Naval Research Laboratory
Stennis Space Center, MS 39529-5004

NRL/FR/7322--97-9666

Complete Oceanographic and Atmospheric Master Library Documentation of the Shallow-Water Wave Refraction and Diffraction Model

Y. Larry Hsu

Ocean Dynamics and Prediction Branch
Oceanography Division

Andrew MacNaughton

Planning Systems Incorporated
Slidell, LA 70458

December 8, 1997

REPORT DOCUMENTATION PAGE

Form Approved
OBM No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December 8, 1997

3. REPORT TYPE AND DATES COVERED
Final

4. TITLE AND SUBTITLE
Complete Oceanographic and Atmospheric Master Library Documentation of the Shallow-Water Wave Refraction and Diffraction Model

5. FUNDING NUMBERS
Job Order No. 573-W100-08
Program Element No. 0603207N
Project No. X2008
Task No.
Accession No.

6. AUTHOR(S)
Y. Larry Hsu and Andrew MacNaughton*

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Research Laboratory
Oceanography Division
Stennis Space Center, MS 39529-5004

8. PERFORMING ORGANIZATION REPORT NUMBER
NRL/FR/7322--97-9666

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Commander, Space and Naval Warfare Systems Command
PMW 185-3B
4301 Pacific Hwy.
San Diego, CA 92110-3127

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
*Planning Systems Incorporated, 115 Christian Lane, Slidell, LA 70458

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Accurate estimates of wave conditions have become increasingly critical to Navy operations in the coastal regions. In the shallow water, wave propagation is affected by many dynamic processes, including shoaling, refraction, diffraction and energy dissipation. The existing RCPWAVE wave model in the Navy-Standard surf model was developed for open coast with slowly varying bathymetry. It cannot handle certain coastline configurations such as those include islands or peninsulas. It also has a serious numerical instability problem. The more recently developed REF/DIF1 wave model has a more robust formulation and does not suffer from the same limitations. REF/DIF1 has been evaluated and validated for Naval applications. In this report, all three OAML documents, i.e. software requirements specification (SRS), software design document (SDD) and software test description (STD) are included.

14. SUBJECT TERMS
coastal wave model, wave refraction and diffraction, surf forecasting

15. NUMBER OF PAGES
82

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Same as report

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Preceding Page Blank

CONTENTS

COMPLETE OCEANOGRAPHIC AND ATMOSPHERIC MASTER LIBRARY DOCUMENTATION OF THE SHALLOW-WATER WAVE REFRACTION AND DIFFRACTION MODEL

SOFTWARE REQUIREMENTS SPECIFICATIONS

1.0 SCOPE

1.1 Identification

This Software Requirements Specification (SRS) describes the functional requirements of a wave prediction model Computer Software Configuration Item (CSCI) identified as the Shallow-Water Wave Refraction and Diffraction Model (REF/DIF1). The CSCI should read in local bathymetry and wave data prior to execution of the actual model, and the model itself should predict wave behavior considering the processes of shoaling, refraction, energy dissipation; and diffraction. The CSCI should at a minimum output wave height data, wave direction data, and tide-corrected depth data.

1.2 Overview

This SRS describes the functional requirements for the Shallow-Water Wave Refraction and Diffraction Model. It describes what is expected from the CSCI and the methods to verify that the requirements are met. The CSCI requirements will include the following: Capability Requirements, External Interface Requirements, Environment Requirements, Computer Resource Requirements, Software Quality Factors, and Design and Implementation Constraints. This report has been prepared for transition into the Oceanographic and Atmospheric Master Library (OAML), in accordance with the Software Documentation Standards for Environmental System Product Development (Naval Oceanographic Office 1995), which is based on Military Standard for Software Development and Documentation, MIL-STD-498.

2.0 REFERENCED DOCUMENTS

Dean, R. G. and R. A. Dalrymple, "Water Wave Mechanics for Engineers and Scientists," Salem: World Scientific Publishing Company, River Ridge, NJ, 1991.

Earle, M. D., "Surf Forecasting Software Scientific Reference Manual," NORDA Tech. Note 351, Naval Research Laboratory, Stennis Space Center, MS, 1989.

Kirby, J. T. and R. A. Dalrymple, "A Parabolic Equation for the Combined Refraction-Diffraction of Stokes Waves by Mildly Varying Topography," *J. Fluid Mechanics* **136**, 543–566 (1983).

Kirby, J. T. and R. A. Dalrymple, "Modeling Waves in Surf Zones and Around Islands," ASCE *J. Waterway, Port, Coast., and Ocean Eng.* **112**(1), 78–93 (1986).

Kirby, J. T. and R. A. Dalrymple, "Combined Refraction-Diffraction Model REF/DIF1, Version 2.5, Documentation and User's Manual," Report 94-22, Center for Applied Coastal Research, University of Delaware, Newark, DE, 1994, 171 pp.

Naval Oceanographic Office, "Software Documentation Standards for Environmental System Product Development," Naval Oceanographic Office Systems Integration Department, Stennis Space Center, MS, 1995.

O'Reilly, W. C. and R. T. Guza, "A Comparison of Two Spectral Wave Models in the Southern California Bight," *Coastal Eng.* **19**, 263–282 (1993).

Weggel, J. R., "Maximum Breaker Height," *J. Waterways, Harbors, Coastal Eng.*, ASCE, 529–548 (1972).

Wiegel, R. L., "Diffraction of Waves by Semi-Infinite Breakwater," *Proc. Am. Soc. Civ. Eng.* **88**, HY 1:27–44 (1962).

3.0 REQUIREMENTS

There are essentially two sets of capabilities required for the CSCI described by this SRS. The first set describes the model itself, i.e., the elements contributing to computation of the solutions to the mild slope model derived from the Laplace equation with nonlinear boundary conditions (e.g., Kirby and Dalrymple 1983). The other set of requirements deal more with the CSCI itself; i.e., software, external interface, and output requirements of the CSCI itself (hereafter referred to as the CSCI, or software).

At a minimum the model must consider the following:

- The combined effects of refraction and diffraction in shallow water;
- Energy dissipation terms, such as laminar or turbulent boundary flow, Darcy flow (porous sand), or depth-induced breaking;
- Wave propagation across irregular bathymetry (must also consider local currents) and;
- Several wave climates.

Requirements for the model are discussed in greater detail in Sec. 3.1.

The CSCI as a whole must also meet the following requirements:

- The CSCI should produce, at a minimum, the following output:
  — Wave amplitude and wave direction:
- The CSCI must interface (directly or indirectly) with the Surf Forecasting Software Model (SURF);
- The CSCI must be portable;
- Mechanisms for data input must be simple;
- It should provide means for easy repetition of the model;
- Output must be in a standard, easy-to-read format.

The individual specifications for the CSCI are discussed in greater detail in Secs. 3.2 through 3.14.

3.1 CSCI CAPABILITY REQUIREMENTS

Outlined in this section are the requirements for the solution model for this CSCI. These include various model solutions, energy dissipation terms, the local wave climate, boundary conditions, and subgridding.

3.1.1 Wave Model

There are only minimal requirements for the CSCI model solution. The solution scheme must be stable and efficient.

3.1.1.1 Linear (Small Amplitude) Theory

The linear (or small amplitude) theory of two-dimensional progressive surface gravity waves is widely used because it supplies useful results for many applications while being much easier to apply than the many available complex finite amplitude theories. For this reason, and because linear theory will provide reasonable solutions for both deep and shallow water, the CSCI described by this report should incorporate linear theory solutions into the model.

3.1.2 Shallow-Water Wave Transformations

The ultimate goal of this CSCI is to provide consistent, reasonably accurate modeling of wave propagation from deep to shallow water, considering the processes of shoaling, refraction, and diffraction. Some models consider only refraction; however, this CSCI must include a solution that combines both refraction and diffraction when necessary. The CSCI model solution must also consider the influence of currents in addition to bathymetry.

3.1.2.1 Wave Shoaling

As a wave moves into increasingly shallow (shoaling) water, changes are manifested in the wave appearance. Wave celerity and wavelength both decrease, but height increases. Wave shoaling (increase of wave height because of shoaling water) must be considered by the CSCI solution when evaluating wave speed and height in the shallow waters.

3.1.2.2 Refraction

Waves are also subject to wave refraction upon entering shallow water. Refraction is the direction change of propagating waves because of changing bottom bathymetry. Because phase speed is depth dependent, the direction of wave propagation shifts so that the crests tend to parallel the depth contours (Dean and Dalrymple 1991). Shoaling and refraction are not isolated processes. They have an interactive effect upon wave height. Both shoaling and refraction influences must be incorporated into the solution model of the CSCI.

3.1.2.3 Diffraction Model

Diffraction is also a process where energy spreads laterally. For example, as a propagating wave interacts with a nontransmitting barrier, the part of the wave hitting the barrier will be partially dissipated and partially reflected. Beyond the barrier, however, energy from the remainder of the propagating wave will bleed into the lee of the barrier (Wiegel 1962).

3.1.2.4 Shallow-Water Wave Refraction and Diffraction Model

Most models include solutions that consider only refraction or diffraction. However, the CSCI model solution must adequately evaluate both refraction and diffraction with complex bathymetry (Kirby and Dalrymple 1986).

3.1.2.5 Wave/Current Interactions

Currents can influence wave celerity, direction, and length. Currents retard (accelerate) wave movement, shortening (lengthening) the wave length, depending on whether the current is flowing against (with) the direction of wave propagation. If current velocities are high enough, waves can be stopped entirely (wave blocking).

3.1.3 Energy Dissipation

Energy dissipation occurs in a number of ways depending on the situation being modeled. Bottom frictional losses because of rough, porous, or viscous bottoms or because of shallow water and wave breaking must be included in the model. Discussion of the specific forms of energy dissipation required for the CSCI are included in Secs. 3.1.3.1 through 3.1.3.4.

3.1.3.1 Laminar Surface and Bottom Boundary Layers

At the water surface and at the bottom, boundary layers occur due to the action of viscosity. For a contaminated surface resulting from surface films (natural or otherwise), a significant amount of damping occurs that is dependent on the value of the fluid viscosity.

3.1.3.2 Turbulent Bottom Boundary Layer

In the field, the likely wave conditions are such that the bottom boundary layer is turbulent. These conditions are most frequent when waves are large or the bottom is particularly rough. The CSCI model must contain alternative terms for energy dissipation because of turbulence. Bottom friction loss can be significant for wave propagation through a long and shallow shelf.

3.1.3.3 Porous Sand

The impact of wave flow into the bottom is usually assumed to be negligible. However, most sea bottoms (particularly sandy ones) are porous, and the waves induce a flow into the bed because of a pressure gradient between bottom waters and bottom pore pressure. This results in wave damping because of Darcy flow into the sand. Terms accounting for Darcy flow must be available within the CSCI model solution.

3.1.3.4 Wave Breaking

Wave breaking is a significant source of energy loss, principally through turbulence and work against bottom friction. Breaking occurs when a wave reaches a certain height/depth ratio and becomes unstable (Dean and Dalrymple 1991) or from the exponential relationship between height and depth based on beach slope (Weggel 1972).

3.1.4 Wave Climate

The CSCI model should be able to completely model monochromatic, discrete direction, or directionally spreading waves for individual frequencies.

3.1.4.1 Monochromatic Waves

Monochromatic waves are the usual prototype for wave models and consist of a single wave train with all waves of the same height (and frequency). This case must be included in the CSCI model.

3.1.4.2 Discrete Direction Waves

It is likely that numerous waves and/or swells coming from many different directions are present on the sea surface at once. The model must be able to model these waves of different amplitudes and directions for a given frequency.

3.1.4.3 Directional Spectrum

The CSCI model solution should also solve for waves of a given frequency propagating along a directional spectrum using superposition. Outside the surf zone where wave/wave interaction is weak, the superposition of wave runs has been proven to be a valid approach (O'Reilly and Guza 1993).

3.1.5 Model Features

There are a number of additional features required of the CSCI model solution; particularly, lateral boundary condition settings and subgrid capabilities must be included as model options.

3.1.5.1 Lateral Boundary Conditions

Many models require *a priori* knowledge of the lateral boundary conditions. Lateral boundary conditions can range from complete transmission of wave energy to total reflection. The requirement for the CSCI model solution is that both transmitting/reflecting lateral boundary conditions are included as options.

3.1.5.2 Subgrids

It is assumed that the model solution will be computed over a large-scale reference grid. However, in addition to this reference grid, the CSCI model must include provisions for subgrid computation. This would allow the user to evaluate waves in complex bathymetry areas to produce direct-calculated, rather than interpolated, results for the subgrid area. This allows the user to evaluate large areas as well as small areas of particular interest at the same time.

3.2 CSCI External Interface Requirements

This CSCI is designed to produce a number of files for plotting and evaluation. Some of these files are destined for use by the SURF program (Earle 1989). Other external interfaces (data files) are for use by and for repeated applications of this CSCI. The structure of the external interfaces described in this section to this CSCI is shown in Fig. 3.2-1.

3.2.1 Interface with Model Parameters File

As there are a number of options associated with CSCI model solution (among them the choice between a Small Amplitude Theory solution or a nonlinear solution, forms of energy dissipation,

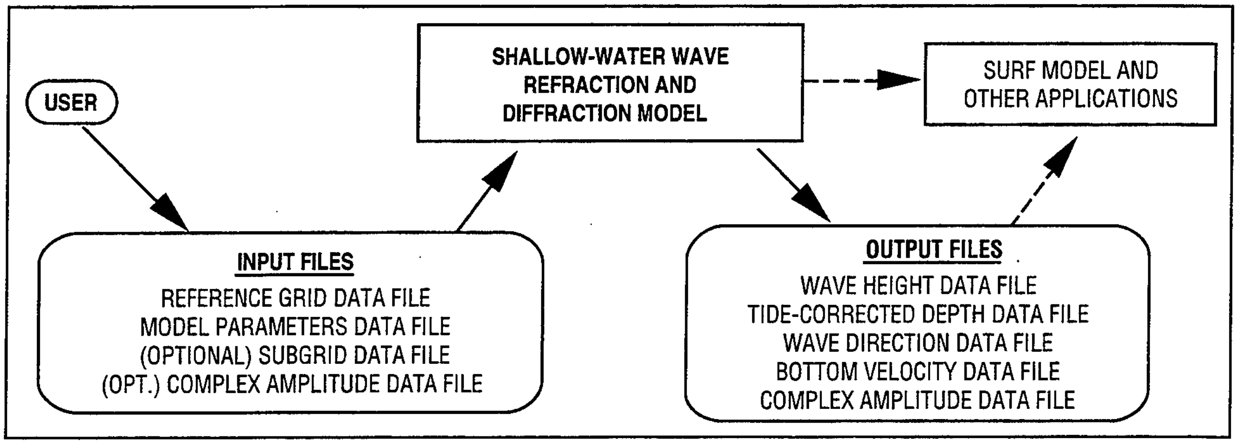

Fig. 3.2-1 — REF/DIF External Interface Structure. The above diagram shows the required (and anticipated) external interfaces between the Combined Refraction and Diffraction Model outlined in this SRS and the SURF model and numerous data files listed in this document.

and whether currents or subgrids are present), the user will have to select these options through a Character User Interface (CUI). The CSCI must produce a model parameters file that will store the options chosen.

Model Parameters File

a. *Priority assigned to interface:* The CSCI must write model parameters to file, but user must be able to decide whether to use existing file or create new one.

b. *Requirements on the type of interface:* Storage and retrieval of data.

c. *Required characteristics of individual data elements:* Individual data elements should be in standard sequential order.

3.2.2 Interface with Reference Grid Data File

The CSCI is required to read in local bathymetry and current data. This data should be found in the user-provided Reference Grid Data File.

Reference Grid Data File

a. *Priority assigned to interface:* The CSCI must read bathymetry and (if available) current data through the interface with this data file.

b. *Requirements on the type of interface:* Retrieval of data.

c. *Required characteristics of individual data elements:* Individual data elements should be real numbers (units are meters for bathymetric data and m/s^{-1} for current data) in standard sequential order.

3.2.3 Interface with Subgrid Data File (Optional)

The user can direct the CSCI to read in subgrid bathymetry data for high-resolution modeling of an area within the reference grid. This data should be found in the user-provided Subgrid Data File.

Subgrid Data File

a. *Priority assigned to interface:* The CSCI has the option to read subgrid bathymetry data through the interface with this data file.

b. *Requirements on the type of interface:* Retrieval of data.

c. *Required characteristics of individual data elements:* Individual data elements should be real numbers (units are in meters) in standard sequential order.

d. *Required characteristics of data element assemblies:* The interface structure with this data file should be similar to that of the Reference Grid.

3.2.4 Interface with First-Row (Last-Row) Complex Amplitude Data File (Optional)

The CSCI must be able to read from (write to) an initial (final) row of complex amplitude data. This data should initialize the model. Storing the last row of data allows another model to be built farther along the direction of wave propagation.

First-Row (Last-Row) Complex Amplitude Data File

a. *Priority assigned to interface:* The CSCI must be able to read from (or write to) this data file.

b. *Requirements on the type of interface:* Retrieval and storage of data.

c. *Required characteristics of individual data elements:* Individual data elements should be complex wave amplitude (units are in meters) in standard sequential order.

3.2.5 Interface with Wave Height Output Data File

The CSCI must be able to write the wave height at each reference gridpoint to this data file.

Wave Height Output Data File

a. *Priority assigned to interface:* The CSCI must be able to write to this data file.

b. *Requirements on the type of interface:* Storage of data.

c. *Required characteristics of individual data elements:* Individual data elements are real (units are in meters) in standard sequential order.

d. *Required characteristics of data element assemblies:* This file must have standard structure.

3.2.6 Interface with Tide-Corrected Depth Output Data File

The CSCI must be able to write the tide-corrected depth at each reference gridpoint to this data file.

Tide-Corrected Depth Output Data File

a. *Priority assigned to interface:* The CSCI must be able to write to this data file.

b. *Requirements on the type of interface:* Storage of data.

c. *Required characteristics of individual data elements:* Individual data elements are real (units are m) in standard sequential order.

d. *Required characteristics of data element assemblies:* This file must have standard structure identical to that of the Wave Height Output Data File (Sec. 3.2.5).

3.2.7 Interface with Wave Direction Output Data File

The CSCI must be able to write the propagation angle of the wave at each reference gridpoint to this data file.

Wave Direction Output Data File

a. *Priority assigned to interface:* The CSCI must be able to write to this data file.

b. *Requirements on the type of interface:* Storage of data.

c. *Required characteristics of individual data elements:* Individual data elements are real (units are in degrees) in standard sequential order.

d. *Required characteristics of data element assemblies:* This file must have standard structure identical to that of the Wave Height Output Data File (Sec. 3.2.5).

3.2.8 Interface with Bottom Velocity Output Data File

The CSCI must be able to write the bottom velocity at each reference gridpoint to this data file.

Bottom Velocity Output Data File

a. *Priority assigned to interface:* The CSCI must be able to write to this data file.

b. *Requirements on the type of interface:* Storage of data.

c. *Required characteristics of individual data elements:* Individual data elements are real (units are in m/s^{-1}) in standard sequential order.

d. *Required characteristics of data element assemblies:* This file must have standard structure identical to that of the Wave Height Output Data File (Sec. 3.2.5).

3.2.9 Interface with Complex Amplitude Output Data File

The CSCI must be able to write the wave complex amplitude at the computational resolution (i.e., at each point, both reference grid and subgrid where the model is evaluated) to this data file.

Computational Resolution Complex Amplitude Output Data File

a. *Priority assigned to interface:* The CSCI must be able to write to this data file.

b. *Requirements on the type of interface:* Storage of data.

c. *Required characteristics of individual data elements:* Individual data elements are complex (units are in meters) in standard sequential order.

d. *Required characteristics of data element assemblies:* This file must be of similar structure (though of different size) to that of the Wave Height Output Data File (Sec. 3.2.5).

3.3 CSCI Internal Interface Requirements

All decisions about internal interface are left to the design.

3.4 CSCI Internal Data Requirements

All decisions about internal data are left to the design.

3.5 Adaptation Requirements

There is no installation-dependent data to be provided by the CSCI and no required operational parameters dictated by operational needs.

3.6 CSCI Environment Requirements

The CSCI should be portable to Unix and MS-DOS.

3.7 Software Quality Factors

Functionality – The CSCI must perform all of the required functions as detailed in Sec. 3.0. These criteria will be graded as outlined in Sec. 4.0.

Reliability – The CSCI must perform with consistent results, i.e., produce the same results for identical situations.

Maintainability – The CSCI must be correct as delivered.

Availability – The CSCI must be compact and self-contained; it must, therefore, be easily accessed and operated.

Flexibility – The CSCI must be structured so that the code itself and model solution can easily be upgraded or adapted.

Portability – The CSCI must be self-contained with no links to proprietary programs or data. The CSCI must also be written in standard code so as to be usable on numerous platforms.

Reusability – The CSCI must be usable for different situations. The CSCI must be able to accommodate different sizes of reference grids.

Testability – The CSCI must meet the reliability, portability, and reusability requirements. Output from the CSCI on different platforms must be comparable.

Usability – The CSCI must be driven by a CUI and, therefore, user-friendly.

3.8 Design and Implementation Constraints

The CSCI must be written in either Ansi-standard or Fortran and must be accepted by most compilers.

3.9 Training-Related Requirements

The CSCI should be user-friendly and, therefore, easily learned from available documentation (e.g., the User's Guide (Kirby and Dalrymple 1994)).

4.0 QUALIFICATIONS TRACEABILITY

Table 4.1 outlines the methods to be used to ensure that the requirements listed in this document are met.

5.0 NOTES

5.1 Acronyms and Abbreviations

CDRL	Contract Data Requirements List
CSCI	Computer Software Configuration Item
CUI	Character User Interface
m	meters
m/s^{-1}	meters per second
MIL-STD	Military Standard
OAML	Oceanographic and Atmospheric Master Library
REF/DIF1	Shallow-Water Wave Refraction and Diffraction Model
SRS	Software Requirements Specification
SURF	Surf Forecasting Software Model

Table 4.1 — Requirements Traceability Lists SRS Requirements with Corresponding SRS Discussion and Method of Testing

REQUIREMENT	SRS SECTION DESCRIBING REQUIREMENT	METHOD FOR EVALUATION
Linear Theory Model Solution	Sec. 3.1.1.1	Testing
Shoaling-Water Applicable	Sec. 3.1.2.1	Testing
Consideration of Refraction/Diffraction	Secs. 3.1.2.2, 3.1.2.3, and 3.1.2.4	Testing
Energy Dissipation Terms	Secs. 3.1.3 and 3.2.1	Inspection of Code
Laminar Surface/Bottom Flow	Sec. 3.1.3.1	Inspection of Code
Turbulent Bottom Flow	Sec. 3.1.3.2	Inspection of Code
Darcy Flow	Sec. 3.1.3.3	Inspection of Code
Depth-Induced Wave Breaking Included	Sec. 3.1.3.4	Testing
Accommodation of Irregular Bathymetry	Secs. 3.1.2.1 and 3.1.2.2	Testing
Consideration of Local Current Data	Secs. 3.1.2.5, 3.2.1, and 3.2.2	Testing
Inclusion of Local Subgrid Data	Secs. 3.1.5.2 and 3.2.3	Testing
Response to Several Wave Climates	Sec. 3.1.4	Inspection of Code and Testing
Monochromatic Waves	Sec. 3.1.4.1	Testing
Discrete Direction Waves	Sec. 3.1.4.2	Inspection of Code
Directional Spectrum Waves	Sec. 3.1.4.3	Inspection of Code
Lateral Boundary Conditions	Sec. 3.1.5.1	Testing
Required Output	Sec. 3.2	Testing
Wave Height	Sec. 3.2.5	Testing
Direction of Wave Propagation	Sec. 3.2.7	Testing
Tide-Corrected Depth	Sec. 3.2.6	Testing
Ease of Data Input	Sec. 3.2.1	Testing

SOFTWARE DESIGN DOCUMENT

1.0 SCOPE

1.1 Identification

This Software Design Document (SDD) describes the Computer Software Configuration Item (CSCI) identified as the Shallow-Water Wave Refraction and Diffraction Model. The wave prediction model is based on the weakly nonlinear combined refraction and diffraction model (REF/DIF1) initially developed by Kirby and Dalrymple (1983a) using parabolic approximations to the equations based on minimax principles by Kirby (1986a) with numerous enhancements. The enhancements include consideration of a strong current jet from Kirby and Dalrymple (1983b), inclusion of laminar and turbulent energy dissipation terms based on Dean and Dalrymple (1984), the allowance for a turbulent bottom boundary layer as described by Dean and Dalrymple (1984), an additional term respecting wave damping for flow in porous sand as discussed by Liu and Dalrymple (1984), and the use of terms describing the dissipation of wave energy due to wave breaking as evaluated by Dally et al. (1985). The model is used to predict the propagation of water waves over irregular bottom bathymetry and around islands considering the processes of shoaling, refraction, energy dissipation, and diffraction. This report follows and adopts from the original REF/DIF1 manual.

1.2 Overview

This SDD describes the design and structure of the REF/DIF1. It describes the CSCI-wide design decisions, the architectural design, and the detailed design. The architecture design partitions the CSCI along procedural steps into individual software units (SU) and details the external and internal interfaces, the overall systemic organization and flow, and the software units organization and flow. In the detailed design the input, output, and processing within each software unit are detailed. This is followed by a requirements traceability, tracing software unit to the requirements being satisfied. This report has been prepared for transition into the Oceanographic and Atmospheric Master Library (OAML), in accordance with the Software Documentation Standards for Environmental System Product Development defined by the Naval Oceanographic Office, which is based on the Military Standard for Software Development and Documentation, MIL-STD-498.

2.0 REFERENCED DOCUMENTS

Berkhoff, J. C. W., "Computation of Combined Refraction-Diffraction," *Proc. 13th Int. Conf. Coastal Engrg.*, ASCE, Vancouver, **1**, 470–490 (1972).

Booij, N., "Gravity Waves on Water with Non-Uniform Depth and Current," Doctoral dissertation, Technical University of Delft, The Netherlands, 1981.

Dally, W. R., R. G. Dean, and R. A. Dalrymple, "Wave Height Variations Across Beaches of Arbitrary Profile," *J. Geophys. Research* **90**, 11,917–11,927 (1985).

Dean, R. G. and R. A. Dalrymple, "Water Wave Mechanics for Engineers and Scientists," World Scientitic, River Ridge, NJ, 1984.

Hedges, T. S., "An Empirical Modification to Linear Wave Theory," *Proc. Inst. Civ. Eng.* **61**, 575–579 (1976).

Kirby, J. T., "Propagation of Weakly-Nonlinear Surface Water Waves in Regions with Varying Depth and Current," Office of Naval Research Tech. Rept. 14, Res. Rept. CE-83-37, Office of Naval Research, Dept. of Civil Engineering, University of Delaware, Newark, DE, 1983.

Kirby, J. T. and R. A. Dalrymple, "A Parabolic Equation for the Combined Refraction-Diffraction of Stokes Waves by Mildly Varying Topography," *J. Fluid Mech.* **136**, 543–566 (1983a).

Kirby, J. T. and R. A. Dalrymple, "The Propagation of Weakly Nonlinear Waves in the Presence of Varying Depth and Current," *Proc. XX^{th} Congress I.A.H.R.*, Moscow, 1983b.

Kirby, J. T., "A Note on Linear Surface Wave-Current Interaction," *J. Geophys. Res.* **89**, 745–747 (1984).

Kirby, J. T., "Higher-Order Approximations in the Parabolic Equation Method for Water Waves," *J. Geophys. Res.* **91**, 933–952 (1986a).

Kirby, J. T., "Rational Approximations in the Parabolic Equation Method for Water Waves," *Coastal Engineering* **10**, 355–378 (1986b).

Kirby, J. T., "Open Boundary Condition in Parabolic Equation Method," *J. Waterway, Port, Coast, and Ocean Eng.* **112**, 460–465 (1986c).

Kirby, J. T., "Damping of High Frequency Noise in the Large Angle Parabolic Equation Method," unpublished manuscript, 1993.

Kirby, J. T. and R. A. Dalrymple, "Combined Refraction/Diffraction Model, REF/DIF1 Version 2.5, Documentation and User's Manual," Center for Applied Coastal Research, Department of Civil Engineering, University of Delaware, Newark, DE, 1994.

Liu, P. L.-F. and R. A. Dalrymple, "On Weak Reflection of Water Waves," *J. Fluid Mech.* **131**, 59–71 (1984).

Mei, C. C. and E. O. Tuck, "Forward Scattering by Thin Bodies," *SIAM J. Appl. Math.* **39**, 178–191 (1980).

O'Reilly, W. C. and R. T. Guza, "A Comparison of Two Spectral Wave Models in the Southern California Bight," *Coastal Engineering* **19**, 263–282 (1993).

Radder, A. C., "On the Parabolic Equation Method for Water-Wave Propagation," *J. Fluid Mech.* **95**, 159–176 (1979).

Yue, D. K. P. and C. C. Mei, "Forward Diffraction of Stokes Waves by a Thin Wedge," *J. Fluid Mech.* **99**, 33–52 (1980).

3.0 CSC-WIDE DESIGN DECISIONS

Among available shallow-water wave models, REF/DIF1 has the most complete physics and features. The design of the model follows the formulation and manual of Kirby and Dalrymple (1994). Details of the model are summarized in the remainder of this section.

3.1 Theoretical Background

The propagation of water waves over irregular bottom bathymetry and around islands involves many processes—shoaling, refraction, energy dissipation, and diffraction. This SDD describes the REF/DIF1 model initially developed by Kirby and Dalrymple (1983a), which incorporates all of the effects mentioned above.

Combined refraction/diffraction models include both refraction and diffraction effects explicitly, thus permitting the modeling of waves in regions where the bathymetry is irregular and where diffraction is important.

3.2 Wave Models

3.2.1 Mild Slope Equation

The problem of water waves propagating over irregular bathymetry in arbitrary directions is a three-dimensional problem and involves complicated nonlinear boundary conditions. Very few solutions to the three-dimensional problem exist, and those that do are only for flat bottoms. To simplify the problem in three dimensions, Berkhoff (1972) noted that the important properties of linear progressive water waves could be predicted by a weighted vertically integrated model. (The vertical integration reduces the problem to only the two horizontal dimensions, x and y.)

Berkhoff's equation is known as the mild slope equation. It is written in terms of the surface displacement, $\eta(x,y)$. The equation, in terms of horizontal gradient operator, is

$$\nabla_h\left(CC_g\nabla_h\eta\right)+\sigma^2\frac{C_g}{C}\eta=0\,. \tag{1}$$

Here,

$$C=\sqrt{(g/k)\tanh\,kh,}\ \text{the wave celerity, and} \tag{1a}$$

$$C_g=C\{1+2kh/\sinh 2kh\}/2,\ \text{the group velocity}\,, \tag{1b}$$

where the local water depth is $h(x,y)$ and g is the acceleration of gravity. The local wave number, $k(x,y)$, is related to the angular frequency of the waves, σ, and the water depth h by the linear dispersion relationship

$$\sigma^2=gk\ \tanh\ kh\,. \tag{2}$$

The model Eq. (1) is an approximation; however, it is good even for moderately large local bottom slopes (see Booij 1983).

For the linear mild slope equation, Radder (1979) developed a parabolic model that had several advantages over the elliptic form presented by Berkhoff. First, the boundary condition at the downwave end of the model area was no longer necessary and secondly, very efficient solution techniques were available for the finite difference form of the model. Radder's approximation for derivatives transverse to the wave direction resulted in a restriction on his parabolic model: the waves must propagate within 45° of the assumed wave direction. Booij (1981) also developed a splitting of the elliptic equation, but his procedure included more terms in the approximation to the lateral derivative and, therefore, his procedure enabled the parabolic model to handle wave propagation within 60° of the assumed direction. More recently, Kirby (1986b) developed an extension to the Booij approximation based on a minimax principle, which further extended the range of validity of the model equations. The wave-current version of the resulting model is integrated into Version 2.5 REF/DIF1.

3.2.2 Diffraction Models

Mei and Tuck (1980) developed a simple parabolic equation for wave diffraction. Their equation is

$$\frac{\delta A}{\delta x} = \frac{i}{2k}\frac{\delta^2 A}{\delta y^2}, \tag{3}$$

where A is a complex amplitude related to the water surface displacement by

$$\eta = Ae^{i(kx-\sigma t)}. \tag{4}$$

Yue and Mei (1980), using a multiple scales approach, developed a nonlinear form of this equation that accurately predicts the propagation of a third order Stokes wave.

3.2.3 Nonlinear Combined Refraction/Diffraction Models

Kirby (1983), using a Lagrangian approach, and Kirby and Dalrymple (1983a), with a multiple scales technique, developed the predecessor to the REF/DIF1 model and bridged the gap between the nonlinear diffraction models and the linear mild slope equation. The hyperbolic form of this model, for time-dependent applications, requires the use of boundary conditions on all sides of the model domain. This is a difficult requirement, as the reflected wave at a boundary is not generally known *a priori*. Kirby (1986a) rederived the equation with a parabolic approximation. Comparisons between linear and nonlinear parabolic models clearly showed the importance of the nonlinear dispersion terms in the governing equations.

3.2.4 Wave-Current Interaction Models

Booij (1981), using a Lagrangian approach, developed a version of the mild slope equation including the influence of current. This model is a weak current model in that the currents are assumed to be small and any products of currents are neglected as small. Kirby (1984) presented the corrected form of this mild slope model. A nonlinear correction and the ability to handle strong currents were added by Kirby and Dalrymple (1983b) and results for waves interacting with a current jet were obtained.

The rederived wide-angle parabolic approximation allowed the study of waves with larger angles of wave incidence with respect to the x axis. This more accurate equation was used as the basis for earlier versions of REF/DIF1 and was extended to include the more accurate minimax approximation (Kirby 1986b) for the present version of REF/DIF1. The revised governing equation is given by the equation below:

$$\left(C_g+U\right)A_x-2\Delta_1 A_y+i\left(\bar{k}-a_0k\right)\left(C_g+U\right)A+\left\{\frac{\sigma}{2}\left(\frac{C_g+U}{\sigma}\right)_x-\Delta_1\sigma\left(\frac{V}{\sigma}\right)_y\right\} \tag{5}$$

$$A+i\Delta'\left[\left(p-V^2\right)\left(\frac{A}{\sigma}\right)_y\right]_y-i\Delta_1\left\{\left[UV\left(\frac{A}{\sigma}\right)_y\right]_x+\left[UV\left(\frac{A}{\sigma}\right)_x\right]_y\right\}+\frac{i\sigma k^2}{2}D|A|^2A+\frac{\omega}{2}$$

$$A+\frac{-b_1}{k}\left\{\left[\left(p-V^2\right)\left(\frac{A}{\sigma}\right)_y\right]_{yx}+2i\left(\sigma V\left(\frac{A}{\sigma}\right)_y\right)_x\right\}$$

$$b_1\beta\left\{2i\omega U\left(\frac{A}{\sigma}\right)_x+2i\sigma V\left(\frac{A}{\sigma}\right)_y-2UV\left(\frac{A}{\sigma}\right)_{xy}+\left[\left(p-V^2\right)\left(\frac{A}{\sigma}\right)_y\right]_y\right\}-\frac{i}{k}$$

$$b_1\left\{\left(\omega V\right)_y+3\left(\omega U\right)_x\right\}\left(\frac{A}{\sigma}\right)_x-\Delta_2\left\{\omega U\left(\frac{A}{\sigma}\right)_x+\frac{1}{2}\omega U_x\frac{A}{\sigma}\right\}+ik\omega U\left(a_0-1\right)\left(\frac{A}{\sigma}\right)=0\,,$$

where $\bar{k}$ is the reference wave number taken as the average wave number along the y axis, U is the mean current velocity in the x-coordinate direction, and V is in the y direction. Additional terms are defined as follows:

$$p=CC_g\,; \tag{5a}$$

$$D=\frac{\cosh 4k+8-\tanh^2 kh}{8\sinh^4 kh}\,,\ \text{where } D \text{ is the nonlinear term;} \tag{5b}$$

$$\beta=\frac{k_x}{k^2}+\frac{\left(k\left(p-U^2\right)\right)_x}{2k^2\left(p-U^2\right)}\,; \tag{5c}$$

$$\Delta_1=a_1-b_1; \tag{5d}$$

$$\Delta_2=1+2a_1-2b_1; \tag{5e}$$

$$\Delta'=a_1-b_1\frac{\bar{k}}{k}\,; \tag{5f}$$

and ω is a dissipation factor discussed in Sec. 3.4. The coefficients a_0, a_1, and b_1 depend on the aperture width chosen to specify the minimax approximation; see Kirby (1986b). The combination

$$a_0 = 1$$

$$a_1 = -0.75$$

$$b_1 = -0.25$$

is used.

The dispersion relationship relating the angular frequency of the wave, the depth and the wave number was changed to reflect the Doppler shift due to currents. The new form of Eq. (2) is

$$(\omega - kU^2) = gk \tanh kh , \tag{6}$$

where the absolute frequency, ω, is related to the intrinsic frequency, σ, by

$$\omega = \sigma + kU , \tag{7}$$

where the assumption that the wave is primarily traveling in the x direction is used.

3.3 Model Assumptions

The REF/DIF1 model, in parabolic form, has a number of inherent assumptions and it is necessary to discuss these directly.

3.3.1 Mild Slope Bottom

The mathematical derivation of the model equations assumes that the variations in the bottom occur over distances that are long in comparison to wavelength.

3.3.2 Weak Nonlinearity

The model is based on a Stokes perturbation expansion and is, therefore, restricted to applications where Stokes solutions are valid. Hedges (1976) developed a heuristic dispersion relationship to provide a model that is valid in very shallow water. This is included as an option in the model. The relationship between the frequency and the water depth is

$$\sigma^2 = gk \tanh(kh(1 + |A|/h)) . \tag{8}$$

In shallow water, this equation matches that of a solitary wave, while in deep water it asymptotically approaches the linear wave result, neglecting real amplitude dispersive effects. For this reason, a model with a dispersion relationship which is a smooth patch between the Hedges form and the Stokes relationship is used. This hybrid model is described in Kirby and Dalrymple (1986b). There are, as a result of the different dispersion relationships possible, three options in REF/DIF1: (1) a linear model, (2) a Stokes-to-Hedges nonlinear model, and (3) a Stokes model. Of these options, the second will cover a broader range of water depths and wave heights than the others.

3.3.3 Wave Direction

The wave direction is confined to a sector ±70° to the principal assumed wave direction due to the use of the minimax wide angle parabolic approximation of Kirby (1986b).

3.4 Energy Dissipation

Energy dissipation in the model occurs in a number of ways depending on the situation being modeled. An energy loss term, due to Booij (1981) and expanded by Dalrymple et al. (1984a), permits the model to treat bottom frictional losses due to rough, porous, or viscous bottoms, surface films, and wave breaking. The linear form of the mild slope equation with dissipation is

$$\frac{\delta A}{\delta x} = \frac{i}{k}\frac{\delta^2 A}{\delta y^2} + \omega a \, , \tag{9}$$

where the dissipation factor, ω, is given by a number of different forms depending on the nature of the energy dissipation.

3.4.1 Laminar Surface and Bottom Boundary Layers

At the water surface and at the bottom, boundary layers occur due to the action of viscosity. For a contaminated surface, resulting from surface films (natural or otherwise), a significant amount of damping occurs, which is dependent on the value of the fluid viscosity.

3.4.2 Turbulent Bottom Boundary Layer

In the field, the likely wave conditions are such that the bottom boundary layer is turbulent. In this case, an alternative specification of the energy dissipation must be provided.

3.4.3 Porous Sand

Most sea bottoms are porous and the waves induce a flow into the bed. This results in wave damping due to the Darcy flow in the sand.

3.4.4 Wave Breaking

For wave breaking the model is more complicated. Dally et al. (1985) showed that the rate of loss of wave energy flux is dependent on the excess of energy flux over a stable value. By using the Kirby and Dalrymple (1986a) dissipation model and a breaking index relation ($H > 0.78$ h, where H is wave height) to determine the onset of breaking, the REF/DIF1 model is able to represent waves both outside and inside of a surf zone. The wave breaking algorithm is always active in the model.

Large surface piercing islands and causeways that would have surf zones are handled by the "thin film" technique of Dalrymple et al. (1984b) and Kirby and Dalrymple (1986a). This procedure permits the easy computation of wave heights around arbitrarily shaped islands by replacing islands with shoals of extremely shallow depth (1 cm). The wave breaking routine reduces the wave heights over the shoal to less than 1/2 cm, which results in a wave that carries negligible energy and, therefore, no longer affects any physical processes.

3.5 Wave Climate

3.5.1 Monochromatic Waves

While the REF/DIF1 model is typically used with monochromatic wave trains propagating in one given direction, there is no intrinsic restriction to this case.

3.5.2 Discrete Direction Waves

For several waves with different directions at a given frequency, the REF/DIF1 model is equipped to calculate the wave field produced by this boundary condition for up to 50 user-supplied A_ns and θ_ns (where A_n is the first-row complex amplitude and θ_n is the angle made by the wave to the x axis).

3.5.3 Directional Spectrum

Waves at a given frequency are often associated with a directional distribution. At the present time, the REF/DIF1 model can only calculate waves at a single frequency per calculation. The model will compute numerous frequencies per computer run; however, wave-wave interactions between different frequencies are not included.

It should be noted that it is often necessary to run the wave model at finer frequency and angular bandwidths over an area with a more complex bathymetry. For example, over the Southern California Bight, 0.01 Hz and 1° angular bandwidth are required (O'Reilly and Guza 1993). In these situations, REF/DIF1 can be run many times at different frequencies and directions to simulate the wave condition that can consist of a combination of swells and directional sea. The results can then be linearly combined. Outside the surf zone, where wave-wave interaction is weak, the superposition approximation has been proven to be a valid approach.

3.6 Numerical Development

3.6.1 Crank-Nicholson Technique

The parabolic model is solved in finite difference form. To accomplish this, the study area bathymetry must be input as a grid with (x,y) directions divided into rectangles of Δx and Δy sizes. The complex amplitude $A(x,y)$ will then be sought at each grid and, therefore, we can keep track of A by denoting its location, not by (x,y) but by (i,j), where $x = (i-1)\,\Delta x$ and $y = (j-1)\,\Delta y$. Then the values of $A(i,j)$ will satisfy Eq. (5) for all i between 1 and m and for all j between 1 and n. The procedure involves expressing all the derivatives in the (x,y) directions in terms of the complex amplitude at various gridpoints.

If a forward difference is used for the x direction and a central difference representation is used for the second derivatives in the lateral direction for all the derivatives in Eq. (5), then an explicit finite difference equation results for $A_{i+1,j}$. The equation can be solved directly for all the $A_{i+1,j}$ for a given row, provided appropriate lateral boundary conditions are prescribed. This explicit representation is not as accurate as an implicit scheme and, therefore, an implicit Crank-Nicholson is used for the amplitude calculations. For a given i row, the Crank-Nicholson scheme can be written

$$aA_{i+1,k+1} + bA_{i+1,j} + cA_{i+1,j-1} = dA_{i,j+1} + eA_{i,j} + fA_{i,j-1}\,, \qquad (10)$$

where the coefficients a, b, c, d, e, and f involve variable, complex, and nonlinear terms.

The scheme can be summarized as a solution of three unknown A_{i+1} terms on one side and three known A_i terms on the other. Due to the nonlinearity of the finite difference equation, the nonlinear terms are approximated on a first pass by using the $A_{i,j}$ values. Once the $A_{i+1,j}$ terms are computed, the equation is solved again for $A_{i+1,j}$ using the just-calculated values in the nonlinear terms. This two-pass iterative method insures that the nonlinearities in the model are treated accurately (Kirby and Dalrymple 1983a). The solution proceeds by moving one grid row in the x direction (incrementing i by one) and, using the two-pass, implicit-implicit technique, determining the complex amplitude $A_{i+1,j}$ for all the values of j on this row.

3.6.2 Initial and Lateral Boundary Conditions

The initial condition is vital for the parabolic model. The farthest seaward grid row corresponding to $i = 1$ is taken as constant depth (in practice, an average depth) and the incident wave(s) is prescribed here. This wave is then propagated over the bathymetry by the model.

The lateral boundary conditions are equally significant. None of the presently existing boundary conditions result in the total transmission of scattered waves. Therefore, for the REF/DIF1 model, a totally reflecting condition is generally used for each side ($j = 1$ and n). This requires that the specification of the model grid be done with care, as the reflection of the incident wave from the lateral boundaries can propagate into the region of interest rapidly and cause erroneous results.

In general, the width of the model should be such that no reflection occurs until far downwave of the region of interest. As a precaution, a graphical representation of the computed wave field should be examined to determine where the reflection from the boundaries is important. Partially transmitting boundaries are incorporated into the model (Kirby 1986c). In general, this boundary condition will result in less reflection in the model domain; however, since some reflection will occur, it is recommended that runs be carried out first with the reflecting boundary conditions to assess the regions potentially affected by reflection from the model boundaries.

3.6.3 Computational Gridding

In the propagation direction, x, the model will automatically determine the computational grid spacing if *ispace* has been set to 1. Otherwise, the user provides the grid spacing using the input *mr*, which permits variable spacing in the x direction.

In the original code, the user had the option of specifying the spacing in the y direction. We have added a new feature that has the code compute y direction spacing based on an input depth. The user then has greater control over the computational grid resolution (and, therefore, the accuracy) of the solution.

The y direction spacing parameter, *nd*, determines computational gridding in that direction. If *nd* is an integer greater than 0, the program uses that value for *nd* (as in the original code). If *nd* is either –1 or –2, the program will compute the wavelength at a minimum depth (*mindep*) and will produce computational grids based on the reference grid spacing (*dyr*) and that wavelength. If *nd* is set to –1, the minimum depth will be set to 1 m; if *nd* is set to –2, the user will be prompted for a real minimum depth.

3.6.4 Subgrids

To reduce the amount of data input and yet provide the user the ability to prescribe the fine scale bathymetry in areas of interest, REF/DIF1 utilizes a coarse scale user-specified reference grid and a fine scale subgrid that can have many times the resolution of the reference grid. The principal purpose of the subgrid is to provide enough computational points to the numerical model to preserve accuracy. The user specifies the number of subgrid divisions in the y direction with the parameter *nd*. If $nd = 1$, then the subgrid spacing in the y direction is the same as the reference grid. If $nd = 2$, then the model uses twice as many computational points in the y direction as there are in the reference grid. If the input flag, *isp*, is set to 1, the subgrid computational grid subdivisions must be specified in the array *isd*.

3.6.5 Damping

When the large-angle parabolic approximation is used as a basis for the computation of wave fields around islands, the presence of wave breaking and resulting sharp lateral variations in wave height leads to the generation of high wave number spectral components in the computed complex amplitude A in the lateral (y) direction. Kirby (1986a) has shown that these components have propagation velocities that can become large in an unbounded fashion; as a result, they can propagate across the grid, filling the computational domain with high wave number noise.

In the present version of REF/DIF1, a new algorithm that was recently developed by Kirby (1993) is being used to attempt to control the high-frequency noise generated by the breaking process. Reference should be made to that document for a description of the algorithm. The damping is built into the computational algorithm and is turned on automatically if breaking has started anywhere in the computational domain.

4.0 CSCI ARCHITECTURAL DESIGN

4.1 CSCI Components

The REF/DIF1 design architecture consists of four Primary Software Units (PSU) that are sequenced to compile, execute, and then manipulate data for the REF/DIF1 model. The four PSUs are the Compile Model Primary Software Unit that compiles the other SUs in accordance with the model-size parameters; the Pre-Model Primary Software Unit that produces the input data file for the model; the REF/DIF1 Primary Software Unit that constitutes the actual REF/DIF1 model; and the Post-Processing Primary Software Unit that allows manipulation of the output data files. The Pre-Model and Post-Processing Software Units are not PSUs *per se*; instead, they define the function of the associated SUs that have those roles within the sequence of execution.

Each of these four PSUs consists of a number of component SUs as shown in Fig. 4.1-1. The Compilation PSU contains two SUs. They are the Included Parameters and the Input File Name files. The Pre-Model PSU also consists of two SUs. They are the Convert Old Input Files and the Create New Input Files Software Units. The REF/DIF1 Primary Software Unit consists of three internal SUs. They are the Read in Reference Data, Read in Wave Data, and the Model Software Units. The last PSU, the Post-Processing Primary Software Unit, consists of two Software Units. They are the Make Surface and Make HDF File Software Units.

4.2 Concept of Execution

The flow of the entire Refraction/Diffraction Model showing the dynamic relationships of the four PSUs is illustrated in Fig. 4.2-1. The model is first compiled before being executed, ensuring

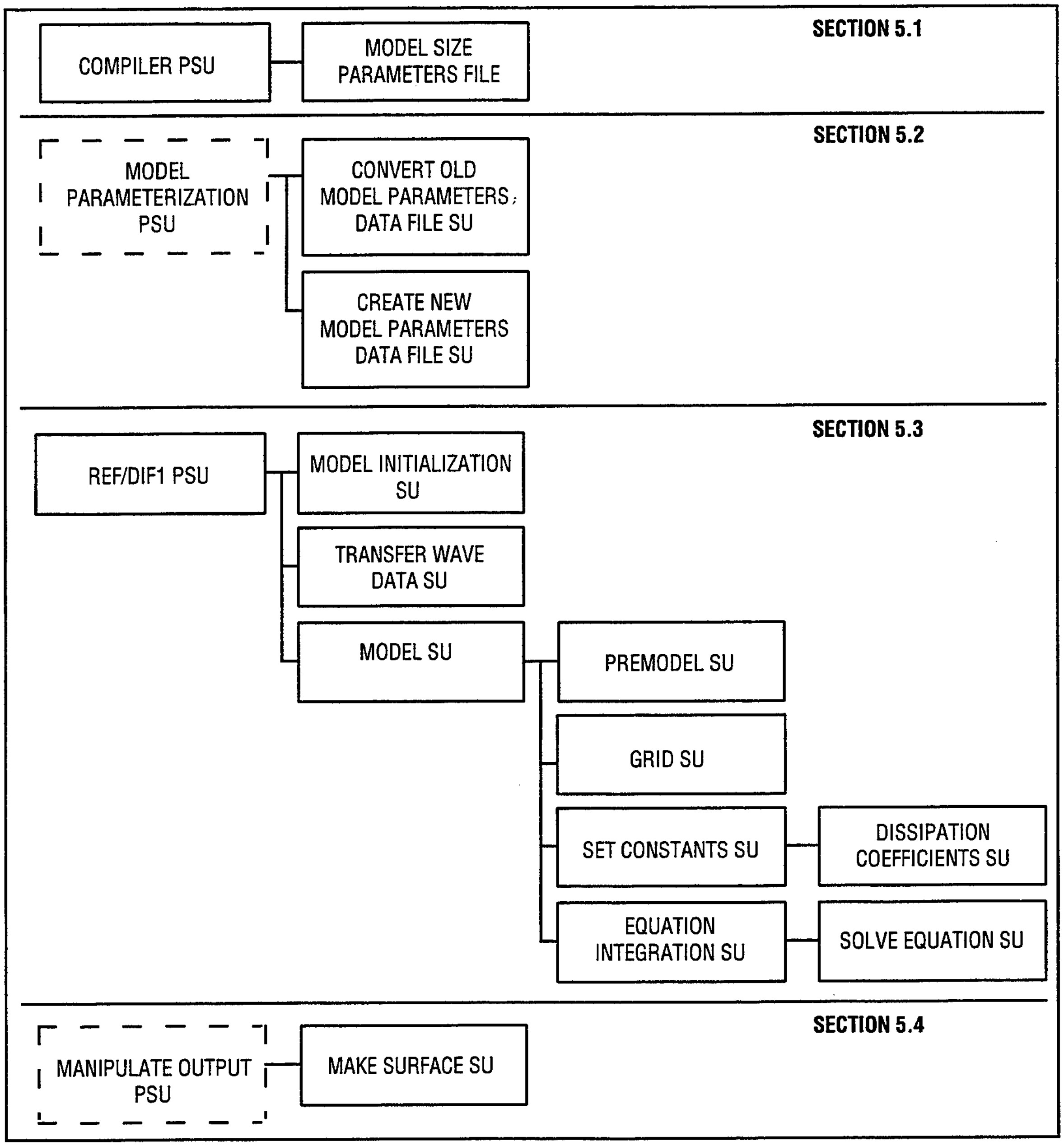

Fig. 4.1-1 — SU Links to PSUs. The PSUs and major constituent SUs are integrated as shown above. PSU discussions are located in the SDD Sections highlighted on the right side.

that all SUs are similarly constructed. Prior to running the REF/DIF1 model itself, a Version 2.5 input data file has to be constructed, either from a Version 2.4 or earlier input data file or from user inputs. After execution of the model, the post-processing layer produces output in different formats as needed.

4.2.1 Concept of Execution, Compiler Primary Software Unit

The Included Parameters and Input File Name files must be edited prior to compilation of the model. The user then directs the compilation of the model using the "Make" command.

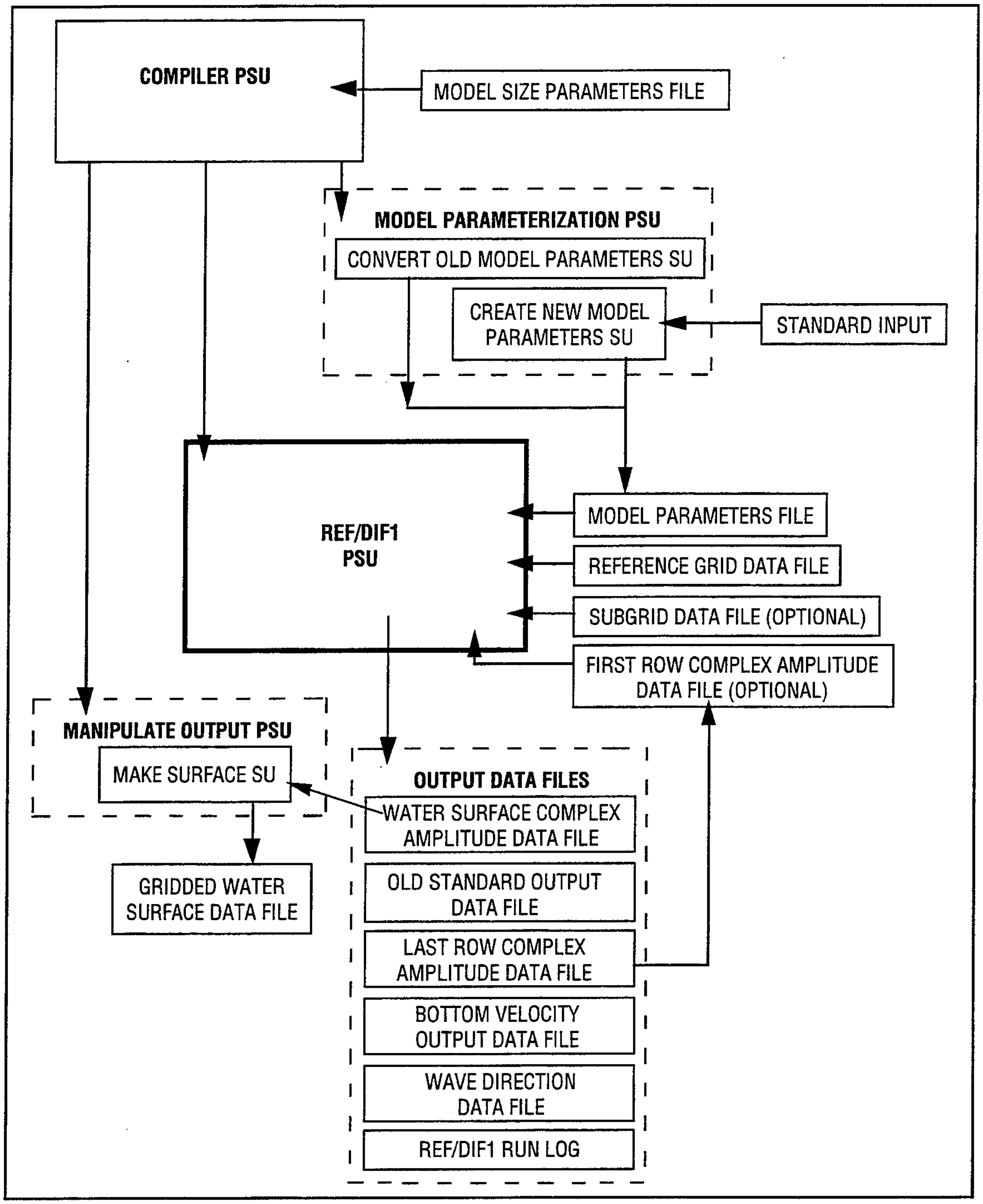

Fig. 4.2-1 — REF/DIF1 Dynamic Flow Chart. The sequence of execution of the Primary Software Units and the arrangement of input and output data files between the PSUs and major SUs are highlighted.

4.2.2 Concept of Execution, Model Parameterization Software Unit

Both the Convert Old Input Files and the Create New Input Files Software Units create a Version 2.5 input data file. The Convert Old Input File Software Unit reads a Version 2.4 or older input data file (filename provide by the Input File Name file within the Compile Model Software

Unit) and produces a Version 2.5 input data file named *indat.new*. This new file has to be renamed or the Input File Name File modified prior to execution of the REF/DIF1 PSU.

4.2.3 Concept of Execution, REF/DIF1 Primary Software Unit

The actual REF/DIF1 Model is a single primary software unit. The internal execution of the model is sequenced by the internal structure of this PSU. The internal sequence of flow for this PSU and its relationship to the external interfaces is shown in Fig. 4.2.3-1.

The first SU shown, the Premodel SU, is only executed once. It calculates the computational grids and spacing in the y direction. The next three SUs pictured are each executed once for each frequency. The Model Initialization and Transfer Wave SUs read in model parameters, switches, and external data that control execution within the PSU. Execution within the Model SU is more complex and is shown on Fig. 4.2.3-2. For each frequency, and then for each reference grid within the area of study, the three major SUs are executed. The Grid and Set Constants SUs prepare the parameters of the reference grid under evaluation. The Equation Integration SU combines all of these elements and then computes the complex amplitude values along each grid row. These solution sets are converted (if necessary) and written to external output. Once each reference grid under each frequency has been evaluated control returns to the REF/DIF1 PSU that closes the output files and ends execution.

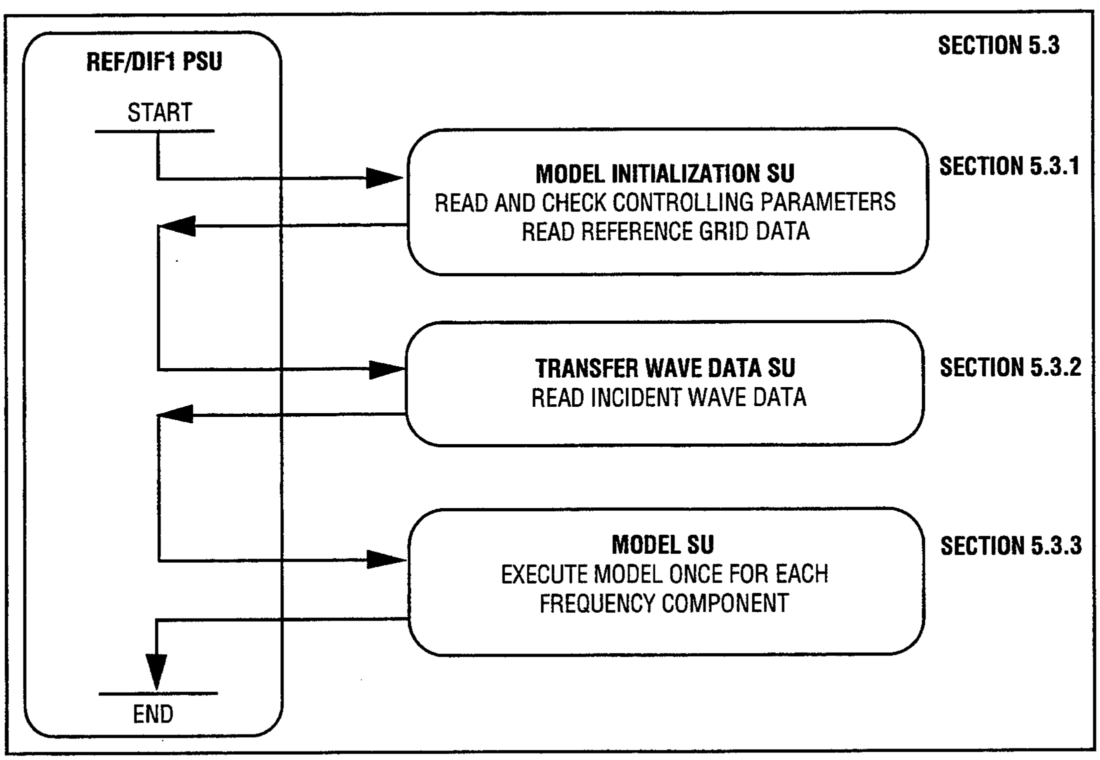

Fig. 4.2.3-1 — Sequence of flow within the REF/DIF1 Primary Software Unit. PSU and SU discussions are located in the SDD Sections highlighted on the right side.

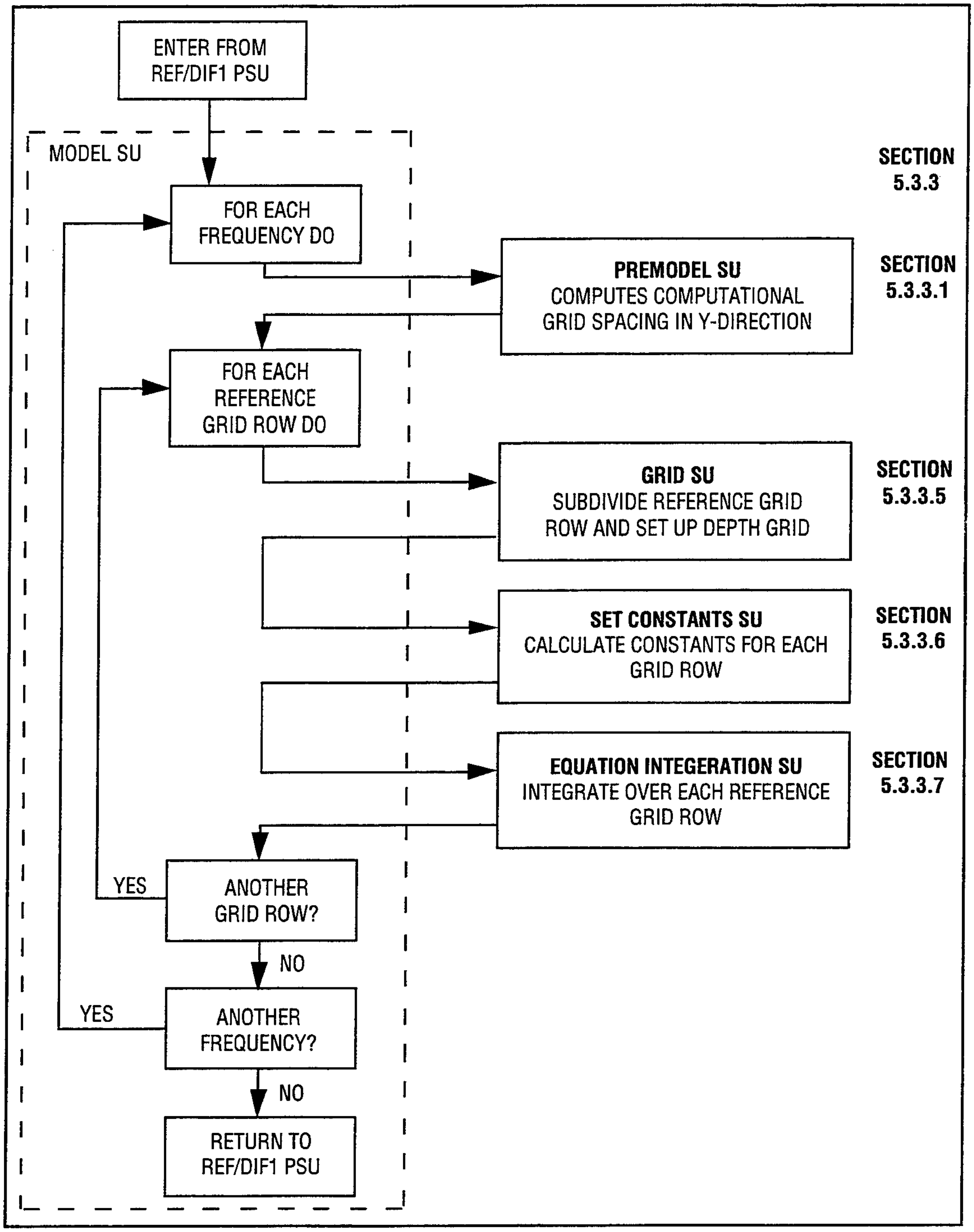

Fig. 4.2.3-2 — Model SU Execution Sequence. The major decisions and SUs within the Model SU are highlighted and the Section within the SDD in which they are discussed are in bold along the right side.

4.2.4 Concept of Execution, Output Manipulation Primary Software Unit

There is currently only one independent SU that constitutes this PSU. The Make Surface SU converts the Water Surface Complex Amplitude Data File produced by the REF/DIF1 PSU into an ASCII format file suitable for Matlab use. The Make Surface SU is contained within a single complete SU.

4.3 Interface Design

4.3.1 Interface Identification and Diagrams

External interfaces to the Combined Refraction/Diffraction Model are incorporated into the dynamic flow chart on Fig. 4.2-1 and are marked bold. Each of the blocks corresponds to individual user-modified or created and program-modified or created logical units, with the exception of the Input Data Files block and the Output Data Files block.

Disk-File Interfaces

Parameter File – Single-line parameter statement dictating the maximum size of arrays within the model.

Input Filename File – Included file with short procedure providing the name of the Model Parameters File (either Old or Version 2.5).

Old Model Param's File – File with model-controlling parameters for Version 2.4 or older REF/DIF1 program.

Model Parameters File – File with model-controlling parameters for Version 2.5 REF/DIF1.

REF/DIF1 Input Files

- *fname1* – File with reference grid data.
- *fname3* – File containing user-specified subgrids.
- *fname4* – File with user-specified complex amplitude on row 1 (for *iinput* = 2).

REF/DIF1 Output Files

- *fname2* – File containing old standard output data.
- *fname5* – File with last-row complex amplitude data (for *ioutput* = 2).
- *fname6* – File with complex amplitude data for constructing an image of the instantaneous water surface at the computational resolution (optional).
- *fname7* – File containing magnitudes of bottom velocity at reference gridpoints (optional).
- *fname8* – File with wave directions θ in degrees at reference gridpoints.
- *fname10* – Run log for REF/DIF1 program.
- *fname11* – File with wave heights at reference gridpoints.
- *fname15* – File with tide-corrected depths at reference grid locations.

Surface Output File – Regularly spaced ASCII file produced from conversion of *fname6*.

Standard I/O Interfaces

Standard Input

User-input Namelist parameters.
User-input name of Surface Output File.

Standard Output

These interfaces are described in Secs. 4.3.2–4.3.4.

The internal interface between SUs within the REF/DIF1 PSU involve the execution of subroutines and the exchange of data via subroutine argument lists and via common blocks. The relationship among software units is shown in Fig. 4.1-1. Their exchange of data via argument list is described for each software unit in Sec. 5.3. The common block interfaces within the REF/DIF1 PSU are described in Sec. 4.3.5.1. A total of 12 common blocks are used to exchange data among software units and are listed:

Common Block Interfaces

Common Block /ref1/
Common Block /ref2/
Common Block /ref3/
Common Block /ref4/
Common Block /block1/
Common Block /con1/
Common Block /con2/
Common Block /nlin/
Common Block /wav1/
Common Block /wav2/
Common Block /comp/
Common Block /names/

The input and output of data between software units via common block for the REF/DIF1 PSU is described in Sec. 5.3. Appendix A contains a complete list of variables contained in the REF/DIF1 common blocks.

4.3.2 Disk File Input Interfaces

4.3.2.1 Model Size Parameters File

The numerous programs constituting the REF/DIF1 system each require that the numerous arrays be properly sized prior to compilation. The values that would otherwise have to be changed in each program were isolated in a single-line file that is included during compilation of the model system. All of the parameters are contained within the Model Size Parameter file (param.h), as presented:

FILE *param.h* - **Input Model Array Size Parameters**

Logical Unit Number: N/A **File Access Method:** file included during compilation

Record	Data	Description
	ixr,	Maximum number of reference grid rows
	iyr,	Maximum number of reference grid columns
	ix,	Maximum number of computational grid rows
	iy,	Maximum number of computational grid columns
	ncomp	Maximum number of frequency components to be run

4.3.2.2 Old Model Parameters File

Version 2.5 of the REF/DIF1 Model contains a program that will read in Version 2.4 or earlier files with the controlling model parameters and will produce a Version 2.5 Model Parameters file.

FILE *fnamein* – **Old Model Parameters File**

Logical Unit Number: 5 **File Access Method:** formatted, sequential access

Record	Data	Description
3	iun(3),	Input file unit numbers (*no longer used)*
1	mr,	Reference grid dimensions in the x direction (maximum value *ixr*).
	nr,	Reference grid dimensions in the y direction (maximum value *iyr*)
	iu,	Switch for physical units (*iu* = 1, MKS, *iu* = 2, English)
	ntype,	Switch for nonlinearity (*ntype* = 0, linear model; *ntype* = 1, composite model; *ntype* = 2, Stokes wave model)
	icur,	Switch for input current data (*icur* = 0, no currents input; *icur* = 1, currents input)
	ibc,	Boundary condition switch (*ibc* = 0, closed boundaries; *ibc* = 1, open boundaries)
	dxr,	Reference grid x spacing
	dyr,	Reference grid y spacing
	dt,	Depth tolerance value
	ispace,	Switch controlling subdivisions (*ispace* = 0, program attempts its own x subdivisions; *ispace* = 1, user specifies x subdivisions)
	nd,	Number of y direction subdivisions
	md(ixr)	Subdivisions in the x direction
3	iff(3),	Dissipation switches (*iff(1)* = 1, turn on turbulent boundary layer; *iff(2)* = 1, turn on porous bottom damping; *iff(3)* = 1, turn on laminar boundary layers; no damping if all values are zero)
1	isp,	Switch for user-specified sub-grid specifications (*isp* = 0, no subgrids to be read; *isp* = 1, subgrids will be read)
	iinput	Specifies whether the program or the user will generate the first row of complex amplitude *A* values (*iinput* = 1, the program constructs *A* based on the input conditions specified below; if *iinput* = 2, the user must specify *A* in an external data file *fname4*)
	ioutput	Specifies whether the last computed row of complex amplitudes *A* are to be saved in file *fname5* (*ioutput* = 1, the values are not saved; if *ioutput* = 2, the values of *A* are saved)
	iinput = 1;	
	Read iwave	Switch for wave field type (*iwave* = 1, discrete wave components, *iwave* = 2, directional spreading model)
	nfreqs	Number of frequency components to be run
	iwave = 1;	
	Read freqs(ncomp)	Wave period for each frequency component (*ncomp* values must be given)
	tide(ncomp)	Tidal offset for each frequency component (*ncomp* values must be given)
	nwavs(ncomp)	Number of wave components for each frequency (*ncomp* values must be given
	amp(ncomp, ncomp)	Amplitude for each component wave
	dir(ncomp, ncomp)	Direction in degrees relative to x-axis for each wave component
	iwave = 2	
	Read thet0	Central direction for the model spectrum
	freqs(ncomp)	Wave period for each frequency component (*ncomp* values must be given)

	tide(ncomp)	Tidal offset for each frequency component (*ncomp* values must be given)
	edens(ncomp)	Variance density for each frequency component
	nwavs(ncomp)	Direction spreading factor
	nseed	Seed value for the random number generator (between 0 and 9999)
iinput = 2		
Read	freqin	Wave period for the single-frequency component
	tidein	Tidal offset for the single-frequency component

4.3.2.3 Model Parameters File

The REF/DIF1 Model contains two program that will create a Version 2.5 Model Parameters file with the controlling model parameters. The Version 2.5 Model Parameters file is then read into the REF/DIF1 PSU. The input parameters control the sequence and calculations within the PSU and dictate the opening and closing of files.

FILE *fnamein* - **Model Parameters File**

Logical Unit Number: 5 **File Access Method:** sequential by Namelist

Record	Data	Description
1	Namelist/fnames/	
	fname1	Reference grid data file
	fname2	Old standard output data file
	fname3	File with user-specified subgrids
	fname4	File with user-specified complex amplitude on last row (for *iinput* = 2)
	fname5	Output file with complex amplitude on last row (for *ioutput* = 2)
	fname6	File with complex amplitude data for constructing an image of the instantaneous water surface at the computational resolution (optional)
	fname7	File with magnitude of bottom velocity at reference gridpoints (optional)
	fname8	File with wave directions θ in degrees at reference gridpoints
	fname9	Not used at present
	fname10	File with run log for REF/DIF1 PSU
	fname11	File with wave heights at reference grid locations
	fname12	File with S_{xx} components at reference grid locations (not yet implemented)
	fname13	File with S_{xy} components at reference grid locations (not yet implemented)
	fname14	File with S_{yy} components at reference grid locations (not yet implemented)
	fname15	File with tide-corrected depths at reference grid locations
1	Namelist/ingrid/	
	mr	Reference grid dimension in x direction (maximum value *ixr*)
	nr	Reference grid dimension in y direction (maximum value *iyr*)
	iu	Switch for physical units (*iu* = 1, MKS; *iu* = 2, English)
	ntype	Switch for nonlinearity (*ntype* = 0, linear model; *ntype* = 1, composite model; *ntype* = 2, Stokes wave model)
	icur	Switch for input current data (*icur* = 0, no currents input; *icur* = 1, currents input)
	ibc	Boundary condition switch (*ibc* = 0, closed boundaries; *ibc* = 1, open boundaries)
	dxr	Reference grid x spacing
	dyr	Reference grid y spacing
	dt	Depth tolerance value

	ispace	Switch controlling subdivisions (*ispace* = 0, program attempts its own x subdivisions; *ispace* = 1, user specifies x subdivisions)
	nd	If *nd*⇓0: Number of y direction subdivisions If $nd < 0$: Indicates that y direction subdivisions are to be computed by the model
	iff(3)	Dissipation switches (*iff(1)* = 1, turn on turbulent boundary layer; *iff(2)* = 1, turn on porous bottom damping; *iff(3)* = 1, turn on laminar boundary layers; no damping if all values are zero)
	isp	Switch for user-specified sub-grid specifications (*isp* = 0, no subgrids to be read; *isp* = 1, subgrids will be read)
	iinput	Specifies whether the program or the user will generate the first row of complex amplitude *A* values (*iinput* = 1, the program constructs *A* based on the input conditions specified below; if *iinput* = 2, the user must specify *A* in an external data file *fname4*)
	ioutput	Specifies whether the last computed row of complex amplitudes *A* are to be saved in file *fname5* (*ioutput* = 1, the values are not saved; if *ioutput* = 2, the values of *A* are saved)
1	Namelist/nddata/	(If $nd < 0$, this value is read)
	mindep	Depth to be used to compute subdivisions in the y direction
1	Namelist/inmd/	(If *ispace* = 1, these values are read)
	md(ixr)	Subdivisions in the x direction
1	Namelist/waves1a/	(If *iinput* = 1, these values are read)
	iwave	Switch for wave field type (*iwave* = 1, discrete wave components, *iwave* = 2, directional spreading model)
	nfreqs	Number of frequency components to be run
1	Namelist/waves1b/	(These values are read if *iwave* = 1)
	freqs(ncomp)	Wave period for each frequency component (*ncomp* values must be given)
	tide(ncomp)	Tidal offset for each frequency component (*ncomp* values must be given)
	nwavs(ncomp)	Number of wave components for each frequency (*ncomp* values must be given
	amp(ncomp,ncomp)	Amplitude for each component wave
	dir(ncomp,ncomp)	Direction in degrees relative to x-axis for each wave component
1	Namelist/waves1c/	(These values are read if *iwave* = 2)
	thet0	Central direction for the model spectrum
	freqs(ncomp)	Wave period for each frequency component (*ncomp* values must be given)
	tide(ncomp)	Tidal offset for each frequency component (*ncomp* values must be given)
	edens(ncomp)	Variance density for each frequency component
	nwavs(ncomp)	Direction spreading factor
	nseed	Seed value for the random number generator (between 0 and 9999)
1	Namelist/waves2/	(If *iinput* = 2, these values are read)
	freqin	Wave period for the single-frequency component
	tidein	Tidal offset for the single-frequency component

4.3.2.4 Input Data Files

There are three other files that provide input data to the REF/DIF1 model. The names of the three are provided to the REF/DIF1 PSU by the Model Parameters file.

4.3.2.4.1 Reference Grid Data File. This file (*fname1*) provides reference grid bathymetry and current velocities to REF/DIF1.

FILE *fname1* - **Reference Grid Data File**

Logical Unit Number: 1 **File Access Method:** formatted, sequential access

Record	Data	Description
1	dr(mr,nr)	Reference grid bathymetry
1	ur(mr,nr)	Reference grid current velocities in the x direction (read if *icur* = 1)
1	vr(mr,nr)	Reference grid current velocities in the y direction (read if *icur* = 1)

4.3.2.4.2 Sub-Grid Data File. This file (*fname3*) provides subgrid bathymetry to the REF/DIF1 PSU.

FILE *fname3* - **Sub-Grid Data File**
Logical Unit Number: 2 **File Access Method:** formatted, sequential access

Record	Data	Description
1	isd(mr,nr)	Sub-grid bathymetry

4.3.2.4.3 First-Row Complex Amplitude File. This file (*fname4*) provides initial first-row complex amplitude values to the REF/DIF1 PSU.

FILE *fname4* - **First-Row Complex Amplitudes File**

Logical Unit Number: 11 **File Access Method:** formatted, sequential access

Record	Data	Description
1	a(1,n)	First-row complex amplitude values

4.3.3 Disk File Output Interfaces

4.3.3.1 REF/DIF1 Output Files

There are a total of eight output files that the REF/DIF1 PSU can produce. Of these, five are optional. The names of the eight are provided to that PSU by Namelist/fnames/ within the Model Parameters File.

4.3.3.1.1 Old Standard Output Data File. This file (*fname2*) saves the output data from the REF/DIF1 PSU in Version 2.4 format.

FILE *fname2* - **Old Standard Output Data File**

Logical Unit Number: 3 **File Access Method:** unformatted, sequential access

Record	Data	Description
1		Data currently stored in many of the data files described below, the Water Surface Complex Amplitude Data File, the Bottom Velocity Output Data File, and the Wave Direction Output Data File, is stored in this file.

4.3.3.1.2 Last-Row Complex Amplitude Data File. This file (*fname5*) saves the last-row complex amplitude data output from the REF/DIF1 PSU. This file can be used to initialize the first row of a different run of REF/DIF1. This file will be produced only if *ioutput* = 2.

FILE *fname5* - **Old Standard Output Data File**

Logical Unit Number: 33 **File Access Method:** formatted, sequential access

Record	Data	Description
1	a(m,n)	Last-row complex amplitude data

4.3.3.1.3 Water Surface Complex Amplitude Data File. This file (*fname6*) contains the complex amplitude data for constructing an image of the instantaneous water surface at the computation resolution. Creation of the file is optional.

FILE *fname6* - **Water Surface Complex Amplitude Data File**

Logical Unit Number: 8 **File Access Method:** formatted, sequential access

Record	Data	Description
1	a(m,n)	Last-row complex amplitude data

4.3.3.1.4 Bottom Velocity Output Data File. This file (*fname7*) saves the REF/DIF1-produced bottom velocity magnitudes at each reference gridpoint. It is produced only if a filename is supplied to the program.

FILE *fname7* - **Old Standard Output Data File**

Logical Unit Number: 17 **File Access Method:** formatted, sequential access

Record	Data	Description
1	bottomu(m,n)	Magnitude of bottom velocity at reference gridpoints

4.3.3.1.5 Wave Direction Data File. This file (*fname8*) saves the direction of wave propagation for each point on the reference grid. This output file is always produced.

FILE *fname8* - **Wave Direction Data File**

Logical Unit Number: 9 **File Access Method:** formatted, sequential access

Record	Data	Description
1	thet(m,n)	Direction of wave propagation for each point on the reference grid

4.3.3.1.6 REF/DIF1 Run Log.This file (*fname10*) contains a log of the execution of REF/DIF1. The file is written to throughout the REF/DIF1 PSU.

FILE *fname10* - **REF/DIF1 Run Log**

Logical Unit Number: 10 **File Access Method:** unformatted access

Record	Data	Description
1	Text	Sequential list of actions, flags, and errors during execution of the REF/DIF1 PSU

4.3.3.1.7 Wave Height Output Data File. This file (*fname11*) saves the output wave heights from each reference gridpoint. The REF/DIF1 PSU always produces this file.

FILE *fname11* - **Wave Height Output Data File**

Logical Unit Number: 12 **File Access Method:** formatted, sequential access

Record	Data	Description
1	2|A|	Wave height at each reference gridpoint

4.3.3.1.8 Tide-Corrected Depth Output Data File. This file (*fname15*) contains the tide-corrected water depth at each reference gridpoint. This output from the REF/DIF1 PSU is always produced.

FILE *fname15* - **Data File**

Logical Unit Number: 16 **File Access Method:** formatted, sequential access

Record	Data	Description
1	d(m,n)	Tide-corrected water column height for each reference gridpoint

4.3.3.2 Gridded Water Surface Data File

This file (*fileout*) contains an instantaneous snapshot of the water surface in a regularly spaced ASCII file, suitable for direct reading into Matlab format.

FILE *fileout* - Data File

Logical Unit Number: 11 File Access Method: formatted, sequential access

Record	Data	Description
1	surface(nx,ny)	Regularly spaced grid with water surface contours

4.3.4 Standard I/O Interfaces

4.3.4.1 Standard Input

4.3.4.1.1 Create New Input Files SU Standard Input. Within the Create New Input Files SU up to six sets of Namelist parameters are user-input via standard device. The Namelists are listed below and were described previously in Sec. 4.3.2.4.

Namelist/ingrid/
Namelist/nddata/
Namelist/inmd/
Namelist/fnames/
Namelist/waves1a/
Namelsit/waves1b/
Namelist/waves1c/
Namelist/waves2/

4.3.4.1.2 Make Surface SU Standard Input. Within the Make Surface SU the name of the file to receive ASCII data converted from the REF/DIF1 surface height output file is user-input via standard device.

4.3.5 Common Block Interfaces

4.3.5.1 REF/DIF1 PSU Common Block Interfaces

The common block interfaces among the SUs within the REF/DIF1 PSU are shown in Fig. 4.3.5.1-1.

5.0 CSCI DETAILED DESIGN

All of the SUs within REF/DIF1 are written in Fortran, with the exception of the Compiler PSU. That PSU uses the Unix Make command.

5.1 Compiler Primary Software Unit

The Compiler Primary Software Unit compiles the other PSUs of the REF/DIF1 model in accordance with the values defined by param.h. The compilation ensures that all arrays used in the executable files of the other three PSUs are dimensioned according to the up-to-date model size parameters.

Design Decisions and Algorithms

The Compiler PSU is structured to use the Unix "Make" command with the file "Makefile" included in the REF/DIF1 set. The param.h program element needs to be modified prior to compilation. Then using a structured set of Make commands, the user can dictate which SUs or PSUs are compiled.

The different elements of this PSU are implemented by using the following commands:

Command	*Executables Produced*
make refdif1	REF/DIF1 PSU
make surface	Surface SU
make all	Both of the above
make convert	Convert Old Input Parameters Data File SU
make create	Create New Input Data File SU
make clean	Remove *.o files

Constraints and Limitations

None.

Input Data Elements

None.

External Input

One file (param.h) is incorporated into the executable code of the other PSUs by this PSU and is detailed in Sec. 4.3.2.

Output Data Elements

None.

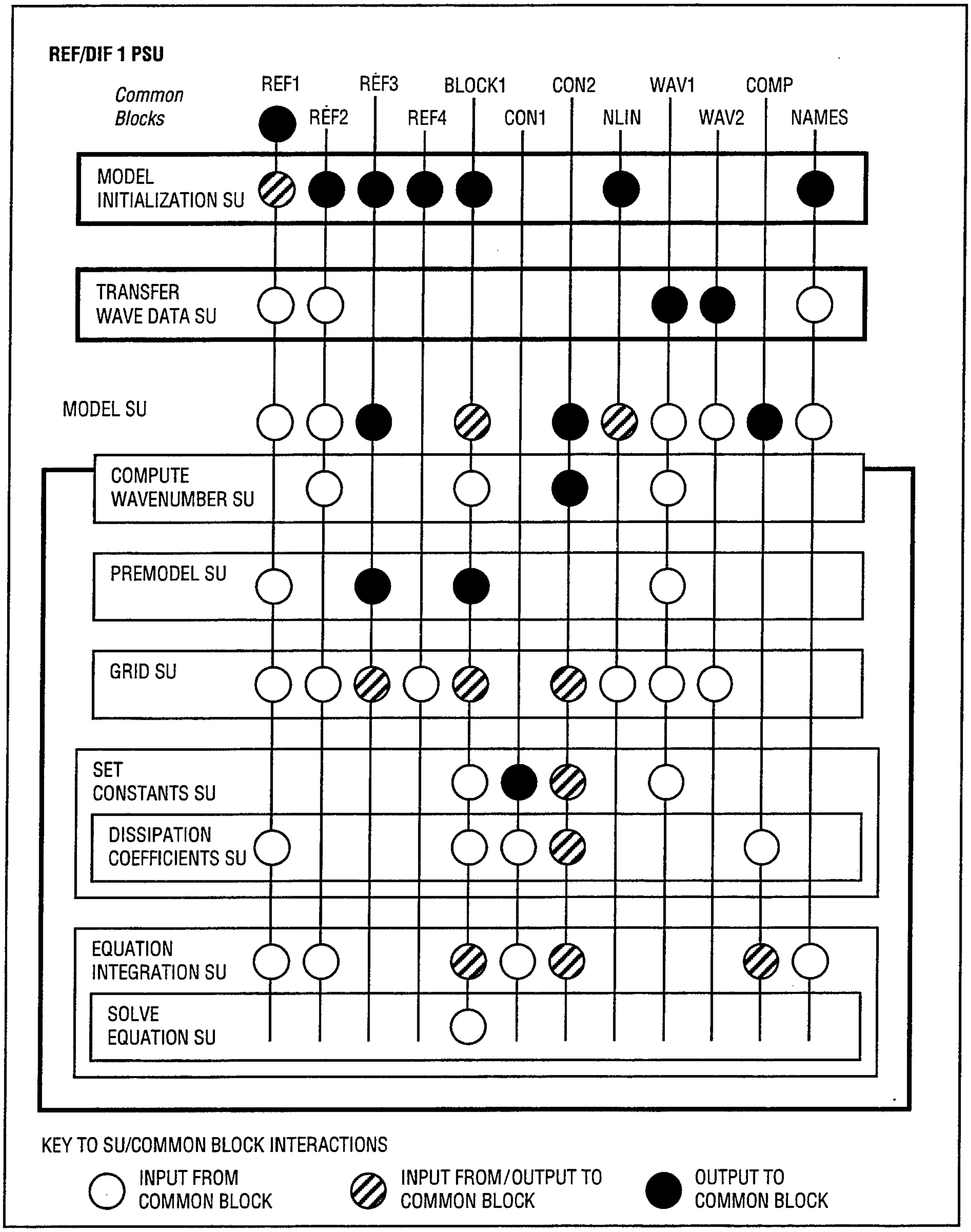

Fig. 4.3.5.1-1 — Common block interface within the REF/DIF1 PSU. The common blocks within the REF/DIF1 PSU link the numerous SUs together. The above diagram shows what is later outlined in the text: the Model Initialization SU and the Transfer Wave Data SU (and the Model SU itself) initialize many of the elements within the common blocks for later use by the actual computational elements, notably the Grid SU, the Set Constants SU, and the Equation Integration SU.

External Output

The executable code for the other three PSUs is produced by this PSU.

5.2 Model Parameterization Primary Software Unit

This PSU is defines a logical step prior to the REF/DIF1 PSU and is not in itself an executable step. The two SUs within this PSU produce an updated Model Parameters Data File and are therefore grouped together because of their similar output. The Convert Old Model Parameters Data File SU reads in a Version 2.4 or earlier Model Parameters Data File and is discussed in Sec. 5.2.1. The Create New Model Parameters Data File SU relies on Standard Input to build the Version 2.5 Model Parameters Data File. The latter SU is discussed in greater detail in Sec. 5.2.2.

Design Decisions and Algorithms

None.

Constraints and Limitations

None.

Input Data Elements

None.

External Input

A Version 2.4 Model Parameters Data File (discussed in Sec. 4.3.2.3) is used as input for the Convert Old Model Parameters Data File, as discussed in Sec. 5.2.1. Standard Input (discussed in Sec. 4.3.4.1.1) is required for the Create New Model Parameters Data File SU as described in Sec. 5.2.2.

Output Data Elements

None.

External Output

A Version 2.5 Model Parameters Data File (discussed in Sec. 4.3.2.4) is produced within this PSU.

5.2.1 Convert Old Model Parameters Data File Software Unit

One of the new features of the Version 2.5 REF/DIF1 program is the use of an Model Parameters Data File in Namelist format. As the new program cannot read the older Model Parameters Data Files, a program to convert the old versions to the new was included in the REF/DIF1 package.

Design Decisions and Algorithms

This SU sequentially reads in the data from logical unit 5, the Old Model Parameters Data File, and writes in Namelist format to logical unit 6, the New Model Parameters Data File.

Constraints and Limitations

This SU must be recompiled every time the size of the model arrays (in param.h) are changed.

Input Data Elements

None.

External Input

Input data elements for this SU is read from the Old Model Parameters Data File, described in Sec. 4.3.2.3.

Output Data Elements

None.

External Output

The Namelist parameters output from this SU are written to the Model Parameters Data File, as described in Sec. 4.3.2.4.

5.2.2 Create New Model Parameters Data File Software Unit

This SU is included in the REF/DIF1 Version 2.5 package and builds a Model Parameters Data File through standard input.

Design Decisions and Algorithms

This SU sequentially reads in the data from standard input and writes to logical unit 10, the new Model Parameters Data File in Namelist format. The values of *iinput* and (if $iinput = 1$) *iwave* determine some sequence of flow within the program. If $iinput = 1$ Namelist/waves1a/ is read; if $iwave = 1$, Namelist/waves1b/ is the read, else Namelist/waves1c/ is read if $iwave = 2$. If $iinput = 2$ Namelist/waves2/ is read.

Constraints and Limitations

This SU must be recompiled every time the size of the model arrays (in param.h) is changed.

Input Data Elements

Input data for this SU is read from Standard Input, as detailed in Sec. 4.3.4.1.1.

External Input Files

None.

Output Data Elements

None.

External Output Files

The Namelist parameters are written to the Model Parameters File, as described in Sec. 4.3.2.4.

5.3 REF/DIF1 Primary Software Unit

The Main Software Unit drives the REF/DIF1 Model. It initializes the physical system constants, calls the major subroutines, and closes the output files after model execution.

Design Decisions and Algorithms

The REF/DIF1 PSU is designed to sequentially call the three major SUs of the REF/DIF1 Model. The Model Initialization Software Unit (Sec. 5.3.1) is called first. The Transfer Wave Data Software Unit (Sec. 5.3.2) is called next, and finally, the Model Software Unit (Sec. 5.3.3) is called. The output data files produced by the Model SU are then closed.

Constraints and Limitations

This PSU must be compiled every time the size of the model arrays (in param.h) is changed.

Input Data Elements

None.

External Input

None.

Output Data Elements

/REF1/

dconv(2)	real	Physical unit conversion factors

External Output

The output data files from the Model Software Unit are closed by this PSU.

5.3.1 Model Initialization Software Unit

This SU (subroutine inref) initializes the REF/DIF1 model. Some of the model-controlling parameters are read, the output files are opened, the array sizes are quality checked for consistency, and the reference grid bathymetry and current data are read. If necessary, the sub-grid bathymetry data is read in. A continuous summary of SU actions are recorded in the REF/DIF1 Run Log.

Design Decisions and Algorithms

This SU sequentially reads in the Namelist parameters from logical unit 5 (file *fnamein*, defined in an external subroutine by the original code, and now defined internally here) and acts accordingly. Namelist fnames is read first, and then the input and output files defined within that Namelist are opened. Once all of the files are opened, a continuous record of SU actions is written to logical unit 10 (file *fname*, the REF/DIF1 Run Log). Namelist ingrid is read next. If $nd < 0$, Namelist nddata is read. If f $ispace = 1$, Namelist inmd is read.

The Model Initialization SU then begins the first of many data quality-check tests. Array sizes, the physical units parameter, and the sub-grid dimensions are all evaluated. Errors in the size of the model arrays or the sub-grid dimensions will cause the program to stop. If the physical units parameter (*iun*) does not equal 1 or 2, the parameter will be reset to the default, the Metric Kelvin System (MKS) physical system.

The SU then reads the depth grid and grid velocities from logical unit 1 (file *fname1*, the Reference Grid Data File). If necessary, the depth and current velocities are converted to MKS units. The depth and current grids are then checked for extreme gradients. Coordinates for the interpolated are then computed and written to logical unit 3 (file *fname3*, the Old Standard Output Data File).

The damping switches are then quality checked, and if the value is illegal, sets the *iff* value to 0, i.e., off.

Finally, the sub-grid sizing is quality checked. The program determines whether user-input or program-determined sub-grid sizing will be used, and if both switches are set to on, the inconsistency is noted in the Run Log. Size of the input arrays is then compared to the model.

Constraints and Limitations

None.

Input Data Elements

/REF1/

dconv(2)	real	Physical unit conversion factors.

External Input

Namelist parameters are read from the Model Parameters File (described in Sec. 4.3.2.4). The depth grid and current data are read from the Reference Grid Data File (described in Sec. 4.3.2.5.1).

Output Data Elements

/REF1/

mr	integer	Reference grid x dimension
nr	integer	Reference grid y dimension
ispace	integer	Switch controlling x subdivisions
nd	integer	Indicator of y direction subdivisions
md(ixr)	integer	Array with x dimension subdivisions per reference grid
iu	integer	Switch for physical units
iff(3)	integer	Dissipation switches
icur	integer	Switch for input current data
ibc	integer	Boundary conditions switch
Mindep	real	Depth to be used to compute y direction subdivisions

/REF2/

dr(ixr,iyr)	real	Reference grid bathymetry data
ur(ixr,iyr)	real	Reference grid x direction current data
vr(ixr,iyr)	real	Reference grid y direction current data
iun(8)	integer	Array with unit numbers for external files
iinput	integer	Switch for user-input, first-row complex amplitude *A* values
ioutput	integer	Switch for last-row complex amplitude *A* values

/REF3/

dxr	real	Reference grid x spacing
dyr	real	Reference grid y spacing
xr(ixr)	real	Spacing of reference grid in x direction
yr(iyr)	real	Spacing of reference grid in y direction

/REF4/

isd(ixr,iyr)	real	Switch for user-supplied subgrid data

/NLIN/

ntype	integer	Switch for nonlinearity

/NAMES/

fname1	character	Reference grid data filename
fname2	character	Old standard output data filename
fname3	character	User-specified sub-grid data filename
fname4	character	User-specified, first-row complex amplitude data filename
fname5	character	Last-row complex amplitude data filename
fname6	character	Complex amplitude data filename
fname7	character	Magnitude of bottom velocity data filename
fname8	character	Wave direction data filename
fname10	character	REF/DIF1 run log filename
fname11	character	Reference grid wave height filename
fname15	character	Reference grid tide-corrected depth data filename.

External Output

The SU sequence of actions and information about the model run are recorded in the REF/DIF1 Run Log (described in Sec. 4.3.3.1.6). The model size is also written to the Old Standard Output Data File (described in Sec. 4.3.3.1.1). The other output files from REF/DIF1 are opened but are not written to.

5.3.2 Transfer Wave Data Software Unit

This SU (subroutine inwave) reads in data specifying the initial wave field along the first row of reference gridpoints. Namelist data is read from the input file and converted to MKS units if necessary. At the end of this subroutine, control is returned to the REF/DIF1 SU.

The Namelist groups read in are determined by the values of switches *iinput* and *iwave*, as follows:

If iinput = 1, read Namelist/waves1a/
If iwave = 1, read Namelist/waves1b/
If iwave = 2, read Namelist/waves1c/
If iinput = 2, read Namelist/waves2/

Design Decisions and Algorithms

The actions of the Transfer Wave Data SU are dictated by the value of *iinput* passed from the previous SU. The value of *iinput* is checked, and if it does not equal 1 or 2, the program stops. A similar quality check is performed on the value of *ioutput*.

If *iinput* = 1, Namelist waves1a is read. If parameter *iwave* from waves1a equals 1, Namelist waves1b is read; otherwise, if *iwave* equals 2, Namelist waves1c is read. Wave angles are converted to radians and tidal parameters are converted to the MKS system If *iwave* equals 1, the discrete components within each wave series is then read and converted. If *iinput* equals 2, the frequency and tide arrays are initialized and converted to MKS units.

A record of SU actions and decisions is written to the Run Log through the SU.

Constraints and Limitations

None.

Input Data Elements

/REF1/

iu	integer	Switch for physical units
dconv(2)	real	Physical unit conversion factors

/REF2/

iun(8)	integer	Array with unit numbers for external files
iinput	integer	Switch for user-input, first-row complex amplitude *A* values
ioutput	integer	

/NAMES/

fname5	character	Last-row complex amplitude data filename

External Input

Namelist parameters are read from the Model Parameters File (described in Sec. 4.3.2.4).

Output Data Elements

/WAV1/

iwave	integer	Switch for wave field type
nfreqs	integer	Number of frequencies
freqs(ncomp)	real	Wave period for each frequency component
edens(ncomp)	real	Variance density for each frequency component
nwavs(ncomp)	real	Number of wave components for each frequency

/WAV2/

amp(ncomp,ncomp)	real	Amplitude of each component wave
dir(ncomp,ncomp)	real	Direction in degrees relative to x-axis for each wave component
tide(ncomp)	real	Tidal offset for each frequency component
seed	integer	Seed value for the random number generator
thet0	real	Central direction for the model spectrum

External Output

The SU sequence of actions and information about the model run are recorded in the REF/DIF1 Run Log.

5.3.3 Model Software Unit

This SU is called by the REF/DIF1 SU and controls execution of the computational part of the program. For each frequency component specified in the input, this SU performs the following series of operations:

(a) Call Premodel SU to calculate y directions subdivisions and computational grid spacing and depth (based on value of *nd*, and if *nd* < 0, *mindep*).

(b) Call Grid SU to perform the grid interpolation specified in the input data.

(c) Call Set Constants SU to calculate constants on the interpolated grid.

(d) Call Equation Integration SU to perform the numerical integration of the parabolic equation over the interpolated subgrid.

The sequence of flow was discussed previously in Sec. 4.2.3. Model execution is then complete and control is returned to the REF/DIF1 PSU.

Design Decisions and Algorithms

This SU first sets the physical constants necessary for analysis. The SU then enters a loop sequenced to execute once for each frequency to be modeled.

Within the loop, the nonlinear parameters are specified and the first-row mean $|kb|$ is calculated with a call to the Compute Wavenumber SU. This value of $|kb|$ is used to specify the initial conditions.

The next sequence within the loop is determined by the value of *iinput* (and if *iinput* = 1, *iwave*). The first row of complex amplitude values is ultimately produced; however, these values are arrived at through three different methods. The first (*iinput* = 1, *iwave* = 1) relies on the summation of the discrete components contributing to the initial condition. The second (*iinput* = 1, *iwave* = 2) employs a directional spreading model and normalizes the randomly distributed distribution of waves to produce the complex amplitude. The last method (*iinput* = 2) directs the SU to read the values of the complex amplitude from logical unit 11 (*fname4*, the First-Row Complex Amplitude Data File).

The SU writes the first row of wave heights to logical unit 8 (*fname6*, Wave Surface Complex Amplitude Data File). The SU then executes the sequence highlighted on Fig. 4.2-2. This SU calls the Grid SU to establish the grid block for the current segment. When done, the SU writes the x positions to the logical unit 10 (*fname10*, the REF/DIF1 Run Log), the x positions, and the subgrid complex amplitude values to logical unit 3 (*fname2*, the Old Standard Output Data File), and writes the tide-corrected depths to logical unit 16 (*fname15*, the Tide-Corrected Depth Output Data File). The model then calls the Set Constants SU to calculate the constants for each grid block before calling the Equation Integration SU that computes and then solves the model equation.

Once the grid block is done, the model writes the complex amplitude from the last row to logical unit 33 (*fname5*, the Last-row Complex Amplitude Data File). The SU also writes a termination value to logical unit 8 (*fname6*, the Water Surface Complex Amplitude Data File). The SU then returns to the beginning of the computation loop to evaluate the area for the next frequency component.

Constraints and Limitations

None.

Input Data Elements

/REF1/

mr	integer	Reference grid x dimension
nr	integer	Reference grid y dimension
iu	integer	Switch for physical units
dconv(2)	real	Physical unit conversion factors

/REF2/

dr(ixr,iyr)	real	Reference grid bathymetry data
ur(ixr,iyr)	real	Reference grid x direction current data
iun(8)	integer	Unit numbers for external files
iinput	integer	Switch for user-input, first-row complex amplitude *A* values

/BLOCK1/

n	integer	Number of y dimension subdivisions

/NLIN/

ntype	integer	Switch for nonlinearity

/WAV1/

iwave	integer	Switch for wave field type
nfreqs	integer	Number of frequencies
freqs(ncomp)	real	Wave period for each frequency component
edens(ncomp)	real	Variance density for each frequency component
nwavs(ncomp)	real	Number of wave components for each frequency

/WAV2/

dir(ncomp,ncomp)1	real	Direction in degrees relative to x-axis for each wave component
tide(ncomp)	real	Tidal offset for each frequency component
seed	integer	Seed value for the random number generator

/NAMES/

fname6	character	Complex amplitude data filename

External Input

If *iinput* = 2 the SU will read the first-row complex amplitude data from the First-Row Complex Amplitude File (described in Sec. 4.3.2.5.3).

Output Data Elements

/REF3/

x(ix)	real	Spacing of subgrids in x direction

/BLOCK1/

d(ix,iy)	real	Sub-grid bathymetry data

/CON2/

k(ix,iy)	real	Array with wavelength at each sub-grid point
kb(ix)	real	Array with mean wavelength for each row

/NLIN/

an	integer	Switch for linear/nonlinear model
an1	integer	Switch for composite or Stokes nonlinear models

/COMP/

a(ix,iy)	complex	Array with complex wavelength amplitude
psibar	real	Reference phase function

External Output

The sequence of SU actions is recorded in the REF/DIF1 Run Log (described in Sec. 4.3.3.1.6). Output data is also written to the Old Standard Output Data File (Sec. 4.3.3.1.1), the Last-Row Complex Amplitude Data File (Sec. 4.3.3.1.2) and the Water Surface Complex Amplitude Data File (Sec. 4.3.3.1.3).

5.3.3.1 Premodel Software Unit

This SU (subroutine premodel) is called by the Model Software Units. In shallow water, the computational grid needs to have at least six points per wavelength in the x direction. The model (in Model SU) computes the x direction subdivisions automatically if *ispace* = 0. Subdivisions in the y direction (*nd*) are an input parameter and are fixed in this SU for the entire model domain. Two options for determining *nd* are available.

(1) If the user entered a non-negative value for *nd* (i.e., 0 or higher), the model treats that value as the number of subdivisions, and relevant computational arrays in the y direction are computed.

(2) This is a new feature added to the code. If *nd* is negative, the user was prompted for a depth value in the Create Model Parameters SU. If *nd* = –1, the depth (*mindep*) was set to 1 m. If *nd* = –2, the user input a real depth value. Using the Compute Wavenumber SU, this SU determines the wavelength of the input wave at *mindep* and computes the appropriate *nd* value to achieve computational spacing equal to that in the x direction. This is done because most of the wave energy is focused in the x direction. Relevant computational grids in the y direction are then computed using the new value of *nd*.

Design Decisions and Algorithms

This SU performs a simple iteration algebraic series to compute resolution in the y direction.

Constraints and Limitations

None.

Input Data Elements

/REF1/

nd	integer	Indicator for or number of y direction subdivisions required

If *nd* = –1 or *nd* = –2

/WAV1/

freqs(ifreq)	real	Wave period of current frequency component

/REF1/

mindep	real	Depth used to compute new value of *nd*

External Input

None.

Output Data Elements

/REF3/

y	integer	Array with y direction computational grid spacing

/BLOCK1/

n	integer	Total number of y dimension gridpoints
dy	real	Unit y direction computational grid spacing

External Output

None.

5.3.3.2 Compute Wavenumber Software Unit

This SU (subroutine wvnum) is called by the Model's Grid and Set Constants Software Units. It performs a Newton-Raphson solution of the linear wave-current dispersion relation to obtain values of the wavenumber k.

Design Decisions and Algorithms

This SU performs a simple iteration to determine the value of the wavenumber k. The value of the wavenumber is initially determined from the equation

$$k = s^2/g \Big/ \sqrt{\tanh\left(s^2 d/g\right)} \;. \tag{11}$$

The following steps are then iterated 20 times. The values of f and fp are then determined from the equations

$$f = s^2 - 2sku - gk\ \tanh\,(kd),\ \text{and} \tag{12}$$

$$fp = -2su + 2ku^2 - g\ \tanh\,(kd) - gkd/\cosh\,(kd^2)\,. \tag{13}$$

From $kn = k - f/fp$ an estimate of k is computed. If the value of $kn - k/_k$ is less than eps (set to 1×10^{-5} in the previous SU), k is set equal to kn and the program returns to the previous SU. If the iteration does not succeed, flags are raised and the iteration failure is recorded in the Run Log.

Constraints and Limitations

None.

Input Data Elements

eps	real	Value equal to 1×10^{-5}
i*	real	X position of current sub-grid element
j*	real	Y position of current sub-grid element
dref	real	Reference depth array
	(when called from Grid SU)	

OR

/BLOCK1/

d(ix,iy)	real	Subgrid bathymetry element
	(when called from Model or Constants SUs)	

/REF2/

ur(ixr,iyr)	real	Reference grid x direction current data

/WAV1/

freqs(ncomp)	real	Wave period for each frequency component

External Input

None.

Output Data Elements

icdw	integer	Error flag

/CON2/

k(ix,iy)	real	Array with wavelength at each point

External Output

Errors within the SU are recorded in the REF/DIF1 Run Log.

5.3.3.3 Normalize Spectrum Software Unit

Called by the Model SU, this SU (subroutine acalc) normalizes the directional spectrum energy density over a 90° sector for each frequency component.

Design Decisions and Algorithms

This SU computes the directional spectrum a with an initial estimate and 10 convergence iterations. The initial estimate is based on the formula

$$a = {}^{thmax \times bn}\!\big/_{2}(2 \times nsp) - 1 \;, \tag{14}$$

where

$$nsp = nwavs(ncomp) \;, \tag{14a}$$

$thmax = {}^{\Pi}\!/_{4}$, and bn is the binary number returned from the next SU.

Constraints and Limitations

None.

Input Data Elements

nsp	real	Equal to /WAV1/nwavs(ncomp) at each individual frequency
thmax	real	Real variable equal to ${}^{\Pi}\!/_{4}$

External Input

None.

Output Data Elements

a1	real	Directional spectrum

External Output

None.

5.3.3.3.1 Compute Bernoulli Software Unit. This SU (subroutine bnum) is called by the Normalize Spectrum SU to compute the Bernoulli number

$$^{n!}/_{k!(n-k)!} \cdot \tag{15}$$

Design Decisions and Algorithms

This SU computes the above value by calling the factorial SU for each of the three values, n, k, and $(n-k)$, before producing and then returning the Bernoulli number.

Constraints and Limitations

None.

Input Data Elements

in	integer	Equal to 2 * *nsp*
n	integer	Equal to *nsp*

External Input

None.

Output Data Elements

bn	real	Bernoulli number

External Output

None.

5.3.3.3.1.1 Factorial software unit. This SU (function fact) is called by the Compute Bernoulli SU to compute $n!$ of an interger n.

Design Decisions and Algorithms

This program executes a DO loop from 1 to n and produces the factorial by multiplying the steps together.

Constraints and Limitations

None.

Input Data Elements

xi	integer	Integer n, as above

External Input

None.

Output Data Elements

fact	integer	Factorial of n

External Output

None.

5.3.3.4 Generate Random Software Unit

This SU (function rand) is called by the Model SU and is a simple random number generator used to initialize random wave phases if the directional spreading model is being used.

Design Decisions and Algorithms

This SU uses the seed value to produce a random real number.

Constraints and Limitations

None.

Input Data Elements

x	integer	Number equal to /wav2/seed

External Input

None.

Output Data Elements

rand	real	Random number

External Input

None.

5.3.3.5 Grid Software Unit

This SU is called by Model SU. It performs the required interpolation over a single grid block of the reference grid as specified in the input data. This SU determines whether or not a user-specified sub-grid feature is available and if necessary reads in the data. The interpolated depth grid is then corrected for tidal offset and checked for surface-piercing features. These features are modified using the thin film approach; see Kirby and Dalrymple (1986a). Control is returned to Model SU.

Design Decisions and Algorithms

This SU first interpolates the reference grid depth and current data to the y divisions in the subgrid. The SU then determines the number of subdivisions in the x direction for the current reference grid. If *ispace* = 0, the program computes the number of x direction subdivisions, *md*(*ir*), for the subgrid. The Model Initialization SU supplies the values of *md* if the *md* array is user-supplied (*ispace* = 1).

With the x direction subdivisions computed, the SU finishes interpolating the depth and current data to the entire subgrid. The SU supersedes the interpolated sub-grid depth values with user-supplied values from logical unit 2 (*fname3*, the Reference Grid Data File). Sub-grid current values are also read from logical unit 2 if *icur* = 1. Depth and current values are converted to the MKS system if necessary.

Finally, the tidal offset is added to the depth subgrid. If necessary, the thin film approach is applied.

Constraints and Limitations

None.

Input Data Elements

/REF1/

nr	integer	Reference grid y dimension
ispace	integer	Switch controlling x subdivisions
nd	integer	Number of y direction subdivisions
md(ixr)	integer	Number of x direction subdivisions per reference grid
iu	integer	Switch for physical units
dconv(2)	real	Physical unit conversion factors
icur	integer	Switch for input current data

/REF2/

dr(ixr,iyr)	real	Reference grid bathymetry data
ur(ixr,iyr)	real	Reference grid x direction current data
vr(ixr,iyr)	real	Reference grid y direction current data
iun(8)	integer	Unit numbers for external files

/REF3/

dxr	real	Reference grid x spacing
dyr	real	Reference grid y spacing
xr(ixr)	real	Spacing of reference grid in x direction
yr(iyr)	real	Spacing of reference grid in y direction
y(iy)	real	Spacing of subgrid in y direction

/REF4/

isd(ixr,iyr)	integer	Switch for user-supplied sub-grid data

/BLOCK1/

n	integer	Number of y dimension subdivisions
dy	real	Current subgrid y dimension spacing

/CON2/

k(ix,iy)	real	Wavelength at each gridpoint

/NLIN/

an	integer	Switch for linear/nonlinear model
an1	integer	Switch for composite or Stokes nonlinear models
ntype	integer	Switch for nonlinearity

/WAV1/

freqs(ncomp)	real	Wave period for each frequency component

/WAV2/

tide(ncomp)	real	Tidal offset for each frequency component

External Input

The Grid SU reads sub-grid depth and current (if $icur = 1$) data from the Sub-grid Data File (described in Sec. 4.3.2.5.2).

Output Data Elements

/REF3/

x(ix)	real	Spacing of subgrid in x direction

/BLOCK1/

d(ix,iy)	real	Subgrid bathymetry data
u(ix,iy)	real	Subgrid x direction current data
v(ix,iy)	real	Subgrid y direction current data
m	integer	number of x dimension subdivisions
dx	real	Current subgrid x dimension spacing

/CON2/

kb(ix)	real	Mean wavelength for each row

External Output

Errors of grid-sizing within the SU are recorded in the REF/DIF1 Run Log.

5.3.3.6 Set Constants Software Unit

This SU is called by the Model SU and calculates constants for the current subgrid.

Design Decisions and Algorithms

This SU computes constants for each element of the subgrid. The wave number for each element is required, and as a result, the Compute Wave Number SU is called for each element. The SU then calls the Dissipation Coefficients SU, and finally computes the mean wave number for each column of the subgrid.

Constraints and Limitations

None.

Input Data Elements

/BLOCK1/

d(ix,iy)	real	Sub-grid bathymetry data
u(ix,iy)	real	Sub-grid x direction current data
m	integer	Number of x dimension subdivisions
n	integer	Number of y dimension subdivisions

/CON2/

k(ix,iy)	real	Wavelengths for each gridpoint

/WAV1/

freqs(ncomp)	real	Wave period for each frequency component

External Input

None.

Output Data Elements

/CON1/

q(ix,iy)	real	Array with coefficient for governing equation
p(ix,iy)	real	Array with values of *p* as for governing equation
sig(ix,iy)	real	Array with intrinsic frequency for governing equation
bottomu(ix,iy)	real	Magnitude of bottom velocity

/CON2/

kb(ix)	real	Mean wavelength for each row
w(ix,iy)	complex	Dissipation terms

External Output

None.

5.3.3.6.1 Dissipation Coefficients Software Unit. This SU (subroutine diss) is called by the Set Constants SU and calculates the frictional dissipation coefficients for the subgrid based on values of the *iff* switches.

Design Decisions and Algorithms

For each element of the subgrid this SU at first sets the damping coefficients to 0 before adding in frictional dissipation. If *iff*(1) = 1 turbulent boundary layer damping is added, if *iff*(2) = 1 porous bottom damping is included, and if *iff*(3) = 1 boundary layer damping is added to the dissipation coefficient. The dissipation coefficient will remain equal to 0 if all three switches within *iff* equal 0.

Constraints and Limitations

None.

Input Data Elements

/REF1/

iff(3)	integer	Dissipation switches

/BLOCK1/

m	integer	Number of x direction subdivisions
n	integer	Number of y direction subdivisions

/CON1/

sig(ix,iy)	real	Array with intrinsic frequency for governing equation

/CON2/

k(ix,iy)	real	Wavelength at each gridpoint

/COMP/

a(ix,iy)	complex	Wavelength complex amplitude

External Input

None.

Output Data Elements

/CON2/

w(ix,iy)	complex	Dissipation terms

External Output

None.

5.3.3.7 Equation Integration Software Unit

This SU (subroutine fdcalc) is also called by the Model SU and performs the integration of the governing parabolic equation over the grid defined in the Grid SU. The coefficients of the finite difference form of the parabolic equation are developed according to the Crank-Nicholson method. A complete description of the equation and the treatment of nonlinearities may be found in Kirby (1986a) and Kirby and Dalrymple (1986b). The sequence of steps in this SU is as follows:

(a) An implicit step is performed to update complex amplitude *A* along an entire grid row.

(b) The model checks for the start or stop of breaking on the updated row.

(c) If the status of breaking changes, the model recomputes the breaking wave dissipation coefficient.

(d) Then, if nonlinearity is being used or breaking status at any point along the row has changed, the model computes a new estimate of *A* on the updated row based on values obtained during the previous iteration.

These of operations is performed for each row in the subgrid until the end of the grid is reached. Control is then returned to Model SU.

Design Decisions and Algorithms

This SU first computes the numerous coefficients for the model equation discussed in Sec. 3.X. The Booij coefficients described in Sec. 4.X and other constants are then defined. If $ir = 1$, the breaking indices *ibr* and *wb* are set.

The SU then sequentially acts over the current reference grid one reference block at a time. The SU solves for the right side of the Crank-Nicholson equation (described in Sec. 3.6.1) and establishes the boundary conditions. If $ibc = 1$, reflecting boundary conditions are included.

The SU also calculates dissipation in rows where breaking occurs; if $ibr(j) = 1$, a dissipation factor *wb* is calculated; otherwise, if $ibr(j) = 0$ the wave-breaking dissipation factor is set to 0. The coefficients for the forward row are then calculated. These values, as well as the solved for right side of the equation, are then sent to the Solve Equation SU, which provides a solution for the complex amplitude at the current grid row.

The solution array for the current grid row is then re-evaluated for wave breaking. If the breaking status is found to have changed the value of *ibr(j),* values are reset and the initial parameter calculations (boundary conditions, wave dissipation, and forward row coefficients) and Solve Equation SU are redone. This sequence of calculations is twice iterated so that the first solution of the complex amplitude *A* is then used to recalculate the boundary and dissipation values for the current grid row.

For the Stokes model alone (*ntype*=2), the SU tests the Ursell parameter to see if it is too large; this is recorded in the REF/DIF1 Output Log if true. The breaking dissipation coefficients are rolled back for each row, and the reference phase function for surface plotting (psibar) is calculated. If necessary, the plotted surface values are stored. The SU also decides sets the filter switch *ifilt* to 1 if breaking was found.

This SU finishes by calculating and storing values to the output file. The wave angles at the reference grid rows are calculated. Wave heights, wave angles, water depths, bottom velocities and reference grid data are all written to external output files if requested. The solution is rolled back to the first grid level and the sequence is repeated for the next grid row.

Constraints and Limitations

None.

Input Data Elements

/REF1/

iu	integer	Switch for physical units
dconv(2)	real	Physical unit conversion factors
ibc	integer	Boundary conditions switch

/REF2/

iun(8)	integer	Unit numbers of external files

/BLOCK1/

u(ix,iy)	real	Sub-grid x direction current data
v(ix,iy)	real	Sub-grid y direction current data
m	integer	Number of x dimension subdivisions
n	integer	Number of y dimension subdivisions
dx	real	Current sub-grid x dimension spacing
dy	real	Current sub-grid y dimension spacing

/CON1/

q(ix,iy)	real	Coefficients for governing equation
p(ix,iy)	real	Array with coefficients for governing equation
siz(ix,iy)	real	Intrinsic frequencies
bottomu(ix,iy)	real	Magnitudes of bottom velocity

/CON2/

k(ix,iy)	real	Wavelengths at each sub-grid point
kb(ix)	real	Average wavelength for each row
w(ix,iy)	complex	Dissipation terms
dd(ix,iy)	real	Nonlinear term *D* in governing equation

/COMP/

a(ix,iy)	complex	Complex wavelength amplitudes

/NAMES/

fname6	character	Complex amplitude data filename
fname7	character	Magnitude of bottom velocity data filename

External Input

None.

Output Data Elements

/BLOCK1/

d(ix,iy)	real	Sub-grid bathymetry data
ibr(iy)	integer	Switch with breaking status of each row

/CON2/

wb(2,iy)	complex	Wave-breaking dissipation terms

/COMP/

a(ix,iy)	complex	Complex wavelength amplitudes
psibar	real	Reference phase function
ifilt	integer	Switch for automatic smooth filtering

External Output Files

This SU writes output to most of the output data files detailed Sec. 4.3.3. For each reference grid row, the values of values of A are written to logical unit 8 (*fname6*, the Water Surface Complex Amplitude Data File); wave heights $2|A|$ are written to logical unit 12 (*fname11*, the Wave Height Output Data File); the wave angles *thet(j)* are written to logical unit 9 (*fname8*, the Wave Direction Data File). Also stored for each reference gridpoint are tide-corrected water depths on logical unit 16 (*fname15*, the Tide-Corrected Depth Output Data File) and the bottom-current magnitude *bottomu(j)* on logical unit 17 (*fname7*, the Output Bottom Velocity Data File). The reference gridpoints in the x direction are recorded in logical unit 10 (*fname10*, the REF/DIF1 Run Log). Finally, the x grid positions and the complex wave amplitude values are stored to logical unit 3 (*fname3*, the Old Standard Output Data File).

5.3.3.7.1 Solve Equation Software Unit. This SU (subroutine ctrida) is a utility routine called by the Equation Integration SU to perform the double sweep elimination to solve the implicit set of equations for the model. The equations are derived from a given row within the Crank-Nicholson scheme as shown in Eq. (10).

Within this model, the terms on the right side are known (from boundary conditions or from previous calculation) and can be solved. The coefficients on the left side are variable, complex, and nonlinear. This CSC employs the two-pass iterative method to ensure that the nonlinearities are treated accurately (Kirby and Dalrymple 1983a). This determines the complex amplitude A for column $i+1$ for row j.

Design Decisions and Algorithms

The coefficients from the above equation are received by this SU and are used to compute the solution array v for the current column. This short SU first computes the two intermediate vectors β and γ as follows:

$$\beta(1) = b(1), \text{ and} \tag{16a}$$

$$\gamma(1) = d(1) / \beta(1) . \tag{16b}$$

For the column in question the solution vector v is solved for by setting

$$v(n) = gamma(n)\,, \tag{16c}$$

where n is the last row of the column. For each element of the column, from element $n-1$ back to element 1, the solution vector is then solved for by setting

$$v(i) = gamma(i) - c(i)^{v(i+1)}/_{beta(i)}\,, \tag{17}$$

where i is the column element in question. After completing the column, solution control is returned to the previous SU.

Constraints and Limitations

None.

Input Data Elements

a(iy)	real	Single-column array of size n with coefficients of $aA_{i+1,j+1}$
b(iy)	real	Single-column array of size n with coefficients of $bA_{i+1,j}$
c(iy)	real	Single-column array of size n with coefficients of $cA_{i+1,j-1}$
d(iy)	real	Right side solution of above equation

/BLOCK1/

n	integer	Subroutine variable 1

External Input

None.

Output Data Elements

v(iy)	real	Single column array with complex amplitude solutions for column $i+1$.

External Output

None.

5.4 Output Manipulation Primary Software Unit

This PSU currently consists of just one SU: the Make Surface SU, which converts the REF/DIF1 Output Files to ASCII format files suitable for Matlab input.

5.4.1 Make Surface Software Unit

This CSC converts an output file (usually surface.dat) containing an instantaneous snapshot of the water surface at the computational grid spacing to a regularly spaced ASCII file, suitable for directly reading into Matlab format.

Design Decisions and Algorithms

This SU first reads in the number of columns (ny) and the actual positions of the y columns. The SU then reads in each x position before reading in the appropriate grid column. The SU interpolates the grid of surface height values to an evenly spaced grid, which is then written to *fileout*.

Constraints and Limitations

None.

Input Data Elements

The name of the output data file (*fileout*) is user-input via Standard Input.

External Input

This SU reads in data from both the Model Parameters File (*indat.dat*, described in Sec. 4.3.2.4) and the file with surface water height (*fname6*, described in Sec. 4.3.3.1.3).

Output Data Elements

None.

External Output

This SU writes an instantaneous snapshot of the water surface to a regularly spaced ASCII file, *fileout*.

6.0 CSC REQUIREMENTS TRACEABILITY

The CSCI described in this SDD was developed from the requirements included in the SRS document. The requirements outlined in the SRS are included in Table 6.0-1, which includes the SRS and SDD sections relating to each requirement, allowing quick reference between the two.

7.0 NOTES

7.1 Acronyms and Abbreviations

CDRL	Contract Data Requirements List
cm	centimeter
CSCI	Computer Software Configuration Item
MIL-STD	Military Standard
OAML	Oceanographic and Atmospheric Master Library
MKS	Metric Kelvin System
PSU	Primary Software Unit
REF/DIF1	Combined Refraction/Diffraction Model Version 2.5
SDD	Software Design Document
SRS	Software Requirements Specification
STD	Software Test Document
SU	Software Unit

Table 6.0-1 — Requirements Traceability Table Outlines Correspondence Between CSCI Capabilities and SRS Requirements

REQUIREMENT	SRS SECTION DESCRIBING REQUIREMENT	COMPARABLE SDD SECTION
Linear Theory Model Solution	Section 3.1.1.1	Section 3.3.1
Shoaling-Water Applicable	Section 3.1.1.2	Section 3.3.2
Consideration of Refraction/Diffraction	Sections 3.1.2.2, 3.1.2.3, and 3.1.2.4	Section 3.2.1, 3.2.2, and 3.2.3
Energy Dissipation Terms	Sections 3.1.3 and 3.2.1	Section 3.4
Laminar Surface/Bottom Flow	Section 3.1.3.1	Section 3.4.1
Turbulent Bottom Flow	Section 3.1.3.2	Section 3.4.2
Darcy Flow	Section 3.1.3.3	Section 3.4.3
Depth-Induced, Wave-Breaking Included	Section 3.1.3.4	Section 3.4.4
Accommodation of Irregular Bathymetry	Sections 3.1.2.1 and 3.1.2.2	Section 3.2.1
Consideration of Local Current Data	Sections 3.1.2.5, 3.2.1, and 3.2.2	Section 3.2.4
Inclusion of Local Sub-grid Data	Sections 3.1.5.2 and 3.2.3	Section 3.6.3
Response to Several Wave Climates	Section 3.1.4	Section 3.5
Monochromatic Waves	Section 3.1.4.1	Section 3.5.1
Discrete Direction Waves	Section 3.1.4.2	Section 3.5.2
Directional Spectrum Waves	Section 3.1.4.3	Section 3.5.3
Lateral Boundary Conditions	Section 3.1.5.1	Section 3.6.2
Required Output	Section 3.2.	Section 4.3.3.1
Wave Height	Section 3.2.5	Section 4.3.3.1.7
Direction of Wave Propagation	Section 3.2.7	Section 4.3.3.1.5
Tide-Corrected Depth	Section 3.2.6	Section 4.3.3.1.8
Ease of Data Input	Section 3.2.1	Section 4.3.2.4

APPENDIX A

COMMON BLOCK VARIABLES

TABLE A-1 — Common Block /ref1/ Variables

mr	Integer	Reference grid x dimension.
nr	Integer	Reference grid y dimension.
ispace	Integer	Switch controlling x-subdivisions.
nd	Integer	Number of y direction subdivisions.
md	Integer (ixr)	Number of x dimension subdivisions per reference grid.
iu	Integer	Switch for physical units.
dconv	Real (2)	Array with physical unit conversion factors.
iff	Integer (3)	Dissipation switches.
icur	Integer	Switch for input current data.
ibc	Integer	Boundary conditions switch.
mindep	Real	Depth to be used to compute y direction subdivisions.

TABLE A-2 — Common Block/ref2/ Variables

dr	Real (ixr,iyr)	Reference grid bathymetry data.
Ur	Real (ixr,iyr)	Reference grid x direction current data.
Vr	Real (ixr,iyr)	Reference grid y direction current data.
Iun	Integer (8)	Unit numbers for external files.
Iinput	Integer	Switch for user-input first-row complex amplitude *A* values.
Ioutput	Integer	Switch for saving last-row complex amplitude *A* values.

TABLE A-3 — Common Block /ref3/ Variables

dxr	Real	Reference grid x spacing.
Dyr	Real	Reference grid y spacing.
Xr	Real (ixr)	Spacing of reference grid in x direction.
Yr	Real (iyr)	Spacing of reference grid in y direction.
X	Real (ix)	Spacing of subgrid in x direction.
Y	Real (ix)	Spacing of subgrid in y direction.

TABLE A-4 — Common Block /ref4/ Variables

isd	Integer (ixr, iyr)	Switch for user-supplied subgrid data.

TABLE A-5 — Common Block /block1/ Variables

d	Real (ix,iy)	Subgrid bathymetry data.
U	Real (ix,iy)	Subgrid x direction current data.
V	Real (ix,iy)	Subgrid y direction current data.
M	Integer	Number of x dimension subdivisions.
N	Integer	Number of y dimension subdivisions.
Dx	Real	Current subgrid x dimension spacing.
Dy	Real	Current subgrid y dimension spacing.
Ibr	Integer (iy)	Switch with breaking status of each row.

TABLE A-6 — Common Block /con1/ Variables

q	Real (ix,iy)	Array with coefficient for governing equation.
P	Real (ix,iy)	Array with coefficient for governing equation.
Sig	Real (ix,iy)	Array with the intrinsic frequency.
Bottomu	Real (ix,iy)	Array with magnitude of bottom velocity.

TABLE A-7 — Common Block /con2/ Variables

k	Real (ix,iy)	Array with wavelength at each grid point.
Kb	Real (ix)	Array with mean wavelength for each row.
W	Complex (ix,iy)	Dissipation terms.
Dd	Real (ix,iy)	Nonlinear term D in governing equation.
Wb	Complex (2,iy)	Array with wave-breaking dissipation factor.

TABLE A-8 — Common Block /nlin/ Variables

an	Integer	Switch for linear/nonlinear model.
an1	Integer	Switch for composite or Stokes nonlinear models.
Ntype	Integer	Switch for nonlinearity.

TABLE A-9 — Common Block /wav1/ Variables

iwave	Integer	Switch for wave field type.
Nfreqs	Integer	Number of frequencies to be modeled.
Freqs	Real (ncomp)	Wave period for each frequency component.
Edens	Real (ncomp)	Variance density for each frequency component.
Nwavs	Integer (ncomp)	Number of wave components for each frequency.

TABLE A-10 — Common Block /wav2/ Variables

amp	Real (ncomp, ncomp)	Amplitude for each component wave.
Dir	Real (ncomp, ncomp)	Direction in degrees relative to x-axis for each wave component.
Tide	Real (ncomp)	Tidal offset for each frequency component.
Seed	Integer	Seed value for the random number generator.
Thet0	Real	Central direction for the model spectrum.

TABLE A-11 — Common Block /comp/ Variables

a	Complex (ix,iy)	Array with complex wave amplitude.
Psibar	Real	Reference phase function.
Ifilt	Integer	Switch for automatic smooth filtering.

TABLE A-12 — Common Block /names/ Variables

fname1	Character*255	Reference grid data filename.
fname2	Character*255	Old standard output data filename.
fname3	Character*255	User-specified subgrid data filename.
fname4	Character*255	User-specified row 1 complex amplitude data filename.
fname5	Character*255	Last-row complex amplitude data filename.
fname6	Character*255	Complex amplitude data filename.
fname7	Character*255	Magnitude of bottom velocity data filename.
fname8	Character*255	Wave direction data filename.
fname9	Character*255	Filename not in use.
fname10	Character*255	REF/DIF1 Run Log filename.
fname11	Character*255	Reference grid wave height filename.
fname12	Character*255	Filename not in use.
fname13	Character*255	Filename not in use.
fname14	Character*255	Filename not in use.
fname15	Character*255	Reference grid tide-corrected depth data filename.
Fnamein	Character*255	Model Parameters Data Filename.

APPENDIX B

MODEL ERRORS RECOGNIZED BY REF/DIF1

REF/DIF1 performs some data checking and checking of calculations during a run. This checking may result in warnings or terminal errors that are beyond calculation errors and would lead to standard Fortran error messages. A list of nine possible errors and the resulting messages follow.

1. Reference grid dimensions were specified as being too large on input: $mr > ixr$ and/or $nr > iyr$.

   Message: Dimensions for reference grid too large; stopping.
   Action: Program stops.
   Error occurs in subroutine *inref.*

2. User specifies a y direction subdivision *nd* that will cause the number of y gridpoints *n* to exceed the maximum *iy*.

   Message: y direction subdivision too fine. Maximum number of y gridpoints will be exceeded. Execution terminating.
   Action: Program stops.
   Error occurs in subroutine *inref.*

3. User specifies an x direction subdivision on one of the grid blocks *ir* that exceeds the maximum amount $(ix - 1)$. As a result, the dimension of the subdivided grid will be too large.

   Message: x direction subdivision too fine on grid block *ir*, execution terminating.
   Action: Program stops.
   Error occurs in subroutine *inref.*

4. If a depth value occurs in the reference grid that differs from the average of its neighbors by more than the depth tolerance value *dt*, the program prints the value but takes no other action. Printed values are in meters.

   Message: Depth *dr* (m) at reference grid location *ir, jr* differs from the average of its neighbors by more than *dt* (m). Execution continuing.
   Action: None by program. Data in file *fname3* should be corrected if wrong.
   Error occurs in subroutine *inref.*

5. An ambient current value occurs that implies that the flow would be supercritical at the given location. This serves as both a check for anomalously large current values and as an indicator of possible subsequent computational problems.

   Message: Ambient current at reference grid location *ir, jr* is supercritical with Froude number = "Froude number," execution continuing.
   Action: None by program. Data in file *fname3* should be corrected if wrong.
   Error occurs in subroutine *inref.*

6. If the user specifies that predetermined subgrids are to be read, while at the same time telling the program to perform its own subdivisions, the computed dimensions of the subgrid may be different than those of the subgrid included in the input. Runs requiring user-specified subgrids should choose the *ispace* = 1 option. If an incompatible set of dimensions occurs, the program will either garble the input array or run out of data.

   Message: Warning: input specifies that user will be supplying specified subgrids (*isp* = 1), while program has been told to generate its own sub-grid spacings (*ispace* = 0). Possible incompatibility in any or all sub-grid blocks.
   Action: None by program. Should restart unit with correct *ispace*, *isp* values.
   Error occurs in subroutine *inref.*

7. While calculating its own subdivision spacings, the model may try to put more division in a reference grid block than is allowed by dimension *ix*. If this occurs, the program uses the maximum number of subdivisions allowed ($ix - 1$), but prints a message indicating that the reference grid spacing is too large with respect to the waves being calculated. This problem may be circumvented by increasing the size of *ix* in param.h.

   Message: Model tried to put more spaces than allowed in grid block *ir*.
   Action: Program uses ($ix - 1$) and continues. Model resolution and accuracy may be poor, and a finer reference grid or increased value of *ix* in the param.h file should be used.
   Error occurs in subroutine *grid.*

8. While using the Stokes wave form of the model, *ntype* = 2, the model may encounter large values of the Ursell number, indicating that the water is too shallow for that model to be appropriate. The cutoff point recognized by the program is $(A/h)/(kh)^2 = 0.5$.

   Message: Warning: Ursell number = *u* encountered at grid location *i,j* should be using Stokes-Hedges model (*ntype* = 1) due to shallow water.
   Action: The program should be rerun with the composite nonlinear model.
   Error occurs in subroutine *fdcalc.*

9. The Newton Raphson iteration for wavenumber *k* may not converge in the specified number of steps. This may occur for waves on strong opposing currents.

   Message: WAVENUMBER FAILED TO CONVERGE ON ROW *i*, COLUMN *j*
   K = last iterated value of wavenumber
   D = depth
   T = period calculated from last iterated value of *k*
   U = x direction velocity
   F = value of objective function (should be = 0 for convergence)
   Action: Program continues with last iterated value of *k*. Computed results are of questionable accuracy.
   Error occurs in subroutine *wvnum.*

Software Test Document

1.0 SCOPE

1.1 Identification

This Software Test Description (STD) outlines the relevant qualifications testing of the Computer Software Configuration Item (CSCI) known as the Shallow-Water Wave Refraction/Diffraction Model (REF/DIF1). This model was based on the weakly nonlinear combined refraction and diffraction model initially developed by Kirby and Dalrymple (1983, 1994). The CSCI was designed to predict the propagation of water waves over irregular bottom bathymetry simulating the processes of shoaling, refraction, diffraction, and energy dissipation.

1.2 Overview

This STD describes the test cases and test procedures used to perform software qualifications testing of the REF/DIF1 CSCI. This STD provides a means to evaluate the quality of the software available to the user. This report outlines the relevant test preparations, including hardware, software, other pre-test preparations, and test descriptions. Individual test descriptions will include the prerequisite conditions, test inputs, and expected test results for each scenario. Many of the model tests and test results from the REF/DIF1 Validation Test Report (VTR) (Hsu et al. 1997) are also included in this report as a means for software testing. This document has been prepared for transition into the Oceanographic and Atmospheric Master Library (OAML) in accordance with the Software Documentation Standards for Environmental System Product Development defined by the Naval Oceanographic Office, which is based on the Military Standard for Software Development and Documentation, MIL-STD-498.

2.0 REFERENCED DOCUMENTS

Berkhoff, J. C., N. Booij, and A. C. Radder, "Verification of Numerical Wave Propagation Models for Simple Harmonic Linear Water Waves," *Coastal Eng.* **6**(3), 255–279 (1982).

Dean, R. G. and R. A. Dalrymple, "Water Wave Mechanics for Engineers and Scientists," World Scientific Publishing Company, Salem, OR, 1991.

Hsu, Y. L., J. M. Kaihatu, and A. MacNaughton, "Validation Test Report for the Shallow-Water Refraction and Diffraction Wave Model (REF/DIF1)," NRL/FR/7322--97-9664, Naval Research Laboratory, Stennis Space Center, MS, 1997.

Kirby, J. T. and R. A. Dalrymple, "Combined Refraction/Diffraction Model REF/DIF1 Version 2.5 Documentation and User's Manual," University of Delaware Department of Civil Engineering Center for Applied Coastal Research, Newark, DE, 1994, 171 pp.

Kirby, J. T. and R. A. Dalrymple, "A Parabolic Equation for the Combined Refraction-Diffraction of Stokes Waves by Mildly Varying Topography," *J. Fluid Mechanics* **136**, 543–566 (1983).

Kirby, J. T., "A Note on Linear Surface Wave-Current Interaction," *J. Geophysical Res.* **89**, 745–747 (1984).

O'Reilly, W. C. and R. T. Guza, "A Comparison of Two Spectral Wave Models in the Southern California Bight," *Coastal Engineering* **19**, 263–282 (1993).

Vincent, C. L. and M. J. Briggs, "Refraction-Diffraction of Irregular Waves Over a Mound," ASCE *J. of Waterway, Port, Coastal, and Ocean Engineering* **115**(2), 269–284 (1989).

3.0 TEST PREPARATIONS

All of the software, scenario input, and output files necessary for testing the REF/DIF1 CSCI are included in the OAML set. The detailed information on input waves and bathymetry is listed in App. A and in the VTR. The files listed below are necessary for testing of the software but are not needed for any other purpose. There is a total of 17 STD-related data files included with this package:

- sandr.inp
- sandr.bth
- sandr.hgt
- vbs.inp
- vbs.hgt
- vbs.bth
- wacur.inp
- oppcur.bth
- wacur.hgt

- bbr.inp
- bbr.bth
- bbr.hgt
- duck.inp
- duck.bth
- duck.hgt

All of the bathymetry files (*.bth) necessary for the test scenarios outlined in this document are provided in the OAML software package and need to be moved or renamed.

REF/DIF1 software must be compiled prior to testing as described by this report.

Very little memory (200 kbytes) is required to perform each test. No hardware preparation is needed for any test.

3.1 Shoaling and Refraction Test

Waves propagating from deep water to shallow water are subject to shoaling and refraction processes (Dean and Dalrymple 1991). This test was designed to evaluate REF/DIF1 for shoaling and refraction processes at a straight, sloping beach.

3.1.1 Pre-Test Preparations

For this test, one of the data files included in the software package must be renamed, as below:

Model Parameters Input File:

sandr.inp $\Rightarrow$ indat.dat

The bottom bathymetry is contained in the Reference Grid Data File (sandr.bth). The bottom bathymetry is a smooth, constant slope inshore beach face.

3.2 Berkhoff-Booij-Radder (BBR) Shoal Experiment

Berkhoff et al. (1982) conducted a set of laboratory wave experiments studying wave focusing by a submerged elliptic shoal resting on a plane beach (slope 1:50), as shown in Fig. 3.2-1. This

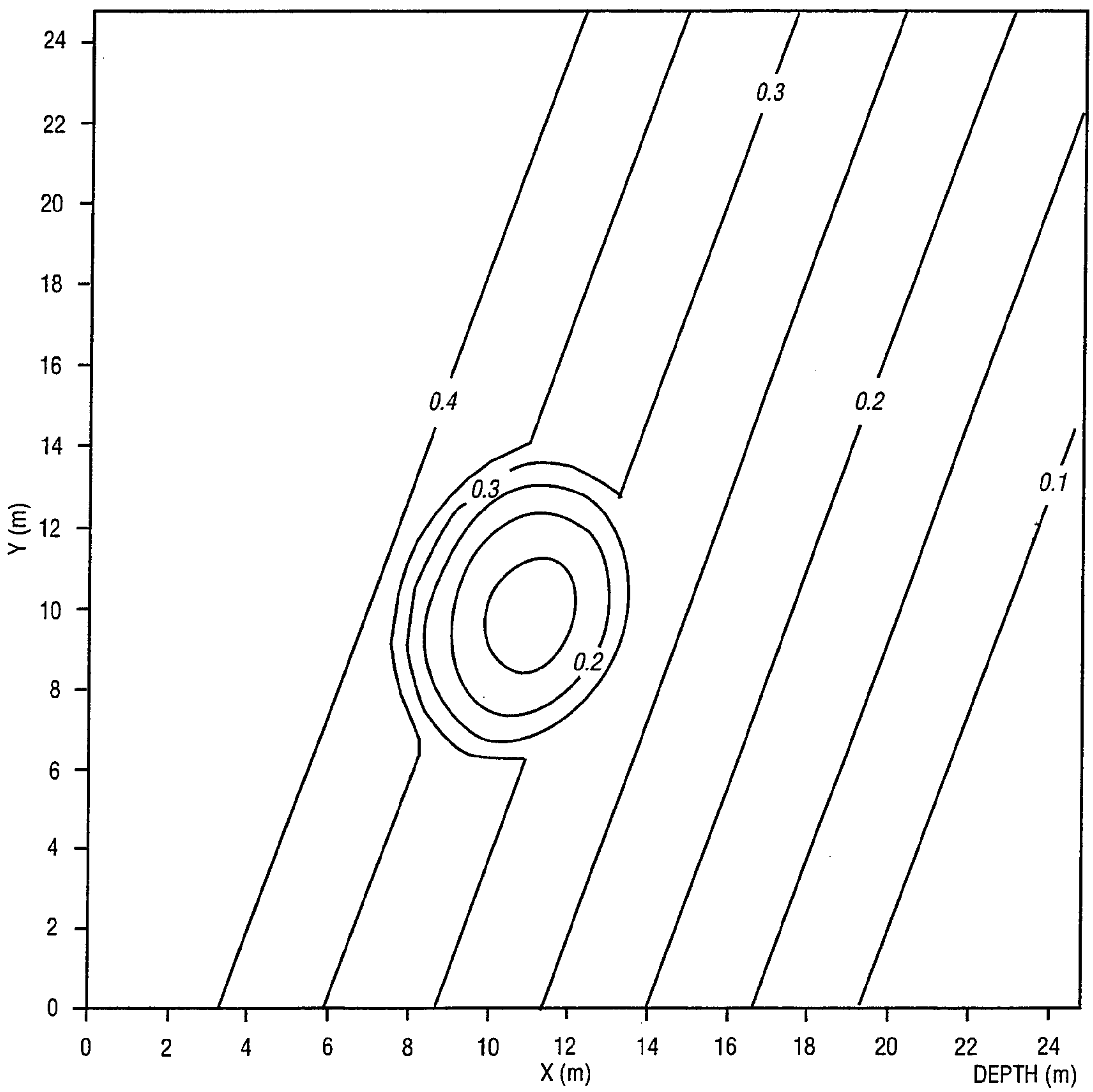

Fig. 3.2-1 — Bottom bathymetry of the Berkhoff-Booij-Radder elliptical shoal test. The bathymetry features an elliptical shoal on an angled, shallowing beach face.

test for REF/DIF1 simulates the conditions outlined in that paper. The test itself was designed to evaluate a situation of combined refraction and diffraction processes.

3.2.1 Pre-Test Preparations

For this test scenario, one of the data files included in the software package must be renamed as below:

Model Parameters Input File:

bbr.inp ⇒ indat.dat

The bottom bathymetry is contained in the Reference Grid Data File (bbr.bth).

3.3 Vincent and Briggs Shoal Experiment

Vincent and Briggs (1989) conducted a set of laboratory wave experiments studying wave focusing by a submerged elliptic shoal. Their shoal was the same size as that in the BBR study and rested on a flat bottom, not a sloping beach. The bottom bathymetry contours are shown on Fig. 3.3-1.

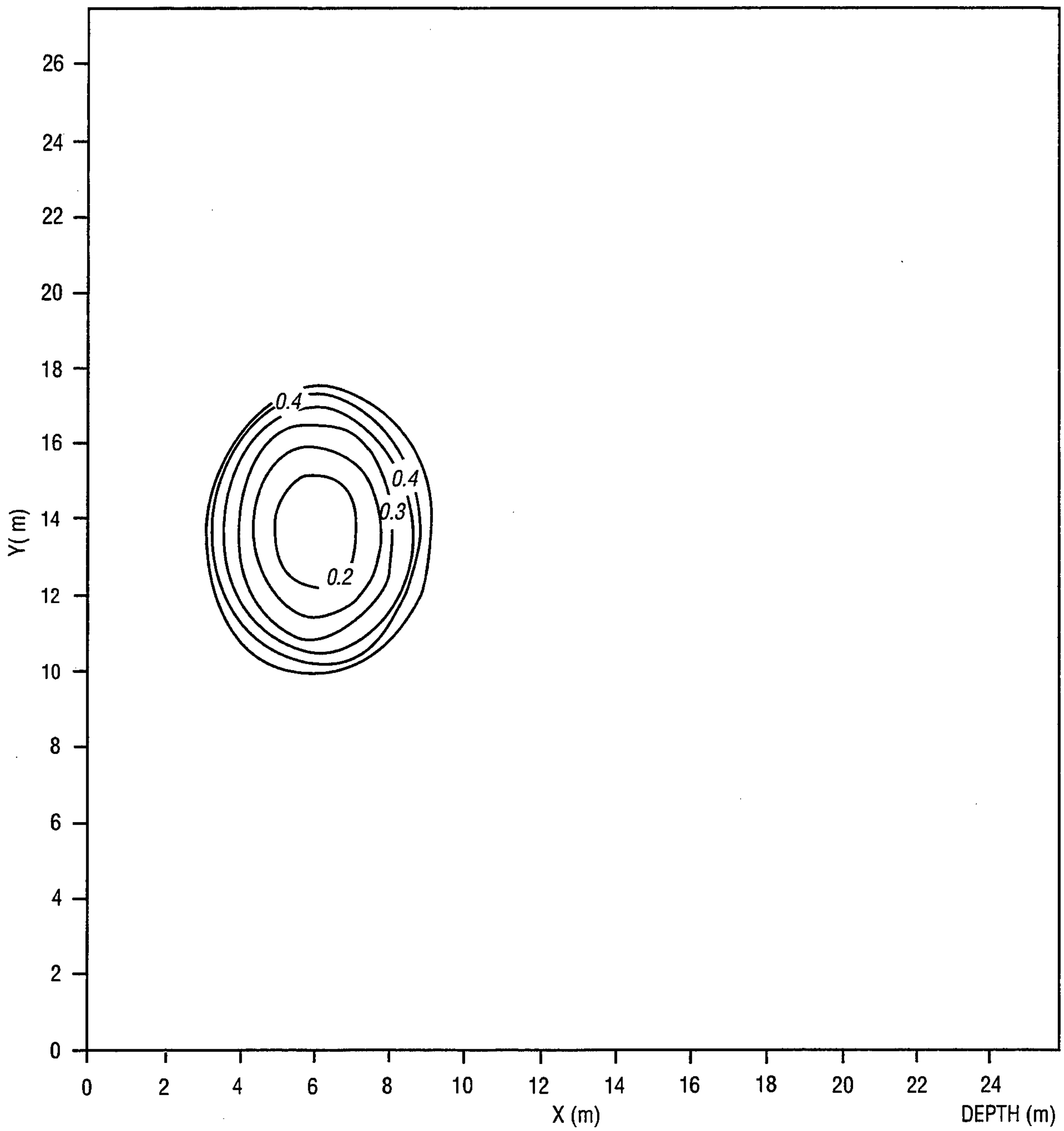

Fig. 3.3-1 — Bottom bathymetry of the Vincent and Briggs elliptical shoal test. Although similar to the test bathymetry of the Berkhoff-Booij-Radder test, the Vincent and Briggs test bathymetry features an elliptical shoal on a flat, otherwise uniform bottom.

3.3.1 Pre-Test Preparations

For the Vincent and Briggs test included in this report, two data files must be separately renamed (one for each test) as below:

Model Parameters Input File:

For this test (with ntype = 1, composite nonlinear dispersion), change the following file name:

vbs.inp ⇒ indat.dat

The bottom bathymetry is contained in the Reference Grid Data File (vbs.bth); the same bathymetry grid is used for both scenarios.

3.4 DELILAH Field Study Comparison Test

The near-shore community undertook a series of detailed field experiments (DELILAH) at Duck, NC. A scenario simulating one of those periods of observation was included in the VTR as a comparison between field study results and REF/DIF1 modeling of the field situation. The test provides a useful evaluation of a true situation with complex bathymetry and an angled incoming wave. The bottom bathymetry is presented in Fig. 3.4-1; the bottom features and slope are irregular.

3.4.1 Pre-Test Preparations

For this test in particular, two data files are included in the software package and one must be renamed as below:

Model Parameters Input File:

duck.inp ⇒ indat.dat

The bottom bathymetry is contained in the Reference Grid Data File (duck.bth).

3.5 Wave-Current Interaction Test

The VTR included a test scenario evaluating the ability of REF/DIF1 to account for wave-current interactions. That test scenario evaluated the Kirby (1984) wave-current interaction solution as implemented within REF/DIF1. The wave-current interaction scenario included in this document is identical to one of the tests in the VTR and examines the propagation of a wave against an "opposing" current on a flat bottom. The current field parallel to the direction of wave propagation (parallel to the x-axis) is shown in Fig. 3.5-1.

3.5.1 Pre-Test Preparations

For this test in particular, two data files are included in the software package and one must be renamed, as below:

Model Parameters Input File:

wacur.inp ⇒ indat.dat

The bottom bathymetry and current data are contained in the Reference Grid Data File (oppcur.bth).

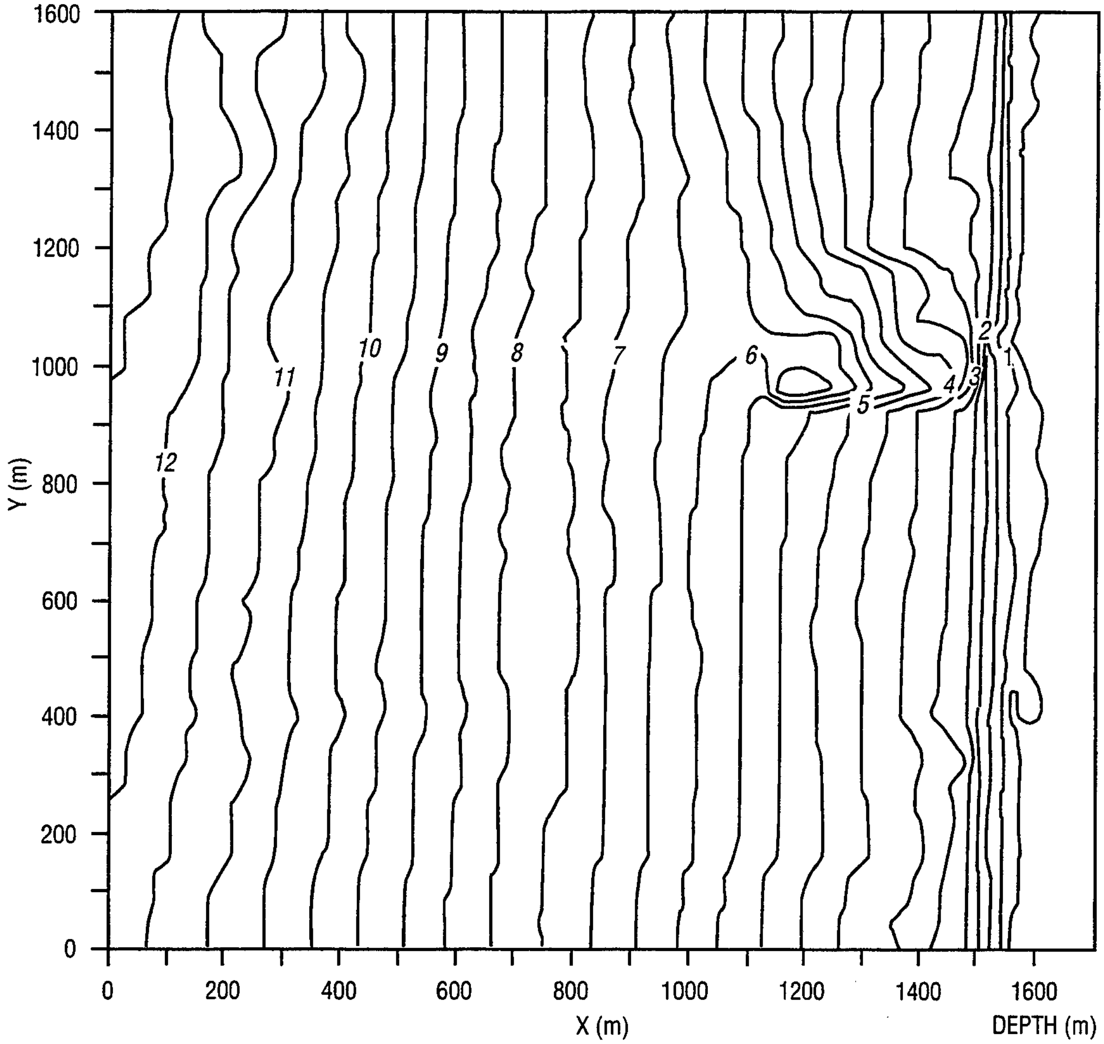

Fig. 3.4-1 — Bottom bathymetry (in meters) at the DELILAH test site in Duck, NC. The survey area stretches from 12-m depths offshore to the beach.

4.0 TEST DESCRIPTIONS

The tests detailed in this document are a subset of those included in the VTR. A copy of the Wave Height Data Files output for each test are included in the OAML software package. This report includes a plot of the appropriate wave height results for each test; some of the test scenarios include a tabular listing of resulting wave heights at particular positions in the study areas. With these resources the user should be able to test whether the software was ported correctly.

As mentioned previously, the information contained in the Model Parameters Input File for each test is included on a set of lists in App. A. These are provided for easy reference. Also, one can easily recreate each Model Parameters Input File for each test using the Model Parameterization Primary Software Unit of the CSCI.

4.1 Shoaling and Refraction Test

The test case was conducted for 1-m and 10-s period waves at 30° to the shore.

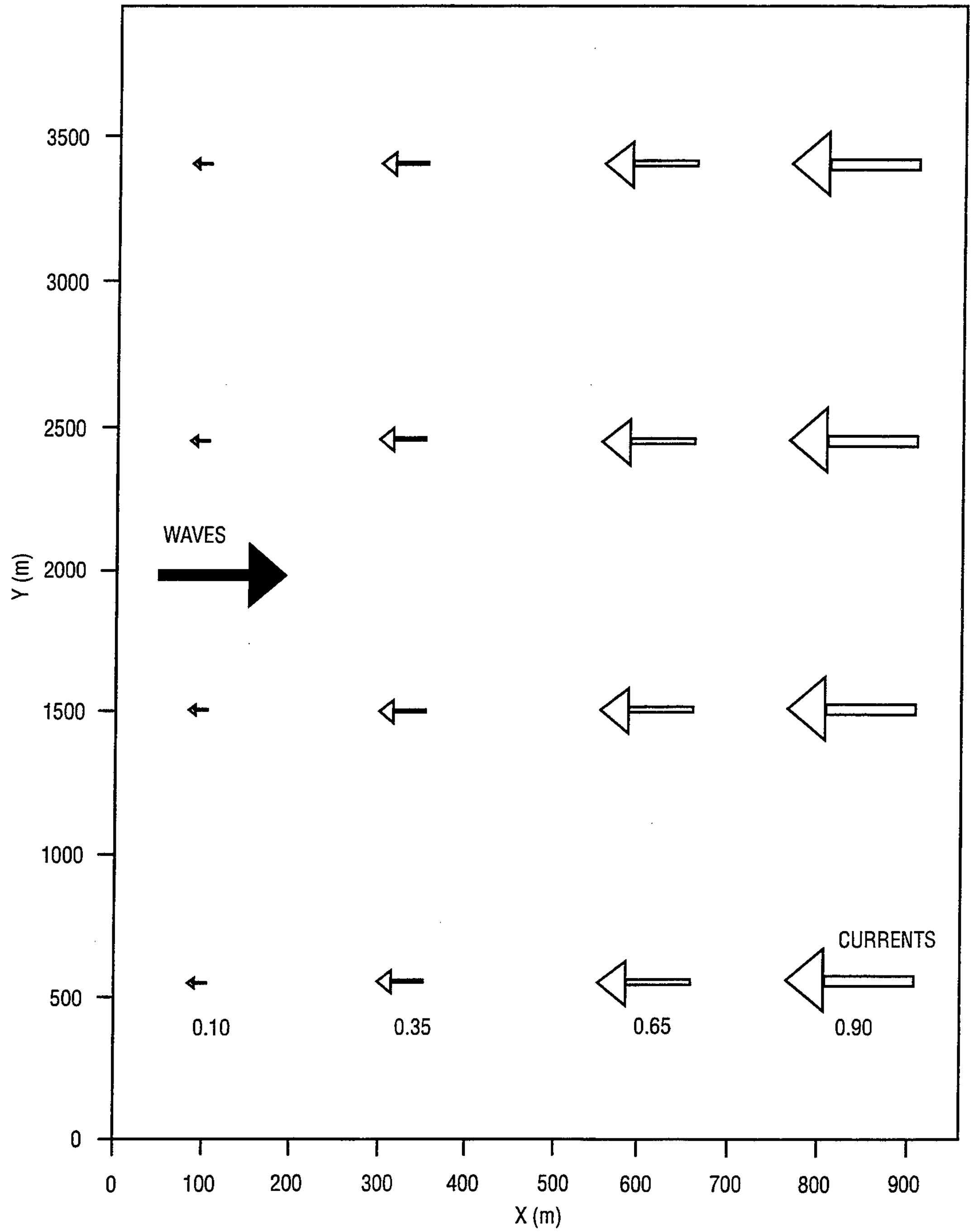

Fig. 3.5-1 — Water column current field parallel to the direction of wave propagation for the wave/opposing current interaction test. Current speeds are in m/s^{-1}.

4.1.1 Test Inputs

The input data file previously prepared and discussed in Sec. 3.1.1 serves as test input. The bathymetry file (sandr.bth) also serves as test input and is correct as included in the OAML package.

4.1.2 Test Results and Evaluation

The results from this test show the increasing of wave height towards towards shore as the result of shoaling and refraction. The decay of wave height passing 600 ft is the result of depth-induced breaking. The expected wave height results along a central axis are plotted in Fig. 4.1.2-1; the square symbols denote the positions as listed on the table below. The output file from testing of the ported software can be compared directly to the output file already included (sandr.hgt). A short list of wave height values and their array and spatial positions is included in the table below.

ARRAY POSITION	X POSITION	Y POSITION	WAVE HEIGHT (m)
(26,1286)	75.0	385.5	1.013
(51,1286)	150.0	385.5	1.028
(76,1286)	225.0	385.5	1.048
(101,1286)	300.0	385.5	1.073
(126,1286)	375.0	385.5	1.105
(151,1286)	450.0	385.5	1.148
(171,1286)	525.0	385.5	1.209
(201,1286)	600.0	385.5	1.031
(226,1286)	675.0	385.5	0.403
(236,1286)	705.0	385.5	0.289

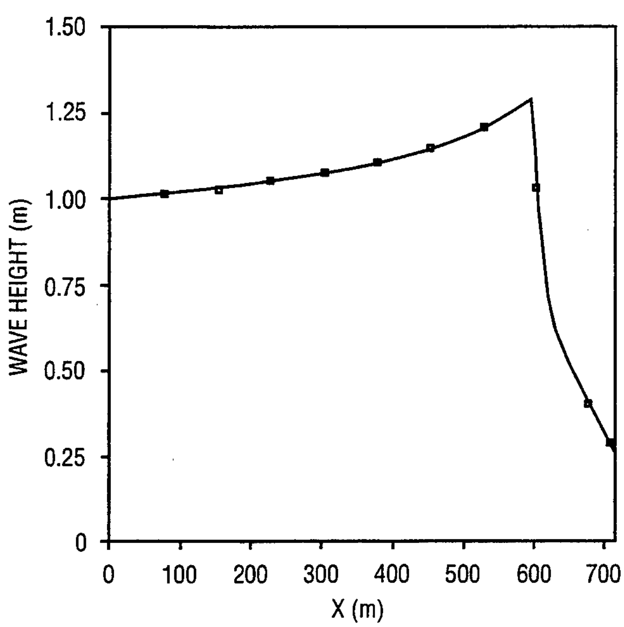

Fig. 4.1.2-1 — Wave height distribution along the central axis Y = 385.5 for the shoaling and refraction test

4.2 Berkhoff-Booij-Radder Shoal Test

The test case was conducted for waves of 0.0464-m height and 1-s periods incoming perpendicular (0°) to the shore.

4.2.1 Test Inputs

The files discussed in Sec. 3.2.1 serve as test input.

4.2.2 Test Results and Evaluation

The test results should show focusing of the wave energy on the shoal with the effects of diffraction. The expected wave height results are plotted in Fig. 4.2.2-1. The output file from testing of the ported software can be compared directly to the output file already included (bbr.hgt). A short table of wave height values and their array and spatial positions are included in the short table below:

ARRAY POSITION	X POSITION	Y POSITION	WAVE HEIGHT (m)
(3,41)	0.5	10.0	0.046
(11,41)	2.5	10.0	0.046
(21,41)	5.0	10.0	0.046
(31,41)	7.5	10.0	0.045
(41,41)	10.0	10.0	0.047
(51,41)	12.5	10.0	0.055
(61,41)	15.0	10.0	0.090
(71,41)	17.5	10.0	0.093
(81,41)	20.0	10.0	0.078
(91,41)	22.5	10.0	0.068
(97,41)	24.0	10.0	0.062

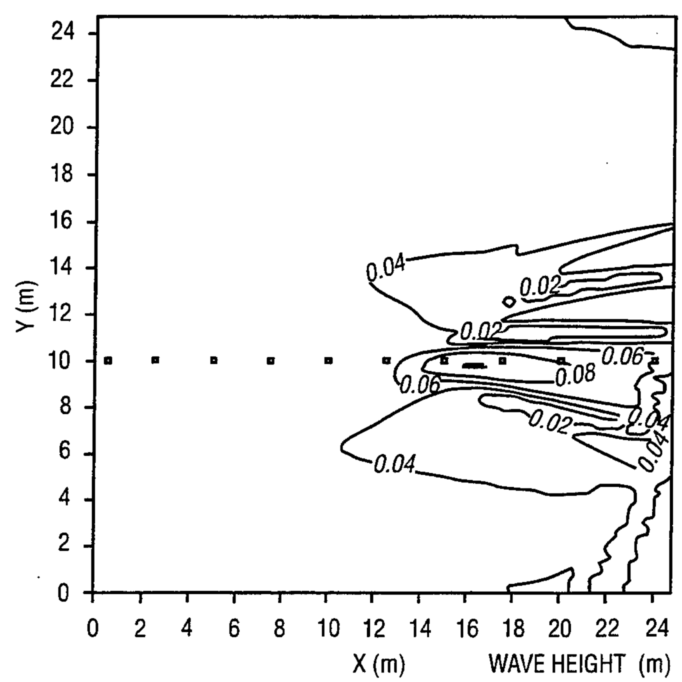

Fig. 4.2.2-1 — Wave height distribution for the Berkhoff-Booij-Radder test

The positions highlighted above are indicated on Fig. 4.2.2-1.

4.3 Vincent and Briggs Shoal Test

The REF/DIF1 solution used with the Vincent and Briggs bathymetry uses Stokes-to-Hedges combined dispersion ($ntype = 1$). All of the results included in this section modeled waves 0.055-m height of 1.3-s periods incoming at 0° to the shoal.

4.3.1 Test Inputs

The files discussed in Sec. 3.3.1 serve as test input.

4.3.2 Test Results and Evaluation

The test results should show focusing of the wave energy by the shoal with the effects of diffraction. The expected wave height results are plotted on Fig. 4.3.2-1. The output file from testing of the ported software can be compared directly to the output file already included (vbs.hgt). A short table of wave height values and their array and spatial positions is included below.

ARRAY POSITION	X POSITION	Y POSITION	WAVE HEIGHT (m)
(9,202)	1.04	26.01	0.055
(25,187)	3.11	24.07	0.055
(40,172)	5.05	22.13	0.055
(56,156)	7.12	20.06	0.055
(71,141)	9.06	18.12	0.046
(87,125)	11.13	16.05	0.042
(102,110)	13.07	14.10	0.102
(117,94)	15.01	12.03	0.038
(133,79)	17.08	10.09	0.052
(148,63)	19.02	8.02	0.058
(164,48)	21.09	6.08	0.038
(179,32)	23.03	4.01	0.065
(195,17)	25.10	2.07	0.058

The positions listed above are highlighted on the solution figure.

4.4 DELILAH Field Study Comparison Test

The test case was conducted for waves of 0.52-m height and 9.71-s periods incoming at –44° (southeast) to the shore. For this test case the tide level was at –0.64 m.

4.4.1 Test Inputs

The files discussed in Sec. 3.4.1 serve as test input.

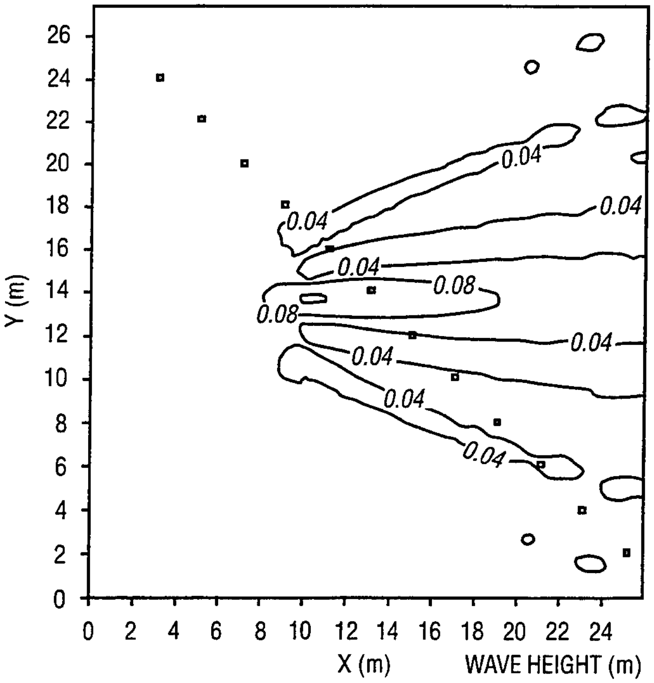

Fig. 4.3.2-1 — Wave height distribution for the Vincent and Briggs shoal test

4.4.2 Test Results and Evaluation

The wave field should exhibit an irregular distribution of wave height contours. The distribution of wave height profiles is shown in Fig. 4.4.2-1. The positions highlighted on the figure are included in the table below. As expected, the wave height distribution is extremely irregular inshore. A short list of wave height values and their array and spatial positions is included below:

ARRAY POSITION	X POSITION	Y POSITION	WAVE HEIGHT (m)
(4,28)	40.36	800.0	0.521
(16,28)	201.80	800.0	0.523
(28,28)	363.24	800.0	0.535
(34,28)	443.96	800.0	0.523
(51,28)	686.11	800.0	0.531
(64,28)	861.01	800.0	0.574
(73,28)	982.08	800.0	0.572
(89,28)	1197.33	800.0	0.574
(100,28)	1358.77	800.0	0.611
(109,28)	1479.85	800.0	0.629
(115,28)	1560.57	800.0	0.724
(119,28)	1587.48	800.0	0.323
(135,28)	1619.43	800.0	0.300
(159,28)	1634.56	800.0	0.227
(171,28)	1640.33	800.0	0.063
(209,28)	1650.43	800.0	0.000

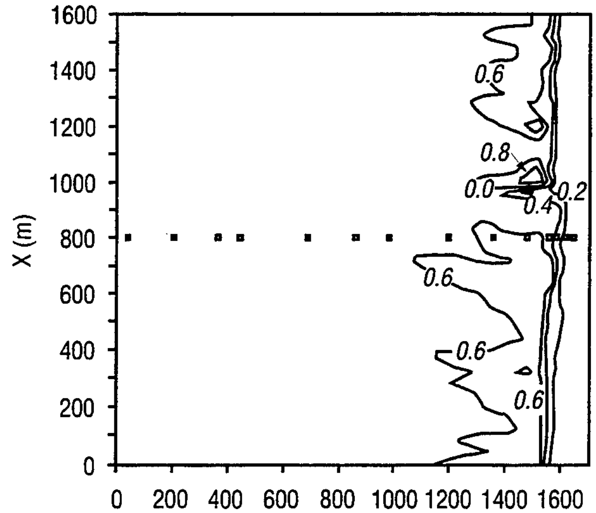

Fig. 4.4.2-1 — Wave height distribution for REF/DIF1 modeling of the DELILAH field test

It should be noted that it is often necessary to run the wave model at finer frequency and angular bandwidths over an area with a more complex bathymetry. For example, over the Southern California Bight, 0.002 Hz and 1° angular bandwidth are required (O'Reilly and Guza 1993).

4.5 Wave-Current Interaction Test

The test case was conducted for waves of 1-m height and 5-s periods incoming perpendicular (0°) to the shore.

4.5.1 Test Inputs

The files discussed in Sec. 3.5.1 serve as test input.

4.5.2 Test Results and Evaluation

The test results show increasing wave height and decreasing wave speed (decreasing wavelength) as the wave is slowed by the increasing opposing current speed. An overhead view of the wave height contours of the REF/DIF1 solution is included in Fig. 4.5.2-1. A profile view of the wave height distribution is provided in Fig. 4.5.2-2. As expected, the wave heights increase and wavelengths decrease as the speed of the opposing current increases.

5.0 REQUIREMENTS TRACEABILITY

The scientific validity and applicability of the REF/DIF1 model were evaluated in the VTR.

6.0 NOTES

6.1 Acronyms and Abbreviations

BBR	Berkhoff-Booij-Radder
CSCI	Computer Software Configuration Item

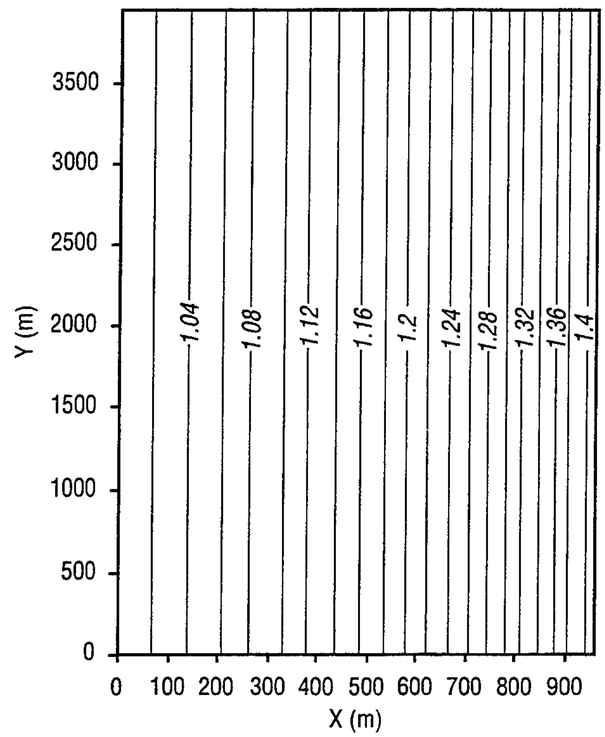

Fig. 4.5.2-1 — Overhead view of the wave height distribution of the wave opposing current interaction test

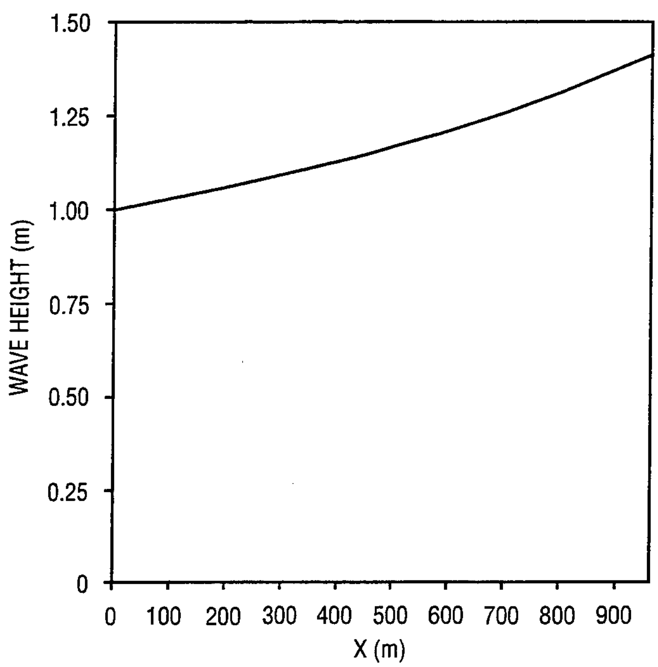

Fig. 4.5.2-2 — Side profile of the wave height distribution of the wave opposing current interaction test

m	meter
m/s^{-1}	meter per second
MIL-STD	Military Standard
OAML	Oceanographic and Atmospheric Master Library
REF/DIF1	Combined Refraction/Diffraction Model, Version 2.5
s	second
STD	Software Test Document
VTR	Validation Test Report

7.0 ACKNOWLEDGMENTS

The authors wish to thank Prof. Edward Thornton/NPS, Prof. Robert Guza/Scripps, and Dr. Chuck Long/CERC for providing the DELILAH data set. This work was sponsored by the Chief of Naval Research (OP-096) under program element 0603207N-Surf Model Upgrade Project, CDR Steve Payne.

Elements of the SDD in particular were derived wholly or in part from the *Combined Refraction/ Diffraction Model REF/DIF I Version 2.5 Documentation and User's Manual* by James T. Kirby and Robert A. Dalrymple (1994).

APPENDIX A

MODEL PARAMETER INFORMATION FOR EACH SCENARIO

Shoaling and Refraction Test Input

fname1 = sandr.bathym	Icur = 0
fname2 = outdat.dat	Ibc = 1
fname3 = subdat.dat	Dxr = 3
fname4 = wave.dat	Dyr = 3
fname5 = owave.dat	Dt = 10
fname6 = surface.dat	Ispace = 0
fname7 = bottomu.dat	Nd = 10
fname8 = angle.dat	Iff = 0,0,0
fname10 = refdif1.log	Isp = 0
fname11 = height.dat	Iinput = 1
fname12 = sxx.dat	Ioutput = 1
fname13 = sxy.dat	Iwave = 1
fname14 = syy.dat	Nfreqs = 1
fname15 = depth.dat	Freqs = 10.0
mr = 240	Tide = 0.0
nr = 240	Nwavs = 1
iu = 1	Amp = 0.5
ntype = 0	Dir = 30.0

Berkhoff, Booij, and Radder Shoal Test Input

fname1 = bbr.bathym	Icur = 0
fname2 = outdat.dat	Ibc = 0
fname3 = subdat.dat	Dxr = 0.25
fname4 = wave.dat	Dyr = 0.25
fname5 = owave.dat	Dt = 10.0
fname6 = surface.dat	Ispace = 0
fname7 = bottomu.dat	Nd = 1
fname8 = angle.dat	Iff = 1,0,0
fname10 = refdif1.log	Isp = 0
fname11 = height.dat	Iinput = 1
fname12 = sxx.dat	Ioutput = 1
fname13 = sxy.dat	Iwave = 1
fname14 = syy.dat	Nfreqs = 1
fname15 = depth.dat	Freqs = 1.0
mr = 100	Tide = 0.0
nr = 100	Nwavs = 1
iu = 1	Amp = 0.0232
ntype = 1	Dir = 0.0

Vincent and Briggs Shoal Test Input

fname1 = vbs.bathym	Icur = 0
fname2 = outdat.dat	Ibc = 1
fname3 = subdat.dat	Dxr = 0.1294
fname4 = wave.dat	Dyr = 0.1294
fname5 = owave.dat	Dt = 0.5
fname6 = surface.dat	Ispace = 0
fname7 = bottomu.dat	Nd = 1
fname8 = angle.dat	Iff = 0,0,0
fname10 = refdif1.log	Isp = 0
fname11 = height.dat	Iinput = 1
fname12 = sxx.dat	Ioutput = 1
fname13 = sxy.dat	Iwave = 1
fname14 = syy.dat	Nfreqs = 1
fname15 = depth.dat	Freqs = 1.3
mr = 200	Tide = 0.0
nr = 212	Nwavs = 1
iu = 1	Amp = 0.0275
ntype = 1	Dir = 0.0

DELILAH Study Comparison Test Input

fname1 = duck.bathym	Icur = 0
fname2 = outdat.dat	Ibc = 1
fname3 = subdat.dat	Dxr = 13.4532
fname4 = wave.dat	Dyr = 13.4532
fname5 = owave.dat	Dt = 10.0
fname6 = surface.dat	Ispace = 0
fname7 = bottomu.dat	Nd = 1
fname8 = angle.dat	Iff = 0,0,0
fname10 = refdif1.log	Isp = 0
fname11 = height.dat	Iinput = 1
fname12 = sxx.dat	Ioutput = 1
fname13 = sxy.dat	Iwave = 1
fname14 = syy.dat	Nfreqs = 1
fname15 = depth.dat	Freqs = 9.71
mr = 128	Tide = -0.64
nr = 55	Nwavs = 1
iu = 1	Amp = 0.26
ntype = 0	Dir = -44.0

Wave-Current Interaction Test Input

fname1 = current.bathym	Icur = 1
fname2 = outdat.dat	Ibc = 1
fname3 = subdat.dat	Dxr = 40.0
fname4 = wave.dat	Dyr = 40.0
fname5 = owave.dat	Dt = 10.0
fname6 = surface.dat	Ispace = 0
fname7 = bottomu.dat	Nd = 1
fname8 = angle.dat	Iff = 0,0,0
fname10 = refdif1.log	Isp = 0
fname11 = height.dat	Iinput = 1
fname12 = sxx.dat	Ioutput = 1
fname13 = sxy.dat	Iwave = 1
fname14 = syy.dat	Nfreqs = 1
fname15 = depth.dat	Freqs = 5.0
mr = 25	Tide = 0.0
nr = 100	Nwavs = 1
iu = 1	Amp = 0.5
ntype = 1	Dir = 0.0