

Interpolation of geophysical data using continuous global surfaces

Stephen D. Billings*, Rick K. Beatson†, and Garry N. Newsam**

ABSTRACT

A wide class of interpolation methods, including thin-plate and tension splines, kriging, sinc functions, equivalent-source, and radial basis functions, can be encompassed in a common mathematical framework involving continuous global surfaces (CGSs). The difficulty in applying these techniques to geophysical data sets has been the computational and memory requirements involved in solving the large, dense matrix equations that arise. We outline a three-step process for reducing the computational requirements: (1) replace the direct inversion techniques with iterative methods such as conjugate gradients; (2) use preconditioning to cluster the eigenvalues of the inter-

polation matrix and hence speed convergence; and (3) compute the matrix–vector product required at each iteration with a fast multipole or fast moment method.

We apply the new methodology to a regional gravity compilation with a highly heterogeneous sampling density. The industry standard minimum-curvature algorithms and several scale-dependent CGS methods are unable to adapt to the varying data density without introducing spurious artifacts. In contrast, the thin-plate spline is scale independent and produces an excellent fit. When applied to an aeromagnetic data set with relatively uniform sampling, the thin-plate spline does not significantly improve results over a standard minimum-curvature algorithm.

INTRODUCTION

Geophysical data are often collected with an irregular spatial distribution. For example, in an airborne geophysical survey, data are collected along roughly parallel transects across the area of interest. The sample spacing both in-line and cross-line can vary substantially from inevitable speed and course fluctuations. Added to this irregularity is the tendency to collect the data at a higher density (by an order of magnitude or more for magnetics) along lines than between them. After preliminary processing, current practice usually involves gridding the data—that is, deriving from the scattered data estimates of the quantity of interest on a regular grid. The gridding occurs for two reasons: (1) for visualization and (2) to simplify and speed up subsequent processing operations.

Gridding has an extensive literature, so this paper does not conduct a thorough review [see Foley and Hagen (1994) and references therein]. Rather, we investigate a number of alternative practices to gridding that have developed in the

geostatistical, terrain modeling, and applied mathematics areas and show how they can be unified in a common mathematical framework. These are methods based on fitting continuous global surfaces (CGSs) to the scattered data, and they include the dual formulation of kriging (e.g., Matheron, 1980; Cressie, 1993), tension splines (Mitasova and Mitas, 1993), radial basis functions (Cheney and Light, 1999), thin-plate splines (Wahba, 1990; Hutchinson 1993), sinc interpolation (Shannon, 1949), and the equivalent-source method of Cordell (1992). The close connections between kriging and splines have been explored by Matheron (1980), Dubrule (1984), Wahba (1990), and Hutchinson and Gessler (1994), while the connection between kriging, radial basis functions, and partial differential equations has been examined by Horowitz et al. (1996).

We are predominantly concerned with 2-D applications, but our methodology also applies to higher dimension data. Consider $f : \mathbb{R}^d \rightarrow \mathbb{R}$, a real-valued function of d variables (observed data), that is to be approximated by some surface $s : \mathbb{R}^d \rightarrow \mathbb{R}$, given the values $\{f_n = f(\mathbf{x}_n) : n = 1, \dots, N\}$, where

Manuscript received by the Editor February 28, 2001; revised manuscript received April 2, 2002.

*University of British Columbia, Geophysical Inversion Facility, 2219 Main Mall, Vancouver, British Columbia V6T 1Z4, Canada. E-mail: sbillings@eos.ubc.ca.

†University of Canterbury, Department of Mathematics and Statistics, Private Bag 4800, Christchurch, New Zealand. E-mail: r.beatson@math.canterbury.ac.nz.

**Defence Science and Technology Organisation, Surveillance Systems Division, P.O. Box 1500, Edinburgh, South Australia 5111, Australia. E-mail: garry.newsam@dsto.defence.gov.au.

© 2002 Society of Exploration Geophysicists. All rights reserved.

$\{\mathbf{x}_n : n = 1, \dots, N\}$ is a set of distinct points in \mathbb{R}^d . The surfaces fitted by each of the techniques described in this paper have the form

$$s(\mathbf{x}) = p(\mathbf{x}) + \sum_{n=1}^N \lambda_n \Phi(\mathbf{x} - \mathbf{x}_n), \quad (1)$$

where $p(\mathbf{x})$ is a polynomial of low-degree k , or is not present, λ_n are a set of weights, and Φ is a fixed function from $\mathbb{R}^d \rightarrow \mathbb{R}$. The approximation in equation (1) is required to satisfy the interpolation conditions

$$s(\mathbf{x}_n) = f_n, \quad \text{for } n = 1, \dots, N \quad (2)$$

together with the side conditions

$$\sum_{n=1}^N \lambda_n q(\mathbf{x}_n) = 0, \quad \text{for all } q \in \pi_k^d, \quad (3)$$

where π_k^d is the space of all polynomials of degree at most k in d variables. For each of the techniques considered, the fitting of the weights reduces to solving a linear system whose structure is always just a slight variation on a generic form. We term this general framework the CGS framework.

CGSs are well suited to geophysical applications for several reasons.

- 1) Many different surface-fitting methods can be encompassed within a common algebraic and computational framework.
- 2) When a CGS is used to interpolate a geophysical survey, the resulting surface inherits certain desirable properties from the basic function. For example, using the thin-plate spline produces the smoothest surface (measured in terms of a smoothness functional depending on second-order derivatives) that passes through all of the data points. On the other hand, using a sinc function generates a band-limited surface, which means it has no power above a certain cutoff frequency. Finally, fitting with a kriging semivariogram gives a surface whose values are the best linear unbiased estimates of the true surface that can be obtained from the data.
- 3) The basic functions do not have to be radially symmetric. Thus, anisotropy arising from different sampling densities or the underlying physics can be accommodated within the CGS framework.
- 4) In contrast to most currently used methods, the results of the fit do not depend on the grid size chosen to display the interpolated image.
- 5) A CGS expansion is effectively a continuous model of the data that can be fed into subsequent processing operations, including Fourier transformations and convolutions (Billings and Newsam, 2002). This means there is no need to grid the data prior to this processing, although of course one would generally still evaluate the fitted surface on a grid for visualization purposes.
- 6) Finally, in practice the available data values f_n are almost always in error, i.e., $f_n = f(\mathbf{x}_n) + \varepsilon_n$, where ε_n is nonnegligible in comparison with the variations in the true function $f(\mathbf{x})$. Since it makes no sense to try to interpolate noise, and since doing so often produces a very unstable interpolant, in practice the surface s is usually constructed so that it only approximately interpolates the

data f_n . The quality of the fit to the data is traded off against some other desirable property in the approximation, such as smoothness. Fortunately, such smooth approximations still fall within the CGS framework. Nevertheless, smoothing data brings up a whole new set of issues, so we restrict our paper to exact interpolation and postpone discussion of smoothing to a companion paper (Billings et al., 2002).

The main impediments to applying CGS to geophysical data to date have been the computational cost of solving the matrix equations that arise in determining the surface and the cost of subsequent evaluations of the surface. Both problems arise because the functions Φ that occur in equation (1) are usually globally supported, i.e., they are nonzero at almost all points in the plane. With N data points, direct solution requires $O(N^3)$ arithmetic operations and $O(N^2)$ storage. Furthermore, the marginal cost of a single extra evaluation of the fitted surface using equation (1) is $O(N)$ operations. Consequently, once N gets beyond a few thousand, direct fitting and evaluation become problematic—even on high-end workstations.

Fortunately, recent developments reported by Beatson and Chacko (2000) and Beatson et al. (2001) help us apply the methodology to large problems. There are essentially four key steps in reducing the computational cost of fitting to manageable levels:

- 1) Replace the direct solution methods by suitable iterative solution methods. This decouples the $O(N^3)$ dependence into the product of the number of iterations, which is at most $O(N)$, and a factor $O(N^2)$ for the work required at each iteration. This is principally the formation of a matrix–vector product.
- 2) Decrease the number of iterations required to $O(\log N)$ or $O(1)$ operations by preconditioning the matrix system.
- 3) Reduce the cost of a single matrix–vector product to $O(N \log N)$ or $O(N)$ operations by using fast algorithms.
- 4) Use the same fast algorithms to reduce the marginal cost of a single extra function evaluation to $O(1)$ operations.

We present the general CGS setting and demonstrate its application to geophysics by applying it to some gravity and airborne magnetic data sets. In the first section we consider the problem of scattered data interpolation and discuss why functions of the form of equation (1) with $\Phi(\mathbf{x})$ radially symmetric are so promising. We then consider variational problems and show how the resulting splines fit within the CGS setting. This is followed by a brief description of kriging, which reveals that it too can be encompassed by the CGS framework. Next we consider computing with CGSs. Estimates of operation counts and memory requirements show that conventional direct approaches are impractical for even very small geophysical data sets. Fortunately, recently developed fast evaluation and iterative strategies make possible the application of CGS techniques to surveys with tens of thousands to millions of points. Finally, we apply the method to both a gravity and an aeromagnetic survey.

INTERPOLATION OF SCATTERED DATA

Scattered data-fitting problems in \mathbb{R}^d , $d > 1$, are inherently more difficult than those in \mathbb{R}^1 . In addition to interpolation by CGS, there are several other standard approaches, each

with their own strengths and weaknesses. Perhaps the two most commonly used are interpolation by a fixed set of functions (e.g., polynomials or Fourier series) and interpolation by finite elements. We briefly compare each of these with CGS fitting to motivate the latter's use.

First, interpolation by a fixed set of functions is often unstable and computationally expensive. Consider the situation of trying to interpolate the data values f_n given at distinct points \mathbf{x}_n with a linear combination of functions g_n . This technique requires solving the following linear system for the function coefficients λ_n :

$$\begin{bmatrix} g_1(\mathbf{x}_1) & g_2(\mathbf{x}_1) & \cdots & g_N(\mathbf{x}_1) \\ \vdots & \vdots & \vdots & \vdots \\ g_1(\mathbf{x}_N) & g_2(\mathbf{x}_N) & \vdots & g_N(\mathbf{x}_N) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix}. \quad (4)$$

For irregular distributions of data points, this system is often numerically unstable; moreover, the surfaces produced are often highly oscillatory. The problem is exacerbated when the configurations contain large data-free areas, as often occurs in geophysics. One source of instability is the relatively large number of configurations of points \mathbf{x}_n for which the matrix in equation (4) will be rank deficient. Another is that most bases of fixed functions are constructed to have relatively uniform approximation properties over some region. Thus, they are unable to easily produce interpolants that are well adapted to a locally varying data density. Finally, for most standard basic functions g_n , equation (4) can usually be solved cheaply only when the points \mathbf{x}_n lie on a regular grid.

In contrast, by siting a basic function at each data point, CGSs automatically adjust to varying data density. As a result, the matrices appearing are far less likely to be rank deficient, and the surface usually oscillates less wildly between nodes.

Finite-element approximations are also automatically adaptable to regular grids, rarely suffer from rank deficiency, and are efficient to compute. However, they are perhaps less adaptable than CGSs to data sets with large data-free regions [see Carr et al., (2001) for examples of how radial basis functions extrapolate across data-free regions]. Moreover, finite-element constructions are not natural interpolants in the sense that they minimize some natural measure of smoothness such as the quadratic penalty for the thin-plate spline over a large function space. Of course, they can approximate the solution to such minimization problems by carrying out the minimization over interpolants from a finite-element subspace of the appropriate Beppo-Levi space rather than over all interpolants in the whole space. For these reasons we feel that CGS interpolants are a worthy alternative to finite elements for geophysical interpolation problems. In particular the CGS framework is rich enough to include interpolants defined by a variety of natural formulations of the properties a good interpolant should have.

Radial basis function approximation

Let us temporarily restrict attention to the special case of radial basis functions in which $\Phi(\mathbf{x}) = \phi(\|\mathbf{x}\|)$ is radial, where $\|\cdot\|$ denotes the Euclidean norm. Equation (1) then has the form

$$s(\mathbf{x}) = \sum_{n=1}^N \lambda_n \phi(\|\mathbf{x} - \mathbf{x}_n\|) + \sum_{j=1}^K a_j q_j(\mathbf{x}). \quad (5)$$

Here, $\{q_j\}$ are a basis for the space of polynomials π_k^d and $\{a_j\}$ are the corresponding coefficients. If this expansion is to satisfy the constraints specified in equations (2) and (3), then the coefficients must satisfy a linear system of $N + K$ equations $N + K$ unknowns. This may be written as

$$\begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \quad (6)$$

with $A_{mn} = \phi(\mathbf{x}_m - \mathbf{x}_n)$, $P_{nj} = q_j(\mathbf{x}_n)$, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_N)^T$, $\mathbf{a} = (a_1, a_2, \dots, a_K)^T$, and $\mathbf{f} = (f_1, f_2, \dots, f_N)^T$.

We next briefly review some important properties of the system in equation (6). When we refer to the whole system in equation (6), we use \mathbf{C} for the matrix, $\boldsymbol{\mu}$ for the CGS weights and the polynomial coefficients together, and \mathbf{z} for the data plus the additional zeroes. The matrix in equation (6) is symmetric but in general is not strictly positive definite (i.e., it does not satisfy the conditions $\boldsymbol{\mu}^T \mathbf{C} \boldsymbol{\mu} > 0$ for all nonzero $\boldsymbol{\mu} \in \mathbb{R}^{N+K}$). The solvability of this matrix system depends on the nonsingularity of the the matrix, which holds for a wide choice of functions $\phi(\mathbf{x})$ and polynomial degrees k . One sufficient condition (Micchelli, 1986) consists of mild restrictions on the distribution of the centers together with the basic function ϕ being strictly conditionally positive definite of degree k . That is, $\boldsymbol{\lambda}^T \mathbf{A} \boldsymbol{\lambda} > 0$ for all $\boldsymbol{\lambda} \neq \mathbf{0}$ such that $\mathbf{P}^T \boldsymbol{\lambda} = \mathbf{0}$. Here the restriction on the centers is that no nontrivial polynomial of degree k vanishes at all points \mathbf{x}_n [that is, $q \in \pi_k^d$ and $q(\mathbf{x}_n) = 0$, $n = 0, \dots, N$ together imply $q(\mathbf{x}) = 0$ for all \mathbf{x}]. This condition on the geometry of the nodes is almost always satisfied. For example, for polynomials of degree 1 and \mathbb{R}^2 , it means there is no single line passing through all of the nodes. Cheney and Light (1999) are a good reference for these and other properties of radial basis functions.

Finally, 3-D perspective plots of some of the particular basic functions. $\Phi(\mathbf{x})$ considered in this paper are shown in Figure 1. They include thin-plate spline, $\|\mathbf{x}\|^2 \log(\|\mathbf{x}\|)$; Gaussian, $\exp(-c\|\mathbf{x}\|^2)$; multiquadric, $(\|\mathbf{x}\|^2 + c^2)^{+1/2}$; inverse multiquadric, $(\|\mathbf{x}\|^2 + c^2)^{-1/2}$; tension spline [see equation (10)]; and sinc function, $\text{sinc}(cx) \text{sinc}(cy)$, where c is a positive constant. The plots show that several basic functions (splines and multiquadric) are unbounded as $\|\mathbf{x}\|$ approaches infinity. Hence, one might expect that these Φ s are bad approximators. However, suitable finite combinations of shifts of Φ can be shown to form highly peaked kernels with rapid decay at infinity, so the growth does not preclude good approximation properties. Indeed, the variational characterizations discussed below show that CGSs have excellent approximation properties.

GENERALITY OF CGSs

We now return to the more general form of interpolation specified in equations (1)–(3) with a (possibly) nonradial basic function. Many surface-fitting methods with their associated basic functions, including thin-plate splines, tension splines, and kriging semivariograms, are defined naturally as the solution of certain variational problems. These arise through defining a reasonable penalty functional $J(s)$ (almost always a quadratic function of s) and then choosing the fitted surface to be the solution of the constrained optimization problem:

$$\min J(s) \quad \text{subject to} \quad s(\mathbf{x}_n) = f_n : n = 1, \dots, N. \quad (7)$$

For quadratic functionals, this problem has a linear solution that can usually be shown to be of CGS form. Let's review some of the more widely used examples of interpolation methods defined in this way.

Thin-plate splines

As already noted, choosing $J(s)$ to emphasize smoothness gives rise to some particularly useful and well-performing families of spline interpolants. Duchon (1976) characterizes the surfaces associated with a general family of smoothness measures, which in two dimensions have the form

$$J(s) = \int_{\mathbb{R}^2} \sum_{i=0}^{m+1} \binom{m+1}{i} \left(\frac{\partial^{m+1} S}{\partial x^i \partial y^{m+1-i}} \right)^2 \partial x \partial y. \quad (8)$$

In particular, he shows that the corresponding basic function is $\phi(\|\mathbf{x}\|) = \|\mathbf{x}\|^{2m} \log(\|\mathbf{x}\|)$. For $m = 1$ the resulting interpolant is the usual thin-plate spline, so called because it is a mathematical model of the physical surface that a thin metal plate would adopt if it were appropriately constrained to pass through the data points. The presence of polynomial terms in smoothing splines is a natural consequence of the form of the smoothness constraint in equation (8). For example, with $m = 1$ the derivatives annihilate constants and linear polynomials, so these can be included without affecting the smoothness measure.

Smoothing splines have been used extensively for interpolating and smoothing data [see Wahba (1990) and references therein]. Geophysicists are familiar with equation (8) for $m = 1$ because it also occurs in the discrete minimum curvature algorithm of Briggs (1974) (referred to as Minq) that is widely used for gridding. Minq, in fact, constructs a discrete interpolant based on minimizing a finite difference approximation to equation (8). Thus, it generates a discrete approximation to the true continuous thin-plate spline interpolant.

Tension splines

When fitting a thin-plate spline to 2-D data that contain regions with rapid change of gradients, the thin-plate's stiffness may result in overshoots. Smith and Wessel (1990) include a tension parameter in the minimum curvature equations that eliminate these overshoot problems. In a variational setting Mitasova and Mitas (1993) modify equation (8) so it includes additional derivative constraints:

$$J(s) = \sum_{k=0}^{\infty} \sum_{\mathbf{a}:|\mathbf{a}|=k} B_{\mathbf{a}} \int_{\mathbb{R}^2} \left(\frac{\partial^{|\mathbf{a}|} s}{\partial x^{a_x} \partial y^{a_y}} \right)^2 \partial x \partial y, \quad (9)$$

where $\mathbf{a} = (a_x, a_y)$, $|\mathbf{a}| = a_x + a_y$, and $B_{\mathbf{a}} = |\mathbf{a}|! \delta^{-2|\mathbf{a}|} / [a_x! a_y! (|\mathbf{a}| - 1)!]$ if $\mathbf{a} \neq 0$ and $B_{\mathbf{a}} = 0$ otherwise. The parameter δ is known as a generalized tension parameter, and it controls the shape of the interpolated surface. When δ is small, the higher order derivatives dominate and the resulting surface resembles a thin-plate spline. Conversely, when δ is large, the lower order derivatives dominate and the surface resembles a thin membrane stretched to fit the data points.

Mitasova and Mitas (1993) show that the function that minimizes $J(s)$ subject to interpolating the data can be cast as a CGS expansion with the basic function

$$\phi(\|\mathbf{x}\|) = 2 \ln\left(\frac{\delta \|\mathbf{x}\|}{2}\right) + E_1\left(\frac{\delta^2 \|\mathbf{x}\|^2}{4}\right) + \gamma, \quad (10)$$

where $E_1(\cdot)$ is the exponential integral function of the first kind and $\gamma = 0.5772 \dots$ is Euler's constant. In the limit of $\|\mathbf{x}\| \rightarrow 0$, the logarithmic term and Euler's constant cancel with the exponential integral so that the definition of ϕ can be extended by continuity to $\phi(0) = 0$.

Kriging

Kriging is a geostatistical technique that has its roots in the estimation of ore grade at an arbitrary location from discrete samples in an orebody (e.g., Krige, 1951; Matheron, 1963). In kriging it is assumed that the ore grade $f(\mathbf{x})$ is a realization of a stochastic process, which in universal kriging (also called kriging with a trend) typically takes the form

$$f(\mathbf{x}) = p(\mathbf{x}) + z(\mathbf{x}), \quad (11)$$

where $z(\mathbf{x})$ is assumed to be a zero-mean, second-order stationary stochastic process, while $p(\mathbf{x}_n)$ represents low-order polynomial drift in the data. The second-order stationarity assumption implies that the second moment of $z(\mathbf{x} + \mathbf{h}) - z(\mathbf{x})$ does not depend on absolute position \mathbf{x} but only on separation \mathbf{h} . That is,

$$\text{var}[z(\mathbf{x} + \mathbf{h}) - z(\mathbf{x})] = 2V(\mathbf{h}), \quad (12)$$

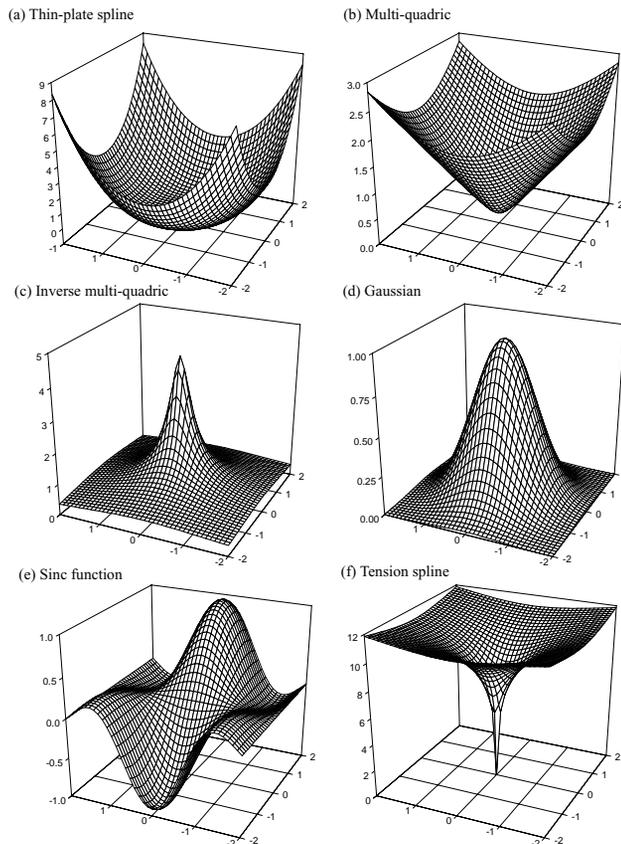


FIG. 1. (a) Thin-plate spline. (b) Multiquadric ($c = 0.1$). (c) Inverse multiquadric ($c = 0.1$). (d) Gaussian ($c = 1$). (e) Sinc function and ($\Delta x = \Delta y = 0.5$). (f) tension spline ($\phi = 1000$).

where $\text{var}[z] = E[z^2] - E[z]^2$ is the variance, $E[z]$ is expected value of z , and $V(\mathbf{h})$ the semivariogram [$2V(\mathbf{h})$ is the variogram]. This definition of the variogram allows anisotropy, although in practice $V(\mathbf{h})$ is often assumed radial.

Kriging now calls on this statistical description of the problem to provide a well-defined estimate of the unknown ore grade at an arbitrary location \mathbf{x} from the samples $f(\mathbf{x}_n)$ collected at the discrete set of points \mathbf{x}_n . The kriging estimate $s(\mathbf{x})$ is taken to be the best linear unbiased estimate of the random variable $f(\mathbf{x})$ obtainable from a weighted linear sum of the sample data, i.e.,

$$s(\mathbf{x}) = \sum_{n=1}^N \lambda_n(\mathbf{x}) f(\mathbf{x}_n). \quad (13)$$

This is a linear inverse problem where the weights $\lambda_n(\mathbf{x})$ must be chosen to ensure the estimate is unbiased and optimal. The solution is unbiased only if the expectation is zero, i.e., $E[s(\mathbf{x}) - f(\mathbf{x})] = 0$, which is equivalent to the interpolation condition satisfied by splines. The optimal solution is now found by minimizing the penalty function:

$$J(s) = \text{Var}[s(\mathbf{x}) - f(\mathbf{x})]. \quad (14)$$

The $J(s)$ here is the equivalent in kriging of the penalty functions for thin-plate and tension splines given in equations (6) and (7), respectively.

The formal equivalence between splines and kriging has been known for some time (Matheron, 1980) and has been examined extensively in the literature (e.g., Dubrule, 1984; Wahba, 1990; Hutchinson and Gessler, 1994). Matheron (1980) shows that the optimality condition in equation (14) implies that the universal kriging estimate $s(\mathbf{x})$ is given by

$$s(\mathbf{x}) = p(\mathbf{x}) + \sum_{n=1}^N \lambda_n V(\mathbf{x} - \mathbf{x}_n), \quad (15)$$

where the polynomial $p(\mathbf{x})$ corresponds to the order of drift of the data. This is analogous to the CGS expansion in equation (1), with the basic function $\Phi(\mathbf{x})$ replaced by the semivariogram $V(\mathbf{x})$. The requirement that the solution be unbiased leads to an equivalent set of conditions on the weights as equation (3) for CGSs. Solution of the kriging equations is then equivalent to equation (6) for CGSs, which in the kriging literature is referred to as the dual form of kriging. The semivariograms are always selected to satisfy a strictly conditional negative definite constraint, which is sufficient to ensure that the linear system has a unique solution (Myers, 1988).

The dual form of kriging is often presented with a generalized covariance function in place of the semivariogram (e.g., Matheron, 1980). This makes the matrix system positive definite instead of negative definite. The second-order stationary random field model of equation (11) is usually replaced with a more general model based on intrinsic random functions of order k , or IRF- k (e.g., Matheron, 1980). The basic idea is to linearly filter the drift terms from the data, for these can cause bias in the estimation of the semivariogram by traditional methods. An IRF-0 is equivalent to the stationarity condition in equation (11), with the generalized covariance function $K(\mathbf{x})$ and the semivariogram related by $K(\mathbf{x}) = -V(\mathbf{x})$. When $k > 0$, the relationship between the variogram and the generalized

covariance function is more complicated (see, e.g., Cressie, 1993).

Often the semivariogram is defined as

$$V(\mathbf{x}) = c_o[1 - \delta(\|\mathbf{x}\|)] + V_o(\mathbf{x}), \quad (16)$$

where $V_o(\mathbf{x})$ is continuous, $\delta(\|\mathbf{x}\|)$ is defined by $\delta(\|\mathbf{x}\|) = 0$ if $\|\mathbf{x}\| \neq 0$ and $\delta(0) = 1$, and c_o is the so-called nugget effect. A nonzero nugget variance arises from both random variation as a result of measurement error and fine-scale variability in the surface not modeled by the semivariogram. Dubrule's (1984) equation (3) gives a suitable form of the dual kriging equations for this situation. Namely, he fits with covariance $-V_o(\mathbf{x})$ and adds a smoothing term c_o to the diagonal entries of an appropriate matrix. With a zero nugget the kriged surface gives an exact interpolation, while including a nonzero nugget smooths the data rather than interpolates it.

Interpolating potential fields

Cordell (1992) has developed an equivalent source method to interpolate point measurements of a potential field. Given data at points \mathbf{x}_n on the surface, this involves postulating the existence of point sources of unknown strengths λ_n at each of these locations at depths z_n . The strengths are then determined so the resulting potential field interpolates the available data. When all equivalent sources are placed at the same depth $z_n \equiv z$, the method can be seen to be a 2-D CGS interpolation with the following kernel:

$$\Phi(x, y) = \frac{1}{(x^2 + y^2 + z^2)^{1/2}}. \quad (17)$$

This basic function is known as an inverse multiquadric in the radial basis function literature. If the sources are at differing depths, then the interpolation problem involves three dimensions.

Interpolation on differing grids

So far we have chosen the centers for the CGS expansion to be the nodes where samples are available. This is a natural choice, but it is not essential; our general ideas can still be applied to solve the more general interpolation problem of choosing a polynomial and weights λ_n so that

$$f_m = p(\mathbf{x}_m) + \sum_{n=1}^N \lambda_n \phi(\mathbf{x}_m - \mathbf{y}_n), \quad (18)$$

where the N centers \mathbf{y}_n differ from the N sampling points \mathbf{x}_m . An example of this occurs when we want to construct a band-limited interpolant. A function is band-limited if it has no frequency content above a certain cutoff frequency u_c . Shannon's sampling theorem (Shannon, 1949) states that it is possible to reconstruct a 2-D band-limited function from its samples $f_{mn} = f(\mathbf{x}_{mn})$ at the points $\mathbf{x}_{mn} = (m\Delta, n\Delta)$, where $\Delta = 1/2u_c$, through the interpolant

$$f(x, y) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} f(n\Delta x, m\Delta y) \text{sinc}\left(\frac{x - n\Delta x}{\Delta x}\right) \times \text{sinc}\left(\frac{y - m\Delta y}{\Delta y}\right). \quad (19)$$

In practice samples are often not available on the regular grid x_{mn} , so unknown function values f_{mn} at these points must be solved for as though they were unknown weights in a CGS expansion with the basic function

$$\phi(x, y) = \text{sinc}\left(\frac{x}{\Delta}\right)\text{sinc}\left(\frac{y}{\Delta}\right). \quad (20)$$

This now becomes a special instance of the general interpolation problem in equation (18) (as long as the number of samples is taken to be the same as the number of gridpoints).

When the nodes \mathbf{x}_m differ from the centers \mathbf{y}_n , the linear system produced by equation (18) is no longer symmetric. In this case some of the iterative methods discussed in the next section can no longer be used. Nevertheless, the basic idea outlined there still holds: by combining iterative methods with appropriate preconditioners and approximations to the basic function, the system can still be solved in $O(N)$ operations.

SOLVING THE INTERPOLATION EQUATIONS

The basic functions forming most CGSs (e.g., standard splines and semivariograms) are zero only at a few exceptional points; most geophysical data are collected with irregular spatial sampling. So the matrix in equation (6) is usually full (i.e., has relatively few zero entries) rather than sparse and has none of the special structures (such as being banded) that are normally exploited in calculating fast solutions of linear systems. Thus, solution of the interpolation equations by direct methods or even simple iterative methods requires $O(N^3)$ operations and $O(N^2)$ storage.

The $O(N^3)$ scaling means that CGS-type interpolations have generally been restricted to at most a few thousand points and have often used a restricted number of nodes to further reduce the computational load (e.g., Bates and Wahba, 1982; Hutchinson, 1993). Consider the implications of this scaling as computer hardware improves. Assume that we can solve a problem of size N in some acceptable timeframe t on one computer and then obtain access to a computer that is an order of magnitude (ten times) faster. In the same time on this computer, we can solve a problem that is $10^{1/3} \sim 2.2$ times bigger. That is, for each tenfold increase in computer speed, we get slightly more than a doubling of the size of the problem that can be solved in an acceptable timeframe. Direct methods require that at least half of the matrix will need to be stored (because it is symmetric) in memory. For a problem with 10 000 points stored in double precision, this requires 382 Mbytes of memory. For a 20 000-point problem, the memory requirements escalate to 1.49 Gbytes. And for 40 000 points, 6 Gbytes are required. Clearly, significant improvements in computational scaling and memory requirements need to be made before the technique can be feasible for large problems.

Brief overview of iterative methods

The rough calculations in the last paragraph show that direct methods are unfeasible for N greater than a few thousand, even on high-end workstations. The key to reducing arithmetic operation and memory requirements is the use of iterative methods. Solution by an iterative method decomposes the operation count into the product of two factors: the number of itera-

tions times the cost per iteration. The former is determined by the rate of convergence of the iterative method, while in most methods the latter is determined by the cost of a matrix–vector multiplication.

To give some insight into how both these costs can be reduced, we briefly review the application of iterative methods to the solution of equation (6). We can write this as $\mathbf{C}\boldsymbol{\mu} = \mathbf{z}$, where

$$\mathbf{C} = \begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}, \quad \boldsymbol{\mu} = \begin{bmatrix} \lambda \\ \mathbf{a} \end{bmatrix} \quad \text{and} \quad \mathbf{z} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}. \quad (21)$$

Classical iterative methods, such as those of Jacobi, Gauss-Seidel, or Richardson, produce iterates of the form

$$\boldsymbol{\mu}_m = \mathbf{B}\boldsymbol{\mu}_{m-1} + \mathbf{b}, \quad (22)$$

where the matrix \mathbf{B} and vector \mathbf{b} are derived in some fashion from \mathbf{C} and \mathbf{z} (e.g., by matrix splitting).

In fitting CGSs, however, these iterative solvers are outperformed by Krylov subspace methods; a review of these methods and their convergence properties can be found in Kelley (1995). At the m th iteration a Krylov subspace method draws its next update $\boldsymbol{\mu}_m$ from the affine space $\mathbf{G}_m = \boldsymbol{\mu}_0 + \mathbf{K}_m$, where \mathbf{K}_m is the Krylov subspace:

$$\mathbf{K}_m = \text{span}\{\mathbf{r}_0, \mathbf{C}\mathbf{r}_0, \dots, \mathbf{C}^{m-1}\mathbf{r}_0\}. \quad (23)$$

Here, $\mathbf{r}_0 = \mathbf{z} - \mathbf{C}\boldsymbol{\mu}_0$ is the residual left by the initial guess $\boldsymbol{\mu}_0$ (often $\boldsymbol{\mu}_0$ is just taken to be zero, so that $\mathbf{G}_m = \mathbf{K}_m$). The two best-known Krylov subspace methods are the conjugate gradient method (Hestenes and Steifel, 1952) and generalized minimization of residuals (GMRES) (Saad and Schultz, 1986). In conjugate gradients $\boldsymbol{\mu}_m$ is the solution of the minimization problem

$$\min_{\boldsymbol{\mu} \in \mathbf{G}_m} \frac{1}{2} \boldsymbol{\mu}^T \mathbf{C}\boldsymbol{\mu} - \boldsymbol{\mu}^T \mathbf{z}, \quad (24)$$

while in GMRES $\boldsymbol{\mu}_m$ is instead chosen to solve

$$\min_{\boldsymbol{\mu} \in \mathbf{G}_m} \|\mathbf{C}\boldsymbol{\mu} - \mathbf{z}\|^2. \quad (25)$$

We shall consider some particular issues in applying these two methods to CGS interpolation in slightly more detail later. For now, however, we focus on the key features that determine their operation count: the number of iterations required to reach an acceptable solution and the work required per iteration

Convergence rates for Krylov subspace methods

We begin with the classic bounds (see, e.g., Kelley, 1995) on convergence rates for the simple iterative method in equation (22). If \mathbf{B} is symmetric and positive definite, then

$$\|\boldsymbol{\mu}_m - \boldsymbol{\mu}^*\| \leq \|\boldsymbol{\mu}_0 - \boldsymbol{\mu}^*\| v_{\max}(\mathbf{B})^m \frac{v_{\max}(\mathbf{B})}{v_{\min}(\mathbf{B})}, \quad (26)$$

where $\boldsymbol{\mu}^*$ is the exact solution and $v_{\max}(\mathbf{B})$ and $v_{\min}(\mathbf{B})$ are the largest and smallest eigenvalues of \mathbf{B} . This algorithm does not converge in a finite number of steps; rather, the error bound reduces by a constant factor at each step.

Clearly the bound depends heavily on the distribution of the eigenvalues of \mathbf{B} , and \mathbf{B} is usually chosen to have a good

distribution. One strategy for achieving this is to precondition the system $\mathbf{C}\boldsymbol{\mu} = \mathbf{z}$ by replacing it by the system

$$(\mathbf{CD})\boldsymbol{\gamma} = \mathbf{z} \quad \text{and} \quad \boldsymbol{\mu} = \mathbf{D}\boldsymbol{\gamma}, \quad (27)$$

where \mathbf{D} is an approximation to \mathbf{C}^{-1} that can be easily computed. This type of strategy is known as right preconditioning. The design of good preconditioners is almost as much an art as a science (see, e.g., Dyn et al., 1986; Beatson et al., 2001). In particular, the price paid for ensuring can \mathbf{D} can be easily computed is that while most of the eigenvalues of \mathbf{CD} are close to one, there are always a few outliers. This limits the effectiveness of preconditioning for accelerating classical iterative schemes but is not a problem for Krylov subspace methods.

To see how preconditioning combined with Krylov subspace iteration rapidly accelerates convergence, we first note that the dimension of the subspaces \mathbf{K}_m increases by one with each iteration. Thus, by $N + K$ iterations, \mathbf{K}_m will eventually span the full space of all possible solutions and, in contrast to classical iterative techniques, the methods will, in theory, converge in at most $N + K$ iterations. To see how preconditioning reduces this to $O(1)$, we note the following bound for the conjugate gradient method (see, e.g., Kelley, 1995). After m iterations the error in the solution is bounded by

$$\|\boldsymbol{\mu}_m - \boldsymbol{\mu}^*\|_C \leq \|\boldsymbol{\mu}_0 - \boldsymbol{\mu}^*\|_C \min_{p \in \pi_m} \max_{v \in \sigma(\mathbf{C})} |p(v)|, \quad (28)$$

where $\boldsymbol{\mu}^*$ is the exact solution $\|\boldsymbol{\mu}\|_C = \sqrt{\boldsymbol{\mu}^T \mathbf{C} \boldsymbol{\mu}}$ and where $\sigma(\mathbf{C})$ is the set of eigenvalues of \mathbf{C} . If the eigenvalues v_m of \mathbf{C} are clustered so there exists an M such that $v_m \sim 1$ for all $m > M$, then choosing the polynomials p_m to have the form

$$p_m(v) = \frac{(v_1 - v)(v_1 - v) \cdots (v_M - v)(1 - v)^{m-M}}{v_1 v_2 \cdots v_M} \quad (29)$$

establishes an upper bound for equation (28). In practice observed performance usually follows this bound: in the early steps each iteration solves for a component of the solution corresponding to one of the outlying eigenvalues. Once the outlying components have been identified, just a few more iterations suffice to refine the solution to the accuracy desired. A similar bound and result hold for GMRES.

Good preconditioners typically produce matrices \mathbf{CB} in which the number M of outlying eigenvalues is a constant size independent of the dimensions of \mathbf{C} . Moreover, the number of iterations required beyond M to reach a sufficiently accurate solution on the clustered eigenvalues is also independent of the problem size. Thus, the number of iterations that Krylov subspace methods take to converge is independent of the problem size.

To illustrate the clustering of eigenvalues that results from applying the domain decomposition preconditioner, we analyzed a small synthetic problem consisting of a 20×20 uniform grid in the unit square. Since the bounds on convergence of GMRES depend on eigenvalue clustering, we plot $1 + \text{relative radius}$ versus index where the relative radius is $|\text{eigenvalue} - \text{central value}| / |\text{central value}|$. The central value is taken as 1 for the preconditioned matrix and as the median eigenvalue for the unpreconditioned matrix. The semilog plot in Figure 2 shows that the eigenvalues of the unpreconditioned matrix are smeared across several orders of magnitude.

Hence, for this matrix GMRES can be expected to converge very slowly. In contrast, the eigenvalues of the preconditioned matrix are clustered about 1, with the farthest away being at a distance less than 0.18 from 1. Hence, the known bounds on the convergence of GMRES guarantee that GMRES iteration with this matrix will converge extremely quickly.

Matrix–vector products

To generate the next basis vector of the Krylov subspace \mathbf{K}_m , the matrix–vector product $\mathbf{C}\mathbf{r}_m$ must be computed at each step. Computing this directly requires $O(N^2)$ operations, while the marginal cost of computing the approximation $s(\mathbf{x})$ at a single point by direct evaluation of equation (1) is $O(N)$ operations. In addition, if the matrix \mathbf{C} is pre-computed and stored, the memory requirements also scale as $O(N^2)$. To get the total cost of an iterative method down to $O(N)$ operations, these costs must be reduced to $O(N)$ operations.

Fortunately, recent work (e.g., Beatson and Newsam, 1998) has shown that for almost all of the standard basic functions $\phi(\mathbf{x})$, it is possible to compute the product $\mathbf{A}\boldsymbol{\lambda}$ [equation (6)] in $O(N)$ operations. These methods were first developed for functions ϕ that were potentials in problems such as simulations of the motion of millions of stars under their own gravitational fields. Where \mathbf{x} is well separated from a cluster of points $\{\mathbf{x}_n : n = 1, \dots, N\}$, the asymptotic expansions available for potential functions can be used to accurately approximate far-field contributions of the form

$$\sum_{n=1}^N \lambda_n \phi(\mathbf{x} - \mathbf{x}_n) \quad (30)$$

as the sum of a small number of standard functions. Clever use of tree structures and translation formulas lets us group clusters into larger clusters and merge the associated approximations, with the ultimate result of reducing the operation count for

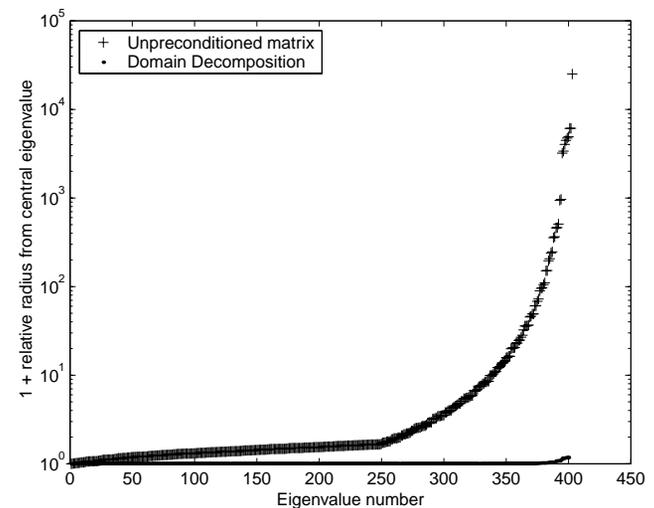


FIG. 2. Semilog plot of $|\text{eigenvalue} - \text{central value}| / |\text{central value}|$ for a test 20×20 -D grid covering the unit square for the unpreconditioned and domain decomposition preconditioned matrices. The central value is taken as 1 for the preconditioned matrix and as the median eigenvalue ($2.37\text{e-}3$) for the unpreconditioned matrix.

computing a very accurate approximation of $\mathbf{A}\boldsymbol{\lambda}$ in $O(N \log N)$ or even $O(N)$ operations.

This approach culminates in the fast multipole method described in Greengard and Rokhlin (1987). The method can be extended to efficiently evaluate CGS sums for many basic functions ϕ that are not, on the face of it, potential functions. For example, Beatson and Newsam (1992) show that the fast multiple method can be used to evaluate sums of thin-plate splines. Unfortunately, for each new ϕ that has a far-field expansion, the formulas and error bounds used to set up the expansions and translations underlying the method need to be rederived, requiring substantial recoding. Moreover, many basic functions do not have suitable asymptotic expansions.

To overcome these problems Beatson and Newsam (1998) evolve the original fast multipole method into a fast moment method (see also Beatson and Chacko, 2000). This approach can easily accommodate changes of the basic function ϕ . Using it with a new ϕ usually requires coding only a one- or two-line function for the slow evaluation of ϕ . The method only requires that ϕ be smooth away from the origin. All of the basic functions presented here (including tension splines) satisfy this condition, along with almost all of the variograms and semivariograms in the kriging literature (see, e.g., Journel and Huijbregts, 1978; the few exceptions are those specifically constructed to have finite support).

Miscellaneous issues

The above discussion indicates how iterative techniques coupled with fast multipole and moment methods can make seemingly intractable calculations. In the interest of clarity, our discussion has necessarily glossed over some areas. We shall briefly touch on one such issue in this section—whether to use conjugate gradients or GMRES.

Most Krylov subspace methods rest on the construction of an orthogonal basis for the subspace \mathbf{K}_m out of the basis vectors $\mathbf{C}\mathbf{r}_m$. If \mathbf{C} is symmetric and positive definite (SPD) then updating \mathbf{K}_m and the approximate solution requires only the new basis vector $\mathbf{C}\mathbf{r}_m$ and the current approximate solution. Thus, generating updates is particularly simple, and there is no need to store a complete basis for \mathbf{K}_m . The resulting update expressions define the conjugate gradients method and make it the most attractive of the Krylov subspace methods.

In most cases the matrix \mathbf{C} will not be SPD; but as we shall presently see, it can be converted into an SPD form. Recall from equation (21) that \mathbf{C} is actually composed of the matrices \mathbf{A} and \mathbf{P} . The discussion following equation (6) establishes that \mathbf{A} is always conditionally SPD, that is, $\boldsymbol{\lambda}^T \mathbf{A}\boldsymbol{\lambda} > 0$ for all $\boldsymbol{\lambda} \neq \mathbf{0}$ such that $\mathbf{P}^T \boldsymbol{\lambda} = \mathbf{0}$. With this constraint it is always possible to find an orthogonal basis \mathbf{Q} such that $\mathbf{P}^T \boldsymbol{\lambda} = \mathbf{0} \Leftrightarrow \boldsymbol{\lambda} = \mathbf{Q}\boldsymbol{\mu}'$, where $\boldsymbol{\mu}' \in \mathbb{R}^{N-K}$, with K the dimension of the polynomial subspace. This idea can be traced to Sibson and Stone (1991) with an extended treatment in Beatson et al. (2001). The matrix system of equation (6) can then be converted to the strictly SPD form $\mathbf{C}'\boldsymbol{\mu}' = \mathbf{z}'$, with $\mathbf{C}' = \mathbf{Q}^T \mathbf{A}\mathbf{Q}$ and $\mathbf{z}' = \mathbf{Q}\mathbf{z}$. Furthermore, once $\boldsymbol{\mu}'$ and hence $\boldsymbol{\lambda}$ are found, the polynomial coefficients $\boldsymbol{\alpha}$ can be obtained by finding the polynomial from π_k^d interpolating to the residual function $\mathbf{r} = \mathbf{f} - \mathbf{A}\boldsymbol{\lambda}$ at K points.

It is always possible to convert CGS equations to a strictly SPD form. However, when preconditioners such as those of Beatson et al. (2001) are used, the matrix system may no longer

be strictly SPD, and conjugate gradients must be replaced by GMRES. To update the Krylov subspace, GMRES requires an orthogonal basis for \mathbf{K}_m . For m iterations, the storage requirements are $O(mN)$ and the operation count for this extension, excluding the cost of the matrix–vector multiplication, grows as $O(m^2N)$. The preconditioners implemented in Beatson et al. (2001) cluster most of the eigenvalues; hence, the number of iterations required for convergence will be small. Thus, the storage and arithmetic overheads are unlikely to be of concern.

APPLICATION TO GEOPHYSICAL DATA

We now present examples of the application of thin-plate and tension splines, multiquadrics, and the equivalent-source technique to some gravity and magnetic data. No examples of kriging or sinc interpolation are presented because they concern smooth fitting rather than interpolation. Examples of these fits are presented in Billings et al. (2002).

Broken Hill gravity survey

Gravity data interpretation often involves compiling several different data sets with widely differing observation spacing. Traditional gridding techniques such as minimum curvature can have difficulty in creating a fit that achieves a good compromise between high-frequency response in densely sampled areas and avoidance of high-frequency noise or ringing in areas of sparse coverage. Murray (1998) recognized this and developed a multiple-pass minimum curvature (MPMinq) algorithm that attempts to create a good fit regardless of the data density. Here, we compare our thin-plate spline fit to Murray's results and to a standard Minq algorithm. Since the thin-plate spline is the correct continuous minimum curvature surface, any deviations from it reflect the inaccuracies of discrete gridding and the artifacts it can introduce.

The data we chose for comparison consist of 55 805 gravity observations in the Broken Hill area, New South Wales, Australia. The data set covers an area of approximately 3.8° by 3.8° , with a highly heterogeneous sampling density; observation spacing varies from 25 m to 7 km (Figure 3a). The thin-plate spline fit required 14 iterations and took about 11 minutes on a generic Intel Celeron processor operating at 450 MHz. We evaluated the surface on a constant grid with a spacing of 15 s in both the north–south and east–west directions, as this was the spacing of the MPMinq grid available. The evaluation required a further 4 minutes. We also used the Minq algorithm in Geosoft to grid the data, with the tolerance parameter set to 0.01% of the data range.

The grids for the thin-plate spline and the MPMinq algorithms are shown in Figures 3c and 3d as sun-shaded images with sun inclinations and declinations of 45° (the Minq result is not shown). The thin-plate spline grid is slightly smoother than the MPMinq grid and has some obvious differences in the areas with sparse data coverage (e.g., just to the east and north of 141°E , 31°S). These differences are further emphasized in a profile across the grid (Figure 4) and in a residual image (Figure 3b) shown with a linear color stretch between -20 and $20 \mu\text{m/s}^2$. Positive residuals larger than this range are white, while negative residuals lying outside the range are black. Almost all of the regions with very large residuals correspond

to areas with poor data coverage. Look in particular at the poor fit in the profile plot between 141°E and 141.5°E. These results show that the MMPMinq algorithm is a poor approximation to the true thin-plate spline surface in areas of sparse data coverage.

The profile plot also shows that a standard Minq algorithm has difficulty in approximating the true thin-plate spline surface. Notice that the MPMinq and Minq results can also deviate significantly from the thin-plate spline fit in areas with very dense coverage (look at the profile plot near 139°E). This is because of the way these algorithms handle multiple values appearing in a single grid cell. Last, the MPMinq does improve the fit over the Minq algorithm, with the residuals for the former having a range of -120 to $105 \mu\text{m/s}^2$ and a standard deviation of $7.1 \mu\text{m/s}^2$ compared with -231 to $238 \mu\text{m/s}^2$ and a standard deviation of $7.9 \mu\text{m/s}^2$ for the latter.

Next, we investigated how successful some of the other CGS interpolation techniques would be on the same data set. We zoomed in on the region shown in Figure 5a because it contained both sparsely and densely sampled regions and fit surfaces with the equivalent-source technique, a tension spline, and a multiquadric. For each of the methods, we first did several trial fits to a small subsection of the data before coming up with estimates of suitable scaling constants. This led us to choose a scaling constant of 2.5 minutes for both the multiquadric and equivalent-source methods and a tension parameter of 50 for the tension spline. A suitable scaling parameter was one that did not cause oscillations in the fitted surface—something that is particularly evident in a spline surface when using inappropriate tension. In fitting the surfaces we did not allow the iterative methods to converge fully for reasons that will become apparent.

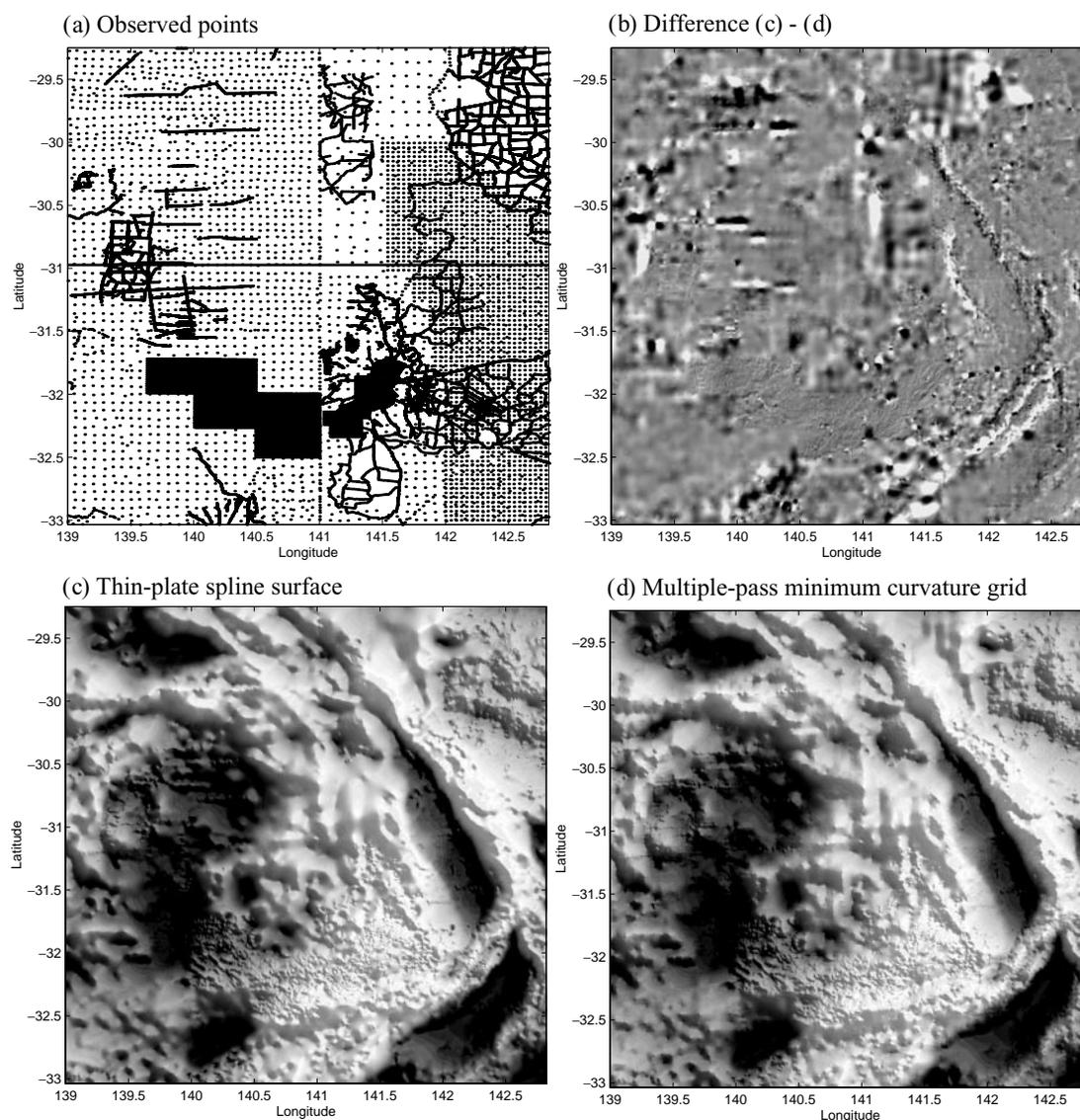


FIG. 3. Australian Geological Survey Organization gravity data set over Broken Hill. (a) Location of observations. (b) Difference between the thin-plate spline and multiple-pass minimum curvature grid. (c) Calculated thin-plate spline surface. (d) Multiple-pass minimum curvature grid.

Contour plots of the different surfaces are shown in Figure 5 (5b, equivalent source; 5d, tension spline; 5e, multiquadric). The contours are fairly similar in the regions with low and medium sampling densities, with the thin-plate spline tending to give a smoother set of contours. In the densely sampled regions (top-right corner and slightly to the left of bottom center) all three of the new surfaces have many bull's-eye-shaped artifacts. At first we thought these must have been concentrated around noisy data in these regions. However, all of these artifacts lie away from data locations and are the result of data overfitting. To show this, we recomputed the multiquadric surface and this time stopped our GMRES iteration early while the residuals were still quite high. The resulting contour plot is shown in Figure 5f. The bull's-eye artifacts have disappeared, and the surface appears to mimic the thin-plate spline quite closely. This tendency of fitting methods to estimate the smoother parts of the surface in the early iterations and the rougher parts later on is a general feature of the GMRES and conjugate gradient iterative methods (e.g., Sibson and Stone, 1991).

The overfitting problems revealed in Figure 5 result from the scaling factors required by the equivalent-source, tension-spline, and multiquadric basic functions. Choosing the scaling factor by trial and error in a small subsection of the survey as we did will result in a scaling factor that is appropriate for that particular data density. When the data density varies by orders of magnitude from this value, the scaling factor will not be appropriate in all areas and artifacts will be introduced. The only way to prevent these artifacts is to stop the iterative process early, which creates a smooth fit. However, when we do this we are not taking advantage of the high-frequency sampling in some parts of the survey area. This implies that these interpolation methods cannot meet our stated aim of achieving a good compromise between fitting high frequencies in the data and avoiding ringing and high-frequency noise in the sparsely sampled areas. The thin-plate spline does not suffer from these disadvantages because it is scale independent. Even when the data points are very dense, the thin-plate spline will be free of high-frequency artifacts (as long as the data are essentially noise free).

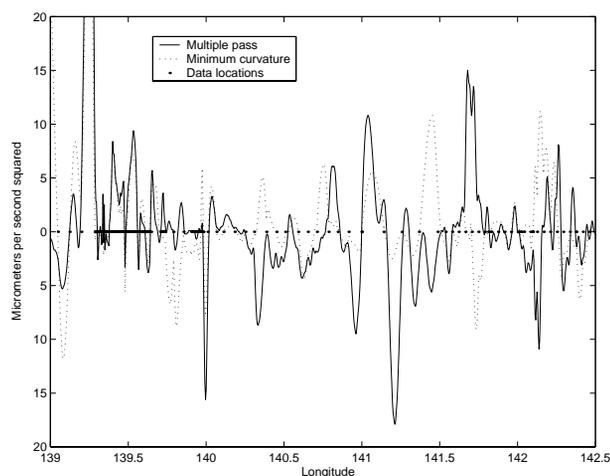


FIG. 4. Difference between the thin-plate spline surface and the standard and multiple-pass minimum curvature grids along the profile marked in Figure 3a.

Last, in Figure 6 we compare the different fits along the profile shown in Figure 5a. The profile lies away from the areas contaminated with bull's-eye artifacts. As one might expect, there are quite large deviations in the fitted surfaces in areas with sparse data coverage (between 141°E and 141.5°E). Where the coverage is a bit denser and more uniform (between 141.5°E and 141.8°E), the different surfaces give very similar fits.

Aeromagnetic survey

We now turn to the problem of gridding aeromagnetic survey data where there is a relatively uniform sampling regime, but with significantly different sampling densities in the x - and y -directions. The survey we chose was flown with a 200-m line spacing at a mean terrain clearance of 60 m with data collected every 7 m along the flight lines. The data are proprietary; so to protect the location we rotated and transformed the coordinates to cover the range 0 to 15.4 km east and 0 to 21.8 km north (Figure 7a). Notice the large gap in the center of the survey where there was a large lake that was not surveyed and the absence of several lines which did not meet the survey specifications.

The survey consisted of 178 545 points. Fitting a thin-plate spline took 31 minutes and 24 s on an Intel Celeron operating at 450 MHz. The evaluation required another 33 s. The resulting thin-plate spline surface evaluated on a 50-m grid cell size is shown in Figure 7c, a standard (Geosoft) Minq fit in Figure 7d, and a difference image in Figure 7b. The Minq fit took 4 minutes, 35 s. Excluding the data-free region in the center of the survey, the standard deviation of the residuals of the Minq fit compared to the thin-plate spline is 7.3 nT with a range of -165 to 452 nT. As one might expect, the largest differences are in the area with missing lines and also along some of the numerous faults evident in the image. This latter observation reflects the thin-plate spline's slightly improved ability to fit linear features transverse to the sampling direction. However, overall the thin-plate spline image differs little from the Geosoft minimum curvature image.

DISCUSSION

Sibson and Stone (1991) felt that interpolation with CGS when there were more than 10 000 data points would never be feasible. Clearly, this restriction no longer applies, with current code able to solve problems with 5 million centers in two dimensions and several hundred thousand centers in three dimensions (Beaston et al., 2001). The key advances have been the implementation of good iterative methods, efficient preconditioners, and fast methods for evaluating matrix-vector products. For the two geophysical examples considered in this paper, our Intel Celeron 450-MHz computer required about 15 minutes for the 55 000-point gravity data set and 34 minutes for the 178 000-point aeromagnetic survey. The code is continually being improved, so these times would be expected to decrease further.

The gravity data set demonstrated that the additional computational time required to compute the full thin-plate spline surface was justified. The thin-plate spline is scale independent and is able to adjust to vastly different data densities without any detrimental effects on other parts of the surface. We found that scale-dependent basic functions were unable to cope with

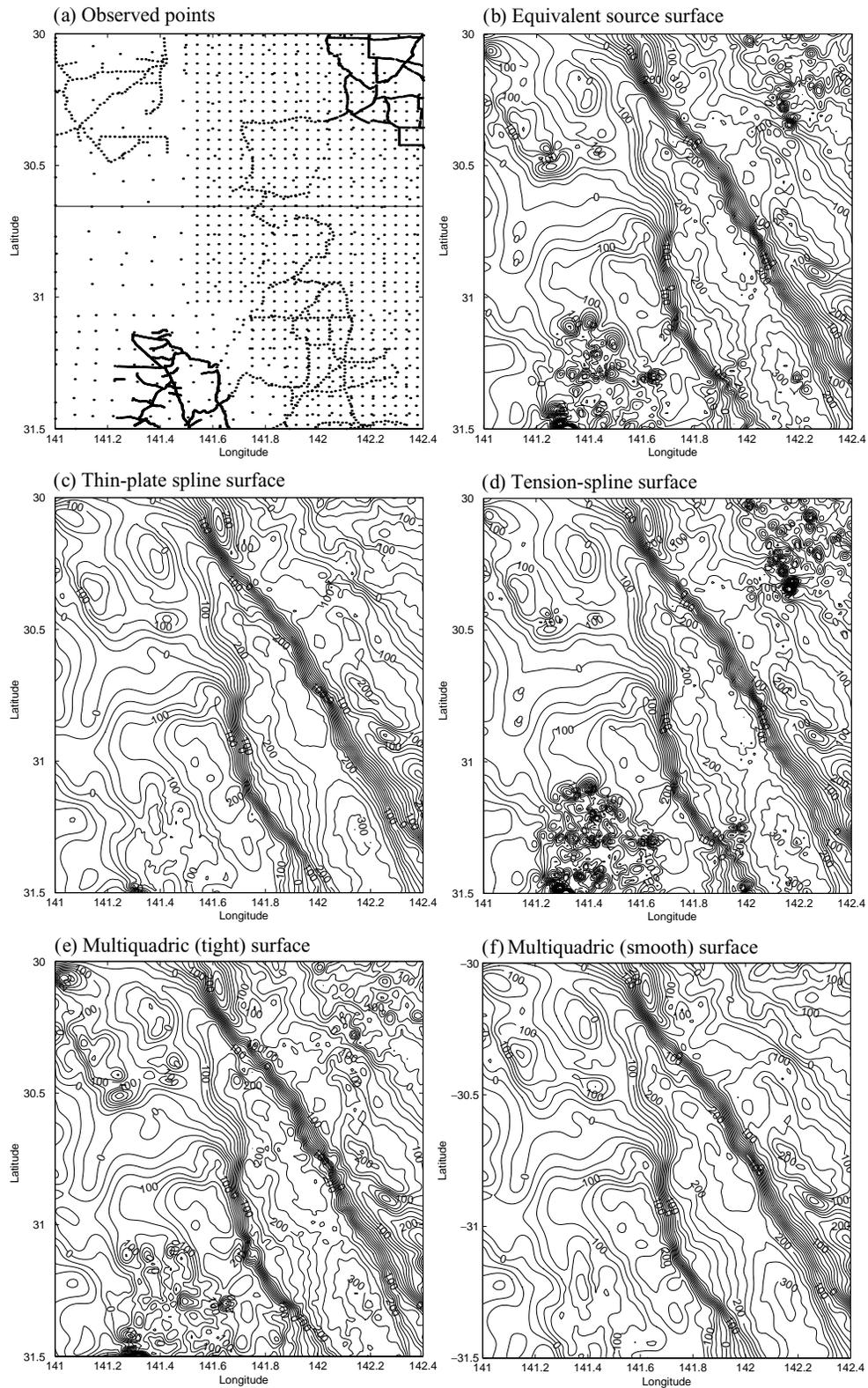


FIG. 5. (a) Data points and contour plots for (b) equivalent source gridding, (c) thin-plate spline, (d) tension spline, (e) multiquadric (tight), and (f) multiquadric (smooth) fits to part of the Broken Hill gravity data set.

such heterogeneous data sampling. Where the scale parameter might be optimum for one sampling density, it became suboptimal in other parts of the data set. Such basic functions would best be applied to interpolating surveys where there is

a fairly consistent spacing between data points, such as in airborne geophysical surveys.

The small difference between the thin-plate spline and minimum-curvature fits on the aeromagnetic survey indicates that the longer execution time required by the thin-plate spline would not be warranted in this case. However, we can envision two situations where computing a CGS might be justified. The first is where additional processing operations, such as upward and downward continuation, need to be applied to the surface. As investigated in Billings and Newsam (2002), CGS surfaces can have exact Fourier processing operations applied without the need for padding and data filling. The second is when the data are noisy and an exact fit is not ideal. In this case the user really desires a smooth surface that sensibly fits the data in some approximate fashion. The CGS framework readily encompasses several proven techniques for constructing such smooth approximations, along with automatic determination of the appropriate degree of smoothing, such as the widely used generalized cross-validation process. These types of smooth fits are required, in particular, for noisy airborne radiometric surveys, Billings et al. (2002a) investigate the application of CGS to this problem.

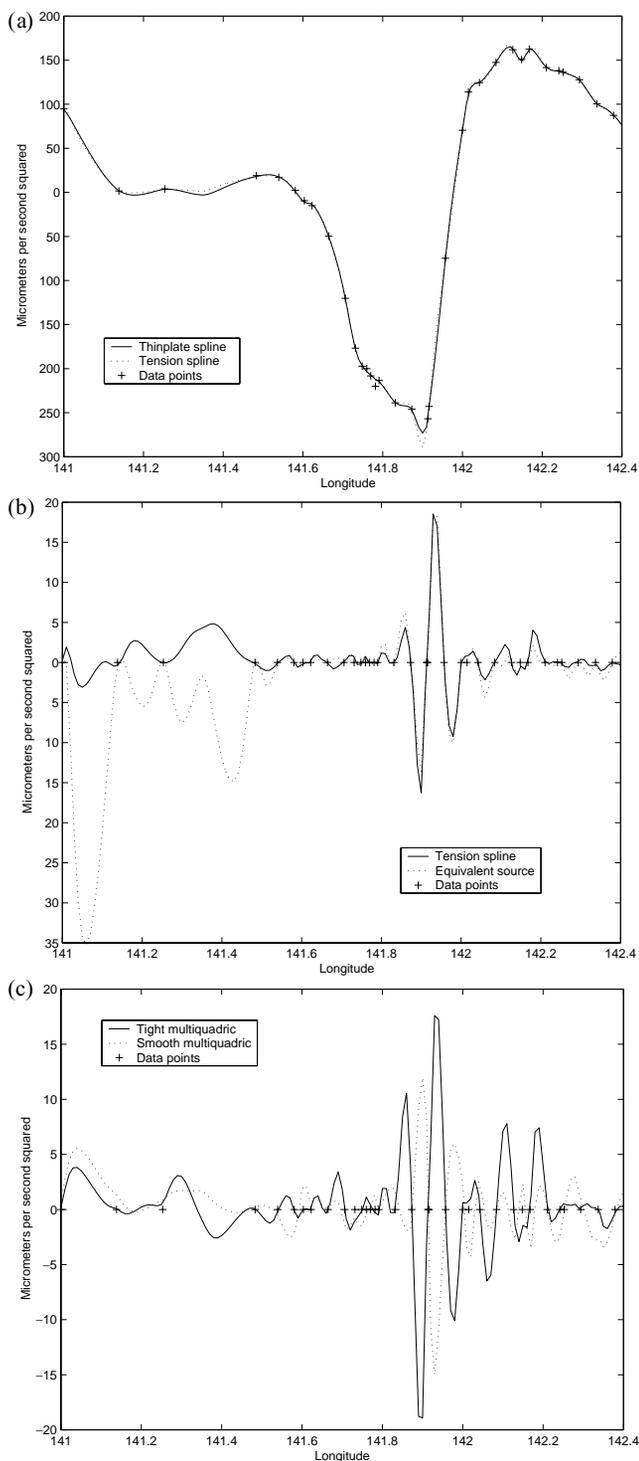


FIG. 6. (a) Profile along the line marked in Figure 5a for the thin-plate spline surface. (b) The difference between the thin-plate spline and the tension and equivalent-source profiles. (c) The difference between the thin-plate spline and the tight and smooth multiquadratic profiles.

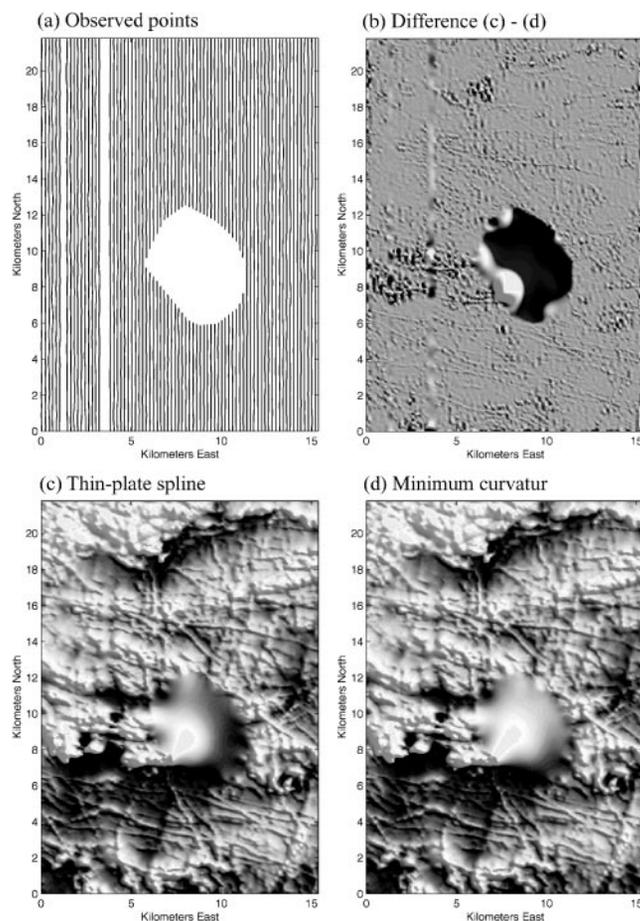


FIG. 7. (a) Data distribution, (b) difference image (c)-(d), (c) thin-plate spline, and (d) minimum-curvature grids for a 60-m elevation, 200-m line spacing airborne magnetic survey interpolated to a 50-m grid cell. The difference image has a linear contrast stretch within the limits -20 to 20 nT.

ACKNOWLEDGMENTS

We thank AGSO–Geoscience Australia for permission to use the Broken Hill gravity data set and Rio Tinto Exploration, Australia Region, for the aeromagnetic data. We are grateful to Tim Mitchell of Canterbury University for his assistance with some of the computations. Matlab code suitable for small- to medium-sized geophysical surveys is available from the authors. See <http://www.geop.ubc.ca/~sbilling/cgs.html> for details.

REFERENCES

- Bates, D. M., and Wahba, G., 1982, Computational methods for generalized cross validation on large data sets, *in* Baker, C., and Miller, G., Eds., *Treatment of integral equations by numerical methods*: Academic Press Inc.
- Beatson, R. K., and Chacko, E., 2000, Fast evaluation of radial basis functions: A multivariate momentary evaluation scheme, *in* Cohen, A., Rabut, C., and Schumaker, L. L., Eds., *Curve and surface fitting*: Saint Malo 1999, Vanderbilt Univ. Press, 37–46.
- Beatson, R. K., and Newsam, G. N., 1992, Fast evaluation of radial basis functions, I: *Comput. Math. Appl.*, **24**, 7–19.
- 1998, Fast evaluation of radial basis functions: Moment based methods: *SIAM J. Sci. Comp.*, **19**, 1428–1449.
- Beatson, R. K., Light, W. A., and Billings, S. D., 2001, Domain decomposition methods for solution of the radial basis function interpolation equations: *SIAM J. Sci. Comp.*, **22**, 1717–1740.
- Billings, S. D., Newsam, G. N., and Beatson, R. K., 2002, Smooth fitting of geophysical data using continuous global surfaces: *Geophysics*, **67**, 1823–1834, this issue.
- Billings, S. D., and Newsam, G. N., 2002, Fourier filtering of continuous global surfaces, *in* Skodras, A. N., and Constantinides, Eds., *Proc. 14th Internat. Conf., Digital Signal Processing*, **2**, 803–806.
- Briggs, I. C., 1974, Machine contouring using minimum curvature: *Geophysics*, **39**, 39–48.
- Carr, J. C., Beatson, R. K., Cherie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., and Evans, T. R., 2001, Reconstruction and representation of 3D objects with radial basis functions: *SIGGRAPH 2001, Proceedings*, 67–76.
- Cheney, E. W., and Light, W. A., 1999, *A course in approximation theory*: Brooks/Cole Publ. Co.
- Cordell, L., 1992, A scattered equivalent-source method for interpolation and gridding of potential-field data in three dimensions: *Geophysics*, **57**, 629–636.
- Cressie, N. A. C., 1993, *Statistics for spatial data*, 2nd ed: John Wiley & Sons, Inc.
- Dubrulle, O., 1984, Comparing splines and kriging: *Comp. and Geosci.*, **10**, 327–338.
- Duchon, J., 1976, Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces: *RAIRO Anal. Numer. Num.*, **10**, 5–12.
- Dyn, N., Levin, D., and Rippa, S., 1986, Numerical procedures for surface fitting of scattered data by radial functions: *SIAM J. Sci. Stat. Comput.*, **7**, 639–659.
- Foley, T., and Hagen, H., 1994, Advances in scattered data interpolation: *Surv. Math. Ind.*, **4**, 71–84.
- Greengard, L., and Rohklin, V., 1987, A fast algorithm for particle simulations: *J. Comput. Phys.*, **73**, 325–348.
- Hestenes, M. R., and Stiefel, E., 1952, Methods of conjugate gradient for solving linear systems: *J. Res. Nat. Bur. Stand.*, **49**, 409–436.
- Horowitz, F. G., Hornby, P., Bone, D., and Craig, M., 1996, Fast multidimensional interpolations: 26th Proc. of Appl. of Comp. and Operat. Res. in the Min. Ind.: Soc. Mining, Metal. and Expl., Abstracts, 53–56.
- Hutchinson, M. F., 1993, On thin plate splines and kriging, *in* Tarter, M. E., and Lock, M. D., Eds., *Computing and science in statistics: Interface Foundation of North America, Univ. of California, Berkeley*, 55–62.
- Hutchinson, M. F., and Gessler, P. E., 1994, Splines—More than just a smooth interpolator: *Geoderma*, **62**, 45–67.
- Journel, A. G., and Huijbregts, C. J., 1978, *Mining geostatistics*: Academic Press Inc.
- Kelley, C. T., 1995, *Iterative methods for linear and non-linear equations*: Soc. Ind. Appl. Math.
- Krige, D. G., 1951, A statistical approach to some basic mine valuation problems on the Witwatersrand: *J. Chem. Metal. Min. Soc. South Africa*, **52**, 119–139.
- Matheron, G., 1963, Principles of geostatistics: *Econ. Geol.*, **58**, 1246–1266.
- 1980, Splines and kriging: Their formal equivalence: *Syracuse Univ. Geol. Contrib.*, **8**, 77–95.
- Micchelli, C. A., 1986, Interpolation of scattered data: Distance matrices and conditionally positive definite functions: *Constr. Approx.*, **2**, 11–22.
- Mitasova, H., and Mitas, L., 1993, Interpolation by regularized spline with tension: I—Theory and implementation: *Math. Geol.*, **6**, 641–655.
- Murray, A., 1998, High precision gridding of gravity data: 13th Aust. Soc. Expl. Geophys. Conf., Extended Abstracts, 107.
- Myers, D. E., 1988, Interpolation with positive definite functions: *Sci. Terre*, **28**, 251–265.
- Saad, Y., and Schultz, M. H., 1986, GMRES: Generalized minimal residual algorithm for solving non-symmetric linear systems: *SIAM J. Sci. Stat. Comput.*, **7**, 856–869.
- Shannon, C. E., 1949, Communications in the presence of noise: *Proc. IRE*, **37**, 10–21.
- Sibson, R., and Stone, G., 1991, Computation of thin-plate splines: *SIAM J. Sci. Stat. Comp.*, **12**, 1304–1313.
- Smith, W. H. F., and Wessel, P., 1990, Gridding with continuous curvature splines in tension: *Geophysics*, **55**, 293–305.
- Wahba, G., 1990, Spline models for observational data: *Soc. Ind. Appl. Math.*