

# Nonparametric Fixed-Interval Smoothing with Vector Splines

Jeffrey A. Fessler

**Abstract**—Spline smoothing has become a popular method for nonparametric exploration and estimation of scalar-valued functions, but its generalizations to vector-valued functions have been underutilized. This paper presents a computationally efficient algorithm for nonparametric smoothing of vector signals with general measurement covariances. This new algorithm provides an alternative to the prevalent “optimal” smoothing algorithms that hinge on (possibly inaccurate) parametric state-space models. We develop and compare automatic procedures that use the measurements to determine how much to smooth; this adaptation allows the data to “speak for itself” without imposing a Gauss–Markov model structure. We present a nonparametric approach to covariance estimation for the case of i.i.d. measurement errors. Monte Carlo simulations demonstrate the performance of the algorithm.

## I. INTRODUCTION

THE goal of fixed-interval smoothing is to estimate a smooth function from a finite number of noisy measurements. We consider here the linear measurement model:

$$y_n = H_n x_n + \varepsilon_n, \quad n = 1, \dots, N \quad (1)$$

where

$$\varepsilon_n, y_n \in \mathbb{R}^L, \quad x_n \in \mathbb{R}^M, \quad \text{and} \quad H_n \in \mathbb{R}^{L \times M}.$$

We assume that the additive measurement error  $\varepsilon_n$  is normally distributed with mean zero and (positive definite) covariance matrix  $\Sigma_n$ , and that the errors are independent between samples. The states  $\{x_n\}$  are (possibly unequally spaced) samples of a process  $g(t_n)$ :

$$x_n = [g_1(t_n), \dots, g_M(t_n)]' \triangleq g(t_n), \quad t_n < t_{n+1} \quad \forall n$$

where “'” denotes matrix transposition. The goal of smoothing is to estimate  $g$  (and/or its derivatives) from the measurements  $\{y_n\}_{n=1}^N$ .

One justifies smoothing by *a priori* knowledge that the component functions of  $g(t)$  vary slowly in some sense. The smoothness of  $g$  is frequently quantified by assuming that the states  $\{x_n\}$  adhere to a parametric Gauss–Markov discrete-time state-space model:

$$x_{n+1} = A_n x_n + B_n u_n, \\ u_n \sim N(0, Q_n), \quad x_0 \sim N(\mu_0, \Pi_0). \quad (2)$$

Manuscript received March 11, 1989; revised May 10, 1990. This work was supported in part by a National Science Foundation Graduate Fellowship, by the National Institute of Health under Contract NO1-HV-38045 and Grants R01-HL-39045, HL-39297, HL-34962, and HL-39478, by the National Science Foundation under Contract ECS-8213959, and by the GE Medical Systems Group under Contract 22-84.

The author was with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA 94305. He is now with the Division of Nuclear Medicine, University of Michigan Medical Center, Ann Arbor, MI 48109.

IEEE Log Number 9042250.

Using such models, one can derive optimal smoothing algorithms [1] that provide minimum mean-square error estimates of the states. However, in many applications the parameters (state evolution matrices and covariances) of the state evolution model (2) are unknown, and they must be estimated from the measurements or from a training set [2]. This estimation may result in an inaccurate parametric model.

Rather than impose a possibly inaccurate parametric model, we would sometimes prefer to “let the data speak for itself,” particularly for off-line data exploration. This motivates nonparametric approaches to smoothing [3].

Nonparametric spline smoothing has proven to be successful at estimating scalar-valued functions from noisy data. Therefore, it is not surprising that the nonparametric approach has also been applied in some situations similar to the vector measurement model (1). In this paper, we derive a computationally efficient algorithm for nonparametric smoothing of vector measurements, allowing for general measurement covariances  $\Sigma_n$ . This has also been considered by Miller and Wegman [4], but their algorithm requires that the covariance matrices be simultaneously diagonalizable. For independent, identically distributed (i.i.d.) measurement errors, we recommend the transformation approach of [4], as it requires fewer computations than the algorithm presented below. Wegman [5], Woltring [6], [7], and Sidhu and Weinert [8] all discuss approaches that effectively assume that the covariance matrices are diagonal. Note that the approach of Sidhu and Weinert [8] does allow for a more general measurement model than (1). In the special case of diagonal covariance matrices, the vector-spline smoothing algorithm reduces to repeated applications of the scalar algorithm. However, one can take advantage of any known similarity between the component functions when choosing the smoothing parameters [6], [7]. Nondiagonal, non-i.i.d. measurement covariances arise, for example, in the medical imaging problems of interest to the authors [9]–[11].

Although, as observed by Silverman [12], “nonparametric regression is not as widely known or adopted as perhaps it should be,” spline smoothing concepts have previously had several other generalizations that we list for didactical reasons: estimating a function’s derivatives [12], [13], estimating branching curves [14], smoothing multivariate functions (scalar valued functions of several variables) [3], [15], and estimating curves with discontinuities [16]. Source code for spline smoothing is available from the GCV and TOMS directories of NETLIB [17].<sup>1</sup>

This paper is organized as follows. In Section II, we review the derivation of the cubic-spline-based algorithm for smooth-

<sup>1</sup>An e-mail message to netlib@research.att.com containing the request “send index” or “send index from gcv” will generate a reply containing instructions.

ing scalar measurements, following the approach and notation of Reinsch [18], [19]. In Section III, we present the new algorithm for smoothing vector measurements. In Section IV, we consider methods that use the measurements to automatically choose the parameters that control the degree of smoothing. In Section V, we describe a nonparametric method for measurement error covariance estimation. In Section VI, we outline the algorithm implementation and discuss computational requirements. In Section VII, we compare these methods using simulated measurements.

## II. SPLINE SMOOTHING OF SCALAR MEASUREMENTS

### A. The Problem

Assume the scalar measurements  $y_n$  satisfy the model

$$y_n = g(t_n) + w_n \epsilon_n, \quad n = 1, \dots, N$$

$$\epsilon_n \sim N(0, \sigma^2), \quad E\{\epsilon_n \epsilon_m\} = 0 \quad \text{if } n \neq m$$

where  $t_1 < \dots < t_N$ . The weights  $w_n$  are assumed known, but the variance  $\sigma^2$  may be unknown. Estimation of  $g$  by smoothing the  $y_n$ 's always involves a tradeoff between fit to the data and smoothness of the estimated function  $\hat{g}$ . For normally distributed measurement errors, the natural measure of fit to the data is the weighted residual sum of squares

$$\text{RSS}(g) \triangleq \sum_{n=1}^N \left( \frac{y_n - g(t_n)}{w_n} \right)^2.$$

Spline smoothing is based on the following nonparametric measure of the roughness (lack of smoothness) of  $g$ :

$$R_k(g) \triangleq \int_{t_1}^{t_N} (g^{(k)}(t))^2 dt.$$

General differential operators have also been considered, e.g., [4], [8]. For simplicity, we consider here only the case  $k = 2$ , though the algorithm derived below is fully generalizable.  $R_2(g)$  is related to the curvature of  $g$ , so it more heavily weights functions that are very wiggly. This measure also has the desirable property that  $R_2(g) = 0$  if and only if  $g$  is linear (for  $k = 2$ ).

We would like to simultaneously minimize  $\text{RSS}(g)$  and  $R_2(g)$ , but these are conflicting goals in general.<sup>2</sup> The standard nonparametric solution is to use the curve that minimizes a weighted combination of the two:

$$\hat{g}_\alpha \triangleq \arg \min_g \left[ \sum_{n=1}^N \left( \frac{y_n - g(t_n)}{w_n} \right)^2 + \alpha \int (g(t))^2 dt \right]. \quad (3)$$

The smoothing parameter  $\alpha$  controls the tradeoff between fit to the data and smoothness. As  $\alpha \rightarrow 0$ ,  $\hat{g}_\alpha$  approaches the cubic-spline interpolant of the measurements, and as  $\alpha \rightarrow \infty$ ,  $\hat{g}_\alpha$  approaches the linear regression of the measurements. Automatic selection of  $\alpha$  will be discussed in Section IV.  $R_2(g)$  acts as a "roughness penalty" [12] that prevents excessive local variation in the curve  $g$ . This idea is related to the regularization methods of computer vision [20].

### B. The Solution

A consequence of the Euler equation corresponding to the variational problem (3) is that the minimizing function  $\hat{g}_\alpha$  is a

<sup>2</sup>The solution to the problem of minimizing  $R_2(g)$  subject to the constraint  $g(t_n) = y_n \forall n$  is cubic-spline interpolation. Interpolation is useful only if the measurements are noiseless [18].

cubic spline [18]. A function  $g$  is a cubic spline with knots  $\{t_n\}_{n=1}^N$  if and only if there exist polynomial coefficients  $\{a_n, b_n, c_n, d_n\}_{n=0}^N$  such that

$$g(t) = a_n + b_n(t - t_n) + \frac{c_n}{2}(t - t_n)^2 + \frac{d_n}{6}(t - t_n)^3,$$

$$t \in [t_n, t_{n+1}] \quad (4)$$

and

$$g(t), \dot{g}(t), \ddot{g}(t) \quad \text{are all continuous.} \quad (5)$$

( $t_0$  is any number less than  $t_1$ , and  $t_{N+1}$  is any number greater than  $t_N$ .)

The continuity conditions (5) impose a system of equations on the polynomial coefficients. These knot dependent equations are [18]

$$b_n = -\frac{1}{6}h_n(c_{n+1} + 2c_n) + h_n^{-1}(a_{n+1} - a_n),$$

$$n = 0, \dots, N - 1$$

$$d_n = h_n^{-1}(c_{n+1} - c_n), \quad n = 0, \dots, N - 1,$$

$$b_N = \frac{1}{6}h_{N-1}c_{N-1} + (a_N - a_{N-1})/h_{N-1}$$

$$h_n^{-1}a_n - (h_n^{-1} + h_{n+1}^{-1})a_{n+1} + h_{n+1}^{-1}a_{n+2}$$

$$= \frac{1}{6}h_n c_n + \frac{1}{3}(h_n + h_{n+1})c_{n+1} + \frac{1}{6}h_{n+1}c_{n+1},$$

$$n = 0, \dots, N - 2 \quad (6)$$

where  $h_n = t_n - t_{n-1}$ . There are a total of  $4(N + 1)$  unknowns, so by adding 4 boundary conditions to the above  $3N$  equations, we can express all of the coefficients in terms of  $(a_1, \dots, a_N)$ . Table I presents two possible boundary conditions. We restrict our attention here to natural cubic-splines by imposing the boundary condition that  $\hat{g}_\alpha(t)$  is linear for  $t > t_N$  and  $t < t_1$ . Other boundary conditions, e.g., periodic and complete splines, are possible, which may be important if derivatives of  $g$  are to be estimated [6], [13].

For natural cubic-splines, the most important constraints are summarized by the following matrix relation:

$$Qa = Tc \quad (7)$$

where  $a = (a_1, \dots, a_N)'$ , and  $c = (c_2, \dots, c_{N-1})'$ .  $Q$  and  $T$  are  $N \times (N - 2)$  and  $(N - 2) \times (N - 2)$  band matrices respectively,

$$Q_{ji} \triangleq \begin{cases} h_i^{-1}, & j = i \\ -(h_i^{-1} + h_{i+1}^{-1}), & j = i + 1 \\ h_{i+1}^{-1}, & j = i + 2 \\ 0, & \text{otherwise} \end{cases}$$

$$T_{ij} \triangleq \begin{cases} \frac{1}{6}h_i, & j = i - 1 \\ \frac{1}{3}(h_i + h_{i+1}), & j = i \\ \frac{1}{6}h_{i+1}, & j = i + 1 \\ 0, & \text{otherwise.} \end{cases}$$

(The  $B$ -spline version of  $Q$  and  $T$  is known to result in a more numerically stable computational algorithm [6]; we present this version for simplicity.)

TABLE I  
BOUNDARY CONDITIONS FOR SPLINE SMOOTHING

Left Boundary	Right Boundary
Natural Splines	
$c_0 = 0$ $d_0 = 0$	$c_N = 0$ $d_N = 0$
Not-a-Knot Splines	
$a_0 = a_1 + b_1 h_1 + \frac{1}{2} c_1 h_1^2 + \frac{1}{6} d_1 h_1^3$	$a_N = a_{N-1} + b_{N-1} h_{N-1} + \frac{1}{2} c_{N-1} h_{N-1}^2 + \frac{1}{6} d_{N-1} h_{N-1}^3$
$d_0 = d_1$	$d_N = d_{N-1}$

Let  $\mathbf{y} = (y_1, \dots, y_N)'$  and  $\mathbf{W} = \text{diag}(w_1, \dots, w_N)$ . If  $g$  is a natural cubic spline with expansion (4), then it was shown in [21] that

$$R_2(g) = \int (\dot{g}(t))^2 dt = \mathbf{c}' \mathbf{T} \mathbf{c} = \mathbf{a}' \mathbf{Q} \mathbf{T}^{-1} \mathbf{Q}' \mathbf{a}$$

and

$$\text{RSS}(g) = (\mathbf{y} - \mathbf{a})' \mathbf{W}^{-2} (\mathbf{y} - \mathbf{a}). \quad (8)$$

Therefore the coefficients of the smoothing spline  $\hat{g}_\alpha$  minimize the quadratic:

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} [(\mathbf{y} - \mathbf{a})' \mathbf{W}^{-2} (\mathbf{y} - \mathbf{a}) + \alpha \mathbf{a}' \mathbf{Q} \mathbf{T}^{-1} \mathbf{Q}' \mathbf{a}].$$

One can find the solution to this minimization by solving the following system of equations for  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{c}}$  [18]:

$$\begin{aligned} \mathbf{Q}' \mathbf{y} &= (\mathbf{T}/\alpha + \mathbf{Q}' \mathbf{W}^2 \mathbf{Q})(\hat{\mathbf{c}} \alpha) \\ \hat{\mathbf{a}} &= \mathbf{y} - \mathbf{W}^2 \mathbf{Q}(\hat{\mathbf{c}} \alpha). \end{aligned} \quad (9)$$

Since  $\mathbf{T}$  and  $\mathbf{Q}$  are band matrices, we can solve (9) in  $O(N)$  operations [22]. These band matrices will also be important to the efficiency of the algorithm for smoothing vector measurements. Having computed  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{c}}$ , we can compute  $\hat{\mathbf{b}}$  and  $\hat{\mathbf{d}}$  from (6), thereby obtaining the piecewise-cubic expansion of  $\hat{g}_\alpha$ . In many cases, only  $\hat{\mathbf{a}}$  is needed since  $\hat{g}_\alpha(t_n) = \hat{a}_n$ .

### III. SPLINE SMOOTHING OF VECTOR MEASUREMENTS

#### A. The Problem

We now generalize the results of the previous section by considering the vector measurement model<sup>3</sup>:

$$\begin{aligned} \mathbf{y}_n &= \mathbf{g}(t_n) + \boldsymbol{\varepsilon}_n, \quad n = 1, \dots, N \\ \mathbf{g}(t_n), \boldsymbol{\varepsilon}_n, \mathbf{y}_n &\in \mathcal{R}^M, \quad \boldsymbol{\varepsilon}_n \sim N(\mathbf{0}, \boldsymbol{\Sigma}_n), \\ E\{\boldsymbol{\varepsilon}_n \boldsymbol{\varepsilon}_m'\} &= \mathbf{0}, \quad n \neq m. \end{aligned} \quad (10)$$

<sup>3</sup>The objection could be raised that model (10) is not as general as model (1), which contains the additional  $\mathbf{H}_n$  term. However, if the measurement matrices  $\mathbf{H}_n$  are all of rank  $M$ , then multiplying both sides of (1) by  $(\mathbf{H}_n' \boldsymbol{\Sigma}_n^{-1} \mathbf{H}_n)^{-1} \mathbf{H}_n' \boldsymbol{\Sigma}_n^{-1}$  transforms (1) into (10). In general, the measurement matrices may not all be of rank  $M$ . If they are not, then even optimal Kalman filters, derived from the state-space model (2), will only be effective if the pairs  $(\mathbf{H}_n, \mathbf{A}_n)$  satisfy the technical condition of *stochastic observability* [23]. This condition is usually satisfied because of the presence of delay or difference terms in  $\mathbf{x}_n$ . Any such (application dependent) *a priori* information should be incorporated into the nonparametric paradigm presented here.

Although we assume the error covariances  $\boldsymbol{\Sigma}_n$  are known for the derivation below, they can be estimated (Section V) if the errors are identically distributed. The goal is to estimate  $\mathbf{g}$  from the measurements  $\{\mathbf{y}_n\}$ .

Again we must compromise between fit to the data and smoothness of the estimated functions. Assuming the errors are normally distributed, the natural measure of fit to the data is

$$\text{RSS}(\mathbf{g}) = \sum_{n=1}^N (\mathbf{y}_n - \mathbf{g}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \mathbf{g}(t_n)).$$

Although we assume that the component functions  $g_m(t)$  of  $\mathbf{g}(t)$  are smooth, they may have different degrees of smoothness, different scales, and different marginal measurement-error variances. Hence,  $M$  smoothing parameters,  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_M)$ , are required to formulate the problem. However, if a group of the component functions are known to have similar properties, then we equate the corresponding smoothing parameters. The multidimensional generalization of (3) is then

$$\begin{aligned} \hat{\mathbf{g}}_\alpha &= \arg \min_{\mathbf{g}} \left[ \sum_{n=1}^N (\mathbf{y}_n - \mathbf{g}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \mathbf{g}(t_n)) \right. \\ &\quad \left. + \sum_{m=1}^M \alpha_m \int (\dot{g}_m(t))^2 dt \right]. \end{aligned} \quad (11)$$

Before presenting the solution to this general minimization problem, we note two special cases. If the covariances  $\boldsymbol{\Sigma}_n$  are diagonal, then (11) separates into  $M$  independent terms of the form (3), so  $\hat{\mathbf{g}}_\alpha$  can be computed by  $M$  evaluations of (9). Similarly, if the covariances are simultaneously diagonalizable, then a transformation of (11) yields a separable expression [4].

#### B. The Solution

Again, by the Euler equations for (11), the solution  $\hat{\mathbf{g}}_\alpha$  is a vector spline with component functions  $\{\hat{g}_{m,\alpha}\}_{m=1}^M$  that are each natural cubic splines. Their piecewise polynomial expansions (4) have coefficients denoted  $\hat{\mathbf{a}}^{(m)}$ ,  $\hat{\mathbf{b}}^{(m)}$ ,  $\hat{\mathbf{c}}^{(m)}$ , and  $\hat{\mathbf{d}}^{(m)}$ , where

$$\begin{aligned} \hat{\mathbf{a}}^{(m)} &= (\hat{a}_1^{(m)}, \dots, \hat{a}_N^{(m)})' \\ \hat{\mathbf{c}}^{(m)} &= (\hat{c}_2^{(m)}, \dots, \hat{c}_{N-1}^{(m)})', \quad \hat{c}_1^{(m)} = \hat{c}_N^{(m)} = 0 \end{aligned}$$

and the coefficients  $\hat{\mathbf{b}}^{(m)}$  and  $\hat{\mathbf{d}}^{(m)}$  satisfy (6).

As shown in Appendix A,  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{c}}$  are computed in  $O(M^3 N)$  operations by solving the following banded equations (cf. (9)):

$$\begin{aligned} (\mathbf{Q}' \otimes \mathbf{I}_M) \mathbf{y} &= ((\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M) \boldsymbol{\Sigma} (\mathbf{Q} \otimes \mathbf{I}_M)) \hat{\mathbf{c}}_\alpha \\ \hat{\mathbf{a}} &= \mathbf{y} - \boldsymbol{\Sigma} (\mathbf{Q} \otimes \mathbf{I}_M) \hat{\mathbf{c}}_\alpha \end{aligned} \quad (12)$$

where  $\otimes$  denotes matrix tensor product,  $\mathbf{I}_M$  is an  $M \times M$  identity matrix,

$$\begin{aligned} \mathbf{a} &= (a_1^{(1)}, \dots, a_1^{(M)}, \dots, a_N^{(1)}, \dots, a_N^{(M)})' \\ \mathbf{c} &= (c_2^{(1)}, \dots, c_2^{(M)}, \dots, c_{N-1}^{(1)}, \dots, c_{N-1}^{(M)})' \\ \mathbf{c}_\alpha &= (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\boldsymbol{\alpha})) \mathbf{c} \\ \mathbf{D}(\boldsymbol{\alpha}) &= \text{diag}(\alpha_1, \dots, \alpha_M) \\ \mathbf{y} &= (y_1', \dots, y_N')' \end{aligned}$$

and

$$\Sigma = \text{diag}(\Sigma_1, \dots, \Sigma_N). \quad (13)$$

$\Sigma$  is the  $NM \times NM$  block diagonal covariance matrix of  $\mathbf{y}$ .

The minimization (11) has resulted in a linear relationship,  $\hat{\mathbf{a}} = \mathbf{A}(\boldsymbol{\alpha})\mathbf{y}$ , between the measurements and the estimates, where from (12)

$$\begin{aligned} \mathbf{A}(\boldsymbol{\alpha}) &= \mathbf{I}_{NM} - \Sigma(\mathbf{Q} \otimes \mathbf{I}_M)((\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) \\ &\quad + (\mathbf{Q}' \otimes \mathbf{I}_M)\Sigma(\mathbf{Q} \otimes \mathbf{I}_M))^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M). \end{aligned} \quad (14)$$

In statistics,  $\mathbf{A}(\boldsymbol{\alpha})$  is called the hat or influence matrix and will be used in the next section for automatic selection of  $\boldsymbol{\alpha}$ .

#### IV. CHOOSING THE SMOOTHING PARAMETERS

If the smoothing parameter  $\boldsymbol{\alpha}$  is too large or too small, then the measurements will be oversmoothed or undersmoothed, respectively. In the scalar case, Reinsch [18] suggested choosing  $\alpha$  so that  $\text{RSS}(\hat{g}_\alpha) \approx N\sigma^2$ . However, Craven and Wahba [24] showed that this led to consistent oversmoothing. Ideally, we would like to choose the smoothing parameters to minimize the mean-square error:

$$\begin{aligned} \text{MSE}(\boldsymbol{\alpha}) &\triangleq \frac{1}{N} \sum_{n=1}^N \|\hat{g}_\alpha(t_n) - g(t_n)\|^2 \\ \boldsymbol{\alpha}_{\text{MSE}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{MSE}(\boldsymbol{\alpha}). \end{aligned} \quad (15)$$

In practice, this minimization is impossible since  $\mathbf{g}$  is unknown. For the scalar measurement problem, several methods have been suggested for estimating  $\boldsymbol{\alpha}_{\text{MSE}}$  from the data [3], including two due to Akaike [25]. We present below three of these methods, each generalized to apply to our vector measurement problem. They are compared empirically in Section VII. Note that for small samples one may prefer to use robust variants of these estimators [25].

The estimators discussed below all depend on the central bands of the influence matrix  $\mathbf{A}(\boldsymbol{\alpha})$ . Hutchinson and deHoog [26], [27] have presented algorithms for computing these bands in  $O(M^3N)$  operations. Their algorithm is directly applicable to the vector measurement problem, so we do not present it here.

##### A. Unbiased Risk

In the scalar case with known error standard deviation, Craven and Wahba [24] have suggested using the value of the smoothing parameter that minimizes an unbiased estimator of the expected mean-square error (risk). This idea can be directly extended to the vector measurement case, and one can easily show that

$$\begin{aligned} \text{UR}(\boldsymbol{\alpha}) &\triangleq \frac{1}{N} \|(I - \mathbf{A}(\boldsymbol{\alpha}))\mathbf{y}\|^2 - \frac{2}{N} \text{tr}(\Sigma(I - \mathbf{A}(\boldsymbol{\alpha}))) \\ &\quad + \frac{1}{N} \text{tr}(\Sigma) \end{aligned}$$

is an unbiased estimator of  $E\{\text{MSE}(\boldsymbol{\alpha})\}$ . The unbiased risk estimate of  $\boldsymbol{\alpha}_{\text{MSE}}$  is thus

$$\boldsymbol{\alpha}_{\text{UR}} \triangleq \arg \min_{\boldsymbol{\alpha}} \text{UR}(\boldsymbol{\alpha}).$$

The estimators  $\boldsymbol{\alpha}_{\text{CV}}$  and  $\boldsymbol{\alpha}_{\text{GCV}}$  discussed below have been more popular than  $\boldsymbol{\alpha}_{\text{UR}}$  in the scalar case, perhaps because the latter

depends on the (often unknown) error variance  $\sigma^2$ . For the vector measurement problem, all three estimates depend on the covariance matrices  $\{\Sigma_n\}$ .

##### B. Cross Validation

Wahba and Wold [28], [29] have suggested using the smoothing parameter that minimizes the cross-validation (CV) score:

$$\begin{aligned} \text{CV}(\boldsymbol{\alpha}) &\triangleq \frac{1}{N} \sum_{n=1}^N (y_n - \hat{g}_{\boldsymbol{\alpha}, -n}(t_n))' \Sigma_n^{-1} (y_n - \hat{g}_{\boldsymbol{\alpha}, -n}(t_n)) \\ \boldsymbol{\alpha}_{\text{CV}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{CV}(\boldsymbol{\alpha}). \end{aligned} \quad (16)$$

$\hat{g}_{\boldsymbol{\alpha}, -n}$  is the solution to the smoothing problem (11) with  $N - 1$  data points, posed without the data pair  $(t_n, y_n)$ , i.e.:

$$\begin{aligned} \hat{g}_{\boldsymbol{\alpha}, -i} &\triangleq \arg \min_g \left[ \sum_{n=1, n \neq i}^N (y_n - g(t_n))' \Sigma_n^{-1} (y_n - g(t_n)) \right. \\ &\quad \left. + \sum_{m=1}^M \alpha_m \int (\dot{g}_m(t))^2 dt \right]. \end{aligned}$$

Each data pair is dropped in turn, the smoothed curve  $\hat{g}_{\boldsymbol{\alpha}, -n}$  is estimated, and the predicted value  $\hat{g}_{\boldsymbol{\alpha}, -n}(t_n)$  is compared with the unused measurement. If the CV score is small, then we have chosen the smoothing parameter that makes the estimated curve a good self-predictor.

Although (16) illustrates the idea behind cross validation, it is computationally inefficient. We show in Appendix B that (16) can be rewritten

$$\text{CV}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{n=1}^N \left\| \Sigma_n^{-(1/2)} (\mathbf{I}_M - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}))^{-1} (y_n - \hat{g}_\alpha(t_n)) \right\|^2 \quad (17)$$

where  $\mathbf{A}_{(nn)}(\boldsymbol{\alpha})$  is the  $n$ th  $M \times M$  block diagonal submatrix of the influence matrix (14). By using the Hutchinson and de Hoog algorithm [26], (17) is computed in only  $O(M^3N)$  operations.

##### C. Generalized Cross Validation

Craven and Wahba [24] have also suggested using the smoothing parameter that minimizes the generalized cross-validation (GCV) score, whose vector spline generation is:

$$\begin{aligned} \text{GCV}(\boldsymbol{\alpha}) &\triangleq \frac{\frac{1}{N} \text{RSS}(\hat{g}_\alpha)}{\left(\frac{1}{N} \text{tr}(I - \mathbf{A}(\boldsymbol{\alpha}))\right)^2} = \frac{\frac{1}{N} (\mathbf{y} - \hat{g}_\alpha)' \Sigma^{-1} (\mathbf{y} - \hat{g}_\alpha)}{\left(\frac{1}{N} \text{tr}(I - \mathbf{A}(\boldsymbol{\alpha}))\right)^2} \\ \boldsymbol{\alpha}_{\text{GCV}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{GCV}(\boldsymbol{\alpha}). \end{aligned}$$

In the scalar case, the GCV score is a weighted version of the CV score that is invariant to rotations of the data when periodic end conditions are imposed [25]. See [3] for discussion of the asymptotic properties of GCV. Again, [26] is used to evaluate  $\text{GCV}(\boldsymbol{\alpha})$  in  $O(M^3N)$  operations.

#### V. ERROR COVARIANCE ESTIMATION

For the scalar measurement case with unknown error variance, Wahba [30] (see also discussion in [31]) proposed the

following estimator:

$$\hat{\delta}^2 = \frac{\|(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))\mathbf{y}\|}{\text{tr}(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))} \quad (18)$$

where the denominator is the natural extension of "degrees of freedom" to nonparametric regression. For the vector measurement case with unknown error covariance, if the measurement errors are identically distributed, that is  $\boldsymbol{\Sigma}_n = \boldsymbol{\Sigma} \mathbf{v}_n$ , then we can generalize this idea to estimate  $\boldsymbol{\Sigma}$  by the following algorithm:

- 1) For each  $m = 1, \dots, M$ , smooth the  $m$ th measurement component  $\{y_{n,m}\}_{n=1}^N$  to compute  $\hat{g}_{m,\alpha_m}$  using the scalar algorithm of Section II, and using, for example, the CV score to choose  $\alpha_m$ .
- 2) Estimate the elements of  $\boldsymbol{\Sigma}$  using the standard correlation estimate:

$$\hat{\Sigma}_{ij} = \frac{\sum_{n=1}^N (y_{n,i} - \hat{g}_{i,\alpha_i}(t_n))(y_{n,j} - \hat{g}_{j,\alpha_j}(t_n))}{\sqrt{\text{tr}(\mathbf{I} - \mathbf{A}(\alpha_i))} \sqrt{\text{tr}(\mathbf{I} - \mathbf{A}(\alpha_j))}}$$

(which simplifies to (18) for the diagonal elements of  $\boldsymbol{\Sigma}$ ). For the non-i.i.d. case, Silverman's iterative reweighting approach may be useful [12].

## VI. ALGORITHM

Table II outlines the organization of the algorithm's implementation (C source code is available as VSPLINE from NETLIB [17]). The first set of computations is independent of  $\boldsymbol{\alpha}$ . The second set computes the smoothed estimates  $\hat{\mathbf{a}}$ , and the third set evaluates the cross-validation score. The computational requirements for this algorithm are of the same order as those for Kalman filter smoothers [32], when  $\boldsymbol{\alpha}$  is known. The second and third set are typically repeated for several values of  $\boldsymbol{\alpha}$  to find the minimum CV score. This search is the computational penalty for our uncertainty about the smoothness of the functions we are estimating. For the i.i.d. case, the transformation method of Miller and Wegman [4] is preferable as it only requires  $O(M^2N)$  computations for the transformations, and  $O(MN)$  computations for smoothing.

## VII. SIMULATION RESULTS

To demonstrate the new smoothing algorithm and to compare the three methods of automatically choosing the smoothing parameter  $\boldsymbol{\alpha}$ , we applied the methods to simulated data. Fig. 1 is a plot of two functions,  $g_1$ , a decaying sinusoid, and  $g_2$ , a hyperbolic tangent, and one realization of their noisy sampled measurements. We generated the measurements by adding pseudorandom Gaussian noise vectors with covariance matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} 2.25 & 2.4 \\ 2.4 & 4 \end{bmatrix}$$

to  $N = 100$  samples of the function drawn uniformly on  $[0, 1]$ .

First we applied the scalar smoothing algorithm to the noisy samples of  $g_1$  shown in Fig. 1. Fig. 2 shows UR( $\alpha$ ), CV( $\alpha$ ), GCV( $\alpha$ ), and MSE( $\alpha$ ) over a range of the smoothing parameter. The minima of the UR, CV, and GCV scores (denoted by the small circles) occur very close to the minimum of the MSE; thus, at least for this scalar example, each of the three methods would select a good smoothing parameter. It is interesting that

TABLE II  
COMPUTATIONAL REQUIREMENTS OF LINEAR SMOOTHING ALGORITHM

Computation	Flops
I. Initialization	
$\mathbf{Q}, \mathbf{T}$	$N(7) - 7$
$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y}$	$N(5M) - 5M$
$\mathbf{B}_0 = (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)$	$N(9M^2 + 6) - 23M^2 - 16$
Subtotal:	$N(9M^2 + 5M + 13) - 23M^2 - 5M - 23$
II. Regression	
$(\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1})$	$N(2M) - 5M$
$\mathbf{B} = (\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) + \mathbf{B}_0$	$N(2M^2) - 4M^2$
$[\mathbf{L}, \mathbf{D}] = \text{Cholesky}(\mathbf{B})$	$N(\frac{3}{2}M^2 - \frac{3}{2}M^2 - M) - 54M^3 + 12M^2 + 2M$
Solve $(\mathbf{LDL}')\hat{\mathbf{e}}_\alpha = (\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y}$	$N(12M^2 - 3M) - 42M^2 + 12M$
$(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{e}}_\alpha$	$N(6M) - 12M$
$\boldsymbol{\Sigma}(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{e}}_\alpha$	$N(2M^2 - M)$
$\hat{\mathbf{a}} = \mathbf{y} - \boldsymbol{\Sigma}(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{e}}_\alpha$	$N(M)$
Subtotal:	$N(\frac{27}{2}M^3 + \frac{29}{2}M^2 + 4M) - 54M^3 - 34M^2 - 3M$
III. Compute CV Score	
$\mathbf{B}^{-1} = \text{Invert}(\mathbf{LDL}') [26]$	$N(18M^3 - 6M^2) - 2M(3M - 1)(12M - 1)$
$\mathbf{F} = (\mathbf{Q} \otimes \mathbf{I}_M)\mathbf{B}^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M)$	$N(\frac{27}{2}M(M + 1)) - \frac{39}{2}M(M + 1)$
$\mathbf{e}_n = \mathbf{y}_n - \hat{\mathbf{g}}_\alpha(t_n)$	$N(M)$
Solve $(\mathbf{F}_{(mm)})\mathbf{f}_n = \boldsymbol{\Sigma}_n^{-1}\mathbf{e}_n$	$N(\frac{1}{2}M^3 + \frac{5}{2}M^2 - M)$
$\text{CV}(\boldsymbol{\alpha}) = 1/N \sum_{n=1}^N \mathbf{f}_n' \boldsymbol{\Sigma}_n^{-1} \mathbf{f}_n$	$N(2M^2 + M - 1)$
Subtotal:	$N(\frac{37}{2}M^3 + 11M^2 + \frac{29}{2}M - 1) - 72M^3 + \frac{21}{2}M^2 - \frac{43}{2}M$

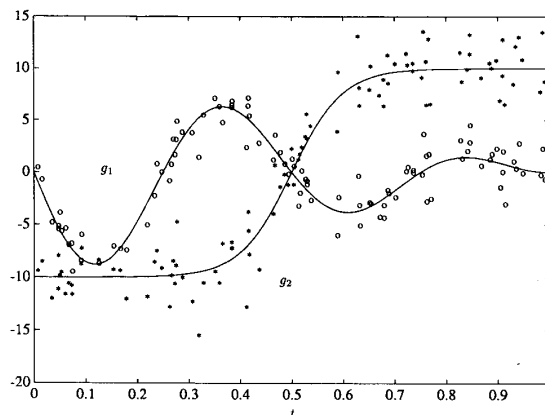


Fig. 1. True curves (-) and noisy measurements (°, \*).

the UR, CV, and GCV scores are all flatter than the MSE near  $\boldsymbol{\alpha}_{\text{MSE}}$ .

To evaluate the three scores in the vector measurement case, we did a Monte Carlo simulation with 400 runs, each with a different measurement noise realization. For each run we computed  $\boldsymbol{\alpha}_{\text{MSE}}$ ,  $\boldsymbol{\alpha}_{\text{UR}}$ ,  $\boldsymbol{\alpha}_{\text{CV}}$ , and  $\boldsymbol{\alpha}_{\text{GCV}}$  using Powell's method for nonlinear minimization as given in [33, p. 315]. Our intent was to compare the estimators' best possible performances, so we initialized the minimization procedure at a value of  $\boldsymbol{\alpha}$  that resulted in low MSE for a few preruns. To compare the estimators, we use their relative efficiencies, defined by

$$\eta_i(\boldsymbol{\alpha}) \triangleq \frac{\text{MSE}(\boldsymbol{\alpha}_{\text{MSE},i})}{\text{MSE}(\boldsymbol{\alpha})}$$

where  $i$  indicates the  $i$ th run. By definition (15),  $\eta_i \in (0, 1]$ .

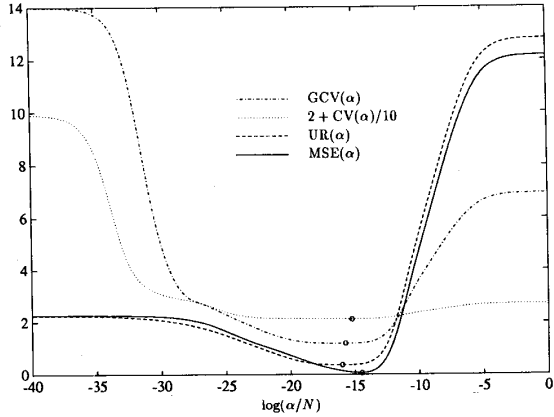


Fig. 2. Comparison of MSE, UR, CV, and GCV for scalar measurements.

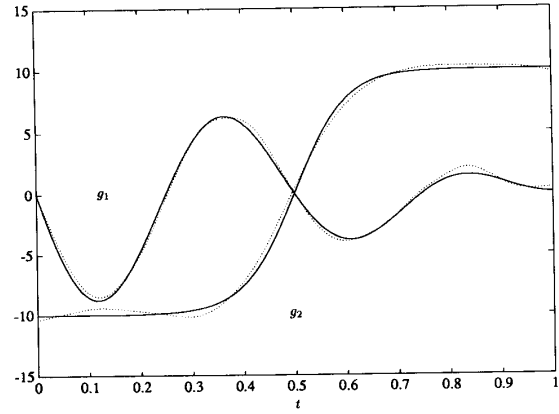


Fig. 3. True curves (solid) and estimated curves (dashed) using CV score.

**TABLE III**  
 RELATIVE EFFICIENCIES OF THE DIFFERENT SMOOTHING APPROACHES

Summary Statistic of $\{\eta_i\}_{i=1}^{400}$	Smoothing Approach					
	None	UR	CV	GCV	Diag ( $\underline{\Sigma}$ )	$\hat{\underline{\Sigma}}$
mean $\bar{\eta}$	0.076	0.812	0.841	0.846	0.754	0.839
5th %-ile	0.034	0.532	0.578	0.605	0.463	0.555
25th %-ile	0.054	0.731	0.762	0.767	0.664	0.757
50th %-ile	0.071	0.836	0.877	0.877	0.777	0.883
75th %-ile	0.092	0.924	0.947	0.945	0.868	0.946
95th %-ile	0.130	0.991	0.992	0.990	0.945	0.993
$\sigma_{\bar{\eta}} = \sigma_{\eta}/\sqrt{400}$	0.002	0.007	0.007	0.006	0.007	0.007

Table III shows summary statistics of the computed relative efficiencies for the 400 runs. Three other cases are included for comparison; ‘‘None’’: no smoothing, ‘‘Diag ( $\underline{\Sigma}$ )’’: smoothing with just the diagonal components of the covariances (with minimum CV score), and ‘‘ $\hat{\underline{\Sigma}}$ ’’: smoothing with the estimated covariance procedure described in Section V (also with minimum CV score).

From the summary statistics for this example, we conclude that the CV and GCV scores perform equally well, and both slightly outperform the UR score. Those three were significantly more efficient than smoothing the components individually, which was expected since the measurement correlation was 0.8. All smoothing approaches decreased the MSE by a factor of approximately 10. It was a pleasant surprise that the performance using the estimated covariance matrix was about as good as the performance using the true covariance. This suggests that the approach described in this paper may be preferable to smoothing the components individually, even when the error covariance is unknown. The off-diagonal elements of the covariance matrices clearly play an important role, even when estimated.

There is still no consensus on the relative theoretical merits of the UR, CV, and GCV scores, even in the scalar case. We have derived and presented the vector generalizations of all three since their performances may be application dependent.

As a representative example, Fig. 3 shows the smoothed estimates (dashed) superimposed on the true curves (solid) for the data shown in Fig. 1, using  $\alpha_{CV}$ . The estimated functions agree well with the true functions, and the overall smoothness is qualitatively similar as well.

## VIII. CONCLUSIONS

We have presented a computationally efficient algorithm for nonparametric fixed-interval smoothing of noisy measurements with arbitrary measurement covariances. The effectiveness of the approach was demonstrated on a numerical example. The approach promises to be an attractive alternative to parametric Kalman smoothing for off-line applications.

Possible extensions of this work would include developing a more robust approach to covariance estimation, and applying Silverman’s iterative reweighting approach [12] for non-i.i.d. covariance estimation. The relationship of nonparametric estimation to state-space methods could also be explored more completely, which could result in a recursive formulation of the solution.

## APPENDIX A

In this Appendix, we derive the solution (12) to the minimization problem (11):

$$\hat{g}_{\alpha} = \arg \min_g \text{RSS}(g) + \sum_{m=1}^M \alpha_m R(g_m).$$

By (13),

$$\text{RSS}(g) = (y - a)' \Sigma^{-1} (y - a)$$

and by (8),

$$\sum_{m=1}^M \alpha_m R(g_m) = (c^{(m)})' T c^{(m)} = c'(T \otimes D(\alpha))c.$$

Since the minimizing component functions are each natural cubic splines, they must each satisfy the constraint  $Q' a^{(m)} = T c^{(m)}$  of (7). These constraints can be aggregated to form the constraint  $(Q' \otimes I_M) a = (T \otimes I_M) c$ . The optimal coefficients thus minimize

$$(y - a)' \Sigma^{-1} (y - a) + c'(T \otimes D(\alpha))c \quad (19)$$

subject to

$$(Q' \otimes I_M) a = (T \otimes I_M) c. \quad (20)$$

Since  $(T \otimes I_M)$  is symmetric and invertible,  $c = (T \otimes I_M)^{-1} (Q' \otimes I_M) a$ , which, substituted into (19) yields

$$\begin{aligned} & (y - a)' \Sigma^{-1} (y - a) + ((T \otimes I_M)^{-1} (Q' \otimes I_M) a)' \\ & \cdot (T \otimes D(\alpha)) (T \otimes I_M)^{-1} (Q' \otimes I_M) a \\ & = y' \Sigma^{-1} y - 2a' \Sigma^{-1} y + a' (\Sigma^{-1} + (Q \otimes D(\alpha)) \\ & \cdot (T \otimes I_M)^{-1} (Q' \otimes I_M)) a. \end{aligned}$$

Minimizing over  $\mathbf{a}$  by setting the partial derivative w.r.t.  $\mathbf{a}$  equal to zero yields

$$\Sigma^{-1}\mathbf{y} = (\Sigma^{-1} + (\mathbf{Q} \otimes \mathbf{D}(\mathbf{a}))(T \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M))\hat{\mathbf{a}}. \quad (21)$$

Solving for  $\hat{\mathbf{a}}$

$$\hat{\mathbf{a}} = (\Sigma^{-1} + \mathbf{S}_\alpha)^{-1}\Sigma^{-1}\mathbf{y} \quad (22)$$

where

$$\begin{aligned} \mathbf{S}_\alpha &\triangleq (\mathbf{Q} \otimes \mathbf{I}_M)(T \otimes \mathbf{D}(\mathbf{a})^{-1})^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M) \\ &= (\mathbf{Q}T^{-1}\mathbf{Q}') \otimes \mathbf{D}(\mathbf{a}). \end{aligned} \quad (23)$$

We could compute  $\hat{\mathbf{a}}$  directly from (22), but a few manipulations [18] yield a more easily evaluated band form. Multiplying both sides of (21) by  $\Sigma$  and using (20), we get

$$\mathbf{y} = \hat{\mathbf{a}} + \Sigma(\mathbf{Q} \otimes \mathbf{I}_M)(\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\mathbf{a}))\hat{\mathbf{c}}.$$

Multiplying both sides by  $(\mathbf{Q}' \otimes \mathbf{I}_M)$  and using (20) yields

$$\begin{aligned} (\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} &= ((T \otimes \mathbf{I}_M) + (\mathbf{Q}' \otimes \mathbf{I}_M)\Sigma(\mathbf{Q} \otimes \mathbf{I}_M) \\ &\quad \cdot (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\mathbf{a})))\hat{\mathbf{c}}. \end{aligned} \quad (24)$$

Symmetric band matrices are the easiest to use, so define  $\mathbf{c}_\alpha = (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\mathbf{a}))\mathbf{c}$ . Combining this definition with (24) yields

$$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} = ((T \otimes \mathbf{D}(\mathbf{a})^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\Sigma(\mathbf{Q} \otimes \mathbf{I}_M))\hat{\mathbf{c}}_\alpha.$$

Thus the minimizing coefficients  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{c}}$  are the solutions to the following system of equations:

$$\begin{aligned} ((T \otimes \mathbf{D}(\mathbf{a})^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\Sigma(\mathbf{Q} \otimes \mathbf{I}_M))\hat{\mathbf{c}}_\alpha &= (\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} \\ \hat{\mathbf{c}} &= (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\mathbf{a})^{-1})\hat{\mathbf{c}}_\alpha \\ \hat{\mathbf{a}} &= \mathbf{y} - \Sigma(\mathbf{Q} \otimes \mathbf{I}_M)\hat{\mathbf{c}}_\alpha. \end{aligned}$$

#### APPENDIX B

In this Appendix, we show that (16) is equivalent to (17). Again we use  $\mathbf{A}_{(nn)}(\mathbf{a})$  to denote the  $n$ th  $M \times M$  central diagonal submatrix of  $\mathbf{A}(\mathbf{a})$ .

The same arguments used in the proof of Lemma 3.1 by Craven and Wahba [24] imply that for all  $n$  and  $i$

$$\hat{\mathbf{g}}_{\alpha, -n}(t_i) = \sum_{k=1, k \neq n}^N \mathbf{A}_{(ik)}(\mathbf{a})\mathbf{y}_k + \mathbf{A}_{(in)}(\mathbf{a})\hat{\mathbf{g}}_{\alpha, -n}(t_n).$$

By the definition of  $\mathbf{A}(\mathbf{a})$

$$\begin{aligned} \mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n) &= \mathbf{y}_n - \sum_{i=1}^N \mathbf{A}_{(ni)}(\mathbf{a})\mathbf{y}_i \\ &= \mathbf{y}_n - \sum_{i=1, i \neq n}^N \mathbf{A}_{(ni)}(\mathbf{a})\mathbf{y}_i - \mathbf{A}_{(nn)}(\mathbf{a})\hat{\mathbf{g}}_{\alpha, -n}(t_n) \\ &\quad - \mathbf{A}_{(nn)}(\mathbf{a})(\mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n)) \\ &= \mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n) - \mathbf{A}_{(nn)}(\mathbf{a})(\mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n)) \\ &= (\mathbf{I} - \mathbf{A}_{(nn)}(\mathbf{a}))(\mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n)). \end{aligned}$$

Therefore

$$\mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n) = (\mathbf{I} - \mathbf{A}_{(nn)}(\mathbf{a}))^{-1}(\mathbf{y}_n - \hat{\mathbf{g}}_{\alpha, -n}(t_n))$$

which can be substituted into (16) to yield (17).

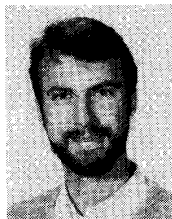
#### ACKNOWLEDGMENT

The author is grateful to Prof. B. Silverman for his helpful suggestions, and to the reviewers for their corrections and references.

#### REFERENCES

- [1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.
- [2] L. Ljung, *System Identification, Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.
- [3] R. L. Eubank, *Spline Smoothing and Nonparametric Regression*. New York: Marcel Dekker, 1988.
- [4] J. J. Miller and E. J. Wegman, "Vector function estimation using splines," *J. Stat. Planning Inference*, vol. 17, pp. 173-180, 1987.
- [5] E. J. Wegman, "Vector splines and the estimation of filter functions," *Technometrics*, vol. 23, pp. 83-89, Feb. 1981.
- [6] H. J. Woltring, "A Fortran package for generalized cross-validated spline smoothing and differentiation," *Adv. Eng. Software*, vol. 8, no. 2, pp. 104-112, 1986.
- [7] H. J. Woltring, A. deLange, J. Kauer, and R. Huiskes, "Instantaneous helical axis estimation via natural cross-validated splines," in *Biomechanics: Basic and Applied Research*, A. R. G. Bergmann and R. Kolbel, Eds. European Society of Biomechanics, Sept. 1986, pp. 121-128.
- [8] G. S. Sidhu and H. L. Weinert, "Vector-valued  $L_g$ -splines, II. Smoothing splines," *J. Math. Anal. Appl.*, vol. 101, pp. 380-396, July 1984.
- [9] J. Fessler and A. Macovski, "Nonparametric tracking of shift and shape functions in medical images," in *Proc. IEEE ASSP Workshop Multidimensional Signal Processing* (Asilomar, CA), Sept. 1989, p. 46.
- [10] J. Fessler, D. Nishimura, and A. Macovski, "Model-based 3-D reconstruction of branching vessels," in *Proc. 11th Ann. Int. Conf. IEEE Eng. Medicine Biol. Soc.* (Seattle, WA), Nov. 1989, pp. 561-562.
- [11] J. Fessler and A. Macovski, "Object-based 3-D reconstruction of arterial trees from a few projections," *IEEE Trans. Med. Imaging*, to be published.
- [12] B. Silverman, "Some aspects of the spline smoothing approach to nonparametric regression curve fitting," *J. Roy. Stat. Soc., ser. B*, vol. 47, no. 1, pp. 1-52, 1985.
- [13] H. J. Woltring, "On optimal smoothing and derivative estimation from noisy displacement data in biomechanics," *Human Movement Sci.*, vol. 4, no. 3, pp. 229-230, 1985.
- [14] B. Silverman and J. Wood, "The nonparametric estimation of branching curves," *J. Amer. Stat. Assoc.*, vol. 82, pp. 551-558, June 1987.
- [15] G. Wahba and J. Wendelberger, "Some new mathematical methods for variational objective analysis using splines and cross validation," *Monthly Weather Rev.*, vol. 108, pp. 1122-1143, 1980.
- [16] D. Lee and T. Pavlidis, "One-dimensional regularization with discontinuities," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 10, pp. 822-829, Nov. 1988.
- [17] J. J. Dongarra and E. Grosse, "Distribution of mathematical software via electronic mail," *Commun. Ass. Comput. Mach.*, vol. 30, pp. 403-407, Oct. 1987.
- [18] C. H. Reinsch, "Smoothing by spline functions," *Numer. Math.*, vol. 10, pp. 177-183, 1967.
- [19] C. H. Reinsch, "Smoothing by spline functions II," *Numer. Math.*, vol. 16, pp. 451-454, 1971.
- [20] M. Bertero, T. Poggio, and V. Torre, "Ill-posed problems in early vision," *Proc. IEEE*, vol. 76, pp. 869-899, Aug. 1988.
- [21] C. de Boor, *A Practical Guide to Splines*. New York: Springer, 1978.
- [22] G. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins University Press, 1983.
- [23] T. Kailath, *Linear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [24] P. Craven and G. Wahba, "Smoothing noisy data with spline functions," *Numer. Math.*, vol. 31, pp. 377-403, 1979.
- [25] T. Robinson and R. Moyeed, "Making robust the cross-validated choice of smoothing parameter in spline smoothing regres-

- sion," *Commun. Statist.—Theory Math.*, vol. 18, no. 2, pp. 523–539, 1989.
- [26] M. Hutchinson and F. de Hoog, "Smoothing noisy data with spline functions," *Numer. Math.*, vol. 47, pp. 99–106, 1985.
- [27] F. de Hoog and M. Hutchinson, "An efficient method for calculating smoothing splines using orthogonal transformations," *Numer. Math.*, vol. 50, pp. 311–319, 1987.
- [28] G. Wahba and S. Wold, "A completely automatic French curve: Fitting spline functions by cross validation," *Commun. Stat.*, vol. 4, no. 1, pp. 1–17, 1975.
- [29] G. Wahba and S. Wold, "Periodic splines for spectral density estimation: The use of cross validation for determining the degree of smoothing," *Commun. Stat.*, vol. 4, no. 2, pp. 125–141, 1975.
- [30] G. Wahba, "Improper priors, spline smoothing, and the problem of guarding against model errors in regression," *J. Roy. Stat. Soc.*, ser. B, vol. 40, no. 3, pp. 364–372, 1978.
- [31] M. J. Buckley, G. K. Eagleson, and B. W. Silverman, "The estimation of residual variance in nonparametric regression," *Biometrika*, vol. 75, no. 2, pp. 189–199, 1988.
- [32] P. Kaminski, A. Bryson, and S. Schmidt, "Discrete square root filtering: A survey of current techniques," *IEEE Trans. Automat. Contr.*, vol. 16, pp. 727–735, Dec. 1971.
- [33] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. Cambridge University Press, 1988.



**Jeffrey A. Fessler** received the B.S.E.E. degree from Purdue University in 1985, the M.S.E.E. degree in 1986, the M.S. degree in statistics in 1989, and the Ph.D. degree in electrical engineering in 1990, all from Stanford University.

From 1985 to 1988 he was a National Science Foundation Graduate Fellow at Stanford. He is currently a postdoctoral fellow in the Department of Nuclear Medicine at the University of Michigan. His research interests include estimation and pattern recognition and their applications in medical imaging and bioengineering.