Gridding with Continuous Curvature Splines in Tension

Article *in* Geophysics · March 1990

DOI: 10.1190/1.1442837

Walter HF Smith
National Oceanic and Atmospheric Administration
136 PUBLICATIONS 39,257 CITATIONS

Paul Wessel
University of Hawaiʻi at Mānoa
161 PUBLICATIONS 26,906 CITATIONS

Dear reader:

Thank you for your interest in our paper, “Gridding with continuous curvatures splines in tension” [*Geophysics*, volume 55, number 3, pages 293–305, March, 1990; doi: 10.1190/1.1442837].

At the time this paper was written and published, *Geophysics* used a private type-setting service and did not allow author-prepared copy. We went through several iterations of correcting page proofs, but even so, errors in the equations remain in the published version. Therefore you cannot write a working computer code to implement our algorithm by coding the equations as they were published. If you want source code for the algorithm, a correctly working and optimized version in the *C* language exists as “surface.c” in our Generic Mapping Tools software (http://gmt.soest.hawaii.edu). This is available for free download under the Gnu Public License.

If there are any errors in the prose explaining the general features of the method and the thinking behind it, those are my responsibility. We were students when we devised this algorithm and wrote the paper. Re-reading it today, 25 years later, I think it has held up fairly well.

Walter H. F. Smith

28 May 2015

GEOPHYSICS, VOL. 55, NO. 3 (MARCH 1990); P. 293-305, 7 FIGS.

Gridding with continuous curvature splines in tension

W. H. F. Smith* and P. Wessel‡

ABSTRACT

A gridding method commonly called minimum curvature is widely used in the earth sciences. The method interpolates the data to be gridded with a surface having continuous second derivatives and minimal total squared curvature. The minimum-curvature surface has an analogy in elastic plate flexure and approximates the shape adopted by a thin plate flexed to pass through the data points. Minimum-curvature surfaces may have large oscillations and extraneous inflection points which make them unsuitable for gridding in many of the applications where they are commonly used. These extraneous inflection points can be eliminated by adding tension to the elastic-plate flexure equation. It is straightforward to generalize minimum-curvature gridding algorithms to include a tension parameter; the same system of equations must be solved in either case and only the relative weights of the coefficients change. Therefore, solutions under tension require no more computational effort than minimum-curvature solutions, and any algorithm which can solve the minimum-curvature equations can solve the more general system. We give common geologic examples where minimum-curvature gridding produces erroneous results but gridding with tension yields a good solution. We also outline how to improve the convergence of an iterative method of solution for the gridding equations.

INTRODUCTION

A wide variety of numerical procedures in the earth sciences require data on a regularly spaced lattice, including Fourier analysis and many map-drawing algorithms. In contrast, most geologic data are acquired at individual observation points or along traverses. It is therefore necessary to construct estimates of the value of a function on a grid, given observations of the value of the function at arbitrary locations in the x, y plane. This operation is called gridding.

There are three areas of concern in evaluating a gridding algorithm. The relative importance of each depends on the intended application. The first concern involves the global properties of the solution. Some procedures construct an interpolant of a priori known functional form. The second concern involves honoring data constraints; e.g. deciding whether the data are fit exactly or approximately. The third concern involves the method of interpolation or extrapolation in poorly constrained regions. It is in this last area that gridding algorithms differ most and where global properties strongly affect the solution.

In general, all gridding algorithms share these underlying assumptions: (a) the function to be gridded is single-valued at any point; (b) the function is continuous within the region to be gridded; and (c) the function is positively autocorrelated over some length scale at least as large as the typical spacing between observation points (Harbaugh et al., 1977; Davis, 1986). Nearly all methods estimate values at grid nodes from weighted averages of nearby data points, a procedure justified by assumption (c) in particular. Although rarely pointed out, some methods also require that the control data not contain information at wavelengths shorter than twice the grid spacing in order that spatial aliasing will not occur. Later in this paper we discuss prefiltering the data to avoid this problem.

Weighted-average schemes differ in how they assign weights to the constraining values. The simplest methods use a polynomial or power law in distance; most other methods use some sort of minimum-norm principle (Wegman and Wright, 1983). We divide these into two groups which we call statistical methods and integral methods. Statistical methods minimize the variance of the grid-value estimator by selecting weights based on the data autocorrelation. The kriging methods used in economic geology (Olea, 1974; Clark, 1979) belong to this class. An advantage of these methods is that they can yield confidence limits for the grid values; a possible disadvantage is that global properties of

Manuscript received by the Editor March 7, 1989; revised manuscript received August 1, 1989.
*Lamont-Doherty Geological Observatory of Columbia University, Route 9W, Palisades, NY 10964.
‡Formerly Lamont-Doherty Geological Observatory of Columbia University; presently Hawaii Institute of Geophysics, 2525 Correa Road, Honolulu, HI 96822.

the surface such as high-order continuity cannot be assured a priori. Integral methods begin with the requirement that the surface should minimize some global norm over some set of functions of the data; the weights are then determined to satisfy this constraint while fitting the data. The advantage of these methods is that they assure a solution with the desired properties; their disadvantage is that they do not easily yield confidence limits.

The method we present here is a generalization of a popular integral method called "minimum curvature." In the minimum-curvature method, an interpolant with continuous second derivatives is constructed such that the squared curvature integrated over the entire surface is minimized. Briggs (1974) derived the equations and suggested their solution by iteration of finite-difference expressions; Swain (1976) gave a Fortran algorithm based on Briggs' method; and Sandwell (1987) solved the same equations by a matrix method. Variations on the Swain algorithm (e.g., the U.S. Navy's SuperMISP) are in widespread use in the earth-science community. For example, the gravity and magnetic anomaly maps of North America (GSA Map Committees, 1987), and the Digital Bathymetric Data Base of the U.S. Navy (Van Wyckhouse, 1973; NGDC, 1988) are prepared by the minimum-curvature method. The heavy solid lines in Figures 1a and 1b are profiles through minimum-curvature surfaces. They honor the data at constrained points, but have large oscillations between these points. This behavior in unconstrained regions may be undesirable in some applications.

In one dimension, the function with continuous second derivatives that interpolates the data constraints exactly and minimizes total curvature is called the interpolating natural cubic spline (cf., de Boor, 1978; Lancaster and Salkauskas, 1986). It may have the large oscillations between constraints shown in Figure 1a. Two modifications to this spline have been used to avoid these oscillations. The first (type 1) modification relaxes the requirement that the data be interpolated exactly; a compromise is made between the misfit to the data and the curvature of the solution. Solutions of type 1 are called smoothing splines (cf., de Boor, 1978; Lancaster and Salkauskas, 1986). The second (type 2) modification, in contrast to the first, interpolates the data exactly but relaxes the constraint that the total curvature must be minimized. One class of type 2 solutions is splines in tension (Schweikert, 1966). The oscillation of the natural cubic spline can result in extraneous inflection points; Schweikert (1966) showed that a spline in tension eliminates these inflections. Note in Figure 1b that the control data can be interpolated with a function which is everywhere concave down (e.g., the thin solid line), yet the minimum-curvature solution (heavy solid line) changes concavity. Späth (1973) has given Schweikert's (1966) equations in a strictly diagonally dominant tridiagonal matrix form, and Cline (1974) has adapted Schweikert's spline to curves in the (x, y) plane.

In two dimensions, the minimum-curvature interpolant is the natural bicubic spline which can have the same oscillations and extraneous inflection points as in the one-dimensional (1-D) case. Again the same methods may be used to suppress these features. Inoue (1986) has given a type 1 modification in which the damping of first and second derivatives is traded off against the data misfit under a least-squares norm. This is essentially a two-dimensional (2-D) smoothing spline which does not fit the data exactly. In this paper we present a type 2 modification. We show how the minimum-curvature gridding method (Briggs, 1974; Swain, 1976; Sandwell, 1987) can be generalized to include tension in the interior and boundary equations, and we show common geologic examples where minimum curvature produces undesirable results but the introduction of tension significantly improves the solution. Our method produces a suite of surfaces with continuous second derivatives of which the minimum-curvature surface is one end member. Increasing the tension parameter relaxes the global minimum-curvature constraint by moving toward a solution with curvature localized at the control data points; at the same time, the surface fits the data exactly. Adjustable tension

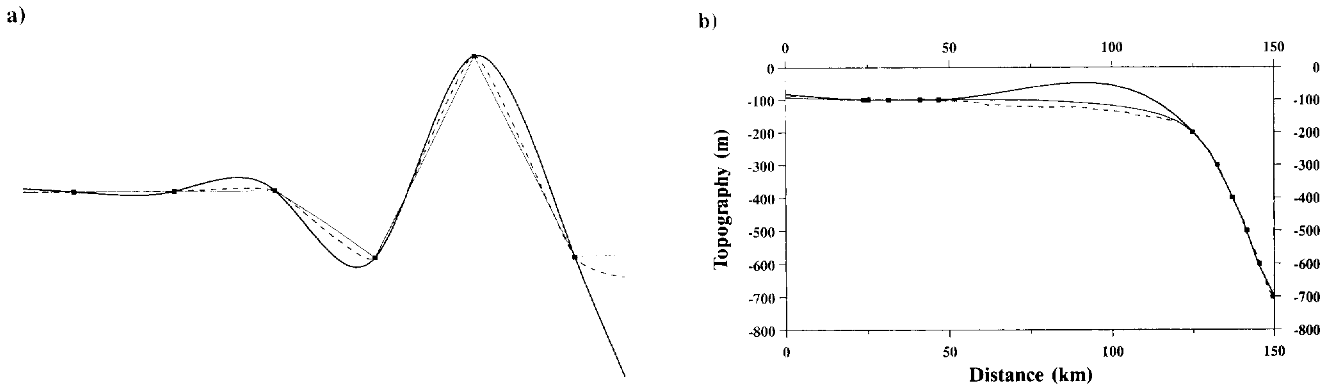

FIG. 1. (a) Cross-sections through surfaces produced with splines in tension. The black squares are data constraints. The heavy line is the minimum-curvature end member, the thin line is the harmonic end member, and the dashed line is an intermediate case using some tension. Note that all solutions honor the data points. (b) Cross-section through a continental shelf and slope. The black squares represent the intersection between the measured bathymetry (dashed line) and 100 m isobath contours. These intersections (contour coordinates) were then gridded using minimum curvature (heavy solid line) and some tension (thin solid line). The minimum-curvature method introduces an extraneous inflection point and exceeds the −100 m level, although we know that bathymetry in this region is bounded by the −200 m and −100 m levels. The surface produced with tension gives a much better approximation since it suppresses local maxima and minima between data constraints.

permits the gridding of data of varying roughness and allows the user to satisfy his own criteria for a good solution. All the lines in Figure 1a and the solid lines in Figure 1b are profiles through surfaces produced by our method.

Swain (1976) gave a method for iterative solution of the finite-difference equations of minimum curvature (Briggs, 1974). We show that including tension in these equations results merely in a change in coefficients, and therefore any minimum-curvature algorithm based on Swain's method can easily be modified to solve our equation. We also discuss how the convergence can be improved to achieve an order of magnitude reduction in run time of the original Swain algorithm. A C language program which incorporates all the features of our method and the sample data gridded in the examples in this paper are available from the authors on request.

MINIMUM CURVATURE, ELASTIC PLATE FLEXURE, AND TENSION

Gridding equations and physical analogs

Minimum-curvature gridding algorithms use the norm

$$C = \iint (\nabla^2 z)^2 \, dx \, dy. \qquad (1)$$

Equation (1) is a valid approximation for the total curvature of z when $|\nabla z|$ is small. Briggs (1974) showed that minimizing equation (1) leads to the differential equation

$$\nabla^2(\nabla^2 z) = \sum_i f_i \, \delta(x - x_i, y - y_i), \qquad (2)$$

where (x_i, y_i, z_i) are constraining data. The f_i must be chosen such that $z \rightarrow z_i$ as $(x, y) \rightarrow (x_i, y_i)$, and the boundary conditions are

$$\frac{\partial^2 z}{\partial n^2} = 0 \qquad (3)$$

and

$$\frac{\partial}{\partial n} (\nabla^2 z) = 0 \qquad (4)$$

along edges, where $\partial/\partial n$ indicates a derivative normal to an edge, and

$$\frac{\partial^2 z}{\partial x \, \partial y} = 0 \qquad (5)$$

at the corners. Equations (3), (4), and (5) are called free-edge conditions; and with these conditions, equation (2) has a unique solution with continuous second derivatives called the natural bicubic spline. The nomenclature comes from an analogy with elastic-plate flexure. Small displacements z of a thin elastic plate of constant flexural rigidity D, subject to a vertical normal stress q and constant horizontal forces per unit vertical length of T_{xx}, T_{xy}, and T_{yy}, approximately satisfy

$$D\nabla^2(\nabla^2 z) - \left[T_{xx} \frac{\partial^2 z}{\partial x^2} + 2T_{xy} \frac{\partial^2 z}{\partial x \, \partial y} + T_{yy} \frac{\partial^2 z}{\partial y^2} \right] = q \qquad (6)$$

(Love, 1927). The minimum-curvature gridding equation (2) is a special case of equation (6) when horizontal forces are zero, and the boundary conditions represent zero bending moment on the edges [equation (3)], zero vertical shear stress on the edges [equation (4)], and zero twisting moment at the corners [equation (5)] (Timoshenko and Woinowsky-Krieger, 1968). Physically, the f_i represent the strengths q/D of point loads on the elastic plate; mathematically, they are coefficients in a solution which is composed of a linear combination of Green's functions for plate flexure due to unit point loads.

The total stored elastic strain energy in the flexed plate is approximately proportional to the curvature (1); of all twice-differentiable surfaces interpolating the data, the minimum-curvature surface stores the least strain energy. If one imagines bending an elastic plate to interpolate the data, then extra work must be done on the plate to create any interpolant other than the minimum-curvature solution. The minimum-curvature solution may seem to be the "natural" way to estimate poorly constrained grid values. It is clear from Figure 1, however, that the minimum-curvature constraint may create extraneous oscillations.

We derived our equation by investigating the role of a uniform isotropic tensile stress T in equation (6). Suppose that $T_{xx} = T_{yy} = T$ and $T_{xy} = 0$. Then equation (6) becomes

$$D\nabla^2(\nabla^2 z) - T\nabla^2 z = q. \qquad (7)$$

When $T = 0$, equation (7) is equivalent to equation (2); but for arbitrarily large T, the solution is dominated by the second term. Here T has units of force per unit length and the T required to adjust the solution scales with D and q; we avoided this by writing

$$(1 - T_I)\nabla^2(\nabla^2 z) - T_I \nabla^2 z = \sum_i f_i \, \delta(x - x_i, y - y_i), \qquad (8)$$

where T_I is a tension parameter and the I subscript indicates internal tension. (We specify tension on the boundaries independently of T_I.) Now we may vary T_I from 0 to 1, with 0 and 1 giving the end-member cases shown as solid lines in Figure 1a. When $T_I = 0$, equation (8) reduces to equation (2); and therefore the minimum-curvature solution is one end-member case of equation (8). When $T_I = 1$, the first term in equation (8) vanishes; and the solution is harmonic between constraining points. In the elastic analogy, $T_I = 1$ represents infinite tension; since infinite tension is not physically meaningful, one may prefer to think of this end member as representing the steady-state temperature field in a conducting plate with heat sources or sinks at the data points. An important property of this solution is that it cannot have local maxima or minima except at constraining data points (this follows from the interior mean value theorem for harmonic functions; e.g., Berg and McGregor, 1966). Note that for any T_I in $0 \leq T_I < 1$, equation (8) gives a solution with continuous curvature, although it does not minimize equation (1) except when $T_I = 0$.

We implemented boundary tension with conditions (4) and (5) above but replacing condition (3) by

$$(1 - T_B) \frac{\partial^2 z}{\partial n^2} + T_B \frac{\partial z}{\partial n} = 0, \qquad (9)$$

where T_B is a tension parameter for the boundary which also varies between 0 and 1. The free-edge condition corresponds to $T_B = 0$; $T_B = 1$ forces the solution to flatten at the edge (see Figure 1). This flattening is sometimes desired, as when gridding potential anomalies which should decay toward a regional background field away from the source region.

Tension as a weighted minimum-norm solution

Rayleigh's theorem (Bracewell, 1978) may be applied to the curvature norm (1) to show that squares of Fourier components of z are integrated with weights proportional to the fourth power of their wavenumber; short wavelengths in z contribute much more to C than long wavelengths. Minimizing C yields a solution with power concentrated at long wavelengths and is suitable for gridding data which vary slowly with distance. The tension parameter in equation (8) effectively defines a weighted minimum-norm problem related to the direct minimum-norm solution (2); when $T_I = 1$, the weights are proportional to the square of the wavenumber. This weighting increases curvature locally near the constraining data and yields a solution with more short-wavelength power. Tension relaxes the global minimum-norm constraint in order to find a solution with more local variation and is suitable for gridding data which vary more rapidly with distance.

CONSTRUCTION OF A SOLUTION

Regional fields

Nearly all minimum-norm gridding equations operate on perturbations from a regional trend. For example, statistical methods require that a regional field be removed from the data so as to make the residuals stationary; and the grid estimates are then constructed as estimates of the local departure from this regional trend. In the integral method above, the complete solution to equation (2) or equation (8) consists of a linear combination of Green's functions for plate flexure by point loads, plus an additional function which is a solution to the homogeneous equation related to equations (2) or (8). This additional function we call a "regional field."

Swain (1976) and Sandwell (1987) did not include a regional field in their solutions to equation (2). Ignoring the regional field means that the true regional function is approximated by a linear combination of Green's functions for plate flexure. We have found that this approximation is not numerically efficient, and the stability and convergence of solutions to equation (2) or equation (8) are enhanced by including a regional field model. In our algorithm, the regional trend is approximated by removing a least-squares plane from the data before solving equation (8). We add the plane back into the grid values after equation (8) is solved. This procedure has the additional effect that the boundary tension T_B drives the solution toward the regional plane rather than toward a horizontal plane.

Finite-difference approach

There are many ways to solve equations (2) or (8). The most direct approach (Sandwell, 1987) is to express $z(x, y)$ as a linear combination of Green's functions for equation (2) or equation (8) and to construct a matrix equation $\underset{\sim}{\mathbf{G}}\mathbf{f} = \mathbf{d}$, where $\underset{\sim}{\mathbf{G}}$ is a matrix of Green's functions (the data kernel, in geophysical parlance), **f** is a vector of unknown coefficients f_i, and **d** is a vector of known data constraints $z(x_i, y_i) = z_i$. This matrix equation is solved for the f_i, and then the z values on the grid nodes are found from the linear combination of Green's functions given by these f_i. Unfortunately, this approach often yields a system which is nearly singular, because the ratio of the distances between the farthest pair and closest pair of data points is large (this is always true for data collected along traverses or tracks), and therefore some column vectors of the Green's matrix $\underset{\sim}{\mathbf{G}}$ are nearly parallel.

Alternatively, one could write the finite-difference expressions for equation (2) or equation (8) in terms of the values at the grid nodes and solve this system. 1-D splines are usually constructed by writing these difference equations in matrix form, generally resulting in a band-diagonal system which is easily solved. However, if the same is done for a 2-D grid, a large and sparse matrix results.

The near-singularity of the Green's matrix and the large and sparse nature of the finite-difference matrix make these matrix methods unstable, and solutions must be found by iterative refinement and possibly also smoothing by zeroing small eigenvalues. We therefore follow the suggestion of Briggs (1974) and solve equation (8) directly by iteration of the difference equations among the grid values (see the Appendix). This procedure is always stable and allows us to control the path taken in interactive refinement of the solution. The numerical solution is path-dependent and so this flexibility is important.

Because equation (8) has one more term than equation (2), it may seem that it is more complicated to solve equation (8). However, when either equation (2) or equation (8) is approximated in terms of central finite differences among grid values, a linear equation among the points shown in Figure 2 results. The Laplacian of the surface at the square involves the four shaded circles; the biharmonic of the surface thus requires all twelve circles. At grid nodes unconstrained by data, the value of the surface at the square is given by a weighted average of the values at the circles. Adjusting the tension parameter T_I in equation (8) determines the weight of the shaded circles relative to the unshaded ones; however, for any value of T_I (except the limiting case $T_I = 1$), the same linear system relating the twelve circles to the square must be solved, and only the relative weights (coefficients) change. Thus, solving equation (8) requires no more computational effort than solving equation (2), and a program which solves equation (2) can be trivially modified to solve equation (8). In fact, because increasing tension gives more weight to the shaded circles in Figure 2 and leads to a more local solution, the solution with tension actually converges faster than the minimum-curvature solution.

Implementing data constraints

Once the user has chosen a discrete grid with lattice spacing (Δx, Δy), the information content of $z(x, y)$ is effectively limited; $z(x, y)$ now has a Nyquist wavelength, and only a limited number of data constraints can be fit exactly. In our algorithm, we assign each datum to its nearest grid node. In Figure 2, the dashed square surrounds all points in the (x, y) plane nearest the grid node indicated with a black square. Data in this dashed region are assigned to this node. If we have at most one data point nearest each grid node, we can fit these data exactly; but if more than one data value constrains a given node, the surface is locally overdetermined and some sort of smoothing or data decimation is required. Swain (1976) uses the datum nearest the node and ignores the others. We do not recommend this procedure, since it can alias information at wavelengths shorter than the Nyquist wavelength of the grid. To make use of all the data, one might solve equation (8) separately for each datum in turn and then average the solutions. However, because the system is linear, it is equivalent to solve equation (8) with one representative constraining value, which is an average of the original data. This method requires only one fourth-order solution for each node; and if it is done outside the gridding process, it reduces the number of data points that need to be stored in the gridding program's memory. Because spatial filtering procedures are generally useful apart from gridding and the best filtering method is application-dependent, we have decoupled the averaging process from the gridding process. That is, our algorithm does not include any provision for smoothing or averaging data at overdetermined nodes; we expect that the data have been preprocessed to give only one filtered value per grid node, which we then interpolate exactly. Often our preprocessing consists simply of finding the mean or median value at the mean or median position in each block of area nearest each node.

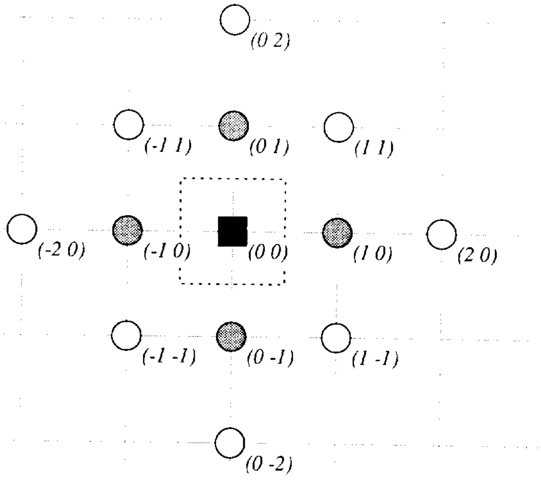

FIG. 2. When either equation (2) or equation (8) is expressed in central finite differences among the grid nodes, the estimate at one node (the black square) is given by a weighted average of the values at 12 nearby nodes (circles). Increasing the tension in equation (8) increases the weight of the shaded circles relative to the unshaded ones, producing a more local solution. In any case the same linear system must be solved and only the weights change; thus any minimum-curvature algorithm can be easily modified to include tension. Data constraints are assigned to their nearest grid node; a datum inside the dashed box is used to constrain the grid value at the square. Subscripts in parentheses illustrate the indexing scheme used in the difference equations in the Appendix.

Briggs (1974) gave an approximation for $\nabla^2 z$ at a grid node in terms of other nearby grid values and one off-grid data constraint. The expression uses a second-order finite-difference Taylor series expansion to predict the value of the interpolating surface away from grid nodes. Since both equations (8) and (2) are equations in $\nabla^2 z$, we can implement data constraints in equation (8) by modifying Briggs' method (see the Appendix). Using this approach, the constraining datum enters the local difference equation through the Taylor expansion; and we do not need to solve for f_i explicitly. If the difference equations were to converge exactly, then the surface would fit the data exactly, in the sense that the Taylor series expansion would match the data constraints with zero prediction error. Because the solution is found to finite precision, the fit is not perfect; the user enters a tolerance for numerical convergence, and the prediction error is of this order. We have found empirically that the mean prediction error is always nearly zero (thus the method is unbiased), and convergence to maximum absolute error of one part in 10^4 can be achieved in short run times (the meaning of "short" is relative to the number of nodes in the lattice). One important feature of the Taylor series method for fitting the data is that it honors a datum exactly when that datum falls on the lattice; if the (x, y) coordinates of the datum match those of a grid node, the value of that grid node is set equal to the datum value.

Convergence

The gridding equations (4), (5), (8), and (9) have a unique solution which we may call the true solution. Because we solve these equations iteratively with finite precision, we do not reach the true solution; and our result depends not only on the convergence limit of the iterations but also on the path taken toward the solution. Optimization of this path is important not only to achieve convergence in only a few iterations, but also because optimization yields a solution closer to the true solution. Convergence in computation is not the same as convergence in mathematics. We consider our iterations "converged" to limit ε when the maximum absolute change at any node during one iteration is less than ε. This does not mean that the result is within ε of the true solution; it means that further improvements in the result will be smaller than ε for each iteration and are therefore not worth the effort.

Details of our solution strategy are given in the Appendix. We generally follow the method of Swain (1976), but we include the tension parameters T_I and T_B and an additional parameter for grid anisotropy α. Users of gridding algorithms often grid data in map coordinates; at high latitudes the anisotropy in distance on a latitude-longitude grid can be significant. We have included an aspect ratio α in our algorithm, where the grid dimensions are such that $dy = dx/\alpha$, and an nth difference in y is scaled by α^n to accommodate the anisotropy. If x = longitude and y = latitude, then α = cosine (latitude). We also generalized the regional grid strategy of Swain (1976) and included successive over-relaxation to accelerate convergence. With these improvements, our algorithm solves the isotropic minimum-curvature problem in one-tenth the time required by Swain's algorithm; applications using tension converge even more rapidly because of the more local nature of the solution in tension.

GEOLOGIC EXAMPLES AND THE USE OF TENSION

The "questionable dipole" example

For this example we use shipboard gravity measurements from offshore Mauritania which are in the Lamont-Doherty marine geophysical data base. The data were cross-over error corrected (Wessel and Watts, 1988; Wessel, 1989) and then 5 by 5 minute block mean values were computed. These mean values were input to our gridding algorithm. In Figure 3 we show two contour maps prepared from these data. Figure 3a was prepared using $T_I = 0$ in equation (8), corresponding to the minimum-curvature method. Figure 3b shows the same data gridded with $T_I = 0.3$ in equation (8); i.e., with some tension in the gridding surface. The locations of the input data values are shown as squares. Both maps

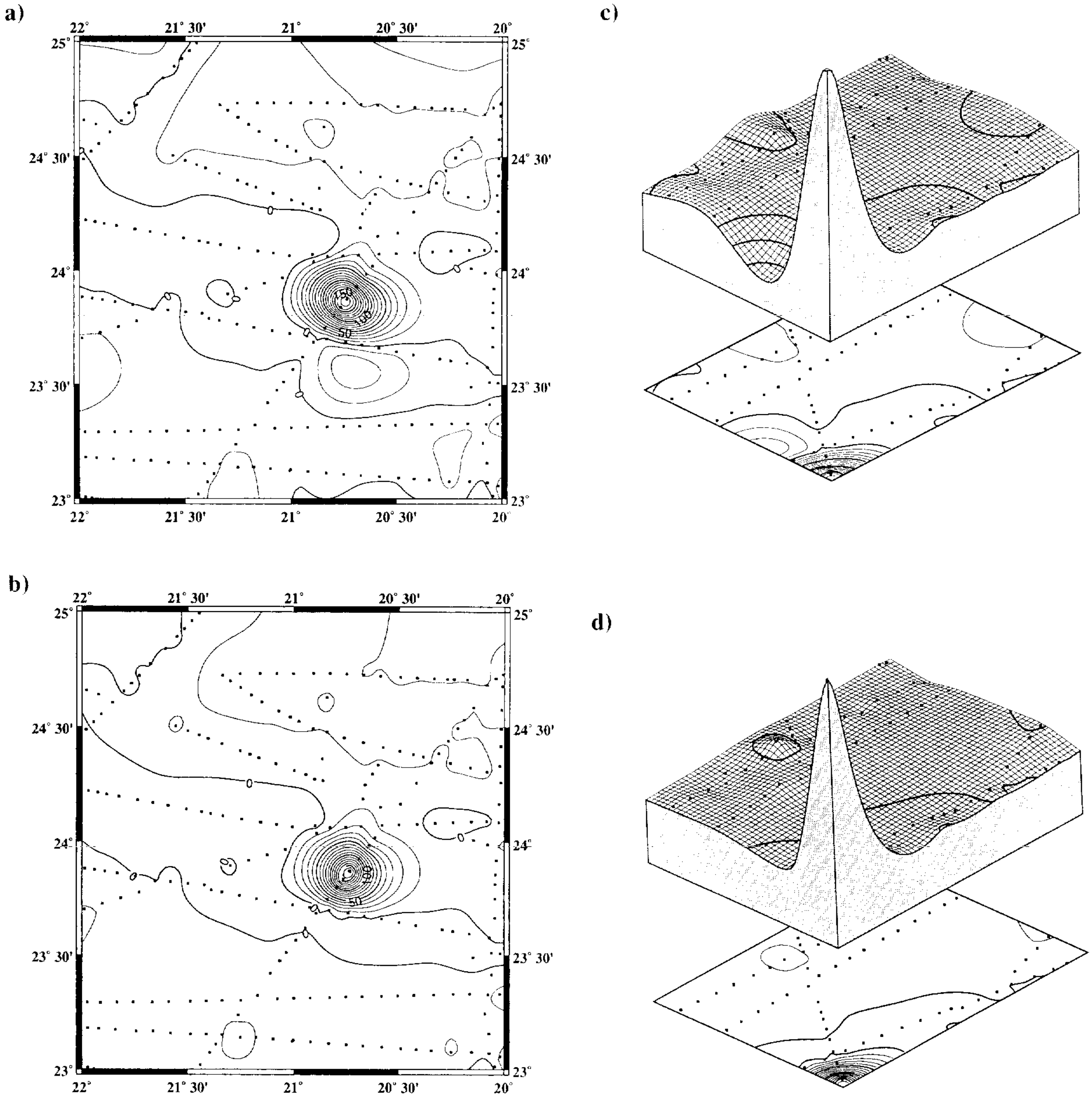

FIG. 3. 10 mGal contour maps of the gravity field from offshore Mauritania. Locations of control data are indicated by black squares. A positive anomaly of ~175 mGal was registered along the northeast trending track, while the other tracks measured values near zero. (a) The surface produced by the minimum-curvature method has an unconstrained low south of the constrained high. (b) The surface produced with some tension ($T_I = 0.3$) looks very similar to (a), except that the unconstrained low has been reduced significantly. (c) A perspective view of the shaded region in (a) from a vantage point above and northeast of the high. The oscillatory nature of the minimum-curvature surface is exemplified in the unconstrained low. (d) The same perspective view as in (c) of the surface produced with tension. Tension suppresses the oscillations found in (a) and (c) and reduces the magnitude of the unconstrained low to less than 10 mGal.

show a feature with a 175 mGal high which was observed along the northeasterly ship track. Other tracks near this feature recorded gravity values near zero. The most significant difference between the two maps is at the unconstrained low to the south of the constrained high. The minimum-curvature map suggests a dipole anomaly; there is a −25 mGal low to the south of the 175 mGal high in Figure 3a. Gridding with tension reduces this low to less than 10 mGal (Figure 3b). Note that there are no data at all in the region of the dipole low; we cannot say whether this low exists or not. By increasing T_I in equation (8), this low can be made to disappear. The user must decide whether he wants this low in the map or not. For example, if the data in Figure 3 were magnetic-intensity measurements, a high-low dipole might be expected; while if the data were bathymetric soundings, the low might be considered spurious; and in gravity data a flexural moat around a seamount high is sometimes reported (e.g., Watts, 1978). The point is that with our method the result may be adjusted as is geologically appropriate.

In Figures 3c and 3d we show perspective views of the shaded portions of the gridding surfaces of Figures 3a and 3b. In these views we have "cut away" the data east and north of the center of the high and are looking at the remaining region from a vantage point above and northeast of the high. These views illustrate the oscillatory flexure of the surface obtained by minimum-curvature gridding. Note in Figure 3d that tension results in a much sharper transition in the unconstrained area between the constrained high and the constrained flat regions.

In this "questionable dipole" example, it is not obvious that minimum-curvature gridding has done anything wrong; the validity of the low anomaly is a subjective decision. A lesson to be drawn from Figure 3 is that we should always plot the locations of constraining data on our contour maps. In the next two examples, we show that the minimum-curvature surface produces clearly undesired results.

The "shelf-break bulge"

Figure 1b contains a vertical cross-section through an actual bathymetric data profile over a continental shelf and slope (dashed line) and two attempts to reproduce this bathymetric surface by gridding the coordinates of isobaths (squares). The heavy solid line is a section through the minimum-curvature solution, and the thin solid line is a section through a solution produced with some tension. The heavy line displays a "shelf-break bulge" which occurs where an unconstrained area lies between two areas constrained to have different gradients, a situation very similar to the one that produced the "questionable dipole." However, while the dipole may or may not be real, the shelf-break bulge is a clear case of extraneous inflection points.

In this example we did not use the actual ship's bathymetry (dashed line) to constrain the gridding; instead, we found the locations of 100 m contours of the ship data and used the coordinates of these contours as the controlling data points. This situation is quite different from the "questionable dipole" example. What we are illustrating here is that attempts to grid surfaces using the coordinates of isopleths of the data suffer from a peculiar lack of information. The shelf-break bulge occurs where there is a large distance between constraining isopleths. When two contours are separated by a large distance, we know that the average gradient in that region is probably small; certainly, the surface is of bounded variation on that interval. However, the gridding algorithm only sees an unconstrained region.

The minimum-curvature solution is clearly wrong in this application. If the surface had a bulge as shown by the heavy solid line in Figure 1b, then there would have been another 100 m contour value somewhere as the bulge turned down to the continental slope. Including tension in the solution works very well (thin solid line). There are two reasons for this success. The first is that the bulge results from an extraneous inflection point, and tension has been shown to eliminate these inflections (Schweikert, 1966). The second is that increasing the tension moves the solution toward the harmonic end member, which can have no local maxima or minima between data points. This feature is well suited to gridding isopleth data, of course.

Shelf-break bulges are common features in the U.S. Navy's Digital Bathymetric Data Base (Van Wyckhouse, 1973; NGDC, 1988). In Figure 4a we have contoured this data set and shaded regions with elevations above −15 m, i.e., shallower than 15 m below sea level. The shading reveals

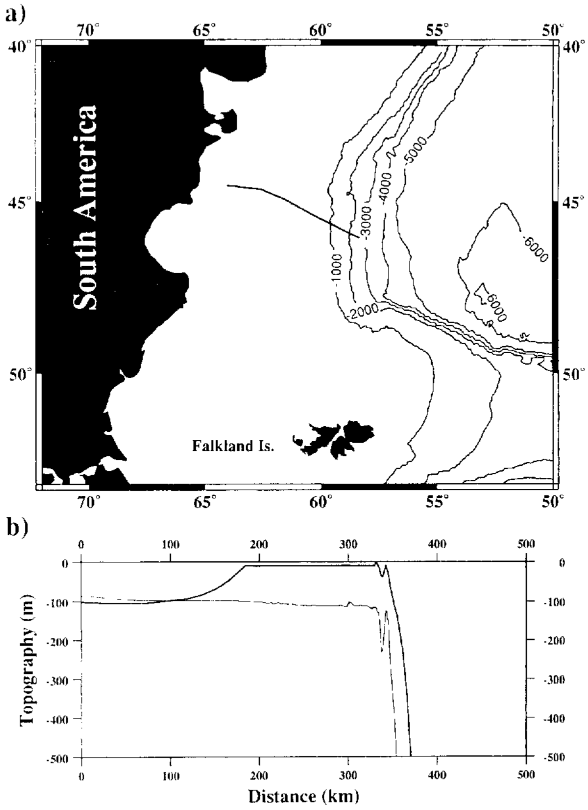

FIG. 4. (a) Map of U.S. Navy DBDB-5 bathymetry contoured at 1000 m intervals. Regions shallower than 15 m below sea level are shaded, revealing "shelf-break bulges." Heavy line indicates ship track of *R/V Vema* cruise 17-11. (b) Profile along the Vema 17-11 track. Thin line: actual bathymetry observed by *R/V Vema*. Heavy line: DBDB-5 data set sampled along the Vema 17-11 track.

prominent bulges in the continental shelf offshore South America. Figure 4b shows a profile along the line in Figure 4a. The thin line in Figure 4b is the actual bathymetry observed by *R/V Vema* cruise 17-11; the heavy line shows the DBDB-5 depths along the Vema track. We believe that the Navy has gridded contours of the original data, resulting in bulges which then have been truncated ad hoc to a −10 m level. In the next example we grid bathymetry data directly; minimum curvature produces a bulge in this example as well.

Bathymetric map of Broken Ridge

Figure 5a is a bathymetric contour map of Broken Ridge in the southern Indian Ocean (Driscoll et al., 1989). This map was hand contoured by a marine geologist at Lamont-Doherty using bathymetric soundings from the Lamont data base and additional data from the Defense Mapping Agency which are not in our digital data base. In Figures 5b and 5c we have tried to approximate the hand-drawn map by machine gridding and contouring the Lamont data only. Figure 5b is made with minimum curvature; and Figure 5c, with $T_I = 0.75$.

While Figure 5c cannot match the hand-drawn map exactly, the major features are quite similar. There are local differences, and we do not know how much the geologist was influenced by the DMA data which are not included in our map. The differences between Figures 5a and 5c and the minimum-curvature map (Figure 5b) are quite clear. In the unconstrained areas near the boundaries of the map, the

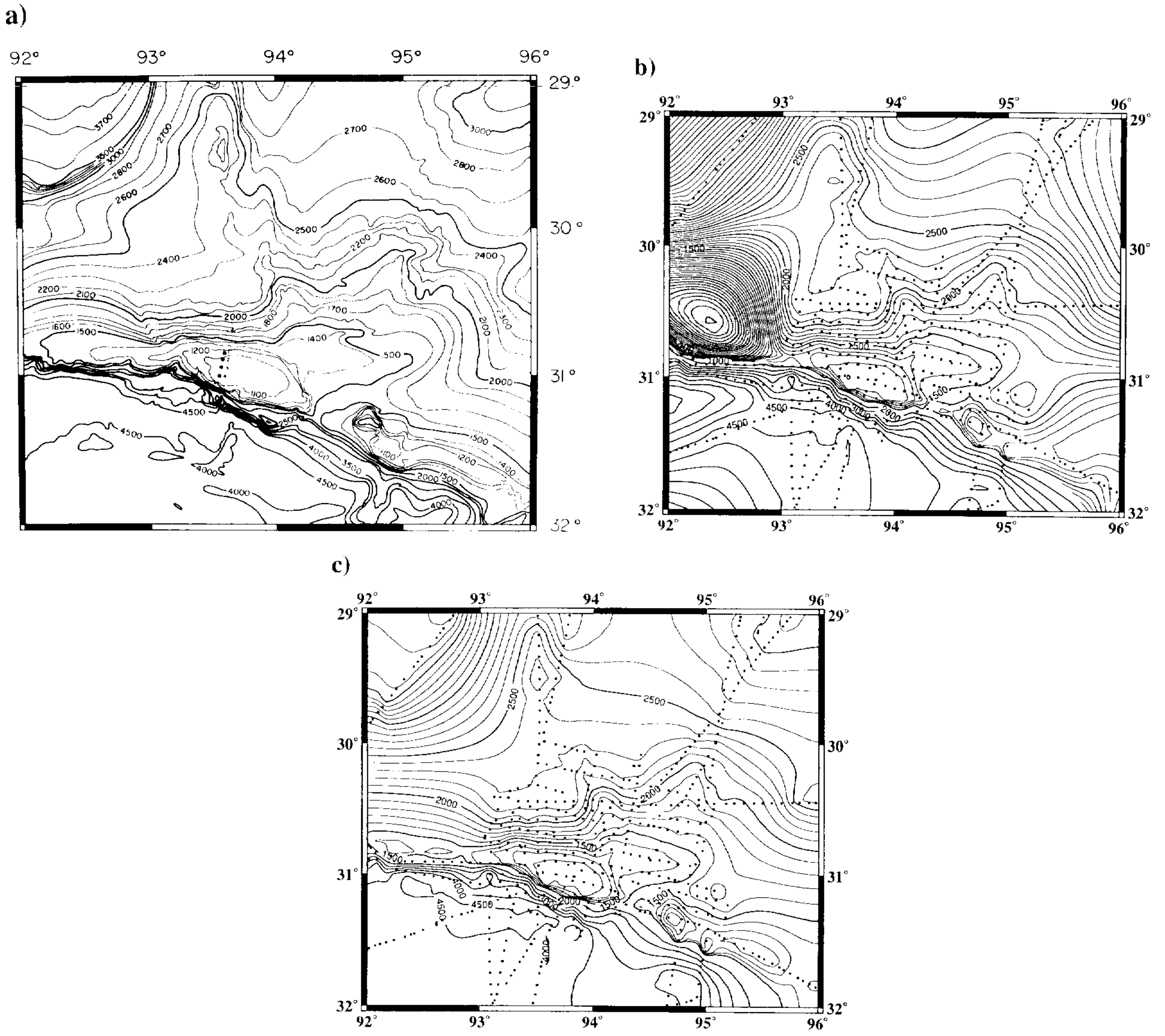

FIG. 5. Bathymetric maps of Broken Ridge in the southern Indian Ocean. (a) This map was hand contoured by a marine geologist using bathymetry from the Lamont-Doherty data base and additional data not available to us (Driscoll et al., 1989). (b) and (c) are generated by machine gridding of only the data in the Lamont-Doherty data base (locations shown as black squares). Both machine-drawn maps are in general agreement with the hand-drawn map in the regions well constrained by data. The major differences occur in regions of interpolation and extrapolation. The minimum-curvature solution (b) has large oscillations and a "false island" rising 900 m above sea level (gray shaded area). A solution using $T_I = 0.75$ (c) shows good agreement with the human interpretation (a) in the unconstrained regions.

minimum-curvature solution has large undulations and actually rises above sea level for a considerable portion (shaded area in Figure 5b). This "island" is certainly an unacceptable feature of this map.

DISCUSSION

It should now be clear that minimum curvature is not the ideal gridding method for all applications, something one might have expected from the start. Gridding with tension is an improvement because it adds a degree of freedom and relaxes the minimum-curvature constraint. We realize that there is no physical reason for using a plate-flexure equation to grid data such as gravity or topography measurements. Integral gridding methods can be designed for the physics of the particular application, such as the norms on harmonic splines used by Shure et al. (1982) for magnetic data. However, it is common for scientists to become familiar with one method of solution and use it in a variety of applications. Also, in the interpretation of potential field anomalies, the geophysicist must usually relate the anomaly to the topography of the source region; and he or she often wants to treat the potential field data and the topography data with the same procedure. A general gridding procedure is therefore a practical necessity.

The minimum-curvature method was advocated by Briggs (1974) for its smoothness properties, and Briggs (1974), Swain (1976), and Sandwell (1987) have all used it for potential-field data, which are expected to define relatively smooth analytic functions. The smoothest possible twice-differentiable surface should not approximate topography very well; our examples show that minimum curvature makes very poor bathymetric maps. It may be that kriging is a better approach, since topography data seem to be well suited to stochastic descriptions such as fractals and ARIMA models (e.g., Malinverno and Gilbert, 1989). However, the complete spatial autocorrelation of data given only along ship tracks is very difficult to compute and include in a simple kriging procedure. We feel that gridding with tension is an acceptable method for topography data: it is a more local scheme than minimum curvature and better reflects the nature of the autocovariance of topography data.

Isopleth data are a challenge for any algorithm. Obviously, one should not grid isopleth coordinates if the original data are available; but frequently gridded values are needed, and a contour map is the only published form of the data. Sometimes it is necessary to grid data which are intrinsically isopleths, such as when one makes a gridded age data set from the locations of magnetic isochrons of the sea floor. Isopleth coordinates contain some information about the locations of values of a function, but, more important, they define regions of bounded variation. We have shown that gridding with tension works well for isopleth data because the end member $T_I = 1$ in equation (8) cannot have local maxima or minima between constrained points. We do not know how kriging would work on isopleth data. The notion of the autocovariance of sets of equal values seems rather problematic.

The minimum-curvature gridding method has been widely applied to bathymetry and other data for which it is not well suited. We have shown that in some applications minimum curvature gives undesired results, and that including tension in the solution overcomes these difficulties. Since it is straightforward to modify a minimum-curvature algorithm to include variable tension, we suggest that earth scientists who already use minimum-curvature algorithms should include this feature. We do not claim that our method is ideal in all applications. However, we feel that continuous curvature gridding with adjustable tension provides the flexibility to handle many cases which arise in the earth sciences.

ACKNOWLEDGMENTS

We wish to thank N. W. Driscoll for allowing us to use his map of Broken Ridge (Figure 5a). W. F. Haxby and A. B. Watts brought to our attention the importance of regional fields and solutions to the related homogeneous equation. W. Menke suggested the temperature field analogy for the case $T_I = 1$. A. Lerner-Lam, A. Malinverno, P. G. Richards, and the reviewers and editors of GEOPHYSICS provided many helpful comments. This work was supported in part by Office of Naval Research contract TO-204 scope I and National Science Foundation contract no. OCE-86-14958.

Lamont-Doherty Geological Observatory contribution number 4555.

REFERENCES

Ahlberg J. H., Nilson, E. N., and Walsh, J. L., 1967, The theory of splines and their applications: Academic Press Inc.

Berg, P.W., and McGregor, J. L., 1966, Elementary partial differential equations: Holden-Day Inc.

Bracewell, R. N., 1978, The Fourier transform and its applications: McGraw-Hill International.

Brandt, A., 1984, Multigrid techniques: 1984 guide with applications to fluid dynamics: Gesellschaft für Mathematik und Datenverarbeitung mbH Bonn.

Briggs, I. C., 1974, Machine contouring using minimum curvature: Geophysics, **39**, 39–48.

Clark, I., 1979, Practical geostatistics: Applied Science Publ. Ltd.

Cline, A., 1974, Scalar and planar-valued curve fitting using splines under tension: Comm. ACM, **17**, 218–223.

Committee for the gravity map of North America, 1987, Gravity anomaly map of North America: Geol. Soc. Am.

Committee for the magnetic map of North America, 1987, Magnetic anomaly map of North America: Geol. Soc. Am.

Davis, J. C., 1986, Statistics and data analysis in geology: 2nd. ed., John Wiley & Sons, Inc.

de Boor, C., 1978, A practical guide to splines: Springer-Verlag New York, Inc.

Driscoll, N. W., Karner, G. D., Weissel, J. K., and the Shipboard Scientific Party, 1989, Stratigraphic and tectonic evolution of Broken Ridge from seismic stratigraphy and Leg 121 drilling: Initial Rep. Ocean Drilling Prog., **121**, 71–91.

Fulton, S. R., Ciesielski, P. E., and Schubert, W. H., 1986, Multigrid methods for elliptic problems: a review: Am. Meteorol. Soc. Monthly Weather Rev., **114**, 943–959.

Hackbush, W., and Trottenberg, U., Eds., 1982, Multigrid methods: Lecture notes in mathematics, **960**, Springer-Verlag.

Harbaugh, J. W., Doveton, J. H., and Davis, J. C., 1977, Probability methods in oil exploration: John Wiley & Sons, Inc.

Inoue, H., 1986, A least-squares smooth fitting for irregularly spaced data: Finite-element approach using the cubic B spline basis: Geophysics, **51**, 2051–2066.

Lancaster, P., and Salkauskas, K., 1986, Curve and surface fitting: Academic Press Inc.

Love, A. E. H., 1927, A treatise on the mathematical theory of elasticity, 4th ed.: Dover Publ. Inc.

Malinverno, A., and Gilbert, L. E., 1989, A stochastic model for the creation of abyssal hill topography at a slow spreading center: J. Geophys. Res., **94**, 1665–1675.

National Geophysical Data Center, 1988, ETOPO-5 Bathymetry/Topography Data, Data Announcement 88-MGG-02: National Oceanic and Atmospheric Administration, U.S. Dept. Commerce.

Olea, R. A., 1974, Optimal contour mapping using universal kriging: J. Geophys. Res., **79**, 696–702.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., 1986, Numerical recipes: Cambridge Univ. Press.

Richardson, L. F., 1910, The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam: Phil. Trans. R. Soc. London, Ser. A, **210**, 307–357.

Roache, P. J., 1982, Computational fluid dynamics: Hermosa Publ.

Sandwell, D. T., 1987, Biharmonic spline interpolation of GEOS-3 and SEASAT altimeter data: Geophys. Res. Lett., **14**, 139–142.

Schweikert, D. G., 1966, An interpolating curve using a spline in tension: J. Math. Physics, **45**, 312–317.

Shure, L., Parker, R. L., and Backus, G. E., 1982, Harmonic splines for geomagnetic modelling: Phys. Earth Plan. Int., **28**, 215–229.

Späth, H., 1973, Spline-Algorithmen zur Konstruktion glatter Kurven and Flächen: R. Oldenbourg Verlag; English translation by Hoskins, W. D., and Sager, H. W., 1974, Spline algorithms for curves and surfaces: Utilitas Mathematica Publ.

Swain, C. J., 1976, A FORTRAN IV program for interpolating irregularly spaced data using the difference equations for minimum curvature: Computers and Geosciences, **1**, 231–240.

Timoshenko, S., and Woinowsky-Krieger, S., 1968, Theory of plates and shells: 2nd ed., McGraw-Hill Book Co.

Van Wyckhouse, R., 1973, SYNBAPS, Tech. Rep. TR-233: U.S. National Oceanographic Office.

Watts, A. B., 1978, An analysis of isostasy in the world's oceans, 1. Hawaiian-Emperor seamount chain: J. Geophys. Res., **83**, 5989–6004.

Wegman, E. J., and Wright, I. W., 1983, Splines in statistics: J. Am. Stat. Assn., **78**, 351–365.

Wessel, P., 1989, XOVER: A cross-over error detector for track data: Computers and Geosciences, **15**, 333–346.

Wessel, P., and Watts, A. B., 1988, On the accuracy of marine gravity measurements: J. Geophys. Res., **93**, 393–413.

Young, D., 1954, Iterative methods for solving partial difference equations of elliptic type: Trans. Am. Math. Soc., **76**, 92–111.

APPENDIX

SOLUTION BY ITERATION OF FINITE-DIFFERENCE EQUATIONS

Difference expression for the homogeneous equation

Our notation is shown in Figure 2. For convenience we use subscripts to refer to the relative position of a grid node with respect to a local origin. Thus z_{00} refers to the current z_{ij}, and z_{1-1} appearing in an equation with z_{00} refers to z_{i+1j-1}. We approximate derivatives by central finite differences, e.g.,

$$\frac{\partial^2 z}{\partial x^2} \approx \frac{z_{10} - 2z_{00} + z_{-10}}{(\Delta x)^2}.$$

We normalize the length of the grid by assuming $\Delta x = 1$. We allow for anisotropy with an aspect ratio $\alpha = \Delta y/\Delta x$. Then

$$\frac{\partial^2 z}{\partial y^2} \approx \alpha^2[z_{01} - 2z_{00} + z_{0-1}]$$

and

$$\nabla^2 z \approx z_{10} + z_{-10} + \alpha^2(z_{01} + z_{0-1}) - 2(1 + \alpha^2)z_{00}. \tag{A-1}$$

Similarly,

$$\begin{aligned}\nabla^2(\nabla^2 z) \approx\ & z_{20} + z_{-20} + \alpha^4(z_{02} + z_{0-2}) \\ & + 2\alpha^2(z_{11} + z_{1-1} + z_{-11} + z_{-1-1}) \\ & - 4(1 + \alpha^2)[z_{10} + z_{-10} + \alpha^2(z_{01} + z_{0-1})] \\ & + (6 + 8\alpha^2 + 6\alpha^4)z_{00}.\end{aligned} \tag{A-2}$$

Using the finite-difference approximations (A-1) and (A-2), the homogeneous equation

$$(1 - T_I)\nabla^2(\nabla^2 z) - T_I\nabla^2 z = 0 \tag{A-3}$$

may be solved for z_{00}:

$$z_{00} = -[(6 + 8\alpha^2 + 6\alpha^4)(1 - T_I) + 2(1 + \alpha^2)T_I]^{-1}\left\{\begin{matrix}(1-T_I)[z_{20}+z_{-20}+\alpha^4(z_{02}+z_{0-2})+2\alpha^2(z_{11}+z_{-11}+z_{1-1}+z_{-1-1})] \\ -[4(1+\alpha^2)(1-T_I)+T_I][z_{10}+z_{-10}+\alpha^2(z_{01}+z_{0-1})]\end{matrix}\right\}. \tag{A-4}$$

We use this expression at those grid nodes z_{ij} which are not constrained by data. In the particular case of minimum curvature ($T = 0$) on an isotropic grid ($\alpha = 1$), this expression reduces to the difference expression given by Briggs (1974) as his equation (12).

Difference expression including a constraining datum

Briggs (1974) gave an expression for $\nabla^2 z$ at a grid point in terms of an off-grid constraining point. Since our equations may be expressed in $\nabla^2 z$, we used his approach, but modified it for our anisotropic grids. To find $\nabla^2 z$ at z_{00}, we construct a second-order Taylor series expansion from z_{00} to a point z_k:

$$\begin{aligned}z_k = z_{00} + \xi_k\frac{\partial z}{\partial x} + \eta_k\frac{\partial z}{\partial y} + \frac{1}{2}\xi_k^2\frac{\partial^2 z}{\partial x^2} \\ + \xi_k\eta_k\frac{\partial^2 z}{\partial x\,\partial y} + \frac{1}{2}\eta_k^2\frac{\partial^2 z}{\partial y^2}.\end{aligned}$$

We do this at five distinct points (ξ_k, η_k), $k = 1, 5$. Then we multiply each expansion by a real number b_k and sum the five expressions:

$$\sum b_k z_k = z_{00} \sum b_k + \sum b_k \xi_k \frac{\partial z}{\partial x} + \sum b_k \eta_k \frac{\partial z}{\partial y} + \frac{1}{2} \sum b_k \xi_k^2 \frac{\partial^2 z}{\partial x^2} + \sum b_k \xi_k \eta_k \frac{\partial^2 z}{\partial x\, \partial y} + \frac{1}{2} \sum b_k \eta_k^2 \frac{\partial^2 z}{\partial y^2}.$$

If the b_k are chosen such that

$$\sum b_k \xi_k = \sum b_k \eta_k = \sum b_k \xi_k \eta_k = 0 \quad \text{and}$$
$$\sum b_k \xi_k^2 = \sum b_k \eta_k^2 = 2$$

then

$$\nabla^2 z = \sum b_k z_k - z_{00} \sum b_k . \tag{A-5}$$

While any five points which yield a nonsingular expression for the b_k may be chosen, it is convenient to use four nearby grid node values and one off-grid constraint. For example, suppose that z_{00} is the location at the square in Figure A-1, and that we wish to implement the datum at E in Figure A-1 as a constraint. Let us assign $k = 1, 4$ to other points on the grid (A–D in Figure A-1), and $k = 5$ to the point E. Then we seek b_k satisfying

$$\begin{bmatrix} -1 & -1 & 0 & 1 & \xi \\ 1 & 0 & -1 & -1 & \alpha\eta \\ 1 & 1 & 0 & 1 & \xi^2 \\ -1 & 0 & 0 & -1 & \xi\alpha\eta \\ 1 & 0 & 1 & 1 & \alpha^2\eta^2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 2\alpha^2 \end{bmatrix},$$

where we have used the fact that the grid dimensions have been normalized by Δx and α is the anisotropy; here ξ and $\alpha\eta$ represent fractional distances on the grid,

$$\xi = \frac{(x_E - x_{00})}{\Delta x},$$

$$\alpha\eta = \frac{(y_E - y_{00})}{\Delta y}$$

or

$$\eta = \frac{(y_E - y_{00})}{\Delta x}.$$

The above matrix expression is easily solved to yield

$$b_5 = \frac{2(1+\alpha^2)}{(\xi + \alpha\eta)(1 + \xi + \alpha\eta)},$$
$$b_4 = 1 - \tfrac{1}{2} b_5 (\xi + \xi^2),$$
$$b_3 = \alpha\eta(1 + \xi) b_5 - 2b_4, \tag{A-6}$$
$$b_2 = \alpha\eta(1 + \xi) b_5 - b_3, \quad \text{and}$$
$$b_1 = \xi b_5 + b_4 - b_2 .$$

For a constraining datum in other quadrants, analogous expressions may be obtained.

The constraint is implemented by substituting equation (A-5) into equation (A-3) and solving for z_{00}:

$$z_{00} = \{T_I \sum b_k - 2(1-T_I)[(1+\alpha^4) - (1+\alpha^2) \sum b_k]\}^{-1} \left\{ (1 - T_I) \left\{ \begin{matrix} z_{20} + z_{-20} + \alpha^4(z_{02} + z_{0-2}) \\ + 2\alpha^2(z_{11} + z_{1-1} + z_{-11} + z_{-1-1} \\ -2(1+\alpha^2)[z_{10} + z_{-10} + \alpha^2(z_{01} + z_{0-1}) + \sum b_k z_k] \end{matrix} \right\} \begin{matrix} \\ -T_I \sum b_k z_k \end{matrix} \right\}. \tag{A-7}$$

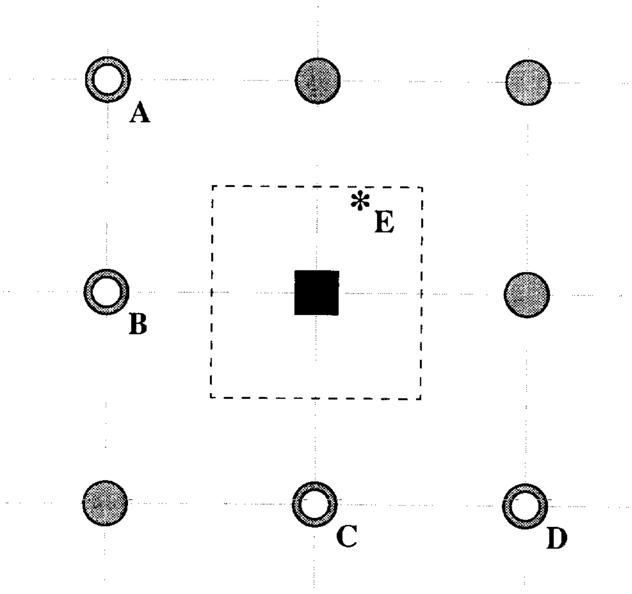

FIG. A-1. The grid node indicated by the black square is to be constrained by the datum at E. A Taylor series expansion is made from the node to the five points (A–E). Since four of these are "known" (they are other points on the grid which are solved in separate steps), they are used to eliminate terms in the Taylor series, leading to an expression for the Laplacian at the node which includes point E as a constraint [equation (A-5)]. When point E is in the first quadrant, A–D may be chosen as shown to yield equation (A-6). For E in another quadrant, A–D may be chosen by rotating this figure appropriately; this will modify the system for equation (A-6).

If the constraining datum (E in Figure A-1) lies exactly on the grid node, then ξ_E and η_E are both zero and the above matrix is singular. However, in this case the grid node value z_{00} may simply be replaced by z_E without using equation (A-7). In this way, the gridded surface always interpolates the constraining datum to second order in the Taylor series.

Difference expressions for boundary conditions

Application of equations (A-4) and (A-7) throughout the desired domain of (x, y) requires two additional outside rows or columns of auxiliary points (Figure A-2). We express the boundary conditions at an x edge here; the expression for the y edges are analogous.

We set the first outside points using boundary condition equation (9):

$$z_{-10} = \frac{2(1 - T_B)z_{00} - \left(1 - \frac{1}{2}T_B\right)z_{10}}{\left(1 - \frac{3}{2}T_B\right)}. \qquad \text{(A-8)}$$

After equation (A-8) has been applied, we use boundary condition equation (5) to set the auxiliary "corner" point:

$$z_{-1-1} = z_{1-1} + z_{-11} - z_{11}. \qquad \text{(A-9)}$$

The second outside points may then be set using boundary condition equation (4):

$$z_{-20} = z_{20} + \alpha^2(z_{11} + z_{1-1} - z_{-11} - z_{-1-1})$$
$$-2(1+\alpha^2)(z_{10} - z_{-10}). \qquad \text{(A-10)}$$

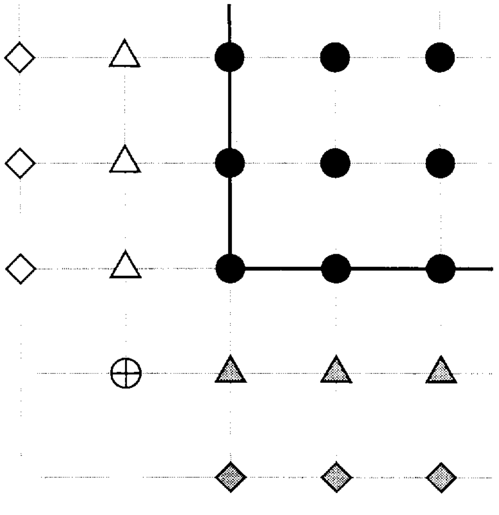

FIG. A-2. Implementing boundary conditions. Points in the lower left corner of the desired grid are represented by the black circles. The array containing the desired grid is augmented by two additional rows and columns surrounding each boundary. These exterior points allow application of equation (A-4) or equation (A-7) at every interior point. During each iteration, equation (A-8) is used to set the values of the white triangles, (A-9) the corner point (circle and cross symbol), and (A-10) the white diamonds. The grey triangles and diamonds on the y boundaries are set using analogous expressions but including the anisotropy factor α. Only one corner of the array is shown here but the entire boundary is set in a similar manner.

Solution by successive overrelaxation

We solve equations (A-4), (A-7), (A-8), (A-9), and (A-10) iteratively. One iteration consists of one application of the appropriate equation at each position z_{ij} to update each value. The updating is done immediately so that typically half of the points in the equation are "new" values and half are "old" values (Gauss-Seidel method; e.g., Press et al., 1986). For programming convenience, we visit these points in sequence in loops over the i and j indices. A nonunity anisotropy factor α effectively "couples" the equations more strongly in one dimension, and the convergence is more efficient if the loops are nested so that the strongly coupled direction is looped first.

The coefficients on the 12 points of Figure 2 are determined from equation (A-4) or equation (A-7) and are constant during the iteration process; only the z_{ij} change. We therefore compute arrays of these coefficients once, prior to entering the iteration loop. This prior step includes sorting the constraining data into the order in which they will be needed in the interaction loop, and computing and storing the b_k used with each constraint.

In the iteration loop itself, we use an overrelaxation parameter $1 < \omega < 2$ to accelerate convergence. The difference equation is used to compute a new value for z_{ij}, and the change in z_{ij} is increased by the overrelaxation factor:

$$z_{ij}^{\text{new}} \Leftarrow (1 - \omega)z_{ij}^{\text{old}} + \omega z_{ij}^{\text{new}}.$$

This method is called successive overrelaxation or simultaneous overrelaxation and is well known (Richardson, 1910; Young, 1954; Roache, 1982; Press et al., 1986). Spectral analysis of the iteration operator may, in theory, yield an optimal value for ω. In our application the best ω depends on the tension used; we have determined empirically that $\omega = 1.4$ works well for $T = 0$, and ω may be increased as T is increased. The system is considered "converged" to the limit ε when

$$\max |z_{ij}^{\text{new}} - z_{ij}^{\text{old}}| < \varepsilon.$$

Multiple grid strategy

We use a system of grids of various mesh sizes to enhance the efficiency of convergence of the system (A-4) and (A-7). We derived our method by a generalization of a technique in the minimum-curvature algorithm of Swain (1976). Our method shares some similarities with the multigrid methods developed for the second-order equations of fluid dynamics (Hackbush and Trottenberg, 1982; Brandt, 1984; Fulton et al., 1986).

In the iterative solution, the array z starts with some initial values which are then changed by an amount Δz when convergence is achieved. From equations (A-4) and (A-7) and Figure 2, it can be seen that in each iteration the new value computed for each z_{ij} is a weighted average of twelve neighboring values. The iteration operator is a local smoothing process, and as a consequence short-wavelength components of Δz are found quickly. Conversely, iteration does not efficiently propagate the effects of data constraints to long wavelengths. For this reason our algorithm does not begin iteration on the array which is ultimately desired; instead, we first find a long-wavelength solution on a coarser mesh consisting of every Nth point in the x and y dimensions of the final (desired) array. We begin with the largest N which divides both grid dimensions (and leaves at least four points in each direction so that some work needs to be done). The above system is solved to convergence on this sparse lattice. Then we divide N by its largest prime factor, exposing new nodes in a finer mesh. These new points are initialized by interpolation from the previous mesh, and then the system of equations on this new mesh is again iterated to convergence. We continue this cycle until $N = 1$ and the full system has been solved.

Multigrid techniques (Hackbush and Trottenberg, 1982; Brandt, 1984; Fulton et al., 1986) include both coarsening and fining mesh transfers in sequences called V-cycles. We use the simpler approach of starting with the coarsest convenient grid and successively fining (one half of one V-cycle). It can be shown (Ahlberg et al., 1967) that a coarse-mesh spline is the best estimator of a fine-mesh spline; in this sense we are starting with optimal initial values at each successive stage, since only short-wavelength perturbations to the solution need to be found. These local corrections are exactly what iteration of equations (A-4) and (A-7) performs efficiently. The computing time spent on the coarse meshes is small because the number of points in each lattice is only $1/N^2$ of the final number of points; the use of a series of meshes results in fewer iterations on the final ($N = 1$) stage and less total run time than if the solution had begun directly on the final grid. The coarse stages run so fast that we actually use ε/N as the convergence limit at each stage, where ε is the convergence limit set by the user for the final stage. This allows the user to choose a reasonable limit to be used on the final grid when the iterations are slow, but makes a better approximation of the long-wavelength components without much increase in total run time.

Equation (A-7) is constructed from the condition that the grid must interpolate the data constraints exactly (to second order in the Taylor series), and thus the prediction error of the surface would be zero if the equations could converge to $\varepsilon = 0$. In practice, we have observed that this sequence of coarse grids with division of N by its largest prime factor at each stage results in a smaller prediction error than any other solution strategies we tried. The minimum-curvature solver of Swain (1976) uses a similar sequence of coarse grids, except that his sequence of grid mesh N values is limited to powers of two and the initial N must be chosen by the user. Our algorithm allows any N and finds the initial N automatically. In map applications using a latitude-longitude mesh in minutes of arc, the grid dimensions commonly have factors of 3 and 5 as well as 2; and in these cases our mesh system goes through more intermediate stages than Swain's algorithm. We find that these extra states result in faster total run time, smaller prediction error, and better overall visual quality of the surface.

With the use of multiple grids, each lattice is initialized by an interpolation from the previous stage, and therefore only the first (coarsest) lattice needs to be seeded with initial values prior to iteration. In Swain's (1976) algorithm, this is done by a weighted average of points inside a user-specified radius. We have retained this feature as an option to be used when the grid dimensions have few common factors and the coarseness factor N starts near 1. However, because we have removed a planar trend from the data prior to iteration, we find that in most cases involving several regional grid stages it is adequate to begin with the coarse-lattice values initialized to zero.