Correspondence

A State-Space Approach to QRD-LSL Interpolation and QRD-LSL Smoothing

Jenq-Tay Yuan

Abstract—**Sayed and Kailath demonstrated the feasibility of directly deriving many known adaptive filtering algorithms in square-root forms by a proper reformulation of the original adaptive problem into a state-space form. This work employs this state-space form to develop adaptive interpolation and smoothing algorithms. In particular, a systematic and concise derivation of the *QR-decomposition least-squares lattice (QRD-LSL) interpolation* and *smoothing* algorithms using correspondences between Kalman filtering and LSL adaptive filtering is given.**

Index Terms—**Interpolation, Kalman filtering, order-recursive adaptive filters, smoothing, state-space models.**

I. Introduction

The Kalman filter [2] provides the linear minimum mean-squared estimator of the state vector $\boldsymbol{x}(k)$, given the observation vectors (or data vectors) $\boldsymbol{y}(1)$ through $\boldsymbol{y}(k)$. However, many applications require the use of the best estimator of the state vector $\boldsymbol{x}(j)$ at some time j, where $j < k$. This is commonly known as the *smoothing* problem [3]. Similarly, linear *interpolation* is the mathematical process of estimating an unknown data sample based on a weighted sum of the surrounding data samples. Naturally, smoothing filters and interpolation filters outperform their filtering and prediction counterparts, respectively, owing to the fact that the former two filters consider the additional "subspace of future data" [4]–[6].

Sayed and Kailath [1] demonstrated that the Kalman filter provides a general framework for the derivation of several different variants of the recursive least-squares (RLS) algorithm including RLS, LSL, and QRD-LSL [4], [7]. However, their approach is concerned only with causal filters. Smoothing filters and interpolation filters are both "noncausal" in the sense that a linear combination of the present, past, and *future* observations can be used to estimate the present signal sample. Consequently, a suitable delay must be introduced for physical realizability. This work comments on the order-recursive QRD-LSL interpolation and QRD-LSL smoothing algorithms developed in [5] and [6], respectively, and shows that both algorithms can be developed by the general framework provided by the Kalman filter theory but in a more systematical and more concise manner relative to the results in [5] and [6]. This development can be accomplished by constructing one-to-one correspondences between Kalman variables and LSL variables, followed by presenting the QRD-LSL interpolation and QRD-LSL smoothing algorithms in a square-root form by translating the square-root information filtering algorithm into the corresponding prearray-to-postarray transformation.

Manuscript received February 21, 2001; revised August 15, 2001. This work was supported by the National Science Council, Taiwan, R.O.C., under Contract, NSC 89-2213-E-030-010. The associate editor coordinating the review of this paper and approving it for publication was Prof. José Carlos M. Bermudez.

The author is with the Department of Electronic Engineering, Fu Jen Catholic University, Taipei, Taiwan, R.O.C. (e-mail: yuan@ee.fju.edu.tw).

Publisher Item Identifier S 1053-587X(01)09598-8.

II. Square-Root Information Filters for the Unforced Dynamical Model

Sayed and Kailath [1], [4] described the special unforced dynamical model that plays a crucial role in formulating a general framework for deriving the RLS family of adaptive filtering algorithms:

$$\boldsymbol{x}(n+1) = \lambda^{-1/2}\boldsymbol{x}(n) \quad (1)$$

$$y(n) = \boldsymbol{u}^H(n)\boldsymbol{x}(n) + v(n) \quad (2)$$

where

- λ positive real scalar;
- $\boldsymbol{x}(n)$ $M \times 1$ state vector;
- $\boldsymbol{u}(n)$ $M \times 1$ input vector;
- $v(n)$ zero-mean white noise process with unit variance;
- $y(n)$ scalar observation.

The *square-root information filters* that propagate the square root $\boldsymbol{K}^{-1/2}(n)$ rather than $\boldsymbol{K}(n)$ itself can be characterized by [1], [4, pp. 594]

$$\begin{bmatrix} \lambda^{1/2}\boldsymbol{K}^{-H/2}(n-1) & \lambda^{1/2}\boldsymbol{u}(n) \\ \hat{\boldsymbol{x}}^H\left(n|Y_{n-1}\right)\boldsymbol{K}^{-H/2}(n-1) & y^*(n) \\ \mathbf{0}^T & 1 \end{bmatrix}\Theta(n) = \begin{bmatrix} \boldsymbol{K}^{-H/2}(n) & \mathbf{0} \\ \hat{\boldsymbol{x}}^H\left(n+1|Y_n\right)\boldsymbol{K}^{-H/2}(n) & r^{-1/2}(n)\alpha^*(n) \\ \lambda^{1/2}\boldsymbol{u}^H(n)\boldsymbol{K}^{1/2}(n) & r^{-1/2}(n) \end{bmatrix} \quad (3)$$

where

$$\boldsymbol{K}(n-1) = E\left\{[\boldsymbol{x}(n-1) - \hat{\boldsymbol{x}}(n-1|Y_{n-1})] \times [\boldsymbol{x}(n-1) - \hat{\boldsymbol{x}}(n-1|Y_{n-1})]^T\right\}$$

is the filtered state-error correlation matrix in which $\hat{\boldsymbol{x}}(n-1|Y_{n-1})$ is the minimum mean-square estimate of the state vector $\boldsymbol{x}(n-1)$, given the data $y(1), y(2), \ldots, y(n-1)$; $\alpha(n)$ is the innovation associated with $y(n)$ and is defined as

$$\alpha(n) = y(n) - \boldsymbol{u}^H(n)\hat{\boldsymbol{x}}(n|Y_{n-1}) \quad (4)$$

and $r^{-1}(n)$ is the conversion factor that converts the Kalman filter innovation $\alpha(n)$ to the Kalman filter estimation error defined by

$$e(n) = y(n) - \boldsymbol{u}^H(n)\hat{\boldsymbol{x}}(n|Y_n) \quad (5)$$

Notably, the matrix $\Theta(n)$ in (3) is an orthogonal rotation that produces a block zero entry in the top block row of the postarray. The following section demonstrates that the square-root information filters described above also provide a general framework for the derivation of both QRD-LSL interpolation and QRD-LSL smoothing algorithms.

TABLE I
SUMMARY OF ONE-TO-ONE CORRESPONDENCES BETWEEN KALMAN VARIABLES AND LSL VARIABLES IN STAGE $(p, f+1)$ AND STAGE $(p+1, f)$ OF THE QRD-LSL INTERPOLATION FILTER

Kalman Variable	LSL Variable			
	Interpolation $(p,f)\to(p,f+1)$		Interpolation $(p,f)\to(p+1,f)$	
	Intermediate Forward Predictor	Interpolator	Intermediate Backward Predictor	Interpolator
y(n)	$\lambda^{-n/2}\varepsilon^{F^*}_{q+1}(n,n-f-1)$	$\lambda^{-n/2}\varepsilon^{I^*}_{p,f}(n-f-1)$	$\lambda^{-n/2}\varepsilon^{B^*}_{q+1}(n,n-f)$	$\lambda^{-n/2}\varepsilon^{I^*}_{p,f}(n-f)$
$\boldsymbol{u}^H(n)$	$\varepsilon^{I^*}_{p,f}(n-f-1)$	$\varepsilon^{F^*}_{q+1}(n,n-f-1)$	$\varepsilon^{I^*}_{p,f}(n-f)$	$\varepsilon^{B^*}_{q+1}(n,n-f)$
$\hat{\boldsymbol{x}}(n\mid Y_{n-1})$	$\lambda^{-n/2}1^{F}_{q+1}(n-1)$	$\lambda^{-n/2}k^{F}_{p,f+1}(n-1)$	$\lambda^{-n/2}1^{B}_{q+1}(n-1)$	$\lambda^{-n/2}k^{B}_{p+1,f}(n-1)$
$\boldsymbol{K}$(n-1)	$\lambda^{-1}I^{-1}_{p,f}(n-f-2)$	$\lambda^{-1}F^{-1}_{q+1}(n-1,n-f-2)$	$\lambda^{-1}I^{-1}_{p,f}(n-f-1)$	$\lambda^{-1}B^{-1}_{q+1}(n-1,n-f-1)$
α(n)	$\gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2}\eta^{F^*}_{q+1}(n)$	$\gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2}\xi^{I^*}_{p,f+1}(n-f-1)$	$\gamma^{1/2}_{p,f}(n)\lambda^{-n/2}\eta^{B^*}_{q+1}(n)$	$\gamma^{1/2}_{p,f}(n)\lambda^{-n/2}\xi^{I^*}_{p+1,f}(n-f)$
r(n)	$\dfrac{\gamma_{p,f}(n-1)}{\gamma'_{p,f}(n-1)}$	$\dfrac{\gamma_{p,f}(n-1)}{\gamma_{p,f+1}(n-1)}$	$\dfrac{\gamma_{p,f}(n)}{\gamma'_{p,f}(n)}$	$\dfrac{\gamma_{p,f}(n)}{\gamma_{p+1,f}(n)}$

III. STATE-SPACE APPROACH TO THE QRD-LSL INTERPOLATION ALGORITHM

In a (p,f)th-order linear interpolation, we linearly estimate the present input data sample $y(i)$ from its p past and f future neighboring data samples, viz.,

$$\hat{y}_{p,f}(i) = -\sum_{\substack{k=-p\\k\neq 0}}^{f} b^*_{(p,f),k}(n-f)y(i+k) \qquad 1-f \le i \le n-f \tag{6}$$

where $b_{(p,f),k}(n-f)$ is the interpolation coefficient at time $n-f$, which remains fixed during the observation interval $1-f \le i \le n-f$. The length of the signal n is variable. The order $q = p+f$. Using (6), the (p,f)th-order *a posteriori* interpolation error at each time unit can be written as

$$e^I_{p,f}(i) = y(i) - \hat{y}_{p,f}(i) = y(i) + \sum_{\substack{k=-p\\k\neq 0}}^{f} b^*_{(p,f),k}(n-f)y(i+k) \qquad 1-f \le i \le n-f. \tag{7}$$

Herein, any qth-order interpolation filter operating on the present data sample as well as p past and f future data samples to produce the (p,f)th-order interpolation error at its output is referred to as a (p,f)th-order *interpolation filter* (or *interpolator*), where $q = p+f$ is assumed implicitly. Order-recursive QRD-LSL interpolation filters requiring only $O(q)$ operations have been developed in [5] by employing a modified version of linear forward and backward predictions, which are referred to as the *intermediate forward and backward predictions*. Herein, we formulate the QRD-LSL interpolation algorithm developed in [5] within the framework provided by the Kalman filter theory. The following two cases are considered since order-updated recursion for the interpolation error can be obtained by increasing either f or p by one.

A. Additional Future Data Sample Is Used: $f \to f+1$

1) Array for Adaptive Intermediate Forward Predictor: This work attempts to present the intermediate forward prediction part of the QRD-LSL interpolation algorithm in an array form using (3) to (5) and the correspondences between the Kalman variables and LSL variables listed in Table I. Notably, the following development of the array form for adaptive intermediate forward (backward) predictor as well as those for adaptive interpolator and adaptive smoother (see Table IV) follows [4, pp. 655–665] in a similar manner. Table II describes all the LSL variables whose order is indicated in the subscript. The one-to-one correspondences between the Kalman variables and the LSL variables shown in the second column of Table I are obtained as follows. The underlying state-space representation for the $(q+1)$th-order least-squares lattice intermediate forward prediction is given by $x(n+1) = \lambda^{-1/2}x(n)$ and $y(n) = \varepsilon^{I^*}_{p,f}(n-f-1)x(n) + v(n)$ using (1) and (2) and the relation shown in (9) of [5], where $x(n)$ is the state-variable, and the observation $y(n)$ is defined by $y(n) = \lambda^{-n/2}\varepsilon^{F^*}_{q+1} f x(n, n-f-1)$ (see [4, pp. 585, 657, 658]). Accordingly, the first four lines of correspondences between the Kalman and LSL variables for intermediate forward prediction listed in Table I can be obtained. The remaining two lines of correspondences can be verified as follows by using (4) and (5). With the aid of the first four lines of correspondences just obtained, the Kalman filter innovation $\alpha(n)$ in (4) is defined by

$$\begin{aligned}\alpha(n) =& \gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2}\\ &\times\left[\eta^{F^*}_{q+1}(n,n-f-1) - 1^F_{q+1}(n-1)\xi^{I^*}_{p,f}(n-f-1)\right]\\ =&\gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2}\eta^{F^*}_{q+1}(n)\end{aligned} \tag{8}$$

where the conversion factor $\gamma^{1/2}_{p,f}(n-1)$ can be expressed as

$$\begin{aligned}\gamma^{1/2}_{p,f}(n-1) =& \frac{\varepsilon^I_{p,f}(n-f-1)}{\xi^I_{p,f}(n-f-1)}\\ =&\frac{e^I_{p,f}(n-f-1)}{\varepsilon^I_{p,f}(n-f-1)}\\ =&\frac{\varepsilon^F_{q+1}(n,n-f-1)}{\eta^F_{q+1}(n,n-f-1)}\\ =&\frac{e^F_{q+1}(n,n-f-1)}{\varepsilon^F_{q+1}(n,n-f-1)}\end{aligned} \tag{9}$$

TABLE II
DESCRIPTION OF LSL VARIABLES

$\eta^F_{q+1}(n)$, $\eta^B_{q+1}(n)$: A priori forward, backward prediction errors

$e^F_{q+1}(n)$, $e^B_{q+1}(n)$: A posteriori forward, backward prediction errors

$\varepsilon^F_{q+1}(n)$, $\varepsilon^B_{q+1}(n)$: Angle-normalized forward, backward prediction errors

$\varepsilon^F_{q+1}(n,n-f-1)$, $\varepsilon^B_{q+1}(n,n-f)$: Angle-normalized intermediate forward, backward prediction errors

$e^F_{q+1}(n,n-f-1)$, $e^B_{q+1}(n,n-f)$: A posteriori intermediate forward, backward prediction errors

$\varepsilon^I_{p,f}(n-f)$, $\varepsilon_{p,f}(n-f)$: Angle-normalized interpolation, smoothing errors

$\xi^I_{p,f}(n-f)$, $e^I_{p,f}(n-f)$: A priori, posteriori interpolation errors

$\xi_{p,f}(n-f)$, $e_{p,f}(n-f)$: A priori, posteriori smoothing errors

$F_{q+1}(n)$, $B_{q+1}(n)$: Sum of weighted angle-normalized forward, backward prediction error squares

$F_{q+1}(n,n-f-1)$, $B_{q+1}(n-1,n-f-1)$: Sum of weighted angle-normalized intermediate forward, backward prediction error squares

$I_{p,f}(n-f-1)$: Sum of weighted angle-normalized interpolation error squares

TABLE III
SUMMARY OF ONE-TO-ONE CORRESPONDENCES BETWEEN KALMAN VARIABLES AND LSL VARIABLES OF THE QRD-LSL SMOOTHING FILTER

Kalman Variable	LSL Variable	
	Smoothing $(p,f)\to(p+1,f)$	Smoothing $(p,f)\to(p,f+1)$
$y(n)$	$\lambda^{-n/2}\varepsilon^*_{p,f}(n-f)$	$\lambda^{-n/2}\varepsilon^*_{p,f}(n-f-1)$
$\boldsymbol{u}^H(n)$	$\varepsilon^{B^*}_{q+1}(n)$	$\varepsilon^{F^*}_{q+1}(n)$
$\hat{\boldsymbol{x}}(n\vert Y_{n-1})$	$\lambda^{-n/2}k^B_{q+1}(n-1)$	$\lambda^{-n/2}k^F_{q+1}(n-1)$
$\boldsymbol{K}(n-1)$	$\lambda^{-1}B^{-1}_{q+1}(n-1)$	$\lambda^{-1}F^{-1}_{q+1}(n-1)$
$\alpha(n)$	$\gamma'^{1/2}_{p,f}(n)\lambda^{-n/2}\xi^*_{p+1,f}(n-f)$	$\gamma'^{1/2}_{p,f}(n-1)\lambda^{-n/2}\xi^*_{p,f+1}(n-f-1)$
$r(n)$	$\dfrac{\gamma'_{p,f}(n)}{\gamma'_{p+1,f}(n)}$	$\dfrac{\gamma'_{p,f}(n-1)}{\gamma'_{p,f+1}(n-1)}$

With the aid of the same correspondences, the Kalman filter estimation error in (5) is defined by

$$\begin{aligned} e(n) =&\gamma_{p,f}^{-1/2}(n-1)\lambda^{-n/2} \\ &\times\left[e^{F^*}_{q+1}(n,n-f-1)-1^F_{q+1}(n)e^{I^*}_{p,f}(n-f-1)\right] \\ =&\gamma_{p,f}^{-1/2}(n-1)\lambda^{-n/2}e^{F^*}_{q+1}(n). \end{aligned} \tag{10}$$

The conversion factor $r^{-1}(n)$ that converts the innovation $\alpha(n)$ to the filtered estimation error $e(n)$ is given by [4, pp. 660]

$$\begin{aligned} r^{-1}(n) =&\frac{e(n)}{\alpha(n)} \\ =&\frac{\gamma_{p,f}^{-1/2}(n-1)e^{F^*}_{q+1}(n)}{\gamma_{p,f}^{1/2}(n-1)\eta^{F^*}_{q+1}(n)} \\ =&\frac{\gamma'_{p,f}(n-1)}{\gamma_{p,f}(n-1)}=c^2_{I,q}(n-1) \end{aligned} \tag{11}$$

where $\gamma'_{p,f}(n-1)=e^F_{q+1}(n)/\eta^F_{q+1}(n)$ [4, pp. 655]

TABLE IV
SUMMARY OF THE QRD-LSL INTERPOLATION AND QRD-LSL SMOOTHING ALGORITHMS

(1) As one additional future data sample is used, $f \rightarrow f+1$

A. Adaptive Intermediate Forward Predictor

$$\begin{bmatrix} \lambda^{1/2} I^{1/2}_{p,f}(n-f-2) & \varepsilon^{I}_{p,f}(n-f-1) \\ \lambda^{1/2} \Delta^{F^*}_{q+1}(n-1) & \varepsilon^{F}_{q+1}(n,n-f-1) \\ 0 & \gamma^{1/2}_{p,f}(n-1) \end{bmatrix} \Theta^{F}_{I,q}(n-1) = \begin{bmatrix} I^{1/2}_{p,f}(n-f-1) & 0 \\ \Delta^{F^*}_{q+1}(n) & \varepsilon^{F}_{q+1}(n) \\ e^{I^*}_{p,f}(n-f-1) I^{-1/2}_{p,f}(n-f-1) & \gamma'^{1/2}_{p,f}(n-1) \end{bmatrix} \quad (12)$$

B. Adaptive Interpolator

$$\begin{bmatrix} \lambda^{1/2} F^{1/2}_{q+1}(n-1,n-f-2) & \varepsilon^{F}_{q+1}(n,n-f-1) \\ \lambda^{1/2} \rho^{F^*}_{p,f+1}(n-1) & \varepsilon^{I}_{p,f}(n-f-1) \\ 0 & \gamma^{1/2}_{p,f}(n-1) \end{bmatrix} \Theta'_{f,q+1}(n) = \begin{bmatrix} F^{1/2}_{q+1}(n,n-f-1) & 0 \\ \rho^{F^*}_{p,f+1}(n) & \varepsilon^{I}_{p,f+1}(n-f-1) \\ e^{F^*}_{q+1}(n,n-f-1) F^{-1/2}_{q+1}(n,n-f-1) & \gamma^{1/2}_{p,f+1}(n-1) \end{bmatrix} \quad (16)$$

(2) As one additional past data sample is used, $p \rightarrow p+1$

A. Adaptive Intermediate Backward Predictor

$$\begin{bmatrix} \lambda^{1/2} I^{1/2}_{p,f}(n-f-1) & \varepsilon^{I}_{p,f}(n-f) \\ \lambda^{1/2} \Delta^{B^*}_{q+1}(n-1) & \varepsilon^{B}_{q+1}(n,n-f) \\ 0 & \gamma^{1/2}_{p,f}(n) \end{bmatrix} \Theta^{B}_{I,q}(n) = \begin{bmatrix} I^{1/2}_{p,f}(n-f) & 0 \\ \Delta^{B^*}_{q+1}(n) & \varepsilon^{B}_{q+1}(n) \\ e^{I^*}_{p,f}(n-f) I^{-1/2}_{p,f}(n-f) & \gamma'^{1/2}_{p,f}(n) \end{bmatrix} \quad (17)$$

B. Adaptive Interpolator

$$\begin{bmatrix} \lambda^{1/2} B^{1/2}_{q+1}(n-1,n-f-1) & \varepsilon^{B}_{q+1}(n,n-f) \\ \lambda^{1/2} \rho^{B^*}_{p+1,f}(n-1) & \varepsilon^{I}_{p,f}(n-f) \\ 0 & \gamma^{1/2}_{p,f}(n) \end{bmatrix} \Theta'_{b,q+1}(n) = \begin{bmatrix} B^{1/2}_{q+1}(n,n-f) & 0 \\ \rho^{B^*}_{p+1,f}(n) & \varepsilon^{I}_{p+1,f}(n-f) \\ e^{B^*}_{q+1}(n,n-f) B^{-1/2}_{q+1}(n,n-f) & \gamma^{1/2}_{p+1,f}(n) \end{bmatrix} \quad (18)$$

(3) *Adaptive Smoother*

A. As one additional past observation is used, $p \rightarrow p+1$

$$\begin{bmatrix} \lambda^{1/2} B^{1/2}_{q+1}(n-1) & \varepsilon^{B}_{q+1}(n) \\ \lambda^{1/2} \rho^{B^*}_{q+1}(n-1) & \varepsilon_{p,f}(n-f) \\ 0 & \gamma'^{1/2}_{p,f}(n) \end{bmatrix} \Theta_{b,q+1}(n) = \begin{bmatrix} B^{1/2}_{q+1}(n) & 0 \\ \rho^{B^*}_{q+1}(n) & \varepsilon_{p+1,f}(n-f) \\ e^{B^*}_{q+1}(n) B^{-1/2}_{q+1}(n) & \gamma'^{1/2}_{p+1,f}(n) \end{bmatrix} \quad (19)$$

B. As one additional future observation is used, $f \rightarrow f+1$

$$\begin{bmatrix} \lambda^{1/2} F^{1/2}_{q+1}(n-1) & \varepsilon^{F}_{q+1}(n) \\ \lambda^{1/2} \rho^{F^*}_{q+1}(n-1) & \varepsilon_{p,f}(n-f-1) \\ 0 & \gamma'^{1/2}_{p,f}(n-1) \end{bmatrix} \Theta_{f,q+1}(n) = \begin{bmatrix} F^{1/2}_{q+1}(n) & 0 \\ \rho^{F^*}_{q+1}(n) & \varepsilon_{p,f+1}(n-f-1) \\ e^{F^*}_{q+1}(n) F^{-1/2}_{q+1}(n) & \gamma'^{1/2}_{p,f+1}(n-1) \end{bmatrix} \quad (20)$$

Using (3) and (9) and referring to the second column of Table I, the array for the adaptive intermediate forward predictor of order $q+1$ is presented as

$$\begin{bmatrix} \lambda^{1/2} I^{1/2}_{p,f}(n-f-2) & \varepsilon^{I}_{p,f}(n-f-1) \\ \lambda^{1/2} \Delta^{F^*}_{q+1}(n-1) & \varepsilon^{F}_{q+1}(n,n-f-1) \\ 0 & \gamma^{1/2}_{p,f}(n-1) \end{bmatrix} \Theta^{F}_{I,q}(n-1)$$

$$= \begin{bmatrix} I^{1/2}_{p,f}(n-f-1) & 0 \\ \Delta^{F^*}_{q+1}(n) & \varepsilon^{F}_{q+1}(n) \\ e^{I^*}_{p,f}(n-f-1) I^{-1/2}_{p,f}(n-f-1) & \gamma'^{1/2}_{p,f}(n-1) \end{bmatrix} \quad (12)$$

where $\Delta^{F^*}_{q+1}(n) \underline{\underline{\Delta}} 1^{F^*}_{q+1}(n) I^{1/2}_{p,f}(n-f-1)$ is the intermediate forward prediction auxiliary parameter, and the 2×2 matrix $\Theta^{F}_{I,q}(n-1) = \begin{bmatrix} c_{I,q}(n-1) & -s_{I,q}(n-1) \\ s^*_{I,q}(n-1) & c_{I,q}(n-1) \end{bmatrix}$ is an orthogonal rotation designed to annihilate the prearray entry $\varepsilon^{I}_{p,f}(n-f-1)$ that is already computed from the previous interpolation lattice stage. Owing to this annihilation, the angle-normalized forward prediction error $\varepsilon^{F}_{q+1}(n)$, which is directly accessible from a QRD-LSL predictor that can be embedded into a QRD-LSL interpolation filter [5], appears in the postarray of (12). The angle-normalized intermediate forward prediction error $\varepsilon^{F}_{q+1}(n, n-f-1)$ can thus be computed using (12) and will be used later to compute the order-updated interpolation error [see (16)]. Notably, this angle-normalized intermediate forward

$$\begin{bmatrix} \lambda^{1/2}F^{1/2}_{q+1}(n-1,n-f-2) & \varepsilon^{F}_{q+1}(n,n-f-1) \\ \lambda^{1/2}\rho^{F^*}_{p,f+1}(n-1) & \varepsilon^{I}_{p,f}(n-f-1) \\ 0 & \gamma^{1/2}_{p,f}(n-1) \end{bmatrix}\Theta'_{f,q+1}(n) = \begin{bmatrix} F^{1/2}_{q+1}(n,n-f-1) & 0 \\ \rho^{F^*}_{p,f+1}(n) & \varepsilon^{I}_{p,f+1}(n-f-1) \\ e^{F^*}_{q+1}(n,n-f-1)F^{-1/2}_{q+1}(n,n-f-1) & \gamma^{1/2}_{p,f+1}(n-1) \end{bmatrix} \tag{16}$$

prediction error, along with other delayed intermediate forward and backward prediction errors of lower orders, form an *orthogonal basis* so that an *order-recursive* QRD-LSL interpolation filter can be realized.

2) Array for Adaptive Interpolator: We now seek to present the interpolation part of the QRD-LSL interpolation algorithm in an array form with the aid of the third column of Table I. The first four lines of correspondences between the Kalman and LSL variables for interpolation listed in Table I follow directly from the state space characterization of the (p,f)th-order interpolation that may be described by $x(n+1)=\lambda^{-1/2}x(n)$ and $y(n)=\varepsilon^{F^*}_{q+1}(n,n-f-1)x(n)+v(n)$ [using (1) and (2) and the relation shown in [5, Eq. (7)]], where $y(n)=\lambda^{-n/2}\varepsilon^{I^*}_{p,f}(n-f-1)$. The remaining two lines of correspondences pertaining to interpolation can be obtained from (4), (5), and (9) in a manner similar to that described in the intermediate forward prediction part. The Kalman filter innovation $\alpha(n)$ in (4) is thus defined by

$$\begin{aligned}\alpha(n) =&\gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2} \\ &\times\Big[\xi^{I^*}_{p,f}(n-f-1) \\ &\quad -k^{F}_{p,f+1}(n-1)\eta^{F^*}_{q+1}(n,n-f-1)\Big] \\ =&\gamma^{1/2}_{p,f}(n-1)\lambda^{-n/2}\xi^{I^*}_{p,f+1}(n-f-1).\end{aligned} \tag{13}$$

The Kalman filter estimation error $e(n)$ in (5) is defined by

$$\begin{aligned}e(n) =&\gamma^{-1/2}_{p,f}(n-1)\lambda^{-n/2} \\ &\times\Big[e^{I^*}_{p,f}(n-f-1)-k^{F}_{p,f+1}(n)e^{F^*}_{q+1}(n,n-f-1)\Big] \\ =&\gamma^{-1/2}_{p,f}(n-1)\lambda^{-n/2}e^{I^*}_{p,f+1}(n-f-1).\end{aligned} \tag{14}$$

The conversion factor $r^{-1}(n)$ is given by

$$\begin{aligned}r^{-1}(n) =&\frac{e(n)}{\alpha(n)} \\ =&\frac{\gamma^{-1/2}_{p,f}(n-1)e^{I^*}_{p,f+1}(n-f-1)}{\gamma^{1/2}_{p,f}(n-1)\xi^{I^*}_{p,f+1}(n-f-1)} \\ =&\frac{\gamma_{p,f+1}(n-1)}{\gamma_{p,f}(n-1)}=\left(c'_{f,q+1}(n)\right)^2.\end{aligned} \tag{15}$$

Using (3) and (9) and referring to the third column of Table I, the array for the adaptive interpolator as the interpolation order is presented as in (16), shown at the top of the page, where $\rho^{F^*}_{p,f+1}(n)\underline{\underline{\Delta}}k^{F^*}_{p,f+1}(n)F^{1/2}_{q+1}(n,n-f-1)$ is the interpolation auxiliary parameter, and the 2×2 matrix $\Theta'_{f,q+1}(n)=\begin{bmatrix} c'_{f,q+1}(n) & -s'_{f,q+1}(n) \\ s^{*}_{f,q+1}(n) & c'_{f,q+1}(n)\end{bmatrix}$ is an orthogonal rotation designed to annihilate the prearray entry $\varepsilon^{F}_{q+1}(n,n-f-1)$ computed from (12). Owing to the orthogonal rotation, the order-updated recursion for the interpolation error $\varepsilon^{I}_{p,f+1}(n-f-1)$ can be computed as an additional future data sample is considered.

B. Additional Past Data Sample Is Used: $p\rightarrow p+1$

The array for the intermediate backward predictor and the array for the interpolator can be similarly derived from the fourth and fifth columns of Table I, respectively. The results are shown as (17) and (18) in Table IV.

The *QRD-LSL smoothing algorithm* in array form can be developed in a manner similar to that described above by using the correspondences between the Kalman variables and LSL variables shown in the second and third columns of Table III (see also [8, (21) and (26)]). Summary of this algorithm is presented as (19) and (20) in Table IV.

IV. Conclusions

This work has demonstrated that the link between Kalman filter theory and adaptive filter theory, as originally developed by Sayed and Kailath [1], can be further extended to adaptive interpolation and smoothing. Accordingly, the already established results in least-squares lattice adaptive interpolation and smoothing can be connected with the general framework provided by the Kalman filter theory.

References

[1] A. H. Sayed and T. Kailath, “A state-space approach to adaptive RLS filtering,” *IEEE Signal Processing Mag.*, vol. 11, pp. 18–60, July 1994.

[2] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Trans. ASME, J. Basic Eng.*, vol. 82, pp. 34–45, Mar. 1960.

[3] H. W. Sorenson, *Kalman Filtering: Theory and Application*, H. W. Sorenson, Ed. New York: IEEE, 1985.

[4] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[5] J.-T. Yuan, “QR-decomposition-based least-squares lattice interpolators,” *IEEE Trans. Signal Processing*, vol. 48, pp. 70–79, Jan. 2000.

[6] ——, “A modified QRD for smoothing and a QRD-LSL smoothing algorithm,” *IEEE Trans. Signal Processing*, vol. 47, pp. 1414–1420, May 1999.

[7] I. K. Proudler, J. G. McWhirter, and T. J. Shepherd, “Computationally efficient QR decomposition approach to least squares adaptive filtering,” *Proc. Inst. Elect. Eng. F*, vol. 138, no. 4, pp. 341–353, Aug. 1991.

[8] J.-T. Yuan, “Least squares order-recursive lattice smoothers,” *IEEE Trans. Signal Processing*, vol. 43, pp. 1058–1067, May 1995.