m \leq p_2\) due to four channel processing. Similar to two-channel and three-channel cases, we solve the optimization problem in (17) using the new data vector, in which case \( \mathbf{\Omega}_f(i) \) and \( \mathbf{\Theta}_f(i) \) are \(4\nu \times 4\nu\) lower triangular matrices, and \(4 \times 4\nu\) forward lattice prediction coefficient matrices at the time instant \(i\) respectively. As in previous sections, these matrices are computed stage-by-stage by use of \(4 \times 4\) lower triangular transformation matrices, \( \mathbf{L}_f(i) \), and \(4 \times 4\) forward prediction coefficient matrices, \( \mathbf{\Delta}_f(i) \), at time instant \(i\) respectively. Since the second delayed and fed back signal is also considered as a new channel in the multichannel sequential lattice filtering, hereafter we have four desired signals, \( d^k(n) \), where \(k = 1, 2, 3, 4\), and this fourth desired signal, \( d^4(n) \), is associated with the MA part of the second subband ARMA modeling. Consequently, we present a diagram of the four channel lattice prediction filter structure under consideration in figure 3.

### III. CONVERSION OF LATTICE COEFFICIENTS TO PROCESS PARAMETERS

Since the mathematical link between process parameters and reflection coefficients of a lattice prediction filter is provided by the Levinson-Durbin algorithm [15], we develop a new Levinson-Durbin type conversion algorithm specifically for SPMLs in order to convert lattice reflection coefficients to ARMA process parameters. Due to the sequential nature of the proposed lattice structure, we carry out the development of the new Levinson-Durbin type multichannel conversion algorithm by taking into consideration each of these sections separately, and therefore we assume that the filter is comprised of three cascaded filters as discussed in Subsection II.B.

We first consider the conversion algorithm for the two-channel section of lattice prediction filter, and we organize the input signal samples to two-channel section as

\[
\mathbf{Y}_{\ell+1}(n) = \begin{bmatrix}
    y_1(n) \\
    y_2(n) \\
    y_1(n - \ell) \\
    y_2(n - \ell)
\end{bmatrix} = \begin{bmatrix}
    y_1(n) \\
    y_2(n) \\
    y_1(n - 1) \\
    y_2(n - 1)
\end{bmatrix} \quad (20)
\]

where we define the data vectors as \( y_1(n) = [y_1(n), \ldots, y_1(n - \ell + 1)]^T, y_2(n) = [y_2(n), \ldots, y_2(n - \ell + 1)]^T, \) and \(0 < m \leq (p_1 - q_1)\). The corresponding forward and backward prediction error coefficient matrices for the \(\ell\)th order transversal filter for the \(4\)th channel are defined as

\[
\hat{a}^k_T(i) = [\hat{a}_1^k(i), \hat{a}_2^k(i), \hat{a}_3^k(i), \hat{a}_4^k(i), \hat{a}_5^k(i), \hat{a}_6^k(i)]
\]

\[
\hat{a}^k_T(i) = [\hat{c}_1^k(i), \hat{c}_2^k(i), \hat{c}_3^k(i), \hat{c}_4^k(i), \hat{c}_5^k(i), \hat{c}_6^k(i)]
\]

where \(k = 1, 2\) due to two-channel processing, and \(\hat{c}_1^k = \hat{c}_2^k = 1.0\). Since the signal time shifting and ordering properties of SPMLs when expressed in matrix form as in equation (11) are different than the organization of input signal samples in matrix form for transversal filtering as shown in equation (20), we use \((2\ell + 1) \times (2\ell + 1)\) shuffling matrices, \(J_{\ell+1}^1\) for the first channel and \(J_{\ell+1}^2\) for the second channel, to reorder the elements of coefficient matrices, \(\hat{a}_1^H(i), \hat{a}_2^H(i)\), and \(\hat{c}_1^H(i), \hat{c}_2^H(i)\), according to the sample ordering of SPMLs. Therefore, the forward and backward prediction errors for the end of the observation interval \(n = i\) at the output of the general \(\ell\)th order filters with transversal structure can be stated as:

\[
\begin{bmatrix}
    f_1^k(n) \\
    f_2^k(n)
\end{bmatrix} = \begin{bmatrix}
    J_{\ell+1}^1 \hat{a}_1^H(n) & 0 \\
    0 & J_{\ell+1}^2 \hat{a}_2^H(n)
\end{bmatrix} \begin{bmatrix}
    \mathbf{Y}_{\ell+1}(n)
\end{bmatrix}
\]

\[(23)\]

\[
\begin{bmatrix}
    b_1^k(n) \\
    b_2^k(n)
\end{bmatrix} = \begin{bmatrix}
    J_{\ell+1}^1 \hat{c}_1^H(n) & 0 \\
    0 & J_{\ell+1}^2 \hat{c}_2^H(n)
\end{bmatrix} \begin{bmatrix}
    \mathbf{Y}_{\ell+1}(n)
\end{bmatrix}
\]

\[(24)\]
where $\mathbf{0}$ is a $1 \times (\ell + 1)$ zero matrix. We then express the $(\ell - 1)$th prediction error for transversal prediction filtering. Note that the size of each coefficient matrix increases by two when the order of prediction filter increases from $(\ell - 1)$ to $\ell$, and $\mathbf{0}$ is a $1 \times (\ell + 1)$ zero matrix. Subsequently, we also define the $\ell$th order lattice prediction errors in terms of lattice parameters and the $(\ell - 1)$th order forward and backward prediction errors. The lattice parameters appear in the lower coefficient triangular and square matrices, which are generated in triangular shaped self-orthogonalization, and square shaped reference-orthogonalization processors in two-channel SPMLs, as previously defined in equations (16) and (18). Accordingly, the $\ell$th and the $(\ell - 1)$th order prediction error matrices for transversal prediction filter, are substituted in the $(\ell)$th order lattice prediction error expressions so as to obtain the pairs of the Levinson-Durbin recursions for two-channel section. Note that we can not provide a complete treatment of the aforementioned matrix operations due to space limitations.

The three-channel section starts with the incorporation of the third channel ($\hat{u}_3(n)$) as the new channel at the $(p_1 - q_1 + 1)$th stage. The Levinson-Durbin algorithm for this section is developed similar to two-channel section by assuming that three-channel section is a separate filter, and thereby considering the input signal samples to the three-channel section as follows:

$$
\mathbf{y}_{\alpha+1}(n) = \begin{bmatrix}
\mathbf{y}_1(n) \\
y_1(n - \alpha) \\
y_1(n - 1)
\end{bmatrix} = \begin{bmatrix}
y_1(n) \\
\mathbf{y}_2(n) \\
y_2(n - \alpha) \\
y_2(n - 1)
\end{bmatrix} = \begin{bmatrix}
\hat{u}_1(n) \\
\hat{u}_1(n - \alpha) \\
\hat{u}_1(n - 1)
\end{bmatrix}
$$

(25)

where $\mathbf{y}_1(n) = [y_1(n), \ldots, y_1(n - \alpha + 1)]^T$, $\mathbf{y}_2(n) = [y_2(n), \ldots, y_2(n - \alpha + 1)]^T$, and $\hat{u}_1(n) = [\hat{u}_1(n), \ldots, \hat{u}_1(n - \alpha + 1)]^T$. Correspondingly, the forward and backward prediction error coefficient matrices for the $\nu$th order transversal filtering are defined as

$$
\hat{a}^kT(n) = [\hat{a}_0^k(n), \hat{a}_1^k(n), \hat{a}_2^k(n), \hat{a}_3^k(n), \ldots, \hat{a}_{q\nu-2}^k(n)], \quad (26)
$$

$$
\hat{c}^kT(n) = [\hat{c}_0^k(n), \hat{c}_1^k(n), \hat{c}_2^k(n), \hat{c}_3^k(n), \ldots, \hat{c}_{q\nu-1}^k(n)], \quad (27)
$$

where $k = 1, 2, 3$ due to three-channel processing. Then, the prediction filtering continues with three-channel lattice stages for $(p_1 - q_1) < m \leq (p_2 - q_2)$. The Levinson-Durbin recursions for three-channel section can be developed similar to two-channel section by establishing the mathematical link between transversal and lattice filter coefficients, that is, by substituting $(\ell - 1)$th order error expressions for transversal filtering into the $\nu$th order prediction error expressions for lattice filtering. Since the organization of signal samples in equation (25) is different than the ordering of signal samples entering into three-channel SPMLs in (19), we use $(3\alpha + 1) \times (3\alpha + 1)$ shuffling matrices, $J^\nu_{\alpha+1}$ for the first channel, $J^\nu_{\alpha+1}$ for the second channel, and $J^\nu_{\alpha+1}$ for the third channel to reorder the elements of coefficient matrices, $\hat{a}_\nu^H(n), \hat{a}_\nu^{2H}(n), \hat{a}_\nu^{3H}(n)$, and $\hat{c}_\nu^H(n), \hat{c}_\nu^{2H}(n), \hat{c}_\nu^{3H}(n), \hat{c}_\nu^{4H}(n)$, according to the sample ordering of SPMLs. The Levinson-Durbin Recursions for three-channel section are subsequently produced similar to two-channel section.

Finally, the fourth channel($\hat{u}_4(n)$), which represents the feed back and delayed signal related to the second subband, is taken into the orthogonalization process at the $(p_2 - q_2 + 1)$th stage, and the prediction filtering continues with four-channel lattice stages through $(p_2 - q_2) < m \leq p_2$. In order to develop the Levinson-Durbin recursions for this section, we define the forward and backward prediction error coefficient matrices for the $\nu$th order transversal filtering similar to the previous sections, and we also organize of the elements of input vectors $\mathbf{y}_1(n) = [y_1(n), \ldots, y_1(n - \nu + 1)]^T$, $\mathbf{y}_2(n) = [y_2(n), \ldots, y_2(n - \nu + 1)]^T$, $\hat{u}_1(n) = [\hat{u}_1(n), \ldots, \hat{u}_1(n - \nu + 1)]^T$, and $\hat{u}_2(n) = [\hat{u}_2(n), \ldots, \hat{u}_2(n - \nu + 1)]^T$ column wise as we did in equations (20) and (25) for two-channel and three-channel sections respectively. Furthermore, the signal sample ordering for four channel transversal filtering is different than the ordering for lattice filtering, hence we use $(4\nu + 1) \times (4\nu + 1)$ shuffling matrices, $J^\nu_{\alpha+1}$ for the first channel, $J^\nu_{\alpha+1}$ for the second channel, $J^\nu_{\alpha+1}$ for the third channel, and $J^\nu_{\alpha+1}$ for the fourth channel to reorder the elements of coefficient matrices $\hat{a}_\nu^{4H}(n), \hat{a}_\nu^{2H}(n), \hat{a}_\nu^{3H}(n), \hat{a}_\nu^{4H}(n)$, and $\hat{c}_\nu^{4H}(n), \hat{c}_\nu^{2H}(n), \hat{c}_\nu^{3H}(n), \hat{c}_\nu^{4H}(n)$, according to the sample ordering of SPMLs. The development of the Levinson-Durbin recursions for this section unfolds as in two and three channel sections. Firstly, the $\nu$th and the $(\nu - 1)$th order forward and backward prediction errors are stated as outputs of transversal filters. Secondly, the prediction order update equations for the the $\nu$th order four-channel lattice section are given in terms of $(\nu - 1)$th prediction errors and lattice parameters, and finally the $\nu$th and the $(\nu - 1)$th order forward and backward transversal filter prediction error expressions are substituted in the lattice prediction order update equations, such that the pairs of Levinson-Durbin order updates for four-channel section are obtained. Eventually, the Levinson-Durbin recursions for two-channel, three-channel, and four-channel sections are concatenated to attain the complete algorithm for the proposed prediction filter.

IV. SPECTRUM ESTIMATION FROM SUBBANDS

By computing process parameters from lattice coefficients, we have established the link between multichannel prediction and spectrum estimation. Hence, the estimated spectrum in subbands can now be expressed in terms of the subband
prediction filter parameters as
\[
S_y(w_k) = \frac{1}{2} \left( 1 + \alpha_1^2 e^{-jw_k} + \cdots + \alpha_q^2 e^{-jqw_k} \right)^2 \sigma_z^2 + \frac{1}{2} \left( 1 + \alpha_1^4 e^{-jw_k} + \cdots + \alpha_2^4 e^{-jqw_k} \right)^2 \sigma_z^2
\]  
(28)

where \( \sigma_z^2 \) represent the prediction error variance for the \( k \)th subband \( \lambda \) and the coefficients, \( \alpha_1^2, \ldots, \alpha_q^2 \), are related to the AR and MA parts of the \( k \)th subband ARMA spectrum. Note that the coefficients related to the first and second subbands in equation (28) are selected from the elements of coefficient vectors in equations (21), (26), and (??) using a coefficient selection rule, which we can not provide herein due to space limitations.

V. COMPUTATIONAL COMPLEXITY

The number of operations required for the two-channel ARMA lattice prediction filter for fullband spectrum estimation is calculated as \( 10p + 16q \) using the number of operations required for single-channel and two-channel sequential processing lattice stages [13], where “one operation is considered as one multiplication/division and one addition”. The Levinson-Durbin recursion for one-channel lattice sections requires \( (p-q)(p-q+1) \) operations, and \( 4q(q+1) \) operations for two-channel lattice sections to compute the ARMA process parameters. Therefore, the number of operations required to compute \( \alpha_1^2, \ldots, \alpha_q^2 \) and \( \alpha_1^4, \ldots, \alpha_2^4 \) is \( 2q^2 + 11p^2 - 2pqq + 5q^2 + 19q \), and then this expression can be extended to the total number of required operations for an \( M \) subband, multichannel implementation as

\[
\text{Total Complexity} = \sum_{k=1}^{M} p_k^2 + 11p_k^2 - 2pq_kq_k + 5q_k^2 + 19q_k. \quad (29)
\]

This computational complexity can be compared to that of the multitaper method using a FFT algorithm, approximately given by \( (2K+2)N \log_2 N \), where \( K \) is the number of tapers used in the computation of power spectral estimate, and \( N \) is the number of signal samples and is a power of 2.

VI. EXPERIMENTAL RESULTS

We focused on spectral estimation of cisoids in noise during simulation experiments due to its relevance in multicarrier cognitive radio applications. In order to visually demonstrate the frequency spacing, whitening, and SNR improvements achieved by subbands implementation, we present and compare the subband spectrum estimation results with the fullband results. We then present the statistical analysis of our method by performing variance and mean bias analysis in estimating the frequencies of cisoids embedded in white noise. In both cases, we compare the performance against that of the multitaper method [5].

We used a data length of \( N = 128 \), and data was zeropadded to 32 times the data length in all simulations. The forgetting factor was \( \lambda = 0.995 \). We repeated simulation experiments one hundred times; the results of these simulations are then ensemble-averaged. In subband decomposition of input signals, we used the Kaiser window based approach in [16] for designing cosine modulated filter banks. In two and four subband decompositions, the filter lengths are thirty and sixty respectively. In simulations involving the multitaper method, the number of tapers (\( K \)) was chosen as \( K = 3 \) for both visual and statistical analysis plots.

The case of two closely spaced cisoids embedded in white as well as colored case was considered in the simulations. The time series for cisoids in white noise are given by

\[
y(n) = A_1 e^{-j2\pi f_1 n + \phi_1} + A_2 e^{-j2\pi f_2 n + \phi_2} + u(n), \quad 1 \leq n \leq N, \quad (30)
\]

where \( u(n) \) is a white Gaussian complex time series with uncorrelated real and imaginary parts, each with a variance \( \sigma_u^2 \) and zero mean, such that the SNR for the \( k \)th cisoid is defined as

\[
SNR_k = 10 \log_{10} \left( \frac{|A_k|^2}{2\sigma_u^2} \right). \quad (31)
\]

The time series generated by two cisoids embedded in colored noise are similarly expressed by passing the white Gaussian complex noise \( u(n) \), with zero mean and unit variance, through a second order AR coloring filter given by

\[
v(n) = 0.1v(n-1) - 0.3v(n-2) + u(n), \quad 1 \leq n \leq N, \quad (32)
\]

and the local SNR in this case for the \( k \)th cisoid is defined as

\[
SNR_k = 10 \log_{10} \left( \frac{|A_k|^2}{V(e^{j\omega_k})} \right) \quad (33)
\]

where \( V(e^{j\omega_k}) \) denotes the spectral density function of the AR process at the frequency of the \( k \)th cisoid. While the cisoidal frequencies were \( f_1 = 0.05 \) and \( f_2 = 0.0656 \) in the white noise case, they were \( f_1 = 0.2106 \) and \( f_2 = 0.2262 \) in the colored noise case. The initial phases \( \phi_1 \) and \( \phi_2 \) were zero in both cases.

A. Visual Results

We first consider the frequency spacing improvement simulations in which case the input time series were generated using equation (30) with no noise. Hence, the individual SNRs are infinite so that the only improvement can be due to frequency spacing. In figure 4, we show that two subband lattice implementation improves the frequency spacing between closely spaced cisoids, and that this improvement is even furthered by four subband lattice when compared to fullband lattice implementation.

In the SNR improvement simulations, we again used the input time series generated by (30) with lower SNRs, \( SNR_1 = SNR_2 = -3 \text{ dB} \), so that the SNR improvement can be better displayed. Figure 5 illustrates that two subband lattice implementation is able to resolve cisoids better than fullband lattice implementation, and that four subband implementation can resolve them much better due to SNR improvement effect of subband filtering.
The next property to be considered is the whitening property of subband filtering. In this experiment, we once again generated the input time series using equation (30), however the white noise $u(n)$ was replaced with the colored noise $v(n)$ in equation (32). The local SNRs for the cisoids were $SNR_1 = SNR_2 = 0$ dB in this case. In figure 6, we compare the estimated spectrums based on fullband, two and four subband ARMA lattice filtering. Evidently, the cisoids are better resolved in two and four subband spectrums than fullband spectrum. Also, they are more clearly resolvable in four subbands than in two subbands. Note that in all of the three visual plots, the spectral estimates with the multitaper method have deeper nulls, wider main lobes, and higher sidelobes comparing to the four subband implementation of the proposed method.

B. Statistical Analysis

We investigate the frequency estimation accuracy performance of the proposed method by performing variance and mean bias simulations for SNRs ranging from $-20$ dB to $20$ dB with $1$ dB increments. We assumed that the number of cisoids are known in these simulations, and then compared the variance performance of the proposed method against the Cramer-Rao Bound (CRB) in [17]. The variance and mean bias plots related to these simulations are presented only for the first frequency in order to make it easier for the reader to separate the individual plots.

The variance and mean bias curves of the proposed lattice method when estimating the spectrum of two cisoids in white noise are presented in figure 7 and 8 respectively. It can be observed in figure 7 that subband filtering improvement effects are more pronounced in the low SNR region ($-20$ dB $\leq SNR \leq -10$ dB), and the threshold SNR, below which the estimation accuracy degrades rapidly, is about $-9$ dB. In the higher SNR region ($-10$ dB $< SNR < 10$ dB), the subband estimation variance curves cross over the CRB while saturating earlier (at approximately $SNR = 7$ dB and $SNR =$ 9 dB in two and four subbands respectively ) than the fullband estimation variance curve due to the shorter data lengths in subbands and the quantization effects induced by the sampling of the frequency variable. The saturation point can be moved to higher SNRs and accordingly the estimation accuracy can be improved by using longer data and more zeropadding at the cost of more computation time. In figure 8, we present mean bias curves for fullband as well as subband implementations. Similar to the variance curves, relatively higher mean bias behavior is observed in the low SNR region while lower values are observed after the threshold SNR. Note the poor variance and mean bias performance of the multitaper method in the low SNR region, which is considered because of its wide main lobes and comparatively higher sidelobes.

VII. CONCLUSIONS

A novel lattice method for adaptive ARMA spectrum estimation in subbands with applications to spectrum sensing in cognitive radio systems has been presented. The proposed method relies on a particular lattice filter structure, which is suitable for reprogrammable VLSI implementations in software defined radios, and can be reconfigured as an equalizer during signal reception. In order to convert lattice reflection coefficients to process parameters, the method includes a new Levinson-Durbin type multichannel conversion algorithm specially developed for SPMLSs. These parameters can therefore be utilized in the classification of the spectrum. Coarse and fine spectrum sensing can be accomplished by using different prediction filter configurations. The computational complexity of the proposed lattice method for $ARMA(p, p)$ spectrum estimation compared to the multitaper method ($N = 128,K=3$) is low as long as filter order ($p$) is smaller than 39 in fullband, 27 in two subbands, and 18 in four subbands. The simulation results demonstrate that the performance of the proposed method improve with increasing number of subbands, and that it provides more accurate frequency estimates than the multitaper method particularly in the low SNR region.

REFERENCES

Fig. 1: Block diagram of the adaptive multichannel ARMA prediction filtering in subbands.

Fig. 2: Block diagram of the adaptive multichannel ARMA lattice prediction filtering in subbands.

Fig. 3: Diagram of the four-channel ARMA lattice filter structure for two-subband spectrum estimation.

Fig. 4: Frequency spacing improvement performance.

Fig. 5: SNR improvement performance.

Fig. 6: Whitening effect performance.

Fig. 7: Variance vs. SNR plots.

Fig. 8: Mean bias vs. SNR plots.