



NRL/FR/7440--08-10,162

Algorithm Design Study for Bathymetry Fusion- Review of Current State-of-the-art and Recommended Design Approach

PAUL A. ELMORE

CHAD A. STEED

Mapping, Charting, and Geodesy Branch

Marine Geosciences Division

December 30, 2008

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 30-12-2008			2. REPORT TYPE Formal Report		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE Algorithm Design Study for Bathymetry Fusion - Review of Current State-of-the-art and Recommended Design Approach					5a. CONTRACT NUMBER	
					5b. GRANT NUMBER	
					5c. PROGRAM ELEMENT NUMBER 0603207N	
6. AUTHOR(S) Paul A. Elmore and Chad A. Steed					5d. PROJECT NUMBER	
					5e. TASK NUMBER	
					5f. WORK UNIT NUMBER 74-7441-J8-5	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory Marine Geosciences Division Stennis Space Center, MS 39529-5004					8. PERFORMING ORGANIZATION REPORT NUMBER NRL/FR/7440--08-10,162	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) SPAWAR (PEO C4I PMW 120) 4301 Pacific Highway San Diego, CA 92110					10. SPONSOR / MONITOR'S ACRONYM(S) SPAWAR PEO C4I	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					11. SPONSOR / MONITOR'S REPORT NUMBER(S)	
13. SUPPLEMENTARY NOTES						
14. ABSTRACT This report reviews current fusion techniques used for bathymetry or other geospatial data, as motivated by the Naval Oceanographic Office's (NAVOCEANO) need for intelligent fusion - combining two or more data sets in a manner that accounts for data uncertainty - of gridded and <i>in situ</i> bathymetric data sets. Based on this review, the recommended approach for building new bathymetry fusion algorithms uses loess interpolation to obtain a trend surface followed by kriging of residuals to recapture finer details lost from smoothing. In addition, if <i>in situ</i> soundings are used, Monte Carlo simulations are run to estimate depth error induced by position errors. The technique also provides the means to liberally estimate errors for navigation safety. The report concludes with plans to build, validate, and transition the algorithm to the Naval Oceanographic Office, Bathymetry Database Division.						
15. SUBJECT TERMS Bathymetry, Data fusion, Interpolation						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			Paul Elmore	
Unclassified	Unclassified	Unclassified	Unlimited	30	19b. TELEPHONE NUMBER (include area code) (228) 688-4613	

CONTENTS

Executive Summary	E-1
1. INTRODUCTION	1
2. REVIEW OF CURRENT STATE-OF-THE-ART	4
2.1 “Splines-In-Tension” Interpolation	4
2.2 Linear Smoothing by Locally Weighted Regression (Loess).....	6
2.3 Kriging.....	9
2.4 Bayesian Approaches	15
3. RECOMMENDED APPROACH	16
3.1 Quadratic Loess Interpolation for Trend Surface	18
3.2 Kriging the Residuals	19
3.3 Monte Carlo Estimation of Depth Error Due to Positional Stability Errors	21
4. ALGORITHM DEVELOPMENT, REQUIREMENTS, AND TEST CASES.....	22
5. SUMMARY	22
ACKNOWLEDGMENTS	23
REFERENCES	23

EXECUTIVE SUMMARY

This report reviews current fusion techniques used for bathymetry or other geospatial data, as motivated by the Naval Oceanographic Office's (NAVOCEANO) need for intelligent fusion – combining two or more data sets in a manner that accounts for data uncertainty – of gridded and *in situ* bathymetric data sets. Currently, NAVOCEANO's bathymetry database, DBDB-V, incorporates the Feathering Algorithm to smooth discontinuities that occur between tiles in the database with dissimilar resolutions. This technique, however, still leaves artifacts and does not provide uncertainty estimates. NAVOCEANO needs intelligent fusion capabilities not only to fuse data sets in a manner that takes the uncertainty of the data into account, but also to generate products that meet hydrography standards and to provide the capability to use one database for multiple purposes. A review of the technical literature indicates that current state-of-the-art fusion techniques that have been used on bathymetric data include splines-in-tension interpolation, locally weighted regression (loess), and kriging. In addition, there are new techniques based on Bayesian inference, but these appear to require further development before being ready for operational implementation. Based on this review, we recommend an approach for building new bathymetry fusion algorithms that was published and validated for bathymetry data recently by Calder*. This approach uses both loess interpolation to obtain a trend surface, followed by kriging of residuals to recapture finer details lost from smoothing. In addition, if *in situ* soundings are used, Monte Carlo simulations are run to estimate depth error induced by position errors. The technique also provides the means to liberally estimate errors for navigation safety. The Naval Research Laboratory (NRL) plans to build and validate a fusion algorithm based on this approach. The work leverages other NRL efforts that developed data fusion capabilities using loess interpolation and adds in additional required components in the build schedule. The algorithm will support the new Bathymetric Attributed Grid (BAG) format of the Open Navigation Surface Project. NRL plans to transition the software to the Naval Oceanographic Office, Bathymetry Database Division.

ALGORITHM DESIGN STUDY FOR BATHYMETRY FUSION – REVIEW OF CURRENT STATE-OF-THE-ART AND RECOMMENDED DESIGN APPROACH

1. INTRODUCTION

The Digital Bathymetry Data Base – Variable Resolution (DBDB-V) is the bathymetric database maintained by the Naval Oceanographic Office (NAVOCEANO) for worldwide bathymetry data. The latest release, Version 5.2, has global coverage at 2 minutes of resolution, primarily based on satellite altimetry data of the ocean surface (which then provides bathymetry from the inferred gravitational anomalies) and ship soundings [1]. Higher resolution data in selected areas also are contained in the database in resolutions of 1, 0.5, 0.1, and 0.05 [2] arc minutes. DBDB-V stores these data sets in rectangular grid tiles.

When the end user extracts data for an area covered by multiple tiles with different resolutions using nearest neighbor interpolation or bicubic spline interpolation, there are discontinuity artifacts at tile boundaries. These discontinuities may be caused by differences in spatial frequencies or data accuracy across tile boundaries or both. As a result, as discussed in Steed and Rankin [3], false cliffs appear in the extracted data at tile boundaries (Fig. 1). The artifacts have negative visual impact and cause errors in oceanographic and acoustic modeling. To mitigate these errors, the OAML Feathering Algorithm [3] was implemented in DBDB-V Version 5.2 to smooth discontinuities at the boundaries. The algorithm produces a minimum-curvature spline grid for the extracted area using the “mb_zgrid” C function of the MB-System software suite [4] (open-source software used for processing and visualizing bathymetric and acoustic backscatter data). As shown in Steed and Rankin [3], the algorithm smoothes discontinuities between dissimilar tiles and should reduce errors for modeling applications. Artifacts, although smoothed, remain. For example, some false cliffs become ramps in between transition regions (Fig. 2). The OAML Feathering Algorithm, however, was not intended for permanent use; it was a “stop-gap measure” to be used until intelligent fusion algorithms could be created.

Intelligent fusion differs fundamentally from feathering in that the uncertainty of the data points is taken into account when generating an interpolation surface from different data sets. The inclusion of uncertainty (or error) allows for the use of different techniques that weight the data using the errors and provides error estimates with the bathymetry. Hence, the products of intelligent fusion are a bathymetry layer and an uncertainty layer.

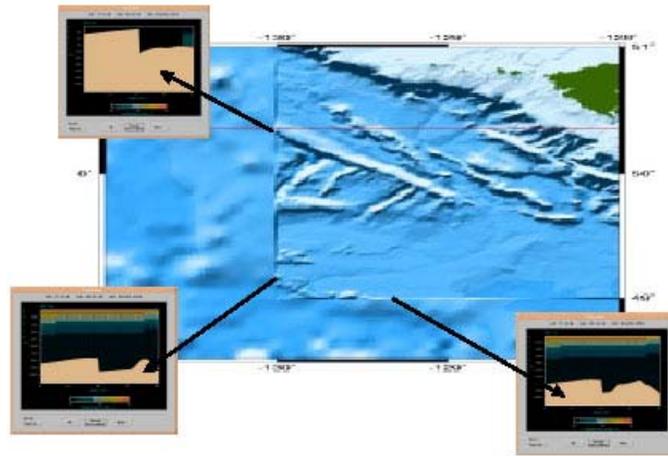


Fig. 1 – Artifact discontinuities between DBDB-V Version 3.0 tiles of dissimilar resolutions (5.0 minutes in coarse region, 0.5 minutes in finer region) [3].

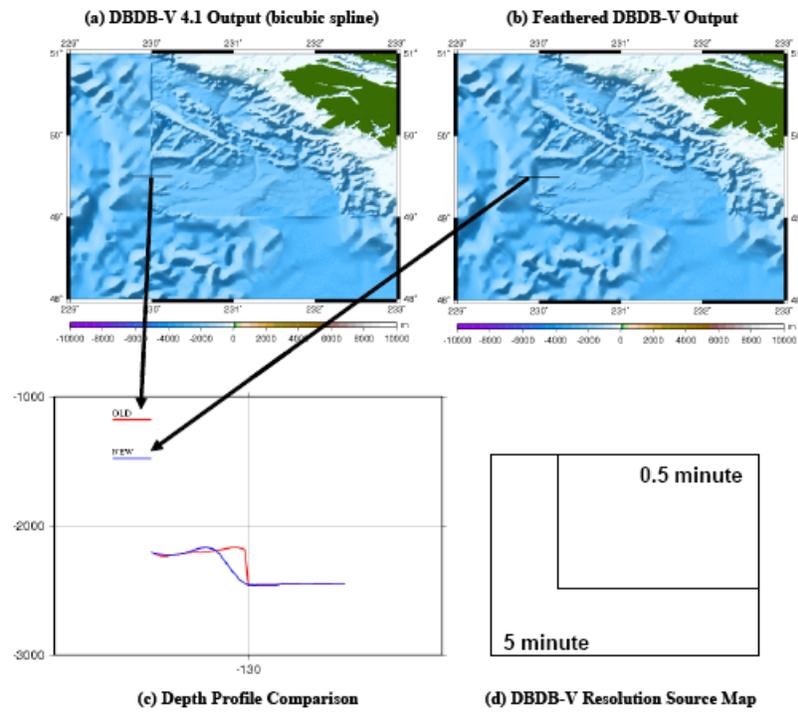


Fig. 2 – Example improvement of artifact discontinuities using the OAML Feathering Algorithm on tiles from Fig. 1 [3].

This result is desirable for a number of reasons. First, data points with smaller errors are more important to surface generation. In an era where highly accurate multibeam echo sounder systems are available, it is desirable to give more credence to these data than to data collected in an era with vertical beam echo sounders (or older technology) and higher uncertainty in navigation. Second, specification of

uncertainty allows for multiple use of a bathymetry database, so that navigation charts can be generated from the same database (by shoaling the data with an appropriate uncertainty level) that would be used by modelers. Third, modelers can, in principle, provide error estimates of predictions based on propagation of bathymetry errors through the models. Lastly, uncertainty in the interpolated points is required for International Hydrography Organization (IHO) standards [5]. Section 5.3 of the fifth edition (draft) dictates:

All data should be attributed with a 95% statistical error estimate for position and depth where appropriate. For soundings this should preferably be done for each individual sounding; however a single error estimate may be recorded for a number of soundings or even for an area, provided any difference between the individual error estimates can be safely expected to be negligible.

Quantifying uncertainty in sparse geospatial data sets has become a focus of research within the past 5 to 7 years. Increased computational capabilities have made it possible to collect high density and high volume data sets, which allows for error estimation based on statistical techniques. Also, navigational and sonar uncertainties have become lower, resulting in higher precision in the data obtained. Both of these factors and others are used by a new data processing algorithm, the Combined Uncertainty and Bathymetric Estimator (CUBE) for providing robust estimates of bathymetry and uncertainty [6]. With regard to product generation, traditional interpolation techniques for sparse data sets have become more pragmatic with newer, more efficient algorithms and increased computational speed to allow for the use of newer robust estimations with more computationally expensive procedures.

In response to the IHO requirements and newer error estimation capabilities, a new data format, the Bathymetric Attributed Grid (BAG) [7], has been designed by the Open Navigation Surface Working Group, a consortium of academic, government and private sector groups. Central to the BAG's design is the requirement that it hold both bathymetry and uncertainty data for rectangular bathymetric tiles. This format has now been adopted by two commercial software suites used for bathymetric data processing and analysis, and is now being used by NAVOCEANO to store output from new multibeam data sets processed by CUBE. Thus, the adoption of intelligent procedures for fusing bathymetry data sets is now not only a pragmatic possibility but also a necessity.

The purpose of this report is to provide a starting point for providing NAVOCEANO with intelligent fusion availabilities for current and future bathymetry data sets. In Section 2, we first review the current state-of-the-art techniques for intelligent fusion of sparse data sets, some of which were developed for bathymetry data. This section not only serves as a review but also provides background for describing this reports recommended technique, which is a new three-step approach by Calder [8]. Section 3 reviews this approach. Section 4 provides discussion of software build design and case studies to use for validation. Finally, Section 5 provides summary remarks.

2. REVIEW OF CURRENT STATE-OF-THE-ART

As discussed in Davis [9] and reiterated by Smith and Wessel [10], algorithms that generate gridded surfaces from sparse data sets assume the following:

- a) Grid points are single-valued.
- b) Continuity exists everywhere on the generated surface.
- c) Autocorrelation of the surface is positive for distance greater than the node spacing.

These assumptions should be maintained by intelligent fusion algorithms. In order to satisfy condition one, the point values, which will now be a statistical quantity with an estimated uncertainty, can be a mean or median.

Crain [11] and Franke [12] provide reviews of earlier interpolation techniques. These methods include polynomial interpolation, inverse-distance weighting, triangulation, piecewise contour line segments, simple finite difference and finite element methods. These techniques are not reviewed again here. The newer techniques discussed below have either been applied to bathymetry or have some form of robust uncertainty estimation, or both. We first discuss “splines-in-tension,” which is used for the OAML Feathering Algorithm and has been applied to bathymetry data since the 1970s; uncertainty estimation was added-on in 2002. Next, are reviews of loess interpolation and kriging; both fundamentally provide uncertainty estimation. Finally, we briefly preview possible future techniques based on Bayesian inference.

2.1 “Splines-in-Tension” Interpolation

Spline interpolation constructs curves between data points to generate a gridded surface. “Splines-in-tension” is a specific technique that calculates a global solution by solving a differential equation. Constraints often include data points, some desired mathematical property (e.g., continuous first and second derivatives, minimum curvature, etc.), and boundary conditions at the end points. The gridded surface is then calculated by finite differences.

2.1.1 Methodology

The geospatial community uses techniques by Briggs [13] and Smith and Wessel [10] to calculate the interpolation surface. The elasticity equation for thin plates derived by Love [14] is the basis for generating the interpolation surface, which is modeled as an elastic plate that is constrained at the locations of the data points. Under this model, the differential equation to be solved is [10]

$$(1 - T)\nabla^2(\nabla^2 z) - T\nabla^2 z = \sum_i f_i \delta(x - x_i, y - y_i), \quad (1)$$

where T is the tension; z is the two-dimensional interpolation surface; $f_i = z(x_i, y_i)$, the i^{th} known value for z at coordinate (x_i, y_i) as $(x, y) \rightarrow (x_i, y_i)$; and $\delta(x - x_i, y - y_i)$ is the Dirac delta function. In Briggs’s “minimum-curvature” technique, $T = 0$ so that Eq. (1) becomes

$$\nabla^2(\nabla^2 z) = \frac{\partial^4 z}{\partial x^4} + \frac{\partial^4 z}{\partial x^2 \partial y^2} + \frac{\partial^4 z}{\partial y^4} = \sum_i f_i \delta(x - x_i, y - y_i) \quad (2)$$

The solution to the surface is one where sum of the curvature of the surface is minimized (as the potential energy of the plate is minimized in this case), and found by imposing the conditions of continuous first and second derivatives, known fixed values at the data points and constraints at the boundaries. Briggs provides a finite difference solution to Eq. (2), which was later coded into Fortran-IV by Swain [15]. This minimum-curvature method was used for earlier versions of DBDB [16].

Smith and Wessel [10] recognized that the solution for Eq. (2) can produce false minima and maxima. They corrected for this potential for artifacts by adding the tension term back into the equations to be solved and provided a solution for Eq. (1) through finite-difference iteration.

2.1.2 Available Software

Smith and Wessel's algorithm is coded in the SURFACE function in the Generic Mapping Tools (GMT) [17] software suite. A similar splines-in-tension algorithm is in the MB_ZGRID in the MB-System [5] suite. As both are widely used open-source software packages.

2.1.3 Advantages and Disadvantages

An advantage to the technique is that it is relatively fast and is exact at the points where there is data. Also, a solution is found globally. A disadvantage is that the setting for the tension term is arbitrary. This tension may need to be set to different values in different settings. For this reason, the OAML Feathering Algorithm sets the tension to the value recommended by the authors of the code. An additional disadvantage is that these routines, as noted by Smith and Wessel, do not provide error estimates for the interpolated grid points.

2.1.4 Error Estimation from Monte Carlo Simulation

Jakobsson et. al. [18] developed a Monte Carlo technique for error estimation for spline-in-tension interpolation. The technique uses positional accuracies of the ships that recorded the soundings to estimate a Gaussian probability density function for horizontal positioning error. Using a subset of bathymetric data from the International Bathymetric Chart of the Arctic Ocean (IBCAO) [19] for validation, the horizontal positions of the soundings are randomly perturbed and new interpolated surfaces are generated. Since the constraining positions are different for each iteration of the simulation, the calculated solution to Eq. (1) changes at the interpolation points as well. After the simulation is complete, an uncertainty layer is then created from the standard deviations of the solutions for each grid point. This error estimation technique is discussed in Section 3 for uncertainty estimation induced by horizontal positioning error.

2.2 Linear Smoothing by Locally Weighted Regression (Loess)

Linear smoothing techniques construct the interpolated surface, $\hat{z}(\mathbf{s})$, $\mathbf{s} \equiv$ matrix of (x,y) coordinates, from a linear weighted average of known data values. Unlike the splines-in-tension technique that finds a global solution to the available data, this technique obtains an interpolated value at a grid point by using only a subset of neighboring points. Mathematically,

$$\hat{z}(\mathbf{s}) = \sum_i \alpha(\mathbf{s} - \mathbf{s}_i) z(\mathbf{s}_i), \quad (3)$$

where $\alpha(\mathbf{s} - \mathbf{s}_i)$ specifies the smoother coefficients and the index i corresponds to the subset of local points, $z(\mathbf{s}_i)$, to be used for the interpolated value at \mathbf{s} , $\hat{z}(\mathbf{s})$. A common technique used to specify these coefficients is locally weighted regression (or “loess”), first published in Cleveland [20] and further developed in Cleveland and Devlin [21]. A recent textbook by Givens and Hoeting [22] also discusses the technique. The methodology determines the smoother coefficients from a weighted least squares polynomial (linear or quadratic) fit of windowed data.

2.2.1 Methodology

To summarize Cleveland’s methodology, the two-dimensional case is considered. (We extend these equations to the three-dimensional case in Section 3; the methodology remains the same.) Let data points x_i and y_i be related as

$$y_i = g(x_i) + \varepsilon_i, \quad (4)$$

where $g(x)$ is a smooth function and ε_i is Gaussian noise with mean 0 and variance σ^2 . Thus we can define \hat{y}_i to be the estimate of $g(x_i)$ (i.e., $\hat{y}_i \approx g(x_i)$). Let

- the number n be the predetermined number of data points to be used for estimating \hat{y}_i
- the set $x_k, k = 1, \dots, n$, be the subset of x_j ’s ($j = 1, \dots, N; N =$ total number of data points, $n < N$) that are closest to x_i
- the distance h_i be the distance from x_i to the furthest x_k
- the windowing weights to be used for the regression, $w_k(x_i) = W([x_k - x_i]/h_i)$, where $W(x)$ is the tricube function, defined as

$$W(x) \equiv \begin{cases} (1 - |x|^3)^3, & x < 1 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

With these definitions, the loess procedure calculates the set of polynomial coefficients, $\hat{\beta}_l(x_i)$, which are the values for the β_l ’s that minimize

$$q(x_i) = \sum_{k=1}^n w_k(x_i) (y_k - \beta_0 - \beta_1 x_k - \dots - \beta_d x_k^d)^2, \quad (6)$$

where $q(x_i)$ is the error function. The interpolated value for $g(x_i)$ is then [20]

$$\hat{y}_i = \sum_{\ell=0}^d \hat{\beta}_\ell(x_i) x_i^\ell = \sum_{k=1}^n \alpha_k(x_i) y_k, \quad (7)$$

where the smoothing terms on the right-hand side correspond to the right-hand side of Eq. (3) rewritten to the present two-dimensional case. Some points of interest are as follows.

- The coefficients obtained have a decreasing trend to the edge of the window so that the data centered in the window generally have the greatest influence.
- The size of the polynomial typically used in Eq. (6) is $d = 1$ or 2 . For $d = 0$, the result is a simple moving average. The $d = 1$ case is called linear loess or “lowess” smoothing, and $d = 2$ is quadratic “loess” smoothing.
- Cubic and higher fits typically are not used as the fits can become over fitted and numerically unstable [23].
- This technique also has a robustness option, so that the interpolation can be shielded from the effects of outliers in the data (details in Cleveland [20]).
- Cleveland choose the tricube weighting function because it allowed the estimate of the error variance to be approximated by a chi-square distribution and usually lowered the variance of the estimate surface as the number of points used for the estimate increased.

Errors propagated into the interpolation by the technique are straight forward to compute. Under the assumption the data follow Eq. (4), the estimate of the variance for \hat{y}_i , $\hat{\sigma}_i^2$, is [20, 21, 24]

$$\hat{\sigma}_i^2 = \sigma^2 \sum_{k=1}^n [\alpha_k(x_i)]^2, \quad (8)$$

which is derivable from independent error propagation.

2.2.2 Equivalent Kernel Approach

Figure 3 plots the set of α_k 's as determined for a centered impulse response using the linear and quadratic loess interpolators [25]. These coefficients are the smoother weights in Eq. (7) and are also called the “equivalent kernel.” They depend only on the grid points, and, due to their finite width, act as a window or low-pass filter function on the spatial data (i.e., the smoothing weights and the data undergo convolution). Hence, one could bypass solving a weighted least square problem and simply compute the convolution of the smoothing window with the data, which should be computationally faster. In addition, other windowing functions could be used to perform the smoothing.

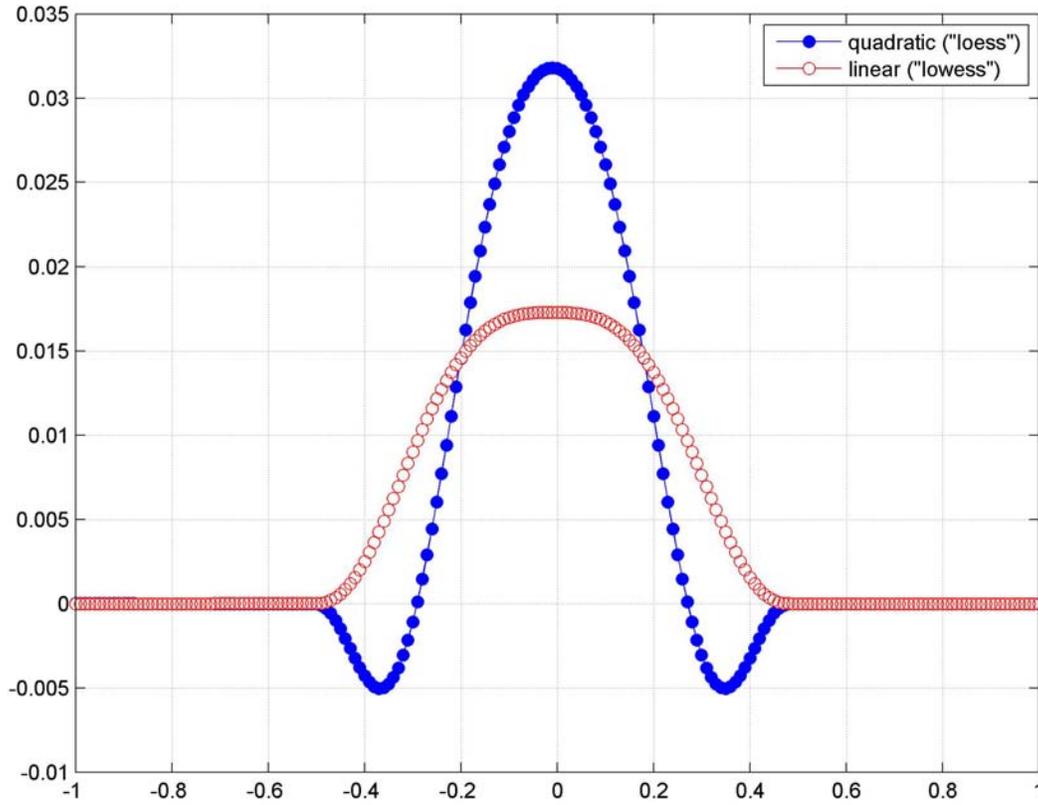


Fig. 3 – Equivalent kernel weights for linear and quadratic loess windows for window size of 0.5

The use of other weighting functions would also allow one to interpolate over data points with differing accuracy. In this case, Eqs. (7) and (8) become

$$\hat{y}_i = \sum_{k=1}^n \alpha_k(x_i) \left(\sigma_k^{-2} / \sum_{k=1}^n \sigma_k^{-2} \right) y_k = \sum_{k=1}^n \tilde{\alpha}_k(x_i) y_k, \quad (9)$$

$$\hat{\sigma}_i^2 = \sum_{k=1}^n \sigma_k^2 \tilde{\alpha}_k^2(x_i) \quad (10)$$

where σ_k^2 is the variance of y_k and $\tilde{\alpha}_k(x_i) \equiv \alpha_k(x_i) \left(\sigma_k^{-2} / \sum_{k=1}^n \sigma_k^{-2} \right)$.

2.2.3 Available Software

Cleveland et al. [26] have written publicly available Fortran and C code for loess interpolation. The technique is also part of the publicly available LOCFIT software package [27]. The commercially available Curve Fitting Toolbox for MATLAB [28] also contains loess interpolation routines.

2.2.4 Advantages and Disadvantages

An advantage of this procedure is that it does not require an *a priori* model or other function to be specified; thus, it is good for modeling surfaces that have complicated topologies [23]. A disadvantage of the loess technique is that it is not exact at the data points and the interpolated bathymetry data can be subject to spatial aliasing, particularly when the data is not uniformly sampled or when there are features not resolved by the sonar system [24]. As a result, additional variance and bias occurs.

Schlax and Chelton [29] examined the aliasing issue for equivalent kernels reviewed by Buja et al. [30], which included running means, linear interpolation, Gaussian windows and cubic splines in addition to linear and quadratic loess kernels. When the data points are evenly sampled, the transmitted errors are lowest for the quadratic loess kernel when interpolation occurs far from grid boundaries. Sparse sampling induces more errors due to aliasing by a factor of between 4 and 8, regardless of the kernel used, with the quadratic loess showing the most increase in error. Nonetheless, in Schlax and Chelton's examples, the quadratic loess kernel still did the best for all but one (linear interpolation) of the kernels. Plant et al. [24] point out that the increased sensitivity is due primarily to the negative sidelobes in the weights.

2.3 Kriging

Kriging is a standard technique for interpolating sparse geospatial data. Geospatial data analysis textbooks, such as Davis [9] and O'Sullivan and Unwin [31], and the monograph by Journel [32] present the technique. Cressie [33] and Chiles [34] provide advanced mathematical treatments. The method was first introduced by Krige [35] and developed by Matheron [36]. Like the linear smoothing technique discussed above, it is a linear regression technique where interpolated values are estimated from a weighted sum of neighboring data points; however, the methodology for finding the weights relies on solving a system of simultaneous linear equations (instead of minimizing an overdetermined system) whose terms are the covariances or, alternatively, variograms of the data. Thus, before discussing the relevant kriging equations for bathymetry, we first introduce covariance and variograms.

2.3.1 Covariance and Variogram Matrices

The covariance, in terms of bathymetry measurements at \mathbf{s}_i and \mathbf{s}_j , $z(\mathbf{s}_i)$ and $z(\mathbf{s}_j)$ respectively, is [33]

$$\text{cov}(\mathbf{s}_i, \mathbf{s}_j) = E\{z(\mathbf{s}_i)z(\mathbf{s}_j)\} - E\{z(\mathbf{s}_i)\}E\{z(\mathbf{s}_j)\} \quad (11)$$

where the operator $E\{\cdot\}$ is the expectation value of the enclosed term. In matrix form

$$\mathbf{C} = \begin{bmatrix} \text{cov}(\mathbf{s}_1, \mathbf{s}_1) & \text{cov}(\mathbf{s}_1, \mathbf{s}_2) & \cdots & \text{cov}(\mathbf{s}_1, \mathbf{s}_n) \\ \text{cov}(\mathbf{s}_2, \mathbf{s}_1) & \text{cov}(\mathbf{s}_2, \mathbf{s}_2) & \cdots & \text{cov}(\mathbf{s}_2, \mathbf{s}_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(\mathbf{s}_n, \mathbf{s}_1) & \text{cov}(\mathbf{s}_n, \mathbf{s}_2) & \cdots & \text{cov}(\mathbf{s}_n, \mathbf{s}_n) \end{bmatrix} \quad (12)$$

Note that for $i = j$, $\text{cov}(\mathbf{s}_i, \mathbf{s}_j)$ is the variance of $z(\mathbf{s}_i)$, $\text{var}(z(\mathbf{s}_i)) = \sigma_{z(\mathbf{s}_i)}^2$. When $z(\mathbf{s})$ is statistically stationary, then the covariance simply depends on the displacement \mathbf{h} from \mathbf{s} (i.e., the variance is independent of position so that $E\{z(\mathbf{s} + \mathbf{h})\} = E\{z(\mathbf{s})\}$). Thus for stationary conditions [37],

$$\text{cov}(\mathbf{h}) = E\{z(\mathbf{s} + \mathbf{h})z(\mathbf{s})\} - [E\{z(\mathbf{s})\}]^2. \quad (13)$$

The variogram, 2γ , is [33]

$$\begin{aligned} 2\gamma(\mathbf{s}_i, \mathbf{s}_j) &\equiv \text{var}(z(\mathbf{s}_i) - z(\mathbf{s}_j)) \\ &= \text{var}(z(\mathbf{s}_i)) + \text{var}(z(\mathbf{s}_j)) - 2\text{cov}(\mathbf{s}_i, \mathbf{s}_j) \end{aligned} \quad (14)$$

The semivariance is half the variogram, or simply γ . Under stationary conditions, Eq. (14) becomes [37],

$$\gamma(\mathbf{h}) = \text{cov}(0) - \text{cov}(\mathbf{h}), \quad (15)$$

where $\text{cov}(0) = \text{var}\{z(\mathbf{s})\}$.

Often, the semivariance is of more practical use than the covariance because an estimate for the semivariance is straightforward to compute from observations, particularly along one dimension. Consider n such observations made at regular intervals. From Eq. (14), the empirical estimate of the semivariance, $\hat{\gamma}(\mathbf{s}_i, \mathbf{s}_j) \approx \gamma(\mathbf{s}_i, \mathbf{s}_j)$, for displacement h , $\hat{\gamma}(\mathbf{s}_i, \mathbf{s}_{i+h}) = \hat{\gamma}(h)$, is

$$\hat{\gamma}(h) = \sum_i^{n-h} (z(\mathbf{s}_i) - z(\mathbf{s}_{i+h}))^2 / 2n. \quad (16)$$

Qualitatively, the semivariogram (Fig. 4), the plot of the semivariance, generally increases with h from a minimum at $h = 0$, which may or may not be zero ($\gamma(0) \neq 0$ is called the “nugget effect”), to a horizontal line, the “sill,” which is equal to overall variance of the observations. This qualitative trend holds when the data is a regionalized variable. A regionalized variable can be considered as a hybrid between deterministic and random variables, where a high amount of correlation exists between two samples that are taken close to each other spatially, but this correlation degenerates as the spatial distance increases until a distance, the “range,” is reached where further observations appear to be random. Thus, the variogram remains at the sill for values of h greater than the range as any correlation between data points is lost.

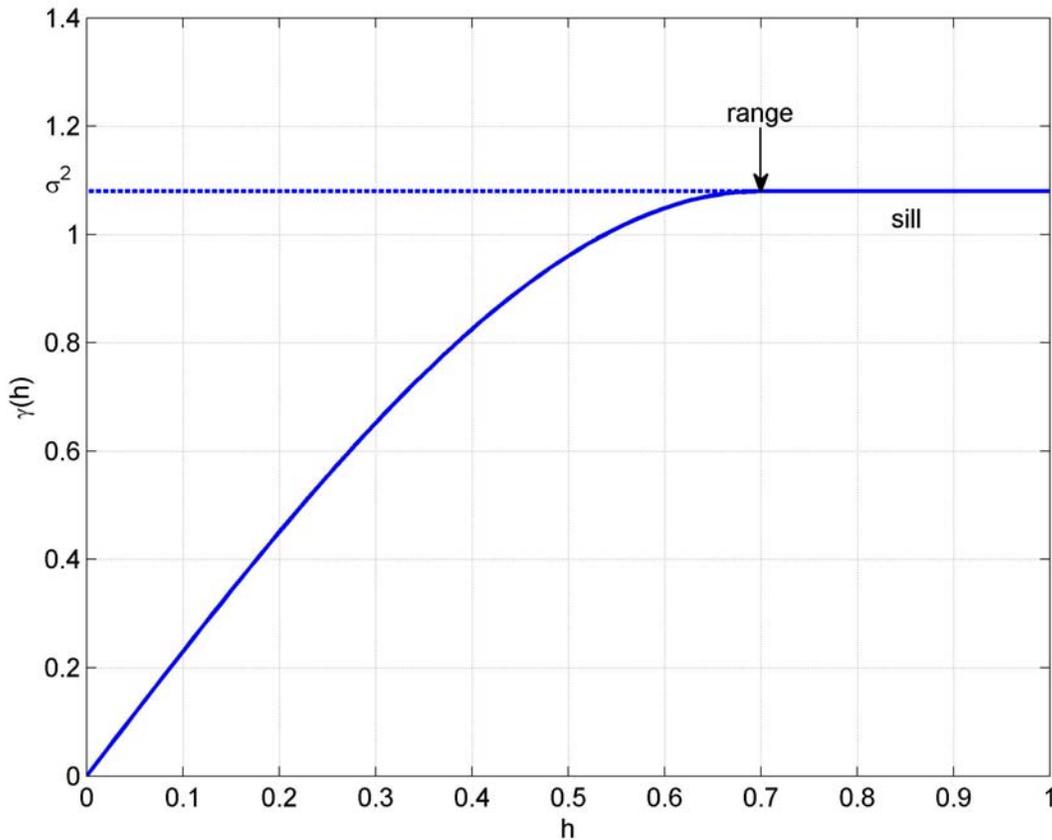


Fig. 4 – Example semivariogram with no nugget effect and the range and sill noted. At the sill, the semivariogram equals the overall variance of the observations. This semivariogram is a plot of the spherical model, Eq. (44) below, with $a_0 = 0$, $a_1 = 1.08$, and $a_2 = 0.7$. Adapted from Davis [9].

This qualitative behavior makes it convenient to fit experimental variograms to modeled ones. We will define one such model, called the “spherical” model, in Section 3.2 below. Other models include those based on linear, exponential, or Gaussian functions. Cressie [33] and Davis [9] present these and other models. Once the variogram is modeled and fitted to the data, the modeled semivariance, or its translation to a covariance, is then used to solve the kriging equations, which is the focus of discussion in the next few sections.

2.3.2 General Kriging Equation

All kriging methodologies used for the interpolation of two-dimensional surfaces have a similar general set of linear equations [37]:

$$\hat{z}(\mathbf{s}) - E\{\hat{z}(\mathbf{s})\} = \sum_{i=1}^n \lambda_i [z(\mathbf{s}_i) - E\{z(\mathbf{s}_i)\}]. \quad (17)$$

The weights, λ_i , are found by minimizing the square of the difference between the two sides of Eq. (17). Depending on assumptions that can be made about the expectation values, different sets of equations for solving the weights (or types of kriging) result. We consider three types of kriging below: simple kriging (SK), ordinary kriging (OK), and universal kriging (UK). (Other forms of kriging – such as block kriging,

cokriging, nonlinear kriging – are outside the scope of this review.) These three types of kriging progressively become more complicated (and with larger variance) as fewer assumptions about the interpolated surface are applicable. Although SK may not be as applicable to bathymetry estimation as the other two, its discussion facilitates the review of OK and UK. The reviews below are adapted primarily from Davis [9] but augmented with other material from Cressie [33] and Deutsch and Journel [37].

2.3.3 Simple Kriging (SK)

In SK, the mean of the data, m , is known and a constant. Hence, $E\{z(\mathbf{s})\} = m$ for all \mathbf{s} in Eq. (17). In addition, covariances and variances are assumed to be independent of position. These assumptions imply that Eq. (17) reduces to

$$\hat{z}_{SK}(\mathbf{s}) = m \left(1 - \sum_{i=1}^n \lambda_i \right) + \sum_{i=1}^n \lambda_i z(\mathbf{s}_i), \quad (18)$$

where the SK subscript means simple kriging estimate. The optimal set of weights at position \mathbf{s} is found from the simultaneous solutions to the equations (see Journel [32] for derivation)

$$\begin{aligned} \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_1) &= \text{cov}(\mathbf{s}, \mathbf{s}_1) \\ \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_2) &= \text{cov}(\mathbf{s}, \mathbf{s}_2) \\ &\vdots \\ \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_n) &= \text{cov}(\mathbf{s}, \mathbf{s}_n) \end{aligned} \quad (19)$$

or in matrix format $\mathbf{C}\boldsymbol{\Lambda}_{SK} = \mathbf{B}_{SK}$, where

$$\begin{aligned} \boldsymbol{\Lambda}_{SK} &= [\lambda_1, \lambda_2, \dots, \lambda_n]^T \\ \mathbf{B}_{SK} &= [\text{cov}(\mathbf{s}, \mathbf{s}_1), \dots, \text{cov}(\mathbf{s}, \mathbf{s}_n)]^T \end{aligned} \quad (20)$$

and T means matrix transposition.

Thus, $\boldsymbol{\Lambda}_{SK} = \mathbf{C}^{-1}\mathbf{B}_{SK}$. Defining $\mathbf{Y}_{SK} = [z(\mathbf{s}_1) - m, \dots, z(\mathbf{s}_n) - m]^T$, Eq. (18) becomes

$$\hat{z}_{SK}(\mathbf{s}) = m + \mathbf{Y}_{SK}^T \boldsymbol{\Lambda}_{SK} = m + \mathbf{Y}_{SK}^T \mathbf{C}^{-1} \mathbf{B}_{SK}. \quad (21)$$

Equation (21) is the solution for the interpolation surface at \mathbf{s} . The estimated variance is

$$\sigma_{SK}^2(\mathbf{s}_o) = \text{var}(z(\mathbf{s}_o)) - \mathbf{B}_{SK}^T \mathbf{C}^{-1} \mathbf{B}_{SK}. \quad (22)$$

2.3.4 Ordinary Kriging (OK)

In OK, the assumption that $E\{z(\mathbf{s})\} = m$ is maintained, but now m is unknown. Hence, the system of simultaneous equations needs an additional constraining condition to obtain a solution. This condition is

that the sum of the weights is one: $\sum_{i=1}^n \lambda_i = 1$. Then, Eq. (18) becomes (with the *SK* subscript replaced with *OK* for ordinary kriging estimate)

$$\hat{z}_{OK}(\mathbf{s}) = \sum_{i=1}^n \lambda_i z(\mathbf{s}_i). \quad (23)$$

The set of equations now needing to be solved are equivalent if either the covariance or semivariance are used (see Journel [32] or Cressie [33] for derivation).

$$\begin{aligned} \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_1) + \mu_0 &= \text{cov}(\mathbf{s}, \mathbf{s}_1) \quad \text{or} \quad \sum_{i=1}^n \lambda_i \gamma(\mathbf{s}_i, \mathbf{s}_1) + \mu_0 = \gamma(\mathbf{s}, \mathbf{s}_1) \\ \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_2) + \mu_0 &= \text{cov}(\mathbf{s}, \mathbf{s}_2) \quad \text{or} \quad \sum_{i=1}^n \lambda_i \gamma(\mathbf{s}_i, \mathbf{s}_2) + \mu_0 = \gamma(\mathbf{s}, \mathbf{s}_2) \\ &\vdots \\ \sum_{i=1}^n \lambda_i \text{cov}(\mathbf{s}_i, \mathbf{s}_n) + \mu_0 &= \text{cov}(\mathbf{s}, \mathbf{s}_n) \quad \text{or} \quad \sum_{i=1}^n \lambda_i \gamma(\mathbf{s}_i, \mathbf{s}_n) + \mu_0 = \gamma(\mathbf{s}, \mathbf{s}_n) \\ \sum_{i=1}^n \lambda_i &= 1 \end{aligned} \quad (24)$$

where μ_0 is a Lagrange multiplier, which is a slack variable used to raise the number of unknowns from n to $n+1$ for the system on $n+1$ equations. In matrix form, Eq. (24) is $\mathbf{W}\mathbf{\Lambda}_{OK} = \mathbf{B}_{OK}$, where

$$\begin{aligned} \mathbf{\Lambda}_{OK} &= [\lambda_1, \lambda_2, \dots, \lambda_n, \mu]^T \\ \mathbf{W} &= \begin{bmatrix} \text{cov}(\mathbf{s}_1, \mathbf{s}_1) & \cdots & \text{cov}(\mathbf{s}_n, \mathbf{s}_1) & 1 \\ \vdots & \ddots & \vdots & 1 \\ \text{cov}(\mathbf{s}_1, \mathbf{s}_n) & \cdots & \text{cov}(\mathbf{s}_n, \mathbf{s}_n) & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \gamma(\mathbf{s}_1, \mathbf{s}_1) & \cdots & \gamma(\mathbf{s}_n, \mathbf{s}_1) & 1 \\ \vdots & \ddots & \vdots & 1 \\ \gamma(\mathbf{s}_1, \mathbf{s}_n) & \cdots & \gamma(\mathbf{s}_n, \mathbf{s}_n) & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \\ \mathbf{B}_{OK} &= [\text{cov}(\mathbf{s}, \mathbf{s}_1), \dots, \text{cov}(\mathbf{s}, \mathbf{s}_n), 1]^T \quad \text{or} \quad [\gamma(\mathbf{s}, \mathbf{s}_1), \dots, \gamma(\mathbf{s}, \mathbf{s}_n), 1]^T \end{aligned} \quad (25)$$

Thus, $\mathbf{\Lambda}_{OK} = \mathbf{W}^{-1}\mathbf{B}_{OK}$. Then, defining $\mathbf{Y}_{OK} = [z(\mathbf{s}_1), \dots, z(\mathbf{s}_n), 0]^T$, Eq. (23) becomes

$$\hat{z}_{OK}(\mathbf{s}) = \mathbf{Y}_{OK}^T \mathbf{\Lambda}_{OK} = \mathbf{Y}_{OK}^T \mathbf{W}^{-1} \mathbf{B}_{OK}. \quad (26)$$

The variance estimate is

$$\sigma_{OK}^2(\mathbf{s}_0) = \text{var}(z(\mathbf{s}_0)) - \mathbf{B}_{OK}^T \mathbf{W}^{-1} \mathbf{B}_{OK} \quad (27)$$

when the covariance is used or

$$\sigma_{OK}^2(\mathbf{s}_0) = \mathbf{B}_{OK}^T \mathbf{W}^{-1} \mathbf{B}_{OK} \quad (28)$$

when the semivariance is used.

2.3.5 Universal Kriging (UK)*

Going one step further in generality from OK is UK. In UK, $E\{z(\mathbf{s})\} = m(\mathbf{s})$ is no longer a constant mean, but now “drifts”; however, $\text{var}\{z(\mathbf{s})\}$ is still assumed constant. With these properties, UK methodology builds upon OK in the following manner. First, note that $z(\mathbf{s})$ can be considered to be composed of two components: the *drift* component, $m(\mathbf{s})$, that specifies the slowly varying expectation value and a *residual* component, $q(\mathbf{s}) = z(\mathbf{s}) - m(\mathbf{s})$, that gives the difference between the observations and the trend. Since observations and the drift should follow the same trend, the residuals should have a constant mean and, given a covariance matrix or semivariance for the residuals, can be found from OK. Next, the drift is modeled as a linear or quadratic polynomial so that (after combining the mean of the residuals with α_0)

$$m(\mathbf{s}_0) = \begin{cases} \alpha_0 + \sum_{i=1}^n (\alpha_1 x_i + \alpha_2 y_i) \\ \alpha_0 + \sum_{i=1}^n (\alpha_1 x_i + \alpha_2 y_i + \alpha_3 x_i^2 + \alpha_4 y_i^2 + \alpha_5 x_i y_i) \end{cases} \quad (29)$$

Then, under the assumption of Eq. (29), the system of equations $z(\mathbf{s}) = q(\mathbf{s}) + m(\mathbf{s})$ is the OK equations for the residuals expanded with extra Lagrange multipliers to account for the coefficients of the drift model. Note that since the residuals are the terms in the OK equations, the covariance and semivariance for the residuals are now $\text{cov}_q(\mathbf{s}_i, \mathbf{s}_j)$ and $\gamma_q(\mathbf{s}_i, \mathbf{s}_j)$.

Thus, for linear drift, the set of equations to solve is (again, $\text{cov}_q(\mathbf{s}_i, \mathbf{s}_j)$ and $\gamma_q(\mathbf{s}_i, \mathbf{s}_j)$ are interchangeable, but we will only print the semivariance):

$$\begin{aligned} \sum_{i=1}^n \lambda_i \gamma_q(\mathbf{s}_i, \mathbf{s}_1) + \alpha_0 + \alpha_1 x_1 + \alpha_2 y_1 &= \gamma_q(\mathbf{s}, \mathbf{s}_1) \\ \sum_{i=1}^n \lambda_i \gamma_q(\mathbf{s}_i, \mathbf{s}_2) + \alpha_0 + \alpha_1 x_2 + \alpha_2 y_2 &= \gamma_q(\mathbf{s}, \mathbf{s}_2) \\ &\vdots \\ \sum_{i=1}^n \lambda_i \gamma_q(\mathbf{s}_i, \mathbf{s}_n) + \alpha_0 + \alpha_1 x_n + \alpha_2 y_n &= \gamma_q(\mathbf{s}, \mathbf{s}_n) \\ \sum_{i=1}^n \lambda_i &= 1; \quad \sum_{i=1}^n \lambda_i x_i = x; \quad \sum_{i=1}^n \lambda_i y_i = y \end{aligned} \quad (30)$$

(The quadratic drift follows the same pattern with the required extra terms.) In matrix form, Eq. (30) becomes $\mathbf{W}\boldsymbol{\Lambda}_{UK} = \mathbf{B}_{UK}$, where

* Although we do not use UK in Section 3, we describe it here for the sake of completeness.

$$\begin{aligned}
\mathbf{\Lambda}_{UK} &= [\lambda_1, \lambda_2, \dots, \lambda_n, \mu_0, \mu_1, \mu_2]^T \\
\mathbf{W} &= \begin{bmatrix} \gamma_q(\mathbf{s}_1, \mathbf{s}_1) & \cdots & \gamma_q(\mathbf{s}_n, \mathbf{s}_1) & 1 & x_1 & y_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \gamma_q(\mathbf{s}_n, \mathbf{s}_1) & \cdots & \gamma_q(\mathbf{s}_n, \mathbf{s}_n) & 1 & x_n & y_n \\ 1 & \cdots & 1 & 0 & 0 & 0 \\ x_1 & \cdots & x_n & 0 & 0 & 0 \\ y_1 & \cdots & y_n & 0 & 0 & 0 \end{bmatrix} \\
\mathbf{B}_{UK} &= [\gamma_q(\mathbf{s}, \mathbf{s}_1), \dots, \gamma_q(\mathbf{s}, \mathbf{s}_n), 1, x, y]^T
\end{aligned} \tag{31}$$

Thus, $\mathbf{\Lambda}_{UK} = \mathbf{W}^{-1}\mathbf{B}_{UK}$. Defining matrix $\mathbf{Y}_{UK} = [z(\mathbf{s}_1), \dots, z(\mathbf{s}_n), 0, 0, 0]^T$, the UK solution for

$$\hat{z}_{UK}(\mathbf{s}) = \sum_{i=1}^n \lambda_i z(\mathbf{s}_i) \tag{32}$$

is

$$\hat{z}_{UK}(\mathbf{s}) = \mathbf{Y}_{UK}^T \mathbf{\Lambda}_{UK} = \mathbf{Y}_{UK}^T \mathbf{W}^{-1} \mathbf{B}_{UK}. \tag{33}$$

The associated variances have the same form as Eq. (27) if covariances are used or Eq. (28) if semivariances are used.

2.3.6 Available Software

Deutsch and Journal [37] developed the open-source geostatistical software suite, GSLib [38], for calculating variograms, covariances and interpolated surfaces by a multitude of kriging algorithms, including SK, OK, and UK. The software is available in both Fortran 77 and 90. Although the user may provide semivariances, GSLib converts semivariances to covariances before solving for the kriging equations ($\gamma(0)$ poses numerical problems).

2.3.7 Advantages and Disadvantages

As Davis discusses (Ref. [9], p. 418), some strengths of kriging are as follows: 1) the technique provides exact interpolation at the data points, 2) error may be estimated for every grid point on the interpolated surface, and 3) the error estimates are the lowest of all linear estimation methods. Pitfalls, as discussed by O'Sullivan and Unwin ([31], pp. 280-1) include the following points: 1) the fit of the interpolation to the data depends on the validity of the variogram, which is often modeled; 2) kriging is computationally intensive, especially the part where the inverse of the \mathbf{C} or \mathbf{W} matrix is required, and may be subject to rounding errors that may or may not be significant.

2.4 Bayesian Approaches

The approaches discussed above appear to be the main techniques for interpolation, fusion and error quantification which have been devised, applied to and validated for bathymetry data. Some newer approaches to data interpolation involve the use of Bayesian inference and Monte Carlo calculations to obtain estimates and uncertainty of geospatial data on a rectangular grid. Implementation of these methods, however, are judged to be immature for operational use as they do not appear to have been

validated for bathymetry and/or require research efforts for adaptation to bathymetry fusion, especially if the fused product is to be used for a hydrographic chart. They are mentioned to provide possible future research directions.

De Oliveira [39] and Hartman [40] present a Bayesian inference approaches to obtain conditional probabilities density functions (cdf) for geospatial variables at regular grid points given measurements and the associated errors at random locations. The interpolated values are then the mean or median of the conditional cdf and the 95% confidence interval provides the uncertainty. Both approaches model the background from which the data is sampled as a stochastic field. (De Oliveira models to background as a Gaussian random field, while Hartman uses a Gaussian Markov random field.) Bayesian inference is then used to estimate parameters that specify these fields, given the information contained in the observations. The Bayesian inference however, requires integration over a highly dimensionalized space, so the integration is approximated from Markov chain Monte Carlo calculations.

Goff et al. [41] use *a posteriori* resampling to remove noise from geospatial data, including the same bathymetry set discussed in Calder [8]. This technique may be adaptable to regridding data at new grid points. Developmental work is needed first to adapt the algorithm to providing this capability. Also, the authors admit that this approach is computationally expensive to implement because the data surface that is generated is dependent on the order that the data is analyzed, so the data needs to be reanalyzed multiple times and an average taken.

3. RECOMMENDED APPROACH

It is recommended that the new fusion algorithm be based on the procedure given by Calder [8], which uses a three-step approach to fuse bathymetry data sets into one surface and provide an uncertainty layer. For hydrography, the methodology also discusses how to apply the uncertainty to assess the order of survey as defined by the International Hydrographic Organization [42]. Figure 5 illustrates the adaptation of Calder's method for the new algorithms.

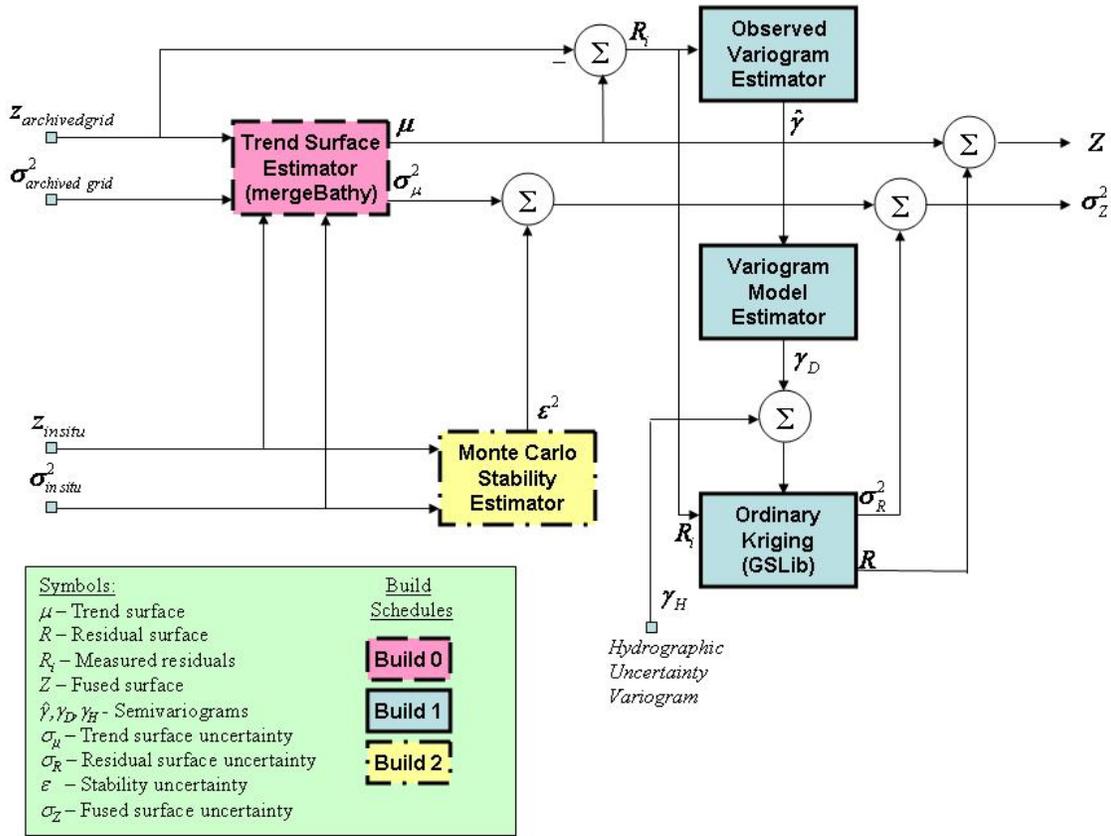


Fig. 5 – New bathymetry fusion algorithm. Data flow is left to right with z 's and σ 's referring to depths and uncertainties for either archived grids (top) or new in situ data (bottom). Other symbols (see text for meaning) and component build schedules are in the key. Adapted from Calder [8].

The steps of the methodology are as follows.

1. Following Plant et al. [24], interpolate the data with the quadratic loess interpolation technique of Cleveland [20] to provide a trend surface for the bathymetry and uncertainty layer.
2. Restore finer details smoothed by the interpolation from ordinary kriging of the residuals; add the errors associated with kriging to the uncertainty layer from Step 1 assuming statistical independence (i.e., the variances add). The surface generated from kriging the residuals is the residual surface.
3. Estimate additional uncertainty caused by positional errors from the Monte Carlo technique of Jakobsson et al. [18], but repeat steps 1 and 2 above instead of using the splines-in-tension algorithm at each iteration. As before, add the estimated error to the uncertainty layer by assuming statistical independence.

In equation form, the final bathymetry surface, $Z(\mathbf{s})$, is the sum of the trend surface, $\mu(\mathbf{s})$, and the residual surface, $R(\mathbf{s})$.

$$Z(\mathbf{s}) = \mu(\mathbf{s}) + R(\mathbf{s}) \quad (34)$$

The uncertainty layer, $\sigma_z(\mathbf{s})$, is

$$\sigma_z(\mathbf{s}) = [\sigma_\mu^2(\mathbf{s}) + \sigma_R^2(\mathbf{s}) + \varepsilon^2(\mathbf{s})]^{1/2}, \quad (35)$$

where $\sigma_\mu(\mathbf{s})$ and $\sigma_R(\mathbf{s})$ are the uncertainties for the trend and residual layers, respectively, and $\varepsilon(\mathbf{s})$ is the uncertainty layer associated with positional errors.

This approach appears to be the most comprehensive, validated methodology for interpolating bathymetry and providing uncertainty estimates. Specifically, it 1) combines accepted techniques to attain a fine detailed bathymetry surface with uncertainty estimation. 2) was demonstrated and validated in [8] using two sets of data for the New Jersey Atlantic Shelf – vertical beam echo-sounder data from the 1970s and multibeam echo-sounder data from the 1990s. In addition, NRL already has code for loess interpolation of bathymetry data based on the work of Plant et al. [24], which has now been augmented with other weighting windows to provide other interpolation options and possibly faster computation. In addition, NRL's code also can correct data sets that have different vertical offsets to alleviate interpolation over false cliffs.

3.1 Quadratic Loess Interpolation for Trend Surface

Extending Eqs. (5) through (8) to two dimensions, the equations for calculating the trend surface, $\mu(\mathbf{s})$ in Eq. (34) are as follows:

$$\mu(\mathbf{s}) = \mathbf{p}^T(\mathbf{s})\hat{\boldsymbol{\beta}}(\mathbf{s}), \quad (36)$$

where $\mathbf{p}^T(\mathbf{s}) \equiv [x^2, y^2, xy, x, y, 1]$ and vector $\hat{\boldsymbol{\beta}}(\mathbf{s}) \equiv [\hat{\beta}_5(\mathbf{s}), \dots, \hat{\beta}_0(\mathbf{s})]^T$ being the set of $\beta_n(\mathbf{s})$'s that minimize the weighted least squares

$$q(\mathbf{s}) = \sum_{k=1}^n w_k(\mathbf{s}) (z_k - \beta_0(\mathbf{s}) - \beta_1(\mathbf{s})y_k - \beta_2(\mathbf{s})x_k - \beta_3(\mathbf{s})x_k y_k - \beta_4(\mathbf{s})y_k^2 - \beta_5(\mathbf{s})x_k^2)^2 \quad (37)$$

using the two-dimensional tri-cube weighting function

$$w_k(\mathbf{s}) = \begin{cases} \left(1 - \left|\frac{\mathbf{s} - \mathbf{s}_k}{d_0}\right|^3\right)^3, & \left|\frac{\mathbf{s} - \mathbf{s}_k}{d_0}\right| < 1 \\ 0, & \text{otherwise} \end{cases} \quad (38)$$

and the user provided value for d_0 . Calder's paper suggests that d_0 be ten times the largest sample spacing. Information lost by oversmoothing is regained when the residuals are calculated in the kriging portion.

Estimates for the variances follow Eq. (8) for like variances in all the data or $\hat{\sigma}^2(\mathbf{s}) = \sum_{k=1}^n [\alpha_k(\mathbf{s})\sigma_k]^2$ for differing variances. The $\alpha_k(\mathbf{s})$'s now have azimuthal symmetry and the same radial dependence as the 1-D case.

In addition, the estimate is made more robust (i.e., eliminates outliers) by flagging estimates greater than three Mahalanobis units as “no data.” The Mahalanobis distance [43], $\mathbf{M}(\boldsymbol{\mu}, \mathbf{m}, \mathbf{C})$, is

$$\mathbf{M}(\boldsymbol{\mu}, \mathbf{m}, \mathbf{C}) \equiv (\boldsymbol{\mu} - \mathbf{m})^T \mathbf{C}^{-1} (\boldsymbol{\mu} - \mathbf{m}) \quad (39)$$

where $\boldsymbol{\mu}(\mathbf{s})$ is the vector of all estimated depths and $\mathbf{m}(\mathbf{s})$ and $\mathbf{C}(\mathbf{s})$ are the corresponding mean depth measurements and covariances for the windowed (c.f. Eq. (38)) data sets. To intuitively explain Eq. (39), suppose that the data points are all independent and have mean m and variance σ^2 . The \mathbf{C} matrix is all zeros except for the diagonal elements, which are all σ^2 . Then, if an estimate differs from its corresponding windowed mean by more than three variances (i.e., $(\mu - m)^2 / \sigma^2 > 3$), that estimate receives the “no data” mark. Equation (39) generalizes this simpler scenario to account for covariance between the data.

3.2 Kriging the Residuals

Since loess interpolation is not exact at the data points, residuals, $R(\mathbf{s}_i)$, exists between actual measurements, $z(\mathbf{s}_i)$, and the corresponding estimated depths along trend surface, $\mu(\mathbf{s}_i)$, such that $R(\mathbf{s}_i) = z(\mathbf{s}_i) - \mu(\mathbf{s}_i)$. The residual surface in Eq. (34) is found from interpolation of the $R(\mathbf{s}_i)$ set using ordinary kriging, as an overall but constant unknown bias may exist in the residuals. The variogram will likely have directional anisotropy, so that $2\gamma(\mathbf{s}_i, \mathbf{s}_j) = 2\gamma(h, \theta)$, where h is the separation distance between the two points and θ is the heading angle (clockwise from the north).

To account for this anisotropy, the following semivariance, $\gamma_D(\mathbf{s}_i, \mathbf{s}_j)$, is used and constructed (unconditionally valid for two-dimensions [44]) in the following manner: 1) obtain an empirical estimate of the variogram, 2) compute the average azimuthal variogram from the empirical estimate to detect the directions of minimum and maximum variation, 3) fit the variograms along these two axes to the spherical model for variograms, and 4) using parameters from the fits to the spherical model and the direction for minimum variance, construct $\gamma_D(\mathbf{s}_i, \mathbf{s}_j)$ from Eqs. (45) – (48) below. These steps are now discussed in more detail.

Step 1: Following Cressie ([33], p.69), the variogram for the residuals is approximated using the methods-of-moments (or classical) estimator, in blocks sizes = $2d_0$ (c.f. Eq. (38)), so that

$$2\gamma(h, \theta) \approx 2\hat{\gamma}(h_i, \theta_i) = \frac{1}{|N(h_i)||N(\theta_j)|} \sum_{(a,b) \in N(h_i) \cap N(\theta_j)} (R(\mathbf{s}_a) - R(\mathbf{s}_b))^2. \quad (40)$$

(Note that Eq. (40) is a more general form of Eq. (16).) The sets $N(h_i)$ and $N(\theta_j)$ contain binned separation distances and heading angles as defined by the equations

$$N(h_i) \equiv \left\{ (a, b) : h_i - \frac{\Delta h_i}{2} \leq d(\mathbf{s}_a, \mathbf{s}_b) \leq h_i + \frac{\Delta h_i}{2} \right\} \quad (41)$$

$$N(\theta_j) \equiv \left\{ (a, b) : \theta_j - \frac{\Delta\theta}{2} \leq \angle(\mathbf{s}_a, \mathbf{s}_b) \leq \theta_j + \frac{\Delta\theta}{2} \right\}, \quad (42)$$

where $d(\mathbf{s}_a, \mathbf{s}_b)$ and $\angle(\mathbf{s}_a, \mathbf{s}_b)$ are the Euclidian distance and heading angle between \mathbf{s}_a and \mathbf{s}_b and $|N(h_i)|$ and $|N(\theta_j)|$ are the number of bins in these sets. For the data set analyzed in [8], $\Delta h_i = 250\text{m}$, $\Delta\theta_i = 45^\circ$, $h_i = i\Delta h + \Delta h / 2$ and $\theta_j = j\Delta\theta$.

Step 2: The average azimuthal variogram, $2\bar{\gamma}(\theta_j) \equiv |N(h_i)|^{-1} \sum_i 2\hat{\gamma}(h_i, \theta_j)$ is then calculated. In [8], $2\bar{\gamma}(\theta_j)$ had two sets of maximum and minima (caused by trending ridges), so it could be modeled by the function

$$2\bar{\gamma}(\theta_j) \approx 0.5g_0 + g_2 \cos(4\pi\theta_j + \phi_2), \quad (43)$$

which is the second Fourier eigenfunction, Equation (43) is fitted to the data by evaluating the second discrete Fourier transform coefficient. The phase constant, ϕ_2 , is the radian angle (which goes counterclockwise from the east) where the first minimum is found. It is changed to a heading angle (again, clockwise from the north), θ_m , from the transformation $\theta_m \equiv -\phi_2/2 + \pi/2$. This angle is obtained for each $2d_0$ block, then loess interpolated to get $\theta_m(\mathbf{s}_i)$ for Step 4 below.

Step 3: Two finer-scaled directional variograms were then calculated; one in the θ_m direction, the other in the perpendicular direction, θ_m^\perp ; using the methods-of-moments estimator with angle bins of $\pm\pi/2$ about each direction. In this case, $\Delta h = 100\text{m}$ and the data were truncated to those within 95% of the mean to reduce outliers. These empirical variograms are fitted to the standard spherical variogram model (Cressie [33], Eq. 2.3.8; Davis [9], Eq. 4.98)

$$2\gamma(h) = \begin{cases} 0, & h = 0 \\ a_0 + a_1 \left(1.5(h/a_2) - 0.5(h/a_2)^3 \right), & 0 \leq h/a_2 \leq 1 \\ a_0 + a_1, & h/a_2 \geq 1 \end{cases} \quad (44)$$

using the Levenberg-Marquardt algorithm [45, 46] for fitting (i.e., solving for a_0 , a_1 , and a_2). Let the modeled variogram along the θ_m direction be $2\gamma_0(h)$ and the fitting constants be a_0 , a_1 , and a_2 . Similarly, let the modeled variogram along the θ_m^\perp direction be $2\gamma_0^\perp(h)$, and the fitting constants be a_0^\perp , a_1^\perp , and a_2^\perp . When solving for both sets of constants, the variograms are constrained to equal the same sill at large distances so that $a_0^\perp = a_0$ and $a_1^\perp = a_1$, but $a_2^\perp \neq a_2$ in general.

Step 4: Define the anisotropy parameter, $\alpha_{aniso} \equiv a_2/a_2^\perp$. Then, $\gamma_D(\mathbf{s}_i, \mathbf{s}_j)$, is evaluated in the following manner.

$$\gamma_D(\mathbf{s}_i, \mathbf{s}_j) = \gamma_0(d'(\mathbf{s}_i, \mathbf{s}_j)), \quad (45)$$

where

$$d'(\mathbf{s}_i, \mathbf{s}_j) \equiv \|\mathbf{A}(\theta_m(\mathbf{s}_i), \alpha_{aniso})(\mathbf{s}_i - \mathbf{s}_j)\|, \quad (46)$$

$$\mathbf{A}(\theta_m(\mathbf{s}_i), \alpha_{aniso}) = \mathbf{R}(-\theta_m(\mathbf{s}_i)) \text{diag}(1, \alpha_{aniso}) \mathbf{R}(\theta_m(\mathbf{s}_i)), \quad (47)$$

and $\mathbf{R}(\theta_m(\mathbf{s}_i))$ is the standard two-dimensional rotational matrix [47]

$$\mathbf{R}(\theta_m(\mathbf{s}_i)) = \begin{bmatrix} \cos(\theta_m(\mathbf{s}_i)) & \sin(\theta_m(\mathbf{s}_i)) \\ -\sin(\theta_m(\mathbf{s}_i)) & \cos(\theta_m(\mathbf{s}_i)) \end{bmatrix}. \quad (48)$$

For applications where only the bathymetry is needed, $\gamma_D(\mathbf{s}_i, \mathbf{s}_j)$, may be sufficient for use. In hydrographic situations, however, where extra caution is required for navigation safety, an additional “hydrographic uncertainty” variogram is added to increase the total uncertainty for safety (see Calder [8] for details). This variogram is defined to be

$$2\gamma_H(h) \equiv b_0 + b_1 h. \quad (49)$$

The choice of the coefficients is arbitrary, but Calder uses $b_0 \equiv 0$ and $b_1 \equiv 6.51 \times 10^{-4}$ m so that the 95% confidence interval of the uncertainty increases by 0.05 m for every meter of horizontal separation in soundings. Thus, for hydrography, the final variogram to use is

$$2\gamma_{total}(\mathbf{s}_i, \mathbf{s}_j) = 2\gamma_D(\mathbf{s}_i, \mathbf{s}_j) + 2\gamma_H(\|\mathbf{s}_j - \mathbf{s}_i\|) \quad (50)$$

3.3 Monte Carlo Estimation of Depth Error Due to Positional Stability Errors

This last part is of use for fusion with soundings data, either with other soundings data or a historical grid, as the position of the soundings contains errors (this part is not applied when fusing historical gridded data sets). As before (c.f. Section 2.1), positioning errors result in errors of depth estimates on the interpolation surface as the position of the soundings affect the interpolated solution; the Monte Carlo technique of Jakobsson et al. [18] is used to estimate this error. At the beginning of each iteration, the locations of the soundings are perturbed according to a probability density function appropriate for the sounding, often assumed to be Gaussian unless otherwise known (c.f. Fig. 5 in Calder [8] for a non-Gaussian example). Loess interpolation and kriging, using the variogram from the unperturbed set, are repeated at each iteration. The standard deviation of the solutions at each interpolation point provides $\varepsilon(\mathbf{s})$ in Eq. (35).

4. ALGORITHM DEVELOPMENT, REQUIREMENTS, AND TEST CASES

NRL plans to develop a prototype build in FY07, followed by two builds in FY08 and FY09, and to deliver these builds to the Bathymetry Database Division of NAVOCEANO. In addition to illustrating data flow, Fig. 5 shows that required components for implementing the procedure discussed in Section 3 and color codes them by build schedule. The prototype build, Build 0, will consist of NRL's "mergeBathy" utility (translated to the C++ programming language) with its capability to loess interpolate *in situ* collected data to one interpolation surface with uncertainty. In addition, Build 0 will include support for exporting grid data to the Open Navigation Surface BAG file format. Build 1 is focused on fusion of gridded historic data. In Build 1, Build 0 will be coupled with variogram and kriging routines to provide the historic bathymetry fusion capability. Build 2 adds the capability to estimate depth errors induced by positional errors from *in situ* data.

As the fusion algorithm develops, NRL can perform validation using the following data.

1. DBDB-V grids of varying resolutions used in the OAML Feathering Algorithm report [3]. Uncertainties can be provided from soundings collected in these areas via CUBE.
2. Data collected as a part of the FY04-FY06 AN/AQS-20A RTP effort [48]. These data were collected off Panama City, FL. They include five sets of multibeam data from AN/AQS-20A flights, high-density multibeam data from dedicated NAVOCEANO bathymetry surveys in 2002 and 2006, historic NGDC single beam soundings, and NOAA gridded data that is now in DBDB-V. Uncertainties have been calculated for these data. Estimates of uncertainty for the historic data will be needed.
3. Scripps Canyon data provided by Plant for validation of his interpolation technique [24]. These data have uncertainty estimates.
4. A NAVOCEANO data set or collection of data sets where we would test the algorithms using all of the lines and then every other line to see how the results differ, particular after applying the interpolation and then kriging as described by Calder's 2006 paper. Error estimates can be provided via CUBE processing.

5. SUMMARY

This report reviews the current state-of-the-art for fusion algorithms applicable to bathymetry data and motivated by NAVOCEANO's need for intelligent fusion algorithms. Currently, NAVOCEANO's DBDB-V incorporates the OAML Feathering Algorithm to smooth discontinuities that occur between tiles in the database with dissimilar resolutions. This solution still leaves artifacts and does not provide estimates of uncertainty. NAVOCEANO needs a better interpolation technique that also provides uncertainty estimates for robust interpolation and to use one database for multiple purposes, including navigation.

State-of-the-art interpolation techniques include splines-in-tension, loess interpolation, and kriging. Splines-in-tension solves a differential equation numerically to obtain an interpolated surface. This technique is the basis of the OAML Feathering Algorithm. It does not provide error estimation readily, although a Monte Carlo technique can be used to provide estimates. Loess interpolation is a localized weighted regression technique to fit data to a low-order polynomial. It provides quantified uncertainty estimates. The technique is not exact and acts as a low-pass spatial filter, so bias in the interpolated surface occurs and finer details are smoothed. Kriging is a methodology for providing interpolation by solving a system of simultaneous equations. It also provides quantified uncertainty estimates and is exact at the observation points. The technique depends on the model or the choice of variogram to be reliable and the matrix inversion is computationally intensive. This report also briefly reviews Bayesian approaches that are being developed in the academic community. These approaches may provide future

fusion methodologies, but need to be demonstrated with bathymetry data first at the research and development level.

The recommended approach for intelligent data fusion was published and validated for bathymetry data recently by Calder [8]. His approach uses both loess interpolation to obtain a trend surface, followed by kriging of residuals to recapture finer details lost by smoothing. In addition, if *in situ* soundings are used, Monte Carlo simulations are run to estimate depth error induced by position errors. The technique provides the means to also liberally estimate errors for navigation safety through the use of a hydrographic comfort variogram.

ACKNOWLEDGMENTS

Funding for this project was provided by US Navy PEO C4I, PMW-120, Program Element 0603207N, with Capt. Robert Parker as program manager. We are grateful to Capt. Parker and Dr. Mozley for their support. We also thank Dr. Brian Calder of the Center for Coastal and Ocean Mapping at the University of New Hampshire, Adeline Wong of the NOAA National Ocean Service, and our NAVOCEANO partners R. Wade Ladner and Bill Rankin for their helpful discussions.

REFERENCES

1. W.H.F. Smith and D.T. Sandwell, "Global Sea Floor Topography from Satellite Altimetry and Ship Depth Soundings," *Science* **277**, 1956-1962, 1997.
2. D.L. Divins and D. Metzger, *NGDC Coastal Relief Model*, retrieved April 2007, <http://www.ngdc.noaa.gov/mgg/coastal/coastal.html>.
3. C.A. Steed and W.E. Rankin, "OAML Feathering Algorithm Overview," NRL/FR/7440--03-10,052, Naval Research Laboratory, Washington, DC, May 16, 2003.
4. D.W. Caress and D.N. Chaves, *MB-System*, Version 5.1.0, Monterey Bay Aquarium Research Institute and the Lamont-Doherty Earth Observatory of Columbia University, Monterey, CA and New York, 2006.
5. *IHO Standards for Hydrographic Surveys*, 5th edition (draft), Special Publication #44: International Hydrographic Bureau, 2007.
6. B.R. Calder and L.A. Mayer, "Automatic Processing of High-rate, High-density Multibeam Echosounder Data," *Geochemistry, Geophysics, Geosystems* **4**, Art. No. 1048, 2003.
7. B.R. Calder, S. Byrne, B. Lamey, R.T. Brennan, J.D. Case, D. Farbe, B. Gallagher, R.W. Ladner, F. Moggert, M. Paton, and The Open Navigation Surface Working Group, "The Open Navigation Surface Project," *International Hydrographic Review* **6**, 1-10, 2005.
8. B. Calder, "On the Uncertainty of Archive Hydrographic Data Sets," *IEEE J. Oceanic Eng.* **31**, 249-265, 2006.
9. J.C. Davis, *Statistics and Data Analysis in Geology*, third edition (Wiley, New York, 2002).
10. W.H.F. Smith and P. Wessel, "Gridding with Continuous Curvature Splines in Tension," *Geophysics* **55**, 293-305, 1990.

11. I.K. Crain, "Computer Interpolation and Contouring of 2-dimensional Data - A Review," *Geoexploration* **8**, 71, 1970.
12. R. Franke, "Scattered Data Interpolation - Tests of Some Methods," *Mathematics of Computation* **38**, 181-200, 1982.
13. I.C. Briggs, "Machine Contouring Using Minimum Curvature," *Geophysics* **39**, 39-48, 1974.
14. A.E.H. Love, *A Treatise on the Mathematical Theory of Elasticity*. 4th ed. (Dover, New York, 1927).
15. C.J. Swain, "A FORTRAN IV Program for Interpolating Irregularly Spaced Data Using the Difference Equations for Minimum Curvature," *Computers and Geosciences* **1**, 231-240, 1976.
16. R.J. VanWyckhouse, "Synthetic Bathymetric Profiling System (SYNBAPS)," TR-233, Naval Oceanographic Office, Stennis Space Center, MS, 1973.
17. P. Wessel and W.H.F. Smith, *The Generic Mapping Tools*, version 4.2.0, retrieved 2007. University of Hawaii, <http://gmt.soest.hawaii.edu/>.
18. M. Jakobsson, B. Calder, and L. Mayer, "On the Effect of Random Errors in Gridded Bathymetric Compilations," *J. Geophys. Res. - Solid Earth* **107**, article 2358, 2002.
19. Editorial Board: IBCAO, 2002. International Bathymetric Chart of the Arctic Ocean (IBCAO). NOAA - National Geophysical Data Center, Boulder, CO. <http://www.ngdc.noaa.gov/mgg/bathymetry/arctic/arctic.html>
20. W.S. Cleveland, "Robust Locally Weighted Regression and Smoothing Scatterplots," *J. American Statistical Association* **74**, 829-836, 1979.
21. W.S. Cleveland and S.J. Devlin, "Locally Weighted Regression - An Approach to Regression-analysis by Local Fitting," *J. American Statistical Association* **83**, 596-610, 1988.
22. G.H. Givens and J.A. Hoeting, *Computational Statistics* (Wiley-Interscience, Hoboken, NJ, 2005).
23. NIST/SEMATECH e-Handbook of Statistical Methods, National Institute of Standards and Technology, retrieved May 2007, <http://www.itl.nist.gov/div898/handbook/pmd/section1/pmd144.htm>.
24. N.G. Plant, K.T. Holland, and J.A. Puleo, "Analysis of the Scale of Errors in Nearshore Bathymetric Data," *Marine Geology* **191**, 71-86, 2002.
25. B.W. Silverman, "Spline Smoothing - The Equivalent Variable Kernel-Method," *Annals of Statistics* **12**, 898-916, 1984.
26. W.S. Cleveland, E. Grosse, and M-J Shyu, *LOESS* (Bell Labs: Murray Hill, NJ, 1992) <http://netlib.bell-labs.com/netlib/a/dloess.gz>.
27. C. Loader, *LOCFIT*, version 2.01, (University of Auckland, New Zealand, 2006) <http://www.locfit.info/>.
28. *MATLAB*, version 7.4 (The Mathworks, Inc., Natick, MA, 2007).

29. M.G. Schlax and D.B. Chelton, "Frequency-domain Diagnostics for Linear Smoothers," *J. American Statistical Association* **87**, 1070-1081, 1992.
30. A. Buja, T. Hastie, and R. Tibshirani, "Linear Smoothers and Additive-models," *Annals of Statistics* **17**, 453-510, 1989.
31. D. O'Sullivan and D.J. Unwin, *Geographic Information Analysis* (Wiley, Hoboken, New Jersey, 2003).
32. A.G. Journel, *Fundamentals of Geostatistics in Five Lessons*, Volume 8 (American Geophysical Union, Washington, 1989).
33. N.A.C. Cressie, *Statistics for Spatial Data*, revised ed. (Wiley, New York, 1993).
34. J. D.P. Chiles, *Geostatistics : Modeling Spatial Uncertainty* (Wiley, New York, 1999).
35. D.G. Krige, "A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand," *J. Chemical, Metallurgical, and Mining Society of South Africa* **52**, 119-139, 1951.
36. G. Matheron, "Principles of Geostatistics," *Economic Geology and the Bulletin of the Society of Economic Geologists* **58**, 1246-1266, 1963.
37. C.V. Deutsch and A.G. Journel, *GSLIB : Geostatistical Software Library and User's Guide*. Version 2.0, 2nd ed. (Oxford, New York, 1998).
38. C.V. Deutsch and A.G. Journel, *GSLIB: Geostatistical Software Library*, Version 2.907, 2000, <http://www.gslib.com/>.
39. V. De Oliveira, "Bayesian Inference and Prediction of Gaussian Random Fields Based on Censored Data," *J. Computational and Graphical Statistics* **14**, 95-115, 2005.
40. L.W. Hartman, "Bayesian Modelling of Spatial Data Using Markov Random Fields, with Application to Elemental Composition of Forest Soil," *Mathematical Geology* **38**, 113-133, 2006.
41. J.A. Goff, C. Jenkins, and B. Calder, "Maximum A Posteriori Resampling of Noisy, Spatially Correlated Data," *Geochemistry Geophysics Geosystems* **7**, 10.1029/2006GC001297, 2006.
42. International Hydrographic Organization, *IHO Standards for Hydrographic Surveys*, 4th edition, Special Publication #44 (International Hydrographic Bureau, Monaco, 1998).
43. P.C. Mahalanobis, "On the Generalised Distance in Statistics," *Proceedings of the National Institute of Science of India* **12**, 49-55, 1936.
44. G. Christakos, "On the Problem of Permissible Covariance and Variogram Models," *Water Resources Research* **20**, 251-265, 1984.
45. S. Brandt, *Data Analysis: Statistical and Computational Methods for Scientists and Engineers*, 3rd edition (Springer, New York, 1999).
46. D.W. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *J. Society for Industrial and Applied Mathematics* **11**, 431-441, 1963.

47. H. Goldstein, *Classical Mechanics* (Addison Wesley, Reading, MA, 1980).
48. P.A. Elmore, W.E. Avera, M.M. Harris, and K.M. Duvieilh, "Environmental Measurements Derived from Tactical Mine-hunting Sonar Data," *IEEE:OES Oceans 2007 - Europe Conference Proceedings*, 18-21 June 2007.