NOAA Technical Report NESDIS 132

Assessing Errors in Altimetric and Other Bathymetry Grids

Washington, D.C.
January 2011

U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
National Environmental Satellite, Data, and Information Service

NOAA TECHNICAL REPORTS

National Environmental Satellite, Data, and Information Service

The National Environmental Satellite, Data, and Information Service (NESDIS) manages the Nation's civil Earth-observing satellite systems, as well as global national data bases for meteorology, oceanography, geophysics, and solar-terrestrial sciences. From these sources, it develops and disseminates environmental data and information products critical to the protection of life and property, national defense, the national economy, energy development and distribution, global food supplies, and the development of natural resources.

Publication in the NOAA Technical Report series does not preclude later publication in scientific journals in expanded or modified form. The NESDIS series of NOAA Technical Reports is a continuation of the former NESS and EDIS series of NOAA Technical Reports and the NESC and EDS series of Environmental Science Services Administration (ESSA) Technical Reports.

An electronic copy of this report may be obtained at http://www.star.nesdis.noaa.gov/star/socd_pub.php

A limited number of copies of earlier reports are available by contacting Susan Devine, NOAA/NESDIS, E/RA, 5200 Auth Road, Room 701, Camp Springs, Maryland 20746, (301) 763-8127 x136. Copies can also be ordered from the National Technical Information Service (NTIS), U.S. Department of Commerce, Sills Bldg., 5285 Port Royal Road, Springfield, VA 22161, (703) 487-4650 (prices on request for paper copies or microfiche, please refer to PB number when ordering). A partial listing of more recent reports appears below:

NESDIS 102 **NOAA Operational Sounding Products From Advanced-TOVS Polar Orbiting Environmental Satellites. Anthony L. Reale, August 2001.**

NESDIS 103 **GOES-11 Imager and Sounder Radiance and Product Validations for the GOES-11 Science Test. Jaime M. Daniels and Timothy J. Schmit, August 2001.**

NESDIS 104 **Summary of the NOAA/NESDIS Workshop on Development of a Coordinated Coral Reef Research and Monitoring Program. Jill E. Meyer and H. Lee Dantzler, August 2001.**

NESDIS 105 **Validation of SSM/I and AMSU Derived Tropical Rainfall Potential (TRaP) During the 2001 Atlantic Hurricane Season. Ralph Ferraro, Paul Pellegrino, Sheldon Kusselson, Michael Turk, and Stan Kidder, August 2002.**

NESDIS 106 **Calibration of the Advanced Microwave Sounding Unit-A Radiometers for NOAA-N and NOAA-N=. Tsan Mo, September 2002.**

NESDIS 107 **NOAA Operational Sounding Products for Advanced-TOVS: 2002. Anthony L. Reale, Michael W. Chalfant, Americo S. Allegrino, Franklin H. Tilley, Michael P. Ferguson, and Michael E. Pettey, December 2002.**

NESDIS 108 **Analytic Formulas for the Aliasing of Sea Level Sampled by a Single Exact-Repeat Altimetric Satellite or a Coordinated Constellation of Satellites. Chang-Kou Tai, November 2002.**

NESDIS 109 **Description of the System to Nowcast Salinity, Temperature and Sea nettle (*Chrysaora quinquecirrha*) Presence in Chesapeake Bay Using the Curvilinear Hydrodynamics in3-Dimensions (CH3D) Model. Zhen Li, Thomas F. Gross, and Christopher W. Brown, December 2002.**

NOAA Technical Report NESDIS 132

Assessing Errors in Altimetric and Other Bathymetry Grids

Karen M. Marks
NOAA/NESDIS/STAR
1335 East West Highway
SSMC1 / Room 5342
Silver Spring, MD 20910

Walter H.F. Smith
NOAA/NESDIS/STAR
1335 East West Highway
SSMC1 / Room 5408
Silver Spring, MD 20910

Washington, DC
January 2011

U.S. DEPARTMENT OF COMMERCE
Gary Locke, Secretary

National Oceanic and Atmospheric Administration
Dr. Jane Lubchenco, Under Secretary of Commerce for Oceans and Atmosphere and NOAA Administrator

National Environmental Satellite, Data, and Information Service
Mary Kicza, Assistant Administrator

Contents

Figures

Abstract

We have developed a method of assessing errors in altimetric bathymetry models. This method was used in the paper "Evolution of errors in the altimetric bathymetry model used by Google Earth and GEBCO" by Marks et al. (2010) to evaluate errors in the Smith and Sandwell (1997) bathymetric model and its updates. The technique involves comparing model depths to multibeam "ground truth" depths, with "errors" being the differences. Other gridded bathymetry data sets and interpolation algorithms can likewise be tested if selected control data are withheld for subsequent comparison to resulting grids. This technical report serves to document our error assessment method in detail.

1. Introduction

Satellite altimeters cover the world's oceans with evenly distributed resolution and at a far greater density than ship tracks, which afford only sparse and irregular coverage. The altimeter measures sea surface height, from which gravity anomalies may be derived and seafloor structure inferred. Even though new bathymetric surveys continue to be collected, there are still gaps as large as 10^5 km^2 that remain untraversed by ship. These gaps can be filled in with estimated depths from bathymetric models that combine depths derived from satellite gravity data with measurements made by ship. The Smith and Sandwell (1997) bathymetric model is widely used in the scientific community and has recently been incorporated into GEBCO (General Bathymetric Charts of the Ocean) products and the popular web application Google Earth. It is important to perform systematic evaluations of the model updates in order to identify errors and assess accuracy, and to confirm the model is being improved. Marks et al. (2010) developed a method of assessing errors in bathymetric models. The technique involves comparing JAMSTEC (Japan Agency for Marine Earth and Science Technology) multibeam "ground truth" depths to model depths, defining "errors" as the differences. This method can also be used on other types of data and to test various gridding algorithms if control data are withheld for subsequent comparison.

2. Data and Software Retrieval

2.1 JAMSTEC Data

Many types of geophysical data are available for download from the JAMSTEC website (http://www.godac.jamstec.go.jp/cruisedata/e/), which is shown in Figure 1. To assess errors in the bathymetric model, we used JAMSTEC multibeam bathymetry and gravity data obtained from this website.

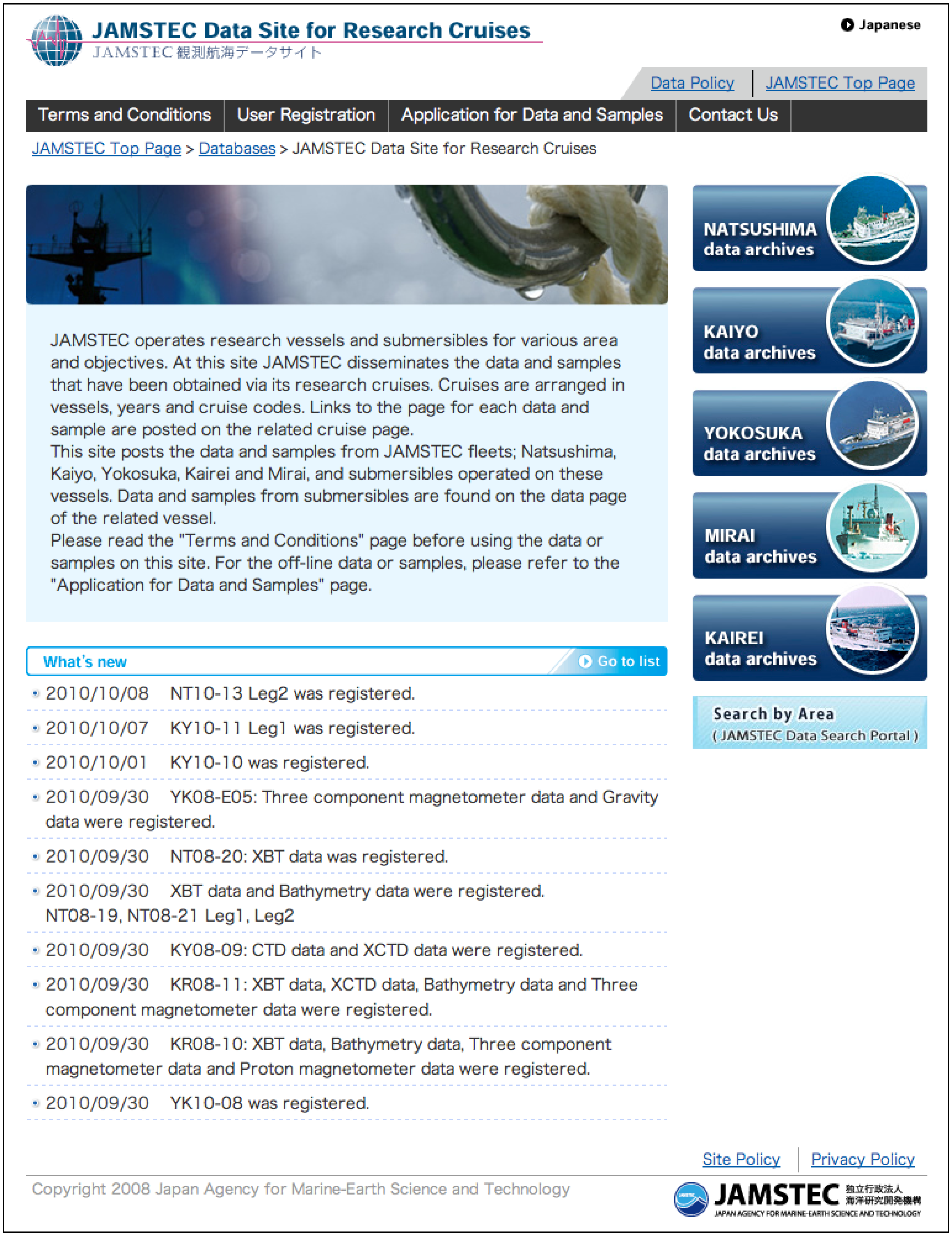


Figure 1. JAMSTEC Data Site for Research Cruises

2.1.1 Multibeam Data

Multibeam data may be downloaded from the JAMSTEC website as follows. The user first selects the “data archives” link for one of the five research vessels on the main JAMSTEC website (Fig. 1). The link opens a global map of cruises color-coded by collection year (Fig. 2). The user selects the desired year, and then the cruise ID. Selecting "Apply" opens the individual cruise web page (Fig. 3). To reach the bathymetry web page, the user selects "Bathymetry" from the right side of the cruise web page (see Fig. 3).

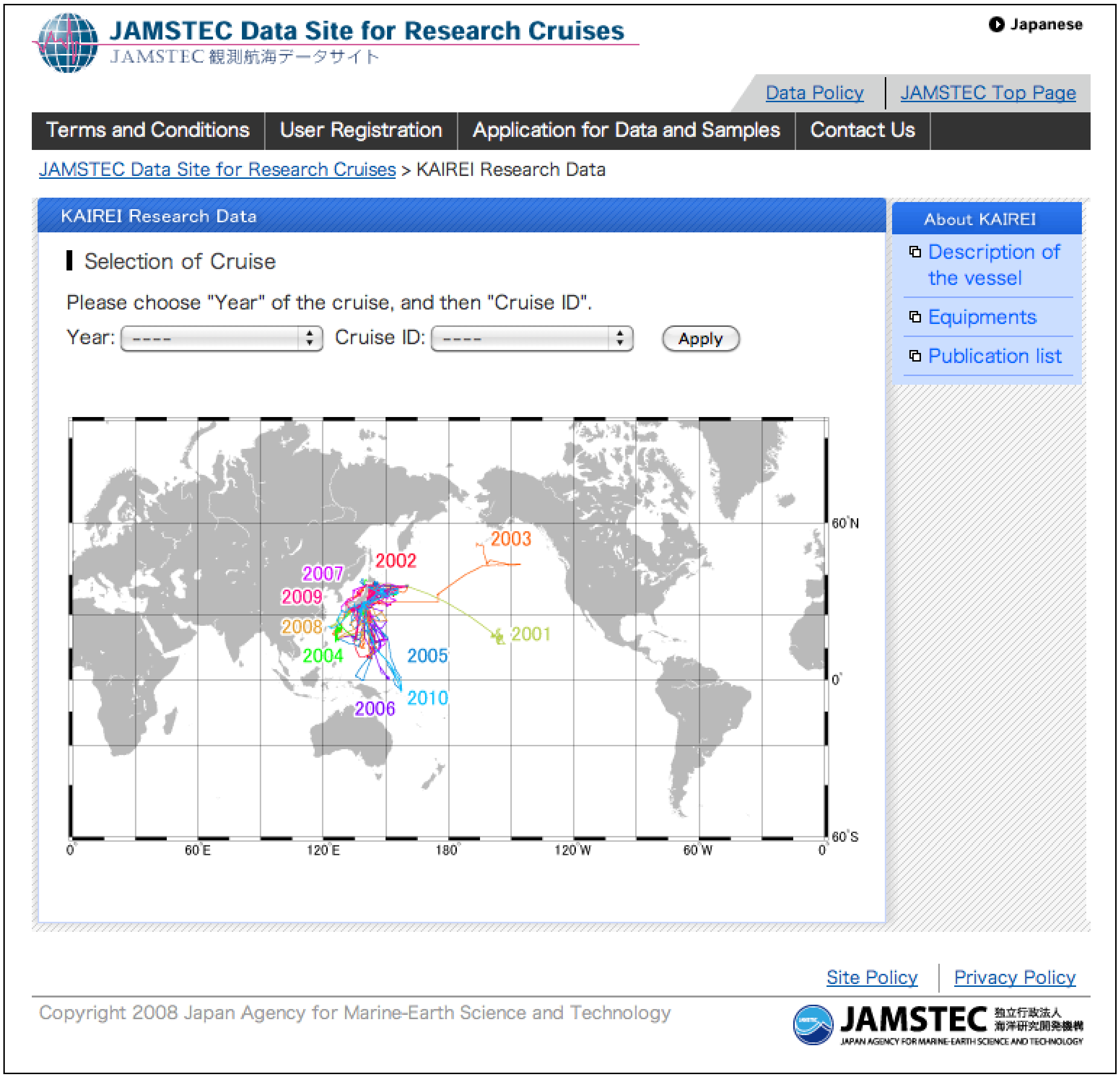


Figure 2. KAIREI Research Data web page.

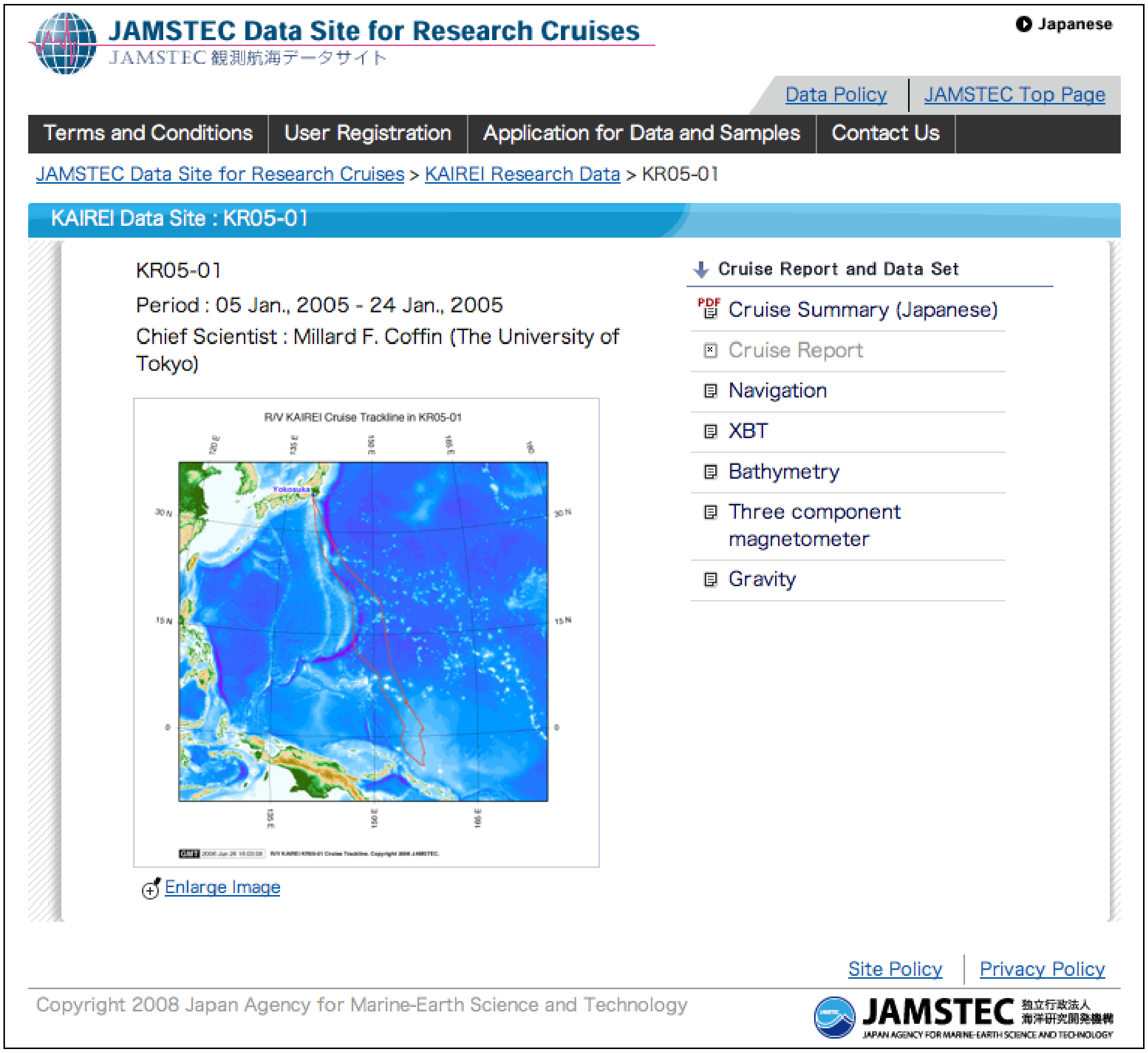


Figure 3. KR05-01 Cruise web page.

Bathymetry data may be downloaded from the bathymetry web page (Fig. 4) via the “Data” link, as well as information on the instruments, data format, and collection system details (i.e., the “Instruments,” “Format description,” and “Readme” links, respectively). The bathymetry data may be downloaded as a number of compressed files in “zip” format (see Fig. 5) that need to be uncompressed (“unzipped”) and then assembled sequentially into a single file. The bathymetry data are ASCII *xyz* (longitude, latitude, depth) multibeam echo sounder ping records. There may not be data along the entire track plot shown on the bathymetry web page and making a plot of the coverage will reveal any gaps.

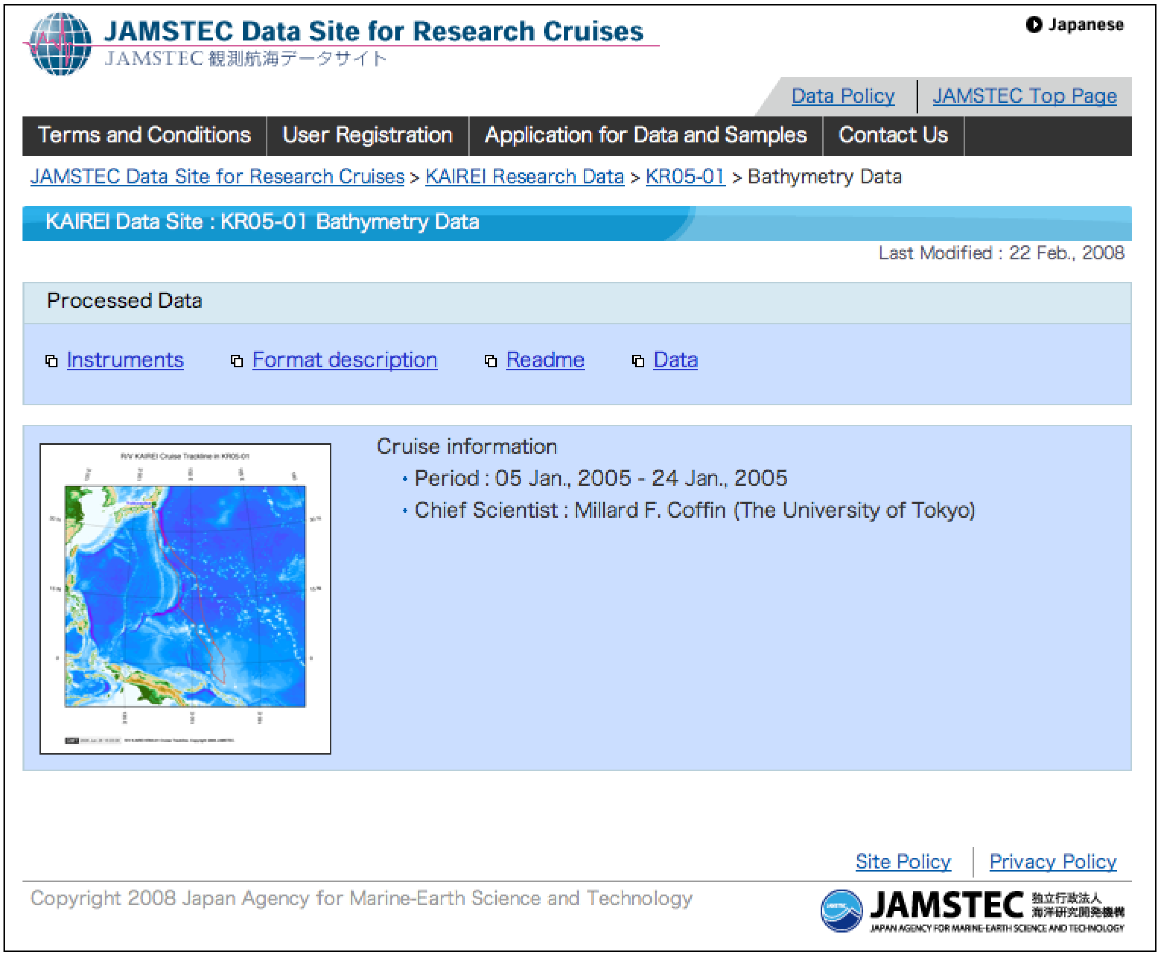
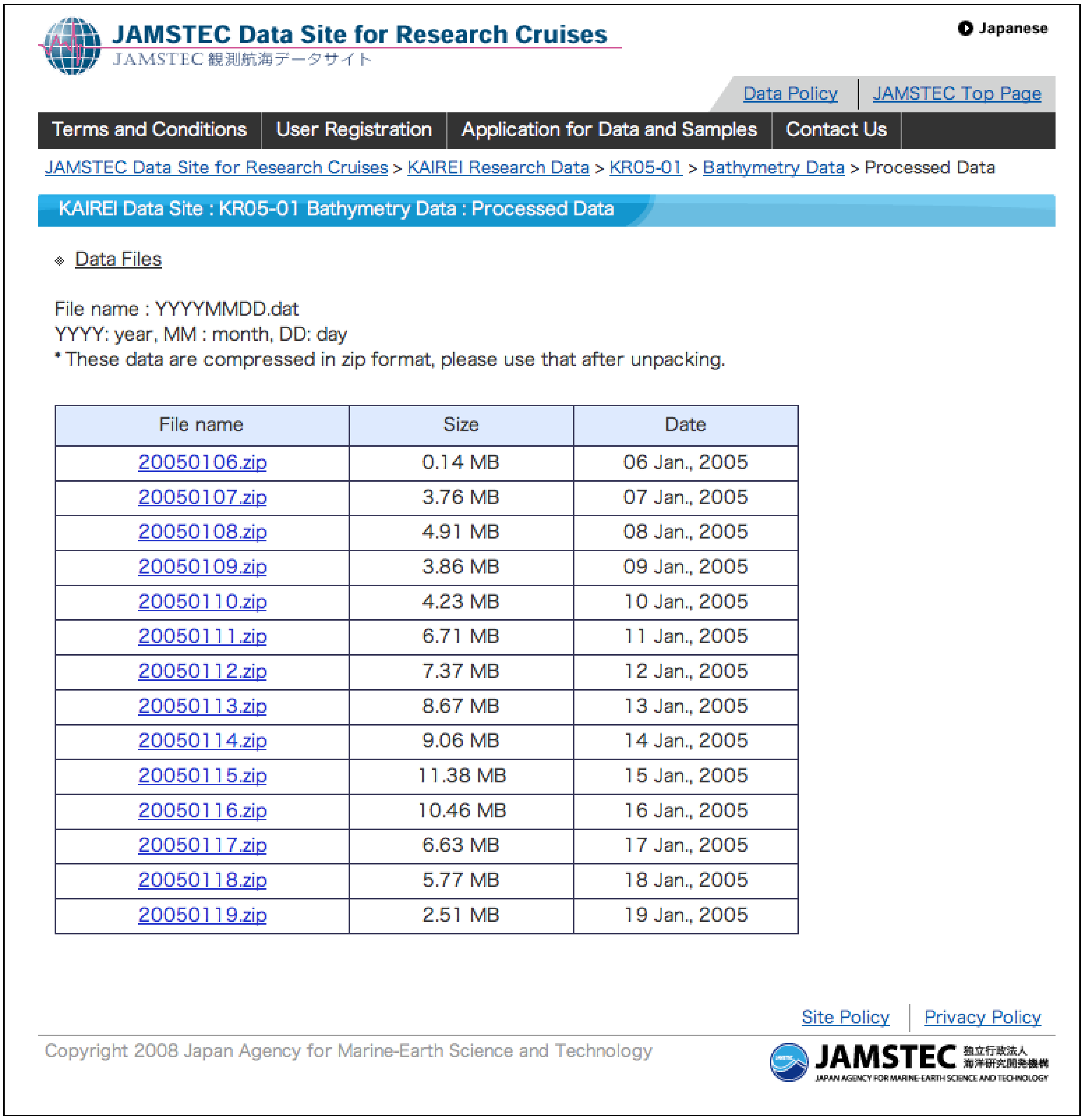


Figure 4. KR05-01 Bathymetry Data web page.



JAMSTEC Data Site for Research Cruises
JAMSTEC 観測航海データサイト

Japanese

Data Policy | JAMSTEC Top Page

Terms and Conditions | User Registration | Application for Data and Samples | Contact Us

JAMSTEC Data Site for Research Cruises > KAIREI Research Data > KR05-01 > Bathymetry Data > Processed Data

KAIREI Data Site : KR05-01 Bathymetry Data : Processed Data

Data Files

File name : YYYYMMDD.dat
YYYY: year, MM : month, DD: day
* These data are compressed in zip format, please use that after unpacking.

File name	Size	Date
20050106.zip	0.14 MB	06 Jan., 2005
20050107.zip	3.76 MB	07 Jan., 2005
20050108.zip	4.91 MB	08 Jan., 2005
20050109.zip	3.86 MB	09 Jan., 2005
20050110.zip	4.23 MB	10 Jan., 2005
20050111.zip	6.71 MB	11 Jan., 2005
20050112.zip	7.37 MB	12 Jan., 2005
20050113.zip	8.67 MB	13 Jan., 2005
20050114.zip	9.06 MB	14 Jan., 2005
20050115.zip	11.38 MB	15 Jan., 2005
20050116.zip	10.46 MB	16 Jan., 2005
20050117.zip	6.63 MB	17 Jan., 2005
20050118.zip	5.77 MB	18 Jan., 2005
20050119.zip	2.51 MB	19 Jan., 2005

Site Policy | Privacy Policy

Copyright 2008 Japan Agency for Marine-Earth Science and Technology

JAMSTEC 独立行政法人 海洋研究開発機構
JAPAN AGENCY FOR MARINE-EARTH SCIENCE AND TECHNOLOGY

Figure 5. KR05-01 Bathymetry Data files available for download.

An alternate way to reach bathymetry web pages is for the user to click on the “JAMSTEC Data Search Portal” tool (see Fig. 1). This accesses instructions on how to use the tool as well as provides an entryway to the tool’s ArcIMS interface (see Fig. 6). On the ArcIMS interface, the user may drag the red box on the global map (upper left corner) to the desired study location, then select “Quick Search” to drag a box on the main map. This brings up a search result panel (see Fig. 7) that lists the cruises that traverse the study box. The bathymetry web page for the cruise of interest can then be reached by clicking on “Jump.”

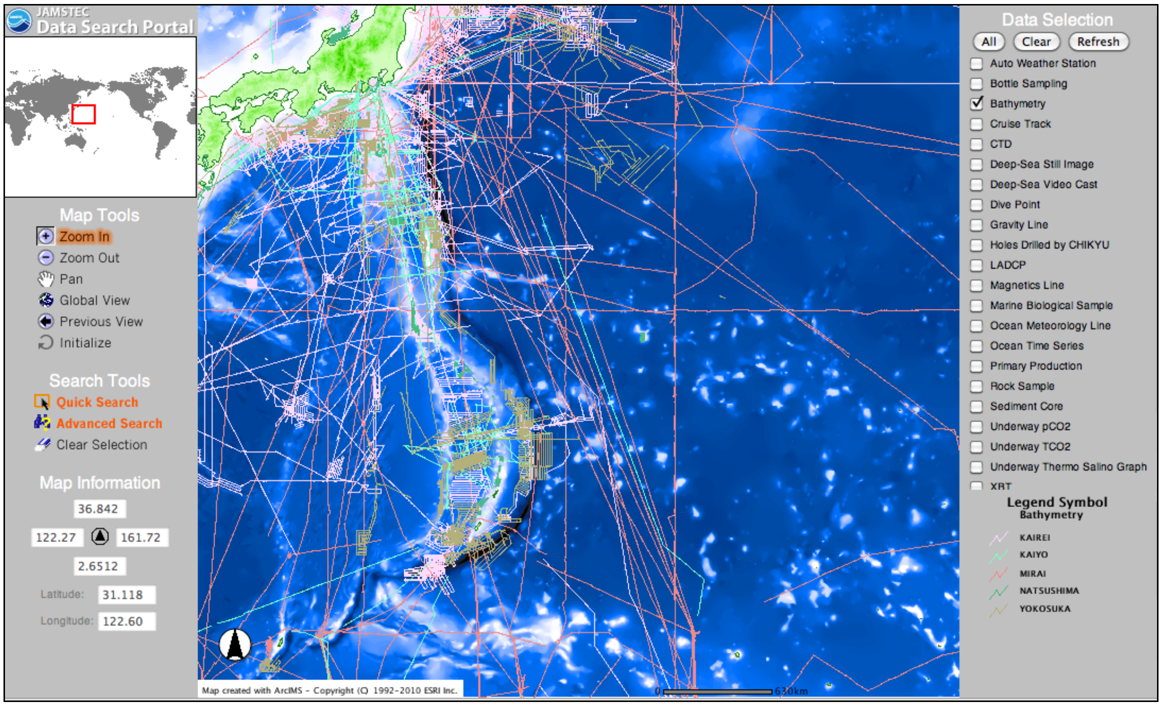


Figure 6. JAMSTEC Data Search Portal.

Search Result (Select Data Set : Bathymetry) 16 Selected

No.	Data ID	Latitude	Longitude	Period (start)	Period (end)	Ship Name	Cruise ID	DSRV Name	Dive No.	Web Site
1	0013-00000027			2002/02/20	2002/03/29	MIRAI	MR02-K02			Jump
2	0013-00000035			2003/02/19	2003/03/29	MIRAI	MR03-K01			Jump
3	0013-00000052			2004/11/16	2004/12/08	MIRAI	MR04-07			Jump
4	0013-00000054			2005/01/13	2005/02/18	MIRAI	MR04-08_leg2			Jump
5	0013-00000056			2005/05/24	2005/06/30	MIRAI	MR05-02			Jump
6	0013-00000064			2006/02/04	2006/03/17	MIRAI	MR06-01			Jump
7	0013-00000073			2007/02/15	2007/03/25	MIRAI	MR07-01			Jump
8	0013-00000075			2007/05/30	2007/07/13	MIRAI	MR07-03			Jump
9	0013-00000085			2008/07/01	2008/08/05	MIRAI	MR08-03			Jump
10	0013-00000211			2005/01/04	2005/01/23	KAIREI	KR05-01			Jump
11	0013-00000228			2005/12/09	2005/12/24	KAIREI	KR05-17			Jump
12	0013-00000242			2006/09/11	2006/09/22	KAIREI	KR06-12			
13	0013-00000264			2007/11/22	2007/12/01	KAIREI	KR07-16			Jump
14	0013-00000326			2002/09/29	2002/10/14	KAIYO	KY02-10_leg1			
15	0013-00000639			2001/01/03	2001/01/14	YOKOSUKA	YK01-01			
16	0013-00000915			2009/11/02	2009/12/11	MIRAI	MR09-04			

Figure 7. Data Search Portal Results.

2.1.2 Gravity Data

JAMSTEC also provides gravity data collected on certain cruises. The user can reach the gravity web page (Fig. 8) from the cruise web page (see Fig. 3) by selecting "Gravity" from the list on the right side of the page.

Figure 8. KR05-01 Gravity Data web page.

The gravity data are in a compressed ("zip" format) file that may be downloaded via the "Data Files" link. The file needs to be uncompressed ("unzipped"). The data are ASCII records containing date, time, latitude, longitude, total gravity, and depth values. We reduced the total gravity data to free-air gravity anomalies relative to the GRS80 Geodetic Reference System (see Appendix A). The depth values are center beam depths at the locations of the gravity measurements. Details on the instruments, data format, and collection systems may be downloaded from the "Instruments," "Format," and "Readme" links, respectively.

As described in Section 2.1.1 above, an alternate way of reaching the gravity web page is via the "JAMSTEC Data Search Portal" tool and then searching for gravity data within a desired study area. As described above, "Quick Search" will list the cruises that collected gravity data within the study area, with web links to the individual gravity web pages (e.g., Figure 8).

2.2 Altimetric Bathymetry Model

Recent versions of the Smith and Sandwell (1997) altimetric bathymetry model are available for download from the Scripps Institution of Oceanography website (http://topex.ucsd.edu/WWW_html/mar_topo.html), which is shown in Figure 9. Selecting "Global Topography" accesses a directory containing models that are available for download

(Fig. 10). We use version 12.1 (topo_12.1.img) for the examples in this report in conformity with the Marks et al. (2010) manuscript. The global topography models are in Mercator projection. SRTM30_Plus, which is also available for download (http://topex.ucsd.edu/WWW_html/srtm30_plus.html; this link is shown in Figure 9), is a 30 arc-second bathymetric grid in a geographic projection.

The user may opt to download ASCII xyz data for a selected area of the global topography model via the "Get an ASCII XYZ file" link (http://topex.uscd.edu/cgi-bin/get_data.cgi), this will download data from the most recent version 13.1. Alternatively, the user may download the entire global bathymetry model via the "Global Topography" link, which supplies a 712 Megabyte Sandwell/Smith "img format" raster file (e.g., topo_12.1.img). The "img format" is binary 2-byte integer in big-endian order. This "img" file can be converted to a netCDF format, and a subset extracted, using GMT (Wessel and Smith, 1998) routine "img2grd." Sample GMT commands to run this routine and others are listed in Appendix B (routine "img2grd" is in Appendix B.1), and a description of how to obtain GMT software follows in the next section.

Figure 9. Scripps Institution of Oceanography web page for Global Topography.

Index of ftp://topex.ucsd.edu/pub/global_topo_1min/

Up to higher level directory

Name	Size	Last Modified	
.DS_Store	7 KB	5/25/09	12:00:00 AM
.gmtcommands4	1 KB	8/21/07	12:00:00 AM
COPYRIGHT.txt	2 KB	7/25/08	12:00:00 AM
README_V11.1.txt	15 KB	12/28/08	12:00:00 AM
REFERENCES_V11.1.txt	7 KB	1/14/09	12:00:00 AM
arcinfo_2min.html	9 KB	10/16/07	12:00:00 AM
global_topo_1min_V13.1.kmz	73 KB	9/21/10	8:32:00 AM
global_topo_1min_V13.1_terra.kmz	68 KB	9/21/10	8:31:00 AM
gmt_examples		8/27/07	12:00:00 AM
matlab		8/15/08	12:00:00 AM
survey_tool.zip	307 KB	1/12/10	12:00:00 AM
topo_11.1.img	729000 KB	9/16/08	12:00:00 AM
topo_11.1.img.ers	1 KB	9/16/08	12:00:00 AM
topo_12.1.img	729000 KB	9/3/09	12:00:00 AM
topo_13.1.img	729000 KB	8/28/10	4:10:00 PM

Figure 10. Global topography models available on the SIO website.

2.3 Generic Mapping Tools

GMT (Generic Mapping Tools) (Wessel and Smith, 1998) is a collection of open source mathematical and mapping routines for use on gridded data sets, data series, and arbitrarily located data. The GMT package is available for download from the University of Hawaii website (http://gmt.soest.hawaii.edu/) (see Figure 11). We utilized GMT routines for all of our data analyses and mapping, and the specific routine command lines that we used are listed in Appendix B. Software packages such as MATLAB, IMSL, ArcGIS and others may also provide similar mathematical and mapping capabilities.

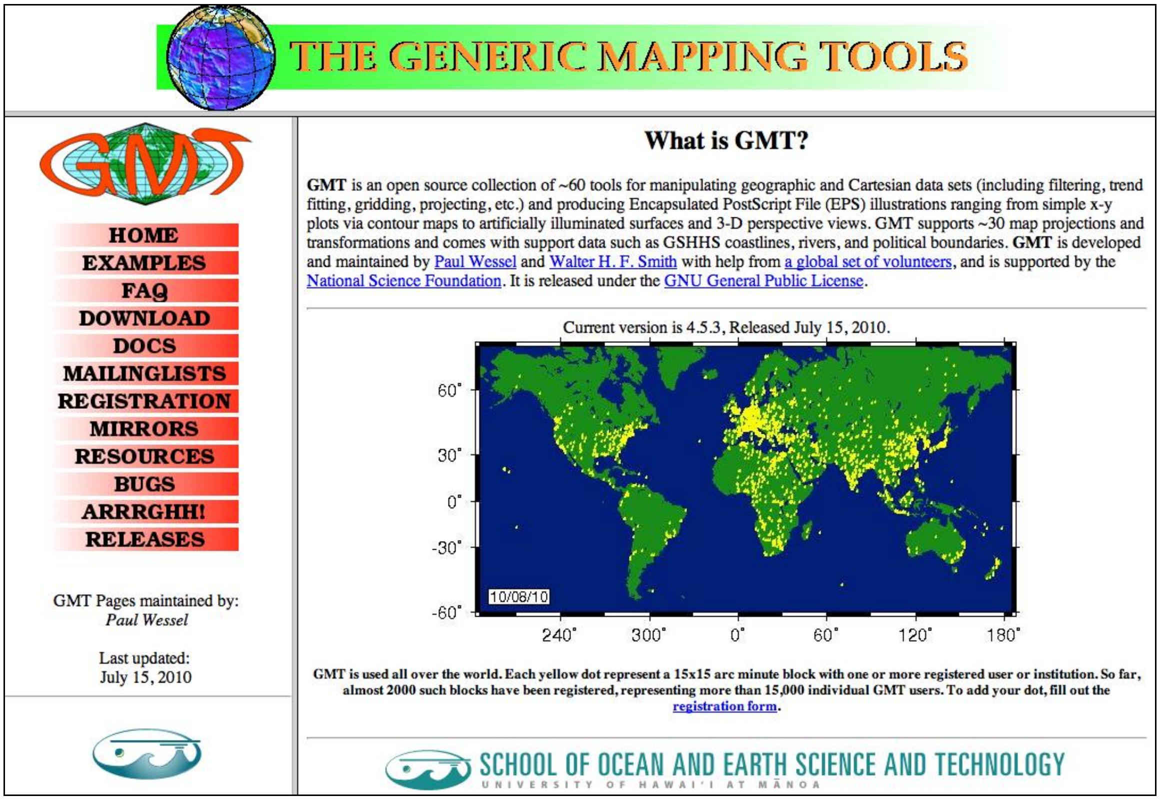


Figure 11. University of Hawaii website for GMT.

3. Data Preparation

Data need to be prepared for subsequent analyses. Multibeam *xyz* ping files which can contain millions of individual points need to be gridded onto both fine-scale grids (6 arc-second spacing) and grids that match the spacing of bathymetry models (1- or 2-minute spacing). It is also necessary to create a grid of distance from sounding controls that are encoded in the bathymetric models.

3.1 Gridding Multibeam Data

A grid can be formed from the individual multibeam *xyz* points downloaded from the JAMSTEC website. Because there are so many data points it is advantageous to first take their block average using GMT routine "blockmedian," calculating the median *z* (at the *x, y* location of the median *z*) for every non-empty grid cell on a 6 arc-second mesh. The next step is to use GMT routine "surface," an adjustable tension continuous curvature surface gridding algorithm, to form a grid at 6 arc-second spacing in longitude and latitude from the median depths. Routine "grdmask" is then used to create a mask that is applied to the grid using "grdmath" so that it holds values only in cells that contain one or more of the original *xyz* points. In Appendix B.2 we list the GMT routines used to create the grid from KR05-01 *xyz* multibeam points that is shown in Figure 12. In Appendix B.3 we list the GMT routines used to produce Figure 12.

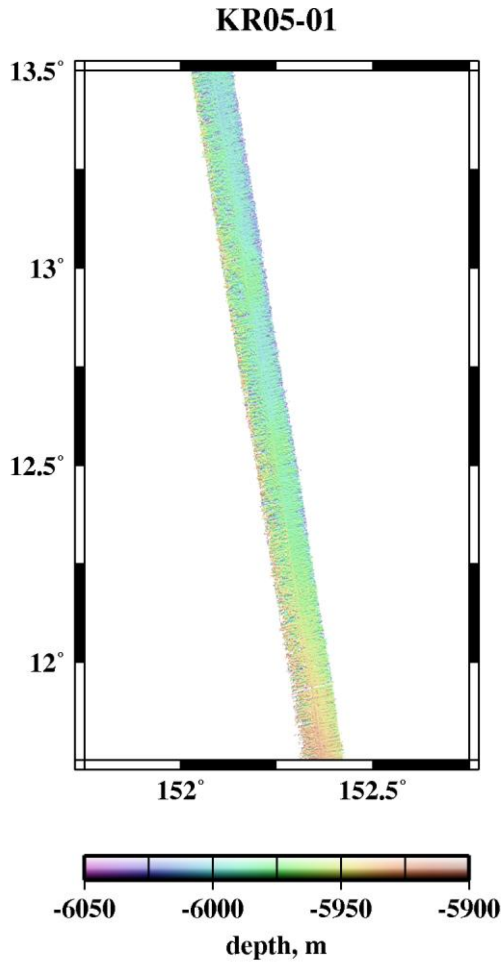


Figure 12. Color shaded-relief image of gridded KR05-01 multibeam points.

3.2 Create a Grid of Distance from Control

Encoded in the Smith and Sandwell (1997) altimetric bathymetry model (e.g., topo_12.1.img) is information on which grid cells contained acoustic echo sounding data to constrain the solution. At grid points constrained by ship measurements, the depth value is the median of all soundings nearest the grid point, rounded to the nearest odd integer meter. At grid points estimated from satellite gravity, the depth value is rounded to the nearest even integer meter. W. H. F. Smith wrote a computer program (Appendix C) that, for each grid cell, searches the neighborhood for the nearest control point and calculates the distance to it, writing the output to a "distance img" grid file. Depending on the user's computer system architecture (i.e., big- or little- endian), it may be necessary to swap adjacent bytes of the Sandwell/Smith "img format" file both prior to and after running Smith's computer program. GMT routine "img2grd" can then used on the byte-swapped "distance img" file to retrieve a netCDF grid of distance from control in a selected area. The command lines used to produce the "distance img" file are listed in Appendix B.4, and the GMT routines used to make the left panel (V12.1) of Figure 13 are listed Appendix B.5.

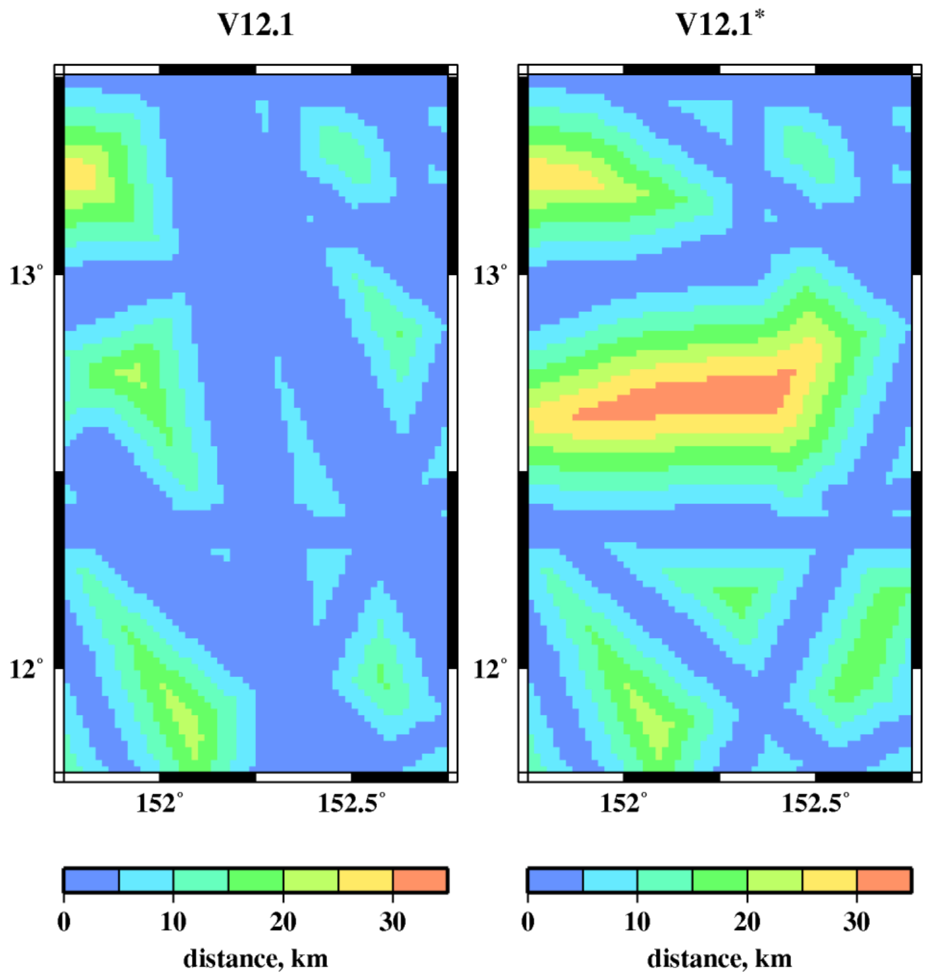


Figure 13. Maps of "distance from control" grids corresponding to bathymetry model versions 12.1 and 12.1*. JAMSTEC data were withheld in V12.1* for testing purposes.

4. Bathymetry Model Errors

Our method of assessing errors in bathymetric models is based on comparing the model depths to JAMSTEC multibeam "ground truth" data that were not available when the model was prepared, or were withheld for testing purposes. This method can be employed on local, regional, and near-global scales, each revealing different aspects of the errors.

4.1 Local Errors

To illustrate our method used locally, we calculate the differences between *xyz* multibeam depth points from KR05-01 and a special version of bathymetry model 12.1 that was constructed without JAMSTEC data (see V12.1*, Figure 14), in a small study area (the GMT routines used to make the left panel in Figure 14 are listed in Appendix B.6). The *xyz* multibeam data were first projected into the Mercator coordinates used in the bathymetry model via GMT routine "mapproject," and then routine "grdtrack" was used to interpolate corresponding bathymetric model depths to the *xy* locations of the multibeam points. The "errors" are the differences between the KR05-01 multibeam depths and special bathymetric model version 12.1* depths created without JAMSTEC data.

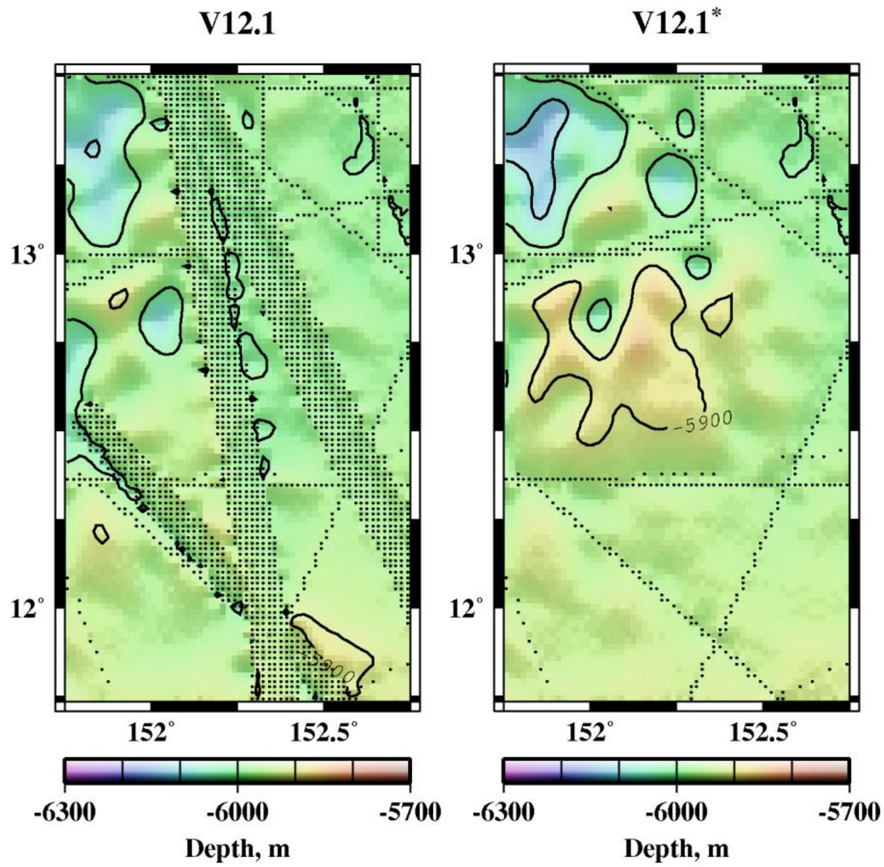


Figure 14. Bathymetry model versions 12.1 (topo_12.1.img) and 12.1* (JAMSTEC data withheld). Black dots are grids cells constrained by ship soundings.

We plot the absolute value of these errors against distance from control in the right panel of Figure 15. We also plot the errors for version 11.1 (JAMSTEC swath KR05-01 was not available when older version 11.1 was constructed). The amplitude of the errors can be quantified, and inspected for any relationship to distance from control: version 11.1 (left panel) demonstrates a strong relationship, and the errors are larger in amplitude. Bathymetry version 12.1 has smaller errors than previous versions because problems in the prediction algorithm that were identified by Marks et al. (2010) were mitigated. The GMT routines used to calculate the errors and plot them in Figure 15 (left panel) are listed in Appendix B.7.

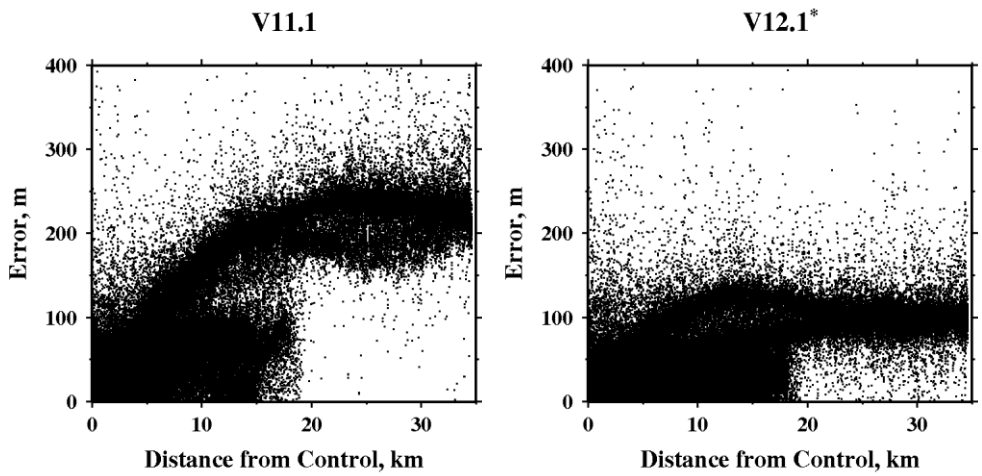


Figure 15. Errors are the absolute value of the differences between KR05-01 multibeam depths and corresponding bathymetry model versions 11.1 and 12.1* depths, plotted against distance from the nearest sounding constraining the bathymetry model.

The distribution of depth differences may be plotted in a histogram (Figure 16). For both versions in this example, the depth differences are generally negative (bathymetry model is shallower than multibeam depths) and they are not normally distributed; and version 11.1 has larger depth differences. The GMT routines used to plot the errors in histogram form as shown in Figure 16 (left panel) are listed in Appendix B.8.

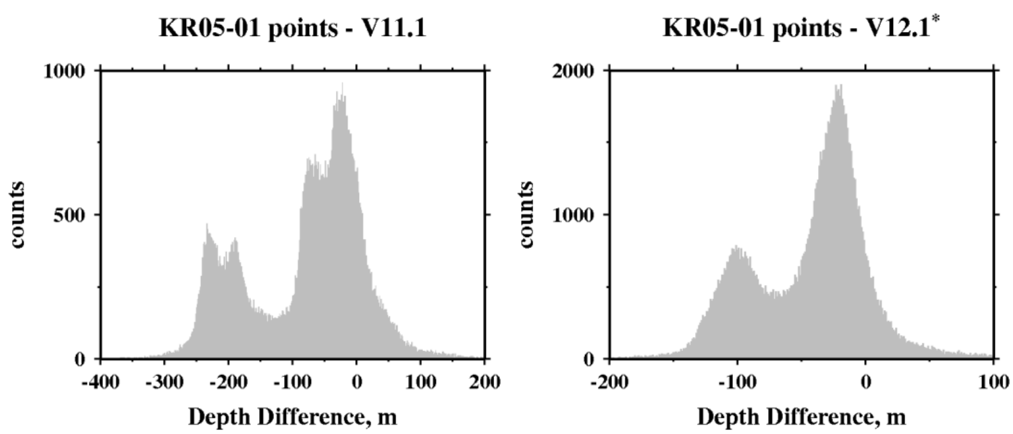


Figure 16. Histogram of the differences between KR05-01 multibeam depths and versions 11.1 and 12.1* depths.

4.2 Regional Errors

Long-wavelength errors can be evaluated on a regional scale, which we demonstrate by using our error assessment method on a large area in the Pacific Ocean. In the left panel of Figure 17, we show the depth differences between Smith and Sandwell bathymetry model version 11.1 and a companion version 11.1 that had JAMSTEC multibeam data withheld. The depth differences are the "errors" and they are colored to enhance their visualization. Version 11.1 without JAMSTEC data contains long-wavelength errors in depth that show up when compared to the better-constrained version 11.1 that incorporated JAMSTEC data. In this example, viewing the errors in map form made evident a long-wavelength problem in the bathymetry model that was not obvious otherwise. As a result, it was possible to mitigate the problem in bathymetry version 12.1 (right panel in Figure 17). Version 12.1 in the right panel has smaller errors and they do not extend far from the JAMSTEC swaths.

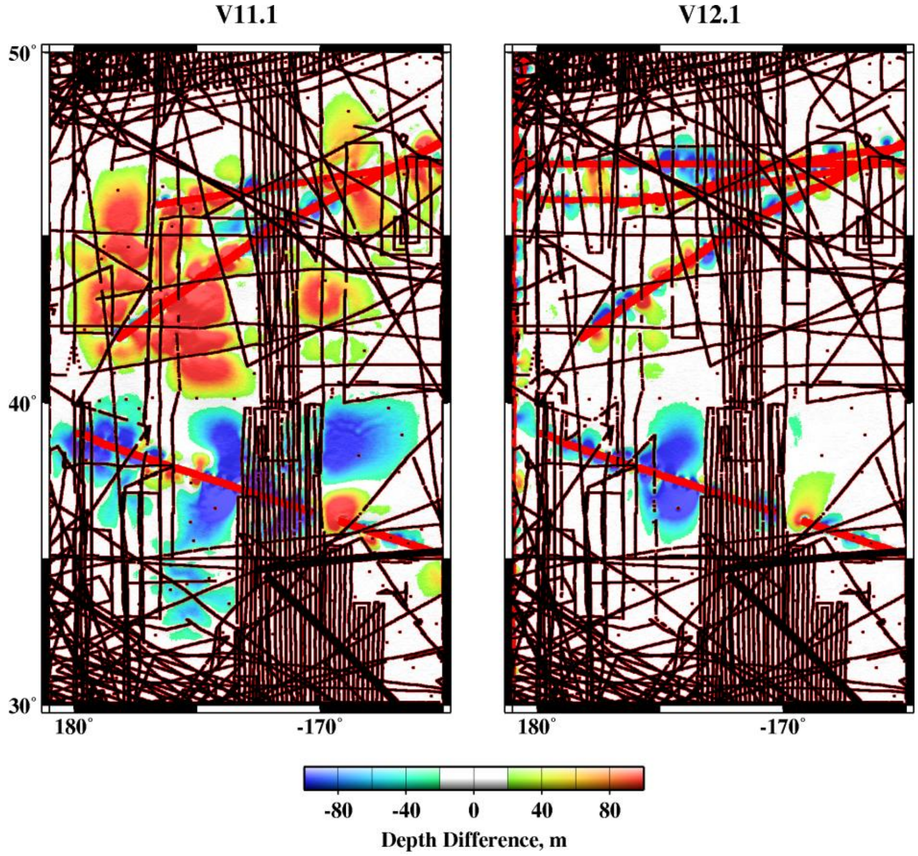


Figure 17. Depth differences ("errors") are between version 11.1 with JAMSTEC data (red dots are controls), and version 11.1 with JAMSTEC withheld (black dots are controls) (left panel). Right panel is same as left except using version 12.1. The problem causing long-wavelength errors in version 11.1 has been mostly corrected in 12.1.

The GMT routines used to produce the right panel in Figure 17 are listed in Appendix B.9.

4.3 Global Errors

Although JAMSTEC data are concentrated in the western Pacific Ocean, they do cover parts of the eastern Indian Ocean, the southern and eastern Pacific Ocean, the Arctic Ocean, and there are cruises that nearly circumnavigate the globe along about 15° or 30° S latitudes. For purposes of this report, we consider this coverage to be near-global enough that we can demonstrate how our error assessment method may be applied globally to bathymetry models.

To obtain statistical measures of global errors, we compared bathymetry models built without JAMSTEC multibeam data to JAMSTEC data incorporated into bathymetry models. The models are on 1-minute Mercator grids and the median of all available soundings in a grid cell is the value of the cell. Accordingly, the median of all JAMSTEC multibeam data falling within grid cells in V12.1 are subtracted from corresponding grid cells in model V12.1* that had JAMSTEC withheld. The differences in depth are the "errors" in the bathymetry model. These errors can be grouped statistically to confirm that overall errors are being reduced in successive bathymetry model versions. The specialized computer program (written by W. H. F. Smith) that we used to calculate these differences directly from bathymetry model "img" files is listed in Appendix D.

Additional information may be gleaned from the depths. Plotting JAMSTEC (measured) depths against model (predicted) depths shows there is no systematic variation of errors with depth (see Figure 18). The GMT routines used to make Figure 18 are listed in Appendix B.10.

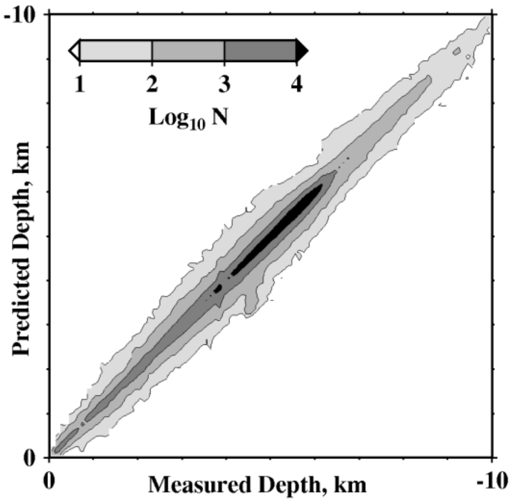


Figure 18. Version 12.1 JAMSTEC controlled (measured) depths plotted against predicted depths (V12.1*, JAMSTEC withheld). The number of points (N) are contoured.

5. Spectral Analyses

Bathymetry (and gravity) data series may also be compared in the frequency (or wavelength) domain. For this example we produce profiles along JAMSTEC track KR05-01 that traverses rough seafloor north of the local study area shown in Figure 14. To sample both bathymetry and gravity at the same *xy* locations, we first downloaded gravity data from the JAMSTEC website (see section 2.1.2). GMT routine "grdtrack" was then used to sample the corresponding 6 arc-second KR05-01 multibeam grid (see section 3.1) at the points were the gravity measurements were made. Next, we used a computer program written by W. H. F. Smith to compute along-track distance from the data record longitude and latitude pairs (Appendix E). Last, the longitudes and latitudes of the gravity points were projected into the Mercator coordinates used in the bathymetry models via GMT routine "mapproject," and "grdtrack" was used to sample bathymetry model versions 11.1 and 12.1 at the gravity measurement locations as well. The resulting profiles are shown in Figure 19.

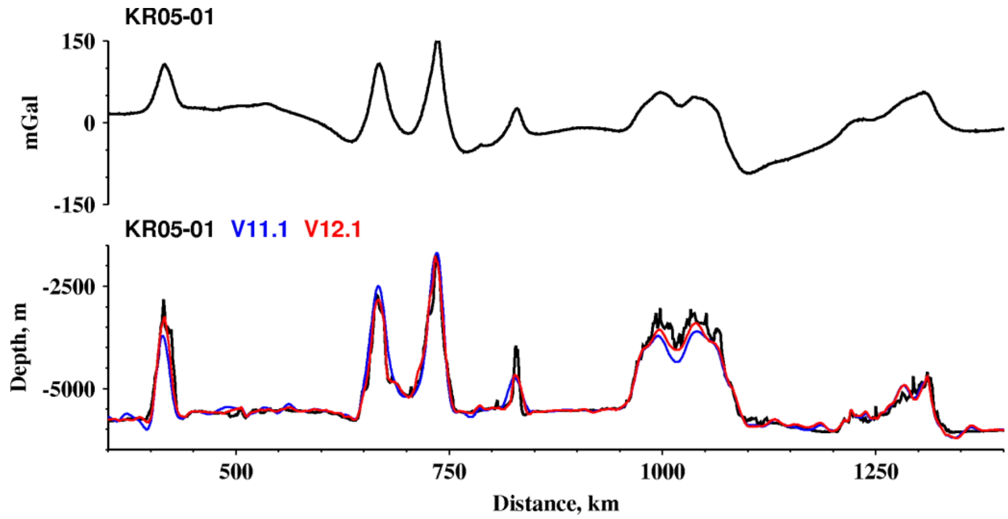


Figure 19. Profiles from gridded JAMSTEC KR05-01 multibeam data, bathymetry model versions 11.1 and 12.1, and KR05-01 along-track gravity measurements.

To form the spectra, the data profiles are resampled to a uniform 1-km spacing using GMT routine “sample1d.” Then the power spectral density (PSD) and cross-spectral coherency are computed using GMT routine “spectrum1d” (Figure 20).

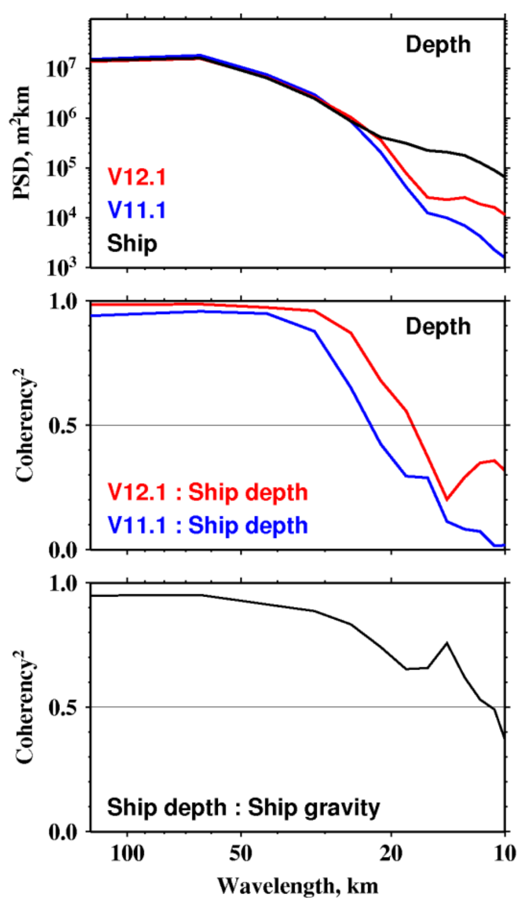


Figure 20. Depth power spectral density (PSD) (top panel), cross-spectral depth coherency (middle panel), and coherency between ship depth and gravity (bottom panel), from profiles shown in Figure 19.

In our example, the power spectral densities (top panel) show that ship bathymetry has more power than bathymetry model estimates at shorter wavelengths. For cross-spectral coherency, perfect correlation has a $coherency^2$ of 1, and no correlation is 0. The coherency plots show that bathymetry version 12.1 is better than version 11.1 (middle panel), but still altimeter improvements could yield higher resolution at shorter wavelengths (bottom panel). This exercise demonstrates the usefulness of spectral analyses in comparing data series.

The GMT routines used to compute PSD and coherency are in Appendix B.11, and those used to make the corresponding plots as in Figure 20 are listed in Appendix B.12.

6. Evaluating Gridding Algorithms

Our method is effective for assessing errors in bathymetry models. The method can also be used to assess errors in any local or regional depth grids where there are multibeam or other "ground truth" depths available to compare the grids against. The "ground truth" data must not be incorporated into the depth grid for the technique to work, and there needs to be an area with a large gap between control points for testing. This method can also be used to evaluate different gridding techniques. A successful gridding algorithm will have smaller errors when compared to the ground truth data.

We demonstrate the use of the error assessment method to evaluate different gridding algorithms. For this example, we select a small study area encompassing several seamounts (left panel in Figure 21) that has large gaps between control points and is traversed by several swaths of JAMSTEC multibeam data that can serve as "ground truth." This study area is also convenient because we can use Smith's computer program (Appendix C) to obtain distance to control (right panel in Figure 21) from bathymetry version 12.1[*].

Using the computer program in Appendix C is appropriate when calculating distance-to-control directly from a bathymetry model "img" file. However distance-to-control may be calculated for any control points using the GMT routine "grdmath" with the PDIST operator. Appendix B.13 lists the steps for calculating distance-to-control from any user-supplied controls points. This will enable users to test any bathymetry grids for errors as a function of distance to control.

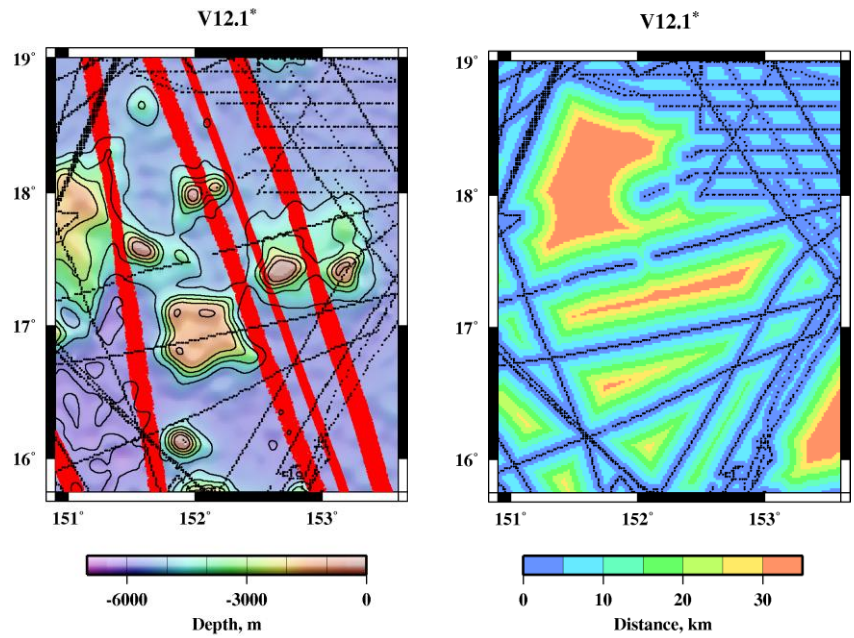


Figure 21. JAMSTEC multibeam swaths (red) plotted on bathymetry version 12.1* (left). Map of "distance to control" corresponding to bathymetry version 12.1* (right). Black dots are V12.1* grid cells constrained by ship soundings. V12.1* was created without JAMSTEC data.

We used depth values from the constrained grid cells from V12.1* in the study area as input into different gridding algorithms: GMT routine "surface," employed with the tension set to "0" and set to "1," and GMT routine "nearneighbor." Surface tension set to "0" gives the minimum curvature solution, and set to "1" gives a harmonic surface where maxima and minima are only possible at control points. "Nearneighbor" uses a nearest neighbor algorithm to assign an average value within a radius centered on a node. The GMT command lines used to produce the gridding solutions shown in Figure 22 are listed in Appendix B.14.

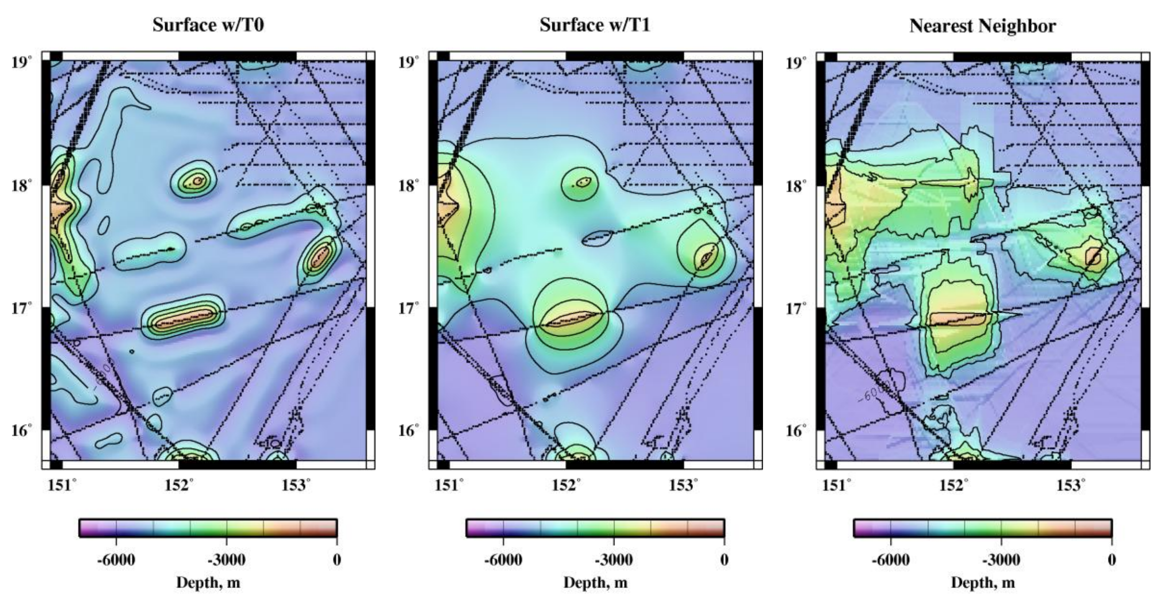


Figure 22. Results of gridding depths from V12.1* at constrained grid cells (black dots): GMT "surface" gridding routine with tension set to 0 (left) and set to 1 (middle), and "nearneighbor" gridding routine (right).

Each gridding algorithm produces a very different result. By comparing these results to the JAMSTEC multibeam "ground truth" data (red dots in Fig. 21), we can assess which algorithm produced the best match to the observed depths. Note that no interpolated solution detects seamounts in gaps between ship controls (Fig. 22). However altimetric bathymetry model V12.1* (Fig. 21) does detect seamounts in gaps between ship soundings. The model fills these gaps with depths estimated from satellite gravity, which reflects the underlying seafloor topography.

In Figure 23 we show the errors (i.e., the absolute value of the differences) between the JAMSTEC multibeam depths and depths produced by the different GMT gridding routines. In our example, GMT routine "surface" with a tension of 1 has the smallest errors.

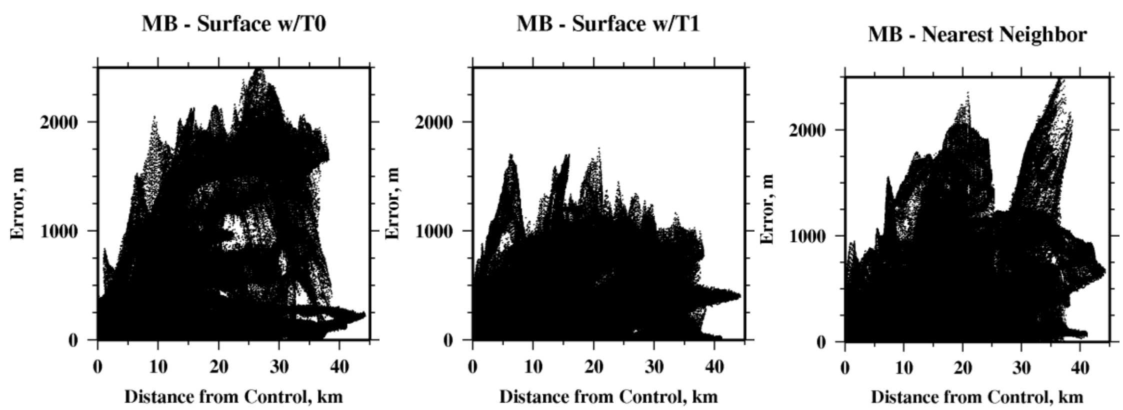


Figure 23. Errors are the absolute value of the differences between multibeam depths from swaths plotted in Figure 21 and gridded depths from GMT routines "surface" with tension set to 0 (left) and set to 1 (middle), and from GMT "nearneighbor" (right), plotted against distance from the nearest sounding control.

The errors can also be displayed in the form of a histogram (Figure 24). The error distribution from gridding with GMT routine "surface" with tension set to 0 shows the largest errors and they are generally negative (depths from "surface" grid are deeper than multibeam depths).

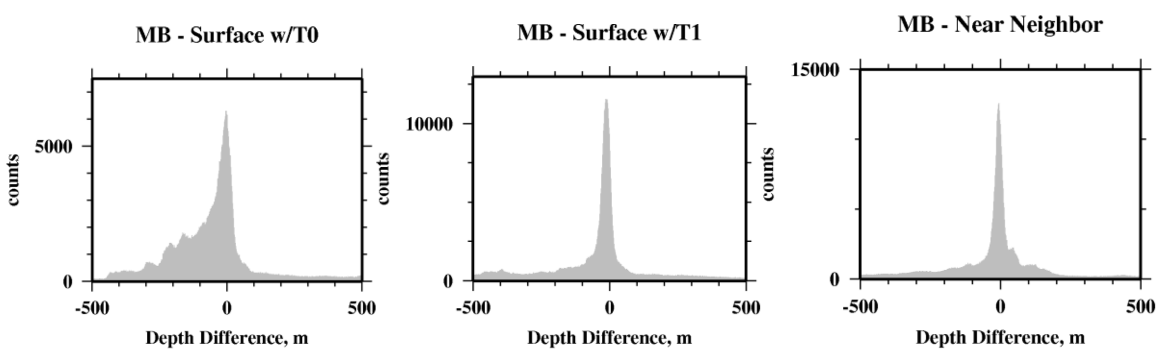


Figure 24. Histogram of the differences between multibeam depths from swaths plotted in Figure 21 and grids from GMT routines "surface" with tension set to 0 (left) and set to 1 (middle), and from GMT "nearneighbor" (right) depths.

It is also possible to map regional errors resulting from the different gridding algorithms. The interpolated grids (shown in Fig. 22) can be subtracted from a gridded solution that incorporated all the ship controls (in this example, bathymetry model V12.1). The pattern of regional depth differences (Fig. 25) can help in evaluating how well a gridding routine is doing.

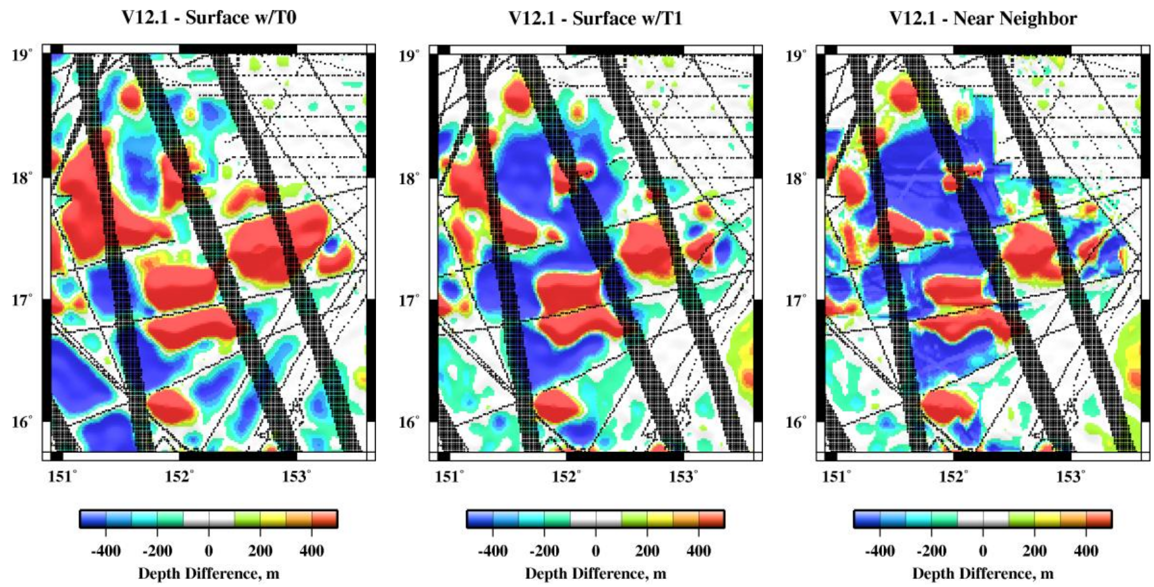


Figure 25. Regional depth differences between V12.1 and results from different GMT gridding routines (shown in Fig. 22). Black dots are constrained grid cells in V12.1.

7. Summary

Marks et al. (2010) developed a method of assessing local, regional, and global errors in the Smith and Sandwell (1997) altimetric bathymetry model and its updates. The method entails comparing model depths to high-quality JAMSTEC multibeam depths, with “errors” being the differences. Locally, the errors are plotted against distance to sounding control to assess their amplitude and view any correlation. Regionally, a special version of the bathymetry model that did not include JAMSTEC multibeam data is subtracted from the bathymetric model that incorporated them; the differences are plotted in map form using a color scale that enhances visualization of the regional-scale errors. Globally, statistical measures are obtained from the differences between the median depths of all JAMSTEC multibeam data that fall within a grid cell and the corresponding bathymetric model grid cell. Additionally, plotting global JAMSTEC depths against model depths can reveal any systematic variations.

The error assessment method may also be employed on any typical local or regional depth grids having multibeam or other “ground truth” depths available for comparison. Algorithms for interpolating bathymetric data points onto a grid can also be evaluated by assessing their respective errors when compared to “ground truth” depths. It is also useful to plot the errors against distance to control.

This report provides simple step-by-step instructions that a user can follow to 1) download freely available data and software from the internet, 2) grid bathymetric data points, 3) test interpolation algorithms used for gridding, 4) assess errors in altimetric bathymetry models, and

5) assess errors in their own gridded data sets. The appendixes show the GMT command lines complete with arguments that enable the user to duplicate the plots, maps, gridding, and error assessment techniques that are presented. Necessary computer software code is also included in the Appendixes.

8. Acknowledgements

Multibeam data used in this report were acquired during the KR05-01 cruise of R/V KAIREI, the MR01-K01 and MR06-01 cruises of R/V MIRAI, and the KY02-10 cruise of R/V KAIYO, Japan Agency for Marine-Earth Science and Technology. We thank JAMSTEC for making multibeam echo-sounding data freely available. We thank David Sandwell for providing bathymetry versions 11.1 and 12.1 constructed without JAMSTEC multibeam data for our testing purposes. Reviews by Rob Hare and Nastia Abramova improved this report.

9. References

Moritz, H. (1980): Geodetic Reference System 1980, Bull. Geod., *54*, pp. 395-405.

Marks, K.M., W. H. F. Smith, and D. T. Sandwell: (2010) Evolution of errors in the altimetric bathymetry model used by Google Earth and GEBCO. *Mar. Geophys. Res*., *31*, 223-238. doi:10.1007/s11001-010-9102-0.

Smith, W. H. F. and D. T. Sandwell (1997): Global sea floor topography from satellite altimetry and ship depth soundings. Sci., *277*, 1956-1962. doi:10.1126/science.277.5334.1956.

Wessel, P. and W. H. F. Smith (1998): New, improved version of Generic Mapping Tools released. EOS Trans. AGU, *79*, 59. doi:10.1029/98EO00426.

Appendix A: GRS80 Geodetic Reference System reduction

The total gravity (g) provided by JAMSTEC is reduced relative to the GRS80 Geodetic Reference System using this equation (Moritz, 1980):

$$g_{\text{reduced}} = g - 978032.7\,(1 + 0.0053024 \sin^2 \phi - 0.0000058 \sin^2 2\phi)\ \text{mGal}$$

where ϕ is latitude

Appendix B: Sample GMT Routine Command Lines

B.1

```
img2grd topo_12.1.img -R151.75/152.75/11.75/13.5 –Gstudy_area.grd -M -m1 -T1 -D -V
```

B.2

```
blockmedian KR05-01.xyz -R151.75/152.75/11.75/13.5 -I6c -Q > KR05-01.blockmedian.6c.xyz
surface KR05-01.blockmedian.6c.xyz -R151.75/152.75/11.75/13.5 -I6c -T0.25 -GKR05-01.surf.6c.grd
grdmask KR05-01.xyz -GKR05-01.mask.grd -R151.75/152.75/11.75/13.5 -I6c -NNaN/1/1 -S6c
grdmath KR05-01.surf.6c.grd KR05-01.mask.grd OR = KR05-01.surf.6c.nan.grd
```

B.3

```
grdgradient KR05-01.surf.6c.nan.grd -A0 -Ne0.2 –Ggradient.grd
grdimage KR05-01.surf.6c.nan.grd –Igradient.grd -Cmb.cpt -Jm2 -K > Fig12.ps
psbasemap -R151.75/152.75/11.75/13.5 -Jm2 -Ba.5f.25:."KR05-01":WeSn -O -K >> Fig12.ps
psscale -D1/-.5/2/.125h -Cmb.cpt -Ba50g25:"depth, m": -I -N300 -O >> Fig12.ps
```

B.4

```
gcc img_d2c.c -lm -o img_d2c
dd if=topo_12.1.img conv=swab of=swap.topo_12.1.img
img_d2c swap.topo_12.1.img swap.dist_12.1.img
dd if=swap.dist_12.1.img conv=swab of=dist_12.1.img
```

B.5

```
img2grd dist_12.1.img -R151.75/152.75/11.75/13.5 -Gdist_12.1.sub.grd -M -T1 -S.01 -m1 -D
grdimage dist_12.1.sub.grd -Cdist.cpt -Jm2 -K > Fig13.ps
psbasemap -R151.75/152.75/11.7337766065/13.5064749466 -Jm2 -Ba.5f.25:."V12.1":WeSn -O -K >> Fig13.ps
psscale -D1/-.5/2/.125h -Cdist.cpt -Ba10g5:"distance, km": -N300 -O >> Fig13.ps
```

B.6

```
img2grd topo_12.1.img -R151.75/152.75/11.75/13.5 -Gstudy.area.constraint.grd -M -m1 -T3 -D -V
grd2xyz study.area.constraint.grd > study.area.constraint.xyz
grdgradient study_area.grd -A0 -Ne0.2 –Ggradient.grd
grdimage study_area.grd –Igradient.grd -Ctopo.cpt -Jx1.8 -K > Fig14.ps
psbasemap -R151.75/152.75/11.7337766065/13.5064749466 -Jm1.8 -Ba.5f.25/a1f.25:.V12.1:WeSn
        -O -K >> Fig14.ps
grdcontour -A100+f10+s8 study_area.grd -R0/1/0/1.81667 -Jx1.8 -C100 -W3 -K -O >> Fig14.ps
awk '($3==1){print $0}' study.area.constraint.xyz | psxy -Sc.01 -G0 -R0/1/0/1.81667 -Jx1.8 -O -K>> Fig14.ps
psscale -D.9/-.3/1.8/.125h -Ctopo.cpt -Ba300g100:"Depth, m": -I -N300 -O>> Fig14.ps
```

B.7

```
img2grd topo_11.1.img -R151.75/152.75/11.75/13.5 -Gtopo_11.1.sub.grd -M -m1 -T1 -D -V
img2grd dist_11.1.img -R151.75/152.75/11.75/13.5 -Gdist_11.1.sub.grd -T1 -M -S.01 -m1 -D
mapproject KR05-01.xyz -Jm1 -R151.75/152.75/11.7337766065/13.5064749466 | grdtrack
-Gtopo_11.1.sub.grd | grdtrack -Gdist_11.1.sub.grd | awk '{print $5, sqrt(($3-$4)*($3-$4))}' | psxy -Sp -R0/35/0/400 -
JX2/1.75 -Ba10f5:"Distance from Control, km":/a100f50:"Error, m"::.V12.1@+*@+:WeSn > Fig15.ps
```

B.8

```
mapproject KR05-01.xyz -Jm1 -R151.75/152.75/11.7337766065/13.5064749466 | grdtrack -Gtopo_11.1.sub.grd | awk
'{print $3-$4}' | pshistogram -F -JX2/1.5 -W1 -G190 -R-400/200/0/1000
-Ba100f50:"Depth Difference, m":/a500f250:counts::."KR05-01 points - V11.1":WeSn > Fig16.ps
```

B.9

```
grdmath topo_12.1.pac.grd topo_12.1.nojamstec.pac.grd - = diff.topo.12.1.grd
grdgradient diff.topo.12.1.grd -A0 -Ne0.2 –Ggrad.grd
grdimage diff.topo.12.1.grd –Igrad.grd -Cdiff.cpt -Jx.16 -K > Fig17.ps
pscoast -G175 -R179/195/29.9945810754/50.0056468984 -Jm.16 -W2 -Df -Ba10f5:."V12.1":weSn -O
        -K>> Fig17.ps
awk '($3==1){print $0}' diff.topo.12.1.controls.xyz | psxy -R0/16/0/26.45 -Jx.16 -Sc.02 -G255/0/0 -K
        -O >> Fig17.ps
awk '($3==1){print $0}' diff.topo.12.1.nojamstec.controls.xyz | psxy -R0/16/0/26.45 -Jx.16 -Sc.01 -G0
        -K -O >> Fig17.ps
psscale -D1.3/-.4/2.2/.15h -Cdiff.cpt -Ba40g20:"Depth Difference, m": -I -N300 -O >> Fig17.ps
```

B.10

```
awk '{print $3, $4, 1}' img_comp.c.output | blockmean -R-10000/0/-10000/0 -I100 -Sw -F
        | awk '{print $1, $2, log($3)/2.30258093}' | xyz2grd -R-10000/0/-10000/0 -I100 -F -Gsum.log.grd
grdview -P -X2.5 -Y4 sum.log.grd -Qs -Wc -JX-2.55 -Clog10n.cpt -K > Fig18.ps
psbasemap -JX-2.55 -Ba10f1/a10f1WeSn -R-10/0/-10/0 -K -O>> Fig18.ps
pstext -X-2.5 -Y-4 -R0/8.5/0/11 -Jx1 -O -K << STOP >> Fig18.ps
2.38    5.2   10    90    5    CB  Predicted Depth, km
3.7     3.8   10    0     5    CB  Measured Depth, km
STOP
psscale -D3.3/6.4/1.25/.12h -E -Clog10n.cpt -B1g1:"Log@-10@- N": -N300 -O >> Fig18.ps
```

B.11

```
cdist < KR05-01.xyg > KR05-01.xygd
awk '{print $4,$3}' KR05-01.xygd | sample1d -I1 -Fl > KR05-01.1km.d.g
grdtrack KR05-01.xygd -m -GKR05-01.surf.6c.nan.grd -S | awk '{print $4,$5}' | sample1d -I1 -Fl >
        KR05-01.1km.d.mb
mapproject KR05-01.xygd -Jm1 -R145/155/3.99675470104/25.0000723601 | grdtrack -m
        -Gtopo_12.1.nojamstec.grd | mapproject -Jm1 -I -R145/155/3.99675470104/25.0000723601 | awk
'{print $4,$5}' | sample1d -I1 -Fl > KR05-01.1km.d.v12.1
mapproject KR05-01.xygd -Jm1 -R145/155/3.99675470104/25.0000723601 | grdtrack -m
        -Gtopo_11.1.grd | mapproject -Jm1 -I -R145/155/3.99675470104/25.0000723601 | awk '{print $4,$5}'
        | sample1d -I1 -Fl > KR05-01.1km.d.v11.1
paste KR05-01.1km.d.mb KR05-01.1km.d.g | awk '{print $2,$4}' | spectrum1d -D1 -S128 -W -Co
        -NKR05-01.mb.g.coh
paste KR05-01.1km.d.mb KR05-01.1km.d.v12.1 | awk '{print $2,$4}' | spectrum1d -D1 -S128 -W -Co
        -NKR05-01.mb.v12.1.coh
paste KR05-01.1km.d.mb KR05-01.1km.d.v11.1 | awk '{print $2,$4}' | spectrum1d -D1 -S128 -W -Co
        -NKR05-01.mb.v11.1.coh
awk '{print $2}' KR05-01.1km.d.mb | spectrum1d -D1 -S128 -W -NKR05-01.mb.xpower
awk '{print $2}' KR05-01.1km.d.v12.1 | spectrum1d -D1 -S128 -W -NKR05-01.v12.1.xpower
awk '{print $2}' KR05-01.1km.d.v11.1 | spectrum1d -D1 -S128 -W -NKR05-01.v11.1.xpower
```

B.12
```
psxy -P -X2 -Y1.5 KR05-01.mb.g.coh -R10/125/0/1 -JX-2.5l/1.5 -Wthin -Ba2f3:"Wavelength,
        km":/a.5f.1g.5:"Coherency@+2@+":WeSn -K > Fig20.ps
psxy -Y1.7 KR05-01.mb.v12.1.coh -Wthick/255/0/0 -R10/125/0/1 -JX-2.5l/1.5
        -Ba2f3/a.5f.1g.5: "Coherency@+2@+":Wesn -K -O >> Fig20.ps
psxy KR05-01.mb.v11.1.coh -R -JX -Wthick/0/0/255 -O -K >> Fig20.ps
psxy -Y1.7 KR05-01.v11.1.xpower -Wthick/0/0/255 -R10/125/1000/100000000 -JX-2.5l/1.5l
        -Ba2f3:"Wavelength, km":/a1f3p:"PSD, m@+2@+km":Wesn -K -O >> Fig20.ps
psxy KR05-01.v12.1.xpower -Wthick/255/0/0 -R -JX -K -O >> Fig20.ps
psxy KR05-01.mb.xpower -Wthick/0/0/0 -R -JX -K -O >> Fig20.ps
pstext -Y-4.9 -X-2 -Jx1 -G0 -R0/8.5/0/11 -K -O <<END >> Fig20.ps
2.1   1.6    10    0    1    LB  Ship depth : Ship gravity
2.1   5.0    10    0    1    LB  Ship
3.9   6.2    10    0    1    LB  Depth
3.9   4.5    10    0    1    LB  Depth
END
pstext -Jx -G255/0/0 -R -K -O <<END >> Fig20.ps
2.1   3.5    10    0    1    LB  V12.1 : Ship depth
2.1   5.4    10    0    1    LB  V12.1
END
pstext -Jx -G0/0/255 -R -O <<END >> Fig20.ps
2.1   3.3    10    0    1    LB  V11.1 : Ship depth
2.1   5.2    10    0    1    LB  V11.1
END
```

B.13
```
grdmath controlpoints.xy –Rminlon/maxlon/minlat/maxlat –Igrdspacing –fg PDIST = distance.grd
```

B.14
```
img2grd -R149/154/14/19 topo_12.1*.img -Gtopo_12.1*.controls.grd -T2 -m1 -D
surface topo_12.1.nojamstec.controls.xyz -R0/2.7/0/3.41666666667 -I1m -T0 -Gsurface.t0.grd
surface topo_12.1.nojamstec.controls.xyz -R0/2.7/0/3.41666666667 -I1m -T1 -Gsurface.t1.grd
nearneighbor topo_12.1.nojamstec.controls.xyz -R0/2.7/0/3.41666666667 -N4/1 -S100k -I1m
        -Gnearneighor.1m.grd
```

Appendix C: Distance from Control Program

```
/* img_dist_to_control.c

Open an img file which has even/odd encoding of control points,
and compute an img file showing distance to nearest control
point.
In accord with img file standards, two byte signed integers
are used.  Therefore the maximum allowed value is 32767.
Distances are expressed as km times 100, i.e., the max
distance found by this program is 327.67 km.

WHF Smith, 14 Jul 2008
*/

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <sys/types.h>
#include <sys/stat.h>
/* #include <float.h>
#include <limits.h>  */

struct SEARCH_ME {
        double  r;
        int     i;
        int     j;
};

int main (int argc, char **argv) {

        double  radius, y, t, d;
        int     nx, ny, nt;    /* nx, ny and n total cells in the img file  */
        int     i, j, k, ii, jj, kk, nhcon, ksearch, maxsearch, not_found;
        short int *h, hr;
        FILE    *fp;
        struct SEARCH_ME *search_list;
        int     check_file_size (char *filename, int *nx, int *ny);
```

```
	int	compare_data_r (struct SEARCH_ME *x, struct SEARCH_ME *y);
	short int	get_hr (double r_in_km);

	/* Get the file names from the argument list  */
	if (argc != 3) {
		fprintf (stderr, "usage:  img_d2c <input_topo_file.img> 
<output_dist_file.img>\n");
		fprintf (stderr, "\tin which the second argument is an img format topo file with 
even/odd control encoding.\n");
		fprintf (stderr, "\tThe output file will be an img format file with distance to 
nearest control point,\n");
		fprintf (stderr, "\texpressed as integer values of (km * 100), ranging from 0.00 
to 327.67 km distances.\n");
		exit (EXIT_FAILURE);
	}

	/* Figure out whether we have a 1 or 2 minute file, to 72 or 81 latitude, from the file 
size.  */

	if (check_file_size (argv[1], &nx, &ny) ) exit (EXIT_FAILURE);
	nt = nx * ny;  /* Total number of elements in file */

	/* Build a search list */

	radius = (nx/2) / M_PI;		/* Radius of img mercator projection  */
	y = (ny/2) - 0.5;	/* Distance in pixels from Equator to center of cell in top row  */
	t = 2.0 * atan ( exp (y / radius) ) - M_PI_2;		/* latitude in radians of center of 
top row of cells  */
	d = 40030.0 * cos(t) / nx;	/* Width and height of a pixel, in km, at this lat.  */
	nhcon = (int) floor (327.67/d);		/* Largest possible search box is 2*nhcon+1 points 
square  */
	fprintf (stderr, "img_d2c:  %s is %d by %d; the largest search halfwidth is %d at %g 
latitude.\n",
		argv[1], nx, ny, nhcon, t * (180.0/M_PI) );

	maxsearch = (nhcon+1)*(nhcon+1);
	if ( (search_list = (struct SEARCH_ME *) malloc ((size_t)(maxsearch * sizeof (struct 
SEARCH_ME)))) == NULL) {
		fprintf (stderr, "img_d2c:  FATAL ERROR cannot allocate %d bytes for search 
table.\n", maxsearch * sizeof(struct SEARCH_ME));
		exit (EXIT_FAILURE);
	}
	for (j = 0, k = 0; j <= nhcon; j++) {
		for (i = 0; i <= nhcon; i++, k++) {
			search_list[k].r = hypot ((double)i, (double)j);
			search_list[k].i = i;
			search_list[k].j = j;
		}
	}
	qsort ((void *)search_list, (size_t) maxsearch, sizeof (struct SEARCH_ME), 
compare_data_r);

	/* Allocate an array, open, read and close the file:  */

	if ( (h = malloc ((size_t)(2*nt)) ) == NULL) {
		fprintf (stderr, "img_d2c:  FATAL_ERROR cannot allocate %d bytes of memory to hold 
%s\n", 2*nt, argv[1]);
		exit (EXIT_FAILURE);
	}

	if ( (fp = fopen (argv[1], "r")) == NULL) {
		fprintf (stderr, "img_d2c:  FATAL_ERROR cannot open %s\n", argv[1]);
		exit (EXIT_FAILURE);
	}

	if ( (fread ((void *)h, (size_t)2, (size_t)nt, fp) ) != (size_t)nt) {
		fprintf (stderr, "img_d2c:  FATAL_ERROR cannot read %s\n", argv[1]);
		exit (EXIT_FAILURE);
	}

	fclose (fp);

	/* Run through the data.  If point is above sea level or odd, set to zero; else set to 
32767  */
	j = 0;
	for (i = 0; i < nt; i++) {
```

```
		h[i] =  (h[i] > 0 || ((abs((int)h[i]))%2) == 1) ? 0 : 32767;
		if (h[i] == 0) j++;
	}
	fprintf (stderr, "img_d2c:  %s has %d constrained and %d unconstrained points.\n",
argv[1], j, nt-j);


	for (j = 0, k = 0; j < ny; j++) {
		y = (ny/2) - (j + 0.5);		/* Distance in pixels from Equator to center of cell
of this row  */
		t = 2.0 * atan ( exp (y / radius) ) - M_PI_2;		/* latitude in radians  */
		d = 40030.0 * cos(t) / nx;	/* Width and height of a pixel, in km, at this lat.
*/

		for (i = 0; i < nx; i++, k++) {

			if (h[k] == 0) continue;

			ksearch = 1;
			not_found = 1;
			while (not_found && ksearch < maxsearch && (hr =
get_hr(search_list[ksearch].r * d)) < 32767) {
				jj = j - search_list[ksearch].j;
				if (jj >= 0) {

					ii = i - search_list[ksearch].i;
					if (ii < 0) ii += nx;
					kk = jj * nx + ii;
					if (h[kk] == 0) {
						not_found = 0;
						h[k] = hr;
					}

					if (not_found) {
						ii = i + search_list[ksearch].i;
						if (ii >= nx) ii -= nx;
						kk = jj * nx + ii;
						if (h[kk] == 0) {
							not_found = 0;
							h[k] = hr;
						}
					}
				}

				jj = j + search_list[ksearch].j;
				if (not_found && jj < ny) {

					ii = i - search_list[ksearch].i;
					if (ii < 0) ii += nx;
					kk = jj * nx + ii;
					if (h[kk] == 0) {
						not_found = 0;
						h[k] = hr;
					}

					if (not_found) {
						ii = i + search_list[ksearch].i;
						if (ii >= nx) ii -= nx;
						kk = jj * nx + ii;
						if (h[kk] == 0) {
							not_found = 0;
							h[k] = hr;
						}
					}
				}

				ksearch++;
			}
		}
		fprintf (stderr, "Finished row j = %d\n", j);
	}

	if ( (fp = fopen (argv[2], "w")) == NULL) {
		fprintf (stderr, "img_d2c:  FATAL_ERROR cannot open %s\n", argv[2]);
		exit (EXIT_FAILURE);
	}
```

```
	if ( (fwrite ((void *)h, (size_t)2, (size_t)nt, fp) ) != (size_t)nt) {
		fprintf (stderr, "img_d2c:  FATAL_ERROR cannot write %s\n", argv[2]);
		exit (EXIT_FAILURE);
	}

	fclose (fp);

	free ((void *)h);

	exit (EXIT_SUCCESS);
}

int	check_file_size (char *filename, int *nx, int *ny) {

	/* Use stat to get the file size.  There are four valid possibilities,
	corresponding to 1 or 2 minute files (nx must be 10800 or 21600) and
	whether the latitude range is 72.006 or 80.738.  If the file size is
	one of the four valid ones, set nx and ny and return zero; else return
	-1 to signal an error condition. */

	struct stat sb;		/* buffer for input file status, used to figure out its
size  */
	const off_t img721 = 2 * 21600 * 12672;		/* file size of a 1-minute file to 72
latitude */
	const off_t img722 = 2 * 10800 * 6336;		/* file size of a 2-minute file to 72
latitude */
	const off_t img811 = 2 * 21600 * 17280;		/* file size of a 1-minute file to 81
latitude */
	const off_t img812 = 2 * 10800 * 8640;		/* file size of a 2-minute file to 81
latitude */

	if (stat (filename, &sb) ) {
		fprintf (stderr, "img_d2c:  FATAL ERROR cannot stat filename %s\n", filename);
		return (-1);
	}

	if (sb.st_size == img721) {
		*nx = 21600;
		*ny = 12672;
		return (0);
	}
	if (sb.st_size == img722) {
		*nx = 10800;
		*ny = 6336;
		return (0);
	}
	if (sb.st_size == img811) {
		*nx = 21600;
		*ny = 17280;
		return (0);
	}
	if (sb.st_size ==  img812) {
		*nx = 10800;
		*ny = 8640;
		return (0);
	}

	fprintf (stderr, "img_d2c:  FATAL_ERROR %s has unrecognized file size of %d bytes.\n",
filename, (int)sb.st_size);
	return (-1);
}

int	compare_data_r (struct SEARCH_ME *x, struct SEARCH_ME *y) {

	if (x->r > y->r) {
		return (1);
	} else if (x->r < y->r) {
		return (-1);
	} else {
		return (0);
	}
}

short int	get_hr (double r_in_km) {
```

```
        double rr;
        short int retval;

        rr = rint (100.0 * r_in_km);
        retval = (rr >= 32767.0) ? 32767 : (short int) rr;
        return (retval);
}
```

Appendix D: Program to Compare Depths between Bathymetry Models

```
/* img_comp.c
6 July 2009, WHFS.

Specialized piece of code for looking at
two versions of prediction and extracting
differences.

Built to look at the "jamstec" and "no jamstec"
versions, extracting also the grav amplitude
file.

*/

#include "img_predict.h"

int main (int argc, char **argv) {

        char fname[3][128];
        short int h[3][21600];
        double  x, y, radius;
        int     i, j, nx, ny, n, err = 0;
        FILE *fp0, *fp1, *fp2 = NULL;

        fname[0][0] = fname[1][0] = fname[2][0] = '\0';

        for (i = 1; !err && i < argc; i++) {
                if (argv[i][0] == '-') {
                        switch (argv[i][1]) {
                                case 'A':
                                        /* Use gravity amplitude file  */
                                        strcpy (fname[2], &argv[i][2]);
                                        break;
                                case 'Y':
                                        /* This file has constraints marked  */
                                        strcpy (fname[0], &argv[i][2]);
                                        break;
                                case 'N':
                                        /* This file has constraints marked  */
                                        strcpy (fname[1], &argv[i][2]);
                                        break;
                                default:
                                        err++;
                                        break;
                        }
                }
                else {
                        err++;
                }
        }

        if (err || fname[0][0] == '\0' || fname[0][0] == '\0') {
                fprintf (stderr, "usage:  img_comp [-A<grav_amp.img>] -Y<yes.img> -N<no.img> >
output\n");
                fprintf (stderr, "\t-Y and -N are two versions of an img file, e.g. a topography
prediction.\n");
                fprintf (stderr, "\tThis program finds points where the Y file is odd and below
zero and the N file is even.\n");
                fprintf (stderr, "\tIt writes lon, lat, Yvalue, Nvalue [Avalue] to stdout.  If -A
file is given, that is written as well.\n");
                fprintf (stderr, "\tIntent is to allow inspection of how things have changed by
adding new data.\n");
                exit (EXIT_FAILURE);
        }
```

```
	if (check_imgfile_size (fname[0], &nx, &ny) ) {
		fprintf (stderr, "img_comp:  FATAL ERROR.  Cannot understand size of %s\n", 
fname[0]);
		exit (EXIT_FAILURE);
	}

	n = nx * ny;

	if (check_imgfile_size (fname[1], &nx, &ny) ) {
		fprintf (stderr, "img_comp:  FATAL ERROR.  Cannot understand size of %s\n", 
fname[1]);
		exit (EXIT_FAILURE);
	}

	if (nx*ny != n) {
		fprintf (stderr, "img_comp:  Files %s and %s are not the same size.\n", fname[0], 
fname[1]);
		exit (EXIT_FAILURE);
	}

	if (fname[2][0] != '\0') {
		if (check_imgfile_size (fname[2], &nx, &ny) ) {
			fprintf (stderr, "img_comp:  FATAL ERROR.  Cannot understand size of 
%s\n", fname[2]);
			exit (EXIT_FAILURE);
		}
		if (nx*ny != n) {
			fprintf (stderr, "img_comp:  Size of file %s does not match the 
others.\n", fname[2]);
			exit (EXIT_FAILURE);
		}
		if ( (fp2 = fopen (fname[2], "r") ) == NULL) {
			fprintf (stderr, "img_comp:  Cannot open %s\n", fname[2]);
			exit (EXIT_FAILURE);
		}
	}

	if ( (fp0 = fopen(fname[0], "r") ) == NULL) {
		fprintf (stderr, "img_comp:  FATAL ERROR.  Cannot open r %s\n", fname[0]);
		exit (-1);
	}
	if ( (fp1 = fopen(fname[1], "r") ) == NULL) {
		fprintf (stderr, "img_comp:  FATAL ERROR.  Cannot open r %s\n", fname[1]);
		exit (-1);
	}

	radius = (nx/2)/M_PI;
	for (j = 0; j < ny; j++) {
		y = ny/2 - 0.5 - j;
		y = 2.0 * atan(exp(y/radius)) - M_PI_2;
		y *= (180/M_PI);

		if ( (fread ( (void *)h[0], (size_t)2, (size_t)nx, fp0) ) != nx) {
			fprintf (stderr, "img_comp:  FATAL ERROR reading %s\n", fname[0]);
			exit (EXIT_FAILURE);
		}
		if ( (fread ( (void *)h[1], (size_t)2, (size_t)nx, fp1) ) != nx) {
			fprintf (stderr, "img_comp:  FATAL ERROR reading %s\n", fname[1]);
			exit (EXIT_FAILURE);
		}
		if (fp2 && (fread ( (void *)h[2], (size_t)2, (size_t)nx, fp2) ) != nx) {
			fprintf (stderr, "img_comp:  FATAL ERROR reading %s\n", fname[2]);
			exit (EXIT_FAILURE);
		}

		for (i = 0; i < nx; i++) {
			/* This would be faster done with a check of the odd bit,
				but I'm not sure if that works on machines of both endians,
				so I'll do it the slow way.  */
			if (h[0][i] < 0 && ((abs((int)h[0][i]))%2) == 1 && ((abs((int)h[1][i]))%2) 
== 0) {
				/* Write this point  */
				x = (nx == 21600) ? (i+0.5)/60 : (i+0.5)/30.0;
				if (fp2) {
```

```
                                        printf ("%10.6lf\t%10.6lf\t%d\t%d\t%d\n", x, y,
(int)h[0][i], (int)h[1][i], (int)h[2][i]);
                                }
                                else {
                                        printf ("%10.6lf\t%10.6lf\t%d\t%d\n", x, y, (int)h[0][i],
(int)h[1][i]);
                                }
                        }
                }
        }

        if (fp2) fclose (fp2);
        fclose (fp1);
        fclose (fp0);
        exit (EXIT_SUCCESS);
}
```

Appendix E: Along-Track Distance Program

```
/* cdist.c

read stdin and extract x y from first two tokens in each row
tack on cumulative distance as last column, write to stdout.

compile:  gcc cdist.c -lm -o cdist

WHF Smith, 19 feb 09
*/

#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

int     main (int argc, char **argv) {

        char    line[256];

        double  a = 6378.137;
        double  f = 1.0 / 298.257;
        double  d, esq, phi, cosphi, sinphi, omesqp, romesq, p, m, x1, x2, y1, y2, dx;
        int     k, n;

        if (argc > 1) {
                fprintf (stderr, "usage:  cdist < input > output\n");
                fprintf (stderr, "\tinput must have lon lat in first two columns\n");
                exit (EXIT_FAILURE);
        }

        d = 0.0;
        n = 0;
        esq = f * (2 - f);

        while (fgets(line, 256, stdin)) {
                k = strlen(line);
                if (line[k-1] == '\n') line[k-1] = '\0';
                k = sscanf (line, "%lf %lf", &x2, &y2);
                if (k != 2) {
                        fprintf (stderr, "cannot read line %d of input.\n", n);
                        exit (EXIT_FAILURE);
                }
                if (n) {
                        phi = 0.5 * (y2 + y1);
                        phi *= (M_PI/180.0);
                        cosphi = cos(phi);
                        sinphi = sin(phi);
                        omesqp = 1.0 - esq * sinphi * sinphi;
                        romesq = sqrt(omesqp);
                        p = (M_PI/180.0) * a / romesq;
                        m = p * (1.0 - esq) / omesqp;
                        p *= cosphi;
                        dx = x2 - x1;
                        while (dx < -180.0) dx += 360.0;
                        while (dx > 180.0)  dx -= 360.0;
```

```
                        d += hypot (dx*p, (y2-y1)*m);
                }
                printf ("%s\t%10.4lf\n", line, d);
                x1 = x2;
                y1 = y2;
                n++;
        }
        exit (EXIT_SUCCESS);
}
```

NESDIS 110 An Algorithm for Correction of Navigation Errors in AMSU-A Data. Seiichiro Kigawa and Michael P. Weinreb, December 2002.

NESDIS 111 An Algorithm for Correction of Lunar Contamination in AMSU-A Data. Seiichiro Kigawa and Tsan Mo, December 2002.

NESDIS 112 Sampling Errors of the Global Mean Sea Level Derived from Topex/Poseidon Altimetry. Chang-Kou Tai and Carl Wagner, December 2002.

NESDIS 113 Proceedings of the International GODAR Review Meeting: Abstracts. Sponsors: Intergovernmental Oceanographic Commission, U.S. National Oceanic and Atmospheric Administration, and the European Community, May 2003.

NESDIS 114 Satellite Rainfall Estimation Over South America: Evaluation of Two Major Events. Daniel A. Vila, Roderick A. Scofield, Robert J. Kuligowski, and J. Clay Davenport, May 2003.

NESDIS 115 Imager and Sounder Radiance and Product Validations for the GOES-12 Science Test. Donald W. Hillger, Timothy J. Schmit, and Jamie M. Daniels, September 2003.

NESDIS 116 Microwave Humidity Sounder Calibration Algorithm. Tsan Mo and Kenneth Jarva, October 2004.

NESDIS 117 Building Profile Plankton Databases for Climate and EcoSystem Research. Sydney Levitus, Satoshi Sato, Catherine Maillard, Nick Mikhailov, Pat Cadwell, Harry Dooley, June 2005.

NESDIS 118 Simultaneous Nadir Overpasses for NOAA-6 to NOAA-17 Satellites from 1980 and 2003 for the Intersatellite Calibration of Radiometers. Changyong Cao, Pubu Ciren, August 2005.

NESDIS 119 Calibration and Validation of NOAA 18 Instruments. Fuzhong Weng and Tsan Mo, December 2005.

NESDIS 120 The NOAA/NESDIS/ORA Windsat Calibration/Validation Collocation Database. Laurence Connor, February 2006.

NESDIS 121 Calibration of the Advanced Microwave Sounding Unit-A Radiometer for METOP-A. Tsan Mo, August 2006.

NESDIS 122 JCSDA Community Radiative Transfer Model (CRTM). Yong Han, Paul van Delst, Quanhua Liu, Fuzhong Weng, Banghua Yan, Russ Treadon, and John Derber, December 2005.

NESDIS 123 Comparing Two Sets of Noisy Measurements. Lawrence E. Flynn, April 2007.

NESDIS 124 Calibration of the Advanced Microwave Sounding Unit-A for NOAA-N'. Tsan Mo, September 2007.

NESDIS 125 The GOES-13 Science Test: Imager and Sounder Radiance and Product Validations. Donald W. Hillger, Timothy J. Schmit, September 2007

NESDIS 126 A QA/QC Manual of the Cooperative Summary of the Day Processing System. William E. Angel, January 2008.

NESDIS 127 The Easter Freeze of April 2007: A Climatological Perspective and Assessment of Impacts and Services. Ray Wolf, Jay Lawrimore, April 2008.

NESDIS 128 Influence of the ozone and water vapor on the GOES Aerosol and Smoke Product (GASP) retrieval. Hai Zhang, Raymond Hoff, Kevin McCann, Pubu Ciren, Shobha Kondragunta, and Ana Prados, May 2008.

NESDIS 129 Calibration and Validation of NOAA-19 Instruments. Tsan Mo and Fuzhong Weng, editors, July 2009.

NESDIS 130 Calibration of the Advanced Microwave Sounding Unit-A Radiometer for METOP-B. Tsan Mo, August 2010

NESDIS 131 The GOES-14 Science Test: Imager and Sounder Radiance and Product Validations. Donald W. Hillger and Timothy J. Schmit, August 2010.

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIONS

The National Oceanic and Atmospheric Administration **was established as part of the Department of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socioeconomic impact of natural and technological changes in the environment and to monitor and predict the state of the solid Earth, the oceans and their living resources, the atmosphere, and the space environment of the Earth.**

The major components of NOAA regularly produce various types of scientific and technical information in the following types of publications

PROFESSIONAL PAPERS – Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS – Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS – Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS – Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS – Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS – Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.

U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
National Environmental Satellite, Data, and Information Service

Washington, D.C. 20233