THE NAVIGATION SURFACE

A MULTIPURPOSE BATHYMETRIC DATABASE

BY

SHEPARD M. SMITH
BS, Cornell University, 1993

THESIS

Submitted to the University of New Hampshire
In Partial Fulfillment of
The Requirements for the Degree of

Master of Science
In
Ocean Engineering
Ocean Mapping Option

May, 2003

This thesis has been examined and approved.

Thesis Director, Dr. Larry Mayer
Director, Center for Coastal and Ocean Mapping
Professor, University of New Hampshire

Capt. Andrew A. Armstrong III, NOAA (ret)
Co-Director, Joint Hydrographic Center
Affiliate Professor, University of New Hampshire

Dr. Brian Calder, Research Assistant Professor, UNH

Dr. Lee Alexander, Research Associate Professor, UNH

Capt. Nicholas Perugini, NOAA
Chief, Marine Charts Division, Coast Survey

Date

FOREWORD

Every field of study has its own vocabulary. Ocean mapping generally and hydrography in particular, is no exception. While I have tried to avoid using unnecessary jargon, there are a few words that express ideas and context beyond their literal meaning, and have therefore been used.

In order not to distract the reader from the flow of the text by including explanations of words in the text, I have included a glossary as an appendix.

ACKNOWLEDGEMENTS

The author wishes to acknowledge the support of the Office of Coast Survey, National Ocean Service, National Oceanic and Atmospheric Administration. Coast Survey supported the assignment of the author to the Joint Hydrographic Center at University of New Hampshire as a research assistant during the time this research was conducted.

The author is a NOAA Corps officer, with salary and personnel support from the NOAA Corps.

The Center for Coastal and Ocean Mapping is supported under NOAA grant NA97OG0241.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

THE NAVIGATION SURFACE

A MULTIPURPOSE BATHYMETRIC DATABASE

By

Shepard M. Smith

University of New Hampshire, May, 2003

The Navigation Surface is a bathymetric database and the methods required to populate it, maintain and manipulate it, and create products from it, including navigation products. The approach has the potential to improve the hydrographic survey and nautical charting processes by streamlining manual tasks and automating cartography. In addition, by modeling, tracking, and reporting uncertainty as well as depth at each model node, it gives the hydrographer and the cartographer a rigorous methodology for managing the survey process and for making decisions about precedence among surveys in a common area.

This approach is demonstrated in the Piscataqua River, Great Bay estuary and approaches (Portsmouth, NH vicinity). The goal of the testbed project is to create a model of the seafloor that is suitable for use in navigation all the way to the shoreline. The existing surveys available in the area consist of a variety of survey techniques from leadline measurements to old reconnaissance-density singlebeam echo soundings to modern multibeam echo soundings.

To create a source model for inclusion in the database, a series of steps were performed. First, each survey was individually modeled to create a collection of models of depth and uncertainty. Next, rules were established for superseding one survey with another. The resulting collection of models represents the Navigation Surface database. To create a chart from the database, the individual models were combined into a single model that covers the geographic area of interest. Next, the model was downsampled and generalized to the scale of the planned product. Finally, the model was contoured and selected depths were taken from it to populate an Electronic Navigational Chart (ENC).

The Navigation Surface database has considerable potential to improve both the speed and the objectivity of the charting process as well as providing a means for hydrographic offices to be interoperable with other users of bathymetric data. Both the high accuracy bathymetry necessary for safe navigation and the high resolution internally consistent bathymetry necessary for marine geology, habitat characterization, and marine modeling are preserved. In addition, the uncertainty estimate inherent in the model provides quality control for the survey, and allows the hydrographer to prioritize further work.

A database of source models could become the source from which the bathymetric portions of nautical charting products are drawn. The cartographic processes required for production of today's paper, raster, and vector charts could be streamlined by generalizing the model to support the navigation purpose of the chart before cartographic features are derived from it. Equally important, this approach preserves the highest resolution data needed for other applications.

INTRODUCTION

> …electronic positioning equipment and depth sounding instruments have been used in semi-automated and automated systems…They have increased the accuracy of the data and the completeness of bottom coverage…A typical survey of this type contains between 2000 and 20,000 data points. These systems increased the data acquisition rate to such an extent that manual data processing methods could not keep up with data acquisition… --LCDR Alan J. Pickrell, NOAA (1979)

The Problem

There are three big problems facing modern hydrographers. First, traditional data validation procedures and tools are overwhelmed by the volumes of soundings associated with modern multibeam sonar systems. Second, the cartographic processes used to create maps and nautical charts from hydrographic data are manually intensive and subjective, and are geared specifically to the needs of today's navigation products, e.g., the paper nautical chart and the Electronic Navigational Chart (ENC). Third, the products created by hydrographers are often incompatible with the needs of other users of marine bathymetric information and vice-versa.

Overall Goals

The first goal of the process described in this thesis is to provide a common tool for two separate disciplines: 1) hydrographic surveying for nautical charting and 2) ocean mapping for marine geology and habitat mapping. While similar sonar technology is typically used by both disciplines, and similar techniques are used to resolve depths from the sonar, the treatment of the measurements after collection and validation has been fundamentally different. The process used to build a chart involves successive downsampling of the data, preserving shoal measurements for use in a smooth sheet and then a nautical chart. Many marine geologists, on the other hand, have begun to use digital terrain models to view and analyze the seafloor bathymetry. The product used by each group often does not meet the needs of the other.

A second goal is to provide an improved method of representing the seafloor in the nautical charting process. By so doing, the process of product creation and quality control of a hydrographic survey can be significantly streamlined. This process can take thousands of staff hours for a full-sized survey. Much of this time is spent cleaning the data, checking the cleaning, and re-cleaning the data to a point where a shoal-biased sounding selection represents the seafloor as it is subjectively understood by the hydrographer. In addition, there are many manual processes involved with creation of the smooth sheet and other cartographic products that could be automated *[NOS Hydrographic Surveys Specifications and Deliverables, 2000; Nautical Chart Manual, 1992]*.

The third goal is to support emerging and future navigation products. These products may contain a three-dimensional model of the seafloor used for visualization, support creation of custom cartographic objects, or may contain a continuous time-variant tidal model.

CHAPTER I

CURRENT PRACTICE

In order to describe the Navigation Surface approach, it is necessary to first outline current practices and requirements of the hydrographic community as well as other disciplines that incorporate ocean mapping. For the purposes of this discussion, the term “hydrographic” will be used to refer to depth measurement and other sonar mapping for the purpose of safety to navigation, i.e. creating or updating a nautical chart.

Current Hydrographic Practice

> “The principal objective of most hydrographic surveys conducted by the National Ocean Survey is to obtain basic data for the compilation of nautical charts with emphasis on the features that may affect safe navigation.”
>
> **--**Hydrographic Manual 4th ed July 4, 1976

> “A basic requirement of a nautical chart is to promote safe navigation by providing the navigator with the proper information to assist in making the right decision at the right time…A nautical chart is a graphic portrayal of the marine environment. It is used by the mariner both as a “road map” and worksheet.”
>
> **--**Nautical Charting Manual page 1.10

To meet this objective, the systems and procedures used by hydrographers must ensure that depth information representing a potential hazard to navigation is carried through the entire data processing pipeline to be available for charting. This concern drives the form of the tools and procedures used through all steps of the process from raw sounding collection to chart production.

Hydrographic Data Processing

In the course of hydrographic data processing, a significant amount of time is spent in the processing chain "cleaning the data." This entails flagging as "rejected" those measurements deemed by the hydrographer to be invalid. The invalid measurements are the result of returns from real targets which are not part of the seafloor (fish, kelp, mooring chains, etc), effects of particular geometries (sidelobes on steep side-slopes), effects of poorly tuned sonars (multiple echoes), interference (other sonars, propeller noise, bubble wash-down), and *measurements that exceed the desired measurement error*, among others. The last of these causes is of particular concern in this thesis.

The bulk of hydrographic errors can be detected and corrected for by a variety of automated filters and manual processes, since they tend to be discontinuous with the rest of the measurements. Measurements that exceed the desired measurement error, however, occur in a continuum. If the hydrographer wants to create a final sounding set that is free of these high-error measurements, painstaking detailed cleaning must be done. This often involves rejecting measurements that are close to the seafloor. *Eeg [2001]* has shown that this can lead to over-cleaning of the data, removing valid measurements on small features. In addition, in a time-analysis study for a single survey, *Calder and Smith [2003]* demonstrated that in that survey, hydrographers were spending much more time per measurement cleaning data in flat deep areas than they did in rugged shallower areas, even though there were many more invalid measurements in the shallow area (see Figure 1). We believe this is typical, and the implication is that hydrographers are spending an inordinate amount of time cleaning measurements that exceed their expectation of the roughness of the seafloor or exceed their expectation of the noise level in the multibeam. This problem is exacerbated by the auto-scaling function built into most hydrographic editing systems.

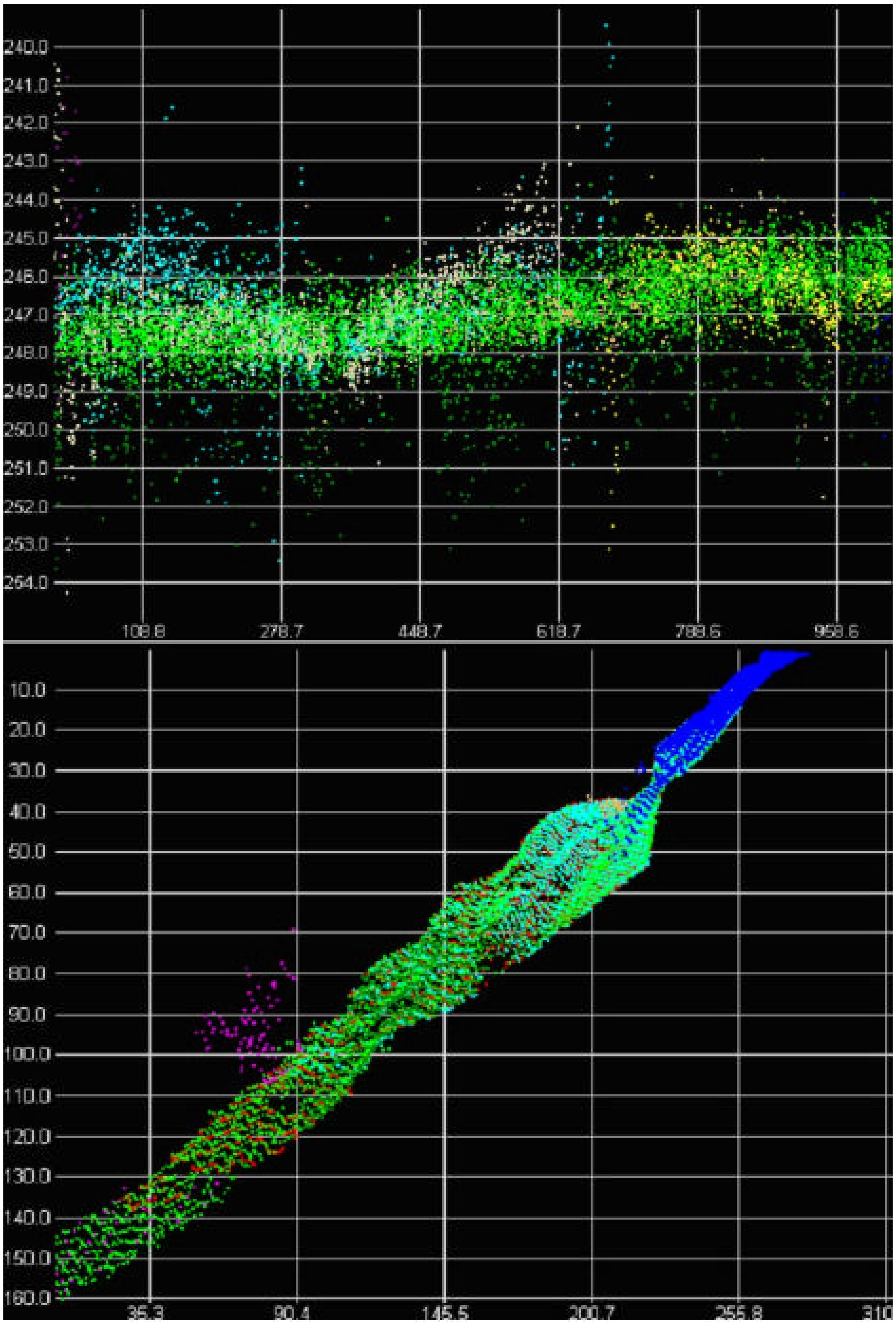


Figure 1 Relative Appearance of Multibeam Noise on a Flat Seafloor v. a Steep Slope. The top figure shows the sounding distribution on a flat seafloor. The bottom figure shows the sounding distribution on a steep slope. In order for a shoal-biased sounding set to adequately represent the seafloor, all shoal-side noise must be removed by hand.

After data are cleaned, all successive processing steps are performed to reduce the number of soundings under consideration. The first step is typically shoal-biased binning, where a single sounding is retained at its true position in each bin. For most modern NOAA surveys, a shoal-biased 5m bin (see Figure 2) is considered an archive data set *[NOS Hydrographic Surveys Specifications and Deliverables, 2000]*, and the one from which successive products are built. The next step is to downsample the binned data to plot scale where every sounding can be physically drawn on paper as a number in prescribed units at prescribed scale using prescribed rounding rules on a sounding plot. This plot is called a “smooth sheet” (see Figure 3).

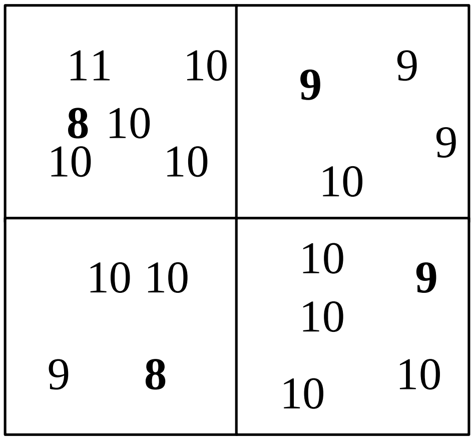


Figure 2 Shoal-biased Binning Process. In the northeast bin, the highlighted 9 was chosen arbitrarily over other equal soundings.

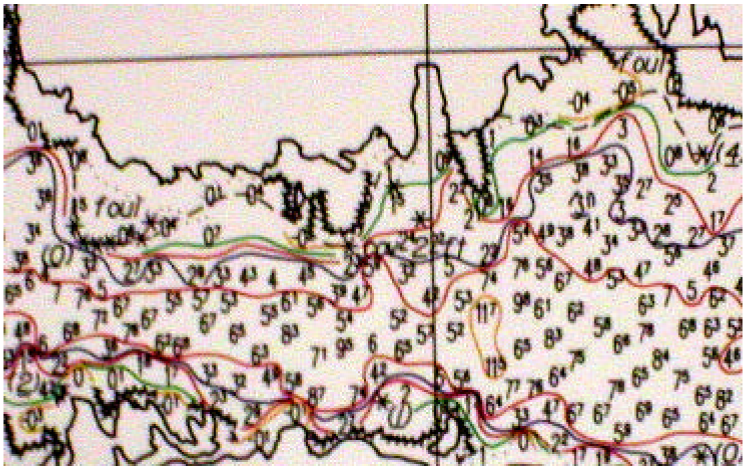


Figure 3 Smooth Sheet. The soundings are spaced at approximately 5mm at the scale of the survey. Contours were drawn by hand based on the visible soundings. Shoreline was derived from photogrammetry and reconciled by hand to the soundings.

The smooth sheet density data is then examined and any anomalous or critical soundings retraced back to their source to verify their accuracy. Depth curves (contours) are added at prescribed intervals using the smooth sheet soundings as the source for the curves. The curves may be machine-generated using a surface created from a Triangulated Irregular Network (TIN) of the smooth sheet soundings or drawn by hand. In the case of machine-generated contours, there are frequently artifacts in the curves caused by noisy shoal-biased source data, and the curves frequently require manual editing. Furthermore, any curves near the boundaries of the survey must be created manually. Other features (e.g., shoreline, point features, and other cartographic interpretation) are then added to the smooth sheet and reconciled with the bathymetry (see Figure 3). The hydrographic office then qualifies this plot as the official and authenticated product of the survey, and the only product that can be used to revise the chart.

Chart Compilation

Chart compilation is done in a Computer Aided Design (CAD) package. After fitting the CAD drawing of a smooth sheet into the projection of the chart, selected soundings and features from the smooth sheet are brought forward to the chart. This selection is done by hand, starting at the most important soundings for navigation “Critical Soundings” through a set of guidelines to “Fill Soundings” *[Nautical Chart Manual, 1992]*. Contours are drawn by hand using smooth sheet density soundings as source. The line is then smoothed to a pleasing shape at chart scale. Successively smaller scale charts use the next largest scale chart as the source in a similar fashion.

Current Practice of Other Disciplines

Marine Geology

Marine geologists use bathymetry to visualize the seafloor at a variety of scales. In order for the visualization to reveal the highest level of detail, it is critical that any internal inconsistency be removed from the data. A common approach is to create a mean grid or Digital Terrain Model (DTM) at a resolution commensurate with the detail in the source data. This has the effect of averaging out the measurement noise while preserving small feature detail key to understanding processes. The absolute depth with respect to datum is less important than the relative positions of adjacent nodes in the model. Figure 4 shows an image that was created from a 100m-resolution grid in approximately 8000m of water in the Puerto Rico Trench. The detail of the location, trend and physiography of the fault is more important to geologists than the absolute depth of the water *[ten Brink and Smith, 2003]*.

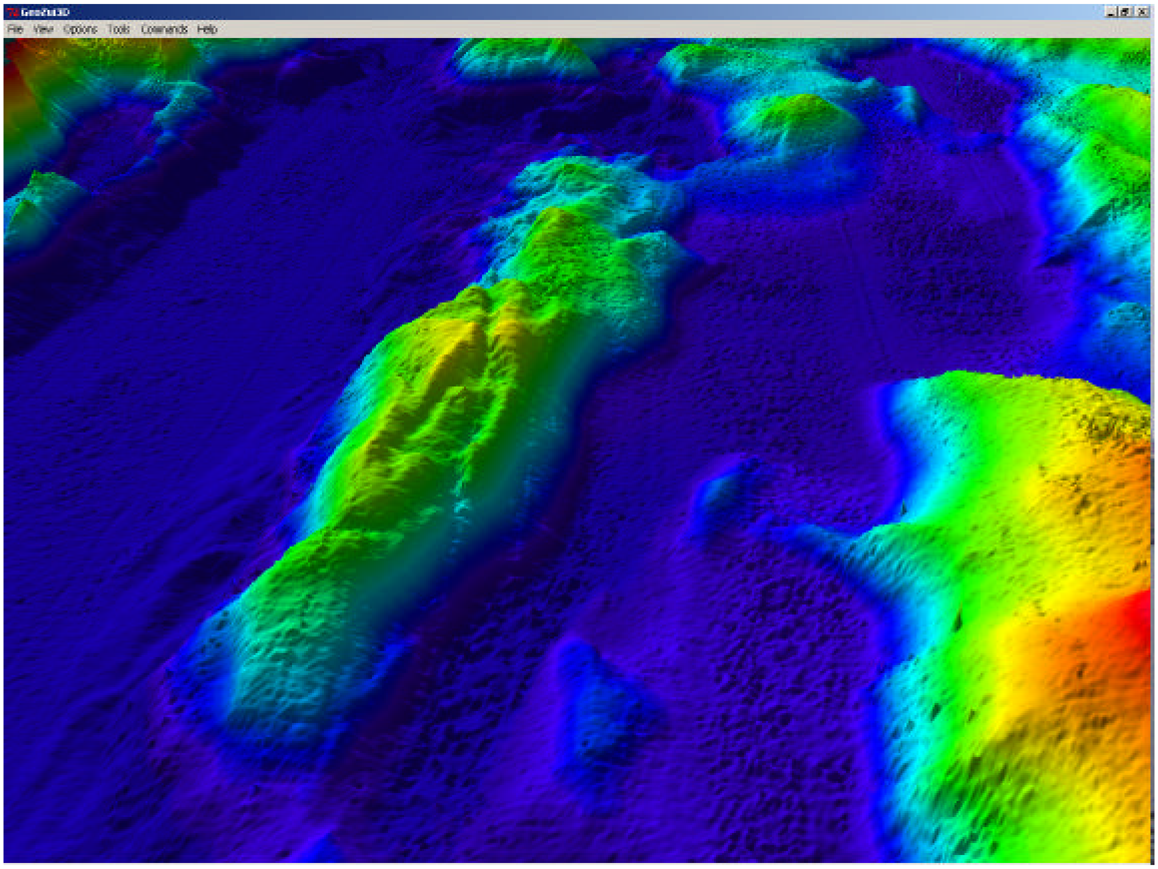


Figure 4 Puerto Rico Trench bathymetry at 100m-resolution in 8000m water depth. The location and physiography of the fault is more important than the absolute depth of the water.

Habitat Characterization

Marine biologists are increasingly using detailed bathymetry and backscatter to segment the seafloor into areas that have significance for fisheries habitat. The requirements for this sort of analysis are the same as for marine geology, and scientists typically use a DTM today. One approach currently being investigated at the University of New Hampshire uses a Local Fourier Histogram on a high resolution DTM as a segmentation vector. [cite G. Cutter] In this case, the local variability in the DTM is being used directly, and any systematic and random errors remaining in the data set would have a significant effect on the segmentation for habitat.

Marine Archaeology

Marine Archaeology has traditionally used backscatter imagery from sidescan sonar as the primary tool for rapid reconnaissance mapping of underwater artifacts. As the resolution of multibeam sonars has increased, some researchers have used high-resolution bathymetry to map archaeological sites in shallow water *[Mayer et al, 2003]*. In these cases, a high resolution DTM was used to visualize and analyze the depth data that contained archaeological targets.

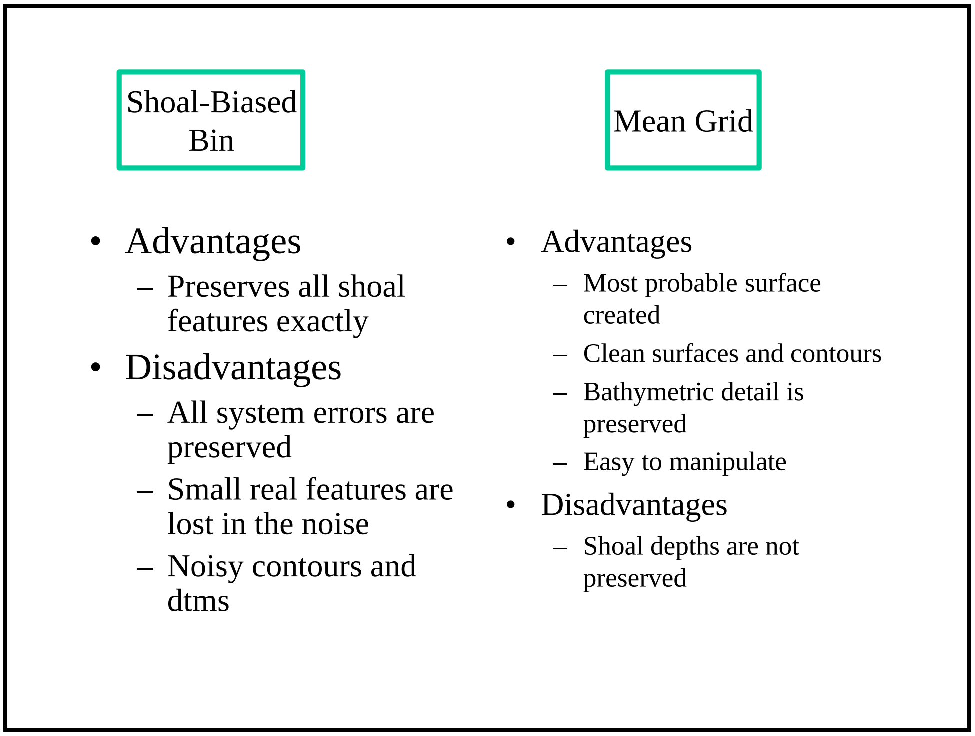
Marine Modeling

Several disciplines use bathymetric models as a boundary condition for an algorithm which creates a model of some other value, such as current, sediment transport, acoustic propagation, or pollution dispersion. Often the modelers are forced to use poor quality DTMs derived from the nautical chart because high-resolution DTMs are not available to them, even though they could have been created from the data that was gathered for the chart. The resolution and the accuracy of the DTM may have a large effect on the performance of the prediction algorithm.

Military Applications

There are military uses for all the above-mentioned applications. In addition, there are requirements for change detection and for the high resolution DTM itself. Though it is an area of research, there are military surveyors using high resolution DTMs to locate mine-like-objects with some success *[Brissette et al, 2001; Mayer et al, 2002]* Martha's Vineyard. The approach used by Brissette is to create pseudo-sun-illuminated images from DTMs at high resolution and display before and after surveys alternately and repeatedly, allowing the eye to do the comparison. This reduces the false contact ratio and improves the probability of detection by reducing the effect of small horizontal and vertical errors.

Bridging Disciplines



Shoal-Biased Bin

- Advantages
  - Preserves all shoal features exactly
- Disadvantages
  - All system errors are preserved
  - Small real features are lost in the noise
  - Noisy contours and dtms

Mean Grid

- Advantages
  - Most probable surface created
  - Clean surfaces and contours
  - Bathymetric detail is preserved
  - Easy to manipulate
- Disadvantages
  - Shoal depths are not preserved

Table 1 Comparison of Shoal-Biased Binning to Mean Gridding.

The products used by the hydrographic community and those created by other ocean mapping communities are often incompatible. Important detail is lost to the other communities when shoal biasing and downsampling of the data is performed. Least depths on important shoal features critical to hydrographers are lost through averaging in the process of creating DTMs using the normal methods of today. However, there are some features of an average surface that are attractive to hydrographers as well, such as the ability to create clean contours, the ability to visualize the data easily, reduced noise (through calculation of mean), and the ability to preserve bathymetric detail. As described in Table 1, the Navigation Surface database is designed to preserve the advantages of both of these approaches.

Summary of Limitations of Current Hydrographic Practice

- Digital Terrain Models that could be most useful to non-navigation users of bathymetric data are not systematically created and preserved. The smooth sheet density soundings are not suitable for most non-navigation uses. See Figure 5.

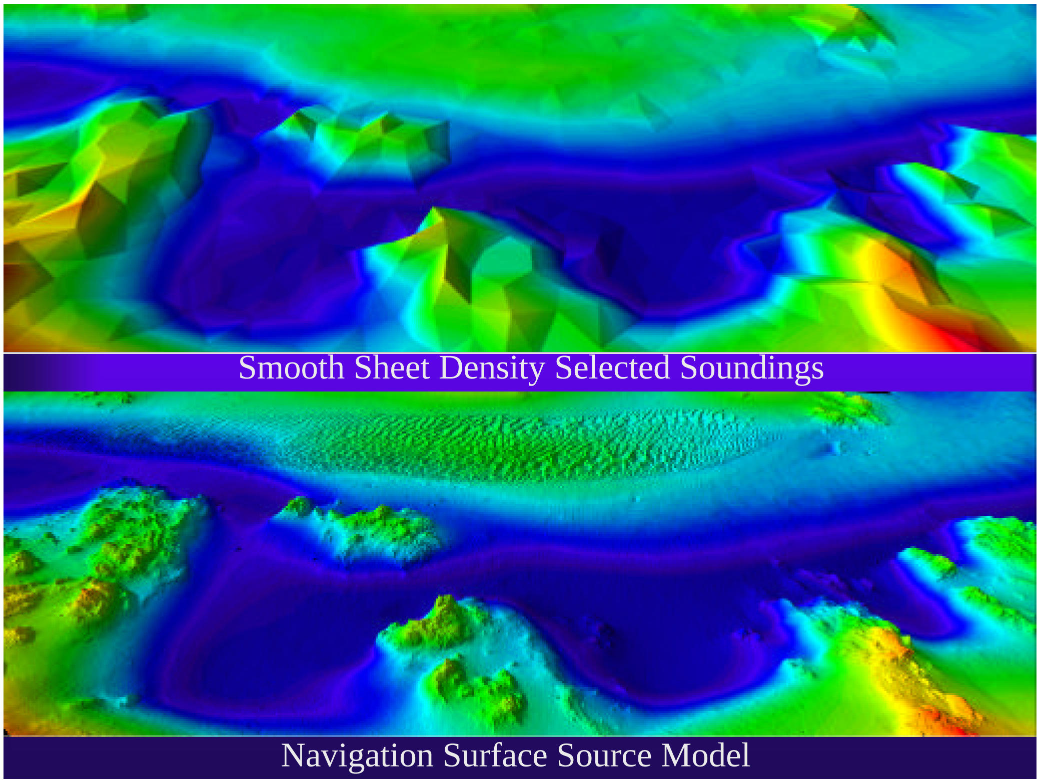


Figure 5 Comparison of a grid derived from a TIN of smooth sheet density soundings (top) and a high resolution mean grid derived from all soundings.

- Shoal-biasing the product preserves noise in the data, distorting the result for both navigation and non-navigation users
- Manual validation and cartographic processes require repetitive, manual, quality control and a significant time expenditure
- Depth data is limited to presentation at a predetermined scale and smaller, regardless of the detail of the data collected.

CHAPTER II

NAVIGATION SURFACE

The Navigation Surface database is designed as a multipurpose database with the following properties:

Single Form – The database consists of a collection of bathymetric models. Each node in each model contains an estimate of depth and the uncertainty of the estimate of depth. The resolution of the model is the highest resolution that the source data will support. In areas critical to navigation, the models have been adjusted to match the hydrographer's understanding of the least depth.

Varied Sources – The modeling technique used to create the source model depends on the source. Techniques for modeling sparse data (singlebeam and leadline surveys) and high-resolution data (multibeam and LIDAR) are discussed later in this chapter. In addition to modeling techniques appropriate for clean data, the database is compatible with the depth and uncertainty grids created by CUBE (Concurrent Uncertainty and Bathymetry Estimation) *[Calder, 2001]* from measurements which have not been cleaned. The projection, datum, units and coordinate system of each model in the database are independent; each model retains its original properties.

Varied Products – The database is deconflicted each time it is populated with a new model, so there is only one best estimate of depth at any location. A product definition includes an area definition, projection, vertical and horizontal datums, resolution, depth units, and intended purpose. Based on these properties, a product model is constructed and populated by re-sampling from the database after applying appropriate transformations. The product model is then manipulated, if necessary, depending on the intended purpose of the product. Finally, if cartographic objects (contours, selected depths, or depth areas) are required, they can be extracted automatically from the product model.

Most hydrographic offices recognize the need for a comprehensive database from which to produce their multiple products and streamline data management. The databases vary considerably from hydrographic office to hydrographic office, but they all share one common feature; they are GIS databases that store attributed points, lines and areas representing soundings, other hydrographic features, and pre-compiled cartographic objects. As an example, the Royal Australian Navy has compiled "a validated, non-conflicting digital data base from which products and services can be generated." *[RAN Internal Report, 2002]* As hydrographic offices move to higher density datasets, the databases get to be very large if every sounding is maintained.

There are three main advantages to a database of models instead of a database of soundings. First, collections of billions of soundings are cumbersome. Second, the billions of soundings are not in agreement with each other. Soundings from adjacent lines in the same location will in general differ. The database needs to be able to report a single depth at a point on earth. The process of reconciling conflicting information during the charting process is called "deconflicting". A sounding is made up of the combination of the actual depth of the water and the measurement noise of the system. It is expected that soundings of the same area of seafloor would differ by an amount proportional to the expected measurement error. However, if the database is to be deconflicted, then inconsistent soundings within the same survey must also be deconflicted. The best time to deconflict the survey is when all information about the soundings is still available. Third, by modeling the bathymetry, it is possible to preserve the information about the seafloor while suppressing the measurement noise. A carefully constructed model retains all the bathymetric information contained in the soundings in a more compact form, and the soundings themselves can be removed to a long-term off-line archive.

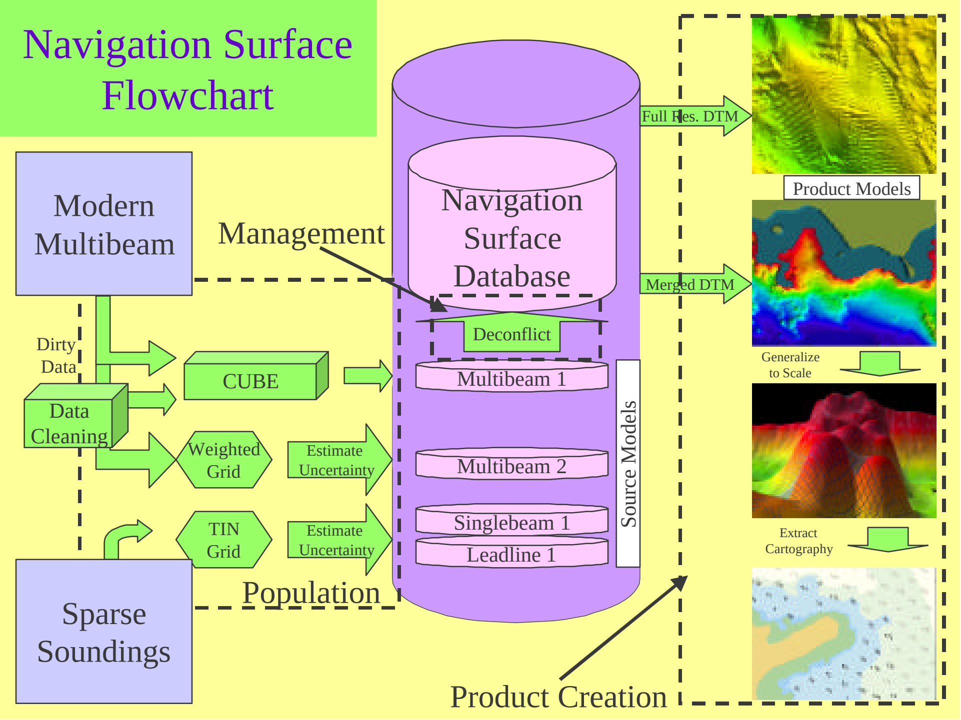


Figure 6 Navigation Surface Flowchart. The purple cylinder in the center is the database, the processes on the left are used to populate the database, the processes on the right are used to make products from the database.

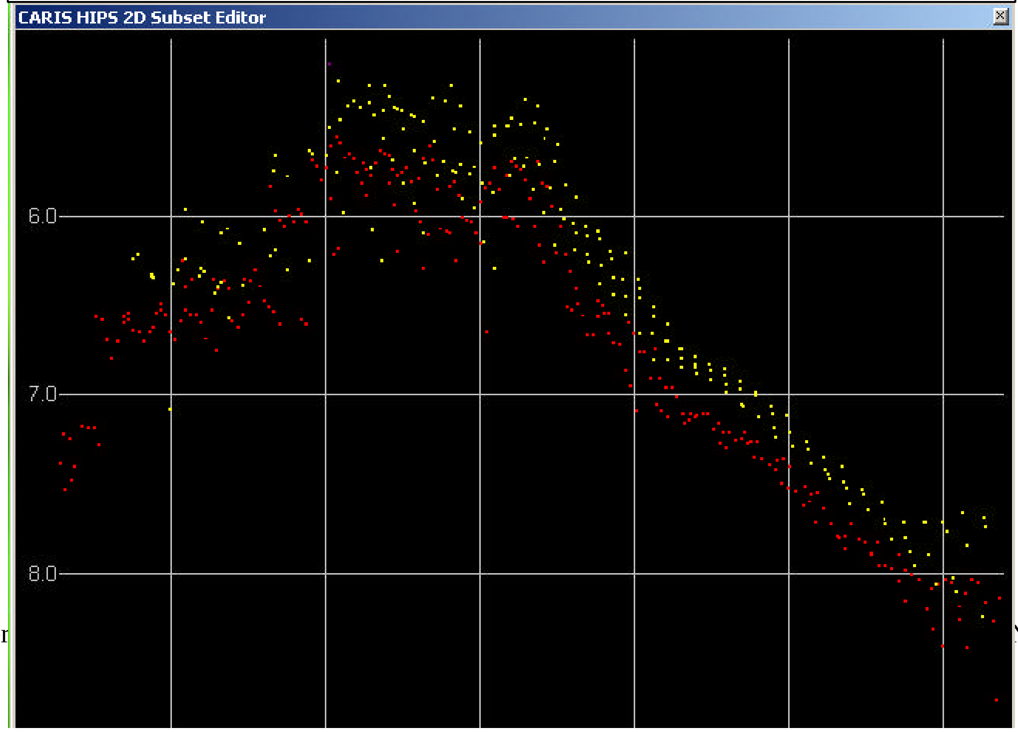
Creating the Source Models

High Resolution Data

Cleaned Data Depth Modeling-The modeling technique used for this project was the weighted gridding technique used by Caris HIPS ® software *[Caris HIPS User's Guide]*. This technique computes a mean depth, weighted by off-nadir angle of the sounding and distance of the sounding from the node. It is described in *[Hughes-Clarke, 1997]*. This technique was designed to create a model which honors the resolution of the varying beam footprint of a multibeam system.

When traditional downsampling techniques are applied to modern multibeam data, the result is biased in the shoal direction by the difference between the shoalest validated measurement and the mean. The difference varies up to the threshold where an operator would declare it an outlier and remove it from further consideration. This threshold, however, is subjectively determined and inconsistently applied during data cleaning. In one typical example (see Figure 7), at the end of a survey area farthest from the

Figure 7 Two separate passes on a shoal feature, shown in different colors. In this case, the difference was caused by tide modeling errors away from the tide gage. The traditional shoal-biased approach would retain the shoalest sounding from the yellow pass, while a mean grid would depict the seafloor as a mean of conflicting soundings.



tide gage, a 0.3m difference was noted between adjacent survey lines collected on different days. The problem was determined to be due to tidal zoning problems that fall within the error budget estimated for tides. Each line was determined to meet specifications and all measurements were accepted. The traditional hydrographic process would select a set of soundings along the top of the yellow line to represent the results of the survey. The weighted mean grid produces a result that takes into account both lines and reports a value between the two.

The potential danger with the weighted mean gridding approach is that critical shoal soundings will be averaged with other soundings in the general area, thereby masking the critical sounding from consideration in any navigation product. There are three properties of the Navigation Surface process that can mitigate this danger. First, the resolution of the model is chosen to approximate the resolution of the sonar. As a result, the extent to which unrelated soundings are integrated into the model is limited. Second, the distribution of the measurements is captured as part of the modeling process. (For more discussion, see “Uncertainty Modeling” below). Third, the hydrographer is given an opportunity to select individual soundings that will be honored at the nearest model node. (For more discussion, see “Golden Soundings” below).

Clean Data Uncertainty Estimation-The goal of the uncertainty model is to estimate the accuracy of the depth reported at each node. Each sounding can have an estimated propagated error based on the errors of the individual sensors combined to estimate the final depth. For example, an error in roll measurement propagates through the depth solution, affecting the measurements farthest from nadir most significantly. Similarly, an error in tide measurement or modeling affects all soundings taken in the area at the time in a equal way. A general-purpose multibeam error model was developed for the Canadian Hydrographic Service *[Hare et al, 1995]*. This error model addresses the expected accuracy of the soundings. Estimation of the uncertainty of a depth model derived from these soundings depends on the depth modeling technique.

If it could be assumed that all soundings are 100% independent, then a mean of the soundings would have an uncertainty lower than the uncertainty of the individual measurements. However, for multibeam collected using current practice, assuming independence of geographically adjacent soundings is not always justified. It is most likely that adjacent soundings are from the same ping. In this case, the two soundings share the same attitude measurement, sound velocity profile, water level estimate, dynamic draft estimate, and static measurements. The only portion of the sounding that is truly independent is the sonar measurement itself. Any error in each of the other values would be the same in both soundings, so the soundings are correlated, significantly dependent on the same sensor measurements. Conversely, soundings from different pings, different lines, different days, and different vessels are increasingly independent. Integration of a test to measure correlation of soundings is beyond the scope of this project.

In the absence of any test for independence, the conservative method is to assume that the measurements are highly correlated. As a result, the estimate of the uncertainty of the model node is simply the average uncertainty of the soundings that were used in the calculation of the depth. For this study, a simple mean was used, but it would be more rigorous to use the same weighting used for the depth model. Uncertainty derived in this way is merely a prediction of uncertainty based on the measurements. In the case of a single sounding supporting a model node, the uncertainty of the model node is simply the total propagated error of the sounding.

When more than one sounding supports a node, we can check the predicted uncertainty by comparing it to the observed distribution in the soundings. If we assume that the distribution of soundings is normal, the 95% bound can be estimated by multiplying the standard deviation by 1.96. In those cases where this scaled standard deviation is greater than the predicted uncertainty, it is appropriate to report the greater value as the uncertainty of the node. In this way, areas with residual random or systematic errors are identified.

This uncertainty surface can be used during the conduct of the survey to prioritize further work and to meet a pre-defined standard. By looking at the uncertainty surface in the context of the depth model, areas of high uncertainty can be further investigated. For example, it is common for a small horizontal error on a steep slope to cause an apparent high vertical uncertainty. This is a completely natural and acceptable result of the uncertainty methodology outlined above. The problem is only apparent when we try to apply a standard to the model that does not allow for more uncertainty on the slopes. Traditional hydrographic practice dictates that comparisons between crossing sounding lines are most valid on “flat or gently sloping bottom” *[Umbach, 1971]*. Furthermore, higher uncertainty on steep slopes is not critical for navigation, since the generalization of the product itself mitigates the vertical uncertainty. This is discussed in more detail in the product creation section. For other users interested in small detail on steep slopes, it is a fundamental limitation of modern systems that we cannot reliably resolve small features on steep slopes when the horizontal error is large.

Combined Uncertainty and Bathymetry Estimation (CUBE) -- Calder *[2002,2003]* has developed a robust method of creating a model of bathymetry and uncertainty directly from unedited data. CUBE works by integrating all measurements in a location into separate, internally consistent hypotheses of depth. CUBE then selects the most likely hypothesis to represent the depth at that node.

After CUBE and the Navigation Surface were presented at the 2nd International Conference on Surveying in Shallow Water in Portsmouth, NH in September 2001 *[Calder, 2001; Smith, 2001]*, NOAA and UNH agreed to undertake a study of three surveys processed with CUBE and Navigation Surface and compare them to the traditionally processed smooth sheets. For these studies, CUBE was used to create the depth and uncertainty models, and the Navigation Surface was used combine and generalize multiple CUBE models, and to join them with singlebeam and shoreline where applicable.

The first survey, a large multi-platform multibeam survey in Snow Passage, Alaska *[H10949]* was collected by the NOAA ship RAINIER. The goal of this trial was to compare the results of the surveys done traditionally to that done through the new CUBE/Navigation Surface procedures. This comparison posed two questions. First, did the modeled product contain the same navigationally critical information? Second, is it possible to integrate the results into NOAA's existing data flow? The results of the comparison were reported at the Canadian Hydrographic Conference 2002 *[Calder and Smith, 2002]*.

The most significant result of the comparison was that choosing the correct grid resolution is critical to achieving an adequate result. In particular, when CUBE was run using a 5 meter regular grid over the entire survey area, significant detail was lost in the shallow areas, including least depths on small features. When a smaller shallow area was selected to be run at 1m resolution, the results matched well with the least depths from the smooth sheet. It is not reasonable to expect that two radically different processes to produce identical results so the goal was to create comparable results and the IHO Order specified for the survey was used to determine an acceptable difference. For example, the IHO vertical error limit for Order 1 surveys in 20 m of water is 0.56m, 95% of the time. With the 1m resolution model, 95% of the smooth sheet soundings were, based on IHO tolerances, consistent with the CUBE grid. This was proposed as being consistent with the specifications for the survey.

Once the entire survey was reprocessed using a resolution approximating the nadir footprint in the area, a further comparison was conducted. Using the navigation surface product creation tools described in Chapter 3, a depth plot was produced which closely mimics the current smooth sheet. As an independent assessment of comparability, personnel at NOAA's Pacific Hydrographic Branch compared the soundings on the CUBE/Navigation Surface-produced depth plot to the soundings ultimately chosen to create the chart. The results of the comparison were reported in an internal memo, where the reviewer noted, "In all cases, the navigation surface depths met the criteria for the 95% confidence level." The report also noted that the Navigation Surface depths are consistently deeper than the shoal-biased smooth sheet, but within the IHO Order 1 accuracy requirements. It is expected that a "most probable" surface would be deeper than the shoal biased measurements. This final comparison not only demonstrated that the content of the model was comparable to the traditional smooth sheet, but that the form was compatible with current practice.

The second comparison was done on data from Woods Hole, Massachusetts, surveyed by the NOAA Ship WHITING *[H11077]*. It is a fairly small area with a depth range of 2-30m. The purpose of this trial was to assess CUBE's ability to resolve very fine features. The survey area contains sand waves and other features at all scales, isolated rocks of various sizes, and cultural features. Since the traditional smooth sheet does not contain detail at this level, this was mostly an internal trial, tuning some of the CUBE algorithms to produce the best result. This effort was described in *Calder [2003]*.

The third comparison was of a near-real time implementation of CUBE and the Navigation Surface on the NOAA ship RAINIER in Valdez Narrows, Alaska in September 2002. The primary purpose of this trial was to compare the level of effort involved with cleaning data in the traditional way to creating a surface with CUBE *[Calder and Smith, 2003]*. Although this is only a single data point, this study led us to expect about a 10-fold increase in the rate of editing data. There are two additional consequences of this processing acceleration. First, the total time spent cleaning, before and after leaving the survey area, will now comfortably reside within the time currently spent processing while the survey is in progress. As a result, the in-process time for data on its way to the chart should be considerably reduced. Second, while the total processing load is reduced, the percentage of total tasks requiring an experienced hydrographer has increased significantly.

In addition to comparing level of effort, the Valdez project provided a testbed to try out the Navigation Surface tools to combine singlebeam and multibeam data. The procedure started with a CUBE grid and a clean singlebeam sounding set. First, any node from the multibeam that was missing any neighbors was exported as a point attributed with uncertainty. Second, the singlebeam lines, shoreline, and the missing neighbors file were combined and used to create a TINned (Triangulated Irregular Network) grid attributed with uncertainty as described later in this chapter. Third, this new grid was merged with the original unmodified multibeam CUBE grid, honoring the multibeam CUBE grid where nodes from both grids were populated. The result is a fully populated depth and uncertainty grid with shore-to-shore coverage.

The combined conclusion of these three comparisons is that the CUBE/Navigation Surface process is both rigorous and efficient for processing hydrographic surveys. In addition, it demonstrates the validity of a hydrographic approach that uses a model and supports the concept of Navigation Surface source models.

Sparse Data, Data Gaps, and Shoreline

The Navigation Surface database is conceived as a broad database with multiple uses. Because high-resolution data (multibeam and LIDAR) currently cover a relatively small percentage of navigable waters worldwide, there needs to be provision for incorporating sparse historic data into a model-based database. Historically, the process in hydrographic offices has been the opposite. The form of the smooth sheet was designed as an efficient way to portray the results of a leadline survey, including shoreline, projection information, depth curves, buoys, shoals, geographic names, etc *[Coast Survey, 1878]*. The scale of the sheet was chosen to display most of the soundings taken. When continuous vertical beam echosounding was introduced into hydrographic offices, the procedure evolved to subsample the continuous trace along line to make it compatible with cartographic procedures designed for leadline *[Hydrographic Manual, 1942]*. Similarly, when 100% coverage multibeam was introduced, a similar procedure was introduced, to suppress excess soundings and reduce the sounding set to one compatible with leadline-derived cartographic procedures *[Specifications and Deliverables, 2000]*.

The aim of the Navigation Surface database is to build the database and products around high-resolution modern data. Instead of attempting to fit multibeam data into a process designed for leadline surveys, the leadline data can be fit into a process designed for multibeam surveys. The goal in this approach is to generalize the concept of a depth and uncertainty model to include sparse data and the uncertainty of both measured and unmeasured areas.

The general case in question is where individual soundings are surrounded by relatively large areas where no soundings have been made. A bathymetric model can be created that honors the soundings and estimates the depth between the soundings. This model is then resampled to a regular grid and uncertainty is estimated at each node. In this way, the form is compatible with the grids created by CUBE and Navigation Surface above. This basic technique is applicable to sparse data from leadline and vertical beam echosounders, as well as discontinuous multibeam surveys.

Pickrell [1979] compared a number of methods for modeling sparse data specifically to support nautical charting. His primary goal for the modeling was to minimize the amount of digital storage necessary to represent the seafloor. He proposed using analytical models as a substitute for soundings in a database, and examined a variety of methods for producing such models. He concluded that these analytical models could save considerable storage space with a negligible distortion of the seafloor surface when compared to a selected sounding set. However, no Hydrographic Offices adopted this approach.

In the mid 1990s, Peter Kielland of the Canadian Hydrographic Service investigated the use of kriging to model both depth and uncertainty for modern singlebeam *[Kielland, 1996]*. The idea was to use the along-track roughness at different scales (a variogram) to estimate the depth and uncertainty between sounding lines. In this way, the uncertainty would grow more quickly in areas with more irregular seafloor. However, due to the way data has been quality controlled and archived in the US, only the smooth sheet soundings are typically preserved in accessible form. These data tend to be fairly uniform in spacing and do not lend themselves to a variogram. Areas that do vary in density tend to be those areas where the hydrographer noted seafloor irregularity and decreased the line spacing. In an experiment using using kriging on archived smooth sheet-density data in the US, we found that unusually shaped features were produced based on only a few data points. This sort of behavior, while mathematically most likely, has properties which make is unhelpful for this particular database.

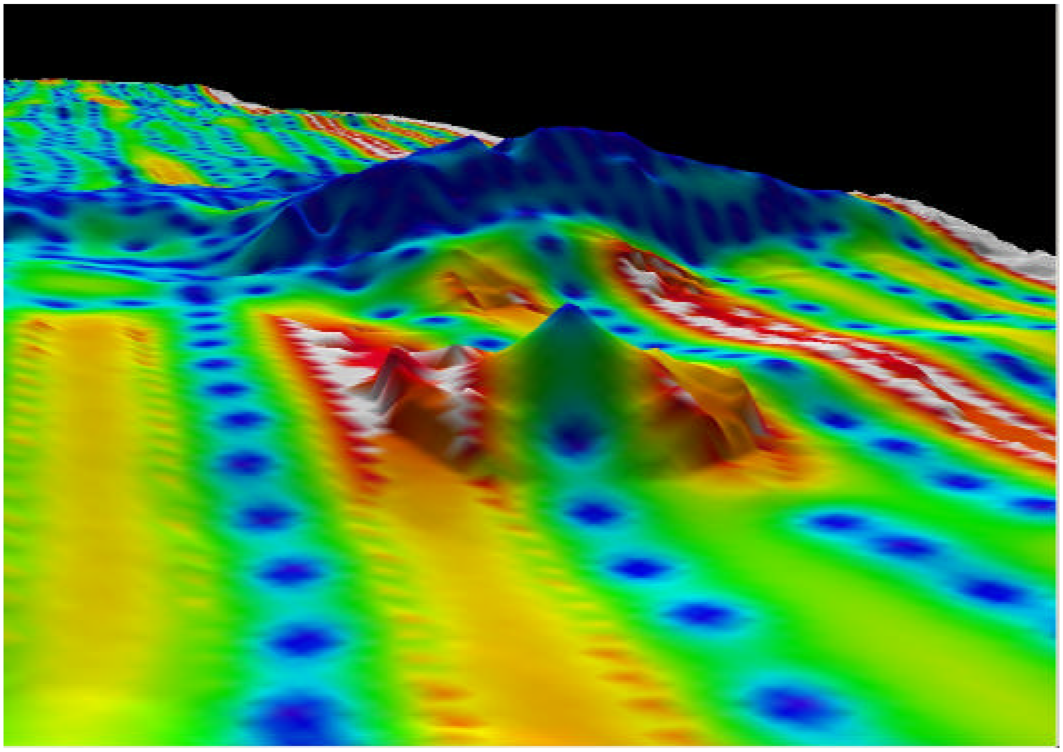


Figure 8 Depth and uncertainty model created by kriging smooth sheet density data. The model is colored by uncertainty. The sombrero-shaped feature in the center of the image is based only on a few data points. There is a danger in over-interpreting the model.

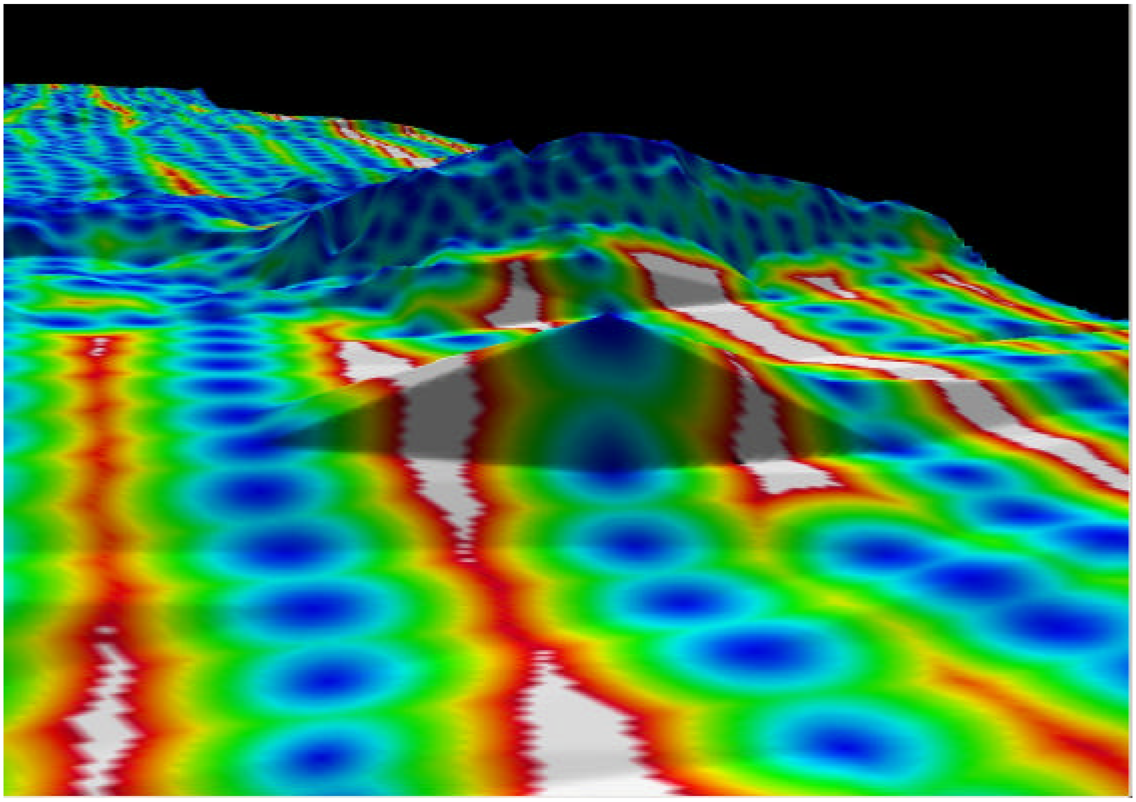


Figure 9 Depth and uncertainty model based on a simple TIN. The model is colored by uncertainty. This model clearly shows what is known and unknown.

In Figure 8, a sombrero-shaped feature can be seen in the foreground. The lines of blue areas are the lines of soundings that were used to create the surface. As can be seen, there is no direct support in the data for the rim of the feature, yet there is a complex structure that might be misinterpreted if incorporated in the database.

One of the useful properties of the Navigation Surface database compared to a chart-scale selected sounding set is that it does not hide what we do and do not know. The introduction of subtle gridding artifacts might tend to distract some users from the fact that the data are really insufficient. In addition, from a practical point of view, the Navigation Surface database must be accepted by a conservative and skeptical community of hydrographers. Gridding algorithms that require too many input parameters and complex structures would meet stiff resistance. There is some historical precedent in Hydrographic Offices for using Triangulated Irregular Networks (TIN) to create surfaces for visualization and contouring, and most hydrographers are comfortable with the approach, and hence it will meet less resistance. In addition, if the type of interpolation between sounding lines makes a difference for navigation, then the sounding lines are likely too sparse to support current use of the chart in the area.

The challenge is then to estimate the uncertainty for the interpolated nodes. The properties of the uncertainty estimation should be:

- For a node coincident with a measurement, the node uncertainty should be set to the measurement uncertainty
- The uncertainty should grow as a function of distance from the measurement.
- The uncertainty should grow more quickly in areas with an irregular seafloor.

A simple quadratic function of distance from the nearest node and the local seafloor roughness was used.

$\sigma_i=\sigma_m+ad+bd^2$

where

σ_i =uncertainty estimate at the interpolated node

σ_m=uncertainty of the nearest measurement

b=arbitrary coefficient, set empirically

a=linear coefficient, which is a function of local seafloor roughness

$$a=c\sqrt{\frac{\sum z*z}{n}-\left(\frac{\sum z}{n}\right)^2}$$

where

c=arbitrary coefficient, set empirically

z=depth at a node

n=number of nodes over which the roughness is computed.

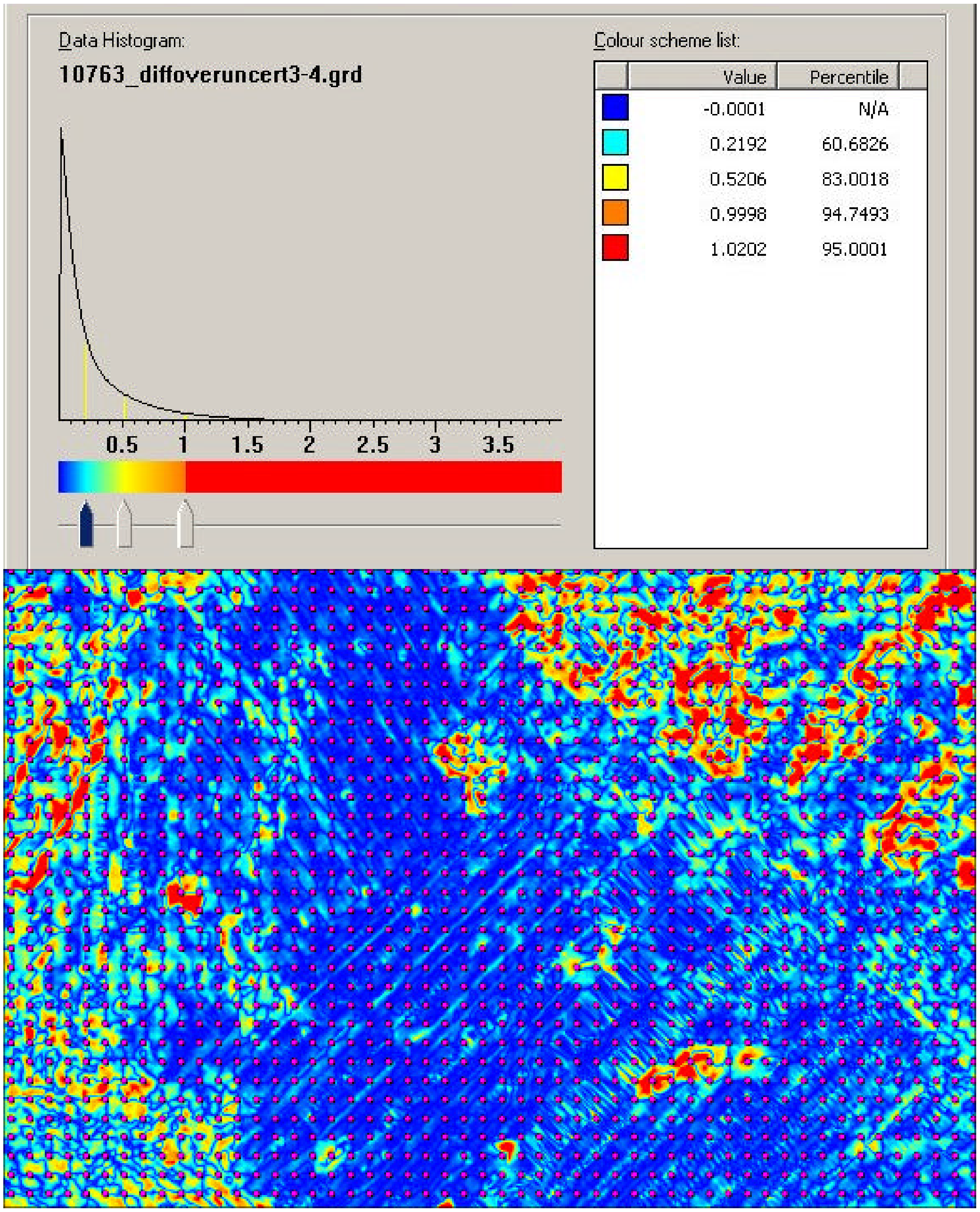
In this equation, b, c and n were set empirically as described below. The number of local nodes, n, was set to 900, a 30×30 node box centered on each node. This value is not computationally overwhelming, but does capture a fairly broad area with respect to the scale of the grid to prevent individual problem nodes from unduly influencing the result. The quadratic coefficient b was chosen so that the uncertainty would grow very quickly far away from the nearest measurement. The roughness coefficient c was chosen empirically by comparing a modern multibeam survey to a hypothetically constructed singlebeam survey of the same area made by sub-sampling the multibeam. At each node, the difference between the two surfaces was compared to the estimated uncertainty of the interpolated surface.

By overlaying the soundings from the singlebeam survey with a grid of the ratio of the depth difference to the reported uncertainty, an evaluation of the uncertainty model can be made. A histogram of the ratios should show 95% of the nodes at a ratio less than 1; that is, the actual difference is predicted in the uncertainty 95% of the time.

In order to estimate reasonable values for the coefficients b and c, a survey *[H10763]* was chosen which contains a variety of seafloor types. Two different grids were created, one at the full resolution of the survey, and another at typical sounding spacing for a 1:10,000 scale survey of 50m. This hypothetical singlebeam survey was TINned and uncertainty estimated as if it were a single beam survey. The difference between the two depth grids was computed and compared to the estimated uncertainty of the hypothetical singlebeam survey. The values of b and c were then adjusted so that the ratio surface contained few systematic problems and 95% of the comparisons were less than 1 (refer to Figure 10). The values selected were:

$b=.0004$

$c=1/120$



	Value	Percentile
🟦	-0.0001	N/A
🟦	0.2192	60.6826
🟨	0.5206	83.0018
🟧	0.9998	94.7493
🟥	1.0202	95.0001

Figure 10 Ratio of estimated uncertainty to difference between high resolution multibeam survey and artificially constructed sparse survey of the same area. The histogram shows the estimated uncertainty is greater than the difference for 95% of the nodes

While the results of the comparison and the coefficients used are valid in this case (95% of the uncertainty estimates are greater than the difference between the two depth grids), it will certainly be possible to improve the above results, especially by generalizing to a wider set of circumstances and seafloor types. Of particular concern are the flat areas where the error model over-predicts the uncertainty systematically. A different form of error model with a stronger function of local roughness might result in a closer estimate.

“Golden Soundings” – There will be circumstances where the model derived depth estimate is in conflict with the hydrographer’s understanding of a feature. This might happen when a feature (such as a rock or wreck) has finer spatial detail than the model can depict, when a definitive measurement was made (by a diver, for example), or when a least depth has critical importance for navigation. In these few manually picked cases, a particular measurement from the dataset is chosen to represent the seafloor at that location, and the nearest node in the model is changed to the exact value of the measurement.

Database Manipulation

The database contains a collection of models of bathymetry and uncertainty. Each model resides in its original form, resolution, datum and projection. Transformations to a desired form will be done at the time of product creation and deconflicting.

Time-dependent Uncertainty

The uncertainty reported for each node in a model is valid at the time the survey was conducted. As time passes, the seafloor in some places changes significantly, due to natural processes and direct and indirect human impacts. Usually, a geologist or a hydrographer can classify areas of high likelihood of change from those with low likelihood of change. To the extent we can bound the expected rate of change, we can increase the uncertainty of the model in the area as a function of time elapsed since the survey. For example, it would be reasonable to assume that rocky outcrops in New England are very stable, so the uncertainty growth with time would be very small. An area of frequent migration of shoals in Cook Inlet, Alaska, might have an uncertainty growth of a meter per year.

It is also reasonable to cap the uncertainty growth based on knowledge of the natural processes. For example, large migrating sand waves might move around significantly, but because little new material is moving into the system, the uncertainty growth could reasonably be capped at the height of the observed sand waves.

Deconflicting the Database

The value of a comprehensive bathymetric database is significantly enhanced if it is deconflicted. In this sense, deconflicted means that there is only one depth in the database for a particular point on earth. Any conflicting information from within a single survey has already been removed through the process of creating a model, described earlier. When each new survey is checked into the database, it needs to be reconciled with any existing models in common area. In general, for a newer high resolution survey in common area with an older singlebeam or leadline survey, the new survey would completely supersede the older one. In more complex cases, the database manager might decide to retain portions of the older survey if the newer survey was not completely adequate to supersede. This might happen for example, where an historic survey discovered a rock and the new survey did not adequately verify or disprove its existence.

In principle, the decision as to whether or not to supersede should be made on the basis of node uncertainty. New higher resolution multibeam surveys would supersede older singlebeam surveys because every node has measurements associated with it (and the nodes have generally lower uncertainty at the time of measurement), and also because the older singlebeam survey has more time-dependent uncertainty associated with it.

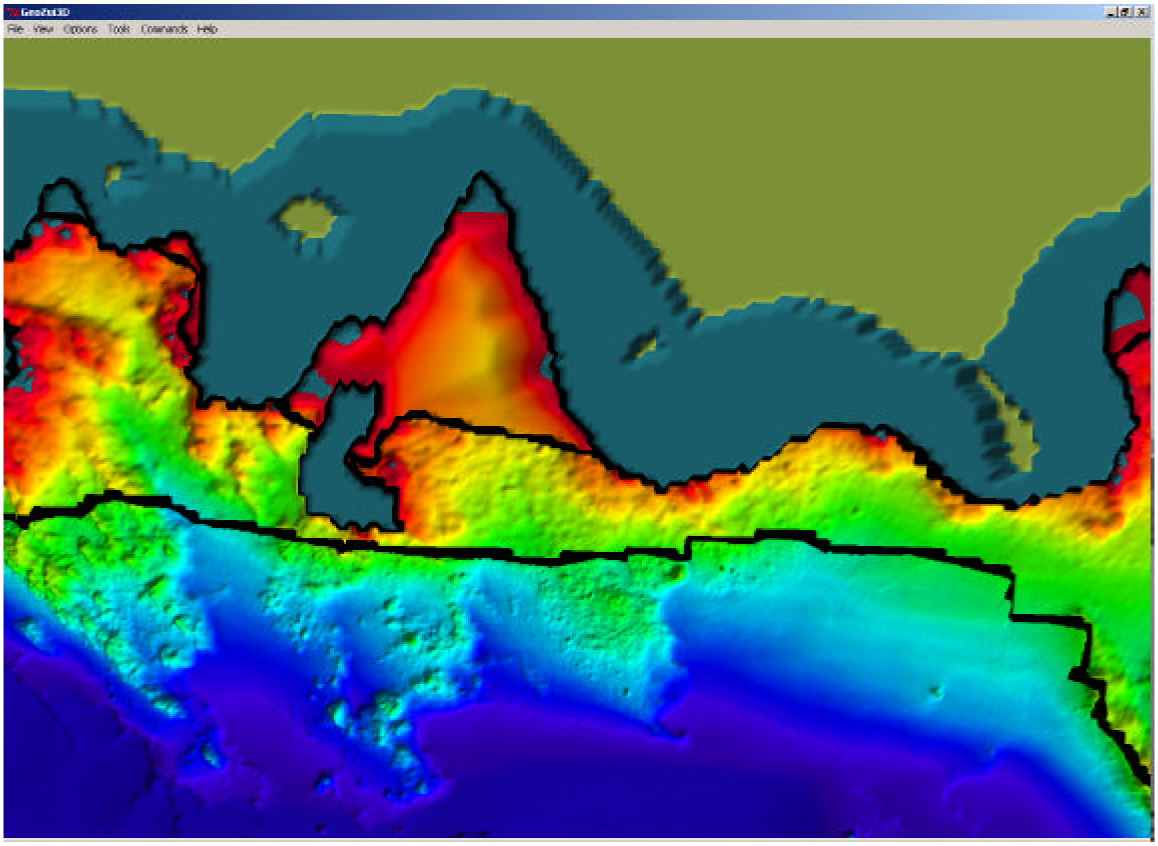


Figure 11 Deconflicted source models in the database. Each model is retained at a resolution appropriate to the survey method.

Datum Management

Because each survey is kept in its original horizontal and vertical datum, both a problem and an opportunity have been created. The problem is that transformation must be done on each model every time a product is created. The opportunity is that it is possible to take advantage of improvements in datum transformations like the VDatum project to model vertical datums currently underway at NOAA *[Parker et al, 2003]*. Surveys can be processed with respect to the ellipsoid or with respect to a traditional tidal datum. A product can be defined with respect to any datum and all models transformed to it.

Product Creation

Merging and Downsampling

The database consists of a collection of deconflicted models, each at their original resolution. The first stage in product creation is to define the properties of the product required, including scale, contouring rules, projection, expected use (navigation, modeling, etc). Once that is done, a simple query into the database will return a new model designed to the product specifications.

For a navigation product, a new grid at an appropriate resolution for the product is constructed. A resolution of 0.5mm at the scale of the product (approximately a line width) was found to be reasonable and would be considered an insignificant distance for a particular scale. Each deconflicted grid node from every model in common area is then shoal-biased binned into the product model using horizontal and vertical transformations appropriate to the datum and projections of the source models and the product model.

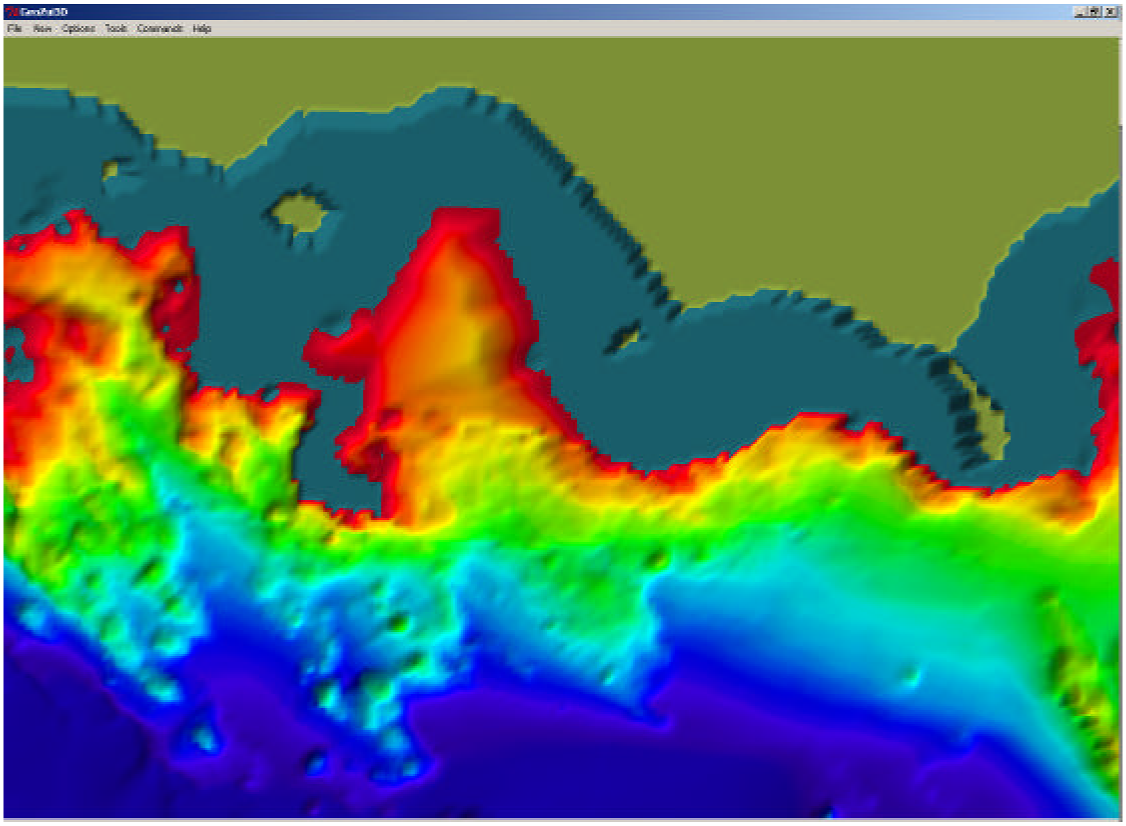


Figure 12 Merged product model before defocusing and generalization. For a navigation product, each node in the product model was produced by shoal-biased binning the closest nodes in the source models. Some details of the highest resolution models are intentionally discarded as part of the downsampling, however, shoal nodes are preserved.

<u>Defocusing for Source Horizontal Error</u>

Due to huge advances in navigation accuracy in the past 30 years, most recently with GPS and its derivative techniques, it is quite common for the average mariner to have better positioning than was used to collect the data on which the chart is based. The horizontal error for an offshore feature collected 50 years ago could be tens of meters, while current Differential GPS systems are accurate to within a few meters. The approach taken in this study is to “defocus” the base model to account for the horizontal error of the source soundings. In effect, shoal depths are spread out over the area defined by the horizontal error. [refer to figure]

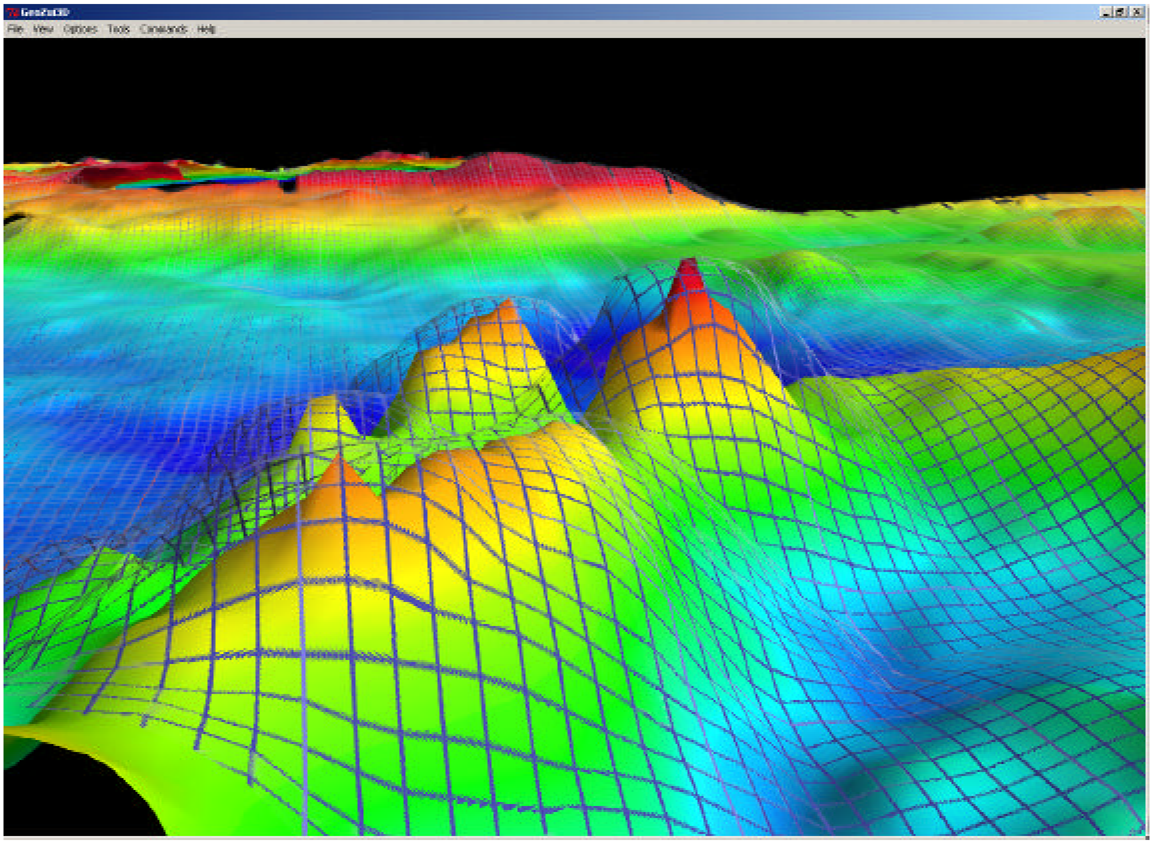


Figure 13 Product model defocused for horizontal error. The model is spread over an area proportional to the horizontal error of the source model.

The first trial algorithm used for this procedure merely spread these depths horizontally at the least depth by the radius of the horizontal error. This approach created a surface that contained many vertical walls and a very artificial appearance. Contours created from it stacked up on top of one another at these vertical surfaces and had strange arced shapes.

The second approach, pictured above, smoothes the transition at the edge of the horizontal error by creating an ellipsoid of rotation whose major axis is the horizontal error and whose minor axis is the difference between the base point (top of the shoal, for instance) and the original depth estimate at the edge of the horizontal error. New depth values are then taken from this ellipse. See Appendix for a detailed description of the algorithm. Throughout the process, shoal depths are preserved.

It is not meaningful to discuss the horizontal error of a node in a model, since the node is exactly defined in space. However, it is likely that horizontal error in the base measurements used to build the source model have been integrated into the model. While it is an imprecise estimate, a single value for horizontal error for a source model can be incorporated into the metadata of the model and used consistently across the model. Also, because the horizontal error is a property of each source model, the defocusing for horizontal error should occur only in the context of a single source model, before the individual models are combined into a product model.

Generalization to Scale

The purpose of the product, i.e. navigation for port approaches, might require further generalization of the product to reduce clutter and highlight the most significant hazards. Usually this is done by generalizing the contours. The approach examined here is to generalize the product model, then produce contours from it. The contours so generated are then appropriate for use at a particular scale. During the generalization process for a navigation product, it is critical that shoal depths are preserved. The requirement for deep channels to be preserved is a function of the scale of the product. The smallest channels are only shown on the largest scale charts.

The approach taken to the generalization process is an extension of the buffering procedure taken from two-dimensional GIS. In 2-D GIS, buffering is used to create a new line or region that lies at least a given distance from an input set of points and/or lines. In the Figure 14, the input lines are the blue ones and the buffered line is the green line, furthest out. In order to more closely fit the input lines, the buffered line is then buffered back in the direction of the original data, resulting in the red line in the diagram. Using this process, the extents of the input lines on the side of the original buffering are honored exactly but fine detail is removed. This is an automatic version of a process that is done manually today whenever a larger scale cartographic product is generalized to a smaller scale.

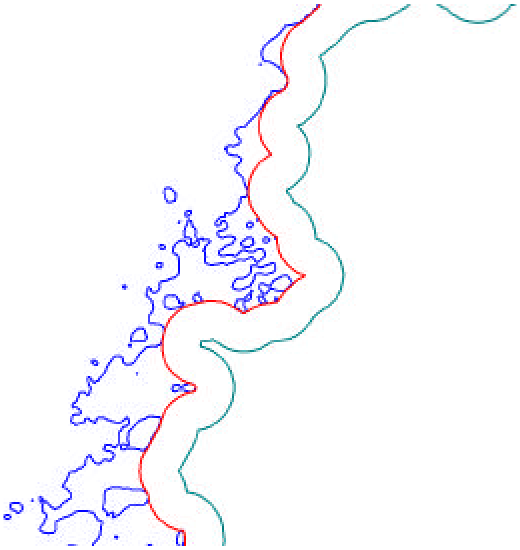


Figure 14 Two dimensional double buffering. The source blue line is buffered by a fixed radius in one direction, then the resulting line is buffered back toward the original line, creating a generalized line honoring points on the side of interest. Image used with permission of the International Hydrographic Review.

The process used in this study merely extends the same concept into three dimensions. First, the model is buffered up (in the shoal direction) by a fixed distance, the buffering radius, producing a new surface that is shoaler than the original surface by at least the buffering distance. Then the buffered model is re-buffered back down (in the deep direction) toward the original model. Each point on the original model which is exactly the buffering distance from the buffered model is exactly preserved in the transformation. These are the shoal points, exactly what we hope to preserve. Deeper points surrounded by shoaler points are not honored in the generalization because of the first buffering step. The greater the value of the buffering distance, the more the model is generalized. See the Appendix for a more detailed description of the algorithm.

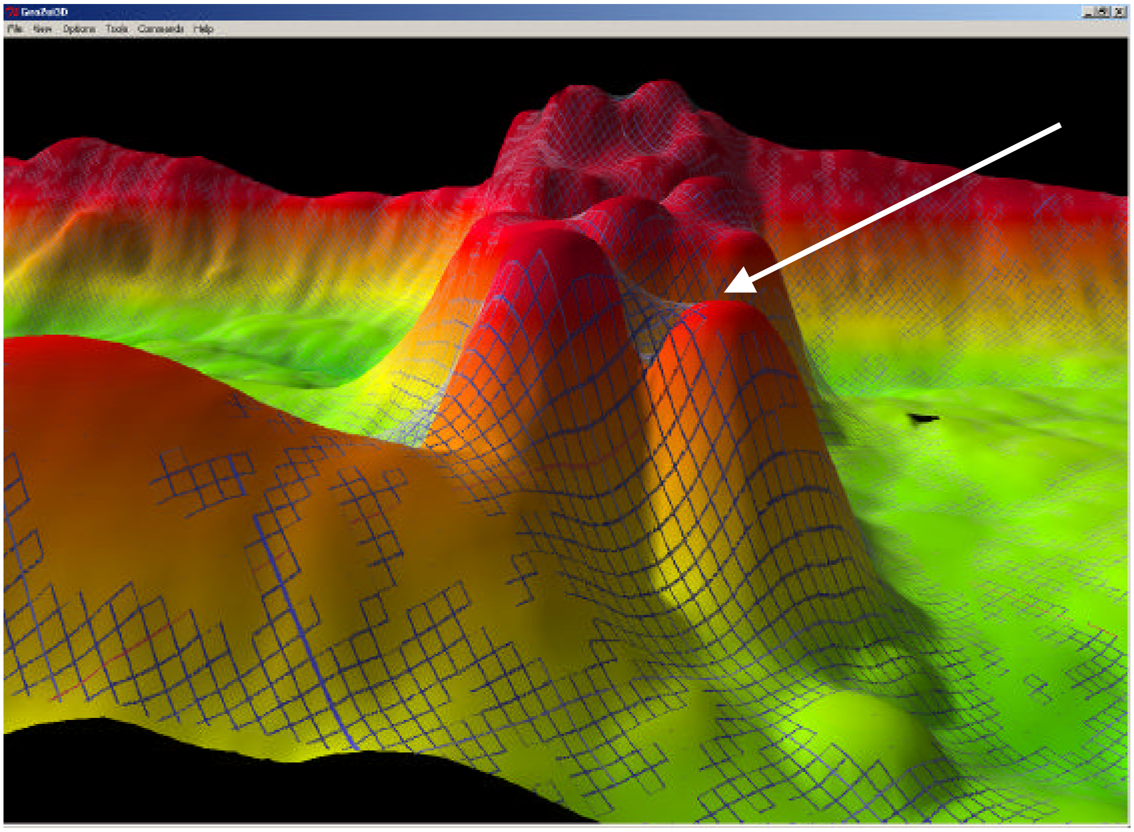


Figure 15 Generalization to Scale. In this example, two isolated peaks near each other are joined in the generalized product model.

Uncertainty of Generalized Nodes

When the depth at a node is changed as part of a generalization process, the associated uncertainty needs to change as well. However, because the uncertainty is no longer symmetrical, the meaning of the uncertainty changes too. For a navigation product, the mariner is primarily concerned with the shoal-side uncertainty, that is remaining uncertainty on the shoal side of the reported depth. As a result, the uncertainty we report should be related to the residual shoal side uncertainty after the depth has been changed. This implementation used the ratio of the area under the probability curve, assuming a Gaussian PDF, on the shoal side of the adjusted depth to the total area under the curve to scale the originally reported uncertainty.

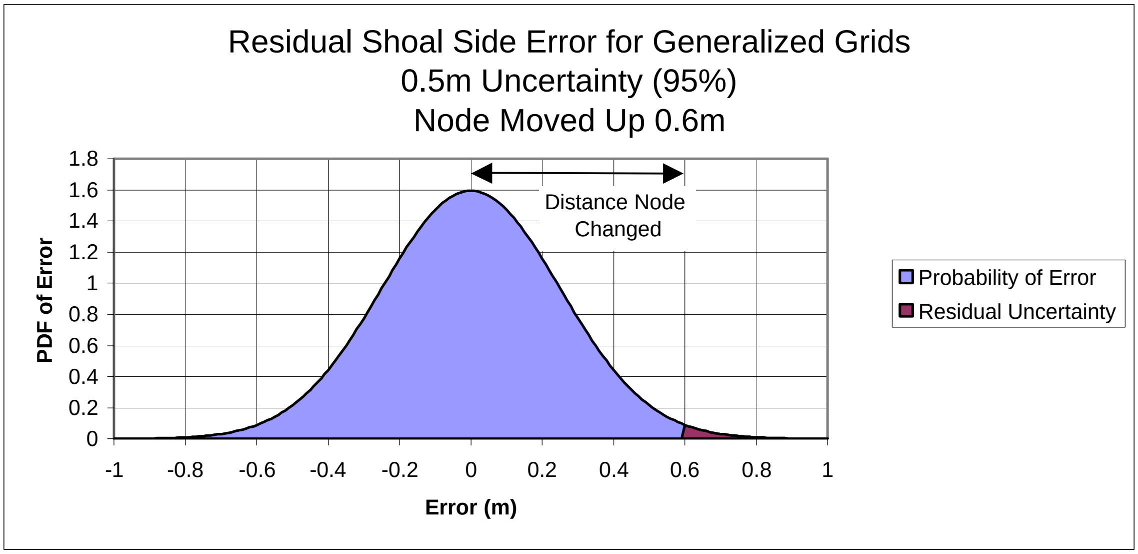


Figure 16 Residual shoal-side error for generalized grids. The residual error is scaled by the ratio of the area under the probability distribution function on the shoal side to the original total area.

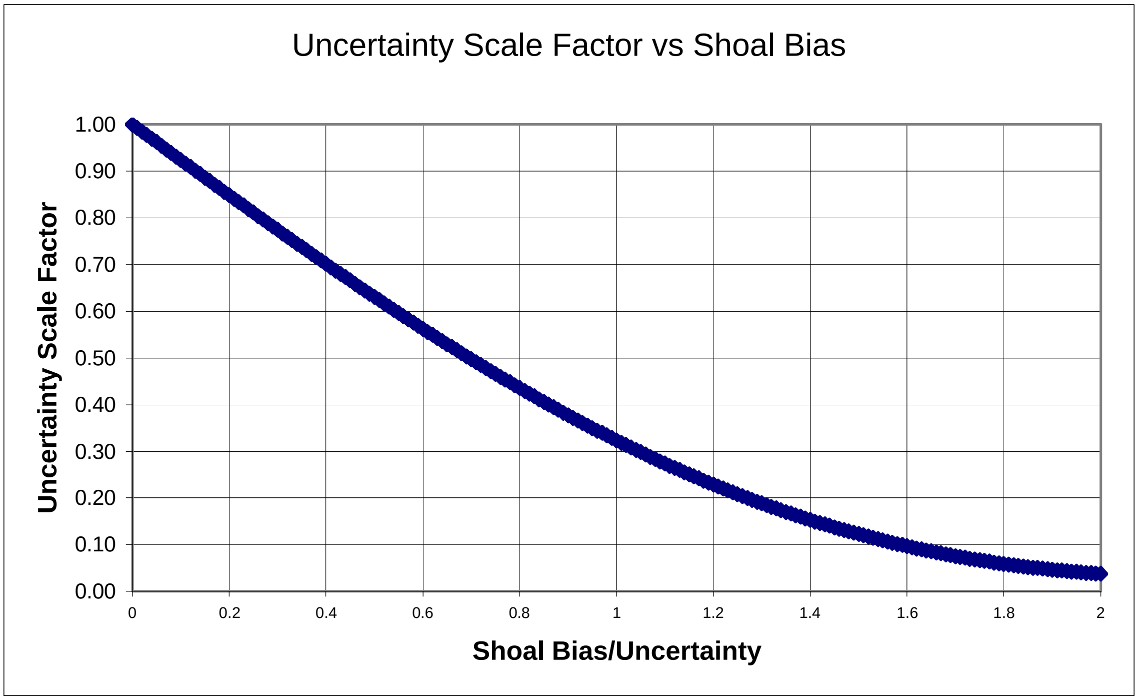


Figure 17 Uncertainty scale factor vs shoal bias. For example, if a node is moved up by its original uncertainty (x=1), the residual shoal side uncertainty is estimated at 33% of the original uncertainty.

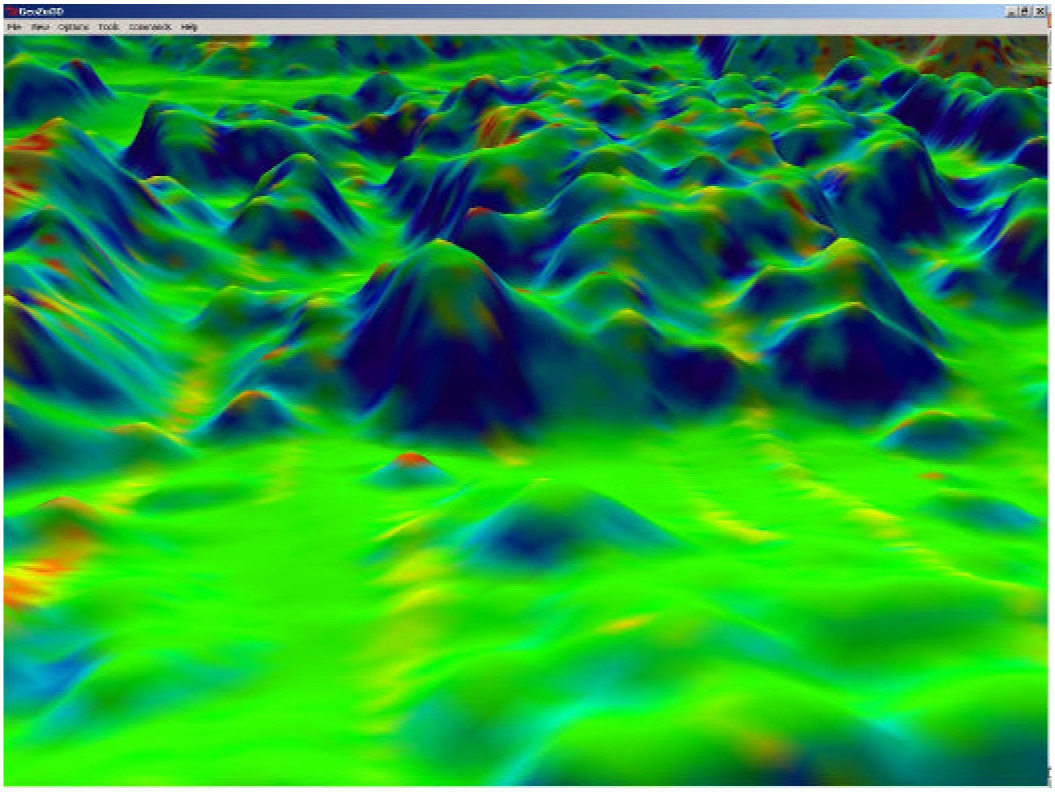


Figure 18 Uncertainty of a generalized product model. Because the generalization honors the shoal points, their uncertainty remains unchanged. Local deeps and steep slopes show low uncertainty because they have been changed the most.

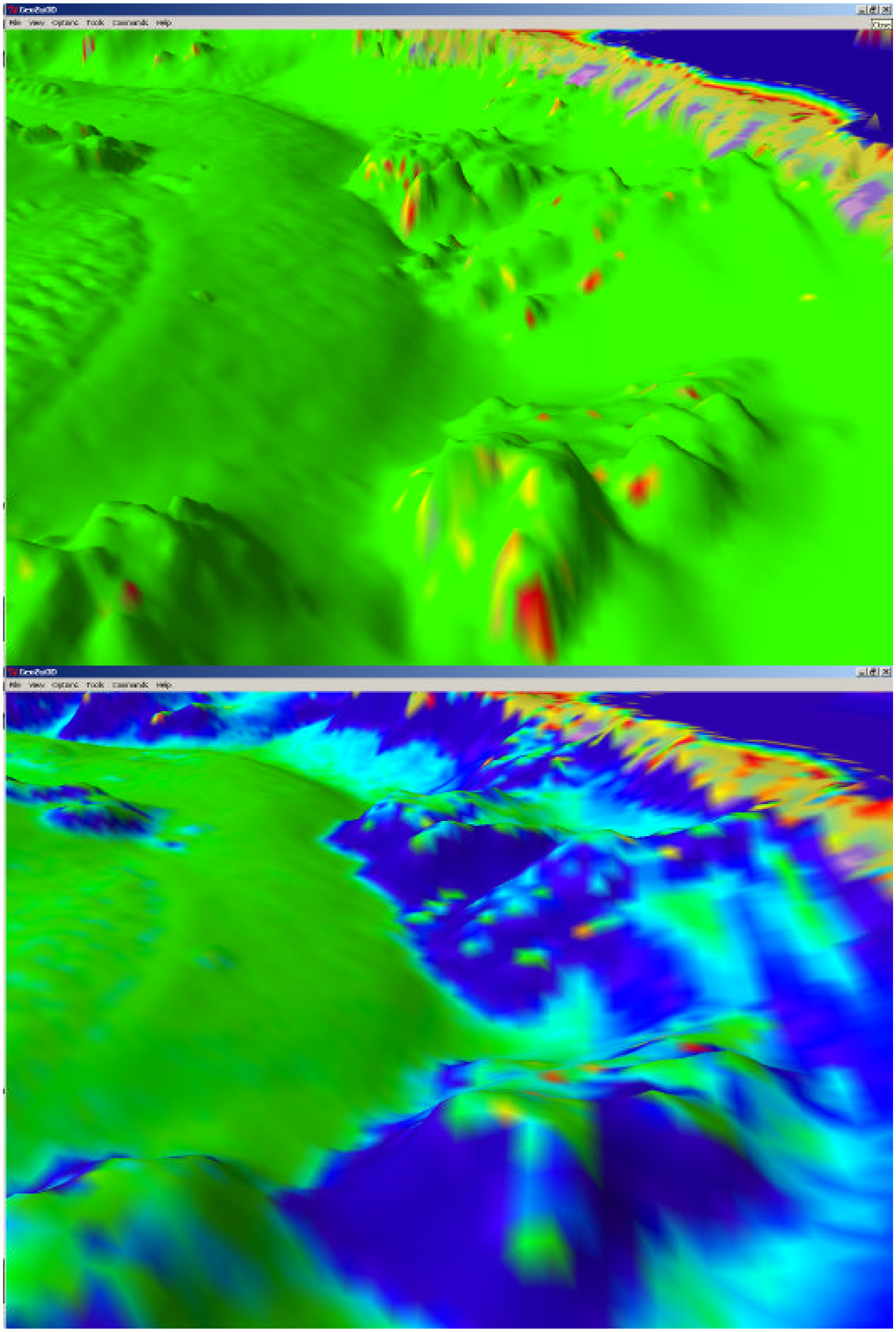


Figure 19 Before generalization (top) and after generalization (bottom). Note that the steep slopes with high uncertainty in the top figure have been generalized so that the product uncertainty is actually very low.

Automated and Semi-Automated Cartography From the Product Model

A primary goal of the Navigation Surface is the facilitation of automated cartography. Because conflicting sources of information are already reconciled and the model is already generalized to the scale required, all that remains is to transform the information contained in the model to another form, specifically depth areas, contours and selected depths. Contouring from a DTM is a well-established practice. For this project, Caris HIPS Field Sheet Editor was used for contouring.

Because the contours are created automatically without manual intervention, any number can be created at any interval, in any units at any time. This is especially important for enhanced Electronic Navigational Charts where the density of depth areas may be significantly increased in areas of low underkeel clearance *[Hudson, 2000]*. When constructing an ENC, some Hydrographic Offices (HO's) have quite complex rules for contour and depth area generation. For example, when contours get to be too close together, some HO's will simply omit some curves for clarity. When this happens, a linear depth area must be created along the boundary to compensate for the topological discrepancy *[Nautical Charting Manual]*. In another example, an HO might have a standard for the minimum area of a deep isolation before the isolation is removed for clarity. While no tools exist today, these rules and others could be applied during automated contouring.

Selected soundings are currently placed on charts for easy reference to depth, typically at the least depths of shoals, at critical inflection points in the bathymetry, on the controlling depths of natural channels, and periodically in between these features for general reference *[Nautical Charting Manual]*. The selected depths must be consistent with the depth areas in which they fall. This is good cartographic practice for paper/raster charts and is enforced for ENCs. Due to the manner in which the Navigation Surface product model was constructed, least depths are preserved. As a result, it is possible to use nodes from the product model as selected depths. There are several advantages to this approach. First, it is not necessary to preserve all possible selected soundings as a separate database that needs to be reconciled to the Navigation Surface database. Second, because the depths and the contours come from the same model, they will coincide without reconciliation, regardless of the level of generalization. Third, the choice of selected depths for the product is not limited to the set that was originally selected at a fixed scale. This makes the choice of scale of final product independent of any expectations of scale at the time the survey was collected.

The process of building a chart from high density data is currently a cumbersome process involving overlaying a rescaled smooth sheet with the proposed chart at chart scale. Soundings are created on the chart that are consistent with the smooth sheet's plotted soundings. Because these are created by hand, they must also be checked by hand, not only for interpretive issues involved with chart creation, but also for transcription errors. A significant short-term downstream advantage of the Navigation Surface approach is the ability to create a better set of tools with which to create the ENC and paper charts.

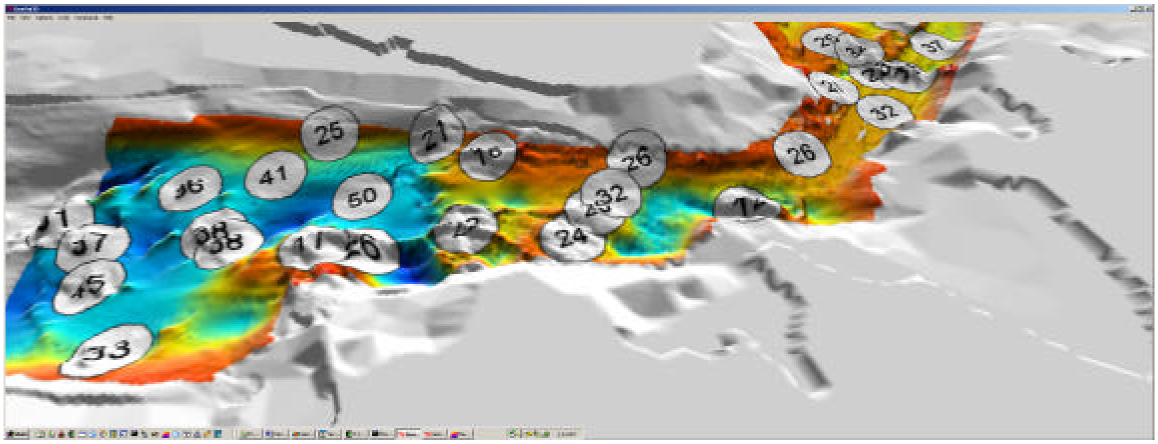


Figure 20 Mock up of the sort of 3D chart constrution environment possible from a Navigation Surface product model.

The creation of this environment is beyond the scope of this project, but could include the following components:

- display of the product model including uncertainty
- icons of "potential" selected depths chosen automatically from the product model
- the ability to select depths from the potential set
- a visual cue of the spacing of selected depths
- depth curves
- other features (rocks, wrecks, obstructions, etc)

Not only would the initial construction of the chart be more efficient, but quality control should be significantly easier, and a production path is established for creation of a new generation of charts.

SUMMARY

The Navigation Surface database is made up of a collection of source models. Each node of each source model is attributed with depth and depth uncertainty. The node spacing is commensurate with the detail available in the source data. There is an opportunity for the hydrographer to manually designate least depths if the model were deemed unrepresentative of a particular feature, assuring items critical to navigation are preserved.

There are several advantages to the Navigation Surface process over current practice. First, the full resolution of the survey is preserved for navigation and non-navigation uses. Second, the use of a manipulable model permits automated cartography. Third, the inherent uncertainty model provides a framework for continuous quality control. Fourth, by changing the framework of the data pipeline to a model based system, CUBE and other innovative advances in ocean mapping may be brought to bear for the hydrographic community.

CHAPTER III

TESTBED PROJECT

Main Objective

The tools described in the previous chapter were developed with reference to only a few datasets. In order to show the broader validity of the Navigation Surface Database, a project was undertaken under more diverse circumstances. The goal was to produce a validated, non-conflicting model-based bathymetric database for a significant geographic area, including uncertainty estimates for every model node. In addition, S-57 cartographic objects appropriate for use at different scales were to be included in a hydrographic product database for use in an ENC and paper chart. For this project, only digitally available data was considered. On the actual chart of the area, there may be data from sources not yet available digitally.

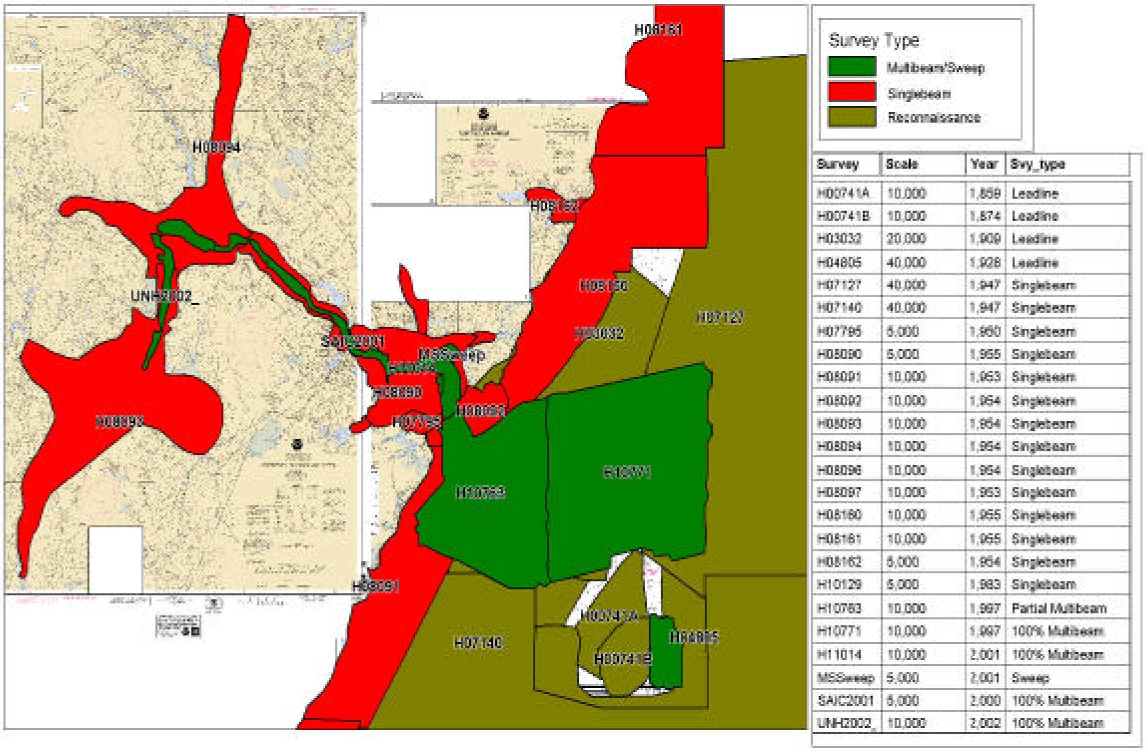
Process

The process followed for the project was as follows:

- Choose a manageable geographic area that contains a variety of source data, varied seafloor type, and a wide depth range.
- Assemble all bathymetric sources for the area at their highest resolution.
- Create Navigation Surface source models from each survey.
- Reconcile conflicting models using clear rules.
- Assemble a single model of the product area.
- Generalize the model to product scale.
- Extract cartographic objects from the generalized product model.

Study Area

The Great Bay estuary and approaches, including Portsmouth, NH, was chosen for its wide range of seafloor morphology and type, including steep slopes, sand areas, rocky outcrops, mud flats, narrow channels and dredged areas. In addition, there is a wide variety of source data, including modern high-resolution multibeam of several types, sweep sonar systems, single beam and leadline surveys. There are also areas that have no digital source data available. In this section, NOS survey registry numbers *[HXXXXX]* are used as a reference to a particular survey.



Survey	Scale	Year	Svy_type
H00741A	10,000	1,859	Leadline
H00741B	10,000	1,874	Leadline
H03032	20,000	1,909	Leadline
H04805	40,000	1,928	Leadline
H07127	40,000	1,947	Singlebeam
H07140	40,000	1,947	Singlebeam
H07795	5,000	1,950	Singlebeam
H08090	5,000	1,955	Singlebeam
H08091	10,000	1,953	Singlebeam
H08092	10,000	1,954	Singlebeam
H08093	10,000	1,954	Singlebeam
H08094	10,000	1,954	Singlebeam
H08096	10,000	1,954	Singlebeam
H08097	10,000	1,953	Singlebeam
H08160	10,000	1,955	Singlebeam
H08161	10,000	1,955	Singlebeam
H08162	5,000	1,954	Singlebeam
H10129	5,000	1,983	Singlebeam
H10763	10,000	1,997	Partial Multibeam
H10771	10,000	1,997	100% Multibeam
H11014	10,000	2,001	100% Multibeam
MSSweep	5,000	2,001	Sweep
SAIC2001	5,000	2,000	100% Multibeam
UNH2002_	10,000	2,002	100% Multibeam

Figure 21 Survey index of surveys used in the testbed project.

Assembling the Sources

The surveys used for the testbed project are shown in Figure 21. The highest resolution validated archive of the historical singlebeam and leadline data is kept by NOAA's National Geophysical Data Center (NGDC), and is accessible through a set of CDs issued by NGDC *[GeoDas-NOS Hydrographic Survey Data]*. Each of these surveys is preserved at the resolution of the smooth sheet originally used to depict the survey. For example, at 1:10,000, there is a sounding approximately every 50m. For the testbed, data for each survey were extracted from the CD and managed separately. Each survey was inspected for blunders by gridding them and looking for outliers. The most common problem found was an incorrect hundreds place at the hundred foot crossover on surveys digitized from paper source. For example, a 198 ft sounding surrounded by 202, 201, 206, would be digitized as 298. For these obvious blunders, the problem was corrected. In other less obvious cases, the sounding would simply be removed from consideration. All of these problems occurred in deep water, greater than 100ft In total, only a dozen or so soundings were suspect in any way.

The smooth sheet data for *H10771* (Reson 9003) was also used, even though it was created from a modern multibeam survey. This was done for two reasons. First, the full density multibeam data was never tide corrected and is stored in a fairly inaccessible form. Second, it is useful to examine how the downsampling affects the ability to create a Navigation Surface source model from the data.

The full multibeam data from *H10763* (Reson 9003) was recovered from archive at the NOAA's Atlantic Hydrographic Branch and tides were reapplied. It had already been cleaned by the NOAA ship RUDE, and no changes were made to this cleaning. When the data were cleaned originally, it was assumed that any derivative product would be made from shoal biased soundings. As a result, there are portions of the survey where deep outliers were not systematically flagged for rejection.

The multibeam data from *H11014* (Reson 8101) was provided by NOAA in HDCS (HIPS) format as part of the common data set for the Shallow Survey 2001 conference. The multibeam data from SAIC (Reson 8125) was provided in XTF (extended Triton Format) format as part of the common data set, and it was converted into HDCS (HIPS) format. The multibeam (Reson 9001) data from the UNH summer hydrographic field class was originally processed in HIPS, so it needed no conversion or further processing. The sweep data from Mirimichi Surveyor (also part of the common data set) was converted to HIPS format using a custom convertor, then cleaned and processed in HIPS.

Creating the Navigation Surface Source Models

The *single beam and leadline surveys* were processed as follows

- The soundings were imported into MapInfo (Mapinfo Corporation, Troy, NY) and reprojected to UTM zone 19.
- The soundings were overlain on the most recent version of chart, and any soundings that plotted on shore were removed. This happened in only a few places where clear evidence was present on the chart for manmade construction. In one case, an area of Seavey Island was filled as part of

Portsmouth Naval Shipyard expansion. In another, the area around the current state pier was filled as part of the construction. Soundings not in direct conflict with the charted shoreline were untouched.

- Each sounding was attributed with an estimated uncertainty. It was beyond the scope of this project to analyze the methods used to conduct the survey in any detail. A simple estimate of 0.3m plus 3% of depth was used. This estimate is based on an estimate for waterlevel and draft error of 0.3m and a depth measurement accuracy of 3%. A horizontal error of 20m was used for all singlebeam and leadline surveys. Using traditional positioning methods, 20m was achievable inshore and more difficult offshore. Better estimates of both horizontal and vertical errors could be made through a detailed examination of sounding and positioning methods *[Jakobsson, 2002]*, but this was beyond the scope of this project.

- Points from the local shoreline were merged with the soundings to provide a boundary condition and a smooth interpolation to the shore. High water lines were attributed with an elevation of 4m (approximate tidal range) and an uncertainty value of 1m. Low water lines were attributed with an elevation of 0 and an uncertainty of 1m. Intertidal rocks with no height given were given elevation 2m (half the tidal range) and an uncertainty of 2m (to encompass the entire tidal range).

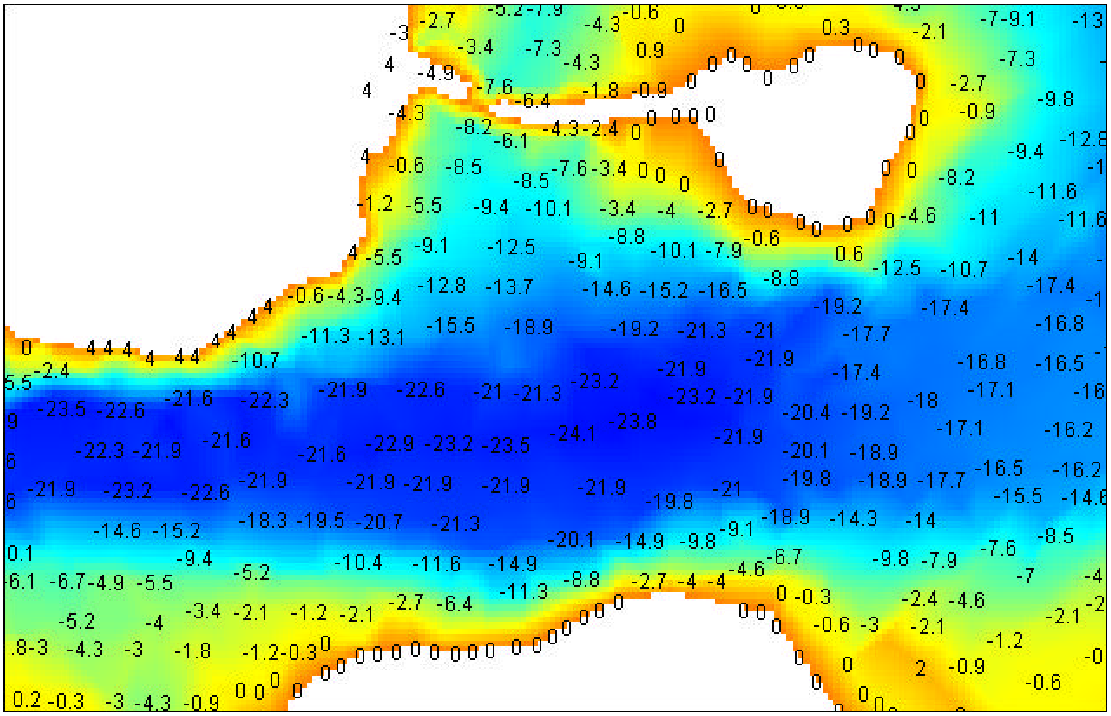


Figure 22 Sounding file merged with points extracted from the shoreline to enable bank-to-bank modeling.

- The merged points were exported as easting, northing, elevation, and uncertainty files.
- The points were TINed in MapInfo using Vertical Mapper, using a max TIN side length of twice the maximum line spacing. The TIN was resampled at a resolution appropriate to the scale of the survey, approximately 0.3-0.5mm at the scale of the survey. The resampled TIN was exported as an ArcView ascii grid.
- Using gzt.exe, the toolbox built for this project, the node closest to each sounding and shoreline point was reset to the exact value of the sounding, which may have been changed due to the resampling of the data in the modeling process.
- Using gzt, an uncertainty for each node in the depth model was estimated using the procedure outlined in Chapter 2.

- The combination model (depth and uncertainty) is then written as a GUTM (Grid, UTM), a GeoZui3D *[Ware, 2001]* grid format used for development and visualization purposes at UNH. This model represents the Navigation Surface source model.

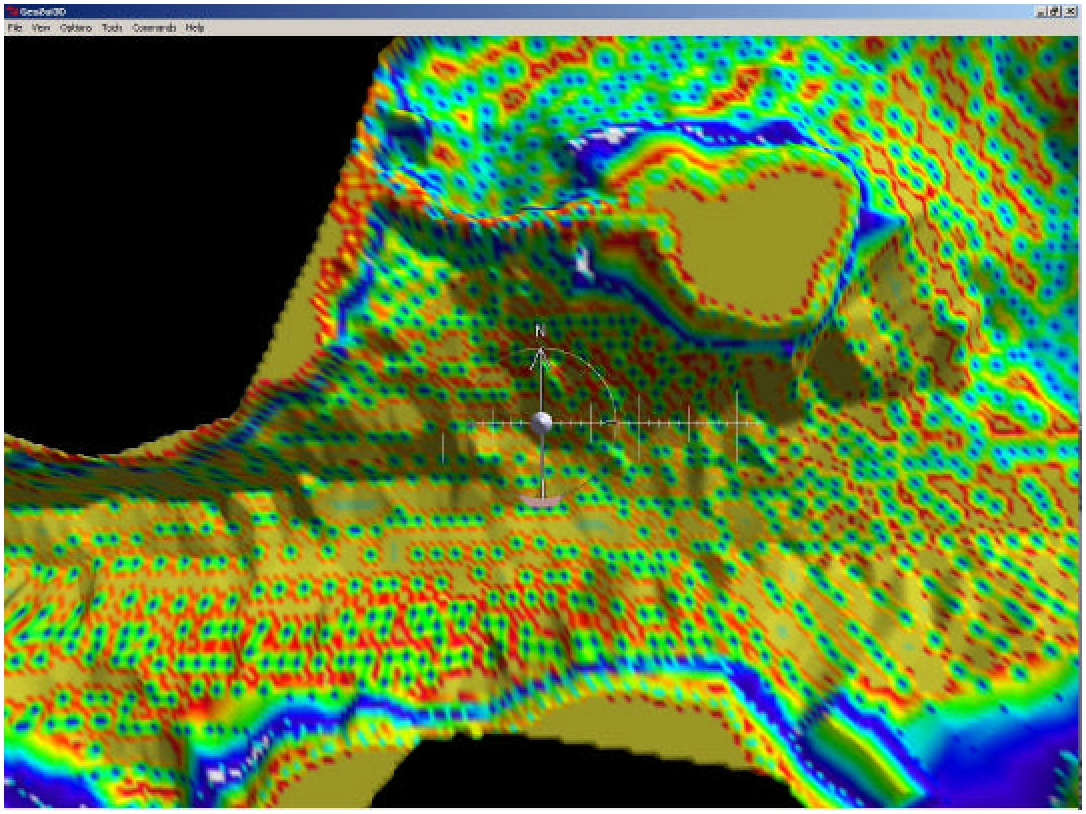


Figure 23 Uncertainty of the source model derived from singlebeam and shoreline information

For *multibeam and sweep systems,* the following procedure was used:

- Using HIPS 5.3, a weighted grid was created as described in Chapter 2. The resolution was chosen to approximate the nadir footprint for the bulk of the survey, since the nadir footprint is the smallest of all data collected.
- Any node which had a missing neighbor was exported to a plain text file. The nodes were TINned and resampled at the resolution of the original model. This model was then used to fill any gaps in the multibeam data.
- The survey was manually examined for significant features not well represented by the weighted grid. For each case, a least depth on the feature was chosen from among the measurements. This

sounding was designated as “outstanding” in the HIPS data structure, and represent the golden soundings chosen by the hydrographer to be honored.

- Using gzt, the HIPS weighted grid was compared to each sounding in a bin around the node, and two values were computed. First, the simple mean of the estimated uncertainties of the soundings was calculated. This is not strictly rigorous, since the same weighting scheme should be used to mean the uncertainties as was used for the depths. Second, the standard deviation was computed from the same binned soundings. This standard deviation was scaled by 1.96 to estimate the 95% threshold. These two values were compared and the greater retained. For the nodes interpolated in step 2, the uncertainty is estimated using the sparse data uncertainty algorithm.
- For each “outstanding” sounding encountered in the line data, the closest node is adjusted to the exact depth and uncertainty of the sounding. This ensures that all golden soundings are honored.
- The combined depth and uncertainty model is then saved as a GUTM file.

The data from Reson 8125 data collected by SAIC was processed by Dr. Brian Calder into a combination GUTM as described in Chapter 2.

The charted shoreline, down to the low water line, was gridded and treated just like a survey for the purposes of this project. It was digitized from the raster charts, assigned depth values, uncertainty was estimated using the sparse data process, and it was saved as a combination grid.

Reconcile Common Areas

Once all the source models were ready to be placed in a database, they were deconflicted. The criteria used was simply the age of the survey with newer surveys superseding the older in common area. The superseded version of each survey was stored separately, so that the collection of surveys represented the deconflicted database. Each survey still retained its original resolution. More subtle rules might be used by comparing uncertainties or retaining portions of an older survey, but there was no need in this case, since each generation of survey was significantly better than the last.

Assemble a Single Product Model

A new product was defined that was intended to be a 1:20,000 scale navigation chart using Universal Transverse Mercator Zone 19. It covers the entire estuary and the approaches as far as the Isles of Shoals. The area is shown in Figure 24. The resolution was chosen as 5m for the base product model, which is an insignificant distance for the purpose of the product. Once the product was defined, each model was sampled into it in turn, retaining the shoalest depth at each node. The movement of nodes associated with this resampling is insignificant, since the node spacing was chosen to be an insignificant distance.

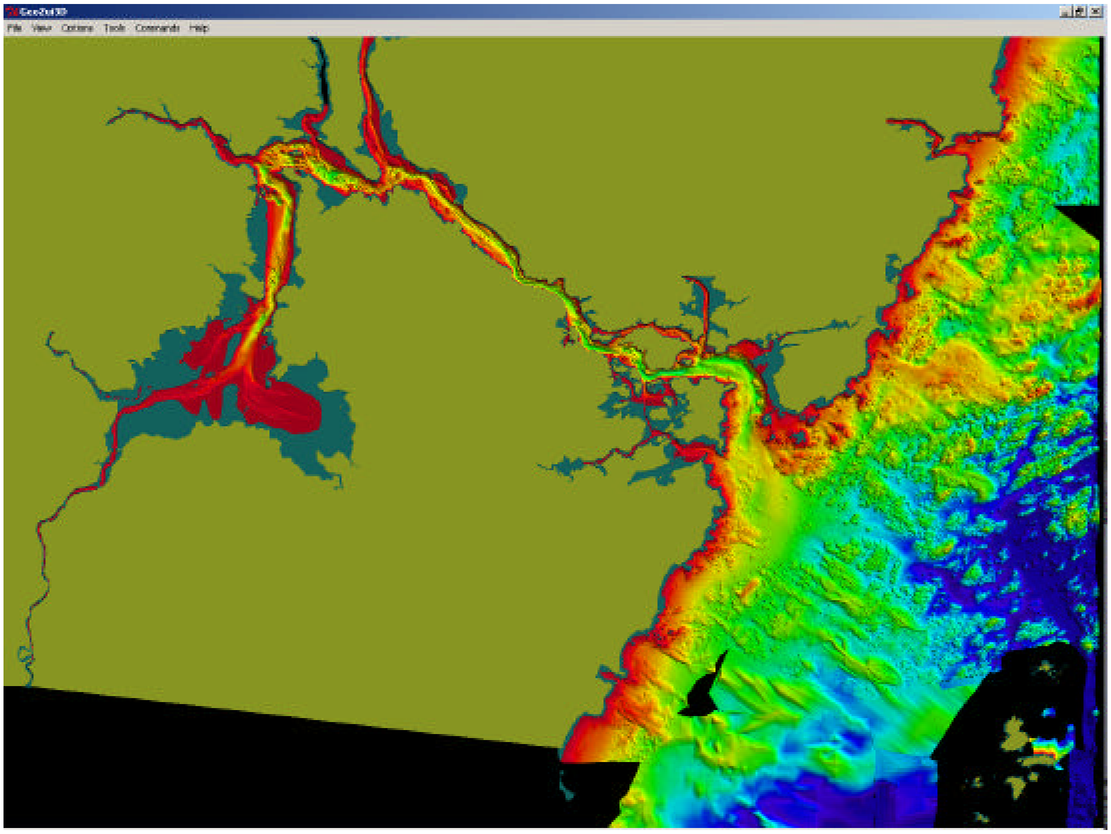


Figure 24 Product model as a combination of all the source models.

Generalize the Product Model to Scale

While the product model is already significantly generalized with respect to the highest resolution source models, further generalization is necessary to simplify the seafloor for clear navigation use. The script used to create the grids is included in the Appendix. The model was downsampled to 10m and double-buffered to a 200m radius. The source model and the final generalized product model are shown in Figure 26. Note that the peaks match in depth, but the peak has been moved within one product model node spacing, and the shape has been significantly generalized. The total movement associated with a shoal sounding through the whole process is half of the source model spacing (chosen to be less than the horizontal error), plus half the product model spacing, chosen to be insignificant at the scale of the product.

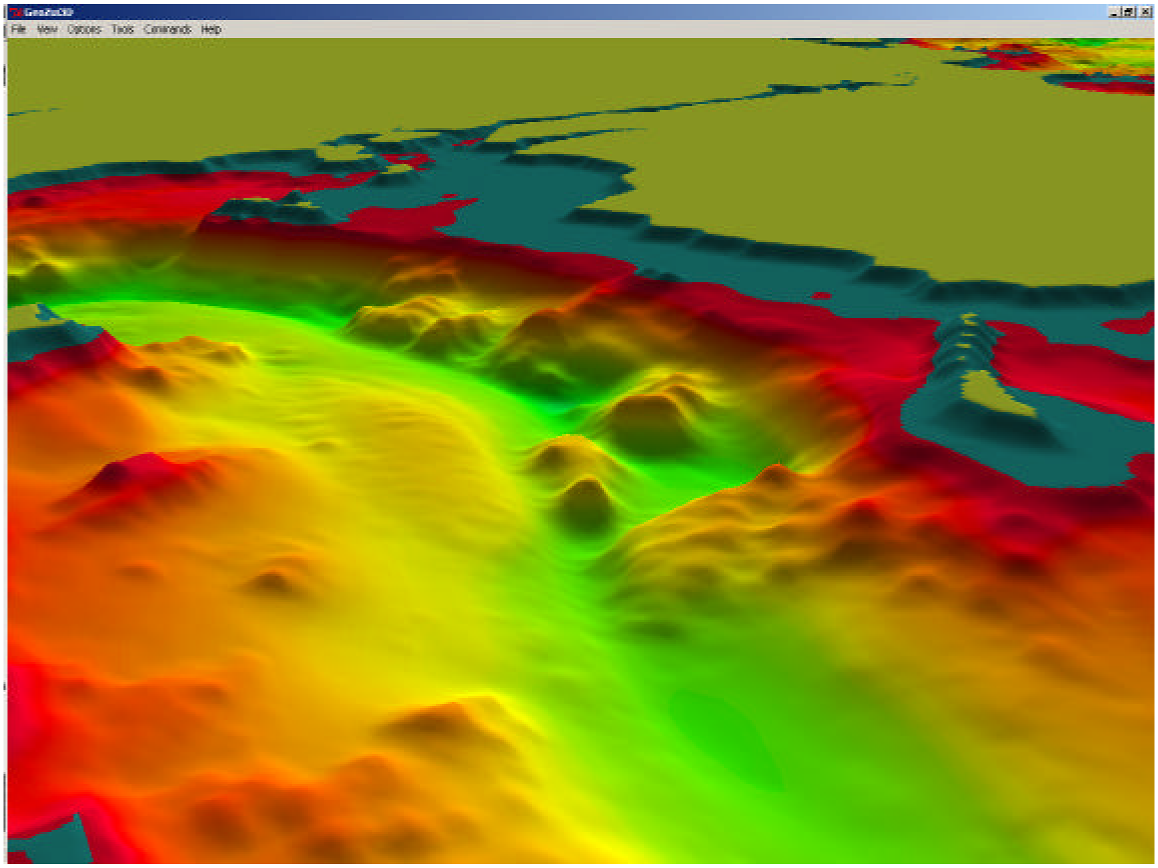


Figure 25 Generalized product model.

Extract Cartographic Objects from Model

Once the product model is complete, cartographic objects can be created based on the model. Contours created based on a generalized model are themselves generalized. For the contouring used in this project, Caris Fieldsheet Editor was used. The generalized model was written into the HIPS weighted grid format using gzt.exe, then the native contouring algorithm was used. One advantage is that the contours created have a direct route into S-57 format.

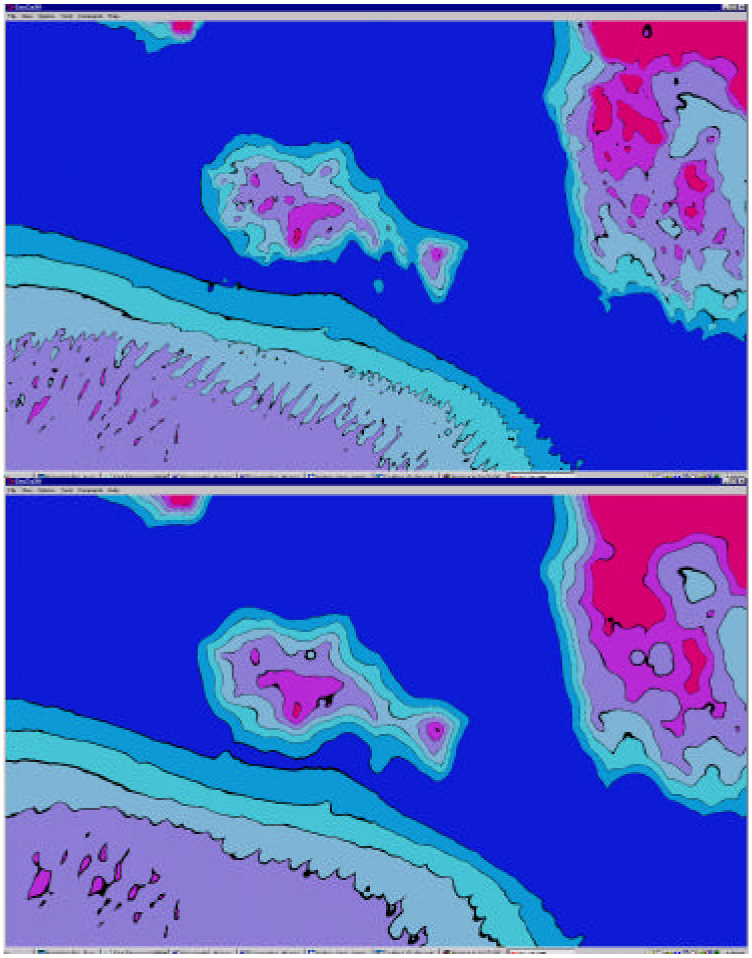


Figure 26 Effect of model generalization on cartographic objects.   The top model is ungeneralized. The bottom model is generalized.

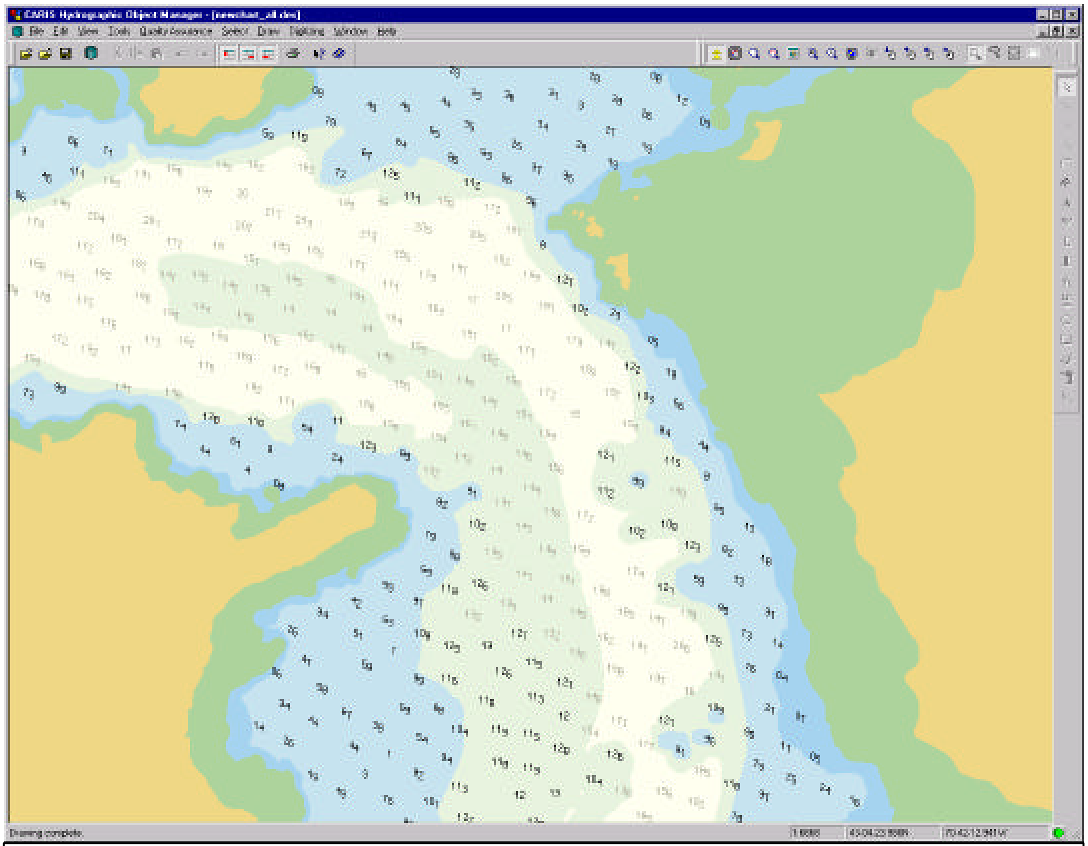


Figure 27 Cartographic objects created from a generalized product model.

Summary

Using the methods described in Chapter 2, a collection of source models was created to simulate a database. The models were deconflicted against one another, and a chart-scale product model was produced from them. The product model was generalized to preserve the appropriate level of detail for the chart, and S-57 format contours and selected depths were created from the product model.

CHAPTER IV

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

The Navigation Surface database has considerable potential to improve both the speed and the objectivity of the charting process as well as providing a means for hydrographic offices to produce products which are suitable for meeting other bathymetric requirements. In particular, by replacing the soundings and contours on a smooth sheet with a high-resolution bathymetric model as described herein, it is possible to produce both the high accuracy bathymetry necessary for safe navigation and the high-resolution internally-consistent bathymetry necessary for marine geology, habitat characterization, and marine modeling. In addition, the uncertainty estimate inherent in the model provides quality control for the survey, and allows the hydrographer to prioritize further work.

A database of source models could provide the source from which the bathymetric portions of nautical charting products are drawn. The cartographic processes required for production of today's paper and raster charts and vector-based Electronic Navigational Charts could be streamlined by generalizing the model to support the navigation purpose of the chart before cartographic features are derived from it. Additional navigation products, such as a chart with a product model embedded into it, could be created in addition to the current suite of charts. Because the database contains both depth and uncertainty, it could be used to assess the adequacy of data underlying the chart with respect to its current use.

Unresolved Problems-Future Research

Refine uncertainty in interpolated areas—While the approach taken appears to be valid on the data presented here, it is necessary to continue testing the uncertainty estimation on different areas before applying it globally. The model has not been validated on very sparse data of greater than 100m line spacing. For modern singlebeam or incomplete coverage multibeam, kriging needs more investigation with more powerful variogram manipulation tools.

Explore the product implications for uncertainty—Once the capability exists to create a product model which contains uncertainty as well as depth, there is an opportunity to present the mariner with uncertainty information in a variety of forms. Existing forms today are source diagrams, zones of confidence (CATZOC), and attibution of individual cartographic items with error estimates. Each have their strengths and limitations. More research should be done on how to get uncertainty information to the mariner in a form best suited to decision support. There might be possibility to tie into risk management tools. A future chart that includes continuous bathymetry could also include continuous uncertainty.

Refine transformations of shoreline and features to model space—Can we define what a rock means at a particular scale? Is there a two-way transformation? If so, we could potentially further automate feature creation and attribution.

Time-based uncertainty growth—There is considerable potential for investigation of this idea, outlined but not pursued in this paper. Research is needed into the form of the rate of uncertainty growth functions, and how to tune them to different seafloor types, different erosion/deposition regimes, and a variety of one-time events (earthquakes and hurricanes, for example). This effort is likely to require an interdisciplinary approach.

Application to LIDAR—LIDAR systems are already being used with DTMs and contouring them for particular purposes, like shoreline identification. Once an error model is established, a blended database of models can be created which spans the intertidal region.

Tie to VDATUM—The navigation surface database, consisting of models in their original datums, requires transformation tools to combine models as the product demands. Any implementation should consider VDatum *[Parker et al, 2003]*-enabled transformations as a critical link.

Build Auto-ENC tools—These may already be available in tool packages from the commercial sector, but there is a need for them in the cartographic process and also potentially in real time if the product model becomes accepted for use as part of a chart.

Build 3D tools for traditional chart compilation—One of the most time-consuming and subjective parts of the chart building process is to pick selected soundings. This new technology raises the possibility of creating a 3D application to interactively build a chart from a product model.

REFERENCES

Brissette, M., J.E.Hughes-Clarke, D. Cartwright, Object Detection Using Multibeam Echosounder Temporal Imagery. U.S. Hydrographic Conference, 2001

Calder, B., L. Mayer, Automatic Processing of High Density, High-Rate Multibeam Echosounder Data. *G3,* DID 2002GC000486 (in press)

Calder, B. Robustness in Automatic Processing of Multi-Beam Echo-Sounder Bathymetry, Shallow Survey 2001, Portsmouth NH, available at http://ccom.unh.edu/shallow/abstracts/robustness_automatic.htm

Calder, B., S. Smith, A Comparison of the Automated Navigation Surface to Traditional Smooth Sheet Compilation, Canadian Hydrographic Conference 2002, Toronto, ON.

Calder, B. Automatic Statistical Processing of Multibeam Echosounder Data, *International Hydrographic Review* 4 (1), April 2003.

Calder, B., S. Smith, A Time/Effort Comparison of Automatic and Manual Bathymetric Processing in Real-Time Mode, US Hydrographic Conference, 2003.

Caris Inc, HIPS 5.3 User's Guide, 2003, Fredericton, NB.

Coast Survey, *General Instructions Hydrographic Surveys Division in regard to Inshore Hydrographic Work of the Coast Survey, 1878.*

Coast and Geodetic Survey, *Hydrographic Manual*, 1942 p. 721.

Cutter, G. R., Rzhanov, Y., Mayer, L. A., 2003. Automated segmentation of seafloor bathymetry from multibeam echosounder data using local Fourier histogram texture features. *Journal of Experimental Marine Biology and Ecology*, 285/286, 355-370.

HXXXXX, *Descriptive Reports for Hydrographic Surveys*, available from Hydrographic Surveys Division, NOAA, 1315 East-West Highway, Silver Spring, MD 20910.

Hare, R., A. Godin and L. A. Mayer, Accuracy Estimation of Canadian Swath (Multi-Beam) and Sweep (Multi-transducer) Sounding Systems, *Tech. Rep.*, Canadian Hydrographic Service, 1995.

Hudson, M., 2000, Electronic Navigational Charts from Survey Source Information-The Australian Experience, *International Hydrographic Review*, Monaco, Vol 1, No. 2. pp 13-23

Hughes-Clarke, J. E. Guide to the Creation and Interpretation of the Comparative Data Products, US/Canada Hydrographic Commission Coastal Multibeam Surveying Course, 1997. Available at http://www.omg.unb.ca/~jhc/uschc97/comp_interp.html

International Hydrographic Organization. *Special Publication No. 44*, IHO Standards for Hydrographic Surveys. 4th edition, April 1998, Monaco.

International Hydrographic Organization. Transfer Standard for Digital Hydrographic Data, *Special Publication S-57*, Edition 3.0, November 1996.

Jakobsson, M., B. Calder, and L. Mayer, On the effect of random errors in gridded bathymetric compilations, *J. Geophys. Res.*, 107(B12), 2358, doi: 10.1029/2001JB000616, 2002.

Kielland, P. and M. Dagbert, Third Generation Electronic Charts: What They Provide Users and How to Produce Them, *Proc. Canadian Hydrographic Conference*, 1996.

Office of Coast Survey, *Nautical Chart Manual*, Seventh Edition (1992), updated to 2003.

Office of Coast Survey, *NOS Hydrographic Surveys Specifications and Deliverables*, June 2000, Silver Spring, MD.

Mayer, L., B. Calder, J. Schmidt, and C. Malzone. Providing the Third Dimension: High-resolution Multibeam Sonar as a Tool for Archeological Investigations - an Example from the D-Day Beaches of Normandy, US Hydrographic Conference, 2003.

Mayer, L. A., R. Raymond, G.Glang, L. Huff. Resolving the Ripples (and a Mine): High-Resolution Multibeam Survey of Martha's Vineyard ONR Mine Burial Program Field Area, *Eos Trans. AGU*, 83(47), Fall Meet. Suppl., Abstract OS61A-0183, 2002

Parker, B., D. Milbert, K. Hess, S. Gill, National VDatum - The Implementation of a National Vertical Datum Transformation Database, US Hydrographic Conference (on CD), 2003.

Pickrell, A., Representation of Hydrographic Surveys and Ocean Bottom Topography by Analytical Models, *Master's Thesis*, Naval Postgraduate School, 1979.

Royan Australian Navy, Internal Specification for Project SEA1430, 2002. Available at http://www.hydro.gov.au/news/sea1430/sea1430.htm

Smith, S. The Navigation Surface: A New Approach to Multiple Product Creation from Hydrographic Surveys. Shallow Survey 2001, Portsmouth, NH available at http://ccom.unh.edu/shallow/abstracts/the_navigation.htm

ten Brink, U.S., and Smith, S., High-resolution Bathymetric Map of the Puerto Rico Trench: Implications for Earthquake and Tsunami Hazards, *Seismological Research Letters*, v. 74, p.342, 2003

Umbach, M. J., *Hydrographic Manual*, Fourth Edition, U.S. Department of Commerce, National Oceanic and Atmospheric Administration, July 1976. pp 6-8.

Ware, C., M. Plumlee, R. Arsenault, L. A. Mayer, S. Smith, GeoZui3D: Data Fusion for Interpreting Oceanographic Data. *Oceans 2001 (CD)*, 2001.

APPENDIX

APPENDIX A

SELECTED SOURCE CODE

Compute Uncertainty

```
int GutmManip::ComputeUncertainty (char * fn, double hor_error, double maxuncertainty)
{
#ifdef WIN32
		FILE * infile;
		FILE * erfile;
		FILE * histfile;
		FILE * histgrid;
		FILE * fliers;
		FILE * areaxyz;
    double x,y,z,u;
		int i,j;
		int cx,cy;
		double base_uncertainty=0.0001;
		double minimum_uncertainty=0.2;
		double minimum_depth_coeff=.02;
		double tmp_backward_error, tmp_forward_error;
		double sumz, sumsquares, tempdist, mindist, growthrate, mingrowthrate, expgrowthrate;;
		float ** sumuncert;
		int l,k,numcells;
		char firstchars [100];
		char tempstring [200];
		int donefile;
		int tempvar;
		int cellsearch;
		mingrowthrate=0.0;
		char project [100];
		char vessel [100];
		char day [100];
		char line [100];
		long ret;

		//Variables for use in the Processed Depth IO code
		HDCS_ProcessedDepths	processedDepthsSensor;
		char	sourceFileName[100];
		unsigned int		toolType;
		unsigned int		numDepths, startDepth;
		unsigned int		year, dayOfYear;
		unsigned int		numLineSegments;
		unsigned int		numProfiles;
		double			seconds, time;
		double			transducerPitch, transducerRoll;
		double			latitude, longitude;
		double			minLat, maxLat, minLong, maxLong;
		double			alongTrack, acrossTrack;
		double			depth, depthAccuracy;
		double			gyro, heave, pitch, roll, tide, vesselVel;
		double			minDepth, maxDepth;
		double			minTime, maxTime;
		unsigned int		coordinateType;
		long					rcode;
		unsigned int		status, summaryStatus, profileStatus;
		double uncert_cap;
```

```
        //Variables for use in the projection code
        Projection proj;
        f64     lat64, lon64, east, north;
        double cmer, clat;
        unsigned int utm_zone;


        // Open files and allocate arrays
        if (maxuncertainty==0){        maxuncertainty=100.0;}
        maxuncertainty=100.0;

        cerr << "Starting CU\n";
        cerr << "Horiz Error=" << hor_error << "\n";
        cerr << "MaxUncertainty=" << maxuncertainty << ".\n";
        uncertainty = new float*[ycount];
        if ( uncertainty == NULL)
        {
                cerr << "Can't allocate uncertainty array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
        {
                uncertainty[i] = new float[xcount];
                if ( uncertainty[i] == NULL)
                {
                        cerr << "Can't allocate uncertainty array! Buy more ram!\n";
                        return 1;
                }
                for (j = 0; j < xcount; j++)
                {
                        uncertainty[i][j]=0.0;
                }
        }
        numpoints = new int*[ycount];
        if ( numpoints == NULL)
        {
                cerr << "Can't allocate numpoints array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
        {
                numpoints[i] = new int[xcount];
                if ( numpoints[i] == NULL)
                {
                        cerr << "Can't allocate numpoints array! Buy more ram!\n";
                        return 1;
                }
    for (j = 0; j < xcount; j++)
    {
      numpoints[i][j]=0;
    }
        }
        sumuncert = new float*[ycount];
        if ( sumuncert == NULL)
        {
                cerr << "Can't allocate sumuncert array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
```

```
    {
        sumuncert[i] = new float[xcount];
        if ( sumuncert[i] == NULL)
        {
            cerr << "Can't allocate sumuncert array! Buy more ram!\n";
            return 1;
        }
        for (j = 0; j < xcount; j++)
        {
            sumuncert[i][j]=0.0;
        }
    }

    //Set bounds for computing

    cerr << "Done Initializing.\n";

    //compute the standard error in each cell from all the soundings


//First determine the type of input file
//Valid types are xyzu and HIPS .def file
    cerr << "Determining file type\n";

    if ((infile = fopen(fn, "r"))==NULL)
    {
        printf("\nCannot open %s\n",fn);
        return(1);
    }
    fscanf(infile, "%s", firstchars);

    cerr << firstchars << "\n";
    fclose (infile);

//In the case of a HIPS weighted grid, the .def file is opened, and each line which was used to create the wg is opened
    //and read.  In addition to computing the standard error from the weighted mean, if any outstanding flags are
set, that
    //measurement is honored in the grid if it is shoaler than the grid value (a safety net for multiple outstanding
flags
    //and badly set outstanding flags.
    //Future implementation should include measurement-by-measurement uncertainty estimation and estimation
of the interdependence
    //of measurements in the integration into the grid.
    //Also need to add a section that creates a gdp file with data about each honored point (source information)
    //Also need to open and parse the sheet *.fsh file which contains the projection information,
    if (strcmp(firstchars,"[HIPS") ==0)
    {
        if ((infile = fopen(fn, "r"))==NULL)
        {
            printf("\nCannot open %s\n",fn);
            return(1);
        }

        //Prepare the projection information
        clat=45.0;                                      //This should come from the *.fsh file
        utm_zone=19;                            //This should come from the *.fsh file
        cmer=-177.0+6*(utm_zone-1);
        if ((proj = projection_new_utm(cmer, clat)) == NULL) {
            cerr << "Bitch about projection...\n";
        }
```

```
        donefile=0;                                             //donefile is a clunky way flag the end
of the *.def file
        while (donefile==0)  //go through each line entry in the def file
        {
            if (fscanf(infile, "%s", tempstring)!=1)  //Gets the next line and checks for the EOF
            {
                donefile=1;
                cerr << "done with lines\n";
            }
            while (strcmp(tempstring, "LINE")!=0  && donefile==0) //Skips to the next "LINE"
statement
            {
                if (fscanf(infile, "%s", tempstring)!=1)
                {
                    donefile=1;
                    cerr << "done with lines\n";
                }
            }
            if (donefile==0)                        //Parses the "LINE" statement for pvdl
            {
                project[0]=0;
                vessel[0]=0;
                day[0]=0;
                line[0]=0;
                //Parse the PVDL Info
                fscanf(infile, "%s", tempstring);
                fscanf(infile, "%s", tempstring);
                strcat(project, strtok(tempstring,"\\"));
                strcat(vessel, strtok(NULL,"\\"));
                strcat(day, strtok(NULL,"\\"));
                strcat(line, strtok(NULL,"\,"));
                cerr << "pvdl " << project << " " << vessel << " " << day << " " << line <<
"\n";
                //Opens the processed depths file

                processedDepthsSensor=ProcessedDepthsOpen(project, vessel, day, line,
"query", &ret);
                if (processedDepthsSensor==NULL){
                    cerr << "Unable to open processed depths file\n";
                    cerr << "Rcode " << ret << "\n";
                    if (ret != 0) exit (1);
                }
                //Gets the Line Summary
                rcode = ProcessedDepthsSummary (processedDepthsSensor, &toolType,
                  &coordinateType, &numLineSegments, &numProfiles, &numDepths,
&minTime,
                  &maxTime, &minDepth, &maxDepth, &minLat, &maxLat, &minLong,
&maxLong,
                  &summaryStatus);
                //Check for line rejection
                if (!PD_REJECT_ENTIRE_LINE(summaryStatus))
                {

                    //Opens each profile in turn
                    for (i=0;i<numProfiles;i++)
                    {

                      rcode = ProcessedDepthsReadProfileSeq (processedDepthsSensor,
&numDepths,
                            &startDepth, &time, &latitude, &minLat, &maxLat,
&longitude, &minLong,
```

```
                                                        &maxLong, &gyro, &heave, &pitch, &roll, &tide,
&vesselVel,
                                                        &transducerPitch, &transducerRoll, &profileStatus);

                                                  // Read the soundings...
      //
                                                  for (j=0;j<numDepths;j++)
                                                  {

                                                        rcode = ProcessedDepthsReadSeq
(processedDepthsSensor, &time,
                                                              &alongTrack, &acrossTrack, &latitude,
&longitude, &depth,
                                                              &depthAccuracy, &status);
                                                        if (rcode != 0) exit (1);
                                                        if (!PD_PROFILE_REJECTED(profileStatus) &&
!PD_DEPTH_REJECTED(status))
                                                        {
                                                              lat64=latitude;
                                                              lon64=longitude;
                                                              projection_ll_to_en_rad(proj, lon64, lat64, &east,
&north);
                                                              //cx, cy, z are in local offset coordinate system
                                                              z = -1*depth;
                                                              z -= zref;
                                                              cx=(int)((east-xcorner)/cellsize); //cell x
                                                              cy=(int)((north-ycorner)/cellsize); //cell y

                                                              if ((north-ycorner>0) && (east-xcorner>0) &&
((xcount*cellsize+xcorner)-east>0) && ((ycount*cellsize+ycorner)-north>0)){
                                                                    if (numpoints[cy][cx]>=0){
//numpoints=-1 means protected point
                                                                          numpoints[cy][cx]+= 1;
                                                                          uncertainty[cy][cx]=
(float)(uncertainty[cy][cx]+pow((dem[cy][cx]-z),2)); //Uncertainty is sum of squares
                                                                          sumuncert[cy][cx]+=
pow(.25+depth*.013*depth*0.013,0.5); //Assign Uncertainty to IHO1
                                                                    }

                                                                    if
(PD_DEPTH_OUTSTANDING(status) && dem[cy][cx]<z){ //Outstanding flag is set
                                                                          cerr << "Found Outstanding
depth at " << east << " " << north << " " << depth << "\n";
                                                                          dem[cy][cx]=z;
                                                                          numpoints[cy][cx]=-
1;//Replace dem value with measurement value

      uncertainty[cy][cx]=IHOlimit(depth,1);
                                                                    }
                                                              } //end geographic test


                                                        } //end rejection test

                                                  } //end loop through all soundings in a profile

                                                 } //end loop through all profiles in a line
                                           } //end line rejection test
                                           ret=ProcessedDepthsClose (processedDepthsSensor);
                                           if (ret !=0) {
                                                 cerr << "Can't close line " << ret << "\n";
                                                 exit (1);
```

```
                }

            } //end EOF test
        } //end while statement, going through each line in *.def file
    } //end HIPS *.def selection


    else // assume xyzu file input  z is positive depth, u is uncertainty, space or tab delimited, also assume that
data has been decimated, and so there should be no coincident points
    {
        if ((infile = fopen(fn, "r"))==NULL)
        {
            printf("\nCannot open %s\n",fn);
            return(1);
        }
        cerr << "Computing Uncertainty from xyzu... ";
        while(fscanf(infile," %lf %lf %lf %lf",&x,&y,&z, &u)==4)
        {
            z -= zref;
            cx=(int)((x-xcorner)/cellsize); //cell x
            cy=(int)((y-ycorner)/cellsize); //cell y

            if ((y-ycorner>0) && (x-xcorner>0) && ((xcount*cellsize+xcorner)-x>0) &&
((ycount*cellsize+ycorner)-y>0) && dem[cy][cx]>0)
            {
                if (numpoints[cy][cx]==0){
                    numpoints[cy][cx]+= 1;
                    uncertainty[cy][cx]= (float)(uncertainty[cy][cx]+pow((dem[cy][cx]-
z),2));  //uncertainty is the sum of the squares of the measurement to grid differences
                    sumuncert[cy][cx]+= u; //sumuncert is the sum of measurement
uncertainty
                }
            }
        }
    } //end of measurement ingest
    cerr << "done scanning\n";

    //This section recomputes the standard error from the sum of the squares

    cerr << "computing backward error\n";
    for (j=0; j<ycount;j++)
    {
        for (i=0; i<xcount;i++)
        {
            if (dem[j][i]==0){uncertainty[j][i]=0.0;}
            if (dem[j][i]>0 && numpoints[j][i]>2) //all points within the grid which are not protected
----dem[j][i]>0 &&
            {
                uncertainty[j][i]=1.96*(float)sqrt(uncertainty[j][i]/numpoints[j][i]); //95%
threshold
                //
            }
            else if (dem[j][i]>0 && numpoints[j][i]<3 && numpoints[j][i]>0) //all points within the
grid which are not protected ----dem[j][i]>0 &&
            {
                uncertainty[j][i]=0.001; //assures that backward error will not be used for
statistically insignificant samples
            }
            else if (numpoints[j][i]==0)  //nodes with no measurements
            {
                uncertainty[j][i]=0.0;
            }
```

```
                else if (numpoints[j][i]==-1)//protected point numpoints=-1
                {
                        cerr << "protected " << uncertainty[j][i] << "\n";
                }
                if (uncertainty[j][i]>5){cerr << " Numpoints= " << numpoints[j][i] << " Uncert=" <<
uncertainty[j][i]<<"\n";}
        }

    }
    //Uncertainty is now backward error

/*  //This section corrects the standard error for local roughness (slope)
//  Commented out after further consideration makes it antithetical to overall approach.  Cool though, maybe still
works.
//  In order to reinvigorate, the roughness computation will need to be reset in main.

    cerr << "Correcting Standard Error\n";
    for (j=0; j<ycount; j++)
    {
        for (i=0; i<xcount; i++)
        {
                if (dem[j][i]>0 && numpoints[j][i]>0)
                {
                        uncertainty[j][i]=(float)(uncertainty[j][i]/(8.5*roughness[j][i]+1))*1.96;
                }
        }

    }
*/
    //This section computes the predicted error based on the measurement uncertainty (forward, predicted error)
    for (j=0; j<ycount; j++)
    {
        for (i=0; i<xcount; i++)
        {
                if (dem[j][i]>0 && numpoints[j][i]>0)
                {
                        tmp_backward_error=uncertainty[j][i];  //cache the backward error
                        //  The following code was eliminated because it assumed independent
measurements. Below is substituted.
//tmp_forward_error=sumuncert[j][i]/numpoints[j][i]/sqrt(__max(numpoints[j][i],1));  //compute the predicted error--
avg uncertainty
                        tmp_forward_error=sumuncert[j][i]/numpoints[j][i];  //compute the predicted
error--avg uncertainty
                        //cerr << "Back " << tmp_backward_error << "  Fwd " << tmp_forward_error
<< "\n";
                        uncertainty[j][i]=(float)__max(tmp_forward_error,tmp_backward_error);
//choose the greater of forward and backward error
                        //cerr << "Greater "<< uncertainty[j][i] << "\n";
//                      uncertainty[j][i]=(float)__max(uncertainty[j][i],0.5*IHOlimit(-
1*(dem[j][i]+zref),1)); //limit uncertainty to half IHO as a first cut at interdependence of measurements
                        //cerr << "HalfIHO " << 0.5*IHOlimit(-1*(dem[j][i]+zref),1) <<"\n";
                        //if (uncertainty[j][i]>5){cerr << "Greater "<< uncertainty[j][i] << "\n";}

                }
        }

    }

    //fill in gaps with sparse data model
    cerr << "Fill in gaps with sparse data model...\n";
    cellsearch=40;//(int) (300/cellsize);
    double dx, dy, sumsq, intdist;
```

```
    for (j=0; j<ycount; j++)
    {
        if (j/100.0==j/100)
            {
                cerr << "Done " << j << " rows of " << ycount << ".\n";
            }

        for (i=0; i<xcount; i++)
        {
            if (dem[j][i]>0 && uncertainty[j][i]==0.0 && numpoints[j][i]==0)
            {
        //  cerr << "Uncertainty " << uncertainty[j][i] << "\n";
                sumz=0.0;
                sumsquares=0.0;
                numcells=0;
                mindist=30*cellsize;
                base_uncertainty=10;
                z=dem[j][i]+zref;
                uncert_cap=__min(pow(z*z,.5),maxuncertainty);
                for (l=__max(0,j-cellsearch);l<__min(ycount,j+cellsearch);l++)
                {
                    for (k=__max(0, i-cellsearch); k<__min(xcount,i+cellsearch);k++)
                    {
                        //Compute Roughness
                        if (dem[l][k]>0.0)
                        {
                            sumz+=dem[l][k];
                            sumsquares+=dem[l][k]*dem[l][k];
                            numcells+=1;
                        }

                        if (numpoints[l][k]>0)
                        {
                            dx=(k-i)*(k-i);
                            dy=(l-j)*(l-j);
                            sumsq=dx+dy;
                            intdist=sqrt(sumsq);
                            tempdist=__max(intdist*cellsize,0);//-hor_error
                            if (tempdist<mindist)
                            {
                                mindist=tempdist;
                                base_uncertainty=uncertainty[l][k];
                            }
                        }

                    }
                }
                //estimate growth rate based on local roughness
                growthrate=__max((float)sqrt(sumsquares/numcells-
pow(sumz/numcells,2))/70,mingrowthrate);
                //cerr << "growthrate " << growthrate << "\n";
                expgrowthrate=.035*growthrate;
                //cerr << "mindist" << mindist << "\n";

    uncertainty[j][i]=__min(base_uncertainty+mindist*growthrate+pow(mindist,2)*expgrowthrate,uncert_cap);

                //cerr << uncertainty[j][i] << "\n";
/*              if (base_uncertainty>5){
                    cerr << "base=" << base_uncertainty << " mindist= " << mindist <<
".  Growthrate=" << growthrate << "uncert=" << uncertainty[j][i] << "\n";
                }
*/
```

```
                        }
                }
        }

        cerr << "done.\n";
#endif
        return 0;
}
```

Defocus and DoubleBuffer

```
int GutmManip::Defocus(double raderr, double zerr)
{
        int i,j,k,l;
        //double raderr, zerr;
        int cellsearch;
        double celldist;
        double newz;
        double zerr_applied;
        double ur, urf;
        //raderr=5.0;
        //zerr=0.3;
   cerr << "Allocating defocused array...";

        defocused = new float*[ycount];
        if ( defocused == NULL)
        {
                cerr << "Can't allocate defocus array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
        {
                defocused[i] = new float[xcount];
                if ( defocused[i] == NULL)
                {
                        cerr << "Can't allocate defocus array! Buy more ram!\n";
                        return 1;
                }
      for (j = 0; j < xcount; j++)
      {
         defocused[i][j]=0.0;
      }
        }

        cerr << "Defocusing DEM with position error " << raderr << " and z error variable" << ".\n";
        cellsearch=(int)(raderr/cellsize);
        cerr << "cellsearch " << cellsearch << "\n";
        for (j=0; j<ycount;j++)
        {
                if (j/100.0==j/100)
                {
                        cerr << "Done " << j << " rows of " << ycount << ".\n";
                }
                for (i=0; i<xcount;i++)
                {
                        if (dem[j][i]>0)
                        {
                                for (k=__max(0,j-cellsearch);k<__min(ycount,j+cellsearch); k++)
                                {
                                        for (l=__max(0,i-cellsearch);l<__min(xcount,i+cellsearch); l++)
```

```
                    {
                        celldist=cellsize*sqrt(pow(k-j,2)+pow(l-i,2));
                        if (dem[k][l] >0 && celldist<raderr)
                        {
                            if (uncertainty==NULL)
                            {
                                zerr_applied=0.0;
                            }
                            else
                            {
                                if (zerr>uncertainty[j][i])
                                {
                                    zerr_applied=0.0;//zerr;
                                }
                                else
                                {

    zerr_applied=__max(uncertainty[j][i]-zerr,1.0);//temporary uncertainty limit 1 m
                                }
                            }

    newz=dem[k][l]+(sin(acos(celldist/raderr))*(dem[j][i]-dem[k][l] + zerr_applied));

    defocused[k][l]=(float)__max(defocused[k][l],newz);
                        }
                    }
                }

            }
        }

    }
    cerr << "Uncertainty " << uncertainty_present << "\n";
    for (i = 0; i < ycount; i++)
    {
        for (j = startline[i]; j < endline[i]; j++)
            {
                if (uncertainty_present && dem[i][j]>0)
                {
                    ur=(dem[i][j]-defocused[i][j])/uncertainty[i][j];
                    urf=0.004657*pow(ur,5) + 3.837 * pow(10,-13) * pow(ur,4) -
0.088403 * pow(ur,3) + 0.7603 * ur + 1.0;
                    if (urf<0.0001)
                    {
                        urf=0.0001;
                    }

                    uncertainty[i][j]=uncertainty[i][j]*urf;
                    if (uncertainty[i][j]<0.1)
                    {
                        uncertainty[i][j]=0.1;
                    }
                }
                dem[i][j]=defocused[i][j];


            }
    }

    return 0;
}
```

```
int GutmManip::DoubleBuffer(double db_radius)
{
        int i,j,k,l;
        int cellsearch;
        double celldist;
        double newz;
        double ur, urf;

  cerr << "Allocating defocused array...";

        defocused = new float*[ycount];
        if ( defocused == NULL)
        {
                cerr << "Can't allocate defocus array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
        {
                defocused[i] = new float[xcount];
                if ( defocused[i] == NULL)
                {
                        cerr << "Can't allocate defocus array! Buy more ram!\n";
                        return 1;
                }
    for (j = 0; j < xcount; j++)
    {
      defocused[i][j]=0.0;
    }
        }

        cerr << "Allocating cache array...";

        cache = new float*[ycount];
        if ( cache == NULL)
        {
                cerr << "Can't allocate cache array! Buy more ram!\n";
                return 1;
        }

        for (i = 0; i < ycount; i++)
        {
                cache[i] = new float[xcount];
                if ( cache[i] == NULL)
                {
                        cerr << "Can't allocate cache array! Buy more ram!\n";
                        return 1;
                }
    for (j = 0; j < xcount; j++)
    {
      cache[i][j]=0.0;
    }
        }


        cerr << "done.\n";

        //cerr << "Defocusing DEM with position error " << raderr << " and z error variable" << ".\n";
        cellsearch=(int)(db_radius/cellsize);
        //cellsearch=cellsearch/2;
```

```
    cerr << "cellsearch " << cellsearch << "\n";
    //First Buffer Up
    for (j=0; j<ycount;j++)
    {
        if (j/200.0==j/200)
        {
            cerr << "Done " << j << " rows of " << ycount << ".\n";
        }
        for (i=0; i<xcount;i++)
        {
            if (dem[j][i]>0)
            {
                for (k=__max(0,j-cellsearch);k<__min(ycount,j+cellsearch); k++)
                {
                    for (l=__max(0,i-cellsearch);l<__min(xcount,i+cellsearch); l++)
                    {
                        celldist=cellsize*sqrt(pow(k-j,2)+pow(l-i,2));
                        if (dem[k][l] >0 && celldist<db_radius)
                        {
                            newz=dem[j][i]+sqrt(pow(db_radius,2) -
pow(celldist,2));

    defocused[k][l]=(float)__max(defocused[k][l],newz);
                        }
                    }
                }

            }
        }

    }
    //Then Buffer Down
    for (j=0; j<ycount;j++)
    {
        for (i=0;i<xcount;i++)
        {
                cache[j][i]=dem[j][i];
                dem[j][i]=0;
        }

    }
    for (j=0; j<ycount;j++)
    {
        if (j/200.0==j/200)
        {
            cerr << "Done " << j << " rows of " << ycount << ".\n";
        }
        for (i=0; i<xcount;i++)
        {
            if (defocused[j][i]>0)
            {
                for (k=__max(0,j-cellsearch);k<__min(ycount,j+cellsearch); k++)
                {
                    for (l=__max(0,i-cellsearch);l<__min(xcount,i+cellsearch); l++)
                    {
                        celldist=cellsize*sqrt(pow(k-j,2)+pow(l-i,2));
                        if (defocused[k][l] >0 && celldist<db_radius)
                        {
                            newz=defocused[j][i]-sqrt(pow(db_radius,2) -
pow(celldist,2));
                            if (dem[k][l]>0)
                            {
```

```
    dem[k][l]=(float)__min(dem[k][l],newz);
                            }
                            else
                            {
                                dem[k][l]=(float) newz;
                            }
                        }
                    }
                }

            }
        }

    }
// Compute new uncertainty

    cerr << "Computing new Uncertainty...";
    for (i = 0; i < ycount; i++)
    {
        for (j = startline[i]; j < endline[i]; j++)
            {
                if (uncertainty_present && dem[i][j]>0)
                {
                    ur=(cache[i][j]-dem[i][j])/uncertainty[i][j];
                    urf=0.004657*pow(ur,5) + 3.837 * pow(10,-13) * pow(ur,4) -
0.088403 * pow(ur,3) + 0.7603 * ur + 1.0;
                    if (urf<.01)
                    {
                        urf=0.01;
                    }

                    uncertainty[i][j]=uncertainty[i][j]*urf;
                }

            }
    }
    cerr << "done\n";



    return 0;
```

GLOSSARY

Bin, binned, binning—One process used to thin dense hydrographic data is to select the shoalest sounding in a N by N meter box, called a bin. The process is called binning.

Cartographic Interpretation—The process of interpreting data for a particular purpose, such as nautical charting. Selected features are retained and highlighted in the data and other features are neglected, depending on the purpose of the product.

Datum—Nautical charts are referenced to a tidal datum, such as mean lower low water (MLLW) used in the US.

Deconflict—The process of reconciling conflicting information based on the age of the informatation, reliability of the source, etc.

Downsampling—The process of reducing the number of data points in a data set, generally by keeping shoaler soundings when they are near deeper soundings.

Electronic Navigational Chart (ENC)—The ENC is a new breed of official vector charts, made up of attributed points and lines, much like a GIS database.

Flagging as Rejected—In the typical hydrographic process, suspect soundings are not deleted. Instead, a flag is set that marks them not to be used. The flag is the "rejected" flag.

Fliers—Outliers. Soundings which are not associated with the seafloor.

Generalization—The process of reducing the complexity of a presentation of map information to make it appropriate for a particular scale of product. For example, shoreline which shows a detailed information about a cove at a large scale may only show a small indentation on a small scale chart.

IHO—International Hydrographic Organization. Body which sets standards for hydrographic surveying.

Least Depth—Shallowest depth over a feature.

Nautical Charts—Special type of map used for navigation on the water. It contains information on the shoreline, depth of the water, buoys, and any special information of interest to the mariner.

Reconnaissance-density surveys—Surveys designed to broadly assess the depth of the water when little other information is known. Usually do not result in obtaining the least depths on features.

Registry Number—The system used by NOAA to track all surveys. The registry number is an index to the survey.

Shoal-biased—The method of downsampling of data where only the shoalest soundings are preserved. This process ensures that least depths on features is preserved during generalization.

Smooth sheet—The printed product of a survey. Contains much of the same information as the chart, but at a scale appropriate to the data collected.

Supersede—When new data is collected and applied to the chart, it is said to supersede the existing information.