The Annals of Statistics
1998, Vol. 26, No. 5, 1733–1759

BACKFITTING IN SMOOTHING SPLINE ANOVA

BY ZHEN LUO

Pennsylvania State University

A computational scheme for fitting smoothing spline ANOVA models to large data sets with a (near) tensor product design is proposed. Such data sets are common in spatial-temporal analyses. The proposed scheme uses the backfitting algorithm to take advantage of the tensor product design to save both computational memory and time. Several ways to further speed up the backfitting algorithm, such as collapsing component functions and successive over-relaxation, are discussed. An iterative imputation procedure is used to handle the cases of near tensor product designs. An application to a global historical surface air temperature data set, which motivated this work, is used to illustrate the scheme proposed.

1. Introduction. Smoothing spline ANOVA (SS-ANOVA) is a multivariate function estimation method based on an ANOVA type decomposition of the function to be estimated. It generalizes the decomposition of multiple factor effects in an ordinary ANOVA and the smoothing spline method for univariate function estimation. It has been applied to various data sets, for example, an environmental data set [Gu and Wahba (1993a, b)], and an epidemiological data set [Wahba, Wang, Gu, Klein and Klein (1996)]. In this article, an application to the spatial-temporal analysis of a global historical temperature data set will be discussed.

A major difficulty in using SS-ANOVA is its large computational demand. Gu (1989) carefully implemented a computational scheme for generic smoothing spline estimation problems. The scheme that he used works well on data sets of small to moderate sizes. Since it does not assume any special data structures, it is inevitably slow and memory-demanding when dealing with large data sets.

A subclass of SS-ANOVA models are those without any kinds of interactions, that is, smoothing spline additive models. Buja, Hastie and Tibshirani (1989) studied these models (and other additive models) and used the backfitting algorithm to transform the problem of multivariate function estimation into many univariate function estimation problems. Because of the existence of sparse matrix representations of the univariate smoothing (polynomial) spline estimate [see, for example, O'Sullivan (1985)], this transformation speeds up computation considerably and saves memory as well.

When interactions are nonnegligible, special structures other than that of complete additivity may be used to save computational memory and time. In

Received December 1996; revised December 1997.

AMS 1991 *subject classifications*. Primary 62G07, 65D10, 65F10; secondary 62H11, 65U05, 86A32.

Key words and phrases. Gauss–Seidel algorithm, tensor product design, spatial-temporal analysis, additive model, collapsing, grouping, SOR, global historical temperature data.

this article, we will make use of tensor product designs that exist in many large data sets, especially in spatial-temporal analyses. The key to our algorithm is also backfitting, which enables us to fit a SS-ANOVA model by way of decomposing matrices much smaller than those without the consideration of tensor product designs. In this way, we can significantly reduce the demand for computational memory. However, unlike the situation of additive models, the "correlation" between component functions, which is a major factor in slowing down the backfitting algorithm, is very often too high in SS-ANOVA models. It makes the backfitting algorithm converge very slowly in its straightforward version. Therefore we will also discuss several techniques to speed up the backfitting algorithm.

In practice, tensor product designs very often hold only approximately, that is, there are no observations at some tensor product grid points. An iterative imputation procedure is used to handle this situation.

The motivation for this work came from the spatial-temporal analysis of a global historical surface air temperature data set. The data set has tens of thousands of observations, which makes the currently available methods to fit SS-ANOVA models, such as the one implemented in Rkpack [Gu (1989)], unusable. This data set will be used to illustrate our proposed computational scheme.

The paper is organized as follows. The SS-ANOVA approach to multivariate function estimation is introduced in Section 2. Section 3 describes our general computational strategy using the backfitting algorithm. Several different techniques, such as grouping, collapsing and successive-overrelaxation, used to speed up the backfitting algorithm, are discussed in Section 4. Section 5 describes an iterative imputation procedure handling the situation of a near tensor product design. An application to a global historical temperature data set is discussed in Section 6. Finally, in Section 7 some comparisons of timing are made between Rkpack and several versions of backfitting.

2. Smoothing spline ANOVA. The SS-ANOVA approach to multivariate function estimation is introduced in this section through its application to the spatial-temporal analysis of a global historical temperature data set. A general formulation of this approach may be found in Wahba (1990), Gu and Wahba (1993a, b) or Wahba, Wang, Gu, Klein and Klein (1995).

Consider the problem of estimating the global winter mean surface air temperature field as a function, denoted by f, of year and geographical location based on scattered noisy data:

$$y_i = f(t_i, P_i) + \varepsilon_i, \qquad i = 1, 2, \ldots, n, \tag{1}$$

where $t_i \in \{1, 2, \ldots, n_t\}$ denotes the year of the ith data point and $P_i =$ (latitude,longitude) denotes the location of the ith data point on the sphere $\mathscr{S}$. Here n is the total number of data points, n_t is the total number of years included in the data set. We also denote the total number of distinct locations in the data set by n_P for later use. The ε_i's are assumed as independent noise terms. (We note that in this particular application, ε_i contains not only the

measurement error of the record y_i, but also the local variation that is of much smaller scale than the resolution of current modeling interest.)

The function f needs only a minimal condition, that $f(t, \cdot)$ is integrable over $\mathscr{S}$ for any t, to have a unique decomposition,

$$f(t, P) = d_1 + d_2\phi(t) + g_1(t) + g_2(P) + g_{\phi,2}(P)\phi(t) + g_{12}(t, P), \tag{2}$$

where $\phi(t) := t - (n_t + 1)/2$, g_1, g_2, $g_{\phi,2}$ and g_{12} satisfy the following side conditions:

$$\begin{aligned} &\sum_{t=1}^{n_t} g_1(t) = \sum_{t=1}^{n_t} g_1(t)\phi(t) = 0, \\ &\sum_{t=1}^{n_t} g_{12}(t, P) = \sum_{t=1}^{n_t} g_{12}(t, P)\phi(t) = 0, \\ &\int_{\mathscr{S}} g_2(P)\,dP = \int_{\mathscr{S}} g_{\phi,2}(P)\,dP = \int_{\mathscr{S}} g_{12}(t, P)\,dP = 0 \end{aligned} \tag{3}$$

for any t and P. The uniqueness of this decomposition is easy to check using these side conditions. To see the existence of such a decomposition, first define three averaging operators on function f:

$$\begin{aligned} (\mathscr{E}_t f)(t, P) &:= \frac{\sum_{t=1}^{n_t} f(t, P)}{n_t}, \\ (\mathscr{E}_t' f)(t, P) &:= \frac{\sum_{t=1}^{n_t} f(t, P)\phi(t)}{\sum_{t=1}^{n_t} \phi^2(t)}\phi(t), \\ (\mathscr{E}_P f)(t, P) &:= \frac{\int_{\mathscr{S}} f(t, P)\,dP}{4\pi}. \end{aligned}$$

Let I denote the identity operator, then f can be decomposed as

$$\begin{aligned} f &= \big[\mathscr{E}_t + \mathscr{E}_t' + (I - \mathscr{E}_t - \mathscr{E}_t')\big]\big[\mathscr{E}_P + (I - \mathscr{E}_P)\big]f \\ &= \mathscr{E}_t\mathscr{E}_P f + \mathscr{E}_t'\mathscr{E}_P f + (I - \mathscr{E}_t - \mathscr{E}_t')\mathscr{E}_P f \\ &\quad + \mathscr{E}_t(I - \mathscr{E}_P)f + \mathscr{E}_t'(I - \mathscr{E}_P)f + (I - \mathscr{E}_t - \mathscr{E}_t')(I - \mathscr{E}_P)f, \end{aligned}$$

which gives the six components in (2). Note that averaging over t is defined in a discrete form because "year" is treated as a discrete variable. While we may consider t as a continuous time variable, we will see in Section 4.1 why it is a better idea to treat it as a discrete one.

The decomposition of f generalizes the decomposition of ordinary two-way ANOVA models. Note that a direct generalization would have only four components in (2), and corresponds to the decomposition using only operators $\mathscr{E}_t$ and $\mathscr{E}_P$. Here we may call $d_2\phi(t)$, $g_1(t)$ and $g_2(P)$ main effect terms, and $g_{\phi,2}(P)\phi(t)$ and $g_{12}(t, P)$ interaction terms. In this article, we will also call $d_1 + d_2\phi$ the "parametric component" and the other four terms in (2) "nonparametric components." The components in (2) are of interest to us because they have clearly defined climatological meanings: d_1 is the grand average winter

temperature over both year and location; d_2 is the linear trend (coefficient) of global average winter temperature; $d_1 + d_2\phi + g_1$ is the global average winter temperature history; g_2, $g_{\phi,2}$ and g_{12} represent local adjustments to the global average terms d_1, d_2 and g_1, respectively. For example, $g_{\phi,2}(P)$ is the local adjustment to the grand linear trend (coefficient) d_2 at location P. In other words, $d_2 + g_{\phi,2}(P)$ is the linear trend (coefficient) at location P. A map of $d_2 + g_{\phi,2}(P)$ would show where the warming areas are and where the cooling areas are over the years covered by the data.

To make any inferences about the function f at points other than the data points, some "smoothness" assumptions about f have to be made. First, as usually done in nonparametric function estimation, we will restrict f to a class of functions with some degree of differentiability. Then, we will estimate f by maximizing a penalized least square criterion with appropriately chosen penalty terms on its component functions.

Let $\mathscr{H}^{(t)}$ denote the space of the functions of t that we will restrict our attention to. Since t has only a finite number (n_t) of possible values, we will add no differentiability restrictions; that is, $\mathscr{H}^{(t)}$ is the Euclidean space of a dimension n_t. For the functions of P, we will restrict our attention to a function space defined in Wahba (1981), denoted there by $\mathscr{H}_2(\mathscr{S})$. Here let $\mathscr{H}^{(P)}$ denote it in order to be consistent with the notation $\mathscr{H}^{(t)}$. Roughly speaking, $\mathscr{H}^{(P)}$ consists of functions g defined on the sphere $\mathscr{S}$ that are twice differentiable and that Δg is square integrable. The Laplace–Beltrami operator for the sphere is denoted by Δ, which is a direct analogue of the Laplace operator for a Euclidean space. For a precise definition of $\mathscr{H}^{(P)}$, please see Wahba (1981), page 7. Both $\mathscr{H}^{(t)}$ and $\mathscr{H}^{(P)}$ are reproducing kernel Hilbert spaces (RKHSs) with squared norms,

$$\begin{aligned}\|g\|^2_{\mathscr{H}^{(t)}} := {} & \left[\frac{\sum_{t=1}^{n_t} g(t)}{n_t}\right]^2 + \left[\frac{\sum_{t=1}^{n_t} g(t)\phi(t)}{\sum_{t=1}^{n_t} \phi^2(t)}\right]^2 \\ & + \sum_{t=1}^{n_t-2} \big[g(t+2) - 2g(t+1) + g(t)\big]^2,\end{aligned} \tag{4}$$

$$\|g\|^2_{\mathscr{H}^{(P)}} := \left(\frac{\int_{\mathscr{S}} g(P)\,dP}{4\pi}\right)^2 + \int_{\mathscr{S}} (\Delta g)^2\,dP, \tag{5}$$

respectively. Note that $\sum_{t=1}^{n_t-2}[g(t+2)-2g(t+1)+g(t)]^2$ is a discrete version of integrated squared second derivative. The function $f(t, P)$ in (1) is restricted to the tensor product space of $\mathscr{H}^{(t)}$ and $\mathscr{H}^{(P)}$, denoted by $\mathscr{H}$. Here $\mathscr{H}$ is also a RKHS since the tensor product space of any two RKHSs is a RKHS. The restriction to RKHSs is barely restrictive because the only requirement for a Hilbert function space to be a RKHS is that every evaluation functional is continuous. That is, a small change of the function measured in the function space should induce only a small change of the value of this function evaluated at any fixed point. For details on reproducing kernel Hilbert spaces and their tensor product spaces, please see Aronszajn (1950). Some introductory

material can also be found in Wahba (1990), or Tapia and Thompson [(1978), Appendix I].

Let $[1^{(t)}]$, $[\phi]$ and $\mathscr{H}_a^{(t)}$ denote the three subspaces of $\mathscr{H}^{(t)}$ defined by $\mathscr{E}_t\mathscr{H}^{(t)}$, $\mathscr{E}_t'\mathscr{H}^{(t)}$, and $(I-\mathscr{E}_t-\mathscr{E}_t')\mathscr{H}^{(t)}$, respectively. They are orthogonal to each other with respect to the inner product corresponding to the norm defined in (4). The three terms on the right side of (4) are the squared norms in these three subspaces. We have

$$\mathscr{H}^{(t)} = [1^{(t)}] \oplus [\phi] \oplus \mathscr{H}_a^{(t)}.$$

Let $[1^{(P)}]$ and $\mathscr{H}_a^{(P)}$ denote the two subspaces of $\mathscr{H}^{(P)}$ defined by $\mathscr{E}_P\mathscr{H}^{(P)}$ and $(I-\mathscr{E}_P)\mathscr{H}^{(P)}$, respectively. These two subspaces are also orthogonal with respect to the inner product corresponding to the norm defined in (5). The two terms on the right side of (5) are the squared norms in these two subspaces. We have

$$\mathscr{H}^{(P)} = [1^{(P)}] \oplus \mathscr{H}_a^{(P)}.$$

Now the function space we restrict f to can be written as an orthogonal sum of six subspaces:

$$\begin{aligned}\mathscr{H} &= \mathscr{H}^{(t)} \otimes \mathscr{H}^{(P)} \\ &= \big([1^{(t)}] \otimes [1^{(P)}]\big) \oplus \big([\phi] \otimes [1^{(P)}]\big) \oplus \big(\mathscr{H}_a^{(t)} \otimes [1^{(P)}]\big) \\ &\quad \oplus \big([1^{(t)}] \otimes \mathscr{H}_a^{(P)}\big) \oplus \big([\phi] \otimes \mathscr{H}_a^{(P)}\big) \oplus \big(\mathscr{H}_a^{(t)} \otimes \mathscr{H}_a^{(P)}\big).\end{aligned} \tag{6}$$

These six subspaces correspond to the six components in (2). Let the first two subspaces be combined into a two-dimensional space, denoted by $\mathscr{H}^0$. This is the parametric subspace. The last four subspaces, denoted by $\mathscr{H}^\alpha$ for $\alpha = 1, 2, 3, 4$, respectively, correspond to the four nonparametric components in (2). We note that the subspaces discussed here are all RKHSs.

The smoothing spline (ANOVA) estimate of f is defined as a penalized least square estimate with the four nonparametric components in (2) penalized by their squared norms in the corresponding subspaces, or a maximum penalized likelihood estimate if the noise terms (ε's) in (1) are assumed to be independent and identically distributed Gaussian random variables. We penalize the nonlinear part of a function of t by its squared norm in the subspace $\mathscr{H}_a^{(t)}$ and the nonconstant part of a function of P by its squared norm in the subspace $\mathscr{H}_a^{(P)}$. Specifically, the smoothing spline estimate of f is defined as the minimizer of

$$\sum_{i=1}^{n}\big(y_i - f(t_i, P_i)\big)^2 + \frac{1}{\theta_1}J_1(f) + \frac{1}{\theta_2}J_2(f) + \frac{1}{\theta_3}J_3(f) + \frac{1}{\theta_4}J_4(f) \tag{7}$$

in $\mathscr{H}$ where θ's are positive numbers called smoothing parameters, J's are penalty terms on four nonparametric components of f in (2). The expression $J_1(f) = \sum_{t=1}^{n_t-2}(g_1(t+2) - 2g_1(t+1) + g_1(t))^2$ is a penalty on the component $g_1 = (I-\mathscr{E}_t-\mathscr{E}_t')\mathscr{E}_P f$; $J_2(f) = \int_{\mathscr{S}}(\Delta g_2)^2\,dP$ is a penalty on the component

$g_2 = \mathscr{E}_t(I - \mathscr{E}_P)f$; $J_3(f) = \int_{\mathscr{S}}(\Delta g_{\phi,2})^2\,dP$ is a penalty on the component $g_{\phi,2}\phi = \mathscr{E}'_t(I-\mathscr{E}_P)f$ and $J_4(f)$ is the squared norm of g_{12} in $\mathscr{H}^4 = \mathscr{H}_a^{(t)} \otimes \mathscr{H}_a^{(P)}$, a penalty on the component $g_{12} = (I - \mathscr{E}_t - \mathscr{E}'_t)(I - \mathscr{E}_P)f$.

The choices of J's given here are not the only ones we may make. How to choose J's and θ's is an important issue in smoothing spline methods. Discussions about this issue can be found in many places, for example, Wahba (1990). For now, suppose that they have been chosen appropriately. Let us move our attention to the computational aspect of this procedure.

3. Computing SS-ANOVA estimates using backfitting. Even though the SS-ANOVA estimate of f is defined as the minimizer of (7) in an infinite-dimensional space $\mathscr{H}$, the solution has a finite-dimensional representation [see Wahba (1990), pages 12, 127, 128],

$$f_\theta(t, P) = d_1 + d_2\phi(t) + \sum_{\alpha=1}^{4}\theta_\alpha \sum_{i=1}^{n} c_i R_\alpha(t_i, P_i; t, P), \tag{8}$$

where R_α is the reproducing kernel of RKHS $\mathscr{H}^\alpha$, a known nonnegative definite function, for $\alpha = 1, 2, 3, 4$; $d := (d_1, d_2)^T$ and $c := (c_1, \ldots, c_n)^T$ are the coefficient vectors to be decided.

In general, the reproducing kernel (RK) of a RKHS F of the functions of x is a two-variable function $R(x, x')$ satisfying (i) for every x', $R(\cdot, x')$ belongs to F; (ii) for every x' and every $f \in F$, $f(x') = \langle f(\cdot), R(\cdot, x')\rangle$ where $\langle\cdot,\cdot\rangle$ denotes the inner product of F. Since $R(\cdot, x')$ is in F, using (ii), we get $R(x, x') = \langle R(\cdot, x'), R(\cdot, x)\rangle$. Therefore $R(x, x')$ is symmetric and nonnegative definite. See Aronszajn (1950) for more information on reproducing kernels.

It is not difficult to verify that the RK of space $[1^{(t)}]$ defined in Section 2 is $R(t, t') = 1$, the RK of space $[\phi]$ is $R(t, t') = \phi(t)\phi(t')$ and the RK of space $\mathscr{H}_a^{(t)}$ is $R(t, t') =$ the (t, t') entry of $(L^TL)^\dagger$, where $\dagger$ denotes the Moore–Penrose generalized inverse and L is a $(n_t - 2) \times n_t$ matrix given by

$$\begin{pmatrix} 1 & -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & \cdots & 0 & 0 & 0 \\ & & & \vdots & \vdots & & & \\ 0 & 0 & 0 & 0 & \cdots & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{pmatrix}. \tag{9}$$

From now on, let $R_t(t, t')$ denote the reproducing kernel of $\mathscr{H}_a^{(t)}$. Note that this form of RK is only suitable for the equally-spaced design of variable t.

It is easy to verify that the RK of space $[1^{(P)}]$ is $R(P, P') = 1$. The RK of space $\mathscr{H}_a^{(P)}$, denoted by $R_P(P, P')$, is given in equations (3.3) and (3.4) of

TABLE 1
The reproducing kernels used in the representation (8) *of the SS-ANOVA estimate*

α	$\mathscr{H}^{\alpha}$	R_{α}
1	$\mathscr{H}_a^{(t)} \otimes [1^{(P)}]$	$R_1(t, P; t', P') = R_t(t, t')$
2	$[1^{(t)}] \otimes \mathscr{H}_a^{(P)}$	$R_2(t, p; t', P') = R_P(P, P')$
3	$[\phi] \otimes \mathscr{H}_a^{(P)}$	$R_3(t, P; t', P') = \phi(t)\phi(t')R_P(P, P')$
4	$\mathscr{H}_a^{(t)} \otimes \mathscr{H}_a^{(P)}$	$R_4(t, P; t', P') = R_t(t, t')R_P(P, P')$

Wahba (1981), and reproduced here:

$$R_P(P, P') = \frac{1}{2\pi}\left[\frac{1}{2}q_2(z) - \frac{1}{6}\right],$$

where z is the cosine of the angle between P and P' on the sphere, and

$$q_2(z) = \frac{1}{2}\left\{\ln\left(1 + \sqrt{\frac{2}{1-z}}\right)\left[12\left(\frac{1-z}{2}\right)^2 - 4\left(\frac{1-z}{2}\right)\right] - 12\left(\frac{1-z}{2}\right)^{3/2} + 6\left(\frac{1-z}{2}\right) + 1\right\}.$$

We note that this R_P, strictly speaking, does not correspond exactly to the squared norm $\int_{\mathscr{S}}(\Delta g)^2\, dP$, but a norm topologically equivalent to it. In other words, we are not computing the minimizer of (7), but the minimizer of (7) with slightly changed J's. The reason is purely computational since the R_P corresponding to $\int_{\mathscr{S}}(\Delta g)^2\, dP$ does not have an explicit form and is expensive to compute. On the other hand, as discussed in Stein (1990), this type of change to the penalty terms should not make much difference in the results when sufficient data are available.

The reproducing kernel of the tensor product space of two RKHSs is the product of their individual reproducing kernels [Aronszajn (1950), Theorem I of Section 8]. Therefore the RKs of $\{\mathscr{H}_{\alpha}, \alpha = 1, 2, 3, 4\}$ are the products of the RK's given above and listed in Table 1.

The f_{θ} expressed in the form of (8) has an obvious decomposition corresponding to (2). For example, $g_1(t) = \theta_1 \sum_{i=1}^{n} c_i R_1(t_i, P_i; t, P) = \theta_1 \sum_{i=1}^{n} c_i R_t(t_i, t)$. Plugging the representation (8) into (7) [note that $\| \sum_{i=1}^{n} c_i R_{\alpha}(t_i, P_i; t, P)\|^2 = \sum_{i,\, j} c_i c_j R_{\alpha}(t_i, P_i; t_j, P_j)$ for $\alpha = 1, 2, 3, 4$], we end up with a finite-dimensional quadratic optimization problem

$$\min_{d,\, c}\left\{\|y - Sd - Q_{\theta}c\|^2 + c^T Q_{\theta} c\right\}, \tag{10}$$

where $y = (y_1, \ldots, y_n)^T$, S is a $n \times 2$ matrix with the ith row given by $(1, \phi(t_i))$, Q_θ is a $n \times n$ matrix defined as $\sum_{\alpha=1}^{4} \theta_\alpha Q_\alpha$ and Q_α is a $n \times n$ matrix with its (i, j) entry given by $R_\alpha(t_i, P_i; t_j, P_j)$, for $\alpha = 1, 2, 3, 4$. Since the objective function in (10) is quadratic and nonnegative (Q_θ is nonnegative definite), it must have a minimizer. It is easy to show that even though the minimizer c and d may not be unique, the corresponding f_θ is unique if S is of full rank [see, e.g., Chen, Gu and Wahba (1989), page 517]. That is, the SS-ANOVA estimate f_θ is uniquely defined. Because the decomposition (2) is unique, all the component functions of f_θ are also uniquely defined.

One of the minimizers of (10) is the solution of

$$\begin{aligned} 0 &= S^T c, \\ (Q_\theta + I)c &= (y - Sd), \end{aligned} \tag{11}$$

which is the system of linear equations usually solved for computing smoothing spline estimates [see, e.g., Wahba (1990), pages 12, 13]. When n, the sample size, is not too large, this system can be solved by direct matrix decompositions of S and Q_θ, which is essentially what Gu (1989) did for general SS-ANOVA models. However, this approach has limitations in terms of both computational time and memory requirement. The memory required is $O(n^2)$ and the time is $O(n^3)$.

When the data and model exhibit some special structures, we may be able to find more efficient ways to compute. The following fact will be used later. Let f_0 denote the vector of the values of the parametric component, $d_0 + d_1\phi$, evaluated at the data points, and f_α denote the vector of the values of the αth nonparametric component of f_θ evaluated at the data points. From the representation (8) we know that $f_0 = Sd$, $f_\alpha = \theta_\alpha Q_\alpha c$ for $\alpha = 1, 2, 3, 4$. If we can get f_0 and f_α's, then we can compute d by solving $Sd = f_0$, and c through the second equation of (11), that is,

$$c = y - Sd - Q_\theta c = y - \sum_{\alpha=0}^{4} f_\alpha. \tag{12}$$

With c and d, we can compute the values of f_θ and its components at any points using (8).

A special structure we have here is the tensor product structure of Q_α's when the data have a complete tensor product design, that is, one observation at every tensor product grid point (t_i, P_j) for $i = 1, 2, \ldots, n_t$ and $j = 1, 2, \ldots, n_P$. In practice we will more likely only have a near tensor product design, that is, one observation at almost every point (t_i, P_j). For the moment, however, we will assume that we do have a complete tensor product design and wait till Section 5 to discuss the way to deal with the situations of near tensor product designs. In the cases of complete tensor product designs, the sample size n is $n_t \times n_P$. Assuming that the data is ordered in such a way that all the data of one location is listed before the data of the next location,

the S and Q_α's in (11) have the following forms:

$$\begin{aligned} S &= 1 \otimes \tilde{S}, \\ Q_1 &= 11^T \otimes Q_t, \\ Q_2 &= Q_P \otimes 11^T, \\ Q_3 &= Q_P \otimes \phi\phi^T, \\ Q_4 &= Q_P \otimes Q_t, \end{aligned} \tag{13}$$

where 1 is a vector of ones of appropriate length, $\phi = (\phi(1), \ldots, \phi(n_t))^T$, $\tilde{S}$ is a $n_t \times 2$ matrix with its ith row given by $(1, \phi(i))$, Q_t is an $n_t \times n_t$ matrix with its (i, j) entry given by $R_t(i, j)$ and Q_P is an $n_P \times n_P$ matrix with its (i, j) entry given by $R_P(P_i, P_j)$. Therefore, $Q_t = (L^T L)^\dagger$ and $Q_t 1 = Q_t \phi = 0$. This fact, which results from the equally spaced design for t and the special form of penalty terms involving t in (7), will be used in Section 4.1.

Even though every Q_α has such a tensor product structure, their linear combination Q_θ may not. Therefore, (11) still cannot be solved using this special structure. In order to make use of this structure, we consider another representation of the SS-ANOVA estimate,

$$f_\theta(t, P) = d_0 + d_1\phi(t) + \sum_{\alpha=1}^{4} \theta_\alpha \sum_{i=1}^{n} c_{i,\alpha} R_\alpha\big((t_i, P_i); (t, P)\big), \tag{14}$$

which is more general than the one in (8). With a slightly abused notation, let c_α denote the vector $(c_{1,\alpha}, c_{2,\alpha}, \ldots, c_{n,\alpha})^T$ for $\alpha = 1, 2, 3, 4$. Plugging (14) into (7), we end up with another finite-dimensional quadratic optimization problem,

$$\min_{d, c_1, c_2, c_3, c_4} \left\{ \left\| y - Sd - \sum_{\alpha=1}^{4} \theta_\alpha Q_\alpha c_\alpha \right\|^2 + \sum_{\alpha=1}^{4} \theta_\alpha c_\alpha^T Q_\alpha c_\alpha \right\}. \tag{15}$$

Any solution of this problem should satisfy

$$\begin{aligned} (S^T S)d &= S^T\left(y - \sum_{\alpha=1}^{4} \theta_\alpha Q_\alpha c_\alpha \right), \\ (\theta_\beta Q_\beta + I) Q_\beta c_\beta &= Q_\beta\left(y - Sd - \sum_{\alpha \neq \beta} \theta_\alpha Q_\alpha c_\alpha \right) \quad \text{for } \beta = 1, 2, 3, 4. \end{aligned} \tag{16}$$

Again, it is easy to show that even though the solution c_α's and d may not be unique, their corresponding f_θ, the SS-ANOVA estimate and all its component functions are uniquely defined if S is of full rank. With the representation (14), the components of f_θ evaluated at the data points can be written as $f_0 = Sd$, $f_\alpha = \theta_\alpha Q_\alpha c_\alpha$ for $\alpha = 1, 2, 3, 4$. According to system (16), they must satisfy

$$f_\beta = S_\beta\left(y - \sum_{\alpha \neq \beta} f_\alpha \right) \quad \text{for } \beta = 0, 1, 2, 3, 4, \tag{17}$$

where $S_0 := S(S^TS)^{-1}S^T$ and $S_\beta := (Q_\beta + (1/\theta_\beta)I)^{-1}Q_\beta$ for $\beta = 1, 2, 3, 4$. Since (15) must have a minimizer, (16), hence (17), must have a solution. For any solution of (17), there exists a corresponding solution to (16) which in turn corresponds to the uniquely defined SS-ANOVA estimate f_θ; therefore the solution of (17) is unique.

The system (17) suggests a natural iterative method to compute f_β's, that is,

$$f_\beta^{(k)} = S_\beta\Big(y - \sum_{\alpha<\beta} f_\alpha^{(k)} - \sum_{\alpha>\beta} f_\alpha^{(k-1)}\Big) \quad \text{for } \beta = 0, 1, 2, 3, 4. \tag{18}$$

This is exactly the backfitting algorithm studied by Buja, Hastie and Tibshirani (1989) in which additive models are fitted. In the case of additive models, each S_β is a one-dimensional smoother and has a sparse matrix representation which makes computation very efficient. Here each S_β has a tensor product structure because S and Q_β do. Therefore, updating in (18) can be done with the decompositions of matrices Q_P and Q_t which are much smaller than Q_β's. In this way, we may save computational memory and time. The tensor product structure plays the same role here as the sparsity structure does in additive models.

Rewrite (17) as

$$\begin{pmatrix} I & S_0 & \cdots & S_0 \\ S_1 & I & \cdots & S_1 \\ & & \cdots & \\ S_4 & S_4 & \cdots & I \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_4 \end{pmatrix} = \begin{pmatrix} S_0 y \\ S_1 y \\ \vdots \\ S_4 y \end{pmatrix}. \tag{19}$$

It is clear that the backfitting algorithm (18) is a (block) Gauss–Seidel algorithm.

Since all the smoothers $\{S_\alpha\}$ here are symmetric with eigenvalues in $[0, 1]$, Theorem 9 of Buja, Hastie and Tibshirani (1989) shows that the backfitting algorithm (18) starting with any initial vectors $\{f_\alpha^{(0)}\}$ does converge to the unique solution of (17). Chen, Gu and Wahba (1989) pointed out the possibility of applying the backfitting algorithm to SS-ANOVA models and the above derivation of (18) is basically theirs. Another interesting discussion about the convergence of backfitting is given by Ansley and Kohn (1994). Note that (17) is the system of stationary equations for the following minimization problem:

$$\min_{f_0\in\mathscr{L}(S),\, f_\alpha\in\mathscr{L}(Q_\alpha)} \Big\| y - \sum_{\alpha=0}^{4} f_\alpha \Big\|^2 + \sum_{\alpha=1}^{4} \frac{1}{\theta_\alpha} f_\alpha^T Q_\alpha^\dagger f_\alpha, \tag{20}$$

where $Q_\alpha^\dagger$ is the Moore–Penrose generalized inverse of Q_α, and $\mathscr{L}(A)$ denotes the space spanned by the columns of A. The backfitting algorithm (18) is the same as the coordinate descent algorithm for this minimization problem where each component f_α is treated as one "coordinate." See also the discussion in Buja, Hastie and Tibshirani [(1989), pages 477 and 488]. As a matter of fact, thinking the backfitting algorithm in this way, there is a direct way to show

its convergence. Let Γ_α be a matrix such that its columns are the eigenvectors of Q_α that correspond to the nonzero eigenvalues. Let Λ_α be a diagonal matrix with its diagonal elements given by the nonzero eigenvalues of Q_α. Then (20) is the same as

$$\min_{d,\, a_1,\, a_2,\, a_3,\, a_4} \left\| y - Sd - \sum_{\alpha=1}^{4} \Gamma_\alpha a_\alpha \right\|^2 + \sum_{\alpha=1}^{4} \frac{1}{\theta_\alpha} a_\alpha^T \Lambda_\alpha^{-1} a_\alpha, \tag{21}$$

which has a system of stationary equations:

$$\begin{pmatrix} S^TS & S^T\Gamma_1 & \cdots & S^T\Gamma_4 \\ \Gamma_1^TS & I + \frac{1}{\theta_1}\Lambda_1^{-1} & \cdots & \Gamma_1^T\Gamma_4 \\ & & \cdots & \\ \Gamma_4^TS & \Gamma_4^T\Gamma_1 & \cdots & I + \frac{1}{\theta_4}\Lambda_4^{-1} \end{pmatrix} \begin{pmatrix} d \\ a_1 \\ \vdots \\ a_4 \end{pmatrix} = \begin{pmatrix} S^Ty \\ \Gamma_1^Ty \\ \vdots \\ \Gamma_4^Ty \end{pmatrix}.$$

The matrix on the left side is symmetric and nonnegative definite because the objective function in (21) is quadratic in d and a_α's and nonnegative for any choice of d and a_α's. It is also nonsingular since (20) has a unique minimizer because (17) has a unique solution. Now, theorems about the convergence of the Gauss–Seidel algorithm for a positive definite matrix in the numerical analysis literature can be applied here. Note that Theorem 10.1.2 in Golub and Van Loan (1989) is also true for the block Gauss–Seidel algorithm, and the proof is exactly the same as the one given there except that all the matrices should be interpreted as corresponding block matrices.

4. Techniques for speeding up the backfitting algorithm. While the backfitting algorithm has significantly reduced the memory demand from $O(n_t^2 n_P^2)$ to $O(\max(n_t, n_P)^2)$, a straightforward implementation of this algorithm often needs many iteration steps to converge. The problem is especially serious here because with interaction terms in the SS-ANOVA models, the "correlation" between component functions, which is a main factor slowing down backfitting, is usually much higher than that in additive models. There are many studies on how to speed up the Gauss–Seidel algorithm in the numerical analysis literature, especially on a technique called successive over-relaxation. See, for example, Young (1971). Here we would like to discuss several techniques in the context of fitting smoothing spline ANOVA models.

4.1. *Orthogonality.* An important reason for the slowness of the convergence of the backfitting algorithm is the correlation between the components. To illustrate this point, consider a trivial problem of minimizing $h(x, y) := x^2 - 2\rho xy + y^2$ where ρ is between -1 and 1. Starting with an arbitrary point $(x^{(0)}, y^{(0)})$, the kth iteration of the coordinate descent or backfitting algorithm is $x^{(k+1)} = \rho y^{(k)}$, $y^{(k+1)} = \rho x^{(k+1)}$. Therefore, $x^{(k+1)} = \rho^2 x^{(k)}$. Hence the speed of $x^{(k)}$ going to the solution 0 depends on ρ^2. The larger the absolute "correlation coefficient" ρ (thinking of h as the negative log-likelihood of

a two-dimensional Gaussian random vector) is, the slower the convergence of the backfitting algorithm will be. If ρ is zero, the backfitting algorithm converges to the solution in one step. Note that the corresponding linear system for $h(x, y)$ is

$$\begin{pmatrix} 1 & -\rho \\ -\rho & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Therefore, if possible, we may want to formulate the original problem in such a way that as many as possible off-diagonal elements of the system matrix [the matrix on the left side of (19)] are zeros.

Recall that $Q_t = (L^T L)^\dagger$ where L is given by (9), hence $Q_t 1 = Q_t \phi = \phi^T 1 = 0$. Considering (13), it is clear that all $Q_\alpha Q_\beta$ for $\alpha \neq \beta$ and $Q_\alpha S$ are zero matrices except $Q_1 Q_4$, $Q_2 S$ and $Q_3 S$. Since $f_0 \in \mathscr{N}(S)$, $f_\alpha \in \mathscr{N}(Q_\alpha)$, the left side of (19) is

$$\begin{pmatrix} I & 0 & S_0 & S_0 & 0 \\ 0 & I & 0 & 0 & S_1 \\ S_2 & 0 & I & 0 & 0 \\ S_3 & 0 & 0 & I & 0 \\ 0 & S_4 & 0 & 0 & I \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix}.$$

Therefore, solving (19) is equivalent to solving the following two smaller systems separately:

$$\begin{pmatrix} I & S_0 & S_0 \\ S_2 & I & 0 \\ S_3 & 0 & I \end{pmatrix} \begin{pmatrix} f_0 \\ f_2 \\ f_3 \end{pmatrix} = \begin{pmatrix} S_0 y \\ S_2 y \\ S_3 y \end{pmatrix} \tag{22}$$

and

$$\begin{pmatrix} I & S_1 \\ S_4 & I \end{pmatrix} \begin{pmatrix} f_1 \\ f_4 \end{pmatrix} = \begin{pmatrix} S_1 y \\ S_4 y \end{pmatrix}. \tag{23}$$

The key to such a reduction is that variable t has an equally spaced design, but the choice of penalty form is also important. For example, if the penalty on the nonlinear part of a function $g(t)$ is chosen as $\int (g''(t))^2 \, dt$ treating t as a continuous variable, then the above orthogonality will not be true even though it may be approximately so. Note that orthogonality here is different from the orthogonality discussed in Section 2. While we can always choose appropriate norms to make the subspaces there orthogonal, their corresponding Q matrices, which are the reproducing kernels evaluated at data points, may not be orthogonal. If the design for variable P is also uniformly spaced in some sense, then by choosing appropriate penalty terms, we may make all $Q_\alpha Q_\beta$ for $\alpha \neq \beta$ and $Q_\alpha S$ zero matrices or at least close to that. In that case, $f_\alpha = S_\alpha y$ or at least approximately, that is, we only need to apply marginal smoothers once

to the data in order to get all the component functions including interaction terms.

4.2. *Grouping and collapsing.* Since, as discussed before, the "correlation" between component functions is a main factor that slows down the backfitting algorithm, sometimes we may want to update two (or more) highly correlated components simultaneously in one iteration step to get rid of the effect of their correlation. This is called "grouping." For example, suppose we want to group f_{α_1} and f_{α_2} together. Rewrite (17) as

$$\begin{pmatrix} I & S_{\alpha_1} \\ S_{\alpha_2} & I \end{pmatrix} \begin{pmatrix} f_{\alpha_1} \\ f_{\alpha_2} \end{pmatrix} = \begin{pmatrix} S_{\alpha_1} \\ S_{\alpha_2} \end{pmatrix} \Big(y - \sum_{\alpha \neq \alpha_1, \alpha_2} f_\alpha \Big)$$

$$f_\beta = S_\beta \Big(y - \sum_{\alpha \neq \beta} f_\alpha \Big) \quad \text{for } \beta \neq \alpha_1, \alpha_2.$$

This suggests an updating scheme similar to (18) except that now f_{α_1} and f_{α_2} are updated simultaneously.

In many cases grouping will reduce the number of iterations needed in the backfitting algorithm [see Varga (1962), page 80], for a discussion and a counter-example of this benefit. Of course, now each updating step is usually more expensive due to the higher dimension of the components to be updated in each step. In an extreme case, if we group all the components together, then the backfitting algorithm will "converge" in one iteration step. But, of course, this just means that we will solve (17) directly. Obviously, a compromise is needed between the cost of each updating step and the number of components we want to group together.

In order to get any real benefits from grouping, we must be able to update $(f_{\alpha_1}, f_{\alpha_2})$ efficiently. In the case of $\alpha_1, \alpha_2 \geq 1$, we can rewrite the updating equations for f_{α_1} and f_{α_2} as

$$\begin{pmatrix} \Big(Q_{\alpha_1} + \dfrac{1}{\theta_{\alpha_1}} I \Big) & Q_{\alpha_1} \\ Q_{\alpha_2} & \Big(Q_{\alpha_2} + \dfrac{1}{\theta_{\alpha_2}} I \Big) \end{pmatrix} \begin{pmatrix} f_{\alpha_1} \\ f_{\alpha_2} \end{pmatrix} = \begin{pmatrix} Q_{\alpha_1} \\ Q_{\alpha_2} \end{pmatrix} \Big(y - \sum_{\alpha \neq \alpha_1, \alpha_2} f_\alpha \Big).$$

When $Q_{\alpha_1} Q_{\alpha_2} = Q_{\alpha_2} Q_{\alpha_1}$, multiplying both sides by

$$\begin{pmatrix} Q_{\alpha_2} + \dfrac{1}{\theta_{\alpha_2}} I & -Q_{\alpha_1} \\ -Q_{\alpha_2} & Q_{\alpha_1} + \dfrac{1}{\theta_{\alpha_1}} I \end{pmatrix},$$

we get explicit updating formulae for f_{α_1} and f_{α_2},

$$
\begin{aligned}
f_{\alpha_1} &= \theta_{\alpha_1}(Q_{\alpha_1+\alpha_2}+I)^{-1}Q_{\alpha_1}\Big(y-\sum_{\alpha\neq\alpha_1,\,\alpha_2} f_\alpha\Big),\\
f_{\alpha_2} &= \theta_{\alpha_2}(Q_{\alpha_1+\alpha_2}+I)^{-1}Q_{\alpha_2}\Big(y-\sum_{\alpha\neq\alpha_1,\alpha_2} f_\alpha\Big),
\end{aligned}
\tag{24}
$$

where $Q_{\alpha_1+\alpha_2}$ denotes $\theta_{\alpha_1}Q_{\alpha_1}+\theta_{\alpha_2}Q_{\alpha_2}$. If $Q_{\alpha_1+\alpha_2}$ has a tensor-product structure of the kind in (13), then each updating step can be done efficiently.

Another way to reduce the number of iterations is to use a technique that we call "collapsing." Suppose that we want to collapse f_{α_1} and f_{α_2} (where $\alpha_1, \alpha_2 \geq 1$) together. By manipulating

$$
f_\beta = \Big(Q_\beta + \frac{1}{\theta_\beta}I\Big)^{-1}Q_\beta\Big(y-\sum_{\alpha\neq\beta} f_\alpha\Big) \quad \text{for } \beta=\alpha_1,\alpha_2
$$

of (17), we get

$$
f_{\alpha_1}+f_{\alpha_2} = (Q_{\alpha_1+\alpha_2}+I)^{-1}Q_{\alpha_1+\alpha_2}\Big(y-\sum_{\alpha\neq\alpha_1,\,\alpha_2} f_\alpha\Big). \tag{25}
$$

This can also be derived from (15) with a restriction $c_{\alpha_1}=c_{\alpha_2}$. Now the collapsed updating equations are

$$
\begin{aligned}
f_{\alpha_1}+f_{\alpha_2} &= (Q_{\alpha_1+\alpha_2}+I)^{-1}Q_{\alpha_1+\alpha_2}\Big(y-\sum_{\alpha\neq\alpha_1,\,\alpha_2} f_\alpha\Big),\\
f_\beta &= S_\beta\Big(y-\sum_{\alpha\neq\beta} f_\alpha\Big) \quad \text{for } \beta\neq\alpha_1,\alpha_2.
\end{aligned}
$$

Note that (24) implies (25), but (25) does not need the condition that Q_{α_1} and Q_{α_2} are commutable. Again, if $Q_{\alpha_1+\alpha_2}$ has a tensor-product structure, updating can be done efficiently. After $f_{\alpha_1}+f_{\alpha_2}$ and other f_α's are computed, c of (8) can be computed using (12), and then f_{α_1} and f_{α_2} can be computed using (8).

Specifically, consider f_1 and f_4,

$$
f_1+f_4 = (Q_{1+4}+I)^{-1}Q_{1+4}(y-f_0-f_2-f_3) = (Q_{1+4}+I)^{-1}Q_{1+4}y,
$$

where the last equality is due to the orthogonality discussed in Section 4.1, and $Q_{1+4} := \theta_1(11^T\otimes Q_t)+\theta_4(Q_P\otimes Q_t) = (\theta_1 11^T+\theta_4 Q_P)\otimes Q_t$. Therefore, f_1+f_4 can be computed by decomposing $(\theta_1 11^T+\theta_4 Q_P)$ and Q_t, directly without any iterations. In the case of f_2 and f_3,

$$
f_2+f_3 = (Q_{2+3}+I)^{-1}Q_{2+3}(y-f_0-f_1-f_4) = (Q_{2+3}+I)^{-1}Q_{2+3}(y-f_0),
$$

where $Q_{2+3} := \theta_2(Q_P\otimes 11^T)+\theta_3(Q_P\otimes\phi\phi^T) = Q_P\otimes(\theta_2 11^T+\theta_3\phi\phi^T)$. Since

$$
\begin{aligned}
f_0 &= S_0(y-f_2-f_3)\\
&= S_0\big(y-(Q_{2+3}+I)^{-1}Q_{2+3}(y-f_0)\big),
\end{aligned}
$$

and $f_0 = Sd$, some simple manipulations lead to

$$d = \left(S^T(I + Q_{2+3})^{-1}S\right)^{-1}S^T(I + Q_{2+3})^{-1}y,$$

which can be computed directly using the eigen-decomposition of $Q_{2+3} = Q_P \otimes (\theta_2 11^T + \theta_3 \phi\phi^T)$. Then f_2 and f_3 can be computed using $f_2 = S_2(y - f_0) = S_2(y - Sd)$, $f_3 = S_3(y - f_0) = S_3(y - Sd)$.

Note that if we collapse all four f_α's together, we will end up with (11), where Q_θ, unlike Q_{1+4} or Q_{2+3}, does not have a tensor product structure. Therefore, a compromise is also needed between the number of components we want to collapse together and the cost of each updating step.

We note that both grouping and collapsing have been used in speeding up the Gibbs sampler which is closely related to the backfitting algorithm. See, for example, Liu (1994) and Roberts and Sahu (1997). In the latter, the relationship between the Gibbs sampler and the Gauss–Seidel algorithm is explicitly used to study the properties of the Gibbs sampler.

4.3. *Successive overrelaxation.* An important technique to speed up the backfitting (Gauss–Seidel) algorithm is the method of successive overrelaxation (SOR). See, for example, Golub and Van Loan (1989) or Young (1971).

The (block) SOR scheme corresponding to the (block) Gauss–Seidel updating scheme (18) is

$$f_\alpha^{(k+1)} = \omega\left\{S_\alpha\left(y - \sum_{\beta<\alpha} f_\beta^{(k+1)} - \sum_{\beta>\alpha} f_\beta^{(k)}\right)\right\} + (1-\omega)f_\alpha^{(k)}, \tag{26}$$

where ω is a real number known as the relaxation factor. With $\omega = 1$, we are back to the Gauss–Seidel algorithm.

By Proposition 11 of Buja, Hastie and Tibshirani (1989), SOR converges for any ω in $(0, 2)$. The trick now is to find a good ω. In general a prescribed optimal ω is available only for some special kinds of matrices. Fortunately our case falls into one of such categories. A $n \times n$ matrix A is said to be consistently ordered if there exists a disjoint partition of $\{1, 2, \ldots, n\} = \bigcup_{k=1}^K W_k$ such that for any $i \in W_k$, a_{ij} or $a_{ji} \neq 0$ implies that $j \in W_{k+1}$ if $j > i$ and $j \in W_{k-1}$ if $j < i$. Basically, it means that after some permutations of rows and columns, a consistently ordered matrix has a block tridiagonal form,

$$\begin{pmatrix} D_1 & H_1 & 0 & \cdots & 0 & 0 \\ G_1 & D_2 & H_2 & \cdots & 0 & 0 \\ 0 & G_2 & D_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & D_{K-1} & H_{K-1} \\ 0 & 0 & 0 & \cdots & G_{K-1} & D_K \end{pmatrix},$$

where the D_i are square diagonal matrices. Let $S(A)$ denote the spectral radius of a matrix A, that is, the largest eigenvalue of A in absolute value. The following theorem in Young (1971) shows the convergence of SOR and the best choice of ω in the case described in the theorem.

THEOREM [Theorem 2.2 and 2.3 of Young (1971), Chapter 6]. *Let A be a consistently ordered matrix with nonvanishing diagonal elements such that $B := I - (diag\ A)^{-1}A$ has real eigenvalues and such that $\overline{\mu} := S(B) < 1$, then $S(\mathscr{L}_\omega) < 1$ for any ω in $(0, 2)$, and*

$$S(\mathscr{L}_\omega) = \begin{cases} \frac{1}{4}\Big[\omega\overline{\mu} + \sqrt{\omega^2\overline{\mu}^2 - 4(\omega-1)}\Big]^2, & \text{if } 0 < \omega \le \omega_b, \\ \omega - 1, & \text{if } \omega_b \le \omega < 2, \end{cases}$$

where

$$\omega_b = \frac{2}{1 + \sqrt{1 - \overline{\mu}^2}}. \tag{27}$$

and $\mathscr{L}_\omega = (I - \omega L)^{-1}(\omega U + (1-\omega)I)$ is the iteration matrix of SOR for solving linear system $Ax = b$, L and U are lower and upper triangular matrices of B, respectively.

From this theorem, it is easy to verify that $S(\mathscr{L}_\omega)$ is a decreasing function of ω for $0 < \omega \le \omega_b$ and an increasing function for $\omega_b \le \omega < 2$; hence it reaches its minimum at ω_b. This means that ω_b is the best choice of ω because the convergence rate of SOR depends on $S(\mathscr{L}_\omega)$. It can also be shown that a small decrease in ω_b results in a much larger relative increase in $S(\mathscr{L}_\omega)$ than a corresponding increase in ω_b. That is, an ω slightly larger than ω_b is better than an ω slightly smaller than ω_b. See Young [(1971), page 202], for details.

In the case of (22),

$$A := \begin{pmatrix} I & S_0 & S_0 \\ S_2 & I & 0 \\ S_3 & 0 & I \end{pmatrix}$$

is obviously consistently ordered with $W_1 := \{1, 2, \ldots, n\}$ and $W_2 := \{n+1, n+2m, \ldots, 3n\}$. In order to apply this theorem, we only need to show that all the eigenvalues of $B := I - (\text{diag } A)^{-1}A$ are real and have absolute values less than 1.

Since

$$\begin{aligned} |B - \lambda I| &= \left| \begin{pmatrix} -\lambda I & -S_0 & -S_0 \\ -S_2 & -\lambda I & 0 \\ -S_3 & 0 & -\lambda I \end{pmatrix} \right| \\ &= (-1)^{3n}\lambda^{2n} \left| \lambda I - (S_0\ S_0)(\lambda I)^{-1} \begin{pmatrix} S_2 \\ S_3 \end{pmatrix} \right| \\ &= (-1)^{3n}\lambda^n \left| \lambda^2 I - S_0(S_2 + S_3) \right| \end{aligned}$$

(this is true for all nonzero λ, hence for all λ, because both sides are continuous). Therefore all the eigenvalues of B are

$$\{0, \pm\sqrt{\mu_i}, i = 1, \ldots, n\},$$

where $\{\mu_1, \ldots, \mu_n\}$ are eigenvalues of $S_0(S_2 + S_3)$ and 0 has a multiplicity n. They are clearly real, since all S_0, S_2, S_3 are nonnegative definite. Matrix S_0 has eigenvalues either 0 or 1 because it is a projection matrix. Because the largest eigenvalue of $S_0(S_2 + S_3)$ is less than or equal to the product of the largest eigenvalues of S_0 and $S_2 + S_3$, we only need to show that all the eigenvalues of $S_2 + S_3$ are less than 1. Let $Q_P = \Gamma_P \Lambda_P \Gamma_P^T$, $Q_t = \Gamma_t \Lambda_t \Gamma_t^T$, $\Lambda_P = \operatorname{diag}(\lambda_j^P)_{j=1}^{n_P}$, $\Lambda_t = \operatorname{diag}(\lambda_i^t)_{i=1}^{n_t}$. Since $Q_t 1 = Q_t \phi = 0$ and $\phi^T 1 = 0$, we can choose Γ_t so that its first two columns are $1/\sqrt{n_t}$ and $\phi/\|\phi\|$, where $\|\phi\| = \sqrt{\sum_{t=1}^{n_t} \phi^2(t)}$. Considering (13), we have

$$\begin{aligned} S_2 &= \left(Q_2 + \frac{1}{\theta_2} I\right)^{-1} Q_2 \\ &= (\Gamma_P \otimes \Gamma_t)\left(\left(\Lambda_P \otimes \Lambda_2 + \frac{1}{\theta_2} I\right)^{-1} (\Lambda_P \otimes \Lambda_2)\right)(\Gamma_P \otimes \Gamma_t)^T, \\ S_3 &= \left(Q_3 + \frac{1}{\theta_3} I\right)^{-1} Q_3 \\ &= (\Gamma_P \otimes \Gamma_t)\left(\left(\Lambda_P \otimes \Lambda_3 + \frac{1}{\theta_3} I\right)^{-1} (\Lambda_P \otimes \Lambda_3)\right)(\Gamma_P \otimes \Gamma_t)^T, \end{aligned}$$

where Λ_2 is a $n_1 \times n_1$ matrix with all its elements being zero except the first diagonal one, which is n_1, Λ_3 is a $n_1 \times n_1$ matrix with all its elements being zero except the second diagonal one, which is $\|\phi\|^2$. Hence, it is clear that the eigenvalues of $(S_2 + S_3)$ are

$$\left\{0, \frac{\lambda_j^P n_t}{\lambda_j^P n_t + 1/\theta_2}, \frac{\lambda_j^P \|\phi\|^2}{\lambda_j^P \|\phi\|^2 + 1/\theta_3}, j = 1, 2, \ldots, n_P\right\}, \tag{28}$$

where 0 has a multiplicity $(n_t - 2) \times n_P$. Therefore, all $(S_2 + S_3)$'s eigenvalues are in $[0, 1)$.

Note that the cited theorems of Young (1971) are only stated for the point Gauss–Seidel or SOR algorithms. In our case, however, point and block versions are the same. Since the diagonal blocks in our linear systems are all identity matrices, updating one element by one element is the same as updating all the elements in one block simultaneously.

In practice, however, we usually do not know $\overline{\mu}$. Therefore in order to apply this theorem, we have to estimate it. The above analysis actually has given us an upper bound on $\overline{\mu}$ which is the maximum of (28). This may be used in (27) to give an overestimated ω_b. Note that a slightly overestimated ω_b is better than a slightly underestimated ω_b. It can also be shown [Young (1971), pages 144 and 147] that $\overline{\mu}^2$ is the spectral radius of the Gauss–Seidel iteration

matrix, which can be estimated by the power method after some Gauss–Seidel iteration steps are taken. See Young [(1971), page 206], for an explanation. This estimated $\overline{\mu}$ can then be used in (27) to get an estimated ω_b.

A similar argument can be used for applying this theorem to system (23).

5. The case of a near tensor product design. So far we have assumed that our data are complete in the sense that there is one observation at every tensor product grid point. But in practice, we will frequently have missing data. The computational procedures discussed in Sections 3 and 4 are still applicable with the help of an iterative imputation procedure.

For simplicity, suppose that we have reordered the data in such a way that the complete data y can be written in the following form:

$$y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \end{pmatrix}, \tag{29}$$

where $y^{(1)}$ is the observed part, and $y^{(2)}$ is the missing part. The iterative imputation procedure starts with any initial values for the missing part, then computes the SS-ANOVA estimate of f based on this "complete" data set using the backfitting algorithm, then calculates the values of the estimated f at the missing part, then imputes $y^{(2)}$ with these new values, then goes back to compute the SS-ANOVA estimate of f again based on the new "complete" data set and so on. It keeps going through this cycle until the estimated f does not change anymore. We note that Yates (1933) used a similar idea to fit an ordinary ANOVA model to the data with a few missing values without solving a general linear model equation.

Assuming that ε's in (1) are independent Gaussian random variables with a common known variance, this iterative imputation procedure can be shown to be equivalent to the EM algorithm applied to a maximum penalized likelihood estimation problem. See Dempster, Laird and Rubin (1977) for general discussions on the EM algorithm and Green (1990) for its application to maximum penalized likelihood estimation.

Treating the missing data as parameters, this iterative imputation procedure is also equivalent to the coordinate descent method for minimizing the objective function (7) corresponding to a complete data set. Here two "coordinates" are f and the vector of missing data. Since backfitting can also be viewed as a coordinate descent method, the whole procedure is essentially a big nested coordinate descent procedure.

Instead of verifying the conditions of the general results about the EM algorithm as in Wu (1983), or the results about the coordinate descent method, Wahba and Luo (1997) directly derived the convergence rate of this iterative imputation procedure for the SS-ANOVA estimate. The results are described briefly here for the convenience of the reader.

Representation (8) and system (11) show clearly that the SS-ANOVA estimate f_θ evaluated at data points can be expressed as some linear combinations of data with coefficients independent of the data. Let $A(\theta)$ denote this linear

transformation matrix from a data vector to the vector of fitted values in the situation of a complete data set. Partition $A(\theta)$, corresponding to (29), as

$$A(\theta) = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

When A_{22}'s spectral radius is less than 1, the estimated f_θ based on the imputed data converges to the estimated f_θ based on the observed data. The smaller the spectral radius of A_{22} is, the faster this iterative imputation procedure converges.

Let Γ_1 be a matrix of orthonormal columns that span the column space of S, partitioned also corresponding to (29) as

$$\begin{pmatrix} \Gamma_{11} \\ \Gamma_{21} \end{pmatrix}.$$

A necessary and sufficient condition for A_{22} to have a spectral radius less than 1 is that $\Gamma_{21}\Gamma_{21}^T$ does not have 1 as an eigenvalue. This condition can be easily checked for a given data set. Considering

$$S(S^TS)^{-1}S^T = \Gamma_1\Gamma_1^T = \begin{pmatrix} \Gamma_{11}\Gamma_{11}^T & \Gamma_{11}\Gamma_{21}^T \\ \Gamma_{21}\Gamma_{11}^T & \Gamma_{21}\Gamma_{21}^T \end{pmatrix},$$

we may interpret $\Gamma_{21}\Gamma_{21}^T$ as a measure of influence similar to the diagonal elements of the hat matrix in an ordinary linear regression. Then $\Gamma_{21}\Gamma_{21}^T$ having an eigenvalue 1 means that the missing part has an infinite influence.

If we expand Γ_1 into an orthonormal square matrix and partition it corresponding to (29) as

$$\Gamma = (\Gamma_1\ \Gamma_2) = \begin{pmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{pmatrix},$$

then A_{22} can be expressed as $I - \Gamma_{22}(\Gamma_2^T(Q_\theta + I)\Gamma_2)^{-1}\Gamma_{22}^T$. It is clear from this expression that the larger the smoothing parameters (θ's) are, the larger the spectral radius of A_{22} is. In other words, in the situation where the amount of smoothing is smaller (larger θ's), the iterative imputation procedure has a slower convergence rate.

6. An application to a global temperature data set. In this section, an application of SS-ANOVA to a historical global surface air temperature data set is briefly described. For more details, readers are referred to Luo, Wahba and Johnson (1998) or Luo (1996).

An accurate assessment of the climate history based on observations taken in the past is clearly of interest to climatologists. It is more so in recent years when scientists have started to model the global climate dynamics and use the models to understand or predict the climate. One way to assess the accuracy

of these complicated dynamic models is to compare their "predictions" of the past climate with what was actually observed. Among the many variables of interest to climatologists, temperature is probably the most important one and certainly the most intensively recorded so far. For the most part of the period after modern instruments were invented, the only available records are those recorded by surface meteorological stations around the world. To facilitate the comparison between dynamic model predictions and the historical records, we would like to reconstruct the global temperature history using these available records scattered in both time and space. In the process of reconstruction, various factors such as relocation of a station, change of instrument and so on may cause biases. One major source of bias is the incompleteness of time and space coverage. Different approaches have been taken to correct this kind of bias. See, for example, Hansen and Lebedeff (1987), Jones, Raper, Bradley Diaz, Kelly and Wigley (1986) and Vinnikov, Groisman and Lugina (1990). There are also some studies on the effect of incomplete sampling on the estimates of the climate history. See, for example, Madden, Shea, Branstator, Tribbia and Weber (1993) and Karl, Knight and Christy (1994).

To get a series of global averages for a very crude assessment of the climate history, one will have to deal with two types of incomplete sampling. One is the nonuniform distribution of meteorological stations around the world. The other is the existence of missing data for many stations. For one thing, not every station was established in the same year or ended its operation in the same year. The first type of incomplete sampling can be taken care of more or less by estimating the temperature field as a function over the sphere and then averaging the estimated field instead of raw data. In this way, the unbalanced influence of more densely sampled regions can be avoided. The second type of incomplete sampling means that the stations included in the data set change from time to time. If in one year cold regions are covered by the data set more extensively than the year after, then we do not know whether an increase in the average temperature is due to a real global change or just the fact that less cold areas are included in the calculation of the average for the year after. A commonly adopted strategy in climate studies including all those cited above is to use anomalies instead of raw temperature data. An anomaly of a temperature record is defined as the difference between the temperature record and the average temperature at the same location over a prespecified reference period. In the notations of (2), the corresponding anomaly of y_i is the difference between y_i and $d_1 + g_2(P_i)$, that is, $d_2\phi(t_i) + g_1(t_i) + g_{\phi,2}(P_i)\phi(t_i) + g_{12}(t_i, P_i) + \varepsilon_i$ if the prespecified reference period is chosen as the same period covered by the data. In this way, the bias resulting from the spatial variation of g_2 is avoided. But the spatial variations of $g_{\phi,2}$ and g_{12} may still cause biases. Put in another way, the anomaly approach basically assumes an additive model, that is, there are no $g_{\phi,2}\phi$ and g_{12} terms in (2). With all those terms included in the model, SS-ANOVA estimates have at least the potential ability to correct the biases related to all three spatial terms. To get any real benefits from the SS-ANOVA approach, the smoothing parameters (θ's) have to be chosen appropriately. Actually, if θ_3 and θ_4 in (7)

are chosen to be very small, $g_{\phi,2}$ and g_{12} will be essentially wiped out and an additive model will be obtained.

Figure 1 shows three series of global average "winter" temperature estimates based on three methods. Here "winter" temperature is defined as the average of monthly temperatures of December, January and February. The first series (dotted lines) is obtained by averaging each year's raw data. Strictly speaking, it is obtained by averaging the temperature field estimated by a smoothing spline method based on each year's raw data. The smoothing spline method is defined in a similar way as in (7) except that there is only one penalty term needed here since we consider each year separately in this approach. The years 1989 and 1990 have outstanding high values in this series. But it is clearly due to the estimation method used because the records for the Antarctic region in our data end in 1988. The data set we have used in this article is a subset of the so-called Jones–Wigley data set [see Jones, Raper, Cherry, Godess, Wigley, Santer, Kelly, Bradley and Diaz (1991)]. We obtained this data set from `http://cdiac.ESD.ORNL.GOV/ftp/`. It is a combination of four files: `ndp020r1/jonesnh.dat`, `ndp020r1/jonessh.dat`, `ndp032/ndp032.tm1` and `ndp032/ndp032.tm2`. In the subset we have used, there are 1000 stations, that is, $n_P = 1000$, 30 years, that is, $n_t = 30$, and 20910 observations, that is, $n = 20910$.

The second series (dashed lines) is obtained by averaging each year's anomalies. In Figure 1, this series is shifted up by an (estimated) grand average, that is, d_1 in (2), to match other two series. The third series (solid lines) is obtained by the SS-ANOVA approach. In this approach, we first choose $\log_{10}(\theta_1) = -0.1$ and $\log_{10}(\theta_2) = 4.5$, which correspond to the case where little smoothing is done to g_1 and g_2. Then we choose θ_3 and θ_4 by a crude

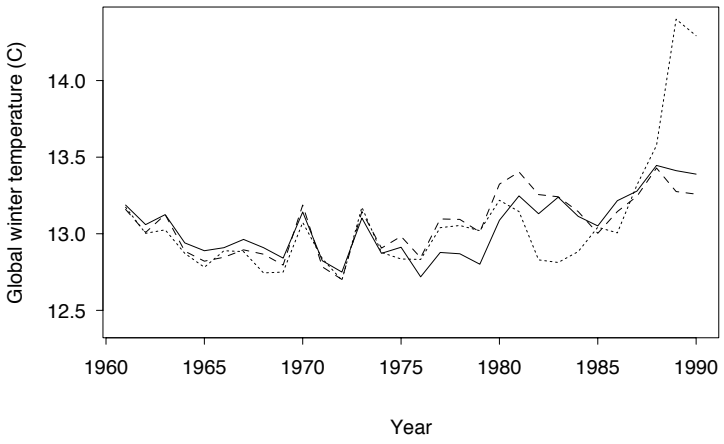

FIG. 1. *Global average "winter" temperature* (°C) *series by three estimation methods. Solid lines*: *SS-ANOVA estimates, that is,* $d_1 + d_2\phi(t) + g_1(t)$ *in* (2); *dotted lines*: *annual averages of raw data*; *dashed lines*: *annual averages of anomalies. The series by the anomaly approach is shifted up by an* (*estimated*) *grand average, that is,* d_1 *in* (2), *to match other two series.*

grid search according to a randomized GCV criterion [see Girard (1989)], which results in $\log_{10}(\theta_3) = 1.25$ and $\log_{10}(\theta_4) = 4.1$. Both the anomaly and SS-ANOVA approaches have corrected the bias in the last two years of the first series. There is not much difference between the anomaly and SS-ANOVA series. One reason for this is that the magnitude of spatial variation in g_2, which has a range roughly $(-40^\circ, 40^\circ)$, is much larger than the magnitude of those in $g_{\phi,2}$ and g_{12}. Another reason, probably a more important one, is that there is not much correlation found between winter temperatures in consecutive years. The advantages of the SS-ANOVA approach are manifest only when there is information that can be borrowed across years. Otherwise, considering each year separately or simultaneously should not make much difference. However, when the correlation over time is stronger, for example, in the case of monthly or daily temperatures, more significant differences between these two approaches will be expected. See Luo, Wahba and Johnson (1998) for more discussions.

From the SS-ANOVA estimate f_θ, the 30-year average "winter" temperature field over the sphere, that is, $d_1 + g_2(P)$, can be obtained and is plotted in Figure 2 for illustration. The 30-year linear trend coefficient as a function of location, that is, $d_2 + g_{\phi,2}(P)$, can also be obtained and is plotted in Figure 3. This plot shows where the cooling regions are and where the warming regions are over the 30-year (1961–1990) period. In Section 3.4 of Luo (1996), some bootstrap estimates of the variation in the SS-ANOVA estimates in Figures 1 and 3 are given. They are particularly useful when the temperature history

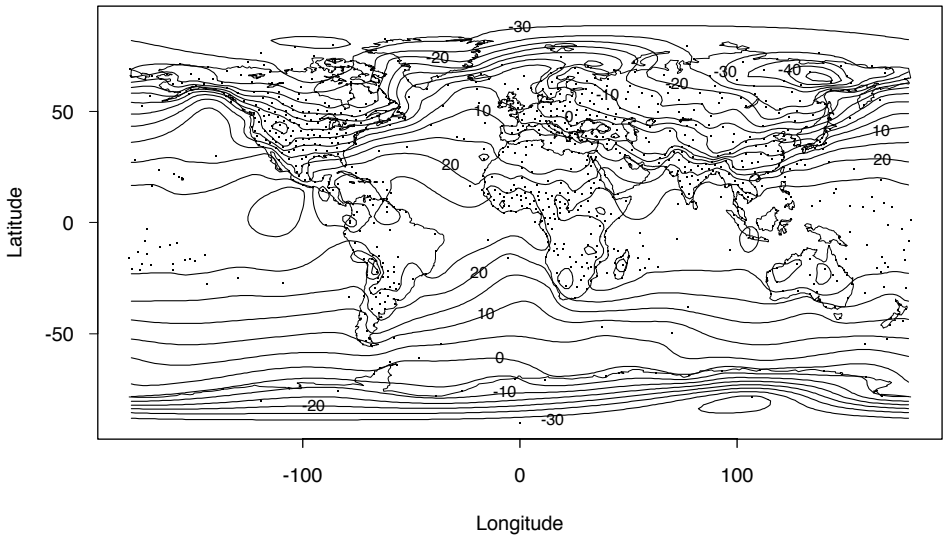

FIG. 2. *A contour plot of* 30*-year* (1961–1990) *average "winter" temperature* (°C) *based on the SS-ANOVA estimate defined by* (7), *that is*, $d_1 + g_2(P)$ *in* (2). *Dots are the locations of the stations included in the analysis.*

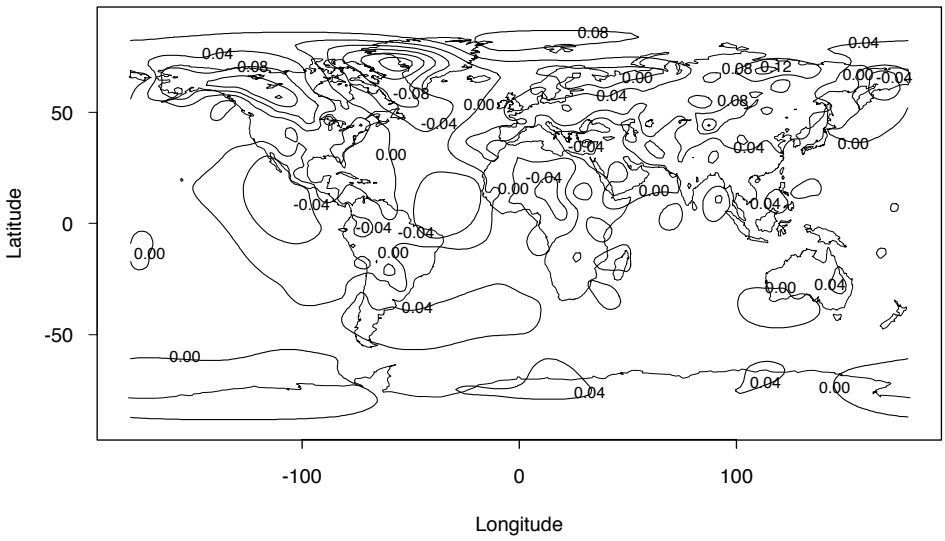

FIG. 3. *A contour plot of* 30*-year* (1961–1990) *linear trend coefficient* (°C/year) *of "winter" temperature based on the SS-ANOVA estimate defined by* (7), *that is*, $d_2 + g_{\phi,2}(P)$ *in* (2).

reconstructed using the method of this paper is compared with dynamic model predictions.

The computational time needed for fitting the SS-ANOVA model to this data set with fixed smoothing parameters is about 2 hours and 17 minutes on a Sun Ultra 1 machine of model 140. To select smoothing parameters according to a data-driven criterion such as GCV, multiple fittings are required. Therefore, for a data set of this size, a few days of computational time are needed.

7. Timing. It is clear that the memory requirement of the backfitting algorithms described in the previous sections is $O(\max(n_t, n_P)^2)$, while the memory requirement of direct matrix decomposition methods such as the one implemented in Rkpack [Gu (1989)] is $O(n^2)$. This means a significant saving in memory when the data have a tensor product design or a near one because in that case n is roughly $n_t n_P$, which is much larger than $\max(n_t, n_P)$. For the application described in Section 6, n is 20910 and $\max(n_t, n_P)$ is 1000. On a machine with 128 MB memory, 2000 is about the largest sample size that Rkpack can handle. In contrast, when the data have a (near) tensor product design, the backfitting algorithm can handle a much larger sample size depending on the ratio of n_t and n_P. The largest size is about $4e+6$ when $n_t = n_P = 2000$.

Comparing the computational time required for Rkpack and backfitting is more complicated. While Rkpack, specifically the procedure *dsidr* with prespecified smoothing parameters, requires $O(n^3)$ computational time, depending largely on the sample size alone, the time needed for the backfitting al-

gorithm depends also on the design of the data and the choice of smoothing parameters.

For iterative procedures such as backfitting, a terminating criterion has to be chosen first before making any comparison of timing. One reasonable choice is to terminate the iteration when the values of (7) in two consecutive iterations have a difference or relative difference smaller than a prespecified small number. To make the comparison of timing more meaningful across different sample sizes or smoothing parameters, however, we decide to terminate the iteration when the absolute difference between the values of every component function in two consecutive iterations is smaller than a fraction, say 1/10,000, of its range. For example, the range of f_2 is roughly 80, which we may obtain from our general knowledge about the problem; therefore an (average) absolute difference less than 0.008 between the values of f_2 in two consecutive iterations will be acceptable for terminating the iteration. For all the calculations made in this article, $1.e-5$, $1.e-5$, $1.e-3$, $1.e-4$ and $1.e-3$ are used for f_0, f_1, f_2, f_3 and f_4, respectively.

Due to the limitation set by the memory requirement of Rkpack, we selected only 100 stations from the data set used in Section 6 and ended up with a data set containing 2046 observations. Then we fit the SS-ANOVA model, using three sets of smoothing parameters corresponding to a less and less amount of smoothing. For each set, we fit the model using Rkpack and three versions of backfitting. The first version, denoted by Bkfit0, is a direct implementation of the backfitting algorithm. The second one, denoted by Bkfit1, implements the backfitting algorithm with the aid of SOR. The last one, denoted by Bkfit2, implements the backfitting algorithm with the aid of collapsing. For simplicity, f_1 and f_4 are collapsed in all three versions. Therefore, the comparison among these three versions is about backfitting f_0, f_2 and f_3. For SOR, the ω is calculated by (27) with an $\overline{\mu}$ estimated through some initial iterations of the Gauss–Seidel scheme. In Bkfit2, after collapsing f_2 and f_3, no iterations are needed for backfitting and the iterations there belong to the iterative imputation procedure alone. The three sets of smoothing parameters are given in Table 2, together with their $\mathrm{tr}(S_\alpha(\theta_\alpha))$. These traces may be interpreted as the "degrees of freedom" of their corresponding component functions, as done in Buja, Hastie and Tibshirani (1989). The bigger they are, the closer the estimated function is to the observations, that is, to an interpolation.

TABLE 2
Three choices of smoothing parameters and their associated "degrees of freedom", $\mathrm{tr}(S_\alpha(\theta_\alpha))$

	Case I		**Case II**		**Case III**	
α	$\log_{10}(\theta_\alpha)$	$\mathrm{tr}(S_\alpha)$	$\log_{10}(\theta_\alpha)$	$\mathrm{tr}(S_\alpha)$	$\log_{10}(\theta_\alpha)$	$\mathrm{tr}(S_\alpha)$
1	0.5	27.5	0.5	27.5	0.5	27.5
2	3.0	98.4	5.0	99.98	6.0	99.998
3	0.0	83.2	0.0	83.2	0.0	83.2
4	1.5	574.2	1.5	574.2	3.0	1498.8

TABLE 3
The timing (in seconds) of Rkpack and three versions of backfitting

	Rkpack	**Bkfit0**	**Bkfit1**	**Bkfit2**
Case I	291.5	19.3	18.4	18.3
Case II	295.2	133.7	77.5	22.3
Case III	289.4	*685.2	344.1	94.8

*Means that after 10,000 iterations still not converged.

The timing of these four procedures is listed in Table 3. All computations are carried out on a Sun Ultra 1 machine of Model 140 with a 128 MB memory. It is apparent from this table that Rkpack needs about the same amount of time, no matter what choice of smoothing parameters. In contrast, backfitting in general needs more time for the cases of larger θ's, or less amount of smoothing. Since in Bkfit2, no iterations are actually done for backfitting, the differences between the three cases are due to the different numbers of iterations needed for the iterative imputation to converge. Therefore it is clear that the iterative imputation procedure converges more slowly for the cases of larger θ's. This is also clear from the discussion at the end of Section 5. In general, backfitting will save computational time (compared with Rkpack) if the smoothing parameters (θ's) are not too large. Note that Case III is an extreme case because the "degrees of freedom" of f_2 is 99.998 (see Table 2) while its maximum "degrees of freedom" is 100. Hence this is a case very close to an interpolation. As a matter of fact, it is not difficult to see that if the smoothing parameters are chosen to be infinities, that is, to force the estimated function to interpolate the data, then the iterative imputation will not work at all.

In Case I, the differences between the three versions of backfitting are very small. The reason for this is that backfitting converges very quickly in this case; therefore the major part of the time is spent on the iterations for the imputation. In Case II where θ_2 is increased from its value in Case I, the differences between the three versions are much greater, since now backfitting converges more slowly and the benefits of speeding-up techniques become evident. The differences are even greater in Case III where θ's are increased further. In this case, after 10,000 iterations, Bkfit0 still has yet to converge.

In conclusion, the saving of computational memory by backfitting is apparent. It is also clear from the experiment described above that if the smoothing parameters are not too large, or in other words, if the estimates are not expected to be too close to an interpolation, then backfitting can also save computational time. The amount of savings in time by backfitting depends on the choice of smoothing parameters, and on the amount of missing data, or strictly speaking, the information contained in the missing data. The effectiveness of speeding up techniques discussed in Section 4 also depends on these factors. When there is a significant amount of missing information, speeding up the iterative imputation will be more important. Effective ways to do that with a limited memory requirement is a topic that merits future research.

Acknowledgments. This work is part of my Ph.D. dissertation in the Department of Statistics, University of Wisconsin–Madison. I thank my advisor, Professor Grace Wahba, for her kind advisory and constant encouragement. I also thank an Associate Editor and two referees for their helpful comments, which have improved the manuscript significantly.

REFERENCES

ANSLEY, C. F. and KOHN, R. (1994). Convergence of the backfitting algorithm for additive models. *J. Austral. Math. Soc. Ser. A* **57** 316–329.

ARONSZAJN, N. (1950). Theory of reproducing kernels. *Trans. Amer. Math. Soc.* **68** 337–404.

BUJA, A., HASTIE, T. and TIBSHIRANI, R. (1989). Linear smoothers and additive models (with discussion). *Ann. Statist.* **17** 453–555.

CHEN, Z., GU, C. and WAHBA, G. (1989). Discussion of "Linear smoothers and additive models" by Buja, Hastie and Tibshirani. *Ann. Statist.* **17** 515–517.

DEMPSTER, A. P., LAIRD, N. M. and RUBIN, D. B. (1977). Maximum likelihood from incomplete data via the *EM* algorithm. *J. Royal Statist. Soc. Ser. B* **39** 1–38.

GIRARD, D. (1989). A fast "Monte Carlo cross-validation" procedure for large least squares problems with noisy data. *Numer. Math.* **56** 1–23.

GOLUB, G. H. and VAN LOAN, C. F. (1989). *Matrix Computations*, 2nd ed. Johns Hopkins Univ. Press.

GREEN, P. J. (1990). On use of the EM Algorithm for Penalized Likelihood Estimation. *J. Royal Statist. Soc. Ser. B* **52** 443–452.

GU, C. (1989). RKPACK and its applications: Fitting smoothing spline models. Technical Report 857, Dept. Statistics, Univ. Wisconsin–Madison.

GU, C. and WAHBA, G. (1993a). Semiparametric analysis of variance with tensor product thin plate splines. *J. Royal Statist. Soc. Ser. B* **55** 353–368.

GU, C. and WAHBA, G. (1993b). Smoothing spline ANOVA with component-wise Bayesian confidence intervals. *J. Comput. Graph. Statist.* **2** 97–117.

HANSEN, J. and LEBEDEFF, S. (1987). Global trends of measured surface air temperature. *J. Geophysical Research* **92** 13,345–13,372.

JONES, P. D., RAPER, S. C. B., CHERRY, B. S. G., GOODESS, C. M., WIGLEY, T. M. L., SANTER, B., KELLY, P. M., BRADLEY, R. S. and DIAZ, H. F. (1991). An updated global grid point surface air temperature anomaly data set: 1851–1988. Environmental Sciences Division Publication 3520, U.S. Dept. Energy, Washington, DC.

JONES, P. D., RAPER, S. C. B., BRADLEY, R. S., DIAZ, H. F., KELLY, P. M. and WIGLEY, T. M. L. (1986). Northern hemisphere surface air temperature variations: 1851–1984. *J. Climate and Applied Meteorology* **25** 161–179.

KARL, T. R., KNIGHT, R. W. and CHRISTY, J. R. (1994). Global and hemispheric temperature trends: uncertainties related to inadequate spatial sampling. *J. Climate* **7** 1144–1163.

LIU, J. S. (1994). The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem. *J. Amer. Statist. Assoc.* **89** 958–966.

LUO, Z. (1996). Backfitting in smoothing spline ANOVA with application to historical global temperature data (thesis). Technical Report 964, Dept. Statistics, Univ. Wisconsin, Madison.

LUO, Z., WAHBA, G. and JOHNSON, D. R. (1998). Spatial-temporal analysis of temperature using smoothing spline ANOVA. *J. Climate* **11** 18–28.

MADDEN, R. A., SHEA, D. J., BRANSTATOR, G. W., TRIBBIA, J. J. and WEBER, R. O. (1993). The effects of imperfect spatial and temporal sampling on estimates of the global mean temperature: Experiments with model data. *J. Climate* **6** 1057–1066.

O'SULLIVAN, F. (1985). Discussion of "Some aspects of the spline smoothing approach to nonparametric regression curve fitting" by Silverman. *J. Roy. Statist. Soc. Ser. B* **47** 39–40.

ROBERTS, G. O. and SAHU, S. K. (1997). Updating schemes, Correlation structure, blocking and parameterization for the Gibbs sampler. *J. Roy. Statist. Soc. Ser. B* **59** 291–317.

STEIN, M. (1990). Uniform asymptotic optimality of linear predictions of a random field using an incorrect second-order structure. *Ann. Statist.* **18** 850–872.

TAPIA, R. A. and THOMPSON, J. R. (1978). *Nonparametric Probability Density Estimation.* Johns Hopkins Univ. Press.

VARGA, R. S. (1962). *Matrix Iterative Analysis*. Prentice-Hall, Englewood Cliffs, NJ.

VINNIKOV, K. YA., GROISMAN, P. YA. and LUGINA, K. M. (1990). Empirical data on contemporary global climate changes (temperature and precipitation). *J. Climate* **3** 662–677.

WAHBA, G. (1981). Spline interpolation and smoothing on the sphere. *SIAM J. Sci. Statist. Comput.* **2** 5–16. [Erratum (1982) **3** 385–386.]

WAHBA, G. (1990). *Spline Models for Observational Data.* SIAM, Philadelphia.

WAHBA, G. and LUO, Z. (1997). Smoothing spline ANOVA fits for very large, nearly regular data sets, with application to historical global climate data. *Ann. Numer. Math.* **4** 579–597.

WAHBA, G., WANG, Y., GU, C., KLEIN, R. and KLEIN, B. (1995). Smoothing spline ANOVA for exponential families, with application to the Wisconsin epidemiological study of diabetic retinopathy. *Ann. Statist.* **23** 1865–1895.

WU, C. F. J. (1983). On the convergence properties of the EM algorithm. *Ann. Statist.* **11** 95–103.

YATES, F. (1933). The analysis of replicated experiments when the field results are incomplete. *Empire J. Experimental Agriculture* **1** 129–142.

YOUNG, D. M. (1971). *Iterative Solution of Large Linear Systems*. Academic Press, New York.

DEPARTMENT OF STATISTICS
PENNSYLVANIA STATE UNIVERSITY
326 THOMAS BUILDING
UNIVERSITY PARK, PENNSYLVANIA 16802-2111
E-MAIL: zhen@stat.psu.edu