ATTACHMENT A: Results of FUNWAVE-TVD verification (Purpose A)

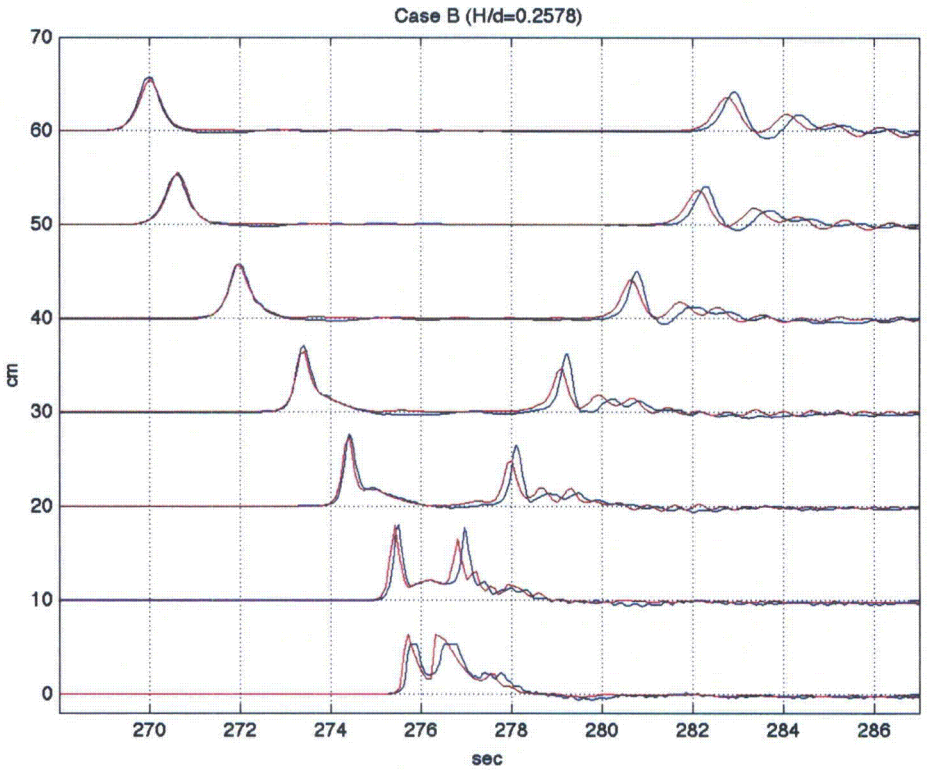

Figure A-1. FUNWAVE-TVD spherical version results (red lines) and laboratory data (blue lines) in the benchmark of solitary wave on a composite beach. Breaking wave case, H=d = 0:2578 initial wave.

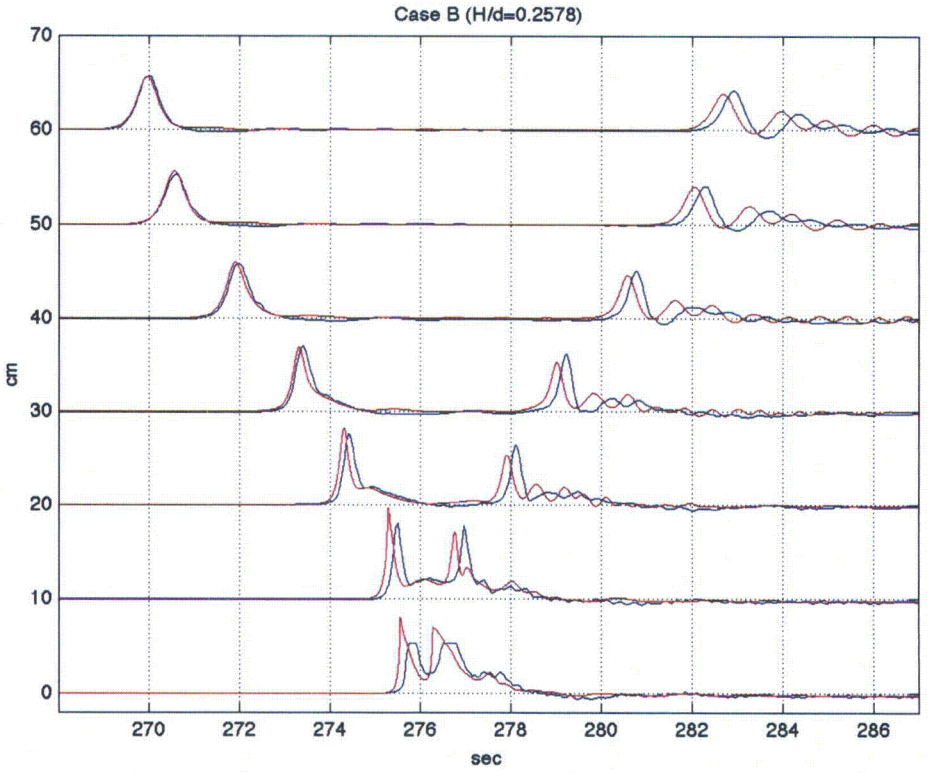

Figure A-2. FUNWAVE-TVD Cartesian version results (red lines) and laboratory data (blue lines) in the benchmark of solitary wave on a composite beach. Breaking wave case, H=d = 0:2578 initial wave.

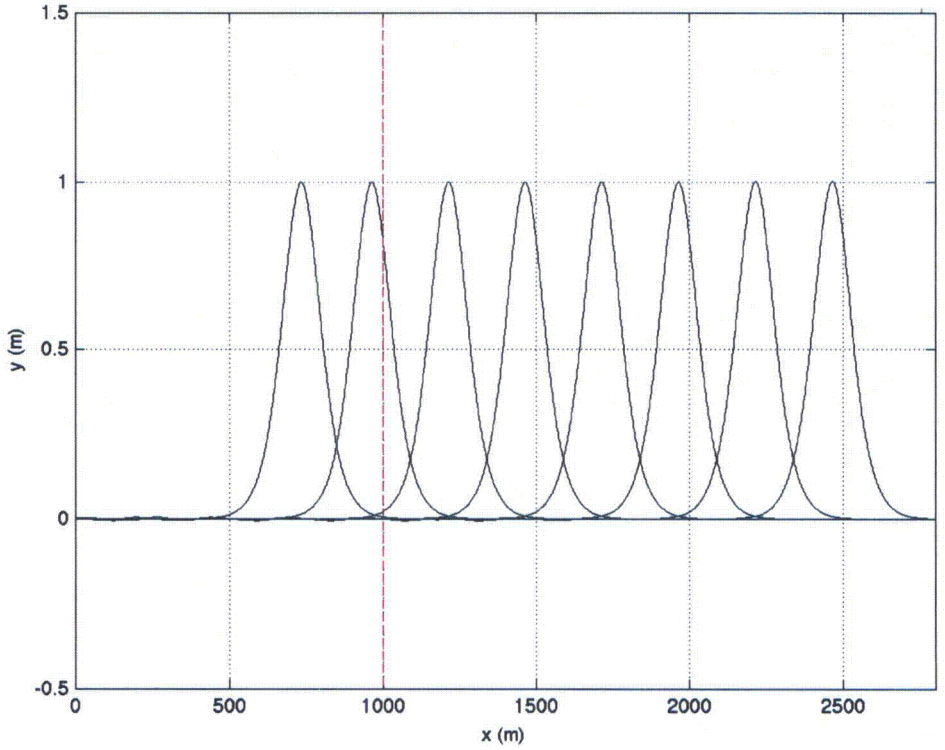

Figure A-3. Solitary wave calculated by FUNWAVE-TVD in the large domain at Time = 25s, 50s, 75s, 100s, 125s, 150s, 175s, and 200s. The red dashed line represents the nesting boundary.

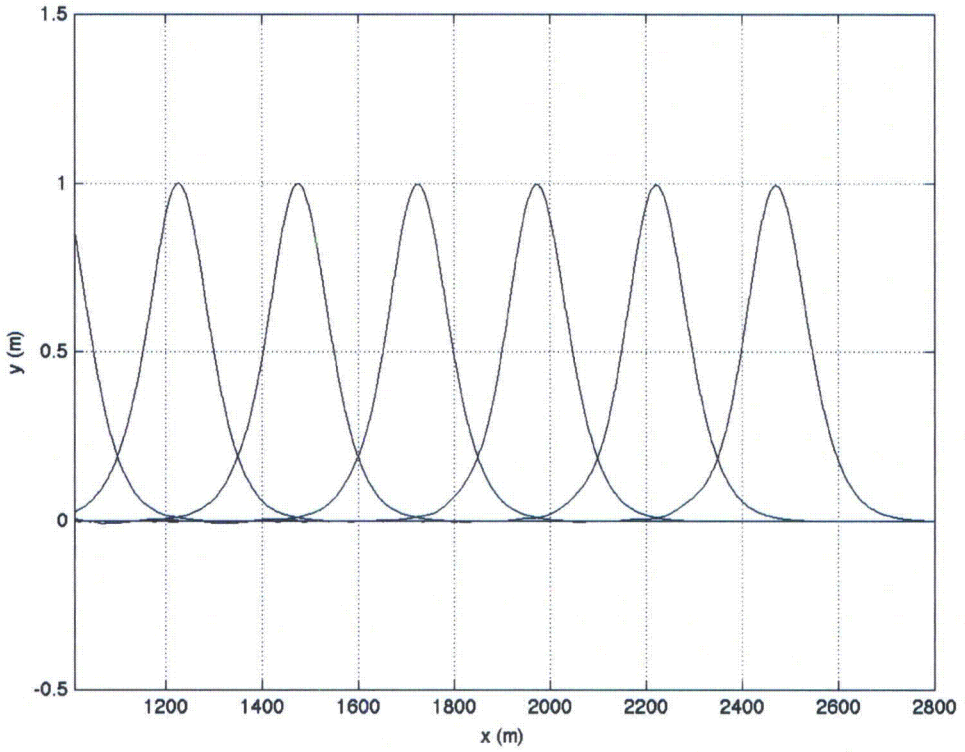

Figure A-4. Solitary wave calculated by FUNWAVE-TVD in the nested small domain at Time = 50s, 75s, 100s, 125s, 150s, 175s, and 200s.

ATTACHMENT B: Results of NHWAVE verification (Purpose B)

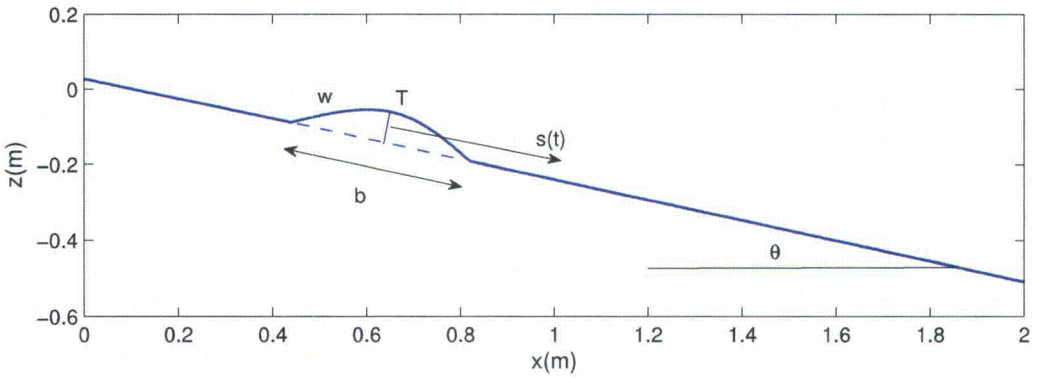

Figure B-1. Vertical cross section for numerical setup of tsunami landslide.

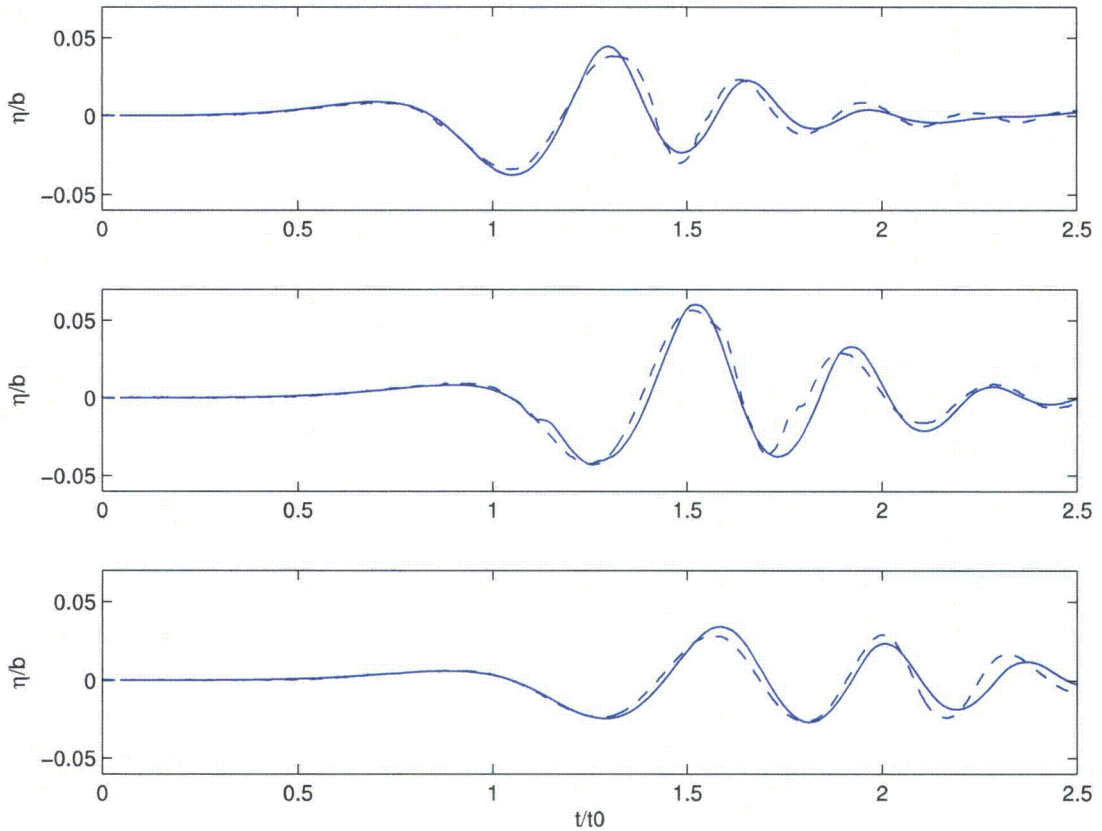

Figure B-2. Comparisons between numerical (solid lines) and experimental (dashed lines) free surface elevation for landslide-generated waves at three wave gauges with d = 61 mm. Gauge coordinates (r, φ)(mm, deg): (a) (1,510,13.4); (b) (1,929,0); (c) (1,992,14.5), where r is radial distance and φ is the azimuth.

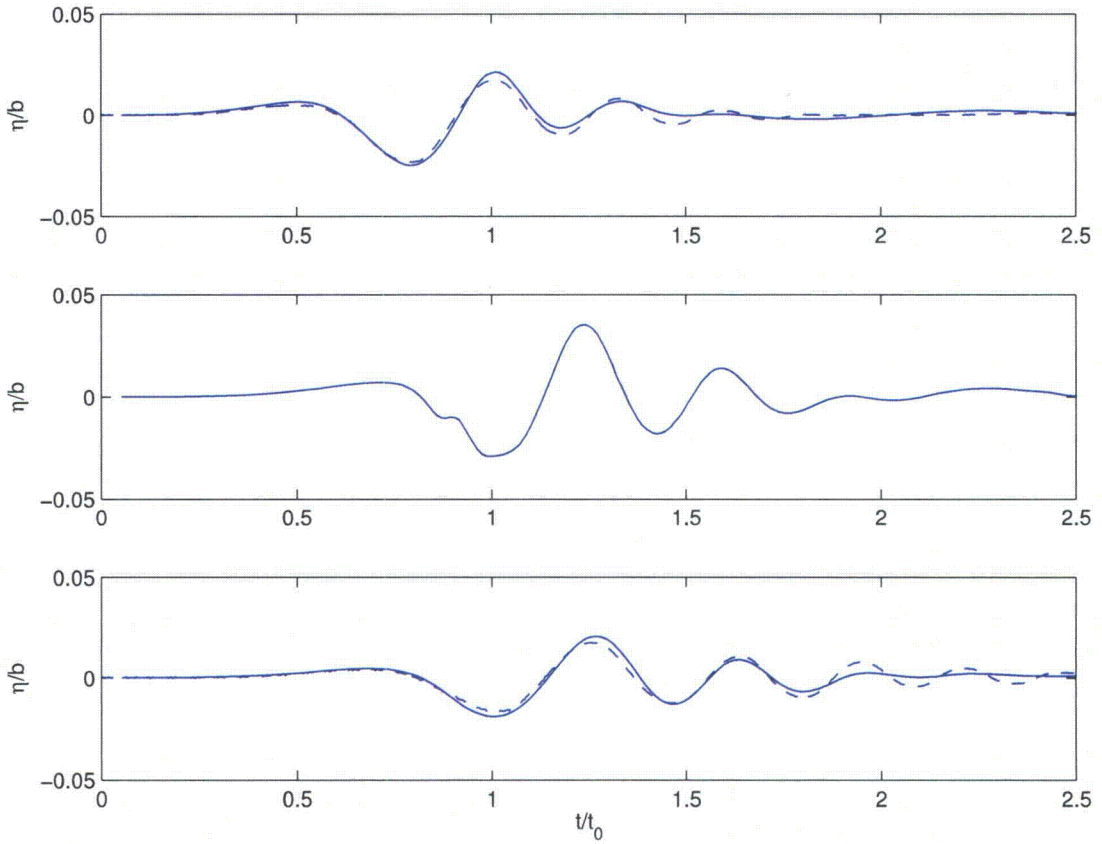

Figure B-3. Comparisons between numerical (solid lines) and experimental (dashed lines) free surface elevation for landslide-generated waves at three wave gauges with d = 120 mm. Gauge coordinates (r, φ)(mm, deg): (a) (1,510,13.4); (b) (1,929,0); (c) (1,992,14.5), where r is radial distance and φ is the azimuth. Note that experimental data are not available in the middle panel.

ATTACHMENT C: Results of verification of bathymetry/topography files (Purpose C)

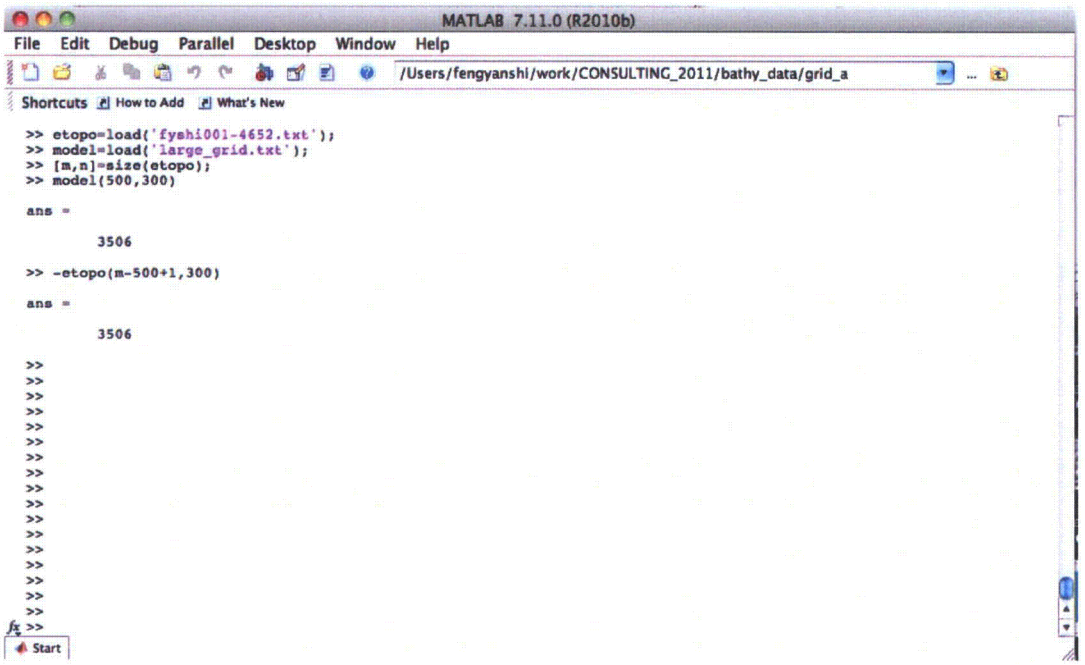

Figure C-1. Direct comparison between the data downloaded from ETOPO 1 and FUNWAVE input file for Grid A, Venezuela case.

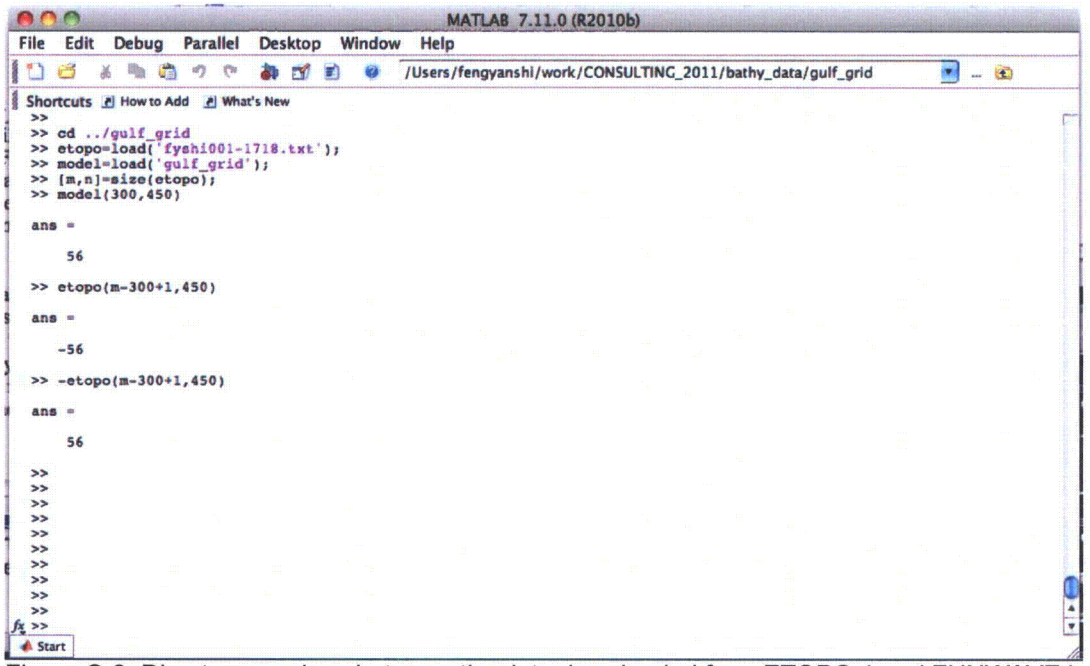

Figure C-2. Direct comparison between the data downloaded from ETOPO 1 and FUNWAVE input file for Grid A, landslide cases.

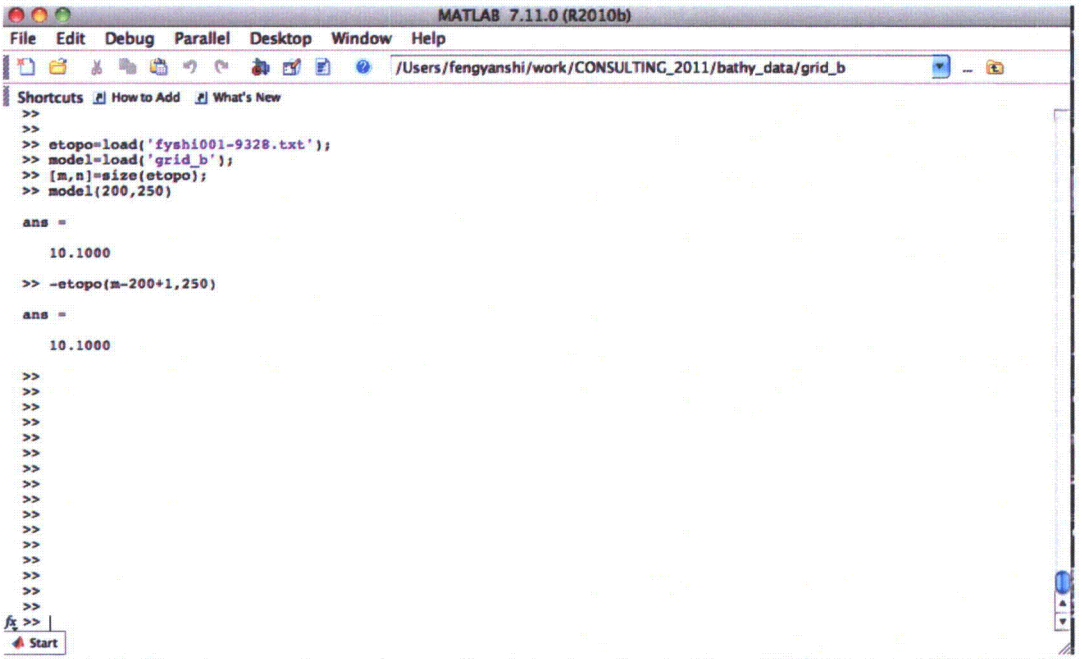

Figure C-3. Direct comparison between the data downloaded from ETOPO 1 and FUNWAVE input file for Grid B.

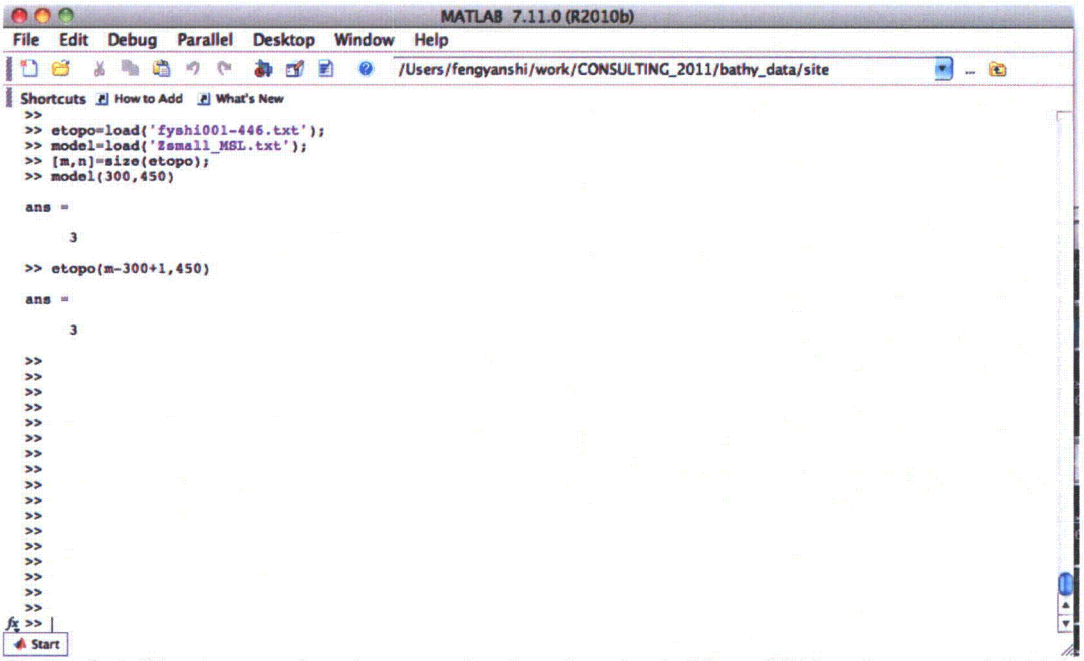

Figure C-4. Direct comparison between the data downloaded from CRM and converted Grid C with the MSL reference.

Program C-1: mkbathy_a_2m.f

```
      program adjust_bathymetry
      parameter(m=2500,n=2500)
          character aa
      real depth1(m,n),x(m),y(n)
      open(1,file='fyshi001-4652.asc')
          read(1,*)aa,m1
          read(1,*)aa,n1
          read(1,*)aa,xstart
          read(1,*)aa,ystart
          read(1,*)aa,size
          read(1,*)aa,nothing
      xstart=360+xstart

          do i=1,m1
           x(i)=xstart+(i-1)*size
          enddo
          do j=1,n1
           y(j)=ystart+(j-1)*size
          enddo

      do j=n1,1,-1
       read(1,*)(depth1(i,j),i=1,m1,1)
      enddo

      open(2,file='large_grid.txt')
      do j=1,n1
       write(2,100)(-depth1(i,j),i=1,m1)
      enddo
100   format(1500f10.2)
      close(2)
     end
```

Program C-2: mkbathy_a_1m.f

```
      program adjust_bathymetry
      parameter(m=2500,n=2500)
          character aa
      real depth1(m,n),x(m),y(n)
      open(1,file='fyshi001-1718.asc')
          read(1,*)aa,m1
          read(1,*)aa,n1
          read(1,*)aa,xstart
          read(1,*)aa,ystart
          read(1,*)aa,size
          read(1,*)aa,nothing
      xstart=360+xstart

          do i=1,m1
           x(i)=xstart+(i-1)*size
          enddo
          do j=1,n1
           y(j)=ystart+(j-1)*size
```

```
            enddo

          do j=n1,1,-1
           read(1,*)(depth1(i,j),i=1,m1,1)
          enddo

          open(2,file='gulf_grid')
          do j=1,n1
           write(2,100)(-depth1(i,j),i=1,m1)
          enddo
100       format(1500f10.2)
          close(2)

          end
```

Program C-3: mkbathy_b_15s.f

```
          program adjust_bathymetry
          parameter(m=2500,n=2500)
              character aa
          real depth1(m,n),x(m),y(n)
          open(1,file='fyshi001-9328.asc')
              read(1,*)aa,m1
              read(1,*)aa,n1
              read(1,*)aa,xstart
              read(1,*)aa,ystart
              read(1,*)aa,size
              read(1,*)aa,nothing
          xstart=360+xstart

              do i=1,m1
               x(i)=xstart+(i-1)*size
              enddo
              do j=1,n1
               y(j)=ystart+(j-1)*size
              enddo

          do j=n1,1,-1
           read(1,*)(depth1(i,j),i=1,m1,1)
          enddo

          open(2,file='grid_b')
          do j=1,n1-1
           write(2,100)(-depth1(i,j),i=1,m1-1)
          enddo
100       format(1500f10.2)
          close(2)

          end
```

Program C-4: mksite_3s.f

```
         program adjust_bathymetry
          parameter(m=2500,n=2500)
              character aa
```

```
      real depth1(m,n),x(m),y(n)
      open(1,file='fyshi001-4161.asc')
          read(1,*)aa,m1
          read(1,*)aa,n1
          read(1,*)aa,xstart
          read(1,*)aa,ystart
          read(1,*)aa,size
          read(1,*)aa,nothing
      xstart=360+xstart

          do i=1,m1
           x(i)=xstart+(i-1)*size
          enddo
          do j=1,n1
           y(j)=ystart+(j-1)*size
          enddo

      do j=n1,1,-1
       read(1,*)(depth1(i,j),i=1,m1,1)
      enddo

      open(2,file='site_grid.txt')
      do j=1,n1
       write(2,100)(-depth1(i,j),i=1,m1)
      enddo
100   format(1500f10.2)
      close(2)

        end
```

Program C-5: intp.m

```
clear all
raw=load('XYZ_coordinates_ft_and_m.txt');
x=raw(:,5);
y=raw(:,6);
z=raw(:,7);

x1=[min(x):30:max(x)];
y1=[min(y):30:max(y)];
[X,Y]=meshgrid(x1,y1);

Z=griddata(x,y,z,X,Y);

save -ASCII z30m.txt Z;
save -ASCII x30m.txt X;
save -ASCII y30m.txt Y;
```

Program C-6: nav_convert.m

```
clear all
x0=-82;
y0=24.3333333333333;

x1= -082.6466616;
```

```
y1=29.0609705;
x2=-082.6018743;
y2=29.0939305;


Xnav=load('x30m.txt');
Ynav=load('y30m.txt');
Znav=load('z30m.txt');

[m,n]=size(Xnav);

Xg=(Xnav-Xnav(1,1))/(Xnav(m,n)-Xnav(1,1))*(x2-x1)+x1;
Yg=(Ynav-Ynav(1,1))/(Ynav(m,n)-Ynav(1,1))*(y2-y1)+y1;

save -ASCII Xg30m.txt Xg;
save -ASCII Yg30m.txt Yg;
save -ASCII Zg30m.txt Znav;
```

Program C-7: process_site.m

```
clear all
fdir='';
zlow=5;
zhigh=25.0;
[B,X1,Y1]=readbatfile([fdir 'site_3s_most.txt']);
X1=X1-360.0;
B=-B;
[X,Y]=meshgrid(X1,Y1);

set(gcf,'units','inches','paperunits','inches','papersize', [8
3],'position',[1 1 9 4],'paperposition',[0 0 8 3]);

Xs=load('Xg30m.txt');
Ys=load('Yg30m.txt');
Zs=load('Zg30m.txt');
Zs=Zs-0.07 ! 0.07 is based on NOAA Cedar Key FL, ID: 8727520;

[m,n]=size(Zs);

Zs90m=griddata(Xs,Ys,Zs,X,Y);

z1=Zs90m;
z1(isnan(z1))=0.0;
z2=Zs90m;
z2(isnan(z2))=B(isnan(z2));

pcolor(X,Y,z2),shading flat
xlabel('Longitude (^\circ)');
ylabel('Latitude (^\circ)');
caxis([zlow,zhigh])
colorbar
axis([-83 -82.5 28.5 29.2])
save -ASCII Xsmall.txt X
save -ASCII Ysmall.txt Y
save -ASCII Zsmall_MSL.txt z2
```

ATTACHMENT D Results from verification of FUNWAVE-TVD input files from NHWAVE outputs (Purpose D)

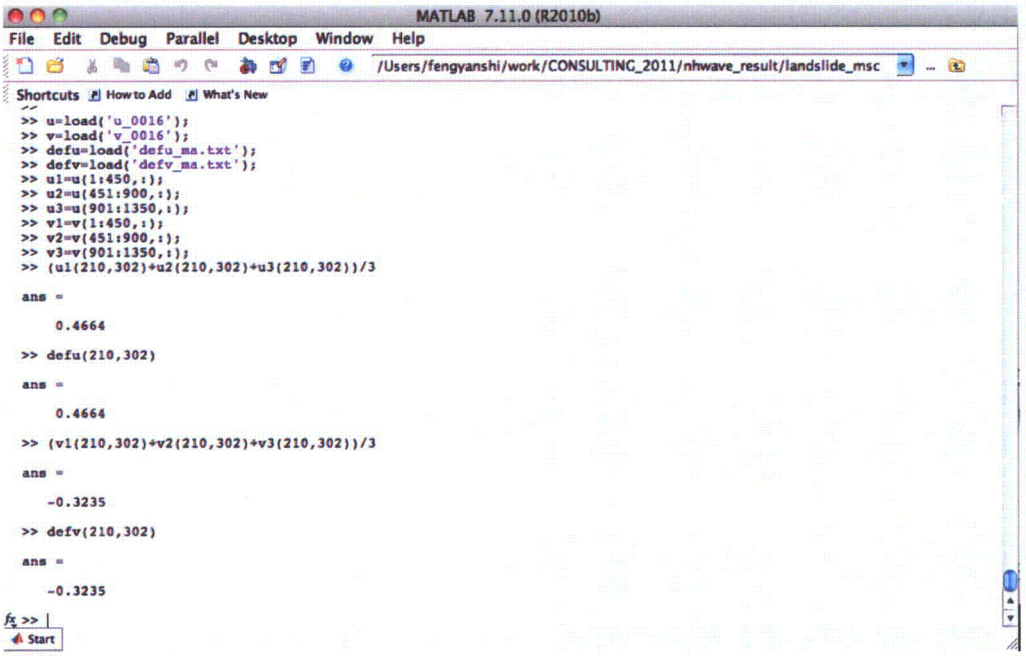

Figure D-1. Direct comparison between NHWAVE output file and FUNWAVE-TVD input file for Mississippi Canyon case.

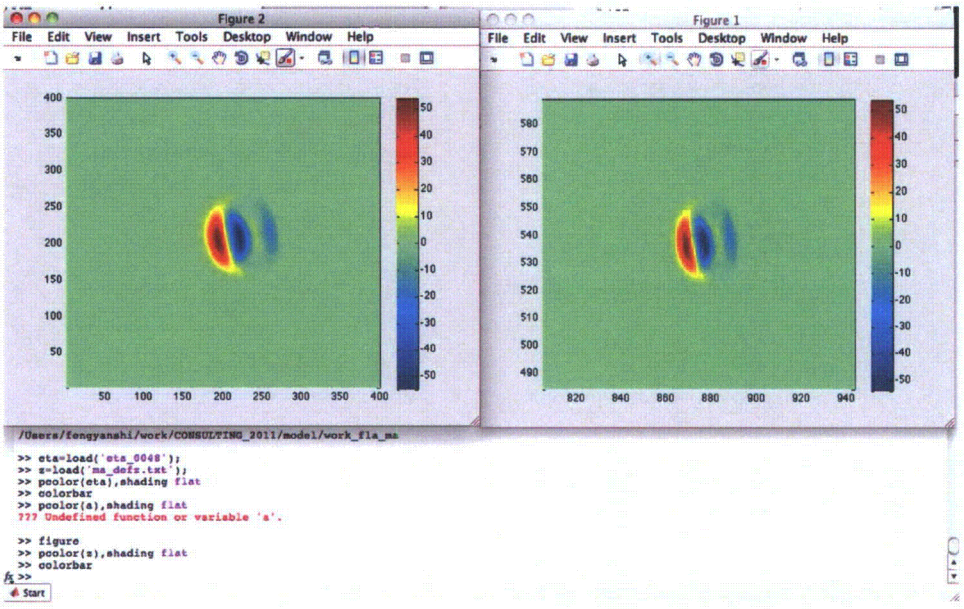

Figure D-2. Conversion between output data from NHWAVE and input data for FUNWAVE-TVD for Florida escarpment case.

Program D-1: interpolation2global.m

```
clear all
x0=-86.0;
y0=25.0;
R_earth = 6371000.0;
z=load('eta_0048');
u=load('u_0048');
v=load('v_0048');
u1=u(1:400,:);
u2=u(401:800,:);
u3=u(801:1200,:);
v1=v(1:400,:);
v2=v(401:800,:);
v3=v(801:1200,:);

uu=(u1+u2+u3)/3.0; % get averaged value
vv=(v1+v2+v3)/3.0;

[n1,m1]=size(z);
dx1=500.0/(R_earth*cos(y0*pi/180.0)*pi/180.0);
dy1=500.0/(R_earth*pi/180.0);
x1=x0+[0:m1-1]*dx1;
y1=y0+[0:n1-1]*dy1;
[X1,Y1]=meshgrid(x1,y1);

m2=1291;
n2=901;
dx2=1/60;
dy2=1/60;
x2=-99.5+[0:m2-1]*dx2;
y2=17.0+[0:n2-1]*dy2;
[X2,Y2]=meshgrid(x2,y2);

Z2=griddata(X1,Y1,z,X2,Y2);
U2=griddata(X1,Y1,uu,X2,Y2);
V2=griddata(X1,Y1,vv,X2,Y2);
Z2(isnan(Z2))=0.0;
U2(isnan(U2))=0.0;
V2(isnan(V2))=0.0;

save -ASCII ma_defz.txt Z2;
save -ASCII ma_defu.txt U2;
save -ASCII ma_defv.txt V2
```

ATTACHMENT E Results from verification of datum conversions (Purpose E)

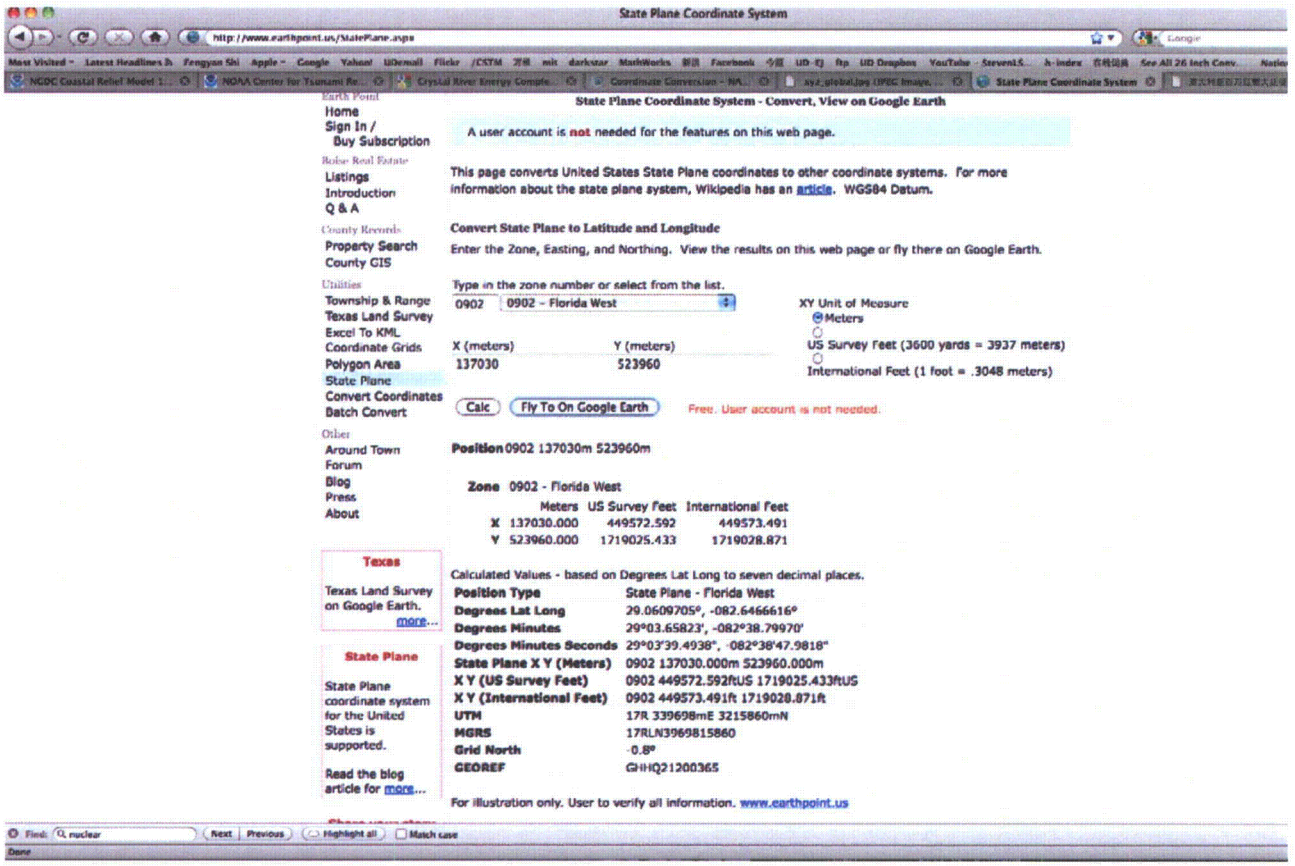

Figure E-1. Horizontal datum conversion between Florida State Plane Coordinates, West Zone, NAD 83 and global Latitude/Longitude coordinates.

Data Retrieval **Page 1 of 1**

TIDES & CURRENTS

Search

Home | Products | Programs | Partnerships | Education | Help

Station Home Page
Station Information
Tide / Water Level Data
Tide Predictions
Current Data
Meteorological Observations
Conductivity
PORTS
Operational Forecast System
Bench Mark Sheets
Datums
Harmonic Constituents
Sea Level Trends

Cedar Key, FL
Station ID: 8727520

Cedar Key, FL: Data Inventory
Page Help

Datums

Click HERE for printable version

Data Units: (•) Feet ( ) Meters [Apply Change]

Jun 29 2011 14:29 **ELEVATIONS ON STATION DATUM**
National Ocean Service (NOAA)

Station:	**8727520**	**T.M.:**	**0 W**
Name:	**CEDAR KEY, GULF OF MEXICO, FL**	**Units:**	**Feet**
Status:	**Accepted**	**Epoch:**	**1983-2001**

Datum	Value	Description
MHHW	5.60	Mean Higher-High Water
MHW	5.27	Mean High Water
DTL	3.70	Mean Diurnal Tide Level
MTL	3.85	Mean Tide Level
MSL	3.84	Mean Sea Level
MLW	2.44	Mean Low Water
MLLW	1.80	Mean Lower-Low Water
GT	3.80	Great Diurnal Range
MN	2.83	Mean Range of Tide
DHQ	0.34	Mean Diurnal High Water Inequality
DLQ	0.63	Mean Diurnal Low Water Inequality
HWI	6.49	Greenwich High Water Interval (in Hours)
LWI	0.34	Greenwich Low Water Interval (in Hours)
NAVD	4.06	North American Vertical Datum
Maximum	10.75	Highest Water Level on Station Datum
Max Date	19961007	Date Of Highest Water Level
Max Time	22:48	Time Of Highest Water Level
Minimum	-2.40	Lowest Water Level on Station Datum
Min Date	19470918	Date Of Lowest Water Level
Min Time	11:30	Time Of Lowest Water Level

To refer Water Level Heights to a Tidal Datum, apply the desired Datum Value.

Click HERE for further station information including New Epoch products.

To refer Water Level Heights to either
NGVD (National Geodetic Vertical Datum of 1929) or
NAVD (North American Vertical Datum of 1988), apply the values located at:
National Geodetic Survey

home | products | programs | partnerships | education | help

Disclaimers Contact Us Privacy Policy About CO-OPS For CO-OPS Employees Only Revised: 11/23/2005 NOAA / National Ocean Service

http://tidesandcurrents.noaa.gov/data_menu.shtml?stn=8727520 Cedar Key, FL&type=Dat... 6/29/2011

Figure E-2. Data at NOAA Cedar Key station used for vertical datum conversion.

ATTACHMENT F Backup of FUNWAVE-TVD Directory Listing (Purpose F)

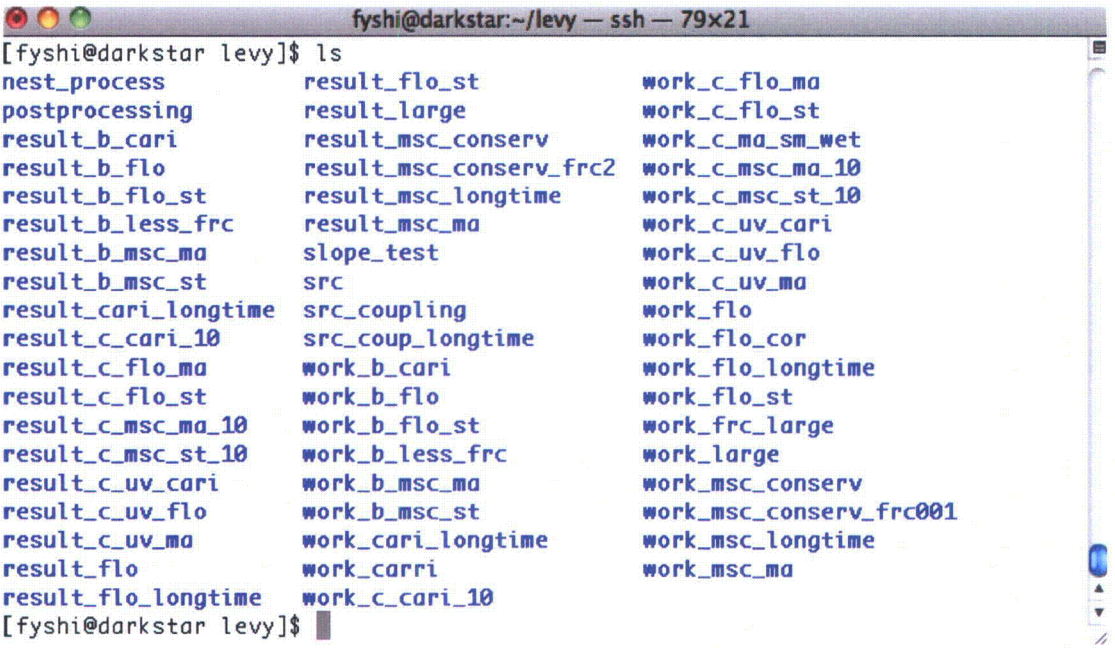

Figure F-1. The structure of FUNWAVE-TVD program folders on DARKSTAR

ATTACHMENT G Backup of NHWAVE Directory Listing (Purpose G)

Figure G-1. The structure of NHWAVE program folders on DARKSTAR

Attachment H. Program for Calculation of Earthquake Magnitude (Purpose H)

Program H-1

```
clear all

L=750.0*1000;
W=50.0*1000;
mu=4.0*10^10; % pa
u0=23.0;
M0=mu*u0*L*W
Mw=(log10(M0)-9)/log10(32)
```

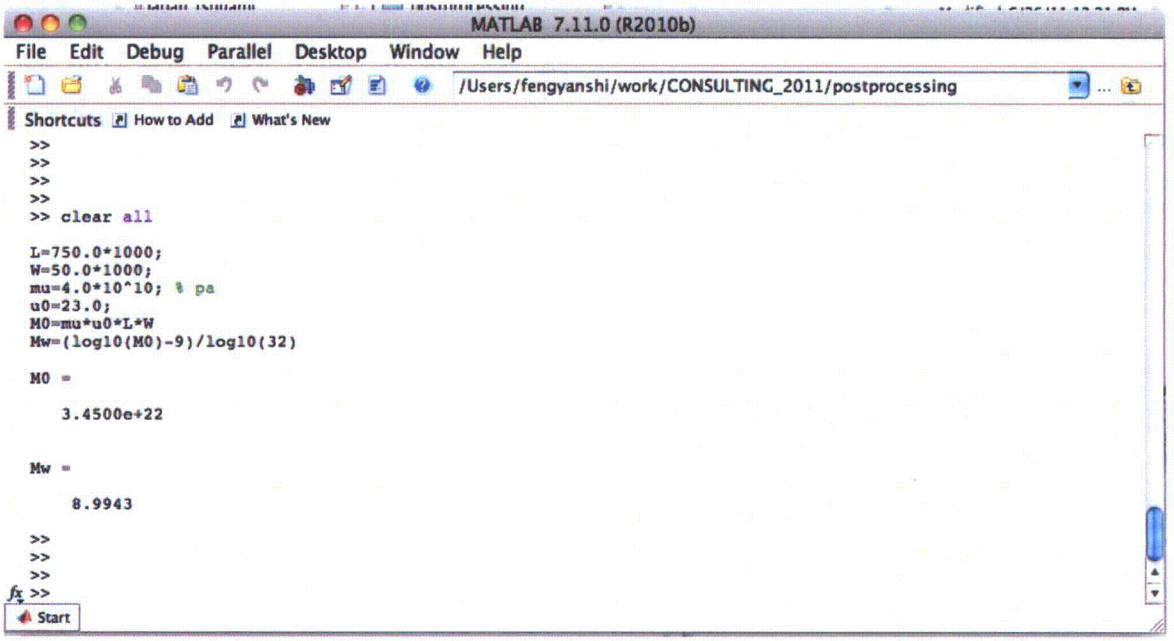

Figure H-1. Validation of earthquake magnitude using parameters in Reference 13.

Attachment I: Verification of Post Processing of FUNWAVE-TVD and NHWAVE (Purpose I)

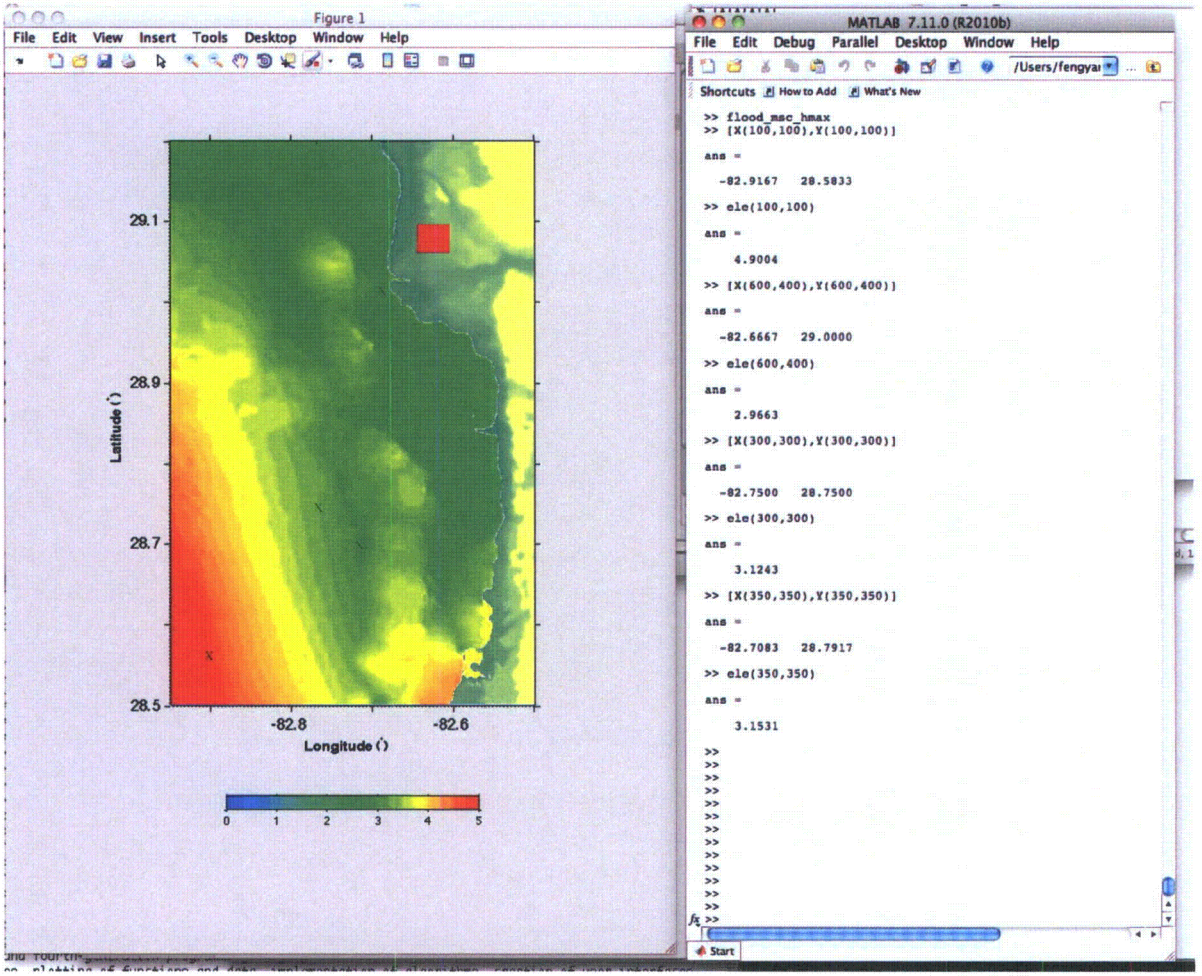

Figure I-1. Comparisons between modeled maximum magnitudes at four locations (marked in the left panel) and approximate values color-scaled in the left figure.

Contents of this DVD are listed as follows.

1. FUNWAVE-TVD Version 1.0

1) Directory listing
DVD/FUNWAVE_TVD
result_b_cari
result_b_flo
result_b_flo_st
result_b_msc_ma
result_b_msc_st
result_c_cari_10
result_c_flo_ma
result_c_flo_st
result_c_msc_ma_10
result_c_msc_st_10
result_flo
result_flo_st
result_large
result_msc_conserv
result_msc_ma
src
src_coupling
work_b_cari
work_b_flo
work_b_flo_st
work_b_msc_ma
work_b_msc_st
work_c_cari_10
work_c_flo_ma
work_c_flo_st
work_c_msc_ma_10
work_c_msc_st_10
work_carri (not used)
work_flo
work_flo_st
work_large
work_msc_conserv
work_msc_ma

2) Users' manual
DVD/Manuals/funwave_tvd_1.0_manual.pdf

3) Computer programs

DVD/FUNWAVE_TVD/src/* - source files (*.F) and executive file (mytvd, compiled without nesting option)

DVD/FUNWAVE_TVD/src_coupling/* - source files (*.F) and executive file (mytvd, compiled with nesting option)

4) Input files

DVD/FUNWAVE_TVD/work_large/input.txt - for Venezuela case, grid A

DVD/FUNWAVE_TVD/work_b_cari/input.txt - for Venezuela case, grid B

DVD/FUNWAVE_TVD/work_c_cari_10/input.txt - for Venezuela case, grid C

DVD/FUNWAVE_TVD/work_msc_ma/input.txt - for MC case, grid A, NHWAVE source

DVD/FUNWAVE_TVD/work_b_msc_ma/input.txt - for MC case, grid B, NHWAVE source

DVD/FUNWAVE_TVD/work_c_msc_ma_10/input.txt - for MC case, grid C, NHWAVE source

DVD/FUNWAVE_TVD/work_msc_conserv/input.txt - for MC case, grid A, static source

DVD/FUNWAVE_TVD/work_b_msc_st/input.txt - for MC case, grid B, static source

DVD/FUNWAVE_TVD/work_c_msc_st_10/input.txt - for MC case, grid C, static source

DVD/FUNWAVE_TVD/work_flo/input.txt - for FE case, grid A, NHWAVE source

DVD/FUNWAVE_TVD/work_b_flo/input.txt - for FE case, grid B, NHWAVE source

DVD/FUNWAVE_TVD/work_c_flo_ma/input.txt - for FE case, grid C, NHWAVE source

DVD/FUNWAVE_TVD/work_flo_st/input.txt - for FE case, grid A, static source

DVD/FUNWAVE_TVD/work_b_flo_st/input.txt - for FE case, grid B, static source

DVD/FUNWAVE_TVD/work_c_flo_st/input.txt - for FE case, grid C, static source

5) Output files

DVD/FUNWAVE_TVD/result_large/* - for Venezuela case, grid A

DVD/FUNWAVE_TVD/result_b_cari/* - for Venezuela case, grid B

DVD/FUNWAVE_TVD/result_c_cari_10/* - for Venezuela case, grid C

DVD/FUNWAVE_TVD/result_msc_ma/* - for MC case, grid A, NHWAVE source

DVD/FUNWAVE_TVD/result_b_msc_ma/* - for MC case, grid B, NHWAVE source

DVD/FUNWAVE_TVD/result_c_msc_ma_10/* - for MC case, grid C, NHWAVE source

DVD/FUNWAVE_TVD/result_msc_conserv/* - for MC case, grid A, static source

DVD/FUNWAVE_TVD/result_b_msc_st/* - for MC case, grid B, static source

DVD/FUNWAVE_TVD/result_c_msc_st_10/* - for MC case, grid C, static source

DVD/FUNWAVE_TVD/result_flo/* - for FE case, grid A, NHWAVE source

DVD/FUNWAVE_TVD/result_b_flo/* - for FE case, grid B, NHWAVE source

DVD/FUNWAVE_TVD/result_c_flo_ma/* - for FE case, grid C, NHWAVE source

DVD/FUNWAVE_TVD/result_flo_st/* - for FE case, grid A, static source

DVD/FUNWAVE_TVD/result_b_flo_st/* - for FE case, grid B, static source
DVD/FUNWAVE_TVD/result_c_flo_st/* - for FE case, grid C, static source

NHWAVE Version 1.0

1) Directory listing

DVD/NHWAVE/LandSlide_FLO
DVD/NHWAVE/LandSlide_MIS

2) NHWAVE documentation

DVD/Manuals/nhwave_documentation.pdf

3) Computer programs
DVD/NHWAVE/LandSlide_FLO/*.f90 – for FE case
DVD/NHWAVE/LandSlide_MIS/*.f90 – for MC case

4) Input files
DVD/NHWAVE/LandSlide_FLO/input.txt - for FE case
DVD/NHWAVE/LandSlide_MIS/input.txt - for MC case

5) Output files
DVD/NHWAVE/LandSlide_FLO/*_0048, *_0055 - for FE case
DVD/NHWAVE/LandSlide_MIS/*_0016 - for MC case

Bathymetric/Topographic Data Files

DVD/Bathy_Topo/fyshi001-4161.asc, GRID C from CRM
DVD/Bathy_Topo/fyshi001-9328.asc, GRID B from CRM
DVD/Bathy_Topo/fyshi001-1718.asc, GRIDA$_1$ from ETOPO 1
DVD/Bathy_Topo/fyshi001-4652.asc, GRIDA from ETOPO 1

Grid Generation (Programs C-1 through C-7)

DVD/Grid_generation/mkbathy_a_2m.f, Program C-1
DVD/Grid_generation/mkbathy_a_1m.f, Program C-2
DVD/Grid_generation/mkbathy_b_15s.f, Program C-3
DVD/Grid_generation/mksite_3s.f, Program C-4
DVD/Grid_generation/intp.m, Program C-5
DVD/Grid_generation/nav_convert.m, Program C-6
DVD/Grid_generation/process_site.m, Program C-7

Post Processing MATLAB Programs

DVD/Postprocessing/

- gulf_b_hmax_flo_ma.m
- cal_max_runup_cari.m
- gulf_b_hmax_flo_st.m
- cari_eta.m
- cari_gulf_eta.m
- gulf_b_hmax_ma.m
- cari_gulf_hmax.m
- gulf_b_hmax_st.m
- depth_b.m
- gulf_eta.m
- depth_gulf.m
- gulf_eta_conserv.m
- depth_large.m
- gulf_eta_flo.m
- gulf_eta_flo_st.m
- gulf_eta_ma.m
- gulf_hmax.m
- gulf_hmax_conserv.m
- gulf_hmax_flo_st.m
- gulf_hmax_ma.m
- flood_cari.m
- max_runup_1d_cari.m
- flood_cari_hmax.m
- max_runup_1d_cari_miles.m
- flood_flo_hmax.m
- max_runup_1d_flo.m
- flood_flo_hmax_st.m
- max_runup_1d_flo_miles.m
- flood_flo_ma.m
- max_runup_1d_flo_st.m
- flood_flo_st.m
- max_runup_1d_flo_st_miles.m
- flood_msc_hmax.m
- max_runup_1d_msc.m
- flood_msc_hmax_st.m
- max_runup_1d_msc_miles.m
- flood_msc_ma.m
- max_runup_1d_msc_st.m
- flood_msc_st.m
- max_runup_1d_msc_st_miles.m
- gulf_b_eta_cari.m
- max_runup_contour_cari.m
- gulf_b_eta_flo_ma.m

max_runup_contour_flo.m
gulf_b_eta_flo_st.m
max_runup_contour_flo_st.m
max_runup_contour_msc.m
max_runup_contour_msc_st.m
gulf_b_eta_ma.m
msc_ma_grid_a.m

Shock-capturing Non-hydrostatic Model for Fully Dispersive Surface Wave Processes

Gangfeng Ma[1], Fengyan Shi[1] and James T. Kirby[1]

[1] *Center for Applied Coastal Research, University of Delaware, Newark DE 19716 USA*

Abstract

This paper describes NHWAVE, a shock-capturing non-hydrostatic model for simulating wave refraction, diffraction, shoaling, breaking and landslide-generated tsunami in finite water depth. The governing equations are the incompressible Navier-Stokes equations in conservative form, written in surface and terrain-following form using a σ coordinate. In order to apply a Godunov-type scheme, the velocities are defined at cell centers. The dynamic pressure is defined at vertically facing cell faces so that the pressure boundary condition at free surface can be precisely imposed. The HLL Riemann approximation is employed to estimate fluxes at horizontal cell faces. The nonlinear strong stability-preserving (SSP) Runge-Kutta scheme is used to obtain second-order accuracy in time. The model is validated using six test cases based on analytical solutions and experimental data. The computed results show that the model can well predict wave propagation over submerged bar, wave refraction and diffraction over an elliptical shoal as well as wave breaking and run-up on a plane beach using relatively few (three to five) vertical layers. The model is also shown to provide accurate reproduc-

tion of a generated tsunami wave resulting from a solid translating bottom feature.

Keywords: non-hydrostatic model, shock-capturing, Godunov-type scheme, nearshore wave processes, landslide generated tsunami

1. Introduction

Wave propagation from deep water to coastal region is subject to wave refraction, diffraction, shoaling and breaking. Accurate prediction of these phenomena is crucial to studying nearshore hydrodynamics and solute transport in the coastal area. Boussinesq-type wave models with improved nonlinearity and dispersion characteristics provide an efficient and well-tested tool for the simulation of wave propagation, especially in shallow water regions (Madsen and Sørensen, 1992; Nwogu, 1993; Wei et al., 1995). Means for extending these models to higher order in dispersion have been developed (see Gobbi et al (2000), Lynett and Liu (2002) and Agnon et al (1999), for example), and more recently, extensions to the model formulation to account for turbulent structure of the flow field and the resulting effects on depth-averaged solute or contaminant transport have been developed (Kim et al, 2009; Kim and Lynett, 2011). All of these extensions lead to a great deal of complexity in the resulting model equations.

An alternative approach is to solve the Navier-Stokes equations directly with proper free surface tracking techniques, such as the marker-and-cell (MAC) method (Harlow and Welch, 1965), the volume-of-fluid (VOF) method

(Hirt and Nichols, 1981) and the level-set method (Osher and Sethian, 1988). These approaches have wide applications on the simulations of wave shoaling and breaking in the surf zone; see, for example, Lin and Liu (1998a,b), Watanabe et al. (2005), Christensen (2006), Shi et al. (2010) and Ma et al. (2011). The main drawbacks of these types of models are: (1) They are computationally expensive, making applications to large-scale domains infeasible; (2) The free surface normally crosses the computational cell arbitrarily, which brings the difficulty of applying the pressure boundary condition precisely on the free surface and may eventually affect the accuracy of velocity computation (Lin and Li, 2002); and (3) The grid resolution in the surf zone and swash zone, where the water depth is relatively shallow, is usually poor due to the use of Cartesian grid system on most of applications.

A direct simplification of the above-mentioned approach is to assume that the free surface elevation is a single value function of the horizontal coordinates. By doing so, the free surface is always located at the upper computational boundary and can be determined by applying the free surface boundary conditions. It is computationally more efficient with the lack of free surface tracking. The pressure boundary condition at the free surface can be accurately prescribed with some proper treatments. This simplification leads to a new set of non-hydrostatic models, which are not only suitable for modeling short wave propagation but also for the simulation of turbulence and solute transport in the surf zone. To solve the non-hydrostatic equations, the pressure is decomposed into hydrostatic and non-hydrostatic

components. The governing equations can be discretized by finite difference method (Casulli and Stelling, 1998; Casulli, 1999; Namin et al., 2001; Casulli and Zanolli, 2002; Lin and Li, 2002; Chen, 2003; Stelling and Zijlema, 2003; Zijlema and Stelling, 2005; Yuan and Wu, 2004; Lee et al., 2006; Young et al., 2007, 2009, 2010; Wu et al., 2010), finite element method (Walters, 2005) and finite volume method (Bradford, 2005; Ai and Jin, 2010; Lai et al., 2010). A major concern addressed in recent developments of non-hydrostatic models is the accurate prediction of wave dispersion characteristics with relatively few vertical grid points. It has been recognized that $10 \sim 20$ vertical layers are normally required to describe wave dispersion up to an acceptable level with some simple treatments of pressure boundary conditions at the top layer, for example, Casulli and Stelling (1998), Casulli (1999), Casulli and Zanolli (2002), Li and Fleming (2001), Namin et al. (2001), Lin and Li (2002), Chen (2003). To address this issue, Stelling and Zijlema (2003) proposed the Keller-box method to replace the staggered grid in the vertical direction, which enables the pressure to be located at the cell faces rather than the cell centers. The pressure boundary condition at the free surface can be exactly assigned to zero without any approximation. Yuan and Wu (2004a,b) proposed an integral method to remove the top-layer hydrostatic assumption using a staggered grid framework. Young and Wu (2010) used the Boussinesq-type-like equations with the reference velocity to provide an analytical-based non-hydrostatic pressure distribution at the top layer. All of these methods significantly reduce the errors in dynamic pressure estima-

tion and allow for use of a very small number of vertical layers for accurate simulation of dispersive waves.

It is non-trivial to apply non-hydrostatic models to the simulation of breaking waves in the surf zone and wave run-up in the swash region, because the numerical scheme involved must treat shock propagation adequately in order to model broken waves (Zijlema and Stelling, 2008). Shock-capturing schemes based on Godunov-type approach, which can deal with discontinuous flow, are well-suited for breaking wave simulations. These schemes are able to track actual location of wave breaking without requiring any criterion that tells the model when and where the wave breaking happens. An application of this approach to simulation of breaking waves in the surf zone was given by Bradford (2011). It was showed that the non-hydrostatic model with Godunov-type scheme can predict wave height distribution, turbulence and undertow under breaking waves at least as accurate as the VOF model. However, eight or more vertical layers are needed in his model to accurately predict the surface elevation around the outer surf zone as well as velocity profiles within the surf zone.

In this paper, we describe a new nonhydrostatic model (called NHWAVE, for Non Hydrostatic WAVE model) based on a Godunov-type scheme. NHWAVE solves the incompressible Navier-Stokes equations in terrain and surface-following σ coordinates. Bottom movement is included in order to simulate tsunami generation by three-dimensional underwater landslides. To apply Godunov-type scheme, the velocities are defined at cell centers. The dy-

namic pressure is defined at vertically-facing cell faces as in the Keller-box method, allowing the pressure boundary condition at the free surface to be precisely imposed. The hydrostatic equations are solved by a well-balanced finite volume method. The fluxes at cell faces are estimated by HLL Riemann approximation. To obtain second-order temporal accuracy, the nonlinear Strong Stability-Preserving (SSP) Runge-Kutta scheme (Gottlieb et al., 2001) is adopted for adaptive time stepping.

The paper is organized as follows. In section 2, the governing equations in conservative form are presented. The numerical method, boundary conditions and wetting-drying scheme are introduced in section 3. Finally, six test cases are given in section 4 to show the model's capability of simulating wave refraction, diffraction, shoaling, breaking and landslide tsunami generation.

2. Governing Equations

The incompressible Navier-Stokes equations in Cartesian coordinates (x_1^*, x_2^*, x_3^*), where $x_1^* = x^*$, $x_2^* = y^*$ and $x_3^* = z^*$ and time t^* are given by

$$\frac{\partial u_i}{\partial x_i^*} = 0 \tag{1}$$

$$\frac{\partial u_i}{\partial t^*} + u_j \frac{\partial u_i}{\partial x_j^*} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i^*} + g_i + \frac{\partial \tau_{ij}}{\partial x_j^*} \tag{2}$$

where $(i, j) = 1, 2, 3$, u_i is velocity component in the x_i^* direction, p is total pressure, ρ is water density, $g_i = -g\delta_{i3}$ is the gravitational body force and $\tau_{ij} = \nu(\partial u_i/\partial x_j^* + \partial u_j/\partial x_i^*)$ is viscous stress with ν the kinematic molecular

viscosity.

In order to accurately represent bottom and surface geometry, a σ-coordinate developed by Phillips (1957) is adopted in this study.

$$t = t^* \quad x = x^* \quad y = y^* \quad \sigma = \frac{z^* + h}{D} \tag{3}$$

where $D(x, y, t) = h(x, y, t) + \eta(x, y, t)$, h is water depth, η is surface elevation. This coordinate transformation basically maps the varying vertical coordinate in the physical domain to a uniform transformed space where σ spans from 0 to 1. Using the principle of chain differentiation, the partial differentiation of a variable $f = f(x^*, y^*, z^*, t^*)$ in the physical domain is transformed as follows.

$$\begin{aligned}
\frac{\partial f}{\partial t^*} &= \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial t^*} \\
\frac{\partial f}{\partial x^*} &= \frac{\partial f}{\partial x} + \frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} \\
\frac{\partial f}{\partial y^*} &= \frac{\partial f}{\partial y} + \frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} \\
\frac{\partial f}{\partial z^*} &= \frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial z^*}
\end{aligned} \tag{4}$$

Plugging equation (4) into (1) and (2), we obtain the governing equations in the new coordinate (x, y, σ) and time t.

$$\frac{\partial D}{\partial t} + \frac{\partial Du}{\partial x} + \frac{\partial Dv}{\partial y} + \frac{\partial \omega}{\partial \sigma} = 0 \tag{5}$$

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial \sigma} = \mathbf{S}_h + \mathbf{S}_p + \mathbf{S}_\tau \tag{6}$$

where $\mathbf{U} = (Du, Dv, Dw)^T$. The fluxes are

$$\mathbf{F} = \begin{pmatrix} Duu + \frac{1}{2}gD^2 \\ Duv \\ Duw \end{pmatrix} \quad \mathbf{G} = \begin{pmatrix} Duv \\ Dvv + \frac{1}{2}gD^2 \\ Dvw \end{pmatrix} \quad \mathbf{H} = \begin{pmatrix} u\omega \\ v\omega \\ w\omega \end{pmatrix}$$

The source terms are given by

$$\mathbf{S}_h = \begin{pmatrix} gD\frac{\partial h}{\partial x} \\ gD\frac{\partial h}{\partial y} \\ 0 \end{pmatrix} \quad \mathbf{S}_p = \begin{pmatrix} -\frac{D}{\rho}(\frac{\partial p}{\partial x} + \frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial x^*}) \\ -\frac{D}{\rho}(\frac{\partial p}{\partial y} + \frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial y^*}) \\ -\frac{1}{\rho}\frac{\partial p}{\partial \sigma} \end{pmatrix} \quad \mathbf{S}_\tau = \begin{pmatrix} DS_{\tau_x} \\ DS_{\tau_y} \\ DS_{\tau_z} \end{pmatrix}$$

where the total pressure has been divided into two parts: dynamic pressure p (use p as dynamic pressure hereinafter for simplicity) and hydrostatic pressure $\rho g(\eta - z)$. ω is the vertical velocity in the σ coordinate image domain, given by

$$\omega = D\left(\frac{\partial \sigma}{\partial t^*} + u\frac{\partial \sigma}{\partial x^*} + v\frac{\partial \sigma}{\partial y^*} + w\frac{\partial \sigma}{\partial z^*}\right) \tag{7}$$

with

$$\begin{aligned}
\frac{\partial \sigma}{\partial t^*} &= \frac{1}{D}\frac{\partial h}{\partial t} - \frac{\sigma}{D}\frac{\partial D}{\partial t} \\
\frac{\partial \sigma}{\partial x^*} &= \frac{1}{D}\frac{\partial h}{\partial x} - \frac{\sigma}{D}\frac{\partial D}{\partial x} \\
\frac{\partial \sigma}{\partial y^*} &= \frac{1}{D}\frac{\partial h}{\partial y} - \frac{\sigma}{D}\frac{\partial D}{\partial y} \\
\frac{\partial \sigma}{\partial z^*} &= \frac{1}{D}
\end{aligned} \tag{8}$$

The diffusion terms are given by

$$\begin{aligned}
S_{\tau_x} &= \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xx}}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xy}}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} + \frac{\partial \tau_{xz}}{\partial \sigma}\frac{\partial \sigma}{\partial z^*} \\
S_{\tau_y} &= \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yy}}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} + \frac{\partial \tau_{yz}}{\partial \sigma}\frac{\partial \sigma}{\partial z^*} \\
S_{\tau_z} &= \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{zx}}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} + \frac{\partial \tau_{zy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} + \frac{\partial \tau_{zz}}{\partial \sigma}\frac{\partial \sigma}{\partial z^*}
\end{aligned} \tag{9}$$

Integrating equation (5) from $\sigma = 0$ to 1 and using the boundary conditions at the bottom and surface for ω, we get the governing equation for free surface movement.

$$\frac{\partial D}{\partial t} + \frac{\partial}{\partial x}(D\int_0^1 u d\sigma) + \frac{\partial}{\partial y}(D\int_0^1 v d\sigma) = 0 \tag{10}$$

3. Numerical Method

A combined finite-volume and finite-difference scheme with a Godunov-type method was applied to discretize equations (6) and (10). It is straightforward to define all dependent variables at cell centers to solve Riemann

problem. However, this treatment results in checkerboard solutions in which the pressure and velocity become decoupled when they are defined at the same location (Patankar, 1980). Therefore, most existing models use a staggered grid in which the pressure is defined at the centers of computational cells and the velocities are defined at cell faces (Bradford, 2005). However, staggered grids do not lend themselves as easily as co-located grids to the use of Godunov-type schemes. Meanwhile, difficulty in treating the cell-centered pressure at the top layer may arise when applying the pressure boundary condition at the free surface (Yuan and Wu, 2004).

With these considerations, a different kind of staggered grid framework is introduced, in which the velocities are placed at the cell centers and the pressure is defined at the vertically-facing cell faces as shown in figure 1. The momentum equations are solved by a second-order Godunov-type finite volume method. The HLL approximate Riemann solver (Harten et al., 1983) is used to estimate fluxes at the cell faces. As in Stelling and Zijlema (2003), the pressure boundary condition at the free surface can be precisely assigned to zero.

3.1. Time Stepping

To obtain second-order temporal accuracy, the two-stage second-order nonlinear Strong Stability-Preserving (SSP) Runge-Kutta scheme (Gottlieb et al., 2001) was adopted for time stepping. At the first stage, an intermediate quantity $\mathbf{U}^{(1)}$ is evaluated using a typical first-order, two-step projection

method given by

$$\frac{\mathbf{U}^* - \mathbf{U}^n}{\Delta t} = -\left(\frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial \sigma}\right)^n + \mathbf{S}_h^n + \mathbf{S}_\tau^n \tag{11}$$

$$\frac{\mathbf{U}^{(1)} - \mathbf{U}^*}{\Delta t} = \mathbf{S}_p^{(1)} \tag{12}$$

where $\mathbf{U}^n$ represents $\mathbf{U}$ value at time level n, $\mathbf{U}^*$ is the interim value in the two-step projection method, and $\mathbf{U}^{(1)}$ is the final first stage estimate. In the second stage, the velocity field is again updated to a second intermediate level using the same projection method, after which the Runge-Kutta algorithm is used to obtain a final value of the solution at the $n+1$ time level.

$$\frac{\mathbf{U}^* - \mathbf{U}^{(1)}}{\Delta t} = -\left(\frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial \sigma}\right)^{(1)} + \mathbf{S}_h^{(1)} + \mathbf{S}_\tau^{(1)} \tag{13}$$

$$\frac{\mathbf{U}^{(2)} - \mathbf{U}^*}{\Delta t} = \mathbf{S}_p^{(2)} \tag{14}$$

$$\mathbf{U}^{n+1} = \frac{1}{2}\mathbf{U}^n + \frac{1}{2}\mathbf{U}^{(2)} \tag{15}$$

Each stage of the calculation requires the specification of the nonhydrostatic component of the pressure force as expressed through the quantities $\mathbf{S}_p$. The pressure field needed to specify these is based on the solution of the Poisson equation described below. Also at each stage, the surface elevation is obtained by solving equation (10) explicitly. The time step Δt is adaptive during the simulation, following the Courant-Friedrichs-Lewy (CFL) crite-

rion

$$\Delta t = C\min\left[\min\frac{\Delta x}{|u_{i,j,k}| + \sqrt{gD_{i,j}}}, \min\frac{\Delta y}{|v_{i,j,k}| + \sqrt{gD_{i,j}}}, \min\frac{\Delta\sigma D_{i,j}}{|w_{i,j,k}|}\right] \quad (16)$$

where C is the Courant number, which is taken to be 0.5 to ensure accuracy and stability in the current model.

3.2. Spatial Finite Volume Scheme

Equation (11) and (13) essentially represent the shallow water equations based on a hydrostatic pressure distribution. We discretize them using a second-order Godunov-type finite volume method. It is noticed that applying a naive finite volume Godunov-type scheme directly to the equation does not lead to an automatic preservation of steady state (Zhou et al., 2001; Kim et al, 2008; Liang and Marche, 2009). Therefore, It is desirable to reformulate the equation so that the flux and source terms can be automatically balanced at the discrete level in the steady state. In this study, the method by Liang and Marche (2009) is employed. Taking the x component source term as an example, notice that the total water depth is $D = h + \eta$. The source term can be rewritten as

$$g(h+\eta)\frac{\partial h}{\partial x} = \frac{\partial}{\partial x}\left(\frac{1}{2}gh^2\right) + g\eta\frac{\partial h}{\partial x} \quad (17)$$

in which the first term in the right hand side can be combined together with the flux terms.

Based on this, the flux and source terms may be expressed as

$$\mathbf{F} = \begin{pmatrix} Duu + \frac{1}{2}g\eta^2 + gh\eta \\ Duv \\ Duw \end{pmatrix} \quad \mathbf{G} = \begin{pmatrix} Duv \\ Dvv + \frac{1}{2}g\eta^2 + gh\eta \\ Dvw \end{pmatrix} \quad \mathbf{S}_h = \begin{pmatrix} g\eta\frac{\partial h}{\partial x} \\ g\eta\frac{\partial h}{\partial y} \\ 0 \end{pmatrix}$$

The main advantages of the above formulation are that they maintain the hyperbolicity of the original formulation and are mathematically balanced for the flux and source terms so that the steady state motionless water surface is automatically preserved for wet-bed applications (Liang and Marche, 2009).

To solve equation (11) and (13), fluxes based on the conservative variables are required at the cell faces. In high-order Godunov-type methods, the values of the conservative variables within a cell are calculated using a reconstruction method based on the cell center data (Zhou et al., 2001). Usually a piecewise linear reconstruction is used, leading to a second order scheme. For $\mathbf{U}$ in the cell i, we have

$$\mathbf{U} = \mathbf{U}_i + (x - x_i)\Delta\mathbf{U}_i \tag{18}$$

where $\Delta\mathbf{U}_i$ is the gradient of U, which is calculated by

$$\Delta\mathbf{U}_i = \text{avg}\left(\frac{\mathbf{U}_{i+1} - \mathbf{U}_i}{x_{i+1} - x_i}, \frac{\mathbf{U}_i - \mathbf{U}_{i-1}}{x_i - x_{i-1}}\right) \tag{19}$$

in which avg is a slope limiter which is used to avoid spurious oscillations in

the reconstruction data at the cell faces. In this study, the van Leer limiter is adopted, which is given by

$$\text{avg}(a,b) = \frac{a|b| + |a|b}{|a| + |b|} \tag{20}$$

The left and right values of $\mathbf{U}$ at cell face $(i + \frac{1}{2})$ are given by

$$\mathbf{U}^L_{i+\frac{1}{2}} = \mathbf{U}_i + \frac{1}{2}\Delta x_i \Delta \mathbf{U}_i \quad \mathbf{U}^R_{i+\frac{1}{2}} = \mathbf{U}_{i+1} - \frac{1}{2}\Delta x_{i+1} \Delta \mathbf{U}_{i+1} \tag{21}$$

The flux $\mathbf{F}(\mathbf{U}^L, \mathbf{U}^R)$ is calculated by solving a local Riemann problem at each horizontally-facing cell face. In the present study, HLL Riemann solver is employed. The flux at the cell interface $(i + \frac{1}{2})$ is determined by

$$\mathbf{F}(\mathbf{U}^L, \mathbf{U}^R) = \begin{cases} \mathbf{F}(\mathbf{U}^L) & \text{if} \quad s_L \geq 0 \\ \mathbf{F}^*(\mathbf{U}^L, \mathbf{U}^R) & \text{if} \quad s_L < 0 < s_R \\ \mathbf{F}(\mathbf{U}^R) & \text{if} \quad s_R \leq 0 \end{cases} \tag{22}$$

where

$$\mathbf{F}^*(\mathbf{U}^L, \mathbf{U}^R) = \frac{s_R \mathbf{F}(\mathbf{U}^L) - s_L \mathbf{F}(\mathbf{U}^R) + s_L s_R (\mathbf{U}^R - \mathbf{U}^L)}{s_R - s_L} \tag{23}$$

with wave speed s_L and s_R defined by

$$s_L = \min(u^L - \sqrt{gD_L}, u_s - \sqrt{gD_s}) \tag{24}$$

$$s_R = \max(u^R + \sqrt{gD_R}, u_s + \sqrt{gD_s}) \tag{25}$$

where u_s and $\sqrt{gD_s}$ are estimated by

$$u_s = \frac{1}{2}(u^L + u^R) + \sqrt{gD_L} - \sqrt{gD_R} \tag{26}$$

$$\sqrt{gD_s} = \frac{\sqrt{gD_L} + \sqrt{gD_R}}{2} + \frac{u^L - u^R}{4} \tag{27}$$

To obtain the non-hydrostatic velocity field, the dynamic pressure p has to be calculated first. From equation (12) and (14), we get

$$u^{(k)} = u^* - \frac{\Delta t}{\rho}\left(\frac{\partial p}{\partial x} + \frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial x^*}\right)^{(k)} \tag{28}$$

$$v^{(k)} = v^* - \frac{\Delta t}{\rho}\left(\frac{\partial p}{\partial y} + \frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial y^*}\right)^{(k)} \tag{29}$$

$$w^{(k)} = w^* - \frac{\Delta t}{\rho}\frac{1}{D^{(k)}}\frac{\partial p^{(k)}}{\partial \sigma} \tag{30}$$

where $k = 1, 2$ represents the kth stage in the Runge-Kutta integration.

Applying equation (3) and (4), the continuity equation (1) is transformed as

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} + \frac{\partial v}{\partial y} + \frac{\partial v}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} + \frac{1}{D}\frac{\partial w}{\partial \sigma} = 0 \tag{31}$$

Substituting equation (28) $\sim$ (30) into (31), we obtain the Poisson equa-

tion in (x, y, σ) coordinate system

$$\frac{\partial}{\partial x}\left[\frac{\partial p}{\partial x}+\frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial x^*}\right]+\frac{\partial}{\partial y}\left[\frac{\partial p}{\partial y}+\frac{\partial p}{\partial \sigma}\frac{\partial \sigma}{\partial y^*}\right]+\frac{\partial}{\partial \sigma}\left(\frac{\partial p}{\partial x}\right)\frac{\partial \sigma}{\partial x^*}+$$
$$\frac{\partial}{\partial \sigma}\left(\frac{\partial p}{\partial y}\right)\frac{\partial \sigma}{\partial y^*}+\left[\left(\frac{\partial \sigma}{\partial x^*}\right)^2+\left(\frac{\partial \sigma}{\partial y^*}\right)^2+\frac{1}{D^2}\right]\frac{\partial}{\partial \sigma}\left(\frac{\partial p}{\partial \sigma}\right)= \tag{32}$$
$$\frac{\rho}{\Delta t}\left(\frac{\partial u^*}{\partial x}+\frac{\partial u^*}{\partial \sigma}\frac{\partial \sigma}{\partial x^*}+\frac{\partial v^*}{\partial y}+\frac{\partial v^*}{\partial \sigma}\frac{\partial \sigma}{\partial y^*}+\frac{1}{D}\frac{\partial w^*}{\partial \sigma}\right)$$

The above equation is discretized with the second-order space-centered finite difference method. The velocities of (u^*, v^*, w^*) at vertical cell faces are interpolated from adjacent cell-centered values. The resulting linear equation is given by

$$a_1 p_{i,j-1,k-1}+a_2 p_{i-1,j,k-1}+a_3 p_{i,j,k-1}+a_4 p_{i+1,j,k-1}+a_5 p_{i,j+1,k-1}+$$
$$a_6 p_{i,j-1,k}+a_7 p_{i-1,j,k}+a_8 p_{i,j,k}+a_9 p_{i+1,j,k}+a_{10} p_{i,j+1,k}+a_{11} p_{i,j-1,k+1}+ \tag{33}$$
$$a_{12} p_{i-1,j,k+1}+a_{13} p_{i,j,k+1}+a_{14} p_{i+1,j,k+1}+a_{15} p_{i,j+1,k+1}=R_p$$

where

$$a_1=-\left(\frac{(\sigma_y)_{i,j-1,k}}{2\Delta y(\Delta\sigma_k+\Delta\sigma_{k-1})}+\frac{(\sigma_y)_{i,j,k}}{2\Delta y(\Delta\sigma_k+\Delta\sigma_{k-1})}\right)$$

$$a_2=-\left(\frac{(\sigma_x)_{i-1,j,k}}{2\Delta x(\Delta\sigma_k+\Delta\sigma_{k-1})}+\frac{(\sigma_x)_{i,j,k}}{2\Delta x(\Delta\sigma_k+\Delta\sigma_{k-1})}\right)$$

$$a_3=-\frac{(\sigma_x^2+\sigma_y^2+\frac{1}{D^2})_{i,j,k}}{0.5(\Delta\sigma_k+\Delta\sigma_{k-1})\Delta\sigma_{k-1}}$$

$$a_4=\frac{(\sigma_x)_{i+1,j,k}}{2\Delta x(\Delta\sigma_k+\Delta\sigma_{k-1})}+\frac{(\sigma_x)_{i,j,k}}{2\Delta x(\Delta\sigma_k+\Delta\sigma_{k-1})}$$

$$a_5 = \frac{(\sigma_y)_{i,j+1,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})} + \frac{(\sigma_y)_{i,j,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})}$$

$$a_6 = a_{10} = -\frac{1}{\Delta y^2} \quad a_7 = a_9 = -\frac{1}{\Delta x^2}$$

$$a_8 = \frac{2}{\Delta x^2} + \frac{2}{\Delta y^2} + \frac{(\sigma_x^2 + \sigma_y^2 + \frac{1}{D^2})_{i,j,k}}{0.5(\Delta\sigma_k + \Delta\sigma_{k-1})\Delta\sigma_k} + \frac{(\sigma_x^2 + \sigma_y^2 + \frac{1}{D^2})_{i,j,k}}{0.5(\Delta\sigma_k + \Delta\sigma_{k-1})\Delta\sigma_{k-1}}$$

$$a_{11} = \frac{(\sigma_y)_{i,j-1,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})} + \frac{(\sigma_y)_{i,j,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})}$$

$$a_{12} = \frac{(\sigma_x)_{i-1,j,k}}{2\Delta x(\Delta\sigma_k + \Delta\sigma_{k-1})} + \frac{(\sigma_x)_{i,j,k}}{2\Delta x(\Delta\sigma_k + \Delta\sigma_{k-1})}$$

$$a_{13} = -\frac{(\sigma_x^2 + \sigma_y^2 + \frac{1}{D^2})_{i,j,k}}{0.5(\Delta\sigma_k + \Delta\sigma_{k-1})\Delta\sigma_k}$$

$$a_{14} = -\left(\frac{(\sigma_x)_{i+1,j,k}}{2\Delta x(\Delta\sigma_k + \Delta\sigma_{k-1})} + \frac{(\sigma_x)_{i,j,k}}{2\Delta x(\Delta\sigma_k + \Delta\sigma_{k-1})}\right)$$

$$a_{15} = -\left(\frac{(\sigma_y)_{i,j+1,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})} + \frac{(\sigma_y)_{i,j,k}}{2\Delta y(\Delta\sigma_k + \Delta\sigma_{k-1})}\right)$$

$$R_p = -\frac{\rho}{\Delta t}\left(\frac{\partial u^*}{\partial x} + \frac{\partial u^*}{\partial \sigma}\frac{\partial \sigma}{\partial x^*} + \frac{\partial v^*}{\partial y} + \frac{\partial v^*}{\partial \sigma}\frac{\partial \sigma}{\partial y^*} + \frac{1}{D}\frac{\partial w^*}{\partial \sigma}\right)$$

Uniform gridding is used in the horizontal direction while gridding in the vertical direction is generalized to be non-uniform in order to capture the bottom and surface boundary layers when desired. The coefficient matrix is asymmetric and has a total of 15 diagonal lines. The linear system is solved using the preconditioned GMRES scheme. With p solved, the non-hydrostatic velocities at each stage can be updated from equation (28) to (30).

3.3. Boundary Conditions

To solve the governing equations, boundary conditions are required for all the physical boundaries. At the free surface, the continuity of normal and tangential stresses is enforced. With wind effects absent, the tangential stress equals zero, resulting in

$$\frac{\partial u}{\partial \sigma}|_{z=\eta} = \frac{\partial v}{\partial \sigma}|_{z=\eta} = 0 \tag{34}$$

The vertical velocity w at the ghost cells is obtained to ensure that w at free surface satisfies the kinematic boundary condition.

$$w|_{z=\eta} = \frac{\partial \eta}{\partial t} + u\frac{\partial \eta}{\partial x} + v\frac{\partial \eta}{\partial y} \tag{35}$$

The zero pressure condition on the free surface is applied when the Poisson equation is solved.

$$p|_{z=\eta} = 0 \tag{36}$$

At the bottom, the normal velocity and the tangential stress are prescribed. The normal velocity w is imposed through the kinematic boundary condition.

$$w|_{z=-h} = -\frac{\partial h}{\partial t} - u\frac{\partial h}{\partial x} - v\frac{\partial h}{\partial y} \tag{37}$$

The free-slip boundary conditions are used to estimate horizontal velocities at the ghost cells.

$$\frac{\partial u}{\partial \sigma}|_{z=-h} = \frac{\partial v}{\partial \sigma}|_{z=-h} = 0 \tag{38}$$

The Neumann boundary condition is used for dynamic pressure.

$$\frac{\partial p}{\partial \sigma}|_{z=-h} = -\rho D \frac{dw}{dt}|_{z=-h} \tag{39}$$

where w at $z = -h$ is given by (37). In the application to an underwater landslide in section 4.6 below, we linearize the resulting boundary condition which gives

$$\frac{\partial p}{\partial \sigma}|_{z=-h} = \rho D \frac{\partial^2 h}{\partial t^2} \tag{40}$$

At the closed boundaries or vertical walls, free-slip boundary conditions are imposed, so that the normal velocity and the tangential stress are set to zero. The normal pressure gradient is zero. At inflow, both free surface and velocities calculated from the analytical solutions are specified.

3.4. Wetting-Drying Treatment

It is straightforward to use a wetting-drying scheme for modeling moving boundaries. In the present study, a simple wetting-drying scheme is implemented. The wet and dry cells are judged by total water depth D. If a cell has the total water depth D greater than D_{min}, it's a wet cell with $\text{Mask}_{i,j} = 1$. Otherwise it's a dry cell with $\text{Mask}_{i,j} = 0$. D_{min} is the minimum water depth allowed for computation. The surface elevation in the dry cells are defined as $\eta_{i,j} = D_{min} - h_{i,j}$. For a dry cell surrounded by wet cells, $\text{Mask}_{i,j}$ has to

reevaluated as

$$\begin{aligned} \text{Mask}_{i,j} &= 1 \quad \text{if} \quad \eta_{i,j} \leq \eta_{neighbor} \\ \text{Mask}_{i,j} &= 0 \quad \text{if} \quad \eta_{i,j} > \eta_{neighbor} \end{aligned} \tag{41}$$

In the dry cells, the normal flux at cell faces are set to zero. The wave speed of equation (24) and (25) for a dry bed are modified as (Zhou et al., 2001)

$$s_L = u^L - \sqrt{gD_L} \quad s_R = u^L + 2\sqrt{gD_L} \quad \text{(right dry bed)} \tag{42}$$

$$s_L = u^R - 2\sqrt{gD_R} \quad s_R = u^R + \sqrt{gD_R} \quad \text{(left dry bed)} \tag{43}$$

4. Numerical Results

The numerical method presented in the above section has been tested with several analytical solutions and laboratory experiments. Six test cases are given in this section: (1) standing wave in closed basin; (2) solitary wave propagation in constant depth; (3) periodic wave over submerged bar; (4) wave transformation over an elliptical shoal on a sloped bottom; (5) breaking solitary wave runup; (6) tsunami generation by three-dimensional rigid underwater landslides. These test cases have been widely used to validate non-hydrostatic models.

4.1. Standing wave in closed basin

The numerical model was first tested by the analytical solution of a standing wave in closed basin with length of $L = 20\ m$ and depth of $D = 10\ m$. This test case has been studied by various researchers, for example, Casulli and Stelling (1998), Casulli (1999) and Chen (2003). They used more than 20 vertical layers in order to correctly simulate wave dispersion. With Keller-box scheme, Stelling and Zijlema (2003) obtained good agreement with analytical solution by using two layers. These models are all solved in Cartesian coordinate system. In this section, we have shown that the present model in σ coordinates can reproduce accurate waves with relatively few vertical layers.

The initial surface elevation is given by

$$\eta = a\cos(\frac{\pi x}{10}) \tag{44}$$

where η is the surface elevation in meters, $a = 0.1\ m$ is the amplitude of the standing wave. The wave length equals the length of the basin. Since $kD = \pi > 1$, the wave is highly dispersive. From the dispersion relationship $\sigma^2 = gk\tanh(kD)$, where $\sigma = 2\pi/T$, $k = 2\pi/L$, we can calculate wave period $T = 3.59\ s$. The analytical solution for this standing wave is

$$\eta = a\cos(kx)\cos(\sigma t) \tag{45}$$

For the numerical setup, a uniform grid spacing of $0.2\ m$ was used in

the horizontal direction. In the vertical direction, five layers are employed. Figure 2 shows the comparisons of numerical and analytical surface elevations at $x = 0.1\ m$ and $x = 17.5\ m$. The computed surface elevations agree very well with the analytical solution. The amplitude of the wave has no significant change. Figure 3 gives the numerical errors at $x = 17.5\ m$ as a function of vertical layers and wave dispersion parameter kD, which is obtained by varying water depth D. The numerical error is defined as $error = \frac{1}{H}\sqrt{\frac{1}{N}\sum_{j=1}^{N}(\eta_a - \eta_j)^2}$, where N is the number of data that are compared, η_a is the analytical solution, H is the wave height at $x = 17.5\ m$. The simulation time is 30 s. As expected, the numerical errors are decreased by increasing the number of vertical layers. The numerical results are less accurate for higher dispersive wave with the same vertical layers. Therefore, more vertical layers are required to obtain sufficiently accurate predictions for highly dispersive waves.

4.2. Solitary wave propagation in constant depth

The second test case is the solitary wave propagation in constant water depth. The computational domain is two-dimensional with length of 200 m and water depth of 1.0 m. The solitary wave is initially at $x = 8.0\ m$. An exact solution of the fully nonlinear equations, obtained using Tanaka's (1986) method, was used to specify the initial surface elevation and velocity field ($t = 0\ s$ at figure 4 and 5). The solitary wave has the wave height to the still water depth ratio $H/h = 0.5$, indicating that the solitary wave is highly

nonlinear. The domain is discretized by a uniform grid in the horizontal direction with $\Delta x = 0.1\ m$ and three layers in the vertical direction. The time step Δt is adjusted during the simulation based on the Courant number, which is taken as 0.5.

Figure 4 shows the comparisons of simulated surface elevations and Tanaka solutions at $t = 10, 20, 40\ s$. We can see that the agreements are almost perfect. As the wave propagates to the right, the wave shape hardly change. Figure 5 shows the comparisons of horizontal and vertical velocities at the middle elevation between numerical results and Tanaka solutions. The good agreement of vertical velocity indicates that the dynamic pressure has been well simulated by the model with three vertical layers.

4.3. Periodic wave over submerged bar (Beji and Battjes, 1993)

In this section, the model is applied to simulate wave shoaling over a submerged bar. The experimental data by Beji and Battjes (1993) is used to validate our non-hydrostatic model. This case has been used to verify a number of non-hydrostatic free surface models including Casulli (1999), Lin and Li (2002), Chen (2003), Stelling and Zijlema (2003), Yuan and Wu (2004) and Bradford (2005). The data has also frequently been used as a test of Boussinesq models, as the case falls outside the range of typical $O(\mu^2)$ models such as Wei et al (1995), but is handled by various higher order approaches such as Gobbi and Kirby (1999) or Lynett and Liu (2002).

The model setup and bottom geometry is shown in figure 6. The wave

flume has a length of 30 m. The still water depth is 0.4 m, which is reduced to 0.1 m at the bar. The offshore slope of the bar is 1/20 and the onshore slope is 1/10. Periodic waves with period 2.02 s and amplitude 1.0 cm are incident at the left boundary. The computational domain is 35 m long with 10 m of sponge layer at the right end. To discretize the computational domain, 1750 constant horizontal grids and three vertical layers are used to ensure that the free higher harmonics can be properly calculated.

Figure 7 shows the comparisons of free surface elevation at six measurement locations between numerical results and experimental data. Wave shoaling at station a and propagation over the bar at station b are well simulated by the model. The bound higher harmonics generated by the nonlinear shoaling wave on the upward slope of the bar become free on the downward slope, resulting in irregular wave pattern at station c to f. The model generally predicts free surface evolution at these stations well, indicating that the dispersion for higher frequency components is well simulated with three vertical layers.

4.4. Wave transformation over an elliptical shoal on a sloped bottom

This example is to test the model's capability of simulating wave refraction and diffraction over a 3D uneven bottom. The corresponding experiment was conducted by Berkhoff et al. (1982). The model setup and bottom geometry is shown in figure 8. An elliptical shoal is located on a plane beach with a slope of 1/50. Let (x', y') be the slope-oriented coordinates, which are

related to (x, y) coordinate system by means of rotation over -20^o. The still water depth without shoal is given by

$$\begin{aligned} h &= 0.45 \quad x' < -5.84 \\ h &= max(0.07, 0.45 - 0.02(5.84 + x')) \quad x' \geq -5.84 \end{aligned} \tag{46}$$

Since the minimum water depth is 0.07 m, the wave is non-breaking. The boundary of the shoal is given by

$$\left(\frac{x'}{3}\right)^2 + \left(\frac{y'}{4}\right)^2 = 1 \tag{47}$$

where the thickness of the shoal is

$$d = -0.3 + 0.5\sqrt{1 - \left(\frac{x'}{3.75}\right)^2 - \left(\frac{y'}{5}\right)^2} \tag{48}$$

Regular wave with wave period of 1.0 s and wave height of 4.64 cm are incident at the left boundary $x = -12$ m. At the right end, waves are completely absorbed by a sponge layer with $L = 5$ m. Two walls are located at $y = -10$ m and 10 m, where free-slip boundary conditions are imposed.

To well simulate wave refraction and diffraction, a fine grid with $\Delta x = 0.025$ m and $\Delta y = 0.05$ m is used. Five vertical layers are used in the vertical direction. The time step is adjusted during the simulation, with courant number 0.5. The simulation period is 30 s. The final five waves are employed to estimate wave height. Figure 9 shows the comparisons of wave

height between numerical results and experiment data at eight measurement sections. Due to refraction, wave focussing occurs behind the shoal with a maximum wave height of approximately 2.2 times the incident wave height (around $x = 5\ m$, $y = 0\ m$). The model slightly under-predicts the peak wave height at section 3 and 5. In other sections, the numerical results and experiment data agree quite well. These results demonstrates that wave refraction and diffraction can be well simulated by the model.

4.5. Breaking solitary wave run-up

To show the model's capability of simulating breaking waves and wetting-drying, we applied the model to study breaking solitary wave run-up and run-down in a slope beach. The corresponding laboratory experiment was conducted by Synolakis (1987). The beach slope is 1/20. The still water depth varies from 0.21 m to 0.29 m. A solitary wave which has a wave height to still water depth ratio of 0.28 was incident on the left. Wave gauges were used to record the free surface displacement during the run-up and run-down.

In the numerical simulation, the solitary wave is initially at 1.5 m on the left from the toe of the beach. An exact solution of the fully nonlinear equations, obtained using Tanaka's (1986) method, was used to specify the initial surface elevation and velocity field, see figure 10. The computational domain extends to a location beyond the maximum run-up point. The entire domain is discretized by 550 uniform grid in the horizontal with $\Delta x = 0.02$ m. Three layers are used in the vertical direction. The minimum water

depth is 5 mm, which determines wetting-and-drying of the computational cells. A Smagorinsky subgrid model is employed to dissipate wave energy during wave breaking, which is given by

$$\nu_t = (C_s\Delta)^2\sqrt{2S_{ij}S_{ij}} \tag{49}$$

where C_s is the Smagorinsky coefficient, which is taken as 0.1, Δ is the filter width, which is calculated as $\Delta = (\Delta x \Delta y \Delta \sigma D)^{1/3}$, and $S_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j^*} + \frac{\partial u_j}{\partial x_i^*})$ is the stress tensor.

The numerical results were compared with the experimental data after normalization. The length scale is normalized by the still water depth d and the time scale is normalized by $\sqrt{g/d}$. Figure 11 shows comparisons of simulated and measured free surface profile during wave shoaling, breaking, run-up and run-down. Panels (a) and (b) show the shoaling process of the solitary wave. The wave becomes more asymmetric and the wave height increases as water depth decreases. Around $t\sqrt{g/d} = 20$, the wave starts to break as shown in panel (c), the surface profile is dramatically changed and the wave height is rapidly reduced. The wave continuously breaks as its turbulent front moves towards the shoreline. After the wave front passes the still-water shoreline, it collapses and the consequent run-up process commences. The run-up process is shown in the panel (d) and (e). After reaching the maximum run-up point, the front starts to run-down which is shown in the panel (f). The comparisons between the simulation and experiment data

are fairly good. The shoaling, breaking, run-up and run-down processes of the solitary wave are well reproduced. In figure 11, we also shown the numerical results without subgrid turbulence model. During wave shoaling and breaking, the predicted surface elevation is seldom impacted by neglecting turbulence. This result is consistent with the findings by Bradford (2011) and Zijlema and Stelling (2008). However, the model without turbulence model slightly overpredicts wave runup as shown in panel (e) and (f).

4.6. Tsunami generation by three-dimensional underwater landslides

Submarine landslides are one of the most dangerous mechanisms for tsunami generation in the coastal areas. In this section, we applied the model to simulate tsunami generation by an idealized three-dimensional underwater landslides. Experiments have recently been performed by Enet and Grilli (2007) in a 3.7 m wide, 1.8 m deep and 30 m long wave tank with a plane underwater slope with $\theta = 15^o$ angle. This data set has also been used recently by Fuhrman and Madsen (2009) to test the accuracy of a higher-order Boussinesq model.

The vertical cross section of the landslide is shown in figure 12. The geometry is defined using truncated hyperbolic secant functions.

$$\zeta = \frac{T}{1-\epsilon}[\mathrm{sech}(k_b x)\mathrm{sech}(k_w y) - \epsilon] \tag{50}$$

where $k_b = 2C/b$, $k_w = 2C/w$ and $C = a\cosh(1/\epsilon)$. The landslide has length $b = 0.395\ m$, width $w = 0.680\ m$ and thickness $T = 0.082\ m$. The truncation

parameter $\epsilon = 0.717$. The landslide is initially located at the submergence depth d. The movement of the landslide is prescribed as

$$s(t) = s_0 \ln(\cosh \frac{t}{t_0}) \tag{51}$$

with

$$s_0 = \frac{u_t^2}{a_0}, \quad t_0 = \frac{u_t}{a_0} \tag{52}$$

where u_t and a_0 are the landslide terminal velocity and initial acceleration, respectively. To represent the landslide, the horizontal domain is discretized by a uniform grid with $\Delta x = 0.02\ m$ and $\Delta y = 0.02\ m$. Three vertical layers are employed in the simulation. The landslide parameters are $u_t = 1.70\ m/s$ and $a_0 = 1.12\ m/s^2$.

Three wave gauges are located at (x, y) locations $(1469, 350), (1929, 0)$ and $(1929, 500)$, where all distances are in mm and where x denotes distance from the still water shoreline and y denotes distances off the centerline axis of the sliding mass. Model results are presented as time series in comparison to measured data at each of the three gages, with two representative tests chosen. Figure 13 shows model/data comparisons for the case of an initial submergence of the landslide center of $d = 61$ mm. The model is seen to represent the amplitude and the phase structure of the generated wave train well. As would be expected, wave heights are highest at the gage lying along the axis of the landslide motion and drop off with distance away from the centerline axis. Figure 13 also displays the results of a hydrostatic model

simulation, which are obtained by neglecting the pressure correction steps indicated in the equation (12) and (14). These results are markedly different from the non-hydrostatic model results, indicating the great importance of dispersion in this test. The hydrostatic result basically consists of a strong drawdown of the water column immediately behind the sliding mass. This drawdown first grows in magnitude and then decreases as the relative depth of submergence becomes larger. In contrast, the nonhydrostatic model result consists of a packet of dispersive waves which lag behind the relatively faster moving slide as the slide accelerates. This behavior is further illustrated in snapshots of the generated wave trains at times $t = 1$, 2 and 3 s shown in figure 14 and 15, with nonhydrostatic behavior shown in the sequence of panels in figure 14 and hydrostatic behavior shown in the sequence in figure 15. The absence of dispersion in the generated waves in figure 15 is clear, emphasizing the importance of frequency dispersion in the present example. We note that the results of Fuhrman and Madsen (2009), obtained using a higher-order Boussinesq model, showed comparable capabilities in predicting wave phase structure as the nonhydrostatic model here, but tended to overpredict crest and trough heights to some degree in comparison to the results here; see their figure 14.

Figure 16 illustrates similar results for the case of an initial depth of submergence of $d = 120$ mm. Generated wave heights are lower here than in the previous case due to the greater depth of submergence, but the pattern of sea surface response is qualitatively similar to the previous case. Hydrostatic

results for this case have not been computed as the problem is in relatively deeper water than the previous case.

5. Conclusions

In this paper, NHWAVE, a shock-capturing non-hydrostatic model for nonlinear free-surface wave processes is presented. The governing equations are solved in a σ coordinate system and discretized by a combined finite volume and finite difference scheme with a Godunov-type method. In order to apply Godunov-type scheme, the velocities are collocated at the cell center. The dynamic pressure is defined at the vertical cell faces, which ensure that the pressure boundary condition at the free surface can be precisely prescribed. The HLL approximate Riemann solver was employed to estimate fluxes at horizontal cell faces. The second-order nonlinear Strong Stability-Preserving (SSP) Runge-Kutta scheme were adopted for adaptive time stepping.

The model was tested using six benchmarks based on analytical solutions or experimental data. Unlike some other non-hydrostatic models which need $10 \sim 20$ vertical layers to well simulate short waves well, the present model can achieve good predictions of wave shoaling, wave dispersion, refraction and diffraction using $3 \sim 5$ vertical layers. With the Godunov-type scheme, which is suitable for solving discontinuous solutions, the model can be used to study complex problems such as wave breaking in the surf zone. For the landslide generated tsunami, the model predicts the evolution of surface

elevation generally well.

Acknowledgements This work was supported by the Office of Naval Research (Coastal Geosciences Program), the National Science Foundation (Physical Oceanography) and the Delaware Sea Grant College Program.

References

Agnon, Y., Madsen, P. A. and Schäffer, H. A., 1999, A new approach to high order Boussinesq models, *J. Fluid Mech.*, 399, 319-333.

Ai C. and Jin S., 2010, Non-hydrostatic finite volume method for non-linear waves interacting with structures, *Comp. Fluids*, 39, 2090-2100

Beji S. and Battjes J.A., 1993, Experimental investigation of wave propagation over a bar, *Coastal Engineering*, 19, 151-162

Berkhoff J.C.W., Booy N. and Radder A.C., 1982, Verification of numerical wave propagation models for simple harmonic linear water waves, *Coastal Engineering*, 6, 255-279

Bradford S.F., 2005, Godunov-based model for nonhydrostatic wave dynamics, *J. Waterway, Port, Coastal and Ocean Engineering*, 131, 226-238

Bradford S.F., 2011, Nonhydrostatic model for surf zone simulation, *J. Waterway, Port, Coastal and Ocean Engineering*, doi:10.1061/(ASCE)WW.1943-5460.0000079

Casulli V. and Stelling G.S., 1998, Numerical simulation of 3D quasi-hydrostatic free-surface flows, *J. Hydr. Engnrg.*, 124, 678-686

Casulli V., 1999, A semi-implicit finite difference method for non-hydrostatic, free-surface flow, *Int. J. Numer. Meth. Fluids*, 30, 425-440

Casulli V. and Zanolli P., 2002, Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems, *Math. Comp. Modeling*, 36, 1131-1149

Chen X., 2003, A fully hydrodynamic model for three-dimensional, free-surface flows, *Int. J. Numer. Meth. Fluids*, 42, 929-952

Christensen E.D., 2006, Large eddy simulation of spilling and plunging breakers, *Coastal Engineering*, 53, 463-485

Enet F. and Grilli S.T., 2007, Experimental study of tsunami generation by three-dimensional rigid underwater landslides, *J. Waterway, Port, Coast. Ocean Engrng.*, 133, 442-454.

Fuhrman, D. R. and Madsen, P. A., 2009, "Tsunami generation, propagation and run-up with a high-order Boussinesq model", *Coastal Engrng.*, 56, 747-758.

Gobbi, M. F. and Kirby, J. T., 1999, Wave evolution over submerged sills: tests of a high-order Boussinesq model, *Coastal Engrng.*, 37, 57-96.

Gobbi, M. F., Kirby, J. T. and Wei, G., 2000, A fully nonlinear Boussinesq model for surface waves. II. Extension to $O(kh^4)$, *J. Fluid Mech.*, 405, 181-210.

Gottlieb S., Shu C.-W. and Tadmor E., 2001, Strong stability-preserving high-order time discretization methods, *SIAM Review*, 43, 89-112

Harlow F.H. and Welch J.E., 1965, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Phys. Fluids*, 8, 2182-2189

Harten A., Lax P. and van Leer B., 1983, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Review*, 25, 35

Hirt C.W. and Nichols B.D., 1981, Volume of fluid (VOF) method for the dynamics of free boundaries, *J. Comp. Phys.*, 39, 201-225

Kim D.-H., Cho Y.-S. and Kim H.-J., 2008, Well-balanced scheme between flux and source terms for computation of shallow-water equations over irregular bathymetry, *J. Engnrng. Mech.*, 134, 277-290

Kim, D. H., Lynett, P. J. and Socolofsky, S. A., 2009, "A depth-integrated model for weakly dispersive, turbulent, and rotational fluid flows", *Ocean Modelling*, 27, 198-214.

Kim, D. H. and Lynett, P. J., 2011, Turbulent mixing and scalar transport in shallow and wavy flows, *Phys. Fluids*, 23, 016603.

Lai Z., Chen C., Cowles G.W. and Beardsley R.C., 2010, A nonhydrostatic version of FVCOM: 1. Validation experiments, *J. Geophys. Res.*, 115, doi:10.1029/2009JC005525

Lee J.W., Terbner M.D., Nixon J.B. and Gill P.M., 2006, A 3-D non-hydrostatic pressure model for small amplitude free surface flows, *Int.*

J. Numer. Meth. Fluids, 50, 649-672

Li B. and Fleming C., 2001, Three-dimensional model of Navier-Stokes equations for water waves, *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 16-25

Liang Q. and Marche F., 2009, Numerical resolution of well-balanced shallow water equations with complex source terms, *Advances in Water Resource*, 32, 873-884

Lin P. and Li C.W., 2002, A σ-coordinate three-dimensional numerical model for free surface wave propagation, *Int. J. Numer. Meth. Fluids*, 38, 1045-1068

Lin P. and Liu P.L.-F., 1998, A numerical study of breaking waves in the surf zone, *J. Fluid Mech.*, 359, 239-264

Lin P. and Liu P.L.-F., 1998, Turbulent transport, vorticity dynamics, and solute mixing under plunging breaking waves in surf zone, *J. Geophy. Res.*, 103, 15677-15694

Lynett, P. and Liu, P. L.-F., 2002, A two-layer approach to wave modelling, *Proc. Roy. Soc. A*, 460, 2637-2669.

Ma G., Shi F. and Kirby J.T., 2011, A polydisperse two-fluid model for surf zone bubble simulation, *J. Geophys. Res.*, 116, doi:10.1029/2010JC006667

Madsen P.A. and Sørensen O.R., 1992, A new form of the Boussinesq equations with improved linear dispersion characteristics. Part A slowly-varying bathymetry, *Coastal Engineering*, 18, 183-204

Namin M., Lin B. and Falconer R., 2001, An implicit numerical algorithm for solving non-hydrostatic free-surface flow problems, *Int. J. Numer. Meth. Fluids*, 35, 341-356

Nwogu O., 1993, Alternative form of Boussinesq equations for nearshore wave propagation, *Journal of Waterway, Port, Coastal and Ocean Engineering*, 119, 618-638

Osher A. and Sethian J.A., 1988, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *J. Comput. Phys.*, 79, 12-49

Patankar S.V., 1980, Numerical heat transfer and fluid flow, McGraw-Hill, New York

Phillips N.A., 1957, A coordinate system having some special advantages for numerical forecasting, *J. Meteor.*, 14, 184-185

Shi F., Kirby J.T. and Ma G., 2010, Modeling quiescent phase transport of air bubbles induced by breaking waves, *Ocean Modelling.* 35, 105-117

Stelling G. and Zijlema M., 2003, An accurate and efficient finite-difference algorithm for non-hydrostatic free-surface flow with application to wave propagation, *Int. J. Numer. Meth. Fluids*, 43, 1-23

Synolakis C.E., 1987, The runup of solitary waves, *J. Fluid Mech.*, 185, 523-545

Tanaka M., 1986, The stability of solitary waves, *Phys. Fluids*, 29, 650-655

Watanabe Y., Saeki H. and Hosking R.J., 2005, Three-dimensional vortex structures under breaking waves, *J. Fluid Mech.*, 545, 291-328

Walters R.A., 2005, A semi-implicit finite element model for non-hydrostatic (dispersive) surface waves, *Int. J. Numer. Meth. Fluids*, 49, 721-737

Wei G., Kirby J.T., Grilli S.T. and Subramanya R., 1995, A fully nonlinear Boussinesq model for surface waves. Part 1. Highly nonlinear unsteady waves, *J. Fluid Mech.*, 294, 71-92

Wu C.H., Young C.-C., Chen Q. and Lynett P.J., 2010, Efficient nonhydrostatic modeling of surface waves from deep to shallow water, *J. Waterway, Port, Coastal, and Ocean Engineering*, 136, 104-118

Young C.-C., Wu C.-H., Kuo J.-T. and Liu W.-C., 2007, A higher-order σ-coordinate non-hydrostatic model for nonlinear surface waves, *Ocean Engineering*, 34, 1357-1370

Young C.-C., Wu C.-H., Liu W.-C. and Kuo J.-T., 2009, A higher-order non-hydrostatic σ model for simulating nonlinear refraction-diffraction of water waves, *Coastal Engineering*, 56, 919-930

Young C.-C. and Wu C.-H., 2010, A σ-coordinate non-hydrostatic model with embedded Boussinesq-type-like equations for modeling deep-water waves, *Int. J. Numer. Meth. Fluids*, 63, 1448-1470

Yuan H. and Wu C.-H., 2004, A two-dimensional vertical non-hydrostatic σ model with an implicit method for free-surface flows, *Int. J. Numer. Meth. Fluids*, 44, 811-835

Yuan H. and Wu C.-H., 2004, An implicit three-dimensional fully non-hydrostatic model for free-surface flows, *Int. J. Numer. Meth. Fluids*, 46, 709-733

Zhou J.G., Gauson D.M., Mingham C.G. and Ingram D.M., 2001, The surface gradient method for the treatment of source terms in the shallow-water equations, *J. Comp. Phys.*, 168, 1-25

Zijlema M. and Stelling G.S., 2005, Further experiences with computing non-hydrostatic free-surface flows involving water waves, *Int. J. Numer. Meth. Fluids*, 48, 169-197

Zijlema M. and Stelling G.S., 2008, Efficient computation of surf zone waves using the nonlinear shallow water equations with non-hydrostatic pressure, *Coastal Engineering*, 55, 780-790

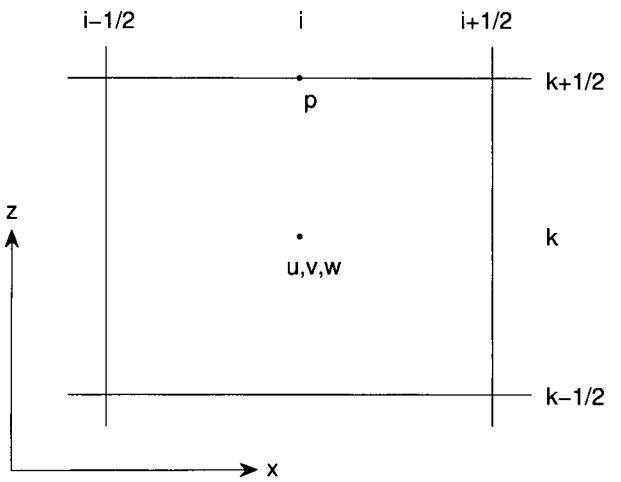

Figure 1: Layout of computational variables. Velocities (u, v, w) are placed at cell center and dynamic pressure (p) is defined at vertical cell face.

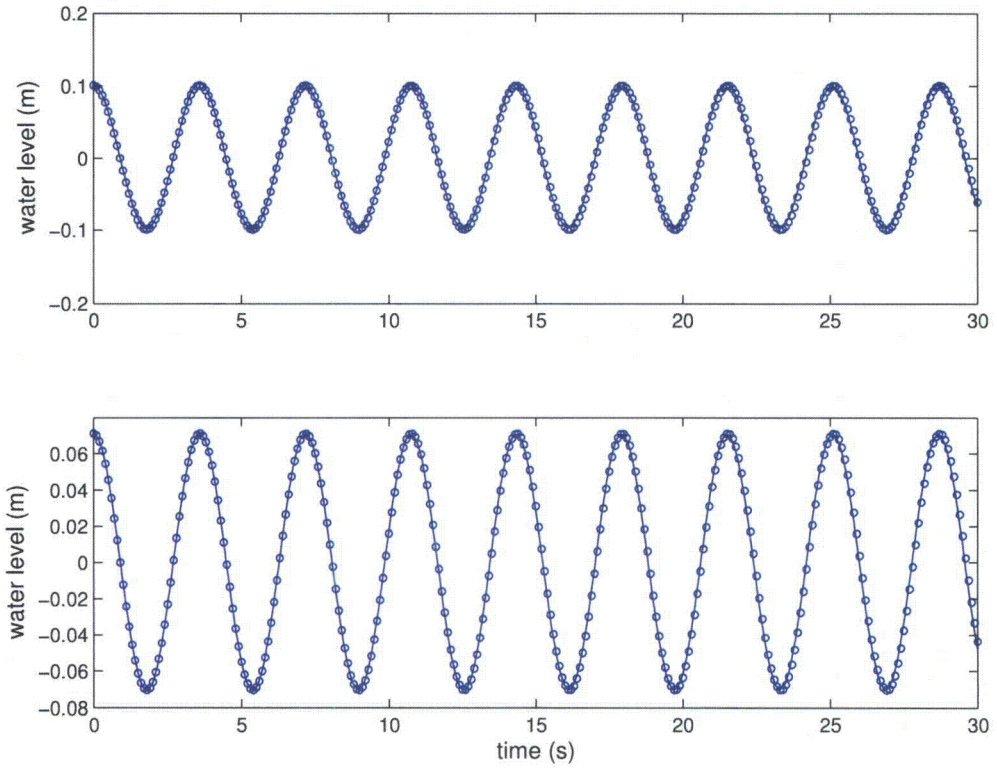

Figure 2: Comparisons between numerical (solid line) and analytical (circles) surface elevations at $x = 0.1\ m$ (upper panel) and $x = 17.5\ m$ (lower panel) for the standing wave in closed basin.

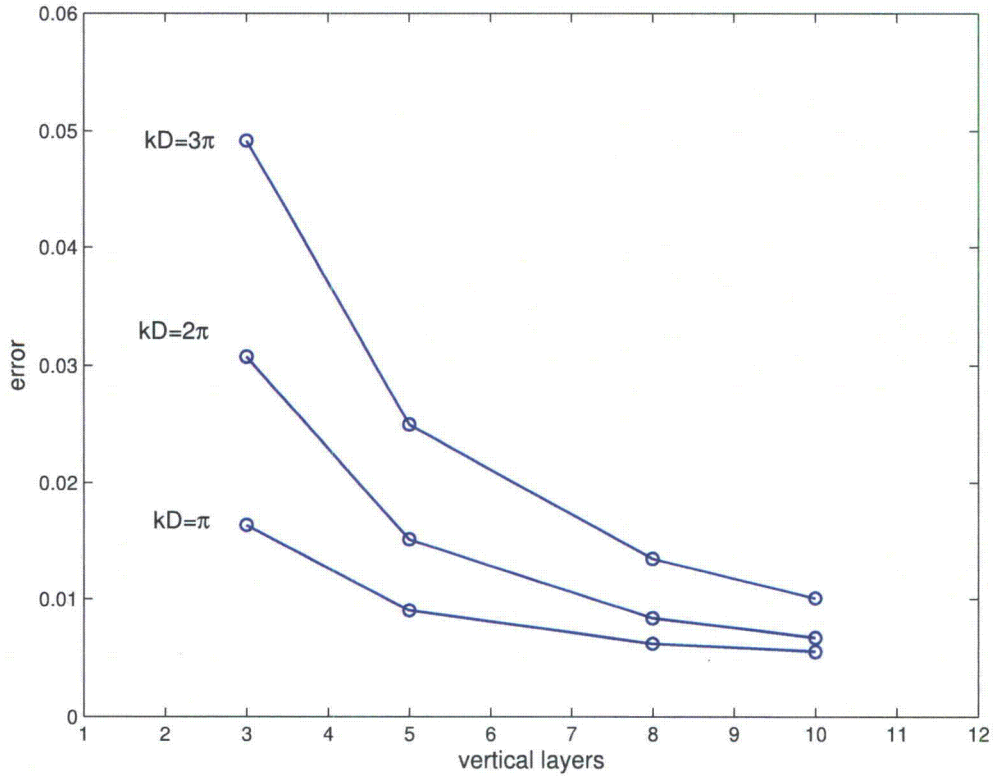

Figure 3: Numerical errors at $x = 17.5\ m$ as a function of the number of vertical layers and wave dispersion parameter kD.

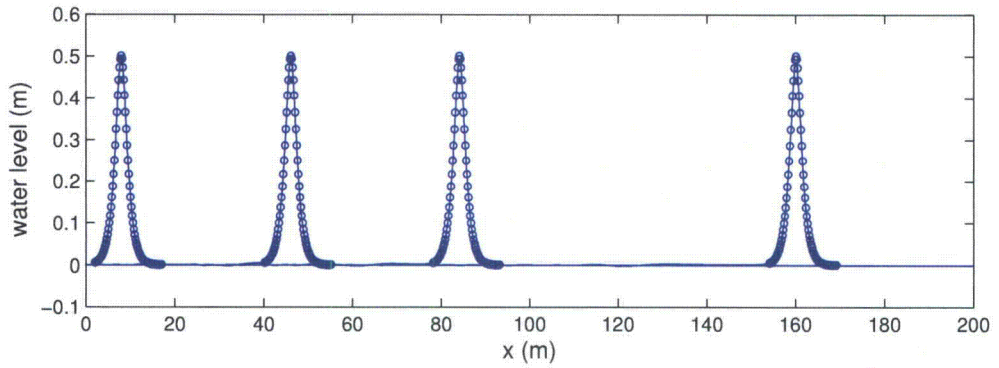

Figure 4: Comparisons between simulated surface elevations (solid line) and Tanaka solutions (circles) at $t = 0, 10, 20, 40$ s for solitary wave propagation in constant depth.

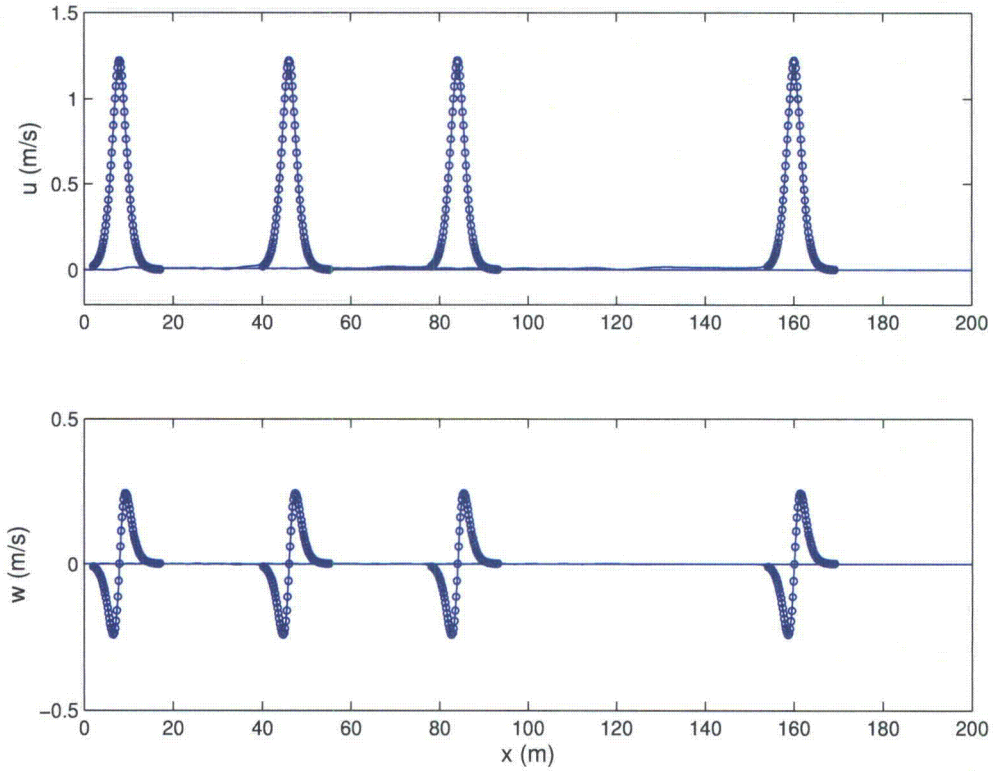

Figure 5: Comparisons between simulated velocities (solid line) and Tanaka solutions (circles) at $t = 0, 10, 20, 40$ s for solitary wave propagation in constant depth.

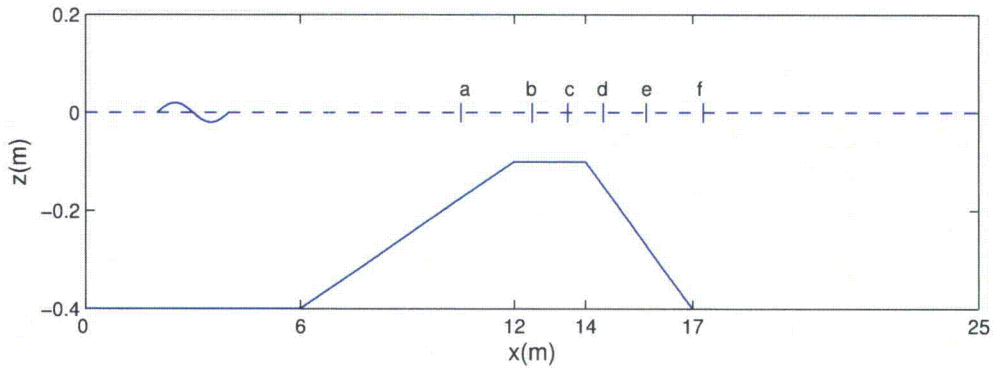

Figure 6: Bottom geometry and location of wave gauges used in the computation (a) $x = 10.5\ m$; (b) $x = 12.5\ m$; (c) $x = 13.5\ m$; (d) $x = 14.5\ m$; (e) $x = 15.7\ m$; (f) $x = 17.3\ m$.

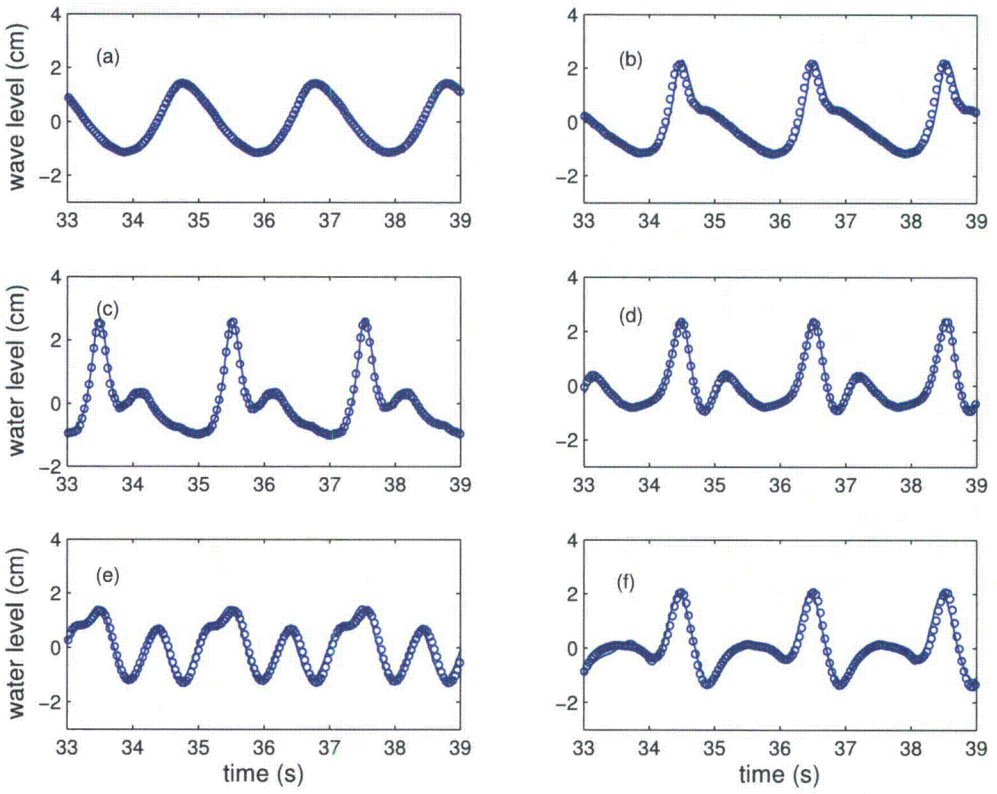

Figure 7: Comparisons between numerical (solid line) and experimental (circles) surface elevations at (a) $x = 10.5\ m$; (b) $x = 12.5\ m$; (c) $x = 13.5\ m$; (d) $x = 14.5\ m$; (e) $x = 15.7\ m$; (f) $x = 17.3\ m$.

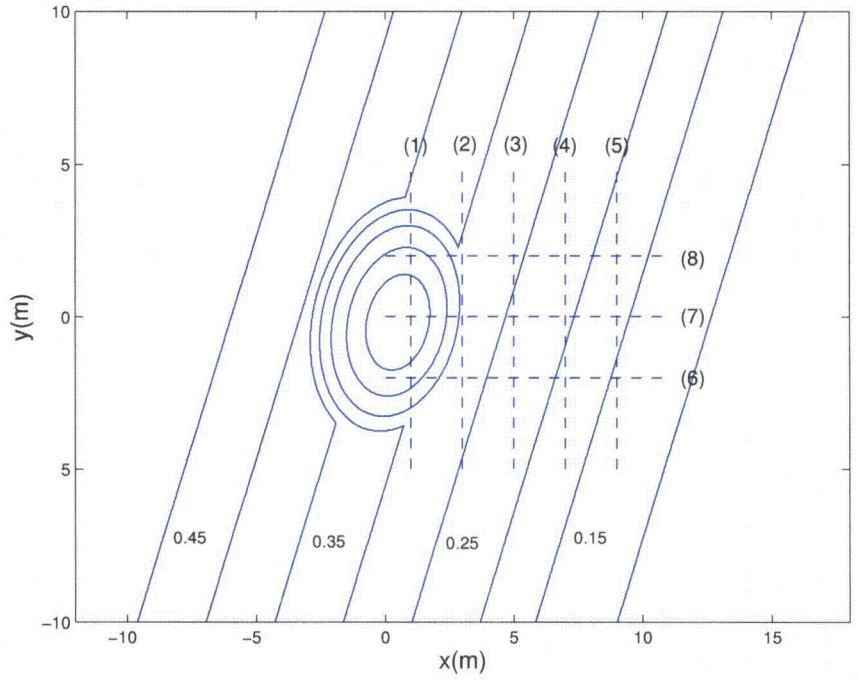

Figure 8: Bottom geometry for periodic wave propagation over an elliptical shoal, experimental setup by Berkhoff et al. (1982)

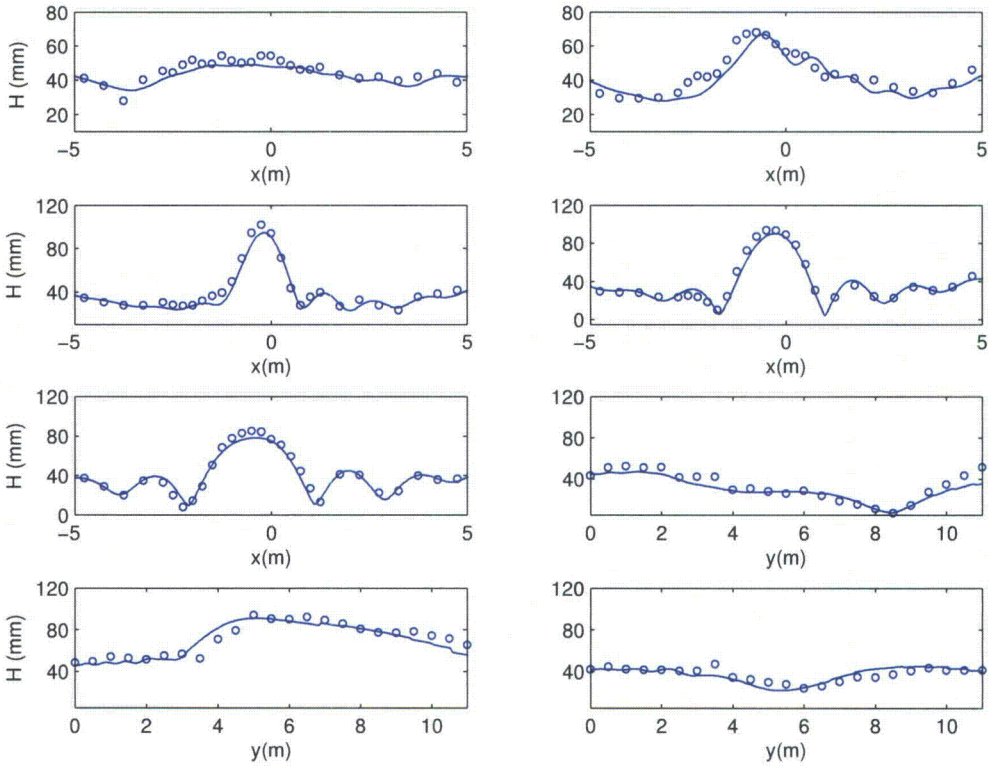

Figure 9: Comparisons between numerical (solid line) and experimental (circles) wave height at eight stations.

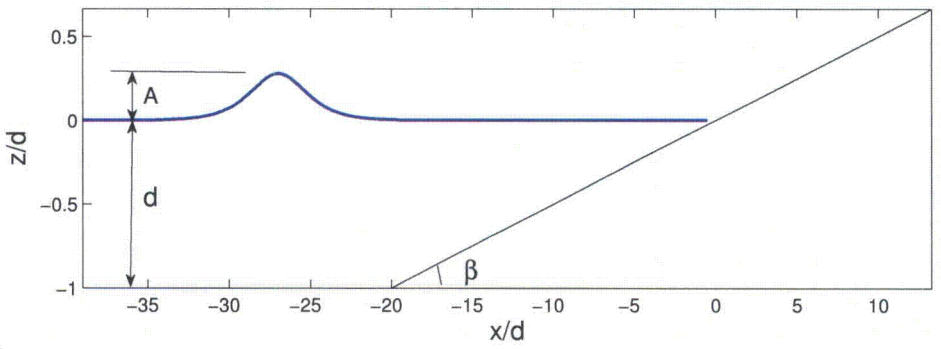

Figure 10: Computational domain and model setup. The beach slope is 1/20. The still water depth is 0.21 m. The amplitude of solitary wave is 0.0588 m.

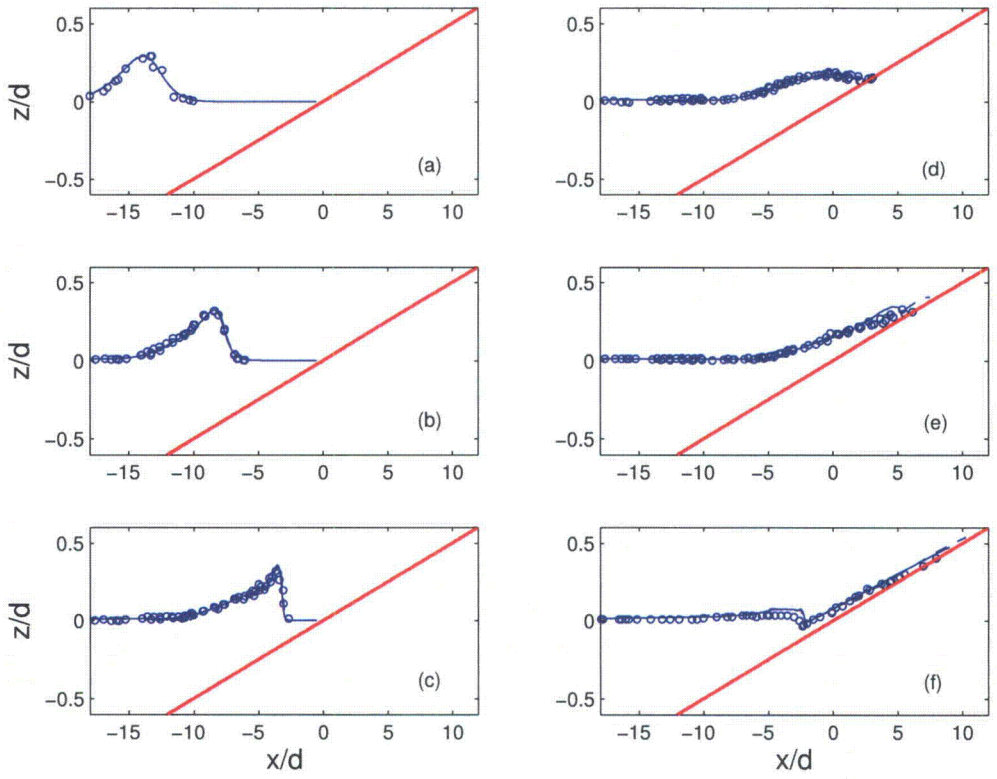

Figure 11: Comparisons between numerical (with subgrid model: solid line; without subgrid model: dashed line) and experimental (circles) free surface elevation for breaking solitary wave run-up and run-down at (a) $t\sqrt{g/d} = 10$; (b) $t\sqrt{g/d} = 15$; (c) $t\sqrt{g/d} = 20$; (d) $t\sqrt{g/d} = 25$; (e) $t\sqrt{g/d} = 30$; (f) $t\sqrt{g/d} = 50$.

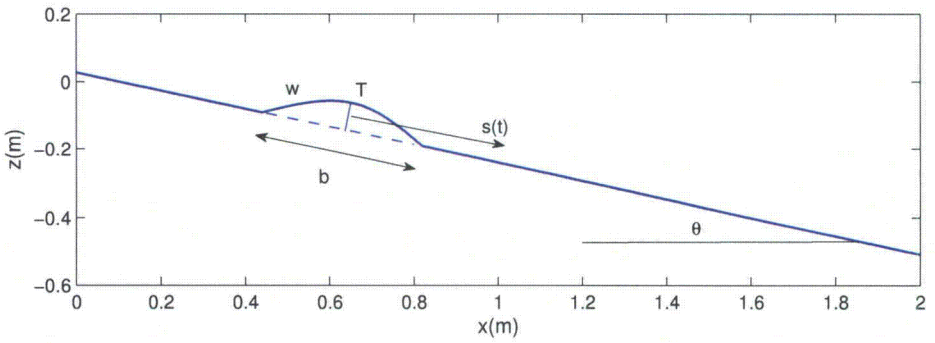

Figure 12: Vertical cross section for numerical setup of tsunami landslide. The gaussian shape landslide model has length $b = 0.395\ m$, width $w = 0.680\ m$ and thickness $T = 0.082\ m$ and is initially located at submergence depth d. The beach slope has an angle of $\theta = 15^o$.

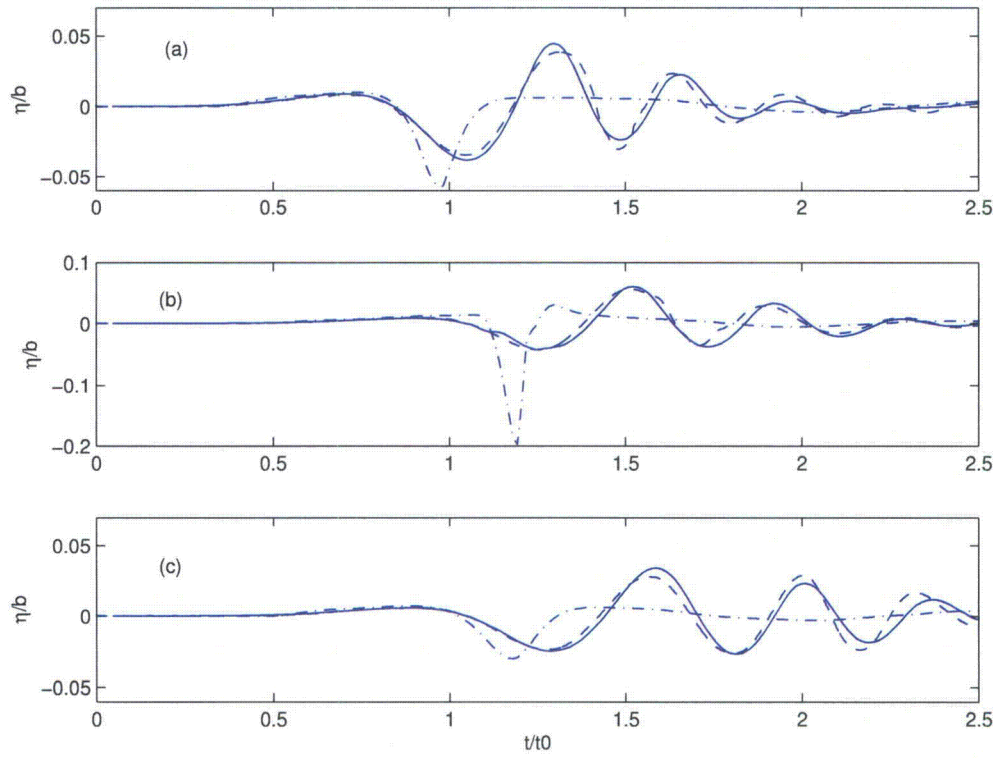

Figure 13: Comparisons between nonhydrostatic numerical results (solid lines), hydrostatic numerical results (dash-dot lines) and experimental data (dashed lines) for free surface elevation for landslide-generated waves at three wave gauges with initial depth of submergence $d = 61$ mm. Gauge coordinates (x, y): (a) (1469, 350)mm; (b) (1,929, 0) mm; (c) (1929, 500) mm, where x is distance from shoreline and y is perpendicular distance from the axis of the shore-normal slide trajectory.

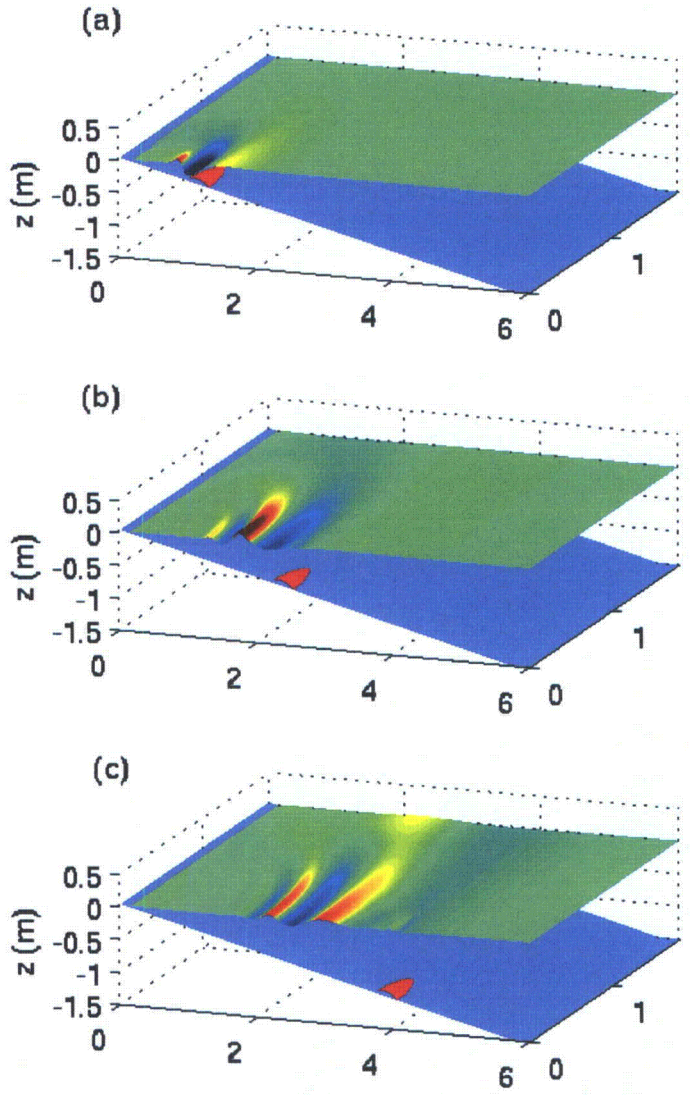

Figure 14: Snapshots of landslide-generated waves simulated using nonhydrostatic model at times (a) $t = 1.0\ s$; (b) $t = 2.0\ s$ and(c) $t = 3.0\ s$ after release of the sliding mass.

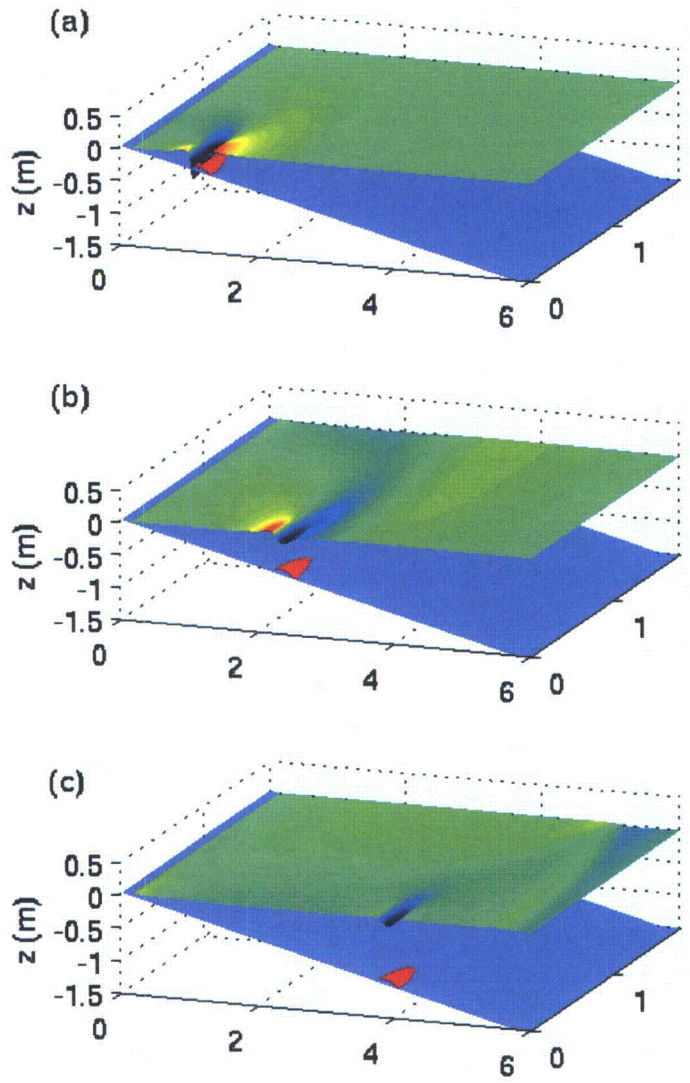

Figure 15: Snapshots of landslide-generated waves simulated using hydrostatic model at times (a) $t = 1.0\ s$; (b) $t = 2.0\ s$ and(c) $t = 3.0\ s$ after release of the sliding mass.

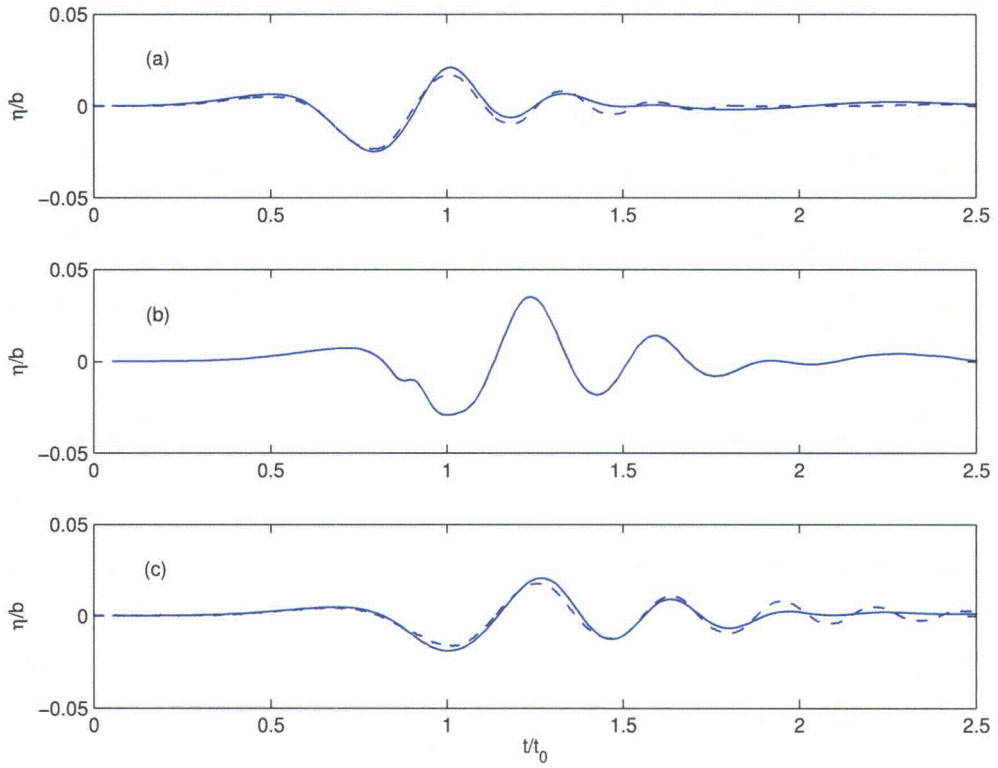

Figure 16: Comparisons between numerical nonhydrostatic results (solid lines) and experimental data(dashed lines) for free surface elevation for landslide-generated waves at three wave gauges with initial depth of submergence $d = 120$ mm. Gauge coordinates (x, y): (a) (1469, 350) mm; (b) (1929, 0) mm; (c) (1929, 500) mm. Experimental data are not available for (b).

A High-Order Adaptive Time-Stepping TVD Solver for Boussinesq Modeling of Breaking Waves and Coastal Inundation

Fengyan Shi[1], James T. Kirby,[1], Jeffrey C. Harris[2], Joseph D. Geiman[1] and Stephan T. Grilli[2]

[1] *Center for Applied Coastal Research, University of Delaware, Newark, DE 19716 USA*
[2] *Department of Ocean Engineering, University of Rhode Island, Narragansett, RI 02882 USA*

Abstract

We present a high-order adaptive time-stepping TVD solver for the fully nonlinear Boussinesq model of Chen (2006), extended to include moving reference level as in Kennedy et al (2001). The equations are reorganized in order to facilitate high-order Runge-Kutta time-stepping and a TVD type scheme with a Riemann solver. Wave breaking is modeled by locally switching to the nonlinear shallow water equations when the Froude number exceeds a certain threshold. The moving shoreline boundary condition is implemented using the wetting-drying algorithm with the adjusted wave speed of the Riemann solver. The code is parallelized using the Message Passing Interface (MPI) with non-blocking communication. Model validations show good performance in modeling wave shoaling, breaking, wave runup and wave-averaged nearshore circulation.

Keywords:

Boussinesq wave model, TVD Riemann solver, breaking wave, coastal

inundation

1. Introduction

The Boussinesq wave model has been a useful tool for modeling surface waves from deep water to the swash zone, as well as wave-induced circulations inside the surfzone. FUNWAVE, the fully nonlinear Boussinesq model developed by the University of Delaware group (Kirby et al.,1998), has come into fairly wide usage in the coastal community. FUNWAVE was based on the fully nonlinear Boussenesq equations derived by Wei et al. (1995) and used the high-order finite difference numerical method.

Since the initial version of FUNWAVE was developed, there were several updates in both theory and numerics for the fully nonlinear Boussinesq model. Gobbi et al. (2000) extended Wei et al. (1995) to higher order, which improved predictions of near-bed kinematics in deeper water. Chen (2006) pointed out missing terms in Wei et al. (1995) which represent the vertical vorticity on a sloping bed. Kennedy et al. (2001) introduced a concept of reference elevation in the derivation of the extended Boussinesq equations to imporve nonlinear performance. Shi et al. (2001) extended the fully nonlinear Boussinesq equations into a non-Cartesian coordinate system.

In the aspect of numerics, Wei and Kirby (1995) initially described a numerical scheme in which time stepping is treated using a fourth-order Adams-Bashforth-Moulton scheme, while spatial differencing is handled using a mixed-order scheme, employing fourth-order accurate centered differ-

ences for first derivatives and second-order accurate derivatives for third derivatives. The choice is made in order to move leading truncation errors to one order higher than the $O(\mu^2)$ dispersive terms, while maintaining the tridiagonal structure of spatial derivatives within time-derivative terms. A non-staggered grid system was used in Wei and Kirby (1995). Shi et al (2001) used similar numerical schemes but a staggered grid approach which has less apparent sensitivity to treatment of boundary conditions. The staggered grid scheme has become the preferred approach in later developments of the Delaware Boussinesq models such as in FUNWAVE 2 (Long and Kirby, 2006).

The moving shoreline condition in FUNWAVE was treated using a so-called slot technique (Kennedy et al., 2000, Chen et al., 2000). In the slot method, deep and narrow slots are added to each grid row, extending down at least to the lowest elevation that will be experienced during shoreface rundown. Recent model tests on the solitary wave runup measured at Oregon State University's O.H. Hinsdale Wave Research Laboratory (ISEC/NEES Workshop, Oregon State University, July, 2009) have raised a concern about considerable errors induced by the slot method. Slots which are too wide relative to the model grid spacing admit too much fluid before filling during runup, and cause both a reduction in amplitude and a phase lag in modeled runup events. At the other extreme, slots which are too narrow tend to induce a great deal of numerical noise, leading to the need for intermittent or even fairly frequent filtering of swash zone solutions.

Wave breaking in FUNWAVE was approached with the eddy viscosity method by Kennedy et al. (2000) following an early eddy viscosity model by Zelt (1991). Kennedy et al. used a model which involves a time history in order to allow the slope of the breaking wave crest to relax after the onset of breaking. Similar approaches were used by other Boussinesq model developers such as Nwogu and Demirbilek (2001) who used a more sophisticated eddy viscosity model in which the eddy viscosity is expressed in terms of turbulent kinetic energy and a length scale.

Recent progress in the development of Boussinesq-type wave models using a hybrid method combining the finite-volume and finite-difference TVD-type schemes (Toro, 2009) have shown robust performance of the shock-capturing method in simulating breaking waves and coastal inundation (Tonelli and Petti, 2009, Roeber et al., 2010, Shiach and Mingham, 2009, Erduran et al., 2005, and others). The shock-capturing scheme makes the treatment of wave breaking straightforward without an artificial viscosity adopted in some breaking wave models such as in Kennedy et al. (2000). The scheme is also able to capture the sharp wave front occurring in the swash zone. The combined finite-volume and finite difference scheme also makes it easy to implement a wetting-drying moving shoreline condition. Recent applications of using such a wetting-drying method have shown to be quite accurate in modeling of wave runup (e.g., Lynett et al., 2002).

This work was motivated by recent needs to model the surfzone and swash zone dynamics and associated breaking wave-induced processes such

as optical properties and sediment transport in a 10 km-scale computational domain, and to model tsunami waves in both a regional/coastal scale for prediction of coastal inundation and a basin scale for wave propagation. We are pursuing a version which is stable and robust in model efficiency and accuracy in a long time wave simulation in a large computational domain.

In this version, we start with a more complete set of fully nonlinear Boussinesq equations developed by Chen (2006), extended to incorporate a moving reference elevation following Kennedy et al (2001). The use of a moving reference elevation is more consistent with a time-varying representation of elevation at a moving shoreline in modeling of a swash zone dynamics and coastal inundation. A conservative form of the equations are derived in order to use a hybrid numerical scheme. Dispersive terms were reorganized with an aim to construct the tridiagonal structure of spatial derivatives within time-derivative terms. The surface elevation gradient term was also rearranged to obtain a numerically well-balanced form which is suitable for any numerical order. In contrast to previous high-order temporal scheme which usually requires uniform time-stepping, we used adaptive time stepping using the third-order Runge-Kutta method. Spatial derivatives were discretized using a combination of finite-volume and finite-difference methods. A fourth-order MUSCL reconstruction technique was used in the Riemann solver. The wave breaking scheme followed the approach of Tonelli and Petti (2009) who used the ability of the nonlinear shallow water equations with a TVD solver to simulate moving hydraulic jumps. Wave breaking is modeled by switching

Boussinesq to NSWE at cells where the Froude number exceeds a certain threshold. A wetting-drying scheme was used to model a moving shoreline.

The model was parallelized using the domain decomposition technique. The Message Passing Interface (MPI) with non-blocking communication is used for data communication between processors.

The paper is organized as follows. Section 2 shows derivations of the conservative form of theoretical equations with a well-balanced pressure gradient term. The numerical implementations, including the hybrid numerical schemes, wetting-drying algorithm, boundary conditions and code parallelization, are described in section 3. Section 4 illustrates four model's applications to problems of wave breaking, wave runup and wave-averaged nearshore circulations. Conclusions are made in section 5.

2. Fully-nonlinear Boussinesq equations

In this section, we describe the development of a set of Boussinesq equations which are accurate to $O(\mu^2)$ in dispersive effects. Here. μ is a parameter characterizing the ratio of water depth to wave length, and is assumed to be small in classical Boussinesq theory. We retain dimensional forms below but will refer to the apparent $O(\mu^2)$ ordering of terms resulting from deviations from hydrostatic behavior in order to identify these effects as needed. The model equations used here follow from the work of Chen (2006). In this and earlier works starting with Nwogu (1993), the horizontal velocity is written

as

$$\mathbf{u} = \mathbf{u}_\alpha + \mathbf{u}_2(z) \tag{1}$$

Here, $\mathbf{u}_\alpha$ denotes the velocity at a reference elevation $z = z_\alpha$, and

$$\mathbf{u}_2(z) = (z_\alpha - z)\nabla A + \frac{1}{2}\left(z_\alpha^2 - z^2\right)\nabla B \tag{2}$$

represents the depth-dependent correction at $O(\mu^2)$, with A and B given by

$$\begin{aligned} A &= \nabla \cdot (h\mathbf{u}_\alpha) \\ B &= \nabla \cdot \mathbf{u}_\alpha \end{aligned} \tag{3}$$

The derivation follows Chen (2006) except for the additional effect of letting the reference elevation z_α vary in time according to

$$z_\alpha = \zeta h + \beta\eta \tag{4}$$

where h is local still water depth, η is local surface displacement and ζ and β are constants, as in Kennedy et al (2001). This addition does not alter the details of the derivation, which are omitted below.

2.1. Governing equations

The equations of Chen (2006) extended to incorporate a possible moving reference elevation follow. The depth-integrated volume conservation equa-

tion is given by

$$\eta_t + \nabla \cdot \mathbf{M} = 0 \tag{5}$$

where

$$\mathbf{M} = H\left\{\mathbf{u}_\alpha + \overline{\mathbf{u}}_2\right\} \tag{6}$$

is the horizontal volume flux. $H = h + \eta$ is the total local water depth and $\overline{\mathbf{u}}_2$ is the depth averaged $O(\mu^2)$ contribution to the horizontal velocity field, given by

$$\overline{\mathbf{u}}_2 = \frac{1}{H}\int_{-h}^{\eta} \mathbf{u}_2(z)dz = \left(\frac{z_\alpha^2}{2} - \frac{1}{6}(h^2 - h\eta + \eta^2)\right)\nabla B + \left(z_\alpha + \frac{1}{2}(h-\eta)\right)\nabla A \tag{7}$$

The depth-averaged horizontal momentum equation can be written as

$$\mathbf{u}_{\alpha,t} + (\mathbf{u}_\alpha \cdot \nabla)\mathbf{u}_\alpha + g\nabla\eta + \mathbf{V}_1 + \mathbf{V}_2 + \mathbf{V}_3 + \mathbf{R} = 0 \tag{8}$$

where g is the gravitational acceleration and $\mathbf{R}$ represents diffusive and dissipative terms including bottom friction and subgrid lateral turbulent mixing. $\mathbf{V}_1$ and $\mathbf{V}_2$ are terms representing the dispersive Boussinesq terms given by

$$\mathbf{V}_1 = \left\{\frac{z_\alpha^2}{2}\nabla B + z_\alpha \nabla A\right\}_t - \nabla\left[\frac{\eta^2}{2}B_t + \eta A_t)\right] \tag{9}$$

$$\mathbf{V}_2 = \nabla\left\{(z_\alpha - \eta)(\mathbf{u}_\alpha \cdot \nabla)A + \frac{1}{2}(z_\alpha^2 - \eta^2)(\mathbf{u}_\alpha \cdot \nabla)B + \frac{1}{2}[A + \eta B]^2\right\} \tag{10}$$

The form of (??) allows for the reference level z_α to be treated as a time-varying elevation, as suggested in Kennedy et al (2001). If this extension is neglected, the term reduced to the form given originally by Wei et al (1995). The expression (??) for $\mathbf{V}_2$ was also given by Wei et al (1995), and is not altered by the choice of a fixed or moving reference elevation.

The term $\mathbf{V}_3$ in (??) represents the $O(\mu^2)$ contribution to the expression for $\boldsymbol{\omega} \times \mathbf{u} = \omega \mathbf{i}^z \times \mathbf{u}$ (with $\mathbf{i}^z$ the unit vector in the z direction) and may be written as

$$\mathbf{V}_3 = \omega_0 \mathbf{i}^z \times \overline{\mathbf{u}}_2 + \omega_2 \mathbf{i}^z \times \mathbf{u}_\alpha \tag{11}$$

where

$$\omega_0 = (\nabla \times \mathbf{u}_\alpha) \cdot \mathbf{i}^z = v_{\alpha,x} - u_{\alpha,y} \tag{12}$$

$$\omega_2 = (\nabla \times \overline{\mathbf{u}}_2) \cdot \mathbf{i}^z = z_{\alpha,x}(A_y + z_\alpha B_y) - z_{\alpha,y}(A_x + z_\alpha B_x) \tag{13}$$

Following Nwogu (1993), z_α is usually chosen in order to optimize the apparent dispersion relation of the linearized model relative to the full linear dispersion in some sense. In particular, the choice $\alpha = (z_\alpha/h)^2/2 + z_\alpha/h = -2/5$ recovers a Padé approximant form of the dispersion relation, while the choice $\alpha = -0.39$, corresponding to the choice $z_\alpha = -0.53h$, minimizes the maximum error in wave pahse speed occuring over the range $0 \leq kh \leq \pi$. Kennedy et al (2001) showed that allowing z_α to move up and down with the passage of the wave field allowed a greater degree of flexibility in optimizing nonlinear behavior of the resulting model equations. In the examples chosen

here, where a great deal of our focus is on the behavior of the model from the break point landward, we adopt Kennedy et al's "datum invariant" form

$$z_\alpha = -h + \beta H = (\beta - 1)h + \beta\eta = \zeta h + (1 + \zeta)\eta \tag{14}$$

with $\zeta = -0.53$ as in Nwogu (1993) and $\beta = 1 + \zeta = 0.47$ This corresponds in essence to a σ coordinate approach which places the reference elevation at a level 53% of the total local depth below the local water surface. This also serves to keep the model reference elevation within the actual water column over the entire wetted extent of the model domain.

2.2. Treatment of the surface gradient term

The hybrid numerical scheme requires a conservative form of continuity equation and momentum equations, thus requiring a modification of the leading order pressure term in the momentum equation. A numerical imbalance problem occurs when the surface gradient term is conventionally split into an artificial flux gradient and a source term that includes the effect of the bed slope for a non-uniform bed. To eliminate errors introduced by the traditional depth gradient method (DGM), a so-called surface gradient method (SGM) proposed by Zhou et al. (2001) was adopted in the TVD based-Boussinesq models in the recent literatures. Zhou et al. discussed an example of SGM in 1-D and verified that the slope-source term may be canceled out by part of the numerical flux term associated with water depth, if the bottom elevation at the cell center is constructed using the average of bottom elevations at

two cell interfaces. Zhou et al. also showed a 2D application but without explicitly describing 2D numerical schemes. Although this scheme can be extended into 2D following the same procedure as in 1D, it was found that the 2D extension may not be trivial in terms of the bottom construction for a 2D arbitrary bathymetry. Kim et al. (2008) pointed out that the water depth in the slope-source term should be written in a discretized form rather than the value obtained using the bottom construction, implying that their revised SGM is valid for general 2D applications.

For the higher-order schemes such as the fourth-order TVD-MUSCL scheme (Yamamoto and Daiguji, 1993, Yamamoto et al., 1998) used in the recent Boussinesq applications, the original SGM and the revised SGM may not be effective in removing the artificial source. This problem was noticed by some recent authors such as Roeber et al. (2010) who kept a first-order scheme (second-order for normal conditions) for the numerical flux term and the slope-source term in order to ensure well-balanced solution without adding noise for a rapidly varying bathymetry.

In fact, the imbalance problem can be solved by a reformulation of this term in terms of deviations away from an unforced but separately specified equilibrium state (see general derivations in Rogers et al., 2003 and recent application in Liang and Marche, 2009). Using this technique, the surface gradient term may be split into

$$gH\nabla\eta = \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] - g\eta\nabla h \tag{15}$$

which is well-balanced for any numerical order under an unforced stationary condition (still water condition).

2.3. Conservative form of fully nonlinear Boussinesq equations

For Chen's (2006) equations or the minor extension considered here, $H\mathbf{u}_\alpha$ can be used as a conserved variable in the construction of a conservative form of Boussinesq equations, but resulting in a source term in the mass conservation equation such as in Shiach and Mingham (2009) and Roeber et al. (2010). An alternative approach is to use $\mathbf{M}$ as a conserved variable in terms of the physical meaning of mass conservation. In this study, we used $\mathbf{M}$, instead of $H\mathbf{u}_\alpha$, in the following derivations of the conservative form of the fully nonlinear Boussinesq equations.

Using M from (??) together with the vector identity

$$\nabla \cdot (\mathbf{uv}) = \nabla \mathbf{u} \cdot \mathbf{v} + (\nabla \cdot \mathbf{v})\mathbf{u} \tag{16}$$

allows (??) to be rearranged as

$$\begin{aligned} &\mathbf{M}_t + \nabla \cdot \left(\frac{\mathbf{MM}}{H}\right) + gH\nabla\eta \\ &= H\left\{\overline{\mathbf{u}}_{2,t} + \mathbf{u}_\alpha \cdot \nabla\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2 \cdot \nabla\mathbf{u}_\alpha - \mathbf{V}_1 - \mathbf{V}_2 - \mathbf{V}_3 - R\right\} \end{aligned} \tag{17}$$

Following Wei et al. (1995), we separate the time derivative dispersion terms in $\mathbf{V}_1$ according to

$$\mathbf{V}_1 = \mathbf{V}'_{1,t} + \mathbf{V}''_1 \tag{18}$$

where

$$\mathbf{V}_1' = \frac{z_\alpha^2}{2}\nabla B + z_\alpha \nabla A - \nabla\left[\frac{\eta^2}{2}B + \eta A\right] \tag{19}$$

and

$$\mathbf{V}_1'' = \nabla\left[\eta_t(A + \eta B)\right] \tag{20}$$

Using (??), (??) and (??), the momentum equation can be rewritten as

$$\begin{aligned}&\mathbf{M}_t + \nabla\cdot\left[\frac{\mathbf{MM}}{H}\right] + \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] = \\ &= H\left\{\overline{\mathbf{u}}_{2,t} + \mathbf{u}_\alpha\cdot\nabla\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2\cdot\nabla\mathbf{u}_\alpha - \mathbf{V}_{1,t}' - \mathbf{V}_1'' - \mathbf{V}_2 - \mathbf{V}_3 - \mathbf{R}\right\} + g\eta\nabla h\end{aligned} \tag{21}$$

A difficulty usually arises in applying the adaptive time-stepping scheme to the time derivative dispersive terms $\bar{u}_{2,t}$ and $\mathbf{V}_{1,t}'$ which are usually calculated using values stored in several time levels in the previous Boussinesq codes such as in Wei et al. (1995) and Shi et al. (2001). The equation can be re-arranged by merging the time derivatives on the right hand side into the time derivative term on the left hand side, giving

$$\begin{aligned}\mathbf{V}_t \quad &+\nabla\cdot\left[\frac{\mathbf{MM}}{H}\right] + \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] = \eta_t(\mathbf{V}_1' - \overline{\mathbf{u}}_2) \\ &+H\left(\mathbf{u}_\alpha\cdot\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2\cdot\nabla\mathbf{u}_\alpha - \mathbf{V}_1'' - \mathbf{V}_2 - \mathbf{V}_3 - \mathbf{R}\right) + g\eta\nabla h\end{aligned} \tag{22}$$

where

$$\mathbf{V} = H(\mathbf{u}_\alpha + \mathbf{V}_1') \tag{23}$$

In (??) η_t can be calculated explicitly using (??) as in Roeber et al. (2010).

Equations (??) and (??) are the governing equations solved in this study. As $\mathbf{V}$ is obtained, the velocity $\mathbf{u}_\alpha$ can be solved by a system of tridiagonal matrix equation formed by (??) in which all cross-derivatives are moved to the right-hand side of the equation.

3. Numerical schemes

3.1. Compact form of governing equations

We define

$$\begin{aligned}
\mathbf{u}_\alpha &= (u, v), \\
\overline{\mathbf{u}}_2 &= (U_4, V_4), \\
\mathbf{M} = (P, Q) &= H\left[u + U_4, v + V_4\right], \\
\mathbf{V}_1' &= (U_1', V_1'), \\
\mathbf{V}_1'' &= (U_1'', V_1''), \\
\mathbf{V}_2 &= (U_2, V_2), \\
\mathbf{V} = (U, V) &= H\left[(u + U_1'), (v + V_1')\right].
\end{aligned}$$

The generalized conservation form of Boussinesq equations can be written as

$$\frac{\partial \mathbf{\Psi}}{\partial t} + \nabla \cdot \mathbf{\Theta}(\mathbf{\Psi}) = \mathbf{S} \tag{24}$$

where $\mathbf{\Psi}$ and $\mathbf{\Theta}(\mathbf{\Psi})$ are the vector of conserved variables and the flux vector function, respectively, and are given by

$$\mathbf{\Psi} = \begin{pmatrix} \eta \\ U \\ V \end{pmatrix}, \qquad \mathbf{\Theta} = \begin{pmatrix} P\mathbf{i} + Q\mathbf{j} \\ \left[\frac{P^2}{H} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{i} + \frac{PQ}{H}\mathbf{j} \\ \frac{PQ}{H}\mathbf{i} + \left[\frac{Q^2}{H} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{j} \end{pmatrix}. \tag{25}$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ g\eta\frac{\partial h}{\partial x} + \psi_x + HR_x \\ g\eta\frac{\partial h}{\partial y} + \psi_y + HR_y \end{pmatrix}, \tag{26}$$

where

$$\psi_x = \eta_t(U_1' - U_4) + H\left(uU_{4,x} + vU_{4,y} + U_4u_x + V_4u_y - U_1'' - U_2 - U_3\right) \tag{27}$$

$$\psi_y = \eta_t(V_1' - V_4) + H\left(uV_{4,x} + vV_{4,y} + U_4v_x + V_4v_y - U_1'' - V_2 - V_3\right) \tag{28}$$

The expanded forms of $(U_1', V_1'), (U_1'', V_1''), (U_2, V_2), (U_3, V_3)$ and (U_4, V_4) can be found in Appendix A. For the term $\mathbf{R}$, the bottom stress is approximated using a quadratic friction equation. A Smagorinsky (1963)-like subgrid turbulent mixing algorithm is implemented following Chen et al. (1999).

3.2. Spatial discretization

A combined finite-volume and finite-difference method was applied to the spatial discretization. For the flux terms and the first-order derivative terms, a fourth-order TVD-MUSCL scheme (Yamamoto and Daiguji, 1993) is used to construct the interface values. For example, in x-direction:

$$\phi^L_{i+1/2} = \phi_i + \frac{1}{6}(\Delta^*\bar{\phi}_{i-1/2} + 2\Delta^*\tilde{\phi}_{i+1/2}), \tag{29}$$

$$\phi^R_{i+1/2} = \phi_{i+1} - \frac{1}{6}(2\Delta^*\bar{\phi}_{i+1/2} + \Delta^*\tilde{\phi}_{i+3/2}), \tag{30}$$

where

$$\Delta^*\bar{\phi}_{i-1/2} = \text{minmod}(\Delta^*\phi_{i-1/2}, b\Delta^*\phi_{i+1/2}), \tag{31}$$

$$\Delta^*\tilde{\phi}_{i+1/2} = \text{minmod}(\Delta^*\phi_{i+1/2}, b\Delta^*\phi_{i-1/2}), \tag{32}$$

$$\Delta^*\bar{\phi}_{i+1/2} = \text{minmod}(\Delta^*\phi_{i+1/2}, b\Delta^*\phi_{i+3/2}), \tag{33}$$

$$\Delta^*\tilde{\phi}_{i+3/2} = \text{minmod}(\Delta^*\phi_{i+3/2}, b\Delta^*\phi_{i+1/2}), \tag{34}$$

$$\Delta^*\phi_{i+1/2} = \Delta\phi_{i+1/2} - \frac{1}{6}\Delta^3\bar{\phi}_{i+1/2}, \tag{35}$$

$$\Delta^3\bar{\phi}_{i+1/2} = \Delta\bar{\phi}_{i-1/2} - 2\Delta\bar{\phi}_{i+1/2} + \Delta\bar{\phi}_{i+3/2}, \tag{36}$$

$$\Delta\bar{\phi}_{i-1/2} = \text{minmod}(\Delta\phi_{i-1/2}, b_1\Delta\phi_{i+1/2}, b_1\Delta\phi_{i+3/2}), \tag{37}$$

$$\Delta\bar{\phi}_{i+1/2} = \text{minmod}(\Delta\phi_{i+1/2}, b_1\Delta\phi_{i+3/2}, b_1\Delta\phi_{i-1/2}), \tag{38}$$

$$\Delta\bar{\phi}_{i+3/2} = \text{minmod}(\Delta\phi_{i+3/2}, b_1\Delta\phi_{i-1/2}, b_1\Delta\phi_{i+1/2}), \tag{39}$$

$$\text{minmod}(i,j) = \text{sign}(i)\max[0, \min\{|i|, \text{sign}(i)j\}], \tag{40}$$

$$\text{minmod}(i,j,k) = \text{sign}(i)\max[0, \min\{|i|, \text{sign}(i)j, \text{sign}(i)k\}], \tag{41}$$

239 in which the coefficients $b_1 = 2$ and $1 \leq b \leq 4$.

240 The numerical fluxes are computed using HLL approximate Riemann
241 solver.

$$\mathbf{\Theta}(\mathbf{\Psi}^L, \mathbf{\Psi}^R) = \begin{cases} \mathbf{\Theta}(\mathbf{\Psi}^L) & \text{if} s_L \geq 0 \\ \mathbf{\Theta}^*(\mathbf{\Psi}^L, \mathbf{\Psi}^R) & \text{if} s_L < 0 < s_R \\ \mathbf{\Theta}(\mathbf{\Psi}^R) & \text{if} s_R \leq 0, \end{cases} \tag{42}$$

242 where

$$\mathbf{\Theta}^*(\mathbf{\Psi}^L, \mathbf{\Psi}^R) = \frac{s_R \mathbf{\Theta}(\mathbf{\Psi}^L) - s_L \mathbf{\Theta}(\mathbf{\Psi}^R) + s_L s_R (\mathbf{\Psi}^R - \mathbf{\Psi}^L)}{s_R - s_L} \tag{43}$$

243 The wave speeds of the Riemann solver are given by

$$s_L = \min(\mathbf{V}^L \cdot \mathbf{n} - \sqrt{g(h+\eta)^L}, u_s - \sqrt{\phi_s}), \tag{44}$$

244

$$s_R = \max(\mathbf{V}^R \cdot \mathbf{n} + \sqrt{g(h+\eta)^R}, u_s + \sqrt{\phi_s}), \tag{45}$$

245 in which u_s and ϕ_s are estimated as

$$u_s = \frac{1}{2}(\mathbf{V}^L + \mathbf{V}^R) \cdot \mathbf{n} + \sqrt{g(\eta+h)^L} - \sqrt{g(\eta+h)^R} \tag{46}$$

246

$$\sqrt{\phi_s} = \frac{\sqrt{g(\eta+h)^L} + \sqrt{g(\eta+h)^R}}{2} + \frac{(\mathbf{V}^L - \mathbf{V}^R) \cdot \mathbf{n}}{4} \tag{47}$$

247 and $\mathbf{n}$ is the normalized side vector for a cell face.

248 Higher derivative terms in ϕ_x and ϕ_y were discretized using the central

differential schemes at the cell centroids as in Wei et al. (2005). No discretization of dispersion terms at the cell interfaces is needed due to using $\mathbf{M}$ as a flux variable. The Surface Gradient Method (Zhou et al, 2001) was used to eliminate unphysical oscillations. Because the pressure gradient term is re-organized as in section 2.2, there is no imbalance issue for the high-order MUSCL scheme.

3.3. Time stepping

The third-order Strong Stability-Preserving (SSP) Runge-Kutta scheme for nonlinear spatial discretization (Gottlieb et al., 2001) was adopted for time stepping. The scheme is given by

$$\begin{aligned}
\mathbf{\Psi}^{(1)} &= \mathbf{\Psi}^n + \Delta t(-\nabla \cdot \mathbf{\Theta}(\mathbf{\Psi}^n) + \mathbf{S}^{(1)}) \\
\mathbf{\Psi}^{(2)} &= \frac{3}{4}\mathbf{\Psi}^n + \frac{1}{4}\left[\mathbf{\Psi}^{(1)} + \Delta t\left(-\nabla \cdot \mathbf{\Theta}(\mathbf{\Psi}^{(1)}) + \mathbf{S}^{(2)}\right)\right] \\
\mathbf{\Psi}^{n+1} &= \frac{1}{3}\mathbf{\Psi}^n + \frac{2}{3}\left[\mathbf{\Psi}^{(2)} + \Delta t\left(-\nabla \cdot \mathbf{\Theta}(\mathbf{\Psi}^{(2)}) + \mathbf{S}^{n+1}\right)\right]
\end{aligned} \tag{48}$$

in which $\mathbf{\Psi}^n$ represents $\mathbf{\Psi}$ value at time level n. $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(2)}$ are values at intermediate stages in the Runge-Kutta integration. As $\mathbf{\Psi}$ is obtained at each intermediate step, the velocity (u, v) can be solved by a system of tridiagonal matrix equation formed by (??). $\mathbf{S}$ need to be updated using (u, v, η) at the corresponding time step and an iteration is needed to achieve convergence.

A adaptive time step is chosen, following the Courant-Friedrichs-Lewy

(CFL) criterion:

$$\Delta t = C\min\left(\min\frac{\Delta x}{|u_{i,j}| + \sqrt{g(h_{i,j}+\eta_{i,j})}}, \min\frac{\Delta y}{|u_{i,j}| + \sqrt{g(h_{i,j}+\eta_{i,j})}}\right) \quad (49)$$

where C is the Courant number and $C = 0.5$ was used in the following examples.

3.4. Wave breaking and wetting-drying schemes for shallow water

The wave breaking scheme follows the approach of Tonelli and Petti (2009) who successfully used the ability of NSWE with a TVD scheme to model moving hydraulic jumps. The fully nonlinear Boussinesq equations are switched to NSWE at cells where the Froude number exceeds an certain threshold. Following Tonelli and Petti, the ratio of wave height to total water depth is chosen to the criterion to switch from Boussinesq to NSWE. The threshold value is set to be 0.8 as suggested by Tonelli and Petti.

The wetting-drying scheme for modeling of a moving boundary is straightforward. The normal flux $\mathbf{n} \cdot \mathbf{M}$ at the cell interface of a dry cell is set to zero. A mirror boundary condition is applied to the fourth-order MUSCL-TVD scheme and discretization of dispersive terms in ψ_x, ψ_y at dry cells. It may be noted that the wave speed of the Riemann solver (??) and (??) for a dry cell are modified as

$$s_L = \mathbf{V}^L \cdot \mathbf{n} - \sqrt{g(h+\eta)^L}, \quad s_R = \mathbf{V}^L \cdot \mathbf{n} + 2\sqrt{g(h+\eta)^L} \text{ (right dry cell)} \quad (50)$$

and

$$s_L = \mathbf{V}^R \cdot \mathbf{n} - \sqrt{g(h+\eta)^R}, \quad s_R = \mathbf{V}^R \cdot \mathbf{n} + 2\sqrt{g(h+\eta)^R} \quad \text{(left dry cell)} \quad (51)$$

3.5. Boundary conditions and wavemaker

We implemented various boundary conditions including wall boundary condition, absorbing boundary condition following Kirby et al. (1998) and periodic boundary condition following Chen et al. (2003).

Wavemakers implemented in the study include Wei and Kirby's (1999) internal wavemakers for regular waves and irregular waves. For the irregular wavemaker, an extension was made to incorporate the alongshore periodicity into wave generation in order to eliminate a boundary effect on wave simulation. The technique exactly follows the strategy in Chen et al. (2003) who adjusted the distribution of wave directions in each frequency bin to obtain alongshore periodicity. This approach is effective in modeling of breaking wave-induced nearshore circulation such as alongshore currents and rip currents.

3.6. Parallelization

In parallelizing the computational model, we use the domain decomposition technique to subdivide the problem into multiple regions and assign each subdomain to a separate processor core. Each subdomain region contains an overlapping area of ghost cells with three rows deep, as dictated by the fourth order MUSCL-TVD scheme. The Message Passing Interface

(MPI) with non-blocking communication is used to exchange the data in the overlapping region between neighboring processors. Velocity components are obtained from Equation (??) by solving tridiagonal matrices using the parallel pipelining tridiagonal solver described in Naik et al. (1993).

To investigate performance of the parallel program, numerical simulations of an idealized case are tested with different number of processors of a linux cluster located at University of Delaware. The test case is set up in a numerical grid of 1800 $\times$ 1800. Figure ?? shows the model speedup versus number of processors. It can be seen that a linear speedup is obtained with a slight decrease with more processors due to the communication cost.

4. Model tests

The model has been validated extensively using laboratory experiments for wave shoaling and breaking as in the FUNWAVE manual by Kirby et al. (1998), and a suite of benchmark tests for wave runup. The interested reader is referred to Shi et al. (2011) and Tehranirad et al.(2011). Here, in this paper, we will present four cases with a focus on examining the shock-capturing scheme for modeling wave breaking, the wetting-drying algorithm for wave runup and the model capability in predicting wave-induced nearshore circulation. The effect of using adaptive time stepping is demonstrated in the wave runup case.

4.1. Breaking waves on a beach

Hansen and Svendsen (1979) carried out laboratory experiments of wave shoaling and breaking on a beach. Waves were generated on a flat bottom at a depth of 3.6 m, and the beach slope was 1:34.26. The experiments included several cases including plunging breakers, plunging-spilling breakers and spilling breakers. In this paper, we simulate two typical cases, a plunging breaker and a spilling breaker, respectively. The wave height and wave period are 4.3 cm and 3.33 s, respectively, for the plunging case, and 2.0 cm and 3.6 s for the spilling case.

Although the shock-capturing breaking algorithm used in Boussinesq wave models has been examined by previous researchers (e.g., Tonelli and Petti, 2009, Shiach and Mingham, 2009 and others), there is a concern about its sensitivity to grid spacing. In this study, we adopted three grid sizes, dx = 0.05 m, 0.025 m and 0.0125 m, respectively, for each cases. Figure ?? shows comparisons of wave height and wave setup between measured data and numerical results from model runs with different grid sizes. The wave breaking location of wave setup/setdown predicted by the three models are in agreement with the data, however, the predicted maximum wave heights are slightly different. The results from the former two models with dx=0.25 m and 0.0125 m are very close, indicating a convergence of grid refinement. All three models underpredict the peak wave height at breaking and overpredict wave height inside of the surfzone. This prediction trend was also found in Kennedy et al. (2000, Figure 2). About 10% underprediction of peak wave

height can be found in our tests with $dx = 0.025$ m and 0.0125 m, which is similar to Kennedy et al. (2000). The model with a coarse grid ($dx = 0.05$ m) underpredicted the maximum wave height by 20%.

To find the cause of the large underprediction of peak wave height by the coarse grid model, we show snapshots of surface elevation from the models with $dx = 0.025$ m and 0.050 m at different times in Figure ??. The model with finer grid resolution switched from the Boussinesq equations to NSWE around $t = 19.9s$ (the model with the coarse grid switched slightly later) at the point where the ratio of surface elevation to water depth reached the threshold value of 0.8. Then, a wave is damped at the sharp front and generates trailing high frequency oscillations. The comparison of wave profiles at an early time (i.e. t= 18.6 s) shows that the coarse grid model underpredicted wave height before the Boussinesq-NSWE switching, indicating that the underprediction is not caused by the the shock-capturing scheme, but by the numerical dissipation due to the coarse grid resolution.

For the case of spilling breaker, the models with three different grid sizes basically predicted slightly different wave peaks as in the plunging wave case, but the differences are less obvious due to the longer waves in the plunging case. Figure ?? shows results with $dx = 0.25$ m, where the model provides very good predictions of wave shoaling to near the breaking limit. Once again, the model overpredicts wave height inside the surf zone.

4.2. Random wave shoaling and breaking on a slope

To study random-wave properties of shoaling and breaking, Mase and Kirby (1992) conducted a laboratory experiment of random wave propagation over a planar beach. The experiment layout is shown in Figure ??, where a constant depth of 0.47 m on the left connects to a constant slope of 1:20 on the right. Two sets of random waves with peak frequencies of 0.6 Hz (run 1) and 1.0 Hz (run 2) were generated by the wavemaker on the left. The target incident spectrum was a Pierson-Moskowitz spectrum. Wave gauges at depths $h = 47$, 35, 30, 25, 20, 17.5, 15, 12.5, 10, 7.5, 5, and 2.5 cm collected time series of surface elevation.

Wei and Kirby (1995) carried out a simulation of run 2 without wave breaking. Later, Kirby et al. (1998) and Kennedy et al. (2000) carried out the same simulation with wave breaking. The present model was set up following Kirby et al. (1998), who used an internal wavemaker located at the toe of the slope where surface elevation is measured (gauge 1). A *FFT* was used to transform between the time domain and frequency domain data required by the wavemaker. The low and high-frequency cutoffs are 0.2 and 10.0 Hz, respectively. The simulation time is the same as the time length of data collection. The computational domain is from x = 0 m to 20 m with a grid size of 0.04 m. The toe of the slope starts at x = 10 m. A sponge layer is applied at the left side boundary to absorb reflected waves, but no sponge layer is needed at the right boundary, which differs from Kirby et al. (1998) who used the slot method combined with a sponge layer at the end of the

domain.

We present the model results for run 2 and compare with the experimental data measured at the other 11 gauges as shown in Figure ??. Figure ?? shows model results (dashed lines) and measured data (solid lines) from $t = 20$ s to $t = 40$ s at those gauges. Both model and data show that most waves start breaking at the depth h= 15 cm. Except for small discrepancies for wave phases, the model reproduces the measured waveform quite well.

To further demonstrate the applicability of the model, we performed third moment computations of the resulting time series of surface elevation. Normalized wave skewness and asymmetry were calculated for both measured and modeled time series of surface elevation according to the following formulations,

$$\begin{aligned} \text{skew} &= \frac{<\eta^3>}{<\eta^2>^{3/2}} \\ \text{asym} &= \frac{<H(\eta)^3>}{<\eta^2>^{3/2}} \end{aligned} \tag{52}$$

where H denotes the Hilbert transform, $<>$ is the mean operator, and the mean has been removed from the time series of surface elevation.

Figure ?? shows the comparisons of skewness and asymmetry between the model results and experiment data. The model predicted skewness and asymmetry reasonably well with a slightly overprediction of wave skewness inside the surf zone.

It is worth mentioning that Kirby et al. (1998) employed more frequent

use of numerical filtering, especially after wave breaking, so that the model run was stable over the entire data time series. The present model did not encounter any stability problem without filtering.

4.3. Solitary wave runup on a shelf with an island

To examine the wetting-drying method used in the present model versus the slot method used in Kennedy et al. (2000) and Chen et al. (2000), we performed a simulation of the solitary wave runup measured recently in a large wave basin at Oregon State University's O.H. Hinsdale Wave Research Laboratory. The basin is 48.8 m long, 26.5 m wide, and 2.1 m deep. A complex bathymetry consists of a 1:30 slope planar beach connected to a triangle shaped shelf and a cylinder structure on the shelf as shown in Figure ??. Solitary waves were generated on the left side by a piston-type wave-maker. Surface elevation and velocity were collected at many locations by wave gauges and ADV's in alongshore and cross-shore arrays (See Swigler, 2011 for details). Figure ?? shows three wave gauges (circles) and three ADV's (triangles) used for model/data comparisons in the present study. Gauge 1 - 3 were located at (x, y) = (7.5, 0.0), (21.0, 0.0) and (13.0, 5.0), respectively. ADV 1 - 3 were located at (13.0 0.0), (21.0, 0.0) and (21.0, -5.0), respectively.

The modeled bathymetry was constructed by combining the measured data of the shelf and the analytical solution of the cylinder which was used for the design of the island in the experiment. The computational domain

was modified by extending the domain from $x = 0.0$ m to -5.0 m with a constant water depth of 0.78 m in order to use a solitary wave solution as an initial condition. The width of the computational domain in the y direction is the same as the basin. Grid spacing used in the model is 0.1 m in both directions. A solitary wave solution based on Nwogu's extended Boussinesq equations was used with centroid located at $x = 0$. The wave height is 0.39 m as the same as that used in the laboratory experiment.

Figure ?? demonstrates modeled water surface at t= 6.4 s, 8.4 s and 14.4 s, respectively. The wave front becomes very sleep as the wave climbs on the shelf, which was well captured by the model. The wave scattering pattern is clearly seen in the bottom panel of Figure ??. Wave breaking on the shelf was observed in the laboratory experiment and was also seen in the present model. Figure ?? shows the variation in time stepping during the simulation. The time step dropped to a minimum around $t = 6.5$ s as the wave collided with the island (top panel of figure ??). The local Froude number reached a maximum at $t = 6.5$ s, reducing the value of the time step based on (??).

Figure ?? shows the time series of modeled surface elevations and measurements at Gauge 1 - 3 (from top to bottom). Good agreement between model and data can be found at the gauge in front of the island (Gauge 1, top panel), as the model successfully predicted the solitary wave propagation and its reflection from the shore. The model also captured the collision of waves from two sides of the island as indicated at the gauge behind the island (Gauge 2, middle panel). The model predicted the timing of wave

collision well but over-predicted the peak of wave runup. The model/data comparison at Gauge 3 (bottom panel), which is located at the north-side shelf, indicates that the model predicted wave refraction and breaking on the shelf reasonably well.

Figure ?? shows model/data comparisons of the velocity time series in the x-direction at ADV 1 (top) and ADV2 (bottom). The model predicted the peak velocity and the entire trend of velocity variation in time at both locations. An underprediction of the seaward velocity can be found at ADV 2. The velocity in the y-direction was not compared at ADV 1 and ADV2 because the measured values are too small. Figure ?? shows the u and v velocity components in the x and y directions at ADV 3, and shows that the model predicted the velocity components in both x and y directions well.

4.4. Wave-averaged nearshore circulation

Boussinesq models have been used to model rip currents (Chen et al., 1999; Johnson and Pattiaratchi, 2006) and alongshore currents (Chen et al., 2003) in field surfzone situations. Most recently, Feddersen et al. (2011) compared results of waves and currents from the Boussinesq model FUNWAVE-C with observations during five surfzone dye release experiments. The comparisons indicated that the Boussinesq model reproduced the observed cross-shore evolution of significant wave height, mean wave angle, bulk directional spread, mean alongshore current, and the frequency-dependent sea-surface elevation spectra and directional moments. Geiman et al. (2011) conducted a

476 numerical study on wave averaging effects on estimates of the surfzone mix-
477 ing using the phase-resolving Boussinesq model FUNWAVE and the wave
478 averaged model Delft3D. Results from both models are compared to field
479 observation at the RCEX field experiment (Brown et al., 2009; MacMahan
480 et al., 2010). Their study shows that each model is able to reproduce 1-hour
481 time-averaged mean Eulerian velocities consistent with field measurements at
482 stationary current meters. However, the spatial distribution of wave height
483 inside the surfzone was different between the two models, due to the different
484 mechanisms for wave breaking.

485 To check the breaking scheme used in the present model and its conse-
486 quences to wave-induced currents, we set up the present model in the same
487 way as in Geiman et al. (2011), except that no sponge layer was applied for
488 the presnet model at the shoreline positon. Both models used a grid size of
489 $\mathrm{d}x = \mathrm{d}y = 2$ m and a north-south periodic boundary condition in a compu-
490 tational domain of 1000 m $\times$ 400 m, as shown in Figure ?? (the bathymetry
491 is also north-south periodic, the figure only shows part of computational do-
492 main). An internal wavemaker for directional spectral wave generation was
493 located at 350 m away from the shoreline. It generated waves with wave
494 height $H_{rms} = 0.92$ m and period $T_{mo} = 10.5$ s , which were from wave data
495 observed at 13 m water depth during the instrument deployment.

496 Figure ?? shows wave-averaged currents calculated by 1-hour averaging
497 over modeled (u_α, v_α). The mean circulation pattern was tied to the rip
498 channels.

The cross-shore and alongshore distribution of wave height H_{rms} was calculated (1-hour averaged) at the measurement locations marked in Figure ??. The top panel in Figure ?? shows the cross-shore distribution of modeled H_{rms} at y= 65 m in comparison with data (circles). A fairly good agreement is obtained from the model/data comparison. The model/data comparison of alongshore distribution of H_{rms} at x =100m is shown in the bottom panel of Figure ??. The model predicted a fairly constant alongshore distribution of H_{rms} as the same as in Geiman et al. (2011). Both the present model and the model used by Geiman et al. show less variance than the distribution calculated by Delft3D due to the two different types of wave breaking parameterizations (Geiman et al., 2011). Figure ?? shows comparison of wave-averaged cross-shore velocity (top panel) and alongshore velocity (bottom panel) between model (solid lines) and data (circles). The alongshore variation in cross-shore velocity is very correspondent with the rip-channel system. The model shows a good agreement with the data.

The present model basically produced similar results compared to the FUNWAVE version used by Geiman et al. (2011) and a similar agreement with the data, suggesting that the shock-capturing breaking scheme is analogous to the artificial eddy viscosity in modeling breaking wave-induced circulations.

5. Conclusions

A new version of the FUNWAVE model was developed and is based on a more complete set of fully nonlinear Boussinesq equations with the vertical vorticity correction derived by Chen et al. (2006) and a time-varying reference elevation introduced by Kennedy et al. (2001). The equations were reorganized in order to facilitate a hybrid numerical scheme which include the third-order Runge-Kutta time-stepping and the fourth-order MUSCL-TVD scheme with the Riemann solver. Wave breaking is modeled by locally switching to the nonlinear shallow water equations where the Froude number exceeds an certain threshold. The wetting-drying method was implemented to model a moving shoreline, instead of the slot method used in the previous FUNWAVE model. The code was parallelized using MPI with non-blocking communication.

Benchmark tests verified the model's capability in simulating wave shoaling, breaking, and wave-induced nearshore circulations, and suggested the following advantages of the new model versus the previous version of FUNWAVE:

(1) The adaptive time stepping is more efficient in a simulation where the local Froude number varies in a large range. The time step used as a constant in the previous FUNWAVE version is usually on an *ad hoc* basis due to unpredictable supercritical fluid conditions.

(2) The shock capturing scheme is robust not only in the treatment of wave breaking, but also in the suppression of numerical instability, especially

in modeling of wave breaking. No filtering is needed in the present model.

(3) The wetting-drying method is more proper than the slot method in modeling of the swash zone and coastal inundation.

In addition, the model accurately predicted wave runup against a suite of benchmark test data (Tehranirad et al., 2011).

Acknowledgements This work was supported by the Office of Naval Research, Coastal Geosciences Program Grant N00014-10-1-0088 (Shi and Kirby), and the National Science Foundation, Physical Oceanography Program Grant OCE-0727376 (Kirby and Geiman), and Geophysics Program Grant EAR-09-11499 (Grilli and Harris).

Appendix A. Expansions of $\mathbf{V}_1'$, $\mathbf{V}_1''$, $\mathbf{V}_2$, $\mathbf{V}_3$ and $\mathbf{V}_4$

The expanded forms of $(U_1', V_1'), (U_1'', V_1''), (U_2, V_2), (U_3, V_3)$ and (U_4, V_4) can be written as

$$\begin{aligned} U_1' = \quad & \frac{1}{2}(1-\beta)^2 h^2 (u_{xx} + v_{xy}) - (1-\beta) h \left[(hu)_{xx} + (hv)_{xy}\right] \\ & - \left[\beta(1-\beta)h\eta - \frac{1}{2}\beta^2\eta^2\right](u_{xx} + v_{xy}) + \beta\eta\left[(hu)_{xx} + (hv)_{xy}\right] \\ & - \left\{\frac{\eta^2}{2}(u_x + v_y) + \eta\left[(hu)_x + (hv)_y\right]\right\}_x \end{aligned} \tag{53}$$

$$\begin{aligned} V_1' = \quad & \frac{1}{2}(1-\beta)^2 h^2 (u_{xy} + v_{yy}) - (1-\beta) h \left[(hu)_{xy} + (hv)_{yy}\right] \\ & - \left[\beta(1-\beta)h\eta - \frac{1}{2}\beta^2\eta^2\right](u_{xy} + v_{yy}) + \beta\eta\left[(hu)_{xy} + (hv)_{yy}\right] \\ & - \left\{\frac{\eta^2}{2}(u_x + v_y) + \eta\left[(hu)_x + (hv)_y\right]\right\}_y \end{aligned} \tag{54}$$

$$U_1'' = \left\{\eta\eta_t(u_x + v_y) + \eta\left[(hu)x + (hv)_y\right]\right\}_x \tag{55}$$

$$V_1'' = \left\{\eta\eta_t(u_x + v_y) + \eta\left[(hu)x + (hv)_y\right]\right\}_y \tag{56}$$

$$U_2 = \quad \left\{(\beta-1)(h+\eta)\left[u\left((hu)_x + (hv)_y\right)_x + v\left((hu)_x + (hv)_y\right)_y\right]\right.$$

$$+\left[\frac{1}{2}(1-\beta)^2h^2-\beta(1-\beta)h\eta+\frac{1}{2}(\beta^2-1)\eta^2\right]\left[u\left(u_x+v_y\right)_x+v\left(u_x+v_y\right)_y\right]$$
$$+\frac{1}{2}\left[(hu)_x+(hv)_y+\eta(u_x+v_y)\right]^2\Bigg\}_x \tag{57}$$

$$V_2 = \Big\{(\beta-1)(h+\eta)\left[u\left((hu)_x+(hv)_y\right)_x+v\left((hu)_x+(hv)_y\right)_y\right]$$
$$+\left[\frac{1}{2}(1-\beta)^2h^2-\beta(1-\beta)h\eta+\frac{1}{2}(\beta^2-1)\eta^2\right]\left[u\left(u_x+v_y\right)_x+v\left(u_x+v_y\right)_y\right]$$
$$+\frac{1}{2}\left[(hu)_x+(hv)_y+\eta(u_x+v_y)\right]^2\Bigg\}_y \tag{58}$$

$$U_3 = -v\omega_1-\omega_0\left\{\left[(\beta-\frac{1}{2})(h+\eta)\right]\left[(hu)_x+(hv)_y\right]_y\right. \tag{59}$$
$$\left.+\left[(\frac{1}{3}-\beta+\frac{1}{2}\beta^2)h^2+(\frac{1}{6}-\beta+\beta^2)\eta h+(\frac{1}{2}\beta^2-\frac{1}{6})\eta^2\right](u_x+v_y)_y\right\}$$

$$V_3 = -v\omega_1-\omega_0\left\{\left[(\beta-\frac{1}{2})(h+\eta)\right]\left[(hu)_x+(hv)_y\right]_x\right. \tag{60}$$
$$\left.+\left[(\frac{1}{3}-\beta+\frac{1}{2}\beta^2)h^2+(\frac{1}{6}-\beta+\beta^2)\eta h+(\frac{1}{2}\beta^2-\frac{1}{6})\eta^2\right](u_x+v_y)_x\right\}$$

555 where

$$\omega_0 = v_x-u_y \tag{61}$$

556

$$\begin{aligned}\omega_1 = \quad & (1-\beta)h_x\left\{\left[(hu)_x+(hv)_y\right]_y+b_2h\left(u_x+v_y\right)_y\right\} \\ & -(1-\beta)h_y\left\{\left[(hu)_x+(hv)_y\right]_x+b_2h\left(u_x+v_y\right)_x\right\}\end{aligned} \tag{62}$$

$$\begin{aligned}U_4 = \quad & (\frac{1}{3}-\beta+\frac{1}{2}\beta^2)h^2(u_{xx}+v_{xy})+(\beta-\frac{1}{2})h\left[(hu)_{xx}+(hv)_{xy}\right] \\ & +\left\{\left[(\frac{1}{6}-\beta+\beta^2)h\eta+(\frac{1}{2}\beta^2-\frac{1}{6})\eta^2\right](u_{xx}+v_{xy})\right. \\ & \left.+(\beta-\frac{1}{2})\eta\left[(hu)_{xx}+(hv)_{xy}\right]\right\}\end{aligned} \tag{63}$$

$$\begin{aligned}V_4 = \quad & (\frac{1}{3}-\beta+\frac{1}{2}\beta^2)h^2(u_{xy}+v_{yy})+a_2h\left[(hu)_{xy}+(hv)_{yy}\right] \\ & +\left\{\left[(\frac{1}{6}-\beta+\beta^2)h\eta+(\frac{1}{2}\beta^2-\frac{1}{6})\eta^2\right](u_{xy}+v_{yy})\right. \\ & \left.+(\beta-\frac{1}{2})\eta\left[(hu)_{xy}+(hv)_{yy}\right]\right\}\end{aligned} \tag{64}$$

References

Bermúdez, A. and Vázquez, M. E., 1994, "Upwind methods for hyperbolic conservation laws with source terms", *Comput. Fluids*, **23**, 1049-1071.

Brown, J., MacMahan, J., Reniers, A. J. H. M. and Thornton, E., 2009, "Surf zone diffusivity on a rip-channeled beach", *J. Geophys. Res.*, **114**, C11015, doi:10.1029/2008JC005158.

Briggs, M. J., Synolakis, C. E., and Harkins, G. S., 1994, "Tsunami run- up on a conical island", *Proc., Waves-Physical and Numerical Modelling*, International Association for Hydraulic Research, Delft, The Netherlands, 446-455.

Chen, Q., Dalrymple, R. A., Kirby, J. T., Kennedy, A. B. and Haller, M. C., 1999, "Boussinesq modelling of a rip current system", *J. Geophys. Res.*, **104**, 20,617-20,637.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Kennedy, A. B. and Chawla, A., 2000, "Boussinesq modeling of wave transformation, breaking and runup. II: 2D", *J. Waterway, Port, Coastal and Ocean Engineering*, **126**, 48-56.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Shi, F. and Thornton, E. B., 2003, "Boussinesq modeling of longshore currents", *J. Geophys. Res.*, **108**(C11), 3362, doi:10.1029/2002JC001308.

Chen, Q., 2006, "Fully nonlinear Boussinesq-type equations for waves and currents over porous beds", *J. Eng. Mech.*, **132**, 220-230.

Erduran, K. S., Ilic, S., and Kutija, V., 2005, "Hybrid finite-volume finite-difference scheme for the solution of Boussinesq equations", *Int. J. Num. Meth. Fluids*, **49**, 1213-1232.

Feddersen, F., Clark, D. B., and Guza, R. T., 2011, "Modeling surfzone tracer plumes, Part 1: waves, mean currents, and low-freqency eddies, *J. Geophys. Res.*, in review.

Geiman, J. D., Kirby, J. T., Reniers, A. J. H. M., and MacMahan, J. H., 2011, "Effects of wave averaging on estimates of fluid mixing in the surf zone", *J. Geophys. Res.*, **116**, C04006, doi:10.1029/2010JC006678.

Gobbi, M. F., Kirby, J. T. and Wei, G., 2000, "A fully nonlinear Boussinesq model for surface waves. II. Extension to $O(kh^4)$", *J. Fluid Mech.*, **405**, 181-210.

Gottlieb, S., Shu C.-W., and Tadmore, E., 2001, "Strong stability-preserving high-order time discretization methods", *SIAM Review*, **43** (1), 89 - 112.

Hansen, J. B., and Svendsen, I. A., 1979, "Regular waves in shoaling water: Experimental data", Series Paper 21, ISVA, Technical Univ. of Denmark, Denmark.

597 Johnson, D., and Pattiaratchi, C., 2006, "Boussinesq modelling of transient
598 rip currents", *Coastal Eng.*, **53**, 419-439.

599 Kennedy, A. B., Chen, Q., Kirby, J. T. and Dalrymple, R. A., 2000, "Boussi-
600 nesq modeling of wave transformation, breaking and runup. I: 1D", *J.*
601 *Waterway, Port, Coastal and Ocean Engineering*, **126**, 39-47.

602 Kennedy, A. B., Kirby, J. T., Chen, Q. and Dalrymple, R. A., 2001, "Boussinesq-
603 type equations with improved nonlinear performance", *Wave Motion*,
604 **33**, 225-243.

605 Kim, D. H., Cho, Y. S. and Kim, H. J., 2008, "Well balanced scheme be-
606 tween flux and source terms for computation of shallow-water equations
607 over irregular bathymetry", *J. Eng. Mech.*, **134**, 277-290.

608 Kim, D. H., Lynett, P. J. and Socolofsky, S. A., 2009, "A depth-integrated
609 model for weakly dispersive, turbulent, and rotational fluid flows",
610 *Ocean Modeling*, **27**, 198-214.

611 Kim, D. H., 2009, "Turbulent flow and transport modeling by long waves
612 and currents", Ph.D. dissertation, Texas A& M University.

613 Kirby, J. T., Wei, G., Chen, Q., Kennedy, A. B. and Dalrymple, R. A.,
614 1998, " FUNWAVE 1.0, Fully nonlinear Boussinesq wave model. Doc-
615 umentation and user's manual", Research Report CACR-98-06, Center
616 for Applied Coastal Research, Department of Civil and Environmental
617 Engineering, University of Delaware.

618 Liang, Q. and Marche, F., 2009, "Numerical resolution of well-balanced
619 shallow water equations with complex source terms", *Adv. Water Res.*,
620 **32**, 873 - 884.

621 Long, W. and Kirby, J. T., 2006, "Boussinesq modeling of waves, currents
622 and sediment transport", Research Report CACR-06-02, Center for Ap-
623 plied Coastal Research, Dept. of Civil and Environmental Engineering,
624 Univ. of Delaware, Newark.

625 Lynett, P. J., Wu, T.-R. and Liu, P. L.-F., 2002, "Modeling wave runup
626 with depth-integrated equations", *Coastal Engineering*, **46**, 89-107.

627 MacMahan, J., Brown, J. Brown, J., Thornton, E., Reniers, A., Stanton,
628 T., Henriquez, M., Gallagher, E., Morrison, J., Austin, M. J., Scott, T.
629 M., and Senechal, N., 2010, "Mean Lagrangian flow behavior on open
630 coast rip-channeled beaches: New perspectives", *Mar. Geol.*, **268**, 1 -
631 15.

632 Madsen, P.A., Sorensen, O.R., 1992, "A new form of the Boussinesq equa-
633 tions with improved linear dispersion characteristics. Part 2. A slowly-
634 varying bathymetry", *Coastal Engineering*, **18**, 183-204.

635 Mase, H., and Kirby, J. T., 1992, Hybrid frequency-domain KdV equation
636 for random wave transformation, *Proc. 23rd Int. Conf. Coast. Eng.*,
637 ASCE, New York, 474-487.

638 Naik, N. H., Naik, V. K., and Nicoules, M., 1993, "Parallelization of a class
639 of implicit finite difference schemes in computational fluid dynamics",
640 *Int. J.High Speed Computing*, **5**, 1-50.

641 Ning, D. Z., Zang, J., Liang, Q., Taylor, P. H., and Borthwick, A. G. L.,
642 2008, "Boussinesq cut-cell model for non-linear wave interaction with
643 coastal structures", *Int. J. Num. Meth. Fluids*, **57**, 1459-1483.

644 Nwogu, O., 1993, "An alternative form of the Boussinesq equations for
645 nearshore wave propagation", *J. Waterway, Port, Coastal, and Ocean*
646 *Engineering*, **119**, 618-638.

647 Nwogu, O. and Demirbilek, Z., 2001, "BOUSS-2D: A Boussinesq wave
648 model for coastal regions and harbors", ERDC/CHL TR-01-25, Coastal
649 and Hydraulics Laboratory, USACOE Engineer Research and Develop-
650 ment Center, Vicksburg, MS.

651 Roeber, V., Cheung, K. F., and Kobayashi, M. H., 2010, "Shock-capturing
652 Boussinesq-type model for nearshore wave processes", *Coastal Engi-*
653 *neering*, **57**, 407-423.

654 Rogers, B. D., Borthwick, A. G. L., and Taylor, P. H., 2003, "Mathemat-
655 ical balancing of flux gradient and source terms prior to using Roe's
656 approximate Riemann solver", *J. Comp. Phys.*, **192**, 422-451.

657 Shi, F., Dalrymple, R. A., Kirby, J. T., Chen, Q. and Kennedy, A., 2001,

"A fully nonlinear Boussinesq model in generalized curvilinear coordinates", *Coastal Engineering*, **42**, 337-358.

Shi, F., Kirby, J. T., Tehranirad, B. and Harris, J. C., 2011, "FUNWAVE-TVD, documentation and users' manual", Research Report, CACR-11-xx, University of Delaware, Newark, Delaware.

Shiach, J. B. and Mingham, C. G., 2009, "A temporally second-order accurate Godunov-type scheme for solving the extended Boussinesq equations", *Coast. Eng.*, **56**, 32-45.

Sitanggang, K. I. and Lynett, P., 2005, "Parallel computation of a highly nonlinear Boussinesq equation model through domain decomposition", *Int. J. Num. Meth. Fluids*, **49**, 57-74.

Smagorinsky, J., 1963, "General circulation experiments with the primitive equations. I. The basic experiment", *Mon. Weather Rev*, **91**, 99-165.

Swigler, D. and Lynett, P. ,2011, "Laboratory study of the three-dimensional turbulence and kinematic properties associated with a solitary wave traveling over an alongshore-variable, shallow shelf", in review.

Synolakis, C. E., Bernard, E. N., Titov, V. V., Kânoglu, U. and González, F. I., 2007, "Standards, criteria, and procedures for NOAA evaluation of tsunami numerical models", *NOAA Tech. Memo. OAR PMEL-135*, National Oceanic and Atmospheric Administration.

Tehranirad, B., Shi, F., Kirby, J. T., Harris, J. C. and Grilli, S., 2011, "Benchmark tests of tsunami wave runup using FUNWAVE-TVD", to be submitted to Proceedings of National Tsunami Workshop.

Ting, F.C.K., and Kirby, J.T., 1994, "Observation of undertow and turbulence in a laboratory surf zone", *Coast. Eng.*, **24**, 51-80.

Tonelli, M. and Petti, M., 2009, "Hybrid finite volume - finite difference scheme for 2DH improved Boussinesq equations", *Coast. Engrng.*, **56**, 609-620.

Tonelli, M. and Petti, M., 2010, "Finite volume scheme for the solution of 2D extended Boussinesq equations in the surf zone", *Ocean Engrng.*, **37**, 567-582.

Toro, E. F., 2009, *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*, Third edition, Springer, New York.

Wei, G. and Kirby, J. T., 1995, "A time-dependent numerical code for extended Boussinesq equations", *J. Waterway, Port, Coastal and Ocean Engineering*, **120**, 251-261.

Wei, G., Kirby, J.T., Grilli, S.T., Subramanya, R., 1995, "A fully nonlinear Boussinesq model for surface waves: Part I. Highly nonlinear unsteady waves", *J. Fluid Mech.*, **294**, 71-92.

Yamamoto, S. and Daiguji, H., 1993, "Higher-order-accurate upwind schemes

for solving the compressible Euler and Navier-Stokes equations", *Computers and Fluids*, **22**, 259-270.

Yamamoto, S., Kano, S. and Daiguji, H, 1998, "An efficient CFD approach for simulating unsteady hypersonic shock-shock interference flows", *Computers and Fluids*, **27**, 571-580.

Zelt, J. A., 1991, "The runup of nonbreaking and breaking solitary waves", *Coast. Eng.*, **15**, 205-246.

Zhou, J. G., Causon, D. M., Mingham C. G., and Ingram, D. M., 2001, "The surface gradient method for the treatment of source terms in the shallow-water equations", *J. Comp. Phys.*, **168**, 1-25.

List of Figures

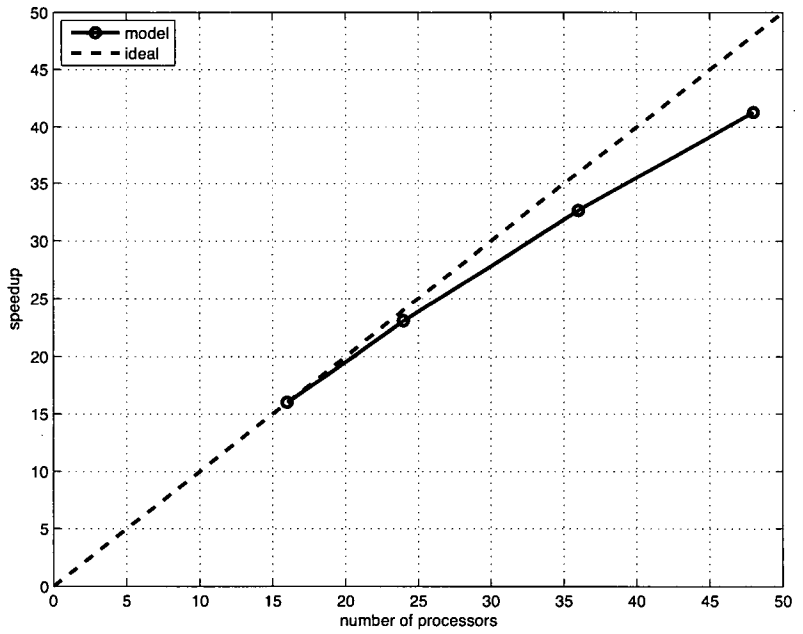

Figure 1: Variation in model performance with number of processors for a 5400 x 3600 domain. Straight line indicates arithmetic speedup. Actual performance is shown in the curved line.

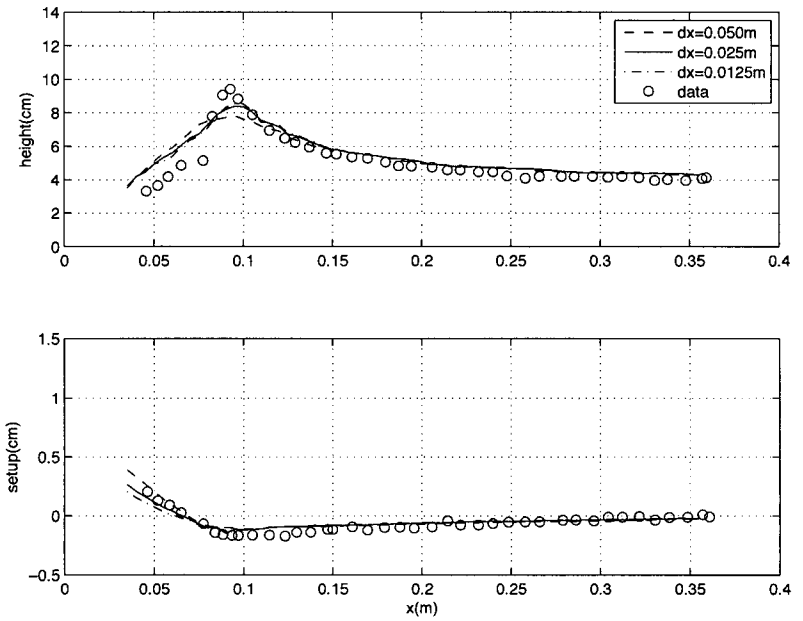

Figure 2: Comparisons of wave height (upper panel) and wave setup (lower panel) between measured data and model results from grid resolutions of $dx = 0.0125$ m, 0.025 m and 0.050 m, respectively. Case: plunging breaker.

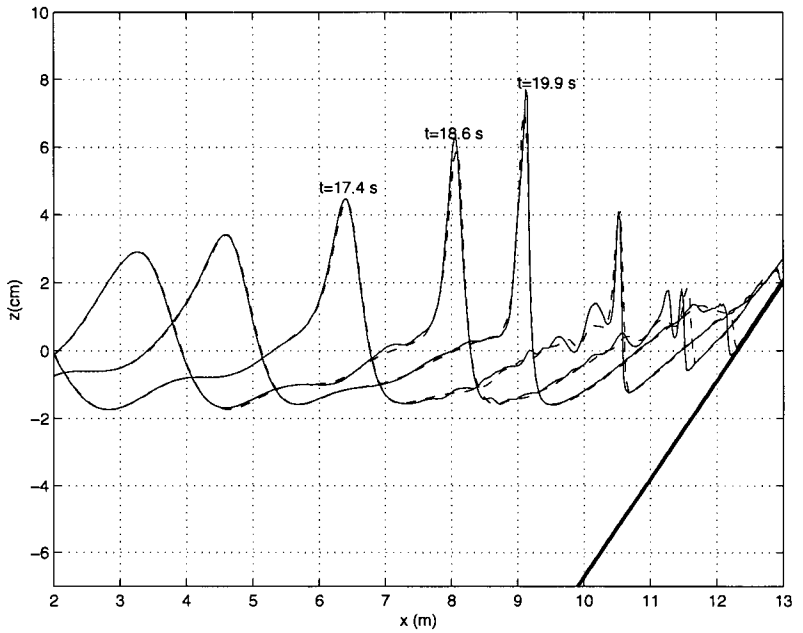

Figure 3: Snapshots of surface elevation at $t = 17.4$, 18.6 and 19.9 s from models with grid resolutions of $\mathrm{d}x = 0.025$ (solid lines) and 0.050 m (dashed lines).

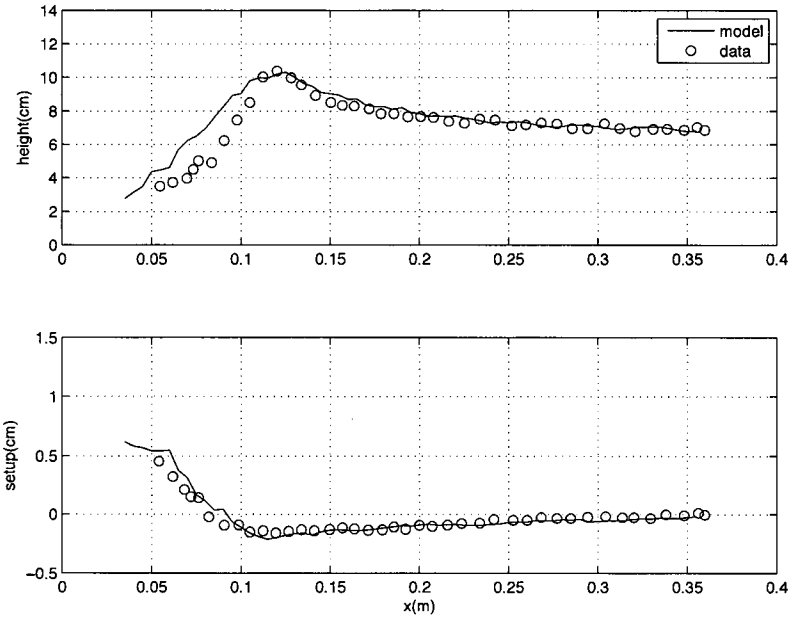

Figure 4: Model/data comparisons of wave height (upper panel) and wave setup (dx = 0.025 m). Case: spilling breaker.

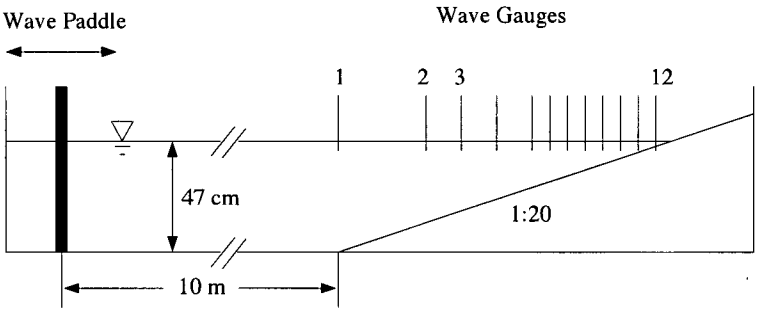

Figure 5: Experiment layout of Mase and Kirby (1992).

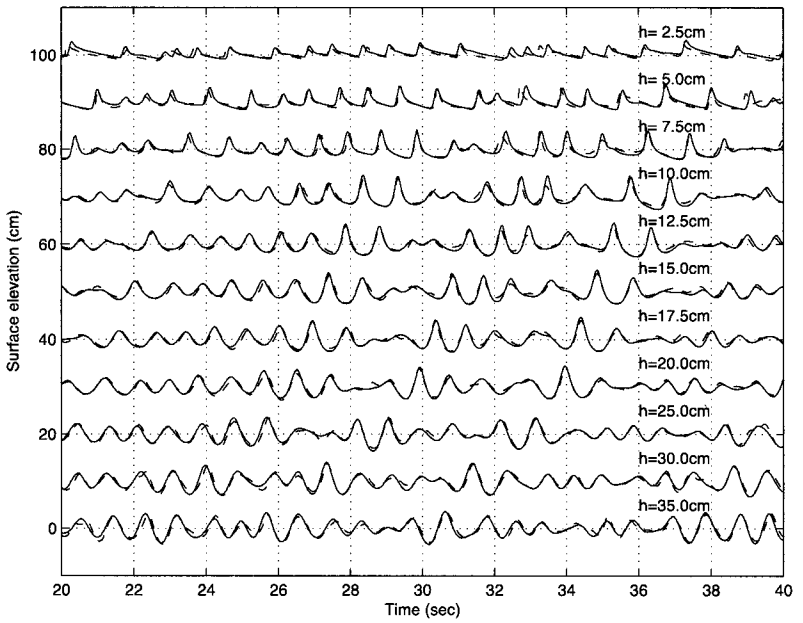

Figure 6: Time series comparison of η between model (dashed lines) and data (solid lines) at 11 wave gauges in Mase and Kirby (1992).

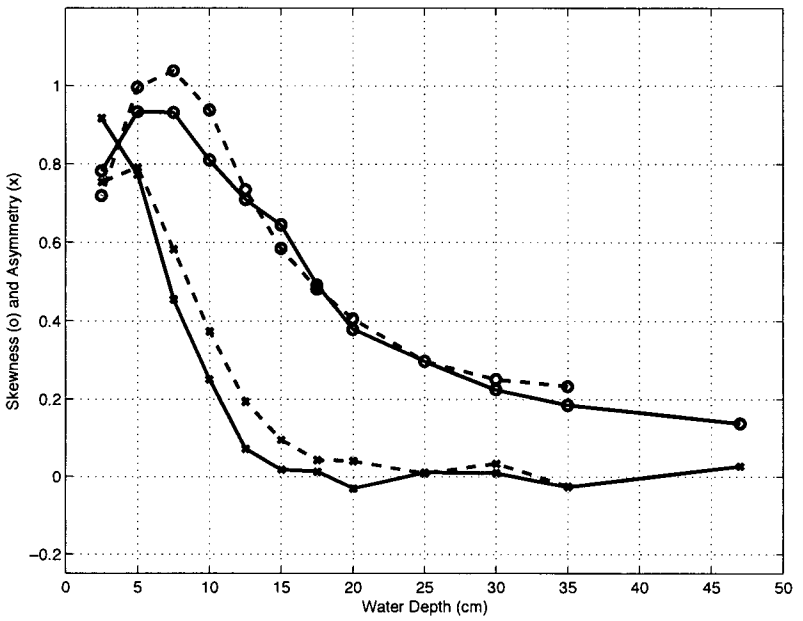

Figure 7: Comparison of skewness (○) and asymmetry (×) at different water depths. Solid lines are experiment data (Mase and Kirby, 1992). Dashed lines are numerical results

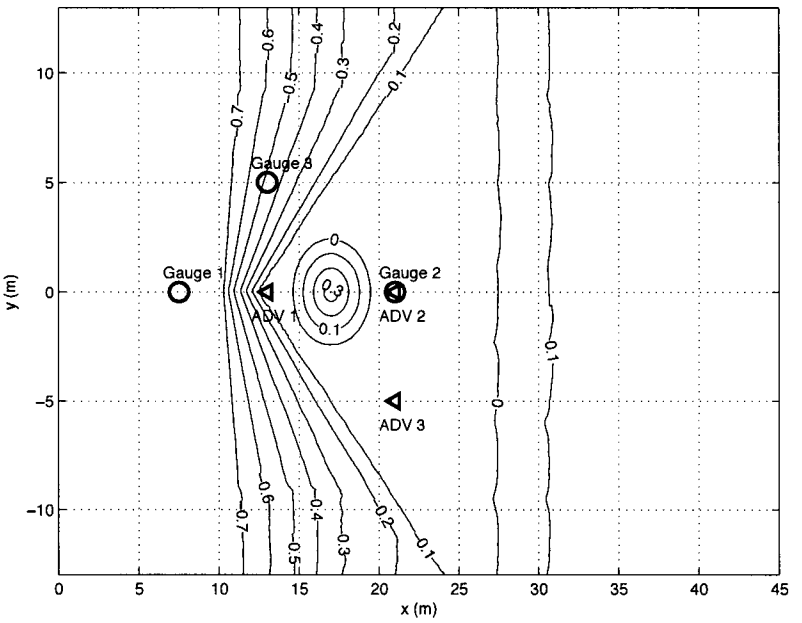

Figure 8: Bathymetry contours (in meters) and measurement locations used in model/data comparisons. Circles: pressure gauges, triangles: ADV.

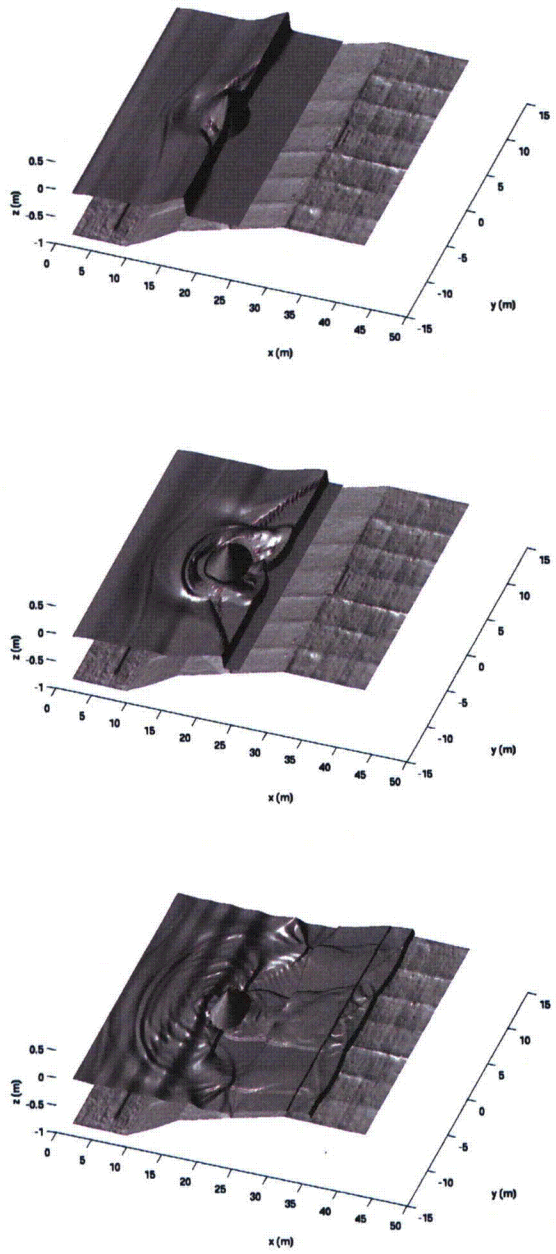

Figure 9: Modeled water surface at (top) $t = 6.4$ s, (middle) $t = 8.4$ s, (bottom) $t = 14.4$ s.

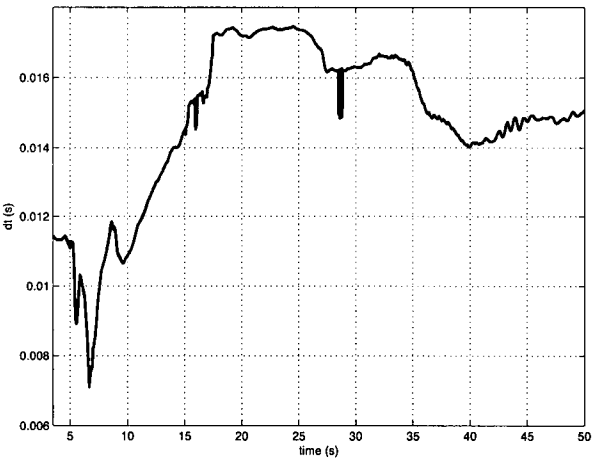

Figure 10: Time step variation.

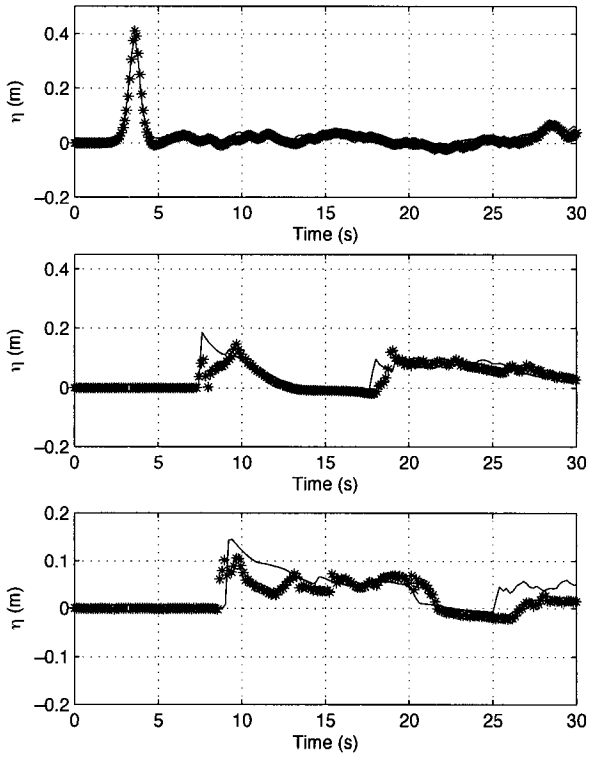

Figure 11: Model/data comparisons of time series of surface elevation at (top) Gauge 1, (middle) Gauge 2,and (bottom) Gauge 3. Solid line: model, stars: data.

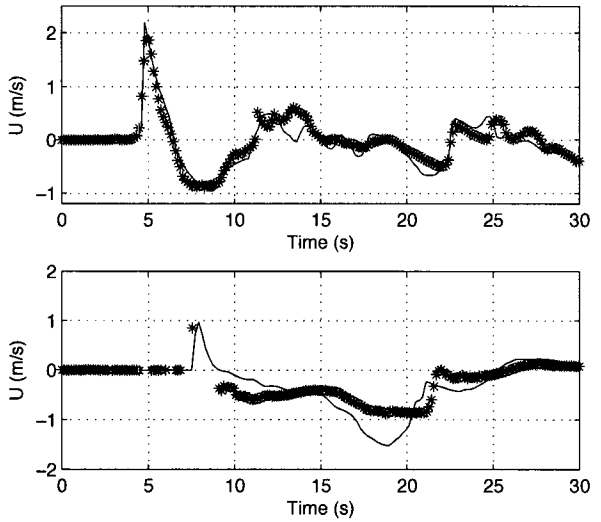

Figure 12: Model/data comparisons of time series of velocity at (top) ADV 1 and (bottom) ADV 2. Solid line: model, stars: data.

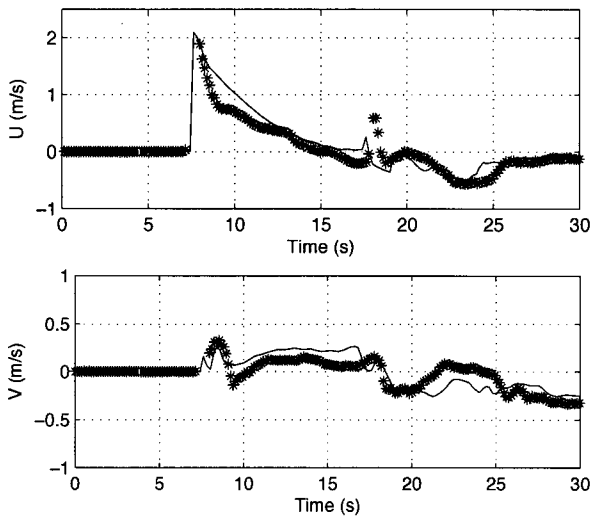

Figure 13: Model/data comparisons of time series of velocity in (top) x direction and (bottom) y direction at ADV 3. Solid line: model, stars: data.

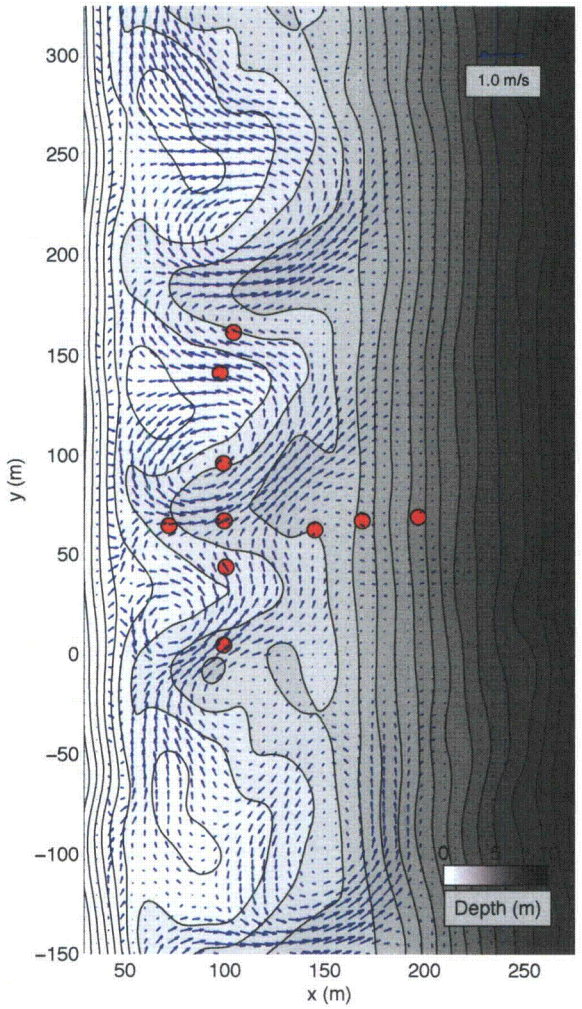

Figure 14: Wave-averaged current, water depth contours and locations of in situ instrumentation (red dots).

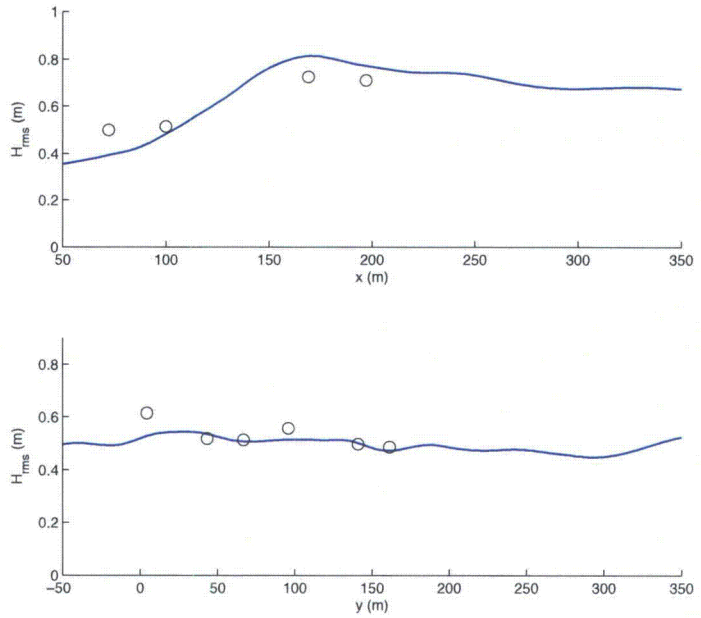

Figure 15: Comparison of H_{rms} between data (circles) and model (solid lines) at y = 65 m as a function of cross-shore location (top) and H_{rms} as a function of alongshore location at x = 100 m inside the surfzone (bottom).

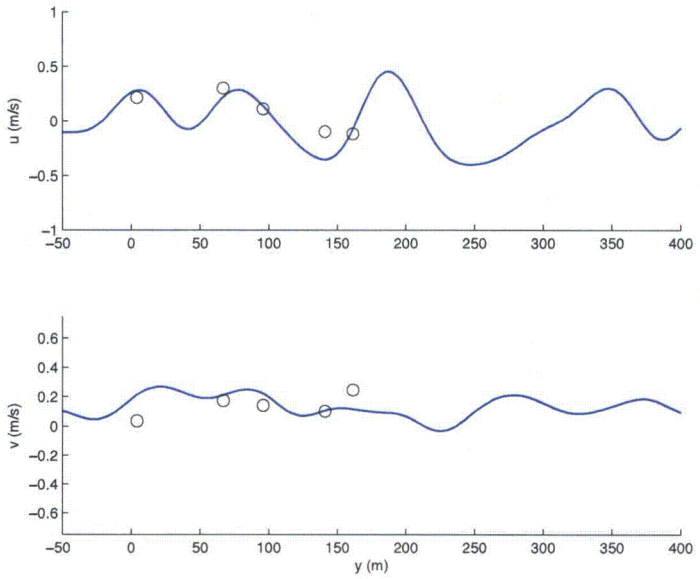

Figure 16: Comparison of wave-averaged cross-shore velocity (top) and alongshore velocity (bottom) between model (solid lines) and data (circles).

FUNWAVE-TVD

Fully Nonlinear Boussinesq Wave Model with TVD Solver Documentation and User's Manual

Fengyan Shi, James T. Kirby and Babak Tehranirad
Center for Applied Coastal Research, University of Delaware, Newark, DE 19716
Jeffrey C. Harris and Stephan Grilli
Department of Ocean Engineering, University of Rhode Island, Narragansett, RI 02882

May 13, 2011

Center for Applied Coastal Research
University of Delaware
Research Report NO. CACR-11-xx

Acknowledgements

This model development was a part of the surfzone optics project sponsored by the Office of Naval Research, Coastal Geosciences Program through grant N00014-10-1-0088.

Abstract

The report documents a new version of the fully nonlinear Boussinesq wave model (FUNWAVE) initially developed by Kirby et al. (1998). The development of the present version was motivated by recent needs for modeling of surfzone-scale optical properties in a Boussinesq model framework, and modeling of Tsunami wave in both a regional/coastal scale for prediction of coastal inundation and a basin scale for wave propagation. This version features several theoretical and numerical improvements, including 1) a more complete set of fully nonlinear Boussinesq equations; 2) MUSCL-TVD solver with adaptive Runge-Kutta time stepping; 3) Shock-capturing wave breaking scheme; 4) wetting-drying moving boundary condition with incorporation of HLL construction method into the scheme; 5) an option for parallel computation. The documentation provides derivations of the conservation form of theoretical equations, re-arrangement of pressure gradient term in order to obtain a numerically well-balanced form, detailed numerical schemes, users' manual and examples.

Contents

. 1

1 Introduction **1**

2 Theory **3**
2.1 Governing equations 3
2.2 Treatment of surface elevation gradient term 4
2.3 Conservation forms of the fully nonlinear Boussinesq equations 5

3 Numerical schemes **7**
3.1 Compact forms of governing equations 7
3.2 Spatial discretization 8
3.3 Time stepping 9
3.4 Wave breaking and wetting-drying schemes for shallow water 10
3.5 Boundary conditions and wavemaker 10
3.6 Parallelization 11
3.7 Implementation of weakly nonlinear Boussinesq equations in spherical coordinates 12

4 Users' Manual **15**
4.1 Program outline and flow chart 15
4.2 Subroutine and function descriptions 15
4.3 Permanent variables 20
4.4 Installation and Compilation 22
4.5 Input 23
4.6 Input for the spherical code 29
4.7 Output 29

5 Benchmark Tests **29**
5.1 Basic hydrodynamic considerations 29
5.1.1 Mass conservation 29
5.1.2 Convergence 30
5.2 Analytical benchmarking 31
5.2.1 Single wave on a simple beach 31
5.2.2 Solitary wave on composite beach 34
5.3 Laboratory benchmarking 37
5.3.1 Solitary wave on a simple beach 37
5.3.2 Solitary wave on a composite beach 38
5.3.3 Solitary wave on a conical island 38
5.4 Field benchmarking 46
5.4.1 Tsunami runup onto a complex three-dimensional beach, Monai Valley . . 46

1 Introduction

The Boussinesq wave model has been a useful tool for modeling surface waves from deep water to the swash zone, as well as wave-induced circulations inside the surfzone. FUNWAVE, the fully nonlinear Boussinesq model developed by the University of Delaware group (Kirby et al.,1998), has come into fairly wide usage in the coastal community. FUNWAVE was based on the fully nonlinear Boussenesq equations derived by Wei et al. (1995) and used the high-order finite difference numerical method.

Since the initial version of FUNWAVE was developed, there were several updates in both theory and numerics for the fully nonlinear Boussinesq model. Gobbi et al. (2000) extended Wei et al. (1995) to higher order, which improved predictions of near-bed kinematics in deeper water. Chen (2006) pointed out missing terms in Wei et al. (1995) which represent the vertical vorticity on a sloping bed. Kennedy et al. (2001) introduced a concept of reference elevation in the derivation of the extended Boussinesq equations to imporve nonlinear performance. Shi et al. (2001) extended the fully nonlinear Boussinesq equations into a non-Cartesian coordinate system.

In the aspect of numerics, Wei and Kirby (1995) initially described a numerical scheme in which time stepping is treated using a fourth-order Adams-Bashforth-Moulton scheme, while spatial differencing is handled using a mixed-order scheme, employing fourth-order accurate centered differences for first derivatives and second-order accurate derivatives for third derivatives. The choice is made in order to move leading truncation errors to one order higher than the $O(\mu^2)$ dispersive terms, while maintaining the tridiagonal structure of spatial derivatives within time-derivative terms. A non-staggered grid system was used in Wei and Kirby (1995). Shi et al (2001) used similar numerical schemes but a staggered grid approach which has less apparent sensitivity to treatment of boundary conditions. The staggered grid scheme has become the preferred approach in later developments of the Delaware Boussinesq models such as in FUNWAVE 2 (Long and Kirby, 2006).

The moving shoreline condition in FUNWAVE was treated using a so-called slot technique (Kennedy et al., 2000, Chen et al., 2000). In the slot method, deep and narrow slots are added to each grid row, extending down at least to the lowest elevation that will be experienced during shoreface rundown. Recent model tests on the solitary wave runup measured at Oregon State University's O.H. Hinsdale Wave Research Laboratory (ISEC/NEES Workshop, Oregon State University, July, 2009) have raised a concern about considerable errors induced by the slot method. Slots which are too wide relative to the model grid spacing admit too much fluid before filling during runup, and cause both a reduction in amplitude and a phase lag in modeled runup events. At the other extreme, slots which are too narrow tend to induce a great deal of numerical noise, leading to the need for intermittent or even fairly frequent filtering of swash zone solutions.

Wave breaking in FUNWAVE was approached with the eddy viscosity method by Kennedy et al. (2000) following an early eddy viscosity model by Zelt (1991). Kennedy et al. used a model which involves a time history in order to allow the slope of the breaking wave crest to relax after the onset of breaking. Similar approaches were used by other Boussinesq model developers such as Nwogu and Demirbilek (2001) who used a more sophisticated eddy viscosity model in which

the eddy viscosity is expressed in terms of turbulent kinetic energy and a length scale.

Recent progress in the development of Boussinesq-type wave models using a hybrid method combining the finite-volume and finite-difference TVD-type schemes (Toro, 2009) have shown robust performance of the shock-capturing method in simulating breaking waves and coastal inundation (Tonelli and Petti, 2009, Roeber et al., 2010, Shiach and Mingham, 2009, Erduran et al., 2005, and others). The shock-capturing scheme makes the treatment of wave breaking straightforward without an artificial viscosity adopted in some breaking wave models such as in Kennedy et al. (2000). The scheme is also able to capture the sharp wave front occurring in the swash zone. The combined finite-volume and finite difference scheme also makes it easy to implement a wetting-drying moving shoreline condition. Recent applications of using such a wetting-drying method have shown to be quite accurate in modeling of wave runup (e.g., Lynett et al., 2002).

This work was motivated by recent needs to model the surfzone and swash zone dynamics and associated breaking wave-induced processes such as optical properties and sediment transport in a 10 km-scale computational domain, and to model tsunami waves in both a regional/coastal scale for prediction of coastal inundation and a basin scale for wave propagation. We are pursuing a version which is stable and robust in model efficiency and accuracy in a long time wave simulation in a large computational domain.

In this version, we started with a more complete set of fully nonlinear Boussinesq equations combined with equations of Wei et al. (1995) and some terms for the vertical vorticity correction reported by Chen (2006). A datum invariant form of the equations introduced by Kennedy et al. (2001) was applied, which is more consistent with a time-varying representation of elevation at a moving shoreline in modeling of a swash zone dynamics and coastal inundation. A conservative form of the equations was derived in order to use a hybrid numerical scheme. Dispersive terms were reorganized with an aim to construct the tridiagonal structure of spatial derivatives within time-derivative terms. The surface elevation gradient term was also rearranged to obtain a numerically well-balanced form which is suitable for any numerical order. Different from the previous high-order temporal scheme which usually requires uniform time-stepping, we used adaptive time stepping using the third-order Runge-Kutta method. The spatial derivatives were discritized using a combination of finite-volume and finite-difference methods. A fourth-order MUSCL reconstruction technique was used in the Riemann solver. The wave breaking scheme followed the approach of Tonelli and Petti (2009) who used the ability of the nonlinear shallow water equations with a TVD solver to simulate moving hydraulic jumps. Wave breaking is modeled by switching Boussinesq to NSWE at cells where the Froude number exceeds an certain threshold. The wetting-drying scheme was used to model a moving shoreline.

The model was parallelized using the domain decomposition technique. The Message Passing Interface (MPI) with non-blocking communication is used for data communication between processors. This report provides derivations of the conservation form of theoretical equations with a well-balanced pressure gradient term, numerical schemes, and users' manual. The last part of report illustrates the model's applications to problems of wave breaking and runup in the context of a standard suite of benchmark tests. In addition, we include a brief documentation of the spherical Boussinesq model used for Tsunami wave simulations. The spherical Boussinesq model was based

on Kirby (2004) and implemented in the same model framework. A detailed documentation for the spherical code will be reported separately.

2 Theory

2.1 Governing equations

Chen (2006) derived fully nonlinear Boussinesq-type equations for waves and currents over porous beds. The equations for the fluid layer are actually equations of Wei et al. (1995) with additional terms representing the second-order effect of the vertical vorticity. In this study, a time-varying reference elevation introduced by Kennedy et al. (2001) is applied to equations of Chen (2006).

The mass equation can be written as

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \mathbf{M} = 0, \tag{1}$$

where

$$\begin{aligned} \mathbf{M} = \quad & (h+\eta)\left\{\mathbf{u}_\alpha + \left[\left(\frac{z_\alpha^2}{2} - \frac{1}{6}(h^2 - h\eta + \eta^2)\right)\nabla(\nabla \cdot \mathbf{u}_\alpha)\right.\right. \\ & \left.\left. + \left(z_\alpha + \frac{1}{2}(h-\eta)\right)\nabla(\nabla \cdot (h\mathbf{u}_\alpha))\right]\right\}. \end{aligned} \tag{2}$$

$\mathbf{u}_\alpha$ represents horizontal velocity vector at a time-varying reference elevation

$$z_\alpha = -h + \alpha(h+\eta) \tag{3}$$

in which h is still water depth, η is the surface elevation, $\alpha = \sqrt{1/5}$ for Padé [2,2] dispersion (Note that Kennedy et al. used the symbol β to distinguish it from α defined by Nwogu, 1993). The reference elevation will always be a fixed fraction of the water column from the bed, which is an advantage in situations where water levels change significantly over time (Kennedy et al., 2001).

The momentum equation can be written as

$$\frac{\partial \mathbf{u}_\alpha}{\partial t} + (\mathbf{u}_\alpha \cdot \nabla)\mathbf{u}_\alpha + g\nabla\eta + \mathbf{V}_1 + \mathbf{V}_2 + \mathbf{V}_3 + \mathbf{R} = 0 \tag{4}$$

where g is the gravitational acceleration, $\mathbf{R}$ represents diffusion terms including bottom friction and subgrid lateral turbulent mixing. $\mathbf{V}_1$ and $\mathbf{V}_2$ are terms representing the dispersive Boussinesq terms given by

$$\begin{aligned} \mathbf{V}_1 = \quad & \left\{\frac{z_\alpha^2}{2}\nabla(\nabla \cdot \mathbf{u}_\alpha) + z_\alpha \nabla[\nabla \cdot (h\mathbf{u}_\alpha)]\right\}_t \\ & -\nabla\left[\frac{\eta^2}{2}\nabla \cdot \mathbf{u}_{\alpha,t} + \eta\nabla \cdot (h\mathbf{u}_{\alpha,t})\right] \end{aligned} \tag{5}$$

$$\mathbf{V}_2 = \nabla\left\{(z_\alpha-\eta)(\mathbf{u}_\alpha\cdot\nabla)[\nabla\cdot(h\mathbf{u}_\alpha)]+\frac{1}{2}(z_\alpha^2-\eta^2)(\mathbf{u}_\alpha\cdot\nabla)(\nabla\cdot\mathbf{u}_\alpha)\right\} + \frac{1}{2}\nabla\left\{[\nabla\cdot(h\mathbf{u}_\alpha)+\eta\nabla\cdot\mathbf{u}_\alpha]^2\right\}. \quad (6)$$

Note that, when the time-varying reference is applied, the only difference between (5) and the original expression of Wei et al. (1995) is in the first part of (5), where the temporal derivative originally found only as $\mathbf{u}_{\alpha,t}$ encompasses the entire term. $\mathbf{V}_3$ in (4) represents additional terms accounting for the second-order effect of the vertical vorticity (Chen, 2006) and may be written as

$$\mathbf{V}_3 = (U_3, V_3) \quad (7)$$

in which

$$U_3 = -v_\alpha\omega_1-\omega_0\left\{\left[z_\alpha-\frac{1}{2}(\eta-h)\right]\frac{\partial}{\partial y}[\nabla\cdot(h\mathbf{u}_\alpha)] + \left[\frac{1}{2}z_\alpha^2-\frac{1}{6}(\eta^2-\eta h+h^2)\right]\frac{\partial}{\partial y}(\nabla\cdot\mathbf{u}_\alpha)\right\} \quad (8)$$

$$V_3 = -u_\alpha\omega_1-\omega_0\left\{\left[z_\alpha-\frac{1}{2}(\eta-h)\right]\frac{\partial}{\partial x}[\nabla\cdot(h\mathbf{u}_\alpha)] + \left[\frac{1}{2}z_\alpha^2-\frac{1}{6}(\eta^2-\eta h+h^2)\right]\frac{\partial}{\partial x}(\nabla\cdot\mathbf{u}_\alpha)\right\} \quad (9)$$

In (8) and (9), ω_0 and ω_1 represent the vertical vorticity in the first- and second-order, respectively. They may be expressed by

$$\omega_0 = \frac{\partial v_\alpha}{\partial x}-\frac{\partial u_\alpha}{\partial y} \quad (10)$$

and

$$\omega_1 = \frac{\partial z_\alpha}{\partial x}\left\{\frac{\partial}{\partial y}[\nabla\cdot(h\mathbf{u}_\alpha)]+z_\alpha\frac{\partial}{\partial y}(\nabla\cdot\mathbf{u}_\alpha)\right\} - \frac{\partial z_\alpha}{\partial y}\left\{\frac{\partial}{\partial x}[\nabla\cdot(h\mathbf{u}_\alpha)]+z_\alpha\frac{\partial}{\partial x}(\nabla\cdot\mathbf{u}_\alpha)\right\} \quad (11)$$

2.2 Treatment of surface elevation gradient term

The hybrid numerical scheme requires a conservative form of continuity equation and momentum equations, resulting in the source terms associated with bottom slope in the momentum equations such as

$$g(h+\eta)\nabla\eta = \nabla\left(\frac{1}{2}gh^2\right)-g\eta\nabla z_b \quad (12)$$

where z_b is the bottom elevation from the datum. The numerical imbalance problem occurs when the surface gradient term is conventionally split into an artificial flux gradient and a source term that includes the effect of the bed slope for a non-uniform bed. To eliminate errors introduced by the traditional depth gradient method (DGM), a so-called surface gradient method (SGM) proposed by Zhou et al. (2001) was adopted in the TVD based-Boussinesq models in the recent literatures. Zhou et al. discussed an example of SGM in 1-D and verified that the slope-source term may be canceled out by part of the numerical flux term associated with water depth, if the bottom elevation at the cell center is constructed using the average of bottom elevations at two cell interfaces. Zhou et al. also showed a 2D application but without explicitly describing 2D numerical schemes. Although this scheme can be extended into 2D following the same procedure as in 1D, it was found that the 2D extension may not be trivial in terms of the bottom construction for a 2D arbitrary bathymetry. Kim et al. (2008) addressed the problem raised from general applications. They pointed out that the water depth in the slope-source term should be written in a discretized form rather than the value obtained using the bottom construction, implying that their revised SGM is valid for general 2D applications.

For the higher-order schemes such as the fourth-order TVD-MUSCL scheme (Yamamoto and Daiguji, 1993, Yamamoto et al., 1998) used in the recent Boussinesq applications, the original SGM and the revised SGM may not be effective in removing the artificial source. This problem was noticed by some recent authors such as Roeber et al. (2010) who kept a first-order scheme (second-order for normal conditions) for the numerical flux term and the slope-source term in order to ensure well-balanced solution without adding noise for a rapidly varying bathymetry.

In fact, the imbalance problem can be solved by a reformulation of this term in terms of deviations away from an unforced but separately specified equilibrium state (see general derivations in Rogers et al., 2003 and recent application in Liang and Marche, 2009). Using this technique, the surface gradient term may be split into

$$g(h+\eta)\nabla\eta = \nabla\left[\frac{1}{2}g(\eta^2+2h\eta)\right] - g\eta\nabla h \tag{13}$$

which is well-balanced for any numerical order under an unforced condition.

2.3 Conservation forms of the fully nonlinear Boussinesq equations

For Nwogu's (1993) and Wei et al's (1995) equations, $(h+\eta)u_\alpha$ can be used as a conserved variable in the construction of a conservative form of Boussinesq equations, but resulting in a source term in the mass conservation equation such as in Shiach and Mingham (2009) and Roeber et al. (2010). An alternative approach is to use $\mathbf{M}$ as a conserved variable in terms of the physical meaning of mass conservation. In this study, we used $\mathbf{M}$, instead of $(h+\eta)u_\alpha$, in the following derivations of the conservative form of the fully nonlinear Boussinesq equations.

We split $\mathbf{M}$ into two parts as follow

$$\mathbf{M} = (h+\eta)(\mathbf{u}_\alpha + \mathbf{V}_4) \tag{14}$$

where

$$\mathbf{V}_4 = \left(\frac{z_\alpha^2}{2} - \frac{1}{6}(h^2 - h\eta + \eta^2)\right)\nabla(\nabla\cdot\mathbf{u}_\alpha) + \left(z_\alpha + \frac{1}{2}(h-\eta)\right)\nabla(\nabla\cdot(h\mathbf{u}_\alpha)) \tag{15}$$

Using the tensor operator

$$\nabla\cdot(\mathbf{uv}) = \nabla\mathbf{u}\cdot\mathbf{v} + (\nabla\cdot\mathbf{v})\mathbf{u} \tag{16}$$

yields

$$\frac{\partial\mathbf{M}}{\partial t} + \nabla\cdot\left(\frac{1}{h+\eta}\mathbf{MM}\right) = (h+\eta)\frac{\partial\mathbf{u}_\alpha}{\partial t} + (h+\eta)\mathbf{u}_\alpha\cdot\nabla\mathbf{u}_\alpha + (h+\eta)\left(\frac{\partial\mathbf{V}_4}{\partial t} + \mathbf{u}_\alpha\cdot\nabla\mathbf{V}_4 + \mathbf{V}_4\cdot\nabla\mathbf{u}_\alpha\right). \tag{17}$$

Following Wei et al. (2005), we separate the time derivative dispersion terms in $\mathbf{V}_1$:

$$\mathbf{V}_1 = \mathbf{V}_{1,t}' + \mathbf{V}_1'' \tag{18}$$

where

$$\mathbf{V}_1' = \frac{z_\alpha^2}{2}\nabla(\nabla\cdot\mathbf{u}_\alpha) + z_\alpha\nabla[\nabla\cdot(h\mathbf{u}_\alpha)] - \nabla\left[\frac{\eta^2}{2}\nabla\cdot\mathbf{u}_\alpha + \eta\nabla\cdot(h\mathbf{u}_\alpha)\right] \tag{19}$$

and

$$\mathbf{V}_1'' = \nabla\left[\eta\eta_t\nabla\cdot\mathbf{u}_\alpha + \eta_t\nabla\cdot(h\mathbf{u}_\alpha)\right] \tag{20}$$

Using (17) and (18), the mass and momentum equations can be rewritten as

$$\frac{\partial\eta}{\partial t} + \nabla\cdot\mathbf{M} = 0. \tag{21}$$

$$\frac{\partial\mathbf{M}}{\partial t} + \nabla\cdot\left[\frac{1}{h+\eta}\mathbf{MM}\right] + \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] = (h+\eta)\left(\frac{\partial\mathbf{V}_4}{\partial t} + \mathbf{u}_\alpha\cdot\nabla\mathbf{V}_4 + \mathbf{V}_4\cdot\nabla\mathbf{u}_\alpha - \frac{\partial\mathbf{V}_1'}{\partial t} - \mathbf{V}_1'' - \mathbf{V}_2 - \mathbf{V}_3\right) + g\eta\nabla h + (h+\eta)\mathbf{R} \tag{22}$$

A difficulty usually arises in applying the adaptive time-stepping scheme to the time derivative dispersive terms ($\mathbf{V}_{4,t}$ and $\mathbf{V}'_{1,t}$) which are usually calculated by values stored in several time levels in the previous Boussinesq codesl such as in Wei et al. (1995) and Shi et al. (2001). The equation can be re-arranged by merging the time derivatives on the right hand side into the time derivative term on the left hand side:

$$\begin{aligned}&\frac{\partial \mathbf{V}}{\partial t}+\nabla\cdot\left[\frac{1}{h+\eta}\mathbf{MM}\right]+\nabla\left[\frac{1}{2}g(\eta^2+2h\eta)\right]\\&=\eta_t(\mathbf{V}'_1-\mathbf{V}_4)+(h+\eta)\left(\mathbf{u}_\alpha\cdot\nabla\mathbf{V}_4+\mathbf{V}_4\cdot\nabla\mathbf{u}_\alpha-\mathbf{V}''_1-\mathbf{V}_2-\mathbf{V}_3\right)\\&+g\eta\nabla h+(h+\eta)\mathbf{R}\end{aligned} \tag{23}$$

where

$$\mathbf{V}=(h+\eta)(\mathbf{u}_\alpha+\mathbf{V}'_1) \tag{24}$$

η_t can be calculated explicitly using (21) as in Roeber et al. (2010). (21) and (23) are the governing equations solved in this study.

3 Numerical schemes

3.1 Compact forms of governing equations

We define

$$\begin{aligned}&\mathbf{u}_\alpha=(u,v),\\&\mathbf{M}=(P,Q)=\left[(h+\eta)u+(h+\eta)U_4,(h+\eta)v+(h+\eta)V_4\right],\\&\mathbf{V}'_1=(U'_1,V'_1),\\&\mathbf{V}''_1=(U''_1,V''_1),\\&\mathbf{V}_2=(U_2,V_2),\\&\mathbf{V}_4=(U_4,V_4),\\&\mathbf{V}=(U,V)=\left[(h+\eta)(u+U'_1),(h+\eta)(v+V'_1)\right].\end{aligned}$$

The generalized conservation form of Boussinesq equations can be written as

$$\frac{\partial \mathbf{\Psi}}{\partial t}+\nabla\cdot\mathbf{\Theta}(\mathbf{\Psi})=\mathbf{S} \tag{25}$$

where $\mathbf{\Psi}$ and $\mathbf{\Theta}(\mathbf{\Psi})$ are the vector of conserved variables and the flux vector function, respectively, and are given by

$$\mathbf{\Psi}=\begin{pmatrix}\eta\\U\\V\end{pmatrix},\qquad \mathbf{\Theta}=\begin{pmatrix}P\mathbf{i}+Q\mathbf{j}\\\left[\frac{P^2}{h+\eta}+\frac{1}{2}g(\eta^2+2\eta h)\right]\mathbf{i}+\frac{PQ}{h+\eta}\mathbf{j}\\\frac{PQ}{h+\eta}\mathbf{i}+\left[\frac{Q^2}{h+\eta}+\frac{1}{2}g(\eta^2+2\eta h)\right]\mathbf{j}\end{pmatrix}. \tag{26}$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ g\eta\frac{\partial h}{\partial x} + \psi_x + (h+\eta)R_x \\ g\eta\frac{\partial h}{\partial y} + \psi_y + (h+\eta)R_y \end{pmatrix}, \tag{27}$$

where

$$\psi_x = \eta_t(U_1' - U_4) + (h+\eta)\left(uU_{4,x} + vU_{4,y} + U_4 u_x + V_4 u_y - U_1'' - U_2 - U_3\right) \tag{28}$$

$$\psi_y = \eta_t(V_1' - V_4) + (h+\eta)\left(uV_{4,x} + vV_{4,y} + U_4 v_x + V_4 v_y - U_1'' - V_2 - V_3\right) \tag{29}$$

The expanded forms of $(U_1', V_1'), (U_1'', V_1''), (U_2, V_2), (U_3, V_3)$ and (U_4, V_4) can be found in Appendix A. For the term $\mathbf{R}$, the bottom stress is approximated using a quadratic friction equation. A Smagorinsky (1963)-like subgrid turbulent mixing algorithm is implemented following Chen et al. (1999).

3.2 Spatial discretization

A combined finite-volume and finite-difference method was applied to the spatial discretization. For the flux terms and the first-order derivative terms, a fourth-order TVD-MUSCL scheme (Yamamoto and Daiguji, 1993) is used to construct the interface values. For example, in x-direction:

$$\phi^L_{i+1/2} = \phi_i + \frac{1}{6}(\Delta^*\bar{\phi}_{i-1/2} + 2\Delta^*\tilde{\phi}_{i+1/2}), \tag{30}$$

$$\phi^R_{i+1/2} = \phi_{i+1} - \frac{1}{6}(2\Delta^*\bar{\phi}_{i+1/2} + \Delta^*\tilde{\phi}_{i+3/2}), \tag{31}$$

where

$$\Delta^*\bar{\phi}_{i-1/2} = \text{minmod}(\Delta^*\phi_{i-1/2}, b\Delta^*\phi_{i+1/2}), \tag{32}$$

$$\Delta^*\tilde{\phi}_{i+1/2} = \text{minmod}(\Delta^*\phi_{i+1/2}, b\Delta^*\phi_{i-1/2}), \tag{33}$$

$$\Delta^*\bar{\phi}_{i+1/2} = \text{minmod}(\Delta^*\phi_{i+1/2}, b\Delta^*\phi_{i+3/2}), \tag{34}$$

$$\Delta^*\tilde{\phi}_{i+3/2} = \text{minmod}(\Delta^*\phi_{i+3/2}, b\Delta^*\phi_{i+1/2}), \tag{35}$$

$$\Delta^*\phi_{i+1/2} = \Delta\phi_{i+1/2} - \frac{1}{6}\Delta^3\bar{\phi}_{i+1/2}, \tag{36}$$

$$\Delta^3\bar{\phi}_{i+1/2} = \Delta\bar{\phi}_{i-1/2} - 2\Delta\bar{\phi}_{i+1/2} + \Delta\bar{\phi}_{i+3/2}, \tag{37}$$

$$\Delta\bar{\phi}_{i-1/2} = \text{minmod}(\Delta\phi_{i-1/2}, b_1\Delta\phi_{i+1/2}, b_1\Delta\phi_{i+3/2}), \tag{38}$$

$$\Delta\bar{\phi}_{i+1/2} = \text{minmod}(\Delta\phi_{i+1/2}, b_1\Delta\phi_{i+3/2}, b_1\Delta\phi_{i-1/2}), \tag{39}$$

$$\Delta\bar{\phi}_{i+3/2} = \text{minmod}(\Delta\phi_{i+3/2}, b_1\Delta\phi_{i-1/2}, b_1\Delta\phi_{i+1/2}), \tag{40}$$

$$\text{minmod}(i,j) = \text{sign}(i)\max[0, \min\{|i|, \text{sign}(i)j\}], \tag{41}$$

$$\text{minmod}(i,j,k) = \text{sign}(i)\max[0, \min\{|i|, \text{sign}(i)j, \text{sign}(i)k\}], \tag{42}$$

in which the coefficients $b_1 = 2$ and $1 \leq b \leq 4$.

The numerical fluxes are computed using HLL approximate Riemann solver.

$$\Theta(\Psi^L, \Psi^R) = \begin{cases} \Theta(\Psi^L) & \text{if} s_L \geq 0 \\ \Theta^*(\Psi^L, \Psi^R) & \text{if} s_L < 0 < s_R \\ \Theta(\Psi^R) & \text{if} s_R \leq 0, \end{cases} \tag{43}$$

where

$$\Theta^*(\Psi^L, \Psi^R) = \frac{s_R\Theta(\Psi^L) - s_L\Theta(\Psi^R) + s_L s_R(\Psi^R - \Psi^L)}{s_R - s_L} \tag{44}$$

The wave speeds of the Riemann solver are given by

$$s_L = \min(\mathbf{V}^L \cdot \mathbf{n} - \sqrt{g(h+\eta)^L}, u_s - \sqrt{\phi_s}), \tag{45}$$

$$s_R = \max(\mathbf{V}^R \cdot \mathbf{n} + \sqrt{g(h+\eta)^R}, u_s + \sqrt{\phi_s}), \tag{46}$$

in which u_s and ϕ_s are estimated as

$$u_s = \frac{1}{2}(\mathbf{V}^L + \mathbf{V}^R) \cdot \mathbf{n} + \sqrt{g(\eta+h)^L} - \sqrt{g(\eta+h)^R} \tag{47}$$

$$\sqrt{\phi_s} = \frac{\sqrt{g(\eta+h)^L} + \sqrt{g(\eta+h)^R}}{2} + \frac{(\mathbf{V}^L - \mathbf{V}^R) \cdot \mathbf{n}}{4} \tag{48}$$

and $\mathbf{n}$ is the normalized side vector for a cell face.

Higher derivative terms in ϕ_x and ϕ_y were discretized using the central differential schemes at the cell centroids as in Wei et al. (2005). No discretization of dispersion terms at the cell interfaces is needed due to using M as a flux variable. The Surface Gradient Method (Zhou et al, 2001) was used to eliminate unphysical oscillations. Because the pressure gradient term is re-organized as in section 2.2, there is no imbalance issue for the high-order MUSCL scheme.

3.3 Time stepping

The third-order Strong Stability-Preserving (SSP) Runge-Kutta scheme for nonlinear spatial discretization (Gottlieb et al., 2001) was adopted in time stepping. The scheme is given by

$$\begin{aligned} \Psi^{(1)} &= \Psi^n + \Delta t(-\nabla \cdot \Theta(\Psi^n) + \mathbf{S}^{(1)}) \\ \Psi^{(2)} &= \frac{3}{4}\Psi^n + \frac{1}{4}\left[\Psi^{(1)} + \Delta t\left(-\nabla \cdot \Theta(\Psi^{(1)}) + \mathbf{S}^{(2)}\right)\right] \\ \Psi^{n+1} &= \frac{1}{3}\Psi^n + \frac{2}{3}\left[\Psi^{(2)} + \Delta t\left(-\nabla \cdot \Theta(\Psi^{(2)}) + \mathbf{S}^{n+1}\right)\right] \end{aligned} \tag{49}$$

in which Ψ^n represents Ψ value at time level n. $\Psi^{(1)}$ and $\Psi^{(2)}$ are values at intermediate stages in the Runge-Kutta integration. As Ψ is obtained at each intermediate step, the velocity (u, v) can

be solved by a system of tridiagonal matrix equation formed by (24). $\mathbf{S}$ need to be updated using (u, v, η) at the corresponding time step and an iteration is needed to achieve convergence.

A adaptive time step is chosen, following the Courant-Friedrichs-Lewy (CFL) criterion:

$$\Delta t = C\min\left(\min\frac{\Delta x}{|u_{i,j}| + \sqrt{g(h_{i,j}+\eta_{i,j})}}, \min\frac{\Delta y}{|u_{i,j}| + \sqrt{g(h_{i,j}+\eta_{i,j})}}\right) \tag{50}$$

where C is the Courant number and $C = 0.5$ was used in the following examples.

3.4 Wave breaking and wetting-drying schemes for shallow water

The wave breaking scheme follows the approach of Tonelli and Petti (2009) who successfully used the ability of NSWE with a TVD scheme to model moving hydraulic jumps. The fully nonlinear Boussinesq equations are switched to NSWE at cells where the Froude number exceeds an certain threshold. Following Tonelli and Petti, the ratio of wave height to total water depth is chosen to the criterion to switch from Boussinesq to NSWE. The threshold value is set to be 0.8 as suggested by Tonelli and Petti.

The wetting-drying scheme for modeling of a moving boundary is straightforward. The normal flux $\mathbf{n} \cdot \mathbf{M}$ at the cell interface of a dry cell is set to zero. A mirror boundary condition is applied to the fourth-order MUSCL-TVD scheme and discretization of dispersive terms in ψ_x, ψ_y at dry cells. It may be noted that the wave speed of the Riemann solver (45) and (46) for a dry cell are modified as

$$s_L = \mathbf{V}^L \cdot \mathbf{n} - \sqrt{g(h+\eta)^L}, \quad s_R = \mathbf{V}^L \cdot \mathbf{n} + 2\sqrt{g(h+\eta)^L} \quad \text{(right dry cell)} \tag{51}$$

and

$$s_L = \mathbf{V}^R \cdot \mathbf{n} - \sqrt{g(h+\eta)^R}, \quad s_R = \mathbf{V}^R \cdot \mathbf{n} + 2\sqrt{g(h+\eta)^R} \quad \text{(left dry cell)} \tag{52}$$

3.5 Boundary conditions and wavemaker

We implemented various boundary conditions including wall boundary condition, absorbing boundary condition following Kirby et al. (1998) and periodic boundary condition following Chen et al. (2003).

Wavemakers implemented in the study include Wei and Kirby's (1999) internal wavemakers for regular waves and irregular waves. For the irregular wavemaker, an extension was made to incorporate the alongshore periodicity into wave generation in order to eliminate a boundary effect on wave simulation. The technique exactly follows the strategy in Chen et al. (2003) who adjusted the distribution of wave directions in each frequency bin to obtain alongshore periodicity. This approach is effective in modeling of breaking wave-induced nearshore circulations such as alongshore currents and rip currents.

3.6 Parallelization

In parallelizing the computational model, we use the domain decomposition technique to subdivide the problem into multiple regions and assign each subdomain to a separate processor core. Each subdomain region contains an overlapping area of ghost cells with three rows deep, as dictated by the fourth order MUSCL-TVD scheme. The Message Passing Interface (MPI) with non-blocking communication is used to exchange the data in the overlapping region between neighboring processors. Velocity components are obtained from Equation (24) by solving tridiagonal matrices using parallel pipelining tridiagonal solver described in Naik et al. (1993).

To investigate performance of the parallel program, numerical simulations of an idealized case are tested with different number of processors (16, 24, 36, and 48 processors) of a linux cluster located at University of Delaware. The test case is set up in a numerical grid of 1800×1800. Figure 1 shows the model speedup versus number of processors. It can be seen that a linear speedup is obtained with a slight decrease with more processors due to the communication cost.

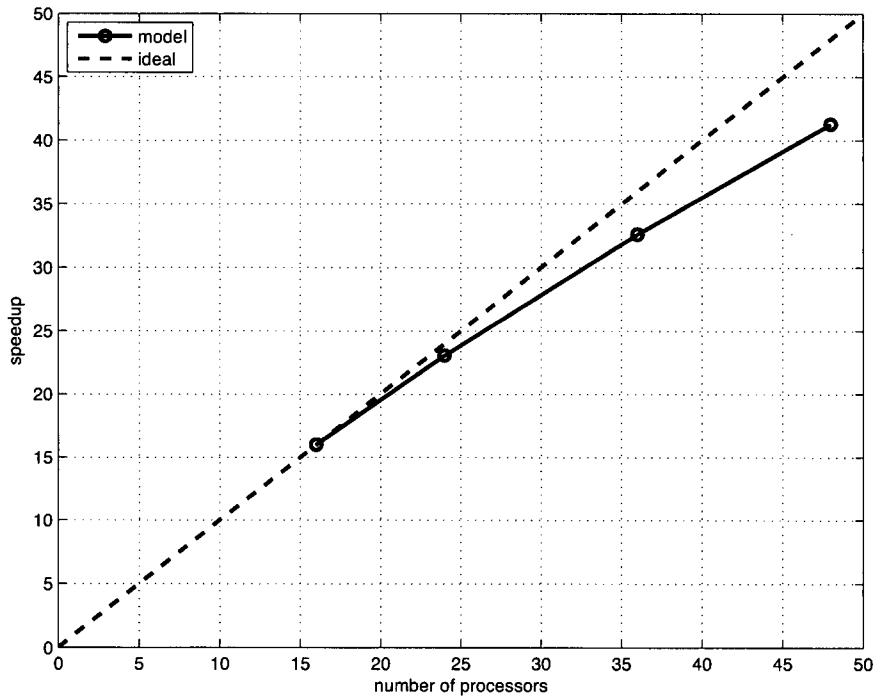

Figure 1: Speedup of the parallel version.

3.7 Implementation of weakly nonlinear Boussinesq equations in spherical coordinates

The weakly nonlinear Boussinesq equations in spherical coordinates are solved in the same model framework. We used the spherical Boussinesq equations derived by Kirby et al. (2004):

$$H_t + \frac{1}{r_0 \cos\theta}\{(Hu)_\phi + (Hv\cos\theta)_\theta\} = 0 \tag{53}$$

$$\begin{aligned} &u_t - fv + \frac{1}{r_0\cos\theta}uu_\phi + \frac{1}{r_0}vu_\theta + \frac{g}{r_0\cos\theta}\eta_\phi \\ &+\frac{1}{r_0^2\cos^2\theta}\left\{\frac{h^2}{6}\left[u_{\phi\phi t} + (v\cos\theta)_{\phi\theta t}\right] - \frac{h}{2}\left[(hu_t)_{\phi\phi} + (h\cos\theta v_t)_{\phi\theta}\right]\right\} \\ &-\tau_b^x + \frac{1}{r_0\cos\theta}(BFT)_\phi = 0 \end{aligned} \tag{54}$$

$$\begin{aligned} &v_t + fu + \frac{1}{r_0\cos\theta}uv_\phi + \frac{1}{r_0}vv_\theta + \frac{g}{r_0}\eta_\theta \\ &+\frac{1}{r_0^2}\left\{\frac{h^2}{6}\left[\frac{1}{\cos\theta}\{u_{\phi t} + (v\cos\theta)_{\theta t}\}\right]_\theta - \frac{h}{2}\left[\frac{1}{\cos\theta}\{(hu_t)_\phi + (h\cos\theta v_t)_\theta\}\right]_\theta\right\} \\ &-\tau_b^y + \frac{1}{r_0}(BFT)_\theta = 0 \end{aligned} \tag{55}$$

where θ and ϕ denote latitude and longitude, respectively, r_0 is the earth radius, f is the Coriolis parameter, $H = h + \eta$, (u, v) represent the depth-averaged velocity. BFT denotes forcing terms resulting from motion of the ocean bottom and it is not taken into account in the present program.

To facilitate solving the spherical equations in the same model framework as in the Cartesian coordinates, we define

$$x = r_0\cos\theta(\phi - \phi_0) \tag{56}$$

$$y = r_0(\theta - \theta_0) \tag{57}$$

where (ϕ_0, θ_0) are the reference longitude and latitude, respectively. The differential $(dx.dy)$ can be written as

$$dx = r\cos\theta d\phi \tag{58}$$

$$dy = rd\theta \tag{59}$$

$d\phi$ and $d\theta$ in (53) - (55) are replaced by dx and dy. Detailed derivations for each term are described below.

$$\frac{1}{r_0\cos\theta}(Hu)\phi = (Hu)_x \tag{60}$$

$$\frac{1}{r_0\cos\theta}(Hv\cos\theta)_\theta = \frac{1}{r_0\cos\theta}\left[\cos\theta(Hv)_\theta - Hv\sin\theta\right] = (Hv)_y - \frac{1}{r_0}\tan\theta Hv \tag{61}$$

$$\frac{1}{r_0\cos\theta}uu_\phi = uu_x \tag{62}$$

$$\frac{1}{r_0}vu_\theta = vu_y \tag{63}$$

$$\frac{1}{r_0\cos\theta}g\eta_\phi = g\eta_x \tag{64}$$

$$\frac{h^2}{6}\frac{1}{r_0^2\cos^2\theta}u_{\phi\phi t} = \frac{h^2}{6}u_{xxt} \tag{65}$$

$$\frac{h^2}{6}\frac{1}{r_0^2\cos^2\theta}(v\cos\theta)_{\phi\theta t} = \frac{h^2}{6}\left(v_{xy} - \frac{1}{r_0}\tan\theta v_x\right)_t \tag{66}$$

$$-\frac{h}{2}\frac{1}{r_0^2\cos^2\theta}(hu)_{\phi\phi t} = -\frac{h}{2}(hu)_{xxt} \tag{67}$$

$$-\frac{h}{2}\frac{1}{r_0^2\cos^2\theta}(hv\cos\theta)_{\phi\theta t} = -\frac{h}{2}\left[(hv)_{xy} - \frac{1}{r_0}\tan\theta(hv)_x\right]_t \tag{68}$$

$$\frac{1}{r_0\cos\theta}uv_\phi = uv_x \tag{69}$$

$$\frac{1}{r_0}vv_\theta = vv_y \tag{70}$$

$$\frac{1}{r_0}g\eta_\theta = g\eta_y \tag{71}$$

$$\frac{h^2}{6}\frac{1}{r_0^2}\left(\frac{u_{\phi t}}{\cos\theta}\right)_\theta = \frac{h^2}{6}\left(u_{xy} + \frac{1}{r_0}\tan\theta u_x\right)_t \tag{72}$$

$$-\frac{h}{2}\frac{1}{r_0^2}\left[\frac{(hu)_{\phi t}}{\cos\theta}\right]_\theta = -\frac{h}{2}\left[(hu)_{xy} + \frac{1}{r_0}\tan\theta(hu)_x\right]_t \tag{73}$$

$$\frac{h^2}{6}\frac{1}{r_0^2}\left[\frac{(v\cos\theta)_{\theta t}}{\cos\theta}\right]_\theta = \frac{h^2}{6}\left[v_{yy} - \frac{1}{r_0}\tan\theta v_y - \frac{1}{r_0^2\cos^2\theta}v\right]_t \tag{74}$$

$$-\frac{h}{2}\frac{1}{r_0^2}\left[\frac{(hv\cos\theta)_{\theta t}}{\cos\theta}\right]_\theta = -\frac{h}{2}\left[(hv)_{yy} - \frac{1}{r_0}\tan\theta(hv)_y - \frac{1}{r_0^2\cos^2\theta}hv\right]_t \tag{75}$$

The equations in (x, y) coordinates are

$$H_t + (Hu)_x + (Hv)_y = \frac{1}{r_0}\tan\theta Hv \tag{76}$$

$$u_t + uu_x + vu_y - fv + g\eta_x + F_t - \tau_b^x + (BFT)_x = 0 \tag{77}$$

$$v_t + uv_x + vv_y + fu + g\eta_y + G_t - \tau_b^y + (BFT)_y \tag{78}$$

where

$$F = \frac{h^2}{6}u_{xx} + \frac{h^2}{6}(v_{xy} - \frac{1}{r_0}\tan\theta v_x) - \frac{h}{2}(hu)_{xx} - \frac{h}{2}\left[(hv)_{xy} - \frac{1}{r_0}\tan\theta(hv)_x\right] \tag{79}$$

$$\begin{aligned} G = {} & \frac{h^2}{6}\left(u_{xy} + \frac{1}{r_0}\tan\theta u_x\right) + \frac{h^2}{6}\left(v_{yy} - \frac{1}{r_0}\tan\theta v_y - \frac{1}{r_0^2\cos^2\theta}v\right) \\ & -\frac{h}{2}\left[(hu)_{xy} + \frac{1}{r_0}\tan\theta(hu)_x\right] - \frac{h}{2}\left[(hv)_{yy} - \frac{1}{r_0}\tan\theta(hv)_y - \frac{1}{r_0^2\cos^2\theta}(hv)\right] \end{aligned} \tag{80}$$

The conservation forms of the spherical equations can be rewritten in the same form of the Cartesian equations (25) and $\mathbf{\Psi}$,$\mathbf{\Theta}$ and $\mathbf{S}$ are defined as

$$\mathbf{\Psi} = \begin{pmatrix} \eta \\ U_s \\ V_s \end{pmatrix}, \qquad \mathbf{\Theta} = \begin{pmatrix} P\mathbf{i} + Q\mathbf{j} \\ \left[\frac{P^2}{h+\eta} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{i} + \frac{PQ}{h+\eta}\mathbf{j} \\ \frac{PQ}{h+\eta}\mathbf{i} + \left[\frac{Q^2}{h+\eta} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{j} \end{pmatrix}. \tag{81}$$

$$\mathbf{S} = \begin{pmatrix} \frac{1}{r_0}\tan\theta(h+\eta)v \\ g\eta\frac{\partial h}{\partial x} + f(h+\eta)v + \tau_b^x + \psi_x \\ g\eta\frac{\partial h}{\partial y} - f(h+\eta)u + \tau_b^y + \psi_y \end{pmatrix}, \tag{82}$$

where $P = (h+\eta)u$, $Q = (h+\eta)v$,

$$U_s = (h+\eta)(u+F) \tag{83}$$

$$V_s = (h+\eta)(v+G) \tag{84}$$

$$\psi_x = \eta_t F \tag{85}$$

$$\psi_y = \eta_t G \tag{86}$$

4 Users' Manual

4.1 Program outline and flow chart

The code was written using Fortran 90 with the c preprocessor (cpp) statements for separation of the source code. Arrays are dynamically allocated at runtime. Precision is selected using the *selected_real_kind* Fortran intrinsic function defined in the makefile. The default precision is single.

The present version of FUNWAVE-TVD includes a number of options including (1) choice of serial or parallel code (2) Cartesian or spherical coordinate (Tsunami propagation mode), (3) samples, and (4) wave breaking index and aging (bubble and foam mode).

The flow chart is shown in Figure 2.

4.2 Subroutine and function descriptions

ALLOCATE_VARIABLES: allocate variables. It is called by MAIN.

BREAKING: print breaking index and breaking age calculated based on Kennedy et al. (2000). This option is defined in input.txt.

BOUNDARY_CONDITION: provide boundary conditions at four side boudaries. It is called by FLUXES.

CAL_DISPERSION: calculate dispersion terms. The first and second derivatives with respect to x, y are also calculated in this subroutine. In addition, the dispersion values in ghost cells are updated in this subroutine. It is called by the main program.

CALCULATE_Cm_Sm: calculate C_m and S_m used in Wei and Kirby's internal wave maker for irregular waves (TMA). Detailed formulation can be found in Shi et al. (2003). The subroutine is called by INITIALIZATION.

CALCULATE_SPONGE: setup sponge layer. It is called by INITIALIZATION.

CONSTRUCTION: low order interface construction. It is called by FLUXES. It calls CONSTRUCT_X and CONSTRUCT_Y.

CONSTRUCTION_HO: high order interface construction. It is called by FLUXES. It calls CONSTRUCT_HO_X and CONSTRUCT_HO_Y.

CONSTRUCT_HO_X: high order interface construction of specific variables in x direction. It is called by CONSTRUCTION_HO.

CONSTRUCT_HO_Y: high order interface construction of specific variables in y direction. It is called by CONSTRUCTION_HO.

CONSTRUCT_X: construct variables in x direction. It is called by CONSTRUCTION.

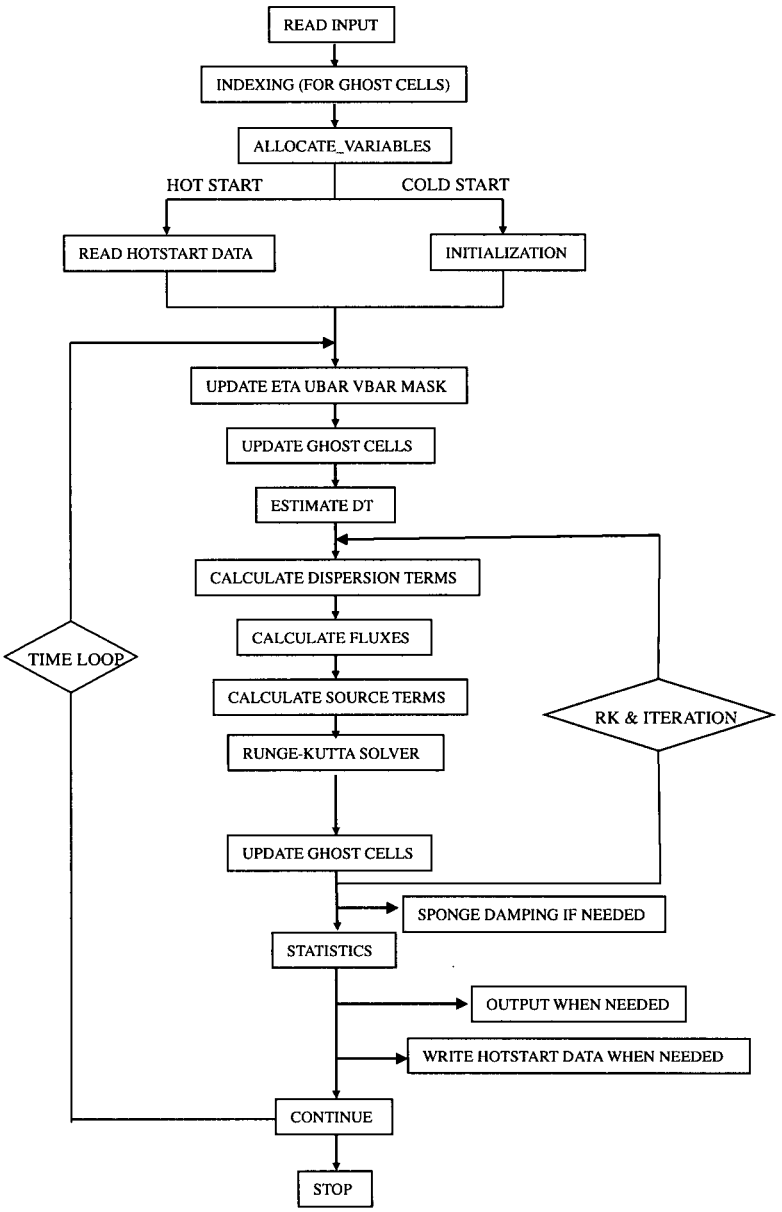

Figure 2: Flow chart of the main program.

CONSTRUCT_Y: construct variables in y direction. It is called by CONSTRUCTION.

CORRECTOR: corrector time scheme. It was used for a comparison between Predictor-corrector scheme and Runge-Kutta. It is not suggested using it in the present program.

DelxFun: calculate derivatives with respect to x. It is called by DelxFun.

DelxyFun: calculate derivatives with respect to x and y. It is called by FLUXES when the lower order construction is applied. It calls DelxFun and DelyFun.

DelyFun: calculate derivatives with respect to y. It is called by DelxyFun.

DERIVATIVE_X: calculate first-derivative respect to x in second order. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_X_High: calculate first-derivative respect to x in higher-order. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_XX: calculate second-derivative respect to x. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_XY: calculate cross-derivative. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_Y: calculate first-derivative respect to y in second order. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_Y_High: calculate first-derivative respect to y in higher-order. The subroutine is called by CAL_DISPERSION.

DERIVATIVE_YY: calculate second-derivative respect to y. The subroutine is called by CAL_DISPERSION.

ESTIMATE_DT: evaluate dt based on CFL criteria. It is called by MAIN.

ESTIMATE_HUV: Runge-Kutta solver. It is called by MAIN. It calls GET_Eta_U_V_HU_HV.

EXCHANGE: update η, u, v, Hu, Hv and MASK in the ghost cells. It is called by MAIN. It calls PHI_COLL.

EXCHANGE_DISPERSION: update dispersion variables in the ghost cells. It is called by MAIN. It calls PHI_COLL.

FLUX_AT_INTERFACE: calculate numerical fluxes at four cell interfaces using the averaging method. It is called by FLUXES.

FLUX_AT_INTERFACE_HLLC: calculate numerical fluxes at four cell interfaces using HLLC scheme. It is called by FLUXES. It calls HLLC.

FLUXES: calculate numerical fluxes. It is called by MAIN. It calls CONSTRUCTION lower order or CONSTRUCTION_HO for higher order, WAVE_SPEED, DelxyFun (for lower order), FLUX_AT_INTERFACE_HLLC for HLL scheme or FLUX_AT_INTERFACE for averaging scheme, and BOUNDARY_CONDITION.

GET_Eta_U_V_HU_HV: calculate η, u, v, Hu and Hv. It is called by ESTIMATE_HUV. The tridiagonal solver is used to get u and v. Froude number cap is applied in this subroutine. It calls TRIG (wall boundary) or TRIG_PERIODIC (periodic boundary). In the parallel mode, it calls TRIDx and TRIDy.

GetFile: read data in the global dimension and distribute to all processors in the parallel mode. Use flag '-DPARALLEL' in Makefile to include this subroutine.

GatherVariable: gather variables from all processors to processor 0 in the parallel mode but with serial tridiagonal solver. Use flag '-DTRID_NO_PARALLEL' to include this subroutine. It is called by TRIDxyNoParallel.

HLLC: HLLC scheme. It is called by FLUX_AT_INTERFACE_HLLC.

INDEX: indexing for ghost cells and MPI subdomains. It is called by MAIN.

INITIALIZATION: initialization subroutine. It is called MAIN. It may call WK_WAVEMAKER_REGULAR_WAVE, WK_WAVEMAKER_IRREGULAR_WAVE, CALCULATE_Cm_Sm, CALCULATE_SPONGE if applied.

INITIAL_HOTSTART: initialization when hotstart option is applied. It is called by MAIN. It may call INITIAL_UVZ, INITIAL_SOLITARY_WAVE, INITIAL_N_WAVE, INITIAL_RECTANGULAR, INITIAL_WAVE, WK_WAVEMAKER_REGULAR_WAVE, WK_WAVEMAKER_IRREGULAR_WAVE, CALCULATE_Cm_Sm, CALCULATE_SPONGE if applied.

INITIAL_N_WAVE: sample to provide initial N wave solution. Use flag '-DSAMPLE' in Makefile to include this subroutine. Parameters are defined in input.txt. The subroutine is called by INITIALIZATION.

INITIAL_SOLITARY_WAVE: sample to provide initial solitary wave solution. Use flag '-DSAMPLE' in Makefile to include this subroutine. Parameters are defined in input.txt. The subroutine is called by INITIALIZATION. It calls SUB_SLTRY to get parameters.

INITIAL_RECTANGULAR: sample of given an initial rectangular hump. Use flag '-DSAMPLE' in Makefile to include this subroutine. Parameters are defined in input.txt. The subroutine is called by INITIALIZATION.

INITIAL_UVZ: read initial u, v, and η data from files defined in input.txt. It is called by INITIALIZATION. It calls GetFile.

MINMOD_LIMITER: function of minmod limiter. It is used in CONSTRUCT_HO_X, and CONSTRUCT_HO_Y.

MPI_INITIAL: initialize MPI environment.

PHI_EXCH: handle float type data exchange between processors in the parallel mode. It is called by PHI_COLL. Use flag '-DPARALLEL' to include this subroutine.

PHI_INT_EXCH: handle integer type data exchange between processors in the parallel mode. It is called by UPDATE_MASK. Use flag '-DPARALLEL' to include this subroutine.

PREDICTOR: predictor scheme. It was used for a comparison between Predictor-corrector scheme and Runge-Kutta. It is not suggested using it in the present program.

PREVIEW: output subroutine. It is called by MAIN. It calls PutFile.

PutFile: print out output in files. In the parallel mode, it gathers data from all processors into processor 0 and prints out in the global dimension. Use flag '-DPARALLEL' in Makefile for the parallel mode. It is called by PREVIEW.

PHI_COLL: update data in ghost cells. It is called by EXCHANGE. in the parallel mode, it calls PHI_EXCH which is a major subroutine to handle data exchange between processors.

READ_INPUT: read input.txt. It is called by MAIN. It calls GET_STRING_VAL, GET_LOGICAL_VAL, GET_INTEGER_VAL, GET_Float_VAL. Input data are written out in LOG.txt.

READ_HOTSTART_DATA: read saved data for hot start and initialize other variables.

ScatterVariable: scatter variables from processor 0 to all processors in the parallel mode but with serial tridiagonal solver. Use flag '-DTRID_NO_PARALLEL' to include this subroutine. It is called by TRIDxyNoParallel.

SOLITARY_WAVE_LEFT_BOUNDARY: nudging boundary condition of solitary wave at left boundary. Use flag '-DSAMPLE' in Makefile to include this subroutine. Parameters are defined in input.txt.

SourceTerms: calculate all source terms including slope term and dispersion terms. It is called by MAIN.

STATISTICS: calculate statistics of total mass volume, energy, maximum and minimum η, u, v, Froude number etc.

SPONGE_DAMPING: use sponge layers to damp waves Parameters are defined in input.txt.

SUB_SLTRY: provide solitary wave solution of Nogu's equations. It is called by INITIAL_SOLITARY_WAVE. Use flag '-DSAMPLE' in Makefile to include this subroutine.

TRI_GE: tridiagonal solver. It allows diagonal variables not equal to unit. It is called by TRIG_PERIODIC.

TRIG: tridiagonal solver. It is called by GET_Eta_U_V_HU_HV. It calls TRI_GE.

TRIDxyNoParallel: no parallel tridiagonal solver but in the parallel mode. It is only for testing. Use flag '-DTRID_NO_PARALLEL' to include this subroutine. It calls GatherVariable, TRIG and ScatterVariable.

TRIG_PERIODIC: cyclic tridiagonal solver. It is called by GET_Eta_U_V_HU_HV. It calls TRI_GE.

UPDATE_MASK: update mask according to wetting and drying. Also update mask and mask9 in ghost cells.

VANLEER_LIMITER: function of Vanleer limiter. It is used in DelxFun and DelyFun.

WAVE_SPEED: calculate numerical wave speed needed by the TVD scheme. It is called by MAIN.

WK_WAVEMAKER_IRREGULAR_WAVE: calculate source function for Wei and Kirby's internal wave maker for irregular waves (TMA). Periodic boundary conditions are included. Parameters are defined in input.txt. The subroutine is called by INITIALIZATION.

WK_WAVEMAKER_REGULAR_WAVE: calculate source function for Wei and Kirby's internal wave maker for regular waves. Periodic boundary conditions are included but for certain wave angles (will ask during a run). Parameters are defined in input.txt. The subroutine is called by **INITIALIZATION**.

WRITE_HOTSTART_DATA: write out data used for hot start.

4.3 Permanent variables

Depth(): still water depth h at element point

DepthNode(): still water depth h at node

DepthX(): still water depth h at x-interface

DepthY(): still water depth h at y-interface

Eta(): surface elevation, for dry point, Eta() = MinDepth - Depth(), MinDepth is specified in input.txt.

Eta0(): η at previous time level

MASK(): 1 - wet, 0 - dry

MASK_STRUC(): 0 - permanent dry point

MASK9: mask to switch from Boussinesq equation to SWE, 1 - Boussinesq, 0 - SWE

U(): depth-averaged u or u at the reference level (u_α) at element

V(): depth-averaged v or v at the reference level (v_α) at element

HU(): $(h+\eta)u$ at element

HV(): $(h+\eta)v$ at element

P(): $(h+\eta)(u+U_4)$ at x-interface

Q(): $(h+\eta)(v+V_4)$ at y-interface

Fx(): numerical flux F at x-interface

Fy(): numerical flux F at y-interface

Gx(): numerical flux G at x-interface

Gy(): numerical flux G at y-interface

Ubar(): U

Vbar(): V

U4(): x-component of U_4

V4(): y-component of V_4

U1p(): x-component of U_1'

V1p(): y-component of V_1'

EtaRxL(): η Left value at x-interface

EtaRxR(): η Right value at x-interface

EtaRyL(): η Left value at y-interface

EtaRyR(): η Right value at y-interface

HxL(): total depth Left value at x-interface

HxR(): total depth Right value at x-interface

HyL(): total depth Left value at y-interface

HyR(): total depth Right value at y-interface

HUxL(): $(h+\eta)u$ Left value at x-interface

HUxR(): $(h+\eta)u$ Right value at x-interface

HVyL(): $(h+\eta)v$ Left value at y-interface

HVyR(): $(h+\eta)v$ Right value at y-interface

PL(): $(h+\eta)(u+U_4)$, Left value at x-interface

PR(): $(h+\eta)(u+U_4)$, Right value at x-interface

QL(): $(h+\eta)(v+V_4)$, Left value at y-interface

QR(): $(h+\eta)(v+V_4)$, Right value at y-interface

FxL = HUxL*UxL + 1/2*g*(EtaRxL^2 + 2*EtaRxL*Depthx)

FxR = HUxR*UxR + 1/2*g*(EtaRxR^2 + 2*EtaRxR*Depthx)

FyL = HyL*UyL*VyL

FyR = HyR*UyR*VyR

GxL = HxL*UxL*VxL

GxR = HxR*UxR*VxR

GyL = HVyL*VyL + 1/2*g*(EtaRyL^2 + 2*EtaRyL*Depthy)

GyR = HVyR*VyR + 1/2*g*(EtaRyR^2 + 2*EtaRyR*Depthy)

4.4 Installation and Compilation

FUNWAVE-TVD is distributed in a compressed fie. To install the programs, first, uncompress the package. Then use

> tar xvf *.tar

to extract files from the uncompressed package. The exacted files will be distributed in two new directories: /src and /work.

To compile the program, go to /src and modify Makefile if needed. There are several necessary flags in Makefile needed to specify below.

-DDOUBLE_PRECISION: use double precision, default is single precision.

-DPARALLEL: use parallel mode, default is serial mode.

-DSAMPLES: include all samples, default is no sample included.

-DINTEL: if INTEL compiler is used, this option can make use of FPORT for the RAND() function

CPP: path to CPP directory.

FC: Fortran compiler.

Then execute
> make
The executable file ''mytvd' will be generated and copied from /src to /work
To run the model, go to /work. Modify input.txt if needed and run.

4.5 Input

Following are descriptions of parameters in input.txt (**NOTE:** all parameter names are capital sensitive).

TITLE: title of your case, only used for log file.

SPECIFICATION OF HOT START

HOT_START: logical parameter, T for hot start, F for cold start.

FileNumber_HOTSTART: number of hotstart file used for a hot start, e.g., 1,2, ...

SPECIFICATION OF MULTI-PROCESSORS

PX: processor numbers in X

PY : processor numbers in Y
NOTE: PX and PY must be consistency with number of processors defined in mpirun command, e.g., mpirun -np n (where n = px×py)

SPECIFICATION OF WATER DEPTH

DEPTH_TYPE: depth input type.

DEPTH_TYPE=DATA: from a depth file.

The program includes several simple bathymetry configurations such as

DEPTH_TYPE=FLAT: flat bottom, need DEPTH_FLAT

DEPTH_TYPE=SLOPE: plane beach along x direction. It needs three parameters: slope,SLP, slope starting point, Xslp and flat part of depth, DEPTH_FLAT

DEPTH_FILE: bathymetry file if DEPTH_TYPE=DATA, file dimension should be Mglob x Nglob with the first point as the south-west corner. The read format in the code is shown below.

```
DO J=1,Nglob
READ(1,*)(Depth(I,J),I=1,Mglob)
ENDDO
```

DEPTH_FLAT: water depth of flat bottom if DEPTH_TYPE=FLAT or DEPTH_TYPE=SLOPE (flat part of a plane beach).

SLP: slope if DEPTH_TYPE=SLOPE

Xslp: starting x (m) of a slope, if DEPTH_TYPE=SLOPE

SPECIFICATION OF RESULT FOLDER

RESULT_FOLDER: result folder name, e.g., RESULT_FOLDER = /Users/fengyanshi/tmp/

SPECIFICATION OF DIMENSION

Mglob: global dimension in x direction.

Nglob: global dimension in y direction.

SPECIFICATION OF TIME

TOTAL_TIME: simulation time in seconds

PLOT_INTV: output interval in seconds (Note, output time is not exact because adaptive dt is used.)

SCREEN_INTV: time interval (s) of screen print.

PLOT_INTV_STATION: time interval (s) of gauge output

HOTSTART_INTV: time interval (s) to save hot start data.

SPECIFICATION OF GRID SIZE

DX: grid size(m) in x direction.

DY: grid size(m) in y direction.

SPECIFICATION OF INITIAL CONDITION

INT_UVZ : logical parameter for initial condition, default is FALSE

ETA_FILE: name of file for initial η, e.g., ETA_FILE= /Users/fengyanshi/work/input/CVV_H.grd, data format is the same as depth data.

U_FILE: name of file for initial u, e.g.,U_FILE= /Users/fengyanshi/work/input/CVV_U.grd, data format is the same as depth data.

V_FILE: name of file for initial v, e.g., V_FILE= /Users/fengyanshi/work/input/CVV_V.grd, data format is the same as depth data.

SPECIFICATION OF WAVEMAKER

WAVEMAKER: wavemaker type.

WAVEMAKER = INI_REC: initial rectangular hump, need Xc,Yc and WID

WAVEMAKER = LEF_SOL: left boundary solitary, need AMP,DEP, and LAGTIME

WAVEMAKER = INI_SOL: initial solitary wave, WKN B solution, need AMP, DEP, and XWAVEMAKER

WAVEMAKER = INI_OTH: other initial distribution specified by users

WAVEMAKER = WK_REG: Wei and Kirby 1999 internal wave maker, need Xc_WK, Tperiod, AMP_WK, DEP_WK, Theta_WK, and Time_ramp (factor of period)

WAVEMAKER = WK_IRR: Wei and Kirby 1999 TMA spectrum wavemaker, need Xc_WK, DEP_WK, Time_ramp, Delta_WK, FreqPeak, FreqMin,FreqMax, Hmo, GammaTMA, ThetaPeak

WAVEMAKER = WK_TIME_SERIES: *fft* a time series to get each wave component and then use Wei and Kirby's (1999) wavemaker. Need input WaveCompFile (including 3 columns: per,amp,pha) and NumWaveComp,PeakPeriod,DEP_WK, Xc_WK,Ywidth_WK

AMP: amplitude (m) of initial η, if WAVEMAKER = INI_REC, WAVEMAKER = INI_SOL, WAVEMAKER = LEF_SOL.

DEP: water depth at wavemaker location, if WAVEMAKER = INI_SOL, WAVEMAKER = LEF_SOL.

LAGTIME, time lag (s) for the solitary wave generated on the left boundary, e.g., WAVEMAKER = LEF_SOL.

XWAVEMAKER: x (m) coordinate for WAVEMAKER = INI_SOL.

Xc: x (m) coordinate of the center of a rectangular hump if WAVEMAKER = INI_REC.

Yc: y (m) coordinate of the center of a rectangular hump if WAVEMAKER = INI_REC.

WID: width (m) of a rectangular hump if WAVEMAKER = INI_REC.

Time_ramp: time ramp (s) for Wei and Kirby (1999) wavemaker.

Delta_WK: width parameter δ for Wei and Kirby (1999) wavemaker. $\delta = 0.3 \sim 0.6$

DEP_WK: water depth (m) for Wei and Kirby (1999) wavemaker.

Xc_WK: x coordinate (m) for Wei and Kirby (1999) wavemaker.

Ywidth_WK: width (m) in y direction for Wei and Kirby (1999) wavemaker.

Tperiod: period (s) of regular wave for Wei and Kirby (1999) wavemaker.

AMP_WK: amplitude (m) of regular wave for Wei and Kirby (1999) wavemaker.

Theta_WK: direction (degrees) of regular wave for Wei and Kirby (1999) wavemaker. Note: it may be adjusted for a periodic boundary case by the program. A warning will be given if adjustment is made.

FreqPeak: peak frequency (1/s) for Wei and Kirby (1999) irregular wavemaker.

FreqMin: low frequency cutoff (1/s) for Wei and Kirby (1999) irregular wavemaker.

FreqMax: high frequency cutoff (1/s) for Wei and Kirby (1999) irregular wavemaker.

Hmo: Hmo (m) for Wei and Kirby (1999) irregular wavemaker.

GammaTMA, TMA parameter γ for Wei and Kirby (1999) irregular wavemaker.

ThetaPeak: peak direction (degrees) for Wei and Kirby (1999) irregular wavemaker.

Sigma_Theta: parameter of directional spectrum for Wei and Kirby (1999) irregular wavemaker.

SPECIFICATION OF PERIODIC BOUNDARY CONDITION

(Note: only south-north periodic condition was implemented)

PERIODIC: logical parameter for periodic boundary condition, T - periodic, F - wall boundary condition.

SPECIFICATION OF SPONGE LAYER

SPONGE_ON: logical parameter, T - sponge layer, F - no sponge layer.

Sponge_west_width: width (m) of sponge layer at west boundary.

Sponge_east_width: width (m) of sponge layer at east boundary.

Sponge_south_width: width (m) of sponge layer at south boundary.

Sponge_north_width width (m) of sponge layer at north boundary

R_sponge: decay rate in sponge layer. Its values are between 0.85 $\sim$ 0.95.

A_sponge: maximum damping magnitude. The value is $\sim$ 5.0.

SPECIFICATION OF OBSTACLES

OBSTACLE_FILE: name of obstacle file. 1 - water point, 0 - permanent dry point. Data dimension is (Mglob $\times$ Nglob). Data format is the same as the depth data.

SPECIFICATION OF PHYSICS

DISPERSION: logical parameter for inclusion of dispersion terms. T - calculate dispersion, F - no dispersion terms

Gamma1: integer parameter for linear dispersive terms. 1 - inclusion of linear dispersive terms, 0 - no linear dispersive terms.

Gamma2: integer parameter for nonlinear dispersive terms. 1 - inclusion of nonlinear dispersive terms, 0 - no nonlinear dispersive terms.

Gamma1=1.0, Gamma2=0.0 for NG's equations. Gamma1=1.0, Gamma2=1.0 for the fully nonlinear Boussinesq equations.

Beta_ref: parameter β defined for the reference level. β = -0.531 for NG's and FUNWAVE equations.

SWE_ETA_DEP: ratio of height/depth for switching from Boussinesq to NSWE. The value is $\sim$ 0.80.

Cd: bottom friction coefficient.

SPECIFICATION OF NUMERICS

Time_Scheme: stepping option, Runge_Kutta or Predictor_Corrector (not suggested for this version).

HIGH_ORDER: spatial scheme option, FOURTH for fourth-order, SECOND for second-order (not suggested for Boussinesq model).

CONSTRUCTION: construction method, HLL for HLL scheme, otherwise for averaging scheme.

CFL: CFL number, CFL $\sim$ 0.5.

FroudeCap: cap for Froude number in velocity calculation for efficiency. The value could be 5 $\sim$ 10.0.

MinDepth: minimum water depth (m) for wetting and drying scheme. Suggestion: MinDepth = 0.001 for lab scale and 0.01 for field scale.

MinDepthFrc: minimum water depth (m) to limit bottom friction value. Suggestion: MinDepthFrc = 0.01 for lab scale and 0.1 for field scale.

SHOW_BREAKING: logical parameter to calculate breaking index. Note that breaking is calculated using shock wave capturing scheme. The index calculated here is based on Kennedy et al. (2000).

Cbrk1: parameter C1 in Kennedy et al. (2000).

Cbrk2: parameter C2 in Kennedy et al. (2000).

SPECIFICATION OF OUTPUT VARIABLES

NumberStations: number of station for output. If NumberStations¿0, need input i,j in STATION_FILE

DEPTH_OUT: logical parameter for output depth. T or F.

U: logical parameter for output u. T or F.

V: logical parameter for output v. T or F.

ETA: logical parameter for output η. T or F.

MASK: logical parameter for output wetting-drying MASK. T or F.

MASK9: logical parameter for output MASK9 (switch for Boussinesq/NSWE). T or F.

SourceX: logical parameter for output source terms in x direction. T or F.

SourceY: logical parameter for output source terms in y direction. T or F.

P: logical parameter for output of momentum flux in x direction. T or F.

Q: logical parameter for output of momentum flux in y direction. T or F.

Fx: logical parameter for output of numerical flux F in x direction. T or F.

Fy: logical parameter for output of numerical flux F in y direction. T or F.

Gx: logical parameter for output of numerical flux G in x direction. T or F.

Gy: logical parameter for output of numerical flux G in y direction. T or F.

AGE: logical parameter for output of breaking age. T or F.

4.6 Input for the spherical code

All input parameters, except the following grid information, are the same as for the Cartesican code.

Lon_West: longitude (degrees) of west boundary.

Lat_South: latitude (degrees) of south boundary.

Dphi: $d\phi$ (degrees)

Dtheta: $d\theta$ (degrees)

In addition, there is no Gamma2 (for nonlinear dispersive terms) in the spherical code.

4.7 Output

The output files are saved in the result directory defined by RESULT_FOLDER in input.txt. For outputs in ASCII, a file name is a combination of variable name and an output series number such eta_0001, eta_0002, ... The format and algorithm are consistent with the depth file. Output for stations is a series of numbered files such as sta_0001, sta_0002 ....

The NetCDF output format is under development.

5 Benchmark Tests

Benchmarking of this numerical model is classified into four categories: 1) Basic hydrodynamic considerations, 2) Analytical benchmarking, 3) Experimental benchmarking, and 4) Field benchmarking. Some of the benchmark problems described here were used as benchmark problems in the (Yeh et al.,1996) and (Liu et al.,2007) Long-Wave Runup Models Workshops in Friday Harbor, Washington, and Catalina Island, Los Angeles, California, respectively.

5.1 Basic hydrodynamic considerations

Two most basic steps are required in ensuring that this numerical model works for predicting evolution and inundation. The first step is ensuring that the model conserves mass; the second basic step is checking convergence of this numerical code to a certain asymptotic limit.

5.1.1 Mass conservation

Conservation of mass can be checked by calculating the water volume at the beginning and at the end of the computation. This should be done by integrating disturbed water depth $\eta(x, y, t)$ over the entire flow domain, i.e., if the flow domain extends from the maximum penetration during

inundation $x = X_{max}$ to the outer location of the source region X_s, and $y = Y_{max}$ to Y_s, then total displaced volume $V(t)$ is

$$V(t) = \int_{X_{max}}^{X_s} V(t) = \int_{Y_{max}}^{Y_s} \eta(x, y, t) dx dy \tag{87}$$

The integral of $\eta(x, y, t)$ should be used instead of the integral of the entire flow depth $h(x, y, t) = \eta(x, y, t) + d(x, yxt)$, where $d(x, y, t)$ is the undisturbed water depth, because the latter is likely to conceal errors in the calculation. Typically, $\eta << d$ at offshore integrating h will simply produce the entire volume of the flow domain and will mask errors. Note that testing of the conservation of mass as above involves placing a closed domain within reflective boundaries (Synolakis et al., 2007).

The code is able to calculate the whole mass of domain and write out the quantity of mass in LOG.txt for each screening time.

Calculations of conservation of mass has been done for all of the following benchmarks in the context such that the total initial displaced volume V (t = 0) was within less than 1% of the total displaced volume at the end of the computation $V(t = T)$ where T represents the computation end time. It is assumed that the end of the computation is when the initial wave is entirely reflected and offshore. However with few changes in Δx and Δy the conservation of mass can be improved.

5.1.2 Convergence

The next basic step that has been covered for all of the benchmarks in the following context was checking convergence of the numerical code to a certain asymptotic limit, presumably the actual solution of the equations solved. The grid steps Δx and Δy has been halved, and the time step Δt automatically reduced appropriately to conform to the CourantFriedricsLevy (CFL) criterion. Convergence has been checked through the extreme runup and rundown. Figure 3 and table 1 provides convergence of the code due to the conical island test problem which will be discussed in experimental benchmarks.

	Maximum Runup		
Grid Size	H/d=0.045	H/d=0.091	H/d=181
0.1	0.02302	0.04061	0.06311
0.05	0.02303	0.04063	0.06315
0.01	0.02303	0.04064	0.06316
0.005	0.02303	0.04064	0.06316

Tabel 1. Maximum runup for Gauge 9 for different grid size

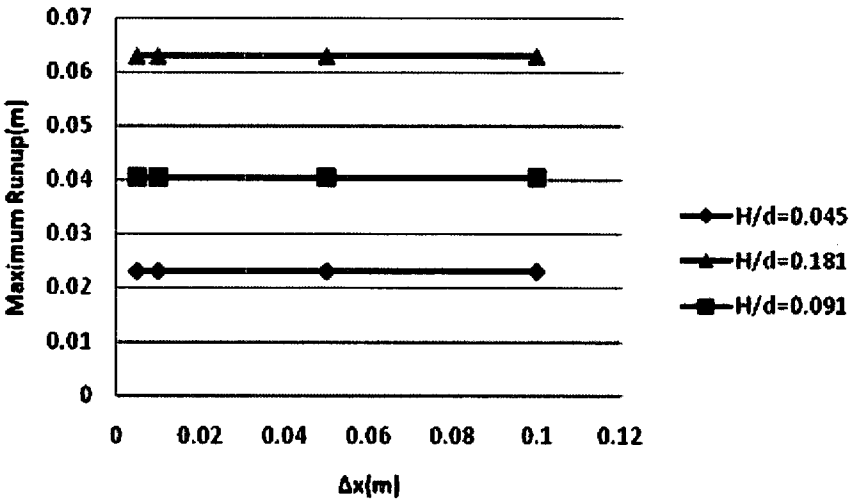

Figure 3: Maximum runup for Gauge 9 for different grid size

5.2 Analytical benchmarking

The shallow water-wave (SW) equations describe the evolution of the water surface elevation and of the depth-averaged water particle velocity of waves with wavelengths large compared with the depth of propagation. The equations assume that the pressure distribution is hydrostatic everywhere, i.e., there is no variation with depth of any of the other flow variables. In this section we present several analytic solutions to the 1+1 version of these equations. As stated in Section 2 of this report, 1+1 models are generally unreliable for inundation mapping and entirely inadequate for operational tsunami forecasting, but they are invaluable to the process of testing and validating models (Synolakis et al., 2007).

In this part following benchmark problems has been studied:

1) Single wave on a simple beach
2) Solitary wave on a composite beach

5.2.1 Single wave on a simple beach

The canonical problem of the shallow water-wave equations is covered here which contains the calculation of a long wave climbing up a sloping beach of angle β attached to a constant-depth region (Figure 4). The origin of the coordinate system is at the initial position of the shoreline and x increases seaward.

a) Solitary wave evolution and runup

It is possible to derive exact results for the evolution and runup of solitary waves based on linear theory (Synolakis, 1986, 1987).Solitary waves have long been used as a model for the leading wave

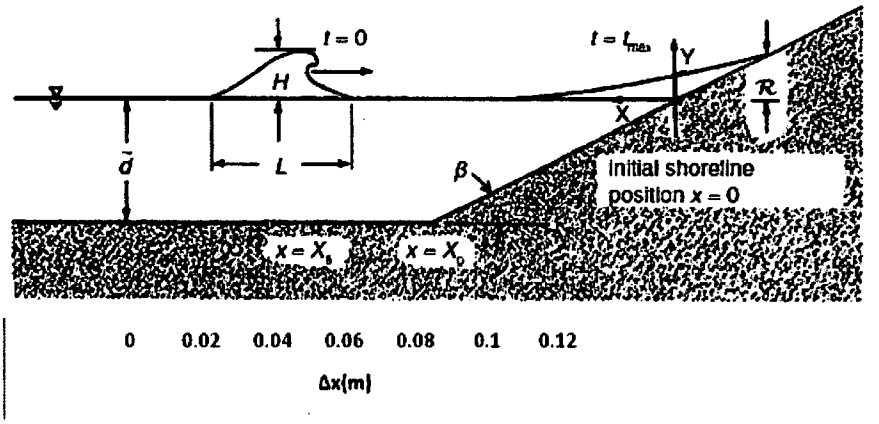

Figure 4: Definition sketch for bathymetry

of tsunamis. Russel (1845) defined solitary waves as the great waves of translation, and consists of a single elevation wave. While capturing some of the basic physics of tsunamis, solitary waves do not model the physical manifestation of tsunamis in nature, which are invariably N-wave like with a leading-depression wave followed by an elevation wave. (Synolakis et al., 2007).

The following expression for the maximum runup R has been provided based on slope of the beach and wave height of the solitary wave:

$$R = 2.831\sqrt{\cot\beta}H^{5/4} \tag{88}$$

The equation (88) is also known as runup law.

The code is able to model this case using the depth format SLOPE in which the slope of the beach, depth of flat region and starting point of the slope should be defined. The cases that have been studied here have different depths from 50cm to 1000m. Also, for each depth different slope and wave heights has been studied. Table 2 provides a list of selected different cases that has been modeled including their maximum runup and the grid size for each case. Figure 5 defines a comparison between numerical simulation and runup law.

b) N-wave runup

Most tsunami eyewitness accounts suggest that tsunamis are N-wave like, i.e., they are dipolar, which means they appear as a combination of a depression and an elevation wave, and frequently as a series of N-waves, sometimes known as double N-waves (Synolakis et al., 2007). Tadepalli and Synolakis (1994) described N-wave with leading-elevation and depression waves of the same height and at a constant separation distance and refer to this wave as an isosceles N- wave with a surface profile given by

				R/d		
d(m)	Δx(m)	H/d	Cot(B)	Runup Law	Numerical calculations	Error(%)
0.5	0.1	0.03	10	0.112	0.110	1.6
0.5	0.1	0.05	10	0.212	0.202	4.6
0.5	0.1	0.05	3.3333	0.122	0.130	6.2
0.5	0.1	0.1	3.3333	0.291	0.282	3.0
0.5	0.1	0.1	1	0.159	0.172	8.0
0.5	0.1	0.2	1	0.379	0.356	6.0
0.5	0.1	0.48	1	1.131	1.100	2.7
0.5	0.1	0.01	20	0.040	0.042	4.9
5	1.0	0.03	10	0.112	0.112	0.4
5	1.0	0.05	10	0.212	0.212	0.3
5	1.0	0.05	3.732	0.129	0.138	6.4
5	1.0	0.1	3.732	0.308	0.315	2.4
5	1.0	0.2	3.732	0.731	0.734	0.3
5	1.0	0.098	2.747	0.257	0.276	7.3
5	1.0	0.294	2.747	1.016	1.056	4.0
5	1.0	0.005	20	0.017	0.017	1.0
5	1.0	0.01	20	0.040	0.039	3.0
100	5.0	0.05	10	0.212	0.202	4.5
100	5.0	0.098	2.747	0.257	0.269	4.6
100	5.0	0.193	2.747	0.600	0.598	0.4
100	5.0	0.01	20	0.040	0.040	0.2
1000	20.0	0.03	10	0.112	0.120	7.4
1000	20.0	0.05	10	0.212	0.200	5.5
1000	20.0	0.05	3.3333	0.122	0.133	8.4
1000	20.0	0.1	3.3333	0.291	0.318	9.5

Tabel 2. Runup data from numerical calculations compared with runup law values.

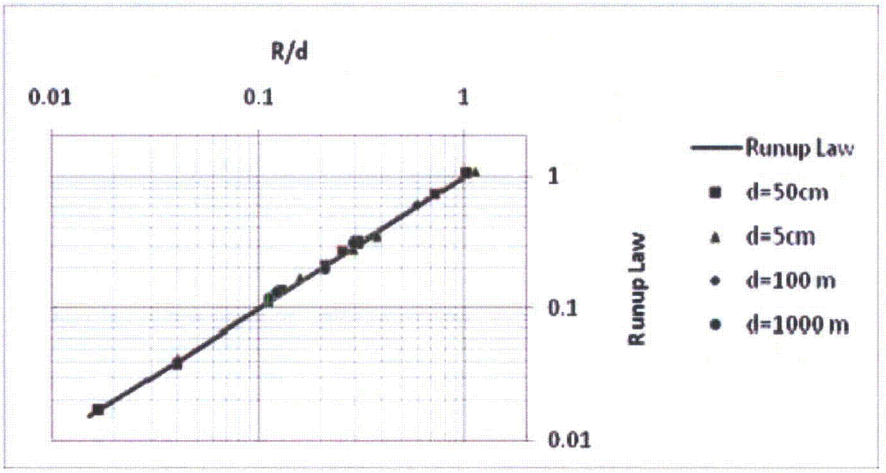

Figure 5: Numerical Simulation data for maximum runup of nonbreaking waves climbing up different beach slopes. Solid line represents the runup law.

$$\eta(x,0) = \frac{3\sqrt{3}H}{2} sech^2[\gamma(x - X_N)] \tanh[\gamma(x - X_N)] \tag{89}$$

where

$$\gamma = \frac{3}{2}\sqrt{\sqrt{\frac{3}{4}H}} \tag{90}$$

The code is able to generate N-waves using parameters H (wave height), X_N(distance from the boundary where the wave generates), and γ.

The following expression for the maximum runup of N-waves has been provided based on slope of the beach and wave height of the N-wave similar to solitary wave problem:

$$R = 3.86\sqrt{\cot\beta}H^{5/4} \tag{91}$$

The code is able to model this case using the depth format SLOPE with the same definitions mentioned for solitary wave problem. The cases that have been studied here have different depths from 50 cm to 1000 m. Also, for each depth different slope and wave heights has been studied. Table 3 provides a list of selected different cases that has been modeled including their maximum runup and the grid size for each case. Figure 6 defines a comparison between numerical simulation and runup law for N-wave problem.

5.2.2 Solitary wave on composite beach

In this benchmark a complex topography consisting of three segments and a vertical wall has been considered (Figure 7). Runup of non-breaking solitary waves on the vertical wall has been simu-

					R/d		
d(m)	γ	Δx(m)	H/d	Cot(B)	Runup Law	Numerical calculations	Error(%)
0.5	0.17	0.1	0.03	10	0.152	0.162	6.3
0.5	0.22	0.1	0.05	10	0.289	0.262	9.2
0.5	0.22	0.1	0.05	3.3333	0.167	0.165	1.0
0.5	0.31	0.1	0.1	3.3333	0.396	0.374	5.6
0.5	0.31	0.1	0.1	1	0.217	0.232	6.9
0.5	0.44	0.1	0.2	1	0.516	0.501	3.0
0.5	0.68	0.1	0.48	1	1.542	1.413	8.4
0.5	0.10	0.1	0.01	20	0.055	0.058	6.2
5	0.54	1.0	0.03	10	0.152	0.157	3.0
5	0.70	1.0	0.05	10	0.289	0.285	1.2
5	0.70	1.0	0.05	3.732	0.176	0.187	6.1
5	0.99	1.0	0.1	3.732	0.419	0.433	3.3
5	1.40	1.0	0.2	3.732	0.997	1.052	5.5
5	0.98	1.0	0.098	2.747	0.351	0.369	5.2
5	1.69	1.0	0.294	2.747	1.385	1.388	0.2
5	0.22	1.0	0.005	20	0.023	0.023	0.2
5	0.31	1.0	0.01	20	0.055	0.056	2.6
100	3.12	5.0	0.05	10	0.289	0.278	3.7
100	4.37	5.0	0.098	2.747	0.351	0.379	8.0
100	6.13	5.0	0.193	2.747	0.818	0.826	0.9
100	1.40	5.0	0.01	20	0.055	0.056	2.6
1000	7.65	20.0	0.03	10	0.152	0.163	7.0
1000	9.87	20.0	0.05	10	0.289	0.275	4.7
1000	9.87	20.0	0.05	3.3333	0.167	0.173	3.8
1000	13.96	20.0	0.1	3.3333	0.396	0.419	5.7

Tabel 2. Runup data from numerical calculations compared with runup law values for N-wave.

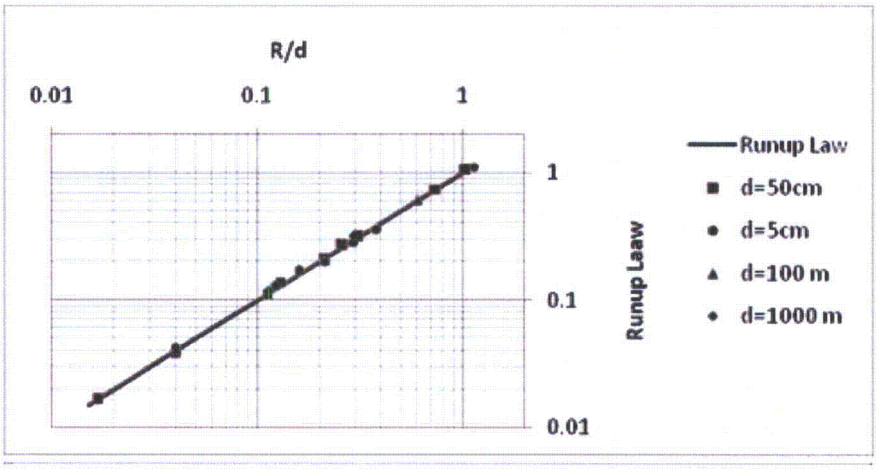

Figure 6: Numerical Simulation data for maximum runup of N-waves climbing up different beach slopes. Solid line represents the runup law.

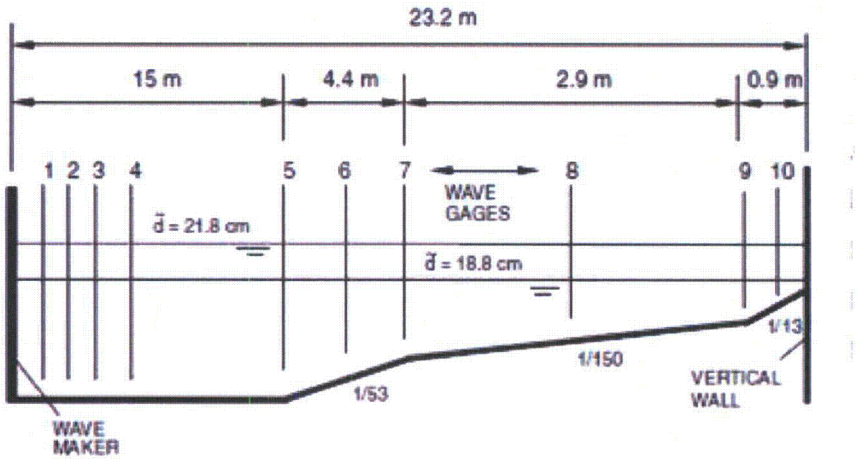

Figure 7: Definition sketch for the revere beach.

lated in this case and also, the results have been compared with the analytical solution. Laboratory data exist for this topography from a U.S. Army Corps of Engineers, Coastal Engineering Research Center, Vicksburg, Mississippi experiment of wave runup on a model of Revere Beach, Massachusetts. However, the maximum runup for solitary waves propagating up Revere Beach (Composite Beach) is given by the runup law

$$R = 2h_w^{-1/4}H \tag{92}$$

where, h_w is the initial depth at the right vertical wall and H is the solitary wave height.

Two different initial depth has been studied here (d=18.8cm, d=21.8cm). For all cases a grid size of $\Delta x = 0.10$m has been considered. Table 4 demonstrates a comparison of runup law with the numerical data for both depths studied here.

d= 21.8 cm				d= 18.8 cm			
H	R(Runup Law)	R(Numerical Calculation)	Error (%)	H	R(Runup Law)	R(Numerical Calculation)	Error (%)
0.005	0.0146	0.0141	3.7	0.005	0.0152	0.0148	2.5
0.01	0.0293	0.0288	1.6	0.01	0.0304	0.0310	2.1
0.03	0.0878	0.0850	3.2	0.03	0.0911	0.0871	4.4
0.05	0.1463	0.1390	5.0	0.05	0.1519	0.1440	5.2

Tabel 4. Maximum runup for solitary wave on a composite beach.

5.3 Laboratory benchmarking

In this part different laboratory benchmarks has been studied and the result of the numerical calculations has been compared with the laboratory data. Following tests has been covered in this part:

1) Solitary wave on a simple beach
2) Solitary wave on a composite beach
3) Solitary wave on a conical island
4) Tsunami runup onto a complex three-dimensional beach; Monai Valley

5.3.1 Solitary wave on a simple beach

In this benchmark, the 31.73 m-long, 60.96 cm-deep and 39.97 cm wide California Institute of Technology, Pasadena, California wave tank was used with water at varying depths. The tank is described by Hammack (1972), Goring (1978), and Synolakis (1986, 1987). The bottom of the tank consisted of painted stainless steel plates. Carriage rails run along the whole length of the tank, permitting the arbitrary movement of instrument carriages. A ramp was installed at one end

of the tank to model the bathymetry of the canonical problem of a constant-depth region adjoining a sloping beach. The ramp had a slope of 1:19.85. The ramp was sealed to the tank side walls. The toe of the ramp was distant 14.95 m from the rest position of the piston generator used to generate waves. This set of laboratory data has been vastly used for many code validations. In this modeling test, the data sets for the $\tilde{H}/\tilde{d} = 0.0185$ nonbreaking and $\tilde{H}/\tilde{d} = 0.3$ breaking solitary waves which are the most frequently used and most appropriate for code validation.

For these cases a grid size of $\Delta x = 0.10m$ has been utilized. Figure 8 and Figure 9 displays the accuracy of the model for both nonbreaking and breaking waves. The error for nonbreaking waves was 3.3% and for breaking waves were 5.8%.

5.3.2 Solitary wave on a composite beach

Revere Beach is located approximately 6 miles northeast of Boston in the City of Revere, Massachusetts. To address beach erosion and severe flooding problems, a physical model of the beach was constructed at the Coastal Engineering Laboratory of the U.S. Army Corps of Engineers, Vicksburg, Mississippi facility, earlier known as Coastal Engineering Research Center.

The beach characteristics are exactly the same as the composite beach described in part 2.2. In this benchmark a nonbreaking wave has been modeled and the data has been compared with the laboratory data. Similar to part 5.2 grid size for this case has been considered $\Delta x = 0.10$m. The code was able to calculate the maximum runup within error percentage of 3.8. Table 5 shows the results.

5.3.3 Solitary wave on a conical island

Motivated by the catastrophe in Babi Island, Indonesia (Yeh et al., 1994), during the 1992 Flores Island tsunami, large-scale laboratory experiments were performed at Coastal Engineering Research Center, Vicksburg, Mississippi, in a 30 m-wide, 25 m-long, and 60 cm-deep wave basin (Figure 10 and 11).

In the physical model, a 62.5 cm-high, 7.2 m toe-diameter, and 2.2 m crest-diameter circular island with a 1:4 slope was located in the basin (Figure 12). Experiments were conducted at depth of 32cm, with three different solitary waves (H/d=0.045, 0.091, 0.181). Water-surface time histories were measured with 27 wave gages located around the perimeter of the island (Figure 12).

For this case, time histories of the surface elevation around the circular island are given at four locations, i.e., in the front of the island at the toe (Gauge 6) and gauges closest to the shoreline with the numbers 9, 16, and 22 located at the $0°, 90°$, and $180°$ radial lines (Figure 14). A grid size of Δx = 0.10 m has been considered for this case. Figures 13 - 15 shows the comparison between the laboratory data with numerical calculations. Table 6 represents the error of the maximum runup for each gauge for different wave heights.

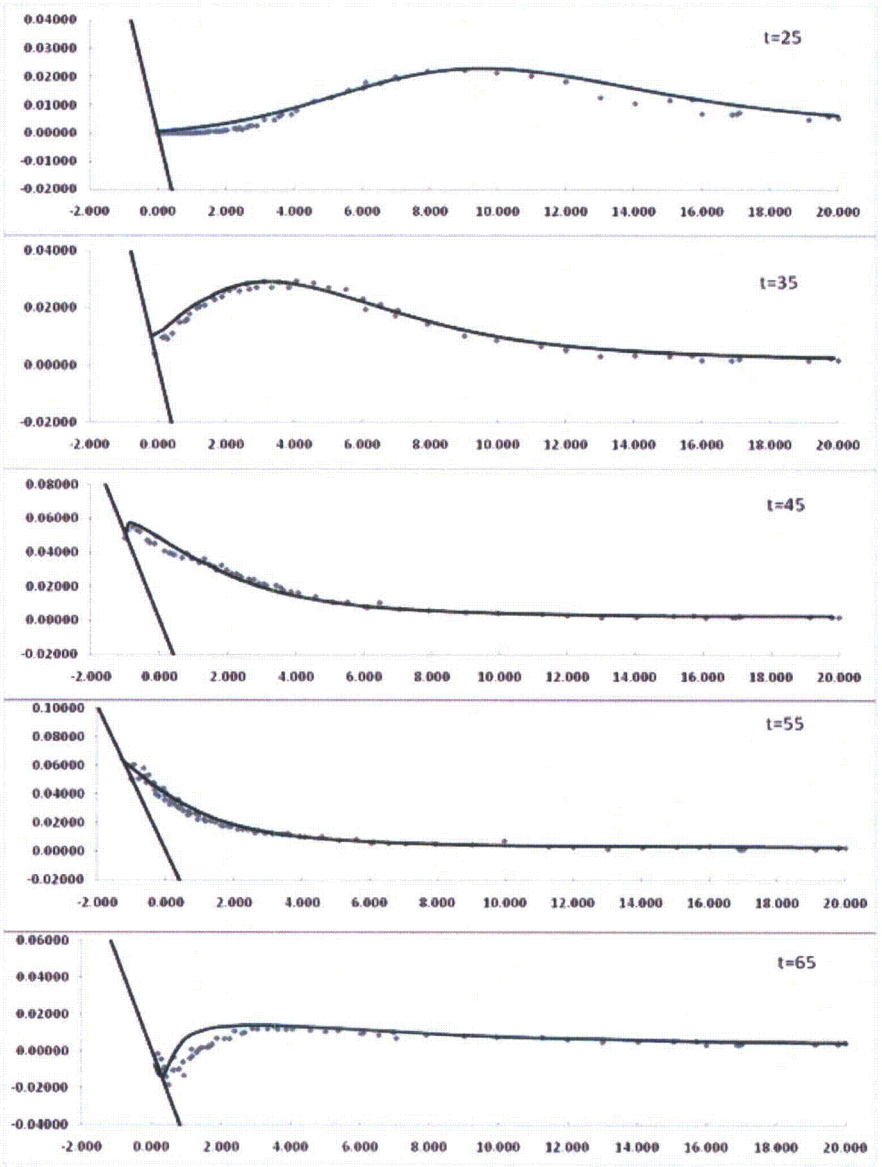

Figure 8: Time evolution of nonbreaking H/d=0.0185 initial wave. The solid line shows the numerical solution and dots represent the laboratory data.

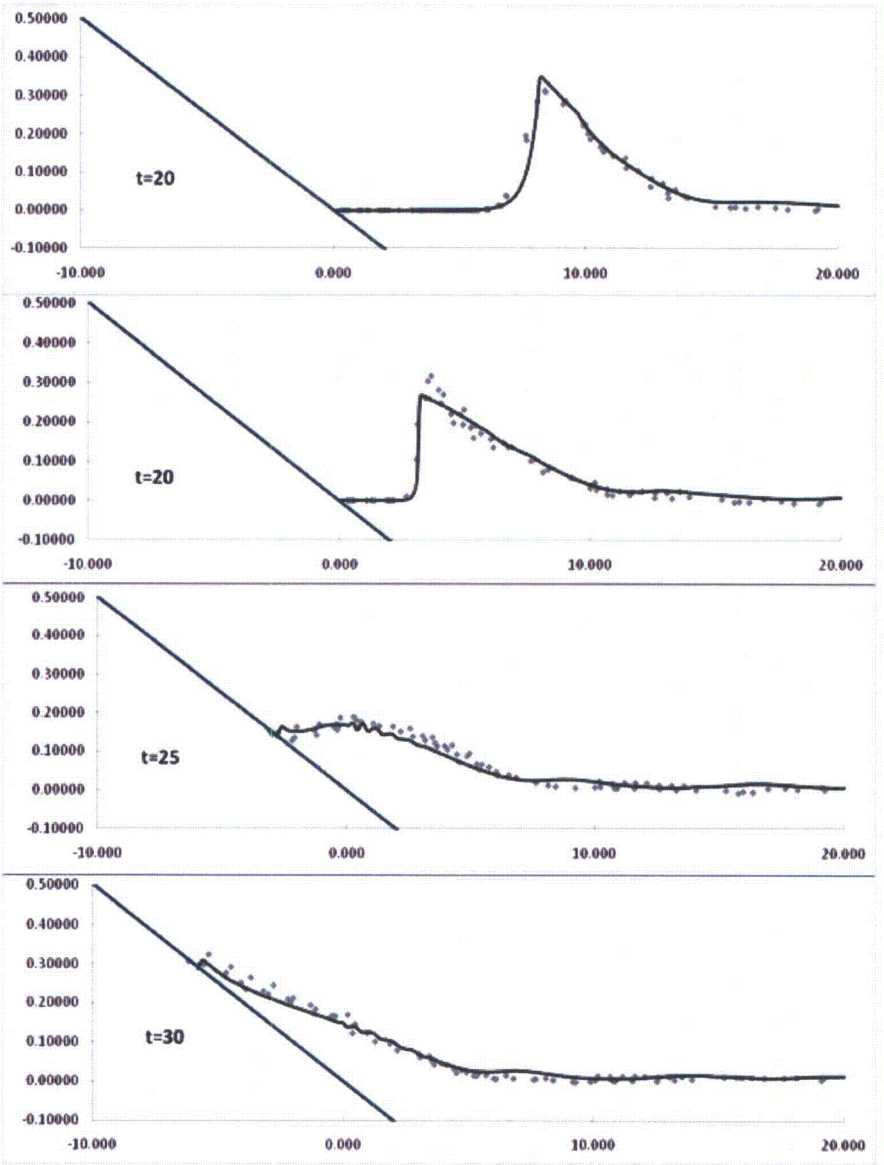

Figure 9: Time evolution of breaking H/d=0.3 initial wave. The solid line shows the numerical solution and dots represent the laboratory data.

H/d	R/d(Laboratory)	R/d (Numerical)
0.038	0.13	0.125

Table 5. Maximum runup for solitary wave on a composite beach.

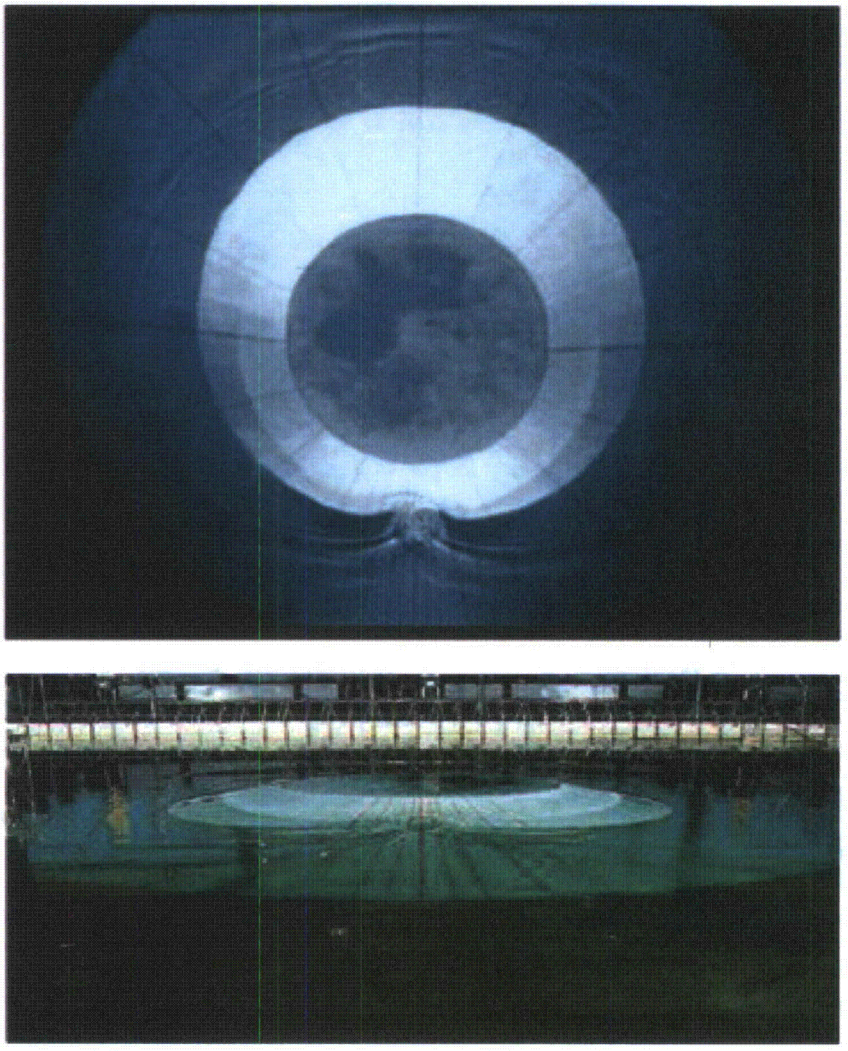

Figure 10: View of conical island (top) and basin (bottom).

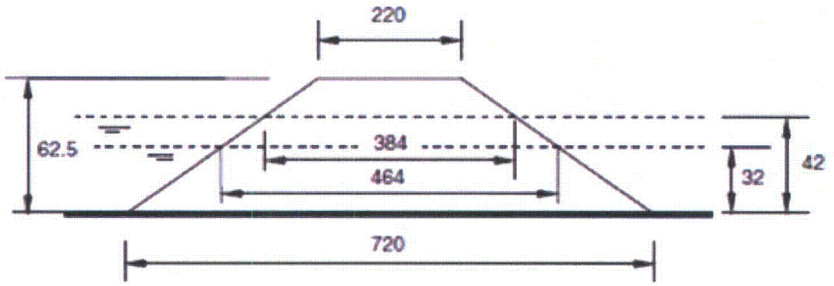

Figure 11: Definition sketch for conical island. All dimensions are in cm.

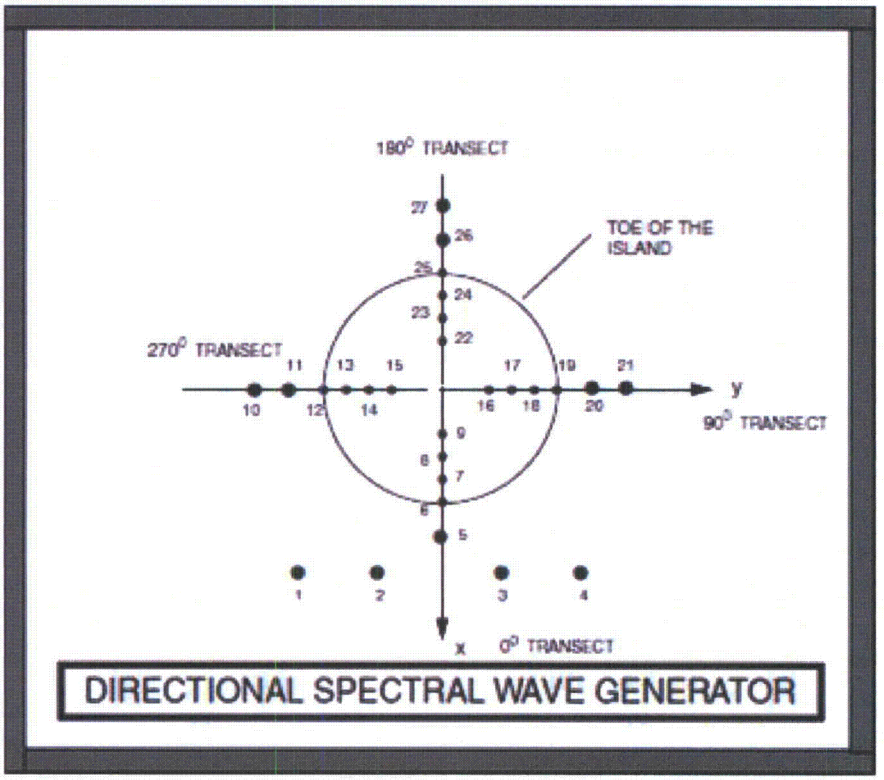

Figure 12: Schematic showing gauge locations around the conical island.

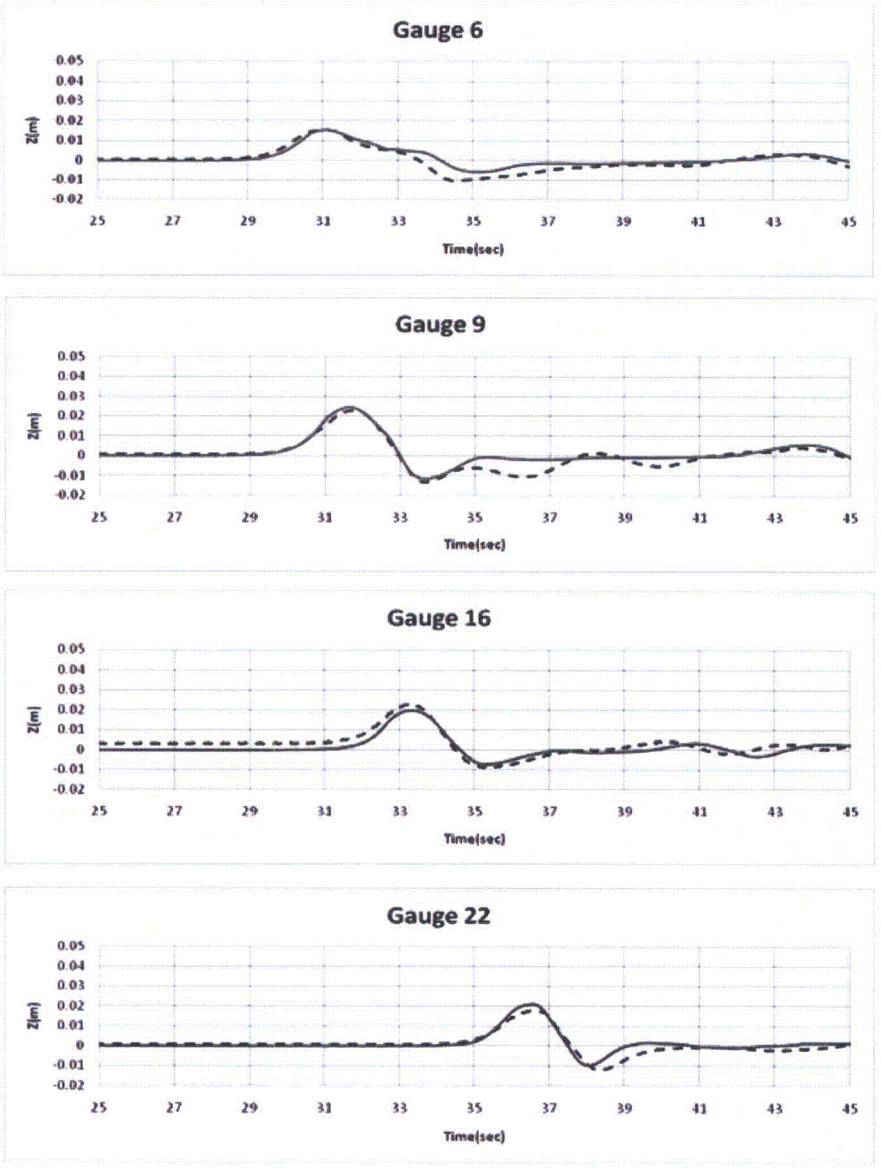

Figure 13: Comparison of computed and measured time series of free surface for H/d=0.045. Dashed lines: measured, Solid lines: Computed.

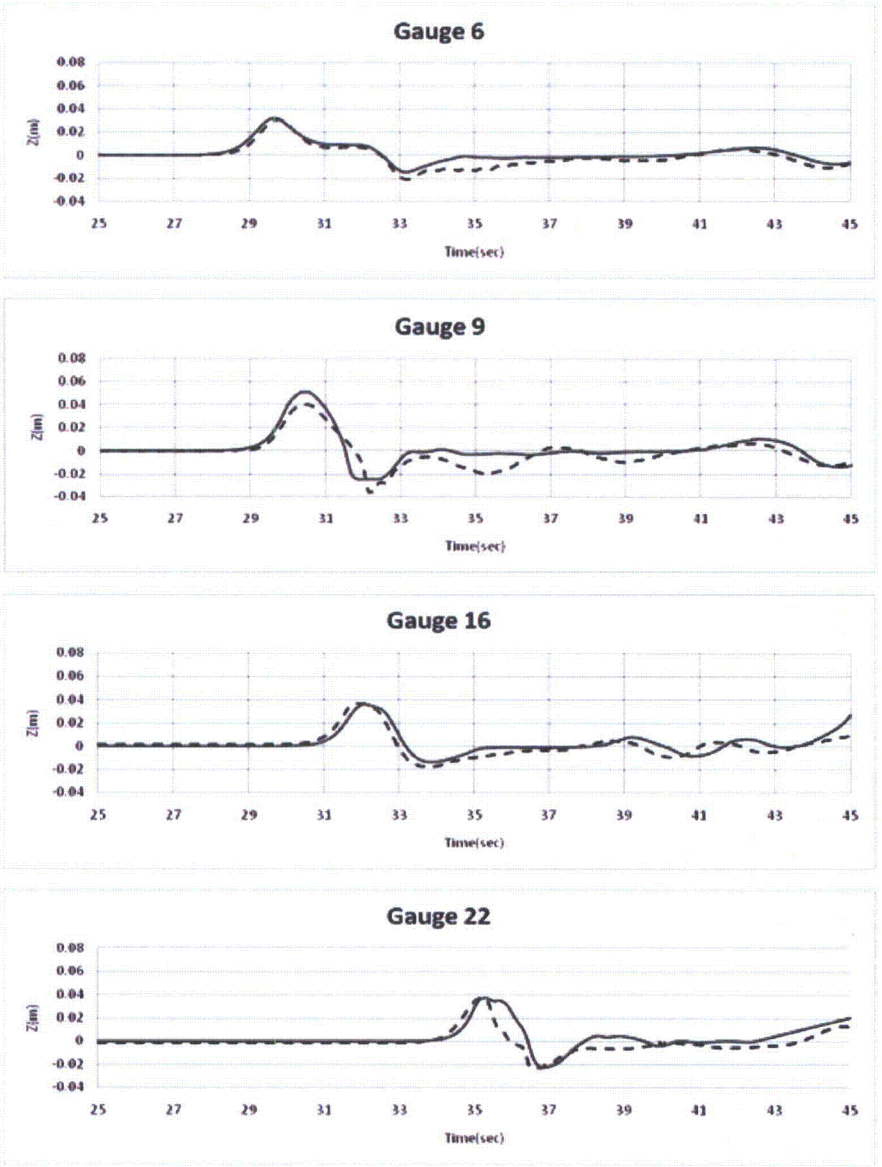

Figure 14: Comparison of computed and measured time series of free surface for H/d=0.091. Dashed lines: measured, Solid lines: Computed.

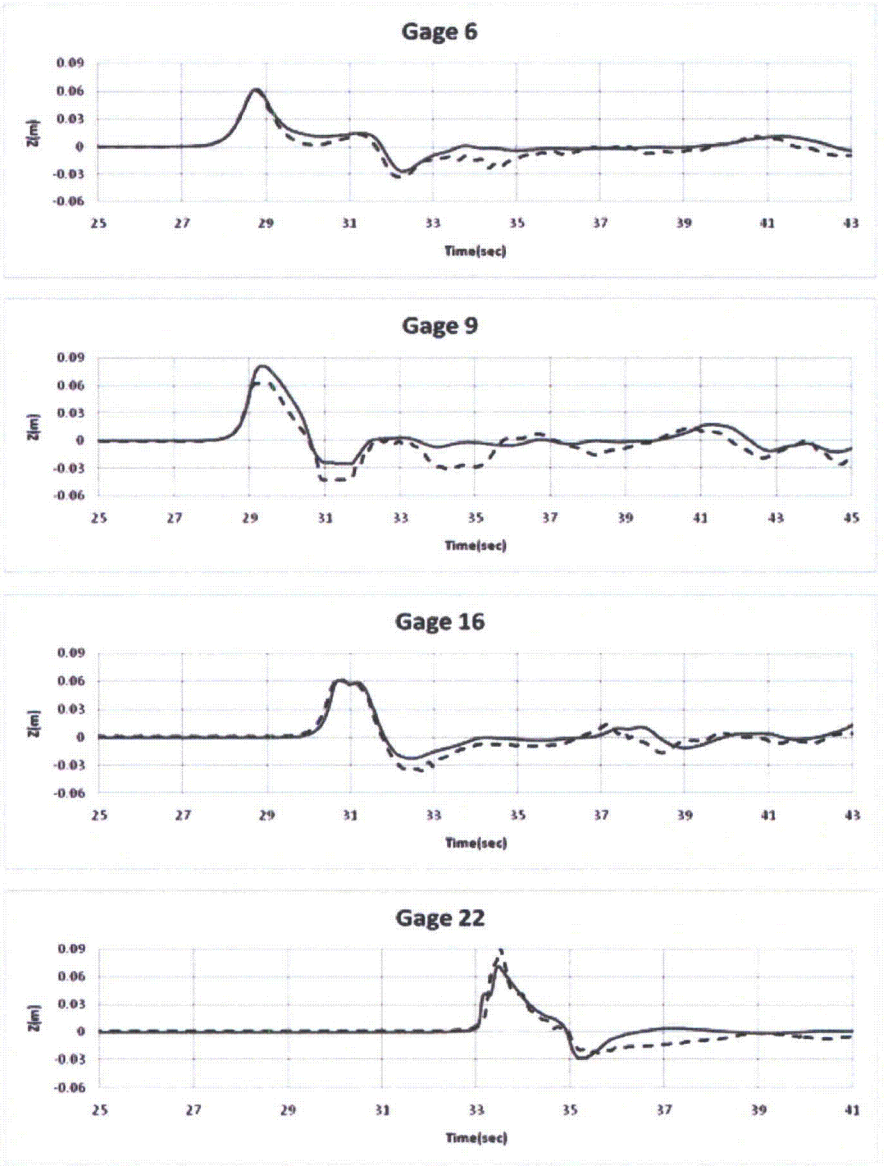

Figure 15: Comparison of computed and measured time series of free surface for H/d=0.181. Dashed lines: measured, Solid lines: Computed.

	Gauge Number			
H/d	6	9	16	22
0.045	0.3	7.4	14.3	8.6
0.091	5.6	19.5	4.6	0.3
0.181	4.3	19.8	1.7	6.2

Tabel 6. Error percentage of maximum runup calculation for each case.

5.4 Field benchmarking

5.4.1 Tsunami runup onto a complex three-dimensional beach, Monai Valley

The Hokkaido-Nansei-Oki tsunami of 1993 that struck Okushiri Island, Japan, with extreme runup height of 30 m and currents of the order of 1018 m/sec was a disaster, but provided fortuitous high-quality data. The extreme tsunami runup mark was discovered at the tip of a very narrow gulley within a small cove at Monai. High resolution seafloor bathymetry existed before the event and when coupled with bathymetric surveys following the event allowed meaningful identification of the seafloor deformation. A 1/400 laboratory model of Monai was constructed in a 205 m-long, 6 m deep, and 3.5 m-wide tank at Central Research Institute for Electric Power Industry (CRIEPI) in Abiko, Japan and partly shown in Figure 16. The laboratory setup closely resembles the actual bathymetry. The incident wave from offshore, at the water depth of d = 13.5 cm is known and it is shown in Figure 18. There are reflective vertical sidewalls at y = 0 and 3.5 m (Figure 19). The entire computational area is 5.448 m $\times$ 3.402 m, and the grid sizes for numerical simulations were $\Delta x = \Delta y = 1.4$cm. The input wave is a LDN with a leading-depression height of 2.5 mm with a crest of 1.6 cm behind it (Figure 18). Waves were measured at thirteen locations and complete time histories are given and compared with numerical simulations at three locations, i.e., (x, y) = (4.521, 1.196), (4.521, 1.696), and (4.521, 2.196) in meters (Figure 19). Also, table 7 provides the error for maximum runup computation for each gauge.

References

Bermùdez, A. and Vàzquez, M. E., 1994, Upwind methods for hyperbolic conservation laws with source terms, *Comput. Fluids*, 23 (8), 1049-1071.

Chen, Q., Dalrymple, R. A., Kirby, J. T., Kennedy, A. B. and Haller, M. C., 1999, Boussinesq modelling of a rip current system, em J. Geophys. Res., 104, 20,617-20,637.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Kennedy, A. B. and Chawla, A., 2000, "Boussinesq

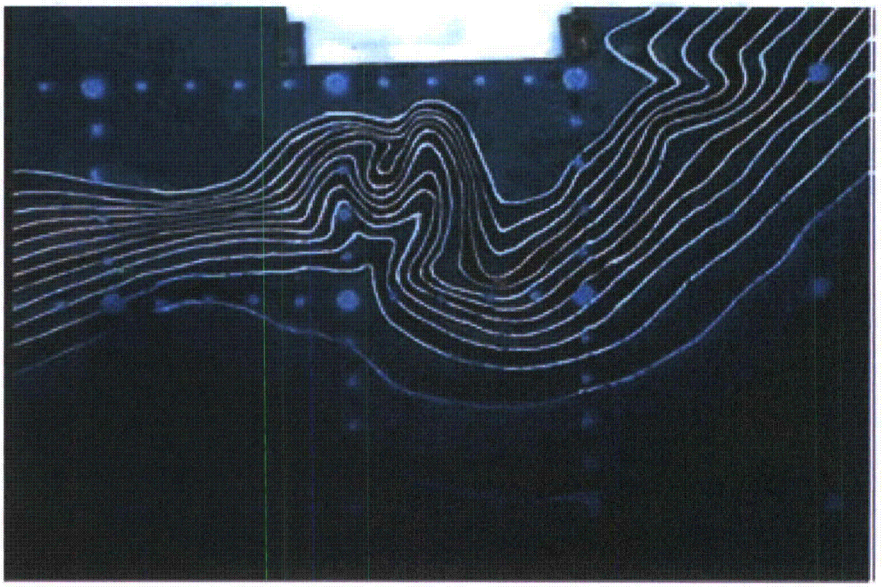

Figure 16: Bathymetric profile for experimental setup for Monai Valley experiment.

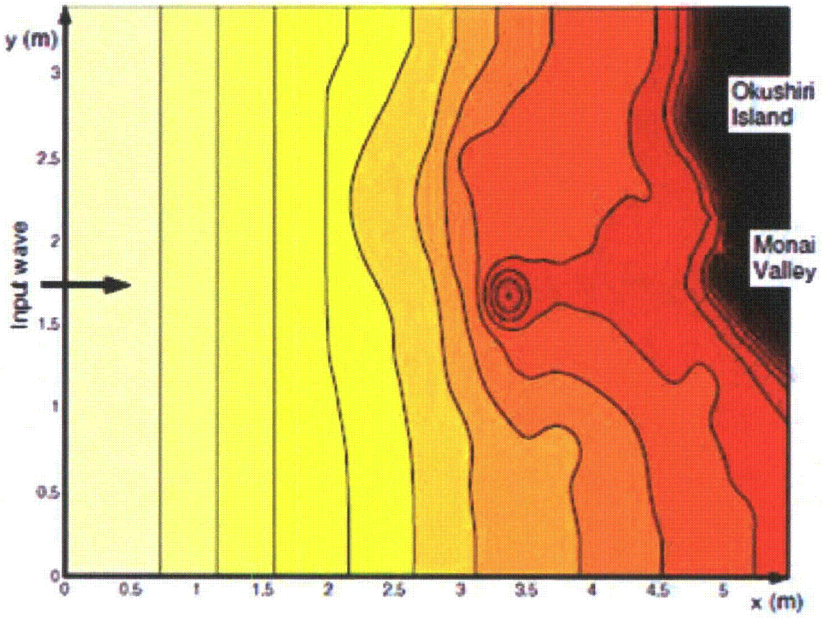

Figure 17: Computational area for Monai Valley experiment.

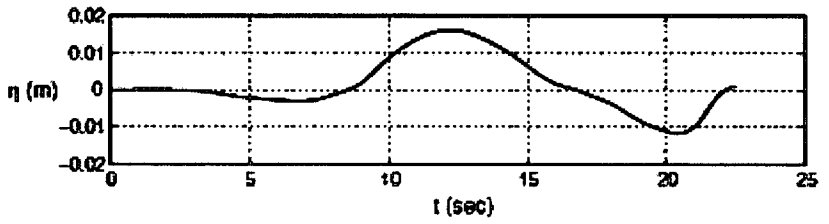

Figure 18: Initial wave profile for Monai Valley experiment.

Gauge Number	Error (%)
1	18.6
2	8.5
3	8.2

Tabel 6. Error percentage for maximum runup calculations.

modeling of wave transformation, breaking and runup. II: 2D", *J. Waterway, Port, Coastal and Ocean Engineering*, **126**, 48-56.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Shi, F. and Thornton, E. B., 2003, Boussinesq modeling of longshore currents, *Journal of Geophysical Research*, 108(C11), 3362, doi:10.1029/2002JC001308.

Chen, Q., 2006, Fully nonlinear Boussinesq-type equations for waves and currents over porous beds, *Journal of Engineering Mechanics.* 132 (2): 220-230.

Erduran, K. S., Ilic, S., and Kutija, V., 2005, Hybrid finite-volume finite-difference scheme for the solution of Boussinesq equations, *Int. J. Numer. Meth. Fluid.*, 49, 1213-1232.

Gobbi, M. F., Kirby, J. T. and Wei, G., 2000, A fully nonlinear Boussinesq model for surface waves. II. Extension to $O(kh^4)$, *Journal of Fluid Mechanics*, 405, pp. 181-210.

Gottlieb, S., Shu C.-W., and Tadmore, E., 2001, Strong stability-preserving high-order time discretization methods, *SIAM Review*, **43** (1), 89 - 112.

Hansen, J. B., and Svendsen, I. A., 1979, Regular waves in shoaling water: Experimental data, Tech. Rep. ISVA Ser., 21, Technical Univ. of Denmark, Denmark.

Kennedy, A. B., Chen, Q., Kirby, J. T. and Dalrymple, R. A., 2000, "Boussinesq modeling of wave transformation, breaking and runup. I: 1D", *J. Waterway, Port, Coastal and Ocean Engineering*, **126**, 39-47.

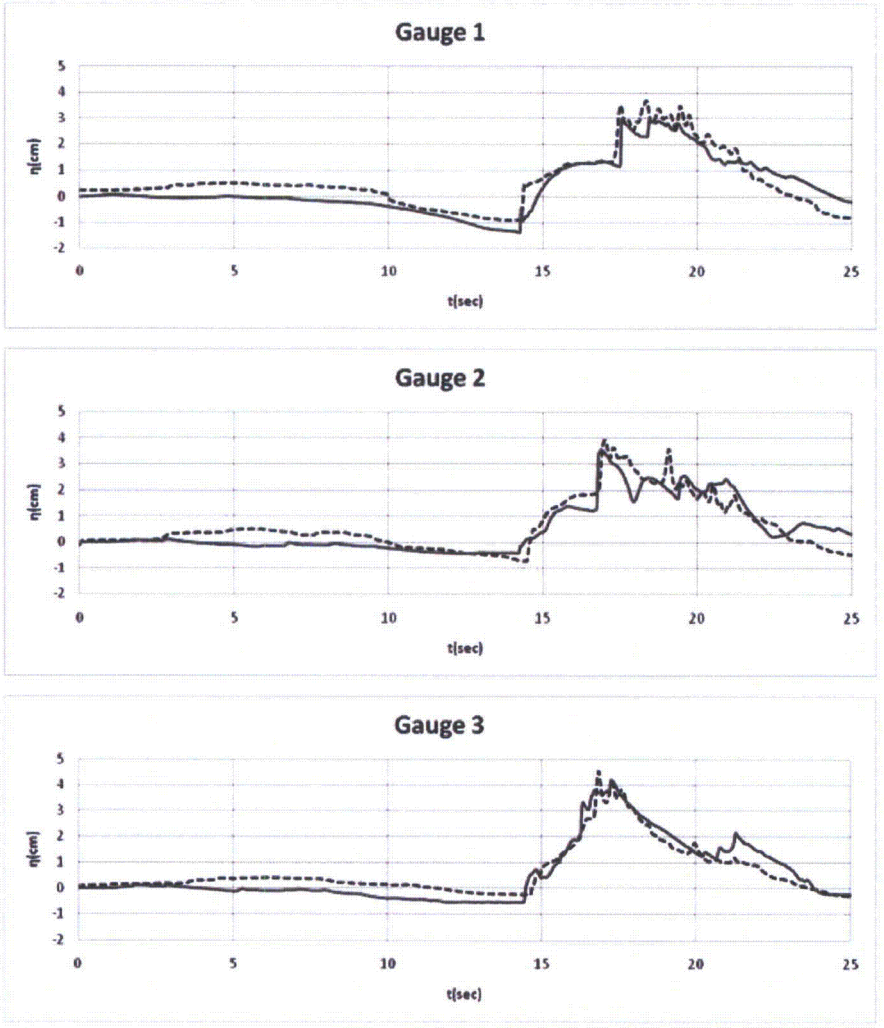

Figure 19: Comparison of computed and measured time series of free surface. Dashed lines: measured, Solid lines: Computed.

Kennedy, A. B., Kirby, J. T., Chen, Q. and Dalrymple, R. A., 2001, Boussinesq-type equations with improved nonlinear performance, *Wave Motion*, 33, pp. 225-243.

Kim D. H., Cho, Y. S., and Kim, H. J., 2008, Well-balanced scheme between flux and source terms for computation of shallow-water equations over irregular bathymetry, *Journal of Engineering Mechanics*, 134, 277-290.

Kim, D. H., Lynett, P. J. and Socolofsky, S. A., 2009, "A depth-integrated model for weakly dispersive, turbulent, and rotational fluid flows", *Ocean Modeling*, **27**, 198-214.

Kim, D. H., 2009, "Turbulent flow and transport modeling by long waves and currents", Ph.D. dissertation, Texas A& M University.

Kirby, J. T., Wei, G., Chen, Q., Kennedy, A. B. and Dalrymple, R. A., 1998, FUNWAVE 1.0, Fully nonlinear Boussinesq wave model. Documentation and users manual. Report CACR-98-06, Center for Applied Coastal Research, Department of Civil and Environmental Engineering, University of Delaware.

Kirby, J.T., Shi, F., Watts, P., Grilli, S.T., 2004, Propagation of short, dispersive tsunami waves in ocean basins. EOS Transactions of the AGU 85 (47) Abstract OS21E-02.

Liang, Q. and Marche, F., 2009, Numerical resolution of well-balanced shallow water equations with complex source terms, *Advances in Water Resources*, **32**, 873 - 884.

Long, W. and Kirby, J. T., 2006, Boussinesq modeling of waves, currents and sediment transport, Research Report No. CACR-06-02, Center for Applied Coastal Research, Dept. of Civil and Environmental Engineering, Univ. of Delaware, Newark.

Lynett, P. J., Wu, T.-R. and Liu, P. L.-F., 2002, Modeling wave runup with depth-integrated equations, *Coastal Engineering*, 46, 89-107.

Madsen, P.A., Srensen, O.R., 1992, A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2. A slowly-varying bathymetry, *Coastal Engineering* 18 (3-4), 183-204.

Mase, H., and Kirby, J. T., 1992, Hybrid frequency-domain KdV equation for random wave transformation, *Proc., 23rd Int. Conf. Coast. Eng.*, ASCE, New York, 474-487.

Naik, N. H., Naik, V. K., and Nicoules, M., 1993, Parallelization of a class of implicit finite difference schemes in computational fluid dynamics, *International Journal of High Speed Computing*, 5: 1-50.

Ning, D. Z., Zang, J., Liang, Q., Taylor, P. H., and Borthwick, A. G. L., 2008, Boussinesq cut-cell model for non-linear wave interaction with coastal structures, *International Journal for Numerical Methods in Fluids*, 57 (10), 1459-1483.

Nwogu, O., 1993, An alternative form of the Boussinesq equations for nearshore wave propagation, *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 119 (6), pp. 618-638.

Nwogu, O. and Demirbilek, Z., 2001, BOUSS-2D: A Boussinesq wave model for coastal regions and harbors, ERDC/CHL TR-01-25, Coastal and Hydraulics Laboratory, USACOE Engineer Research and Development Center, Vicksburg, MS.

Roeber, V., Cheung, K. F., and Kobayashi, M. H., 2010, Shock-capturing Boussinesq-type model for nearshore wave processes, *Coastal Engineering*, 57, 407-423.

Rogers, B. D., Borthwick, A. G. L., and Taylor, P. H., 2003, Mathematical balancing of flux gradient and source terms prior to using Roe's approximate Riemann solver, *Journal of Computational Physics*, 192, 422-451.

Shi, F., Dalrymple, R. A., Kirby, J. T., Chen, Q. and Kennedy, A., 2001, A fully nonlinear Boussinesq model in generalized curvilinear coordinates. *Coastal Engineering*, Vol. 42, pp. 337-358.

Shiach, J. B. and Mingham, C. G., 2009, A temporally second-order accurate Godunov-type scheme for solving the extended Boussinesq equations, *Coastal Engineering*, 56, 32-45.

Sitanggang, K. I. and Lynett, P., 2005, "Parallel computation of a highly nonlinear Boussinesq equation model through domain decomposition", *Int. J. Num. Meth. Fluids*, **49**, 57-74.

Smagorinsky, J., 1963, General circulation experiments with the primitive equations. I. The basic experiment, *Mon. Weather Rev*, 91, 99-165.

Synolakis, C. E., Bernard, E. N., Titov, V. V., Kânoglu, U. and González, F. I., 2007, "Standards, criteria, and procedures for NOAA evaluation of tsunami numerical models", *NOAA Tech. Memo. OAR PMEL-135*, National Oceanic and Atmospheric Administration.

Tehranirad, B., Shi, F., Kirby, J. T., Harris, J. C. and Grilli, S. , 2011, Benchmark tests of tsunami wave runup using FUNWAVE-TVD, to be submitted to Proceedings of National Tsunami Workshop.

Ting, F.C.K., Kirby, J.T., 1994, Observation of undertow and turbulence in a laboratory surf zone. *Coast. Eng.* 24, 5180.

Tonelli, M. and Petti, M., 2009, "Hybrid finite volume - finite difference scheme for 2DH improved Boussinesq equations", *Coast. Engrng.*, **56**, 609-620.

Tonelli, M. and Petti, M., 2010, "Finite volume scheme for the solution of 2D extended Boussinesq equations in the surf zone", *Ocean Engrng.*, **37**, 567-582.

Wei, G., Kirby, J.T., Grilli, S.T., Subramanya, R., 1995, A fully nonlinear Boussinesq model for surface waves: Part I. Highly nonlinear unsteady waves *Journal of Fluid Mechanics* 294, 7192.

Wei, G. and Kirby, J. T., 1995, A time-dependent numerical code for extended Boussinesq equations, *Journal of Waterway, Port, Coastal and Ocean Engineering*, 120, pp. 251-261.

Yamamoto, S., Daiguji, H., 1993, Higher-order-accurate upwind schemes for solving the compressible Euler and NavierStokes equations, *Computers and Fluids*, 22 (2/3), 259270.

Yamamoto, S., Kano, S. and Daiguji, H, 1998, An efficient CFD approach for simulating unsteady hypersonic shockshock interference flows, *Computers and Fluids* 27 (56), pp. 571-580.

Zelt, J. A., 1991, The runup of nonbreaking and breaking solitary waves, *Coastal Engineering*, 15, pp. 205-246.

Zhou, J. G., Causon, D. M., Mingham C. G., and Ingram, D. M., 2001, The surface gradient method for the treatment of source terms in the shallow-water equations, *Journal of Computational Physics*, 168, 1-25.

NOAA Technical Memorandum OAR PMEL-135

STANDARDS, CRITERIA, AND PROCEDURES FOR NOAA EVALUATION OF TSUNAMI NUMERICAL MODELS

Costas E. Synolakis[1]
Eddie N. Bernard[2]
Vasily V. Titov[3]
Utku Kânoğlu[4]
Frank I. González[2]

[1] Viterbi School of Civil Engineering
University of Southern California
Los Angeles, CA

[2] Pacific Marine Environmental Laboratory
Seattle, WA

[3] Joint Institute for the Study of the Atmosphere and Ocean (JISAO)
University of Washington, Seattle, WA

[4] Department of Engineering Sciences
Middle East Technical University
Ankara, TURKEY

Pacific Marine Environmental Laboratory
Seattle, WA
May 2007

UNITED STATES DEPARTMENT OF COMMERCE

Carlos M. Gutierrez
Secretary

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION

VADM Conrad C. Lautenbacher, Jr.
Under Secretary for Oceans and Atmosphere/Administrator

Office of Oceanic and Atmospheric Research

Richard W. Spinrad
Assistant Administrator

NOTICE

Mention of a commercial company or product does not constitute an endorsement by NOAA/OAR. Use of information from this publication concerning proprietary products or the tests of such products for publicity or advertising purposes is not authorized.

Contribution No. 3053 from NOAA/Pacific Marine Environmental Laboratory

Also available from the National Technical Information Service (NTIS)
(http://www.ntis.gov)

Contents

1. Introduction 1
2. Model Evaluation Standards 3
2.1 Basic considerations 4
2.2 Analytical benchmarking 4
2.3 Laboratory benchmarking 5
2.4 Field data benchmarking 7
2.5 Scientific evaluation 8
2.6 Operational evaluation 9
3. Criteria for Evaluating Operational Forecasting and Inundation Mapping Models 9
3.1 Model computational time constraints 10
3.2 Model accuracy constraints 11
3.3 Model reliability and realism constraints 11
4. Summary of Model Evaluation Procedures 12
5. Acknowledgments 14
6. References 14
Appendix A: Existing Methods for Model Validation and Verification 17
1 Basic hydrodynamic considerations 17
1.1 Mass conservation 17
1.2 Convergence 18
2 Analytical benchmarking 18
2.1 Single wave on a simple beach 18
2.2 Solitary wave on composite beach 30
2.3 Subaerial landslide on simple beach 33
3 Laboratory benchmarking 34
3.1 Solitary wave on a simple beach 36
3.2 Solitary wave on a composite beach 37
3.3 Solitary wave on a conical island 39
3.4 Tsunami runup onto a complex three-dimensional beach; Monai Valley 45
3.5 Tsunami generation and runup due to three-dimensional landslide 47
4 Field benchmarking 48
4.1 Okushiri Island 48
4.2 Rat Islands tsunami 54

Standards, criteria, and procedures for NOAA evaluation of tsunami numerical models

C.E. Synolakis[1], E.N. Bernard[2], V.V. Titov[3], U. Kânoğlu[4], F. González[2]

1. Introduction

The National Oceanic and Atmospheric Administration (NOAA) is the federal agency charged with mitigating tsunami hazards in the United States. NOAA's National Weather Service operates the two Tsunami Warning Centers (TWCs) in the U.S., and NOAA has spearheaded the national effort to develop inundation maps for evacuation planning through the National Tsunami Hazard Mitigation Program (NTHMP). The latter was formed through a directive of the U.S. Senate Appropriations Committee in 1994 to develop a plan for a tsunami warning system that reduces the risk to coastal residents. Following the horrific Indian Ocean tsunami of 26 December 2004, the U.S. expanded the role of NTHMP to serve as the organizational framework to implement the recommendations of the National Science and Technology council report "Tsunami Risk Reduction for the United States: A Framework for Action" (NSTC, 2005):

- Develop standardized and coordinated tsunami hazard and risk assessments for all coastal regions of the United States and its territories.
- Improve tsunami and seismic sensor data and infrastructure for better tsunami detection and warning.
- Enhance tsunami forecast and warning capability along our coastlines (Pacific, Atlantic, Caribbean, and Gulf of Mexico) by increasing the number of Deep-ocean Assessment and Reporting of Tsunamis (DART) buoys, tide gauges, and seismic sensors feeding real-time data into on-line forecast models.
- Ensure interoperability between the U.S. national system and other regional tsunami warning systems.
- Provide technical expertise and assistance, as appropriate, to facilitate development of international tsunami and all-hazard warning systems, including for the Indian Ocean.

[1]Viterbi School of Civil Engineering, University of Southern California, Los Angeles, CA 90089-2531, USA
[2]NOAA/Pacific Marine Environmental Laboratory, 7600 Sand Point Way NE, Seattle, WA 98115-6349, USA
[3]Joint Institute for the Study of the Atmosphere and Ocean (JISAO), University of Washington, Box 357941, Seattle, WA 98195-4235, USA
[4]Department of Engineering Sciences, Middle East Technical University, 06531 Ankara, TURKEY

- Encourage data exchange and interoperability among all regional tsunami and all-hazard warning systems, such as the Intergovernmental Oceanographic Sub-commission for the Caribbean (IOCARIBE).

- Promote development of model mitigation measures and encourage communities to adopt construction, critical facilities protection, and land-use planning practices to reduce the impact of future tsunamis.

- Increase outreach to all communities, including all demographics of the at-risk population, to raise awareness, improve preparedness, and encourage the development of tsunami preparedness plans.

- Conduct an annual review of the status of tsunami research and develop a strategic plan for tsunami research in the United States.

Since standards for modeling tools do not currently exist, and given the increased number of states in the Atlantic, the Pacific, and territories in the Caribbean that will need to develop tsunami mitigation plans, there is the risk that forecast products may be produced with older or untested methodology. This is not just a U.S. problem, but worldwide, as UNESCO's plans are to build local capacity for developing maps in most member nations facing tsunami hazards. Incorrectly assessing possible inundation can be costly both in terms of lives lost, or in unnecessary evacuations in areas larger than warranted that may put lives at risk and reduce the credibility of the system, even in areas that were not directly affected. Standards are urgently needed to ensure a minimum level of quality and reliability for forecasting and inundation products.

To calculate tsunami currents, forces and runup on coastal structures, and inundation of coastlines, one must numerically calculate the evolution of the tsunami wave from the deep ocean to its target coastal community. No matter what the numerical model, both validation (the process of ensuring that the model solves the parent equations of motion accurately) and verification (the process of ensuring that the model used represents geophysical reality appropriately) are an essential part of the model development. Validation ensures that the model performs well in a wide range of circumstances and is accomplished through comparison with analytical solutions. Verification ensures that the computational code performs well over a range of geophysical problems. Many existing numerical models have been validated through comparison with analytical solutions. Very few have also been verified with laboratory and field measurements. Even fewer have been tested in an operational forecast setting. Numerical models that have been tested under all three conditions represent the gold standard for both inundation mapping and real-time forecasting.

In the last ten years, the process of model validation and verification has shown that terminal effects of tsunamis can be described by depth-averaged equations. Interest in these equations has arisen because comparisons with both large-scale laboratory data and field data have demonstrated a remarkable and perhaps surprising capability to model complex evolution phenomena, and in particular the maximum runup and inundation, extremely well

over wide ranges of tsunami waves. In the current state of knowledge, the main uncertainty arises from the ambiguities of the initial condition, presuming that the solution methodology solves the equations of motion satisfactorily. With the increasing deployment of tsunameters that monitor the tsunami evolution in the deep ocean and allow for real-time updates of the initial data, this remaining uncertainty is rapidly diminishing.

While equation solvers for more comprehensive approximations of the parent Navier-Stokes equations now exist, they are presently too computationally intensive for inundation mapping or operational forecasting, and are generally used for free-surface flows of very limited geographical extent. These models remain largely unvalidated over wide ranges of tsunami events. Further, the rapid development of packaged numerical modeling tools that allow for the production of high-end animations even by untrained users, underscores the issue of validating specific calculation tools used in tsunami inundation and forecasting to avoid producing mathematically correct, but physically unrealistic solutions.

In this report, benchmark tests for validating and verifying computational tools for predicting the coastal effect of tsunamis are discussed. State-of-the-art methods for validation of tsunami codes are reviewed. Standards and guidelines for validation and verification are recommended for operational codes used for inundation mapping and tsunami forecasting.

2. Model Evaluation Standards

Tsunami inundation models have evolved in the last two decades through careful and explicit validation through comparisons of their predictions with 1+1 (one directional and time) and 2+1 (two directional and time) analytical solutions, laboratory experiments, and field measurements. (See Synolakis and Bernard (2006) for a comprehensive discussion of the evolution of tsunami hydrodynamics.)

Numerical codes often include terms with "friction" factors in an attempt to model physical dissipation, but primarily to stabilize what is an inherently marginal inundation computation, given the small flow depths and large velocities. Large friction factors make calculations more stable, but may under-predict inundation distances and runup heights. Smaller friction factors (or just relying on the inherent numerical dissipation of any numerical calculation) result in more realistic predictions, but they tend to be less stable, depending on the particular differencing scheme used. Thus it is recommended that any code used for modeling tsunami inundation at geophysical scales be tested with all three types of validation data: analytical solutions, laboratory measurements, and field measurements.

While there is in principle no absolute certainty that a numerical code that has performed well in all the benchmark tests will also produce realistic inundation predictions with any given source motions, validated codes largely reduce the level of uncertainty in their results to the uncertainty in the geophysical initial conditions. Further, when coupled with real-time free-field tsunami measurements from tsunameters, validated codes are the only

choice for realistic forecasting of inundation; the consequences of failure are too serious to take chances with less-validated numerical procedures.

In what follows, the steps for validating 2+1 codes are discussed leaving the details of the benchmark problems to Appendix A. Although 1+1 codes were used two decades ago for developing early inundation maps, are generally unreliable for inundation mapping and entirely inadequate for tsunami forecasting. Nonetheless, analytic solutions to the corresponding 1+1 hydrodynamic equations are invaluable to the process of model validation, and several are presented in Appendix A. Specific steps are recommended for the approval of modeling tools, their further development, and their transfer to operations. These steps can be classified into six categories: basic hydrodynamic considerations, analytical benchmarking, laboratory benchmarking, field data benchmarking, scientific evaluations, and operational evaluations.

2.1 Basic considerations

Two basic steps are required before benchmarking the numerical codes: mass conservation and convergence.

Mass conservation: The first basic step in ensuring that a numerical model accurately simulates tsunami evolution is ensuring that the model conserves mass. While the conservation of mass equation is one of the equations of motion that are solved in any numerical procedure, cumulative numerical approximations can sometimes produce results that violate mass conservation. This is particularly true when friction factors or smoothing to stabilize inundation computations are used. Calculations of conservation of mass should be such that the total initial displaced volume should be within 5% of the total displaced volume at the end of the computation, i.e., when the initial wave is entirely reflected and offshore.

Convergence: The next basic step is checking convergence of the numerical code to a certain asymptotic limit, ideally the actual solution of the equations solved, if one exists. The optimal locations to check convergence are the extreme runup and rundown. The numerical predictions should be seen to converge to a certain value, and further reductions in step sizes should not change the computed results.

2.2 Analytical benchmarking

Why is analytical benchmarking important? To calculate tsunami currents, forces, and runup on coastal structures and coastal inundation, one must calculate the evolution of the tsunami wave from deep ocean to its target.

Exact solutions of the shallow water-wave equations are useful for validating the complex numerical models that are used for final design. These often involve ad-hoc assumptions, particularly during inundation computations when grid points are introduced and withdrawn as the shoreline recedes and advances. Comparisons of numerical predictions with analytical solutions can identify systematic errors, as when using friction factors or dissipative terms to augment the idealized equations of motion.

Here, we only present certain common 1+1 propagation problems. The

waves evolve over constant-depth and then over plane or composite beaches. Even though most results are derived for idealized waveforms often used in tsunami engineering to describe the leading wave of a tsunami, the generalization to realistic spectral distributions of geophysical tsunamis is trivial with the closed-form integrals provided.

It is important to note that validation should take place with non-periodic waves. During runup, individual monochromatic waves reflect with slope-dependent phase shifts. Whereas a code may model a periodic wave well, it may not model wave superposition well. This was a problem of earlier shallow water-wave formulations that did not account for reflection. While their predictions for the Carrier–Greenspan (Carrier and Greenspan, 1958) sinusoids was correct, they exhibited significant errors when modeling solitary waves or N-waves.

Further, the analytical solutions allow for using cyber-waves of the same scale as geophysical tsunamis. This is in contrast to laboratory experiments that require modeling of small-scale experiments.

Analytical solutions for three problems are presented for analytical benchmarking:

Single wave on simple beach: Solitary and N-wave propagation first over the constant-depth then sloping beach is considered. Details of the analytical solutions are given in Appendices A2.1.1, A2.1.2, and A2.1.3. Numerical models should calculate the maximum runup of nonbreaking solitary and N-waves within 5% of the calculated value from the analytical solution. Numerical experiments should be undertaken in a range of depths from 50 cm to 1000 m, and the initial waves scaled appropriately, i.e., for a $\tilde{H}/\tilde{d} = 0.02$ initial solitary wave; at 50 cm the initial height should be 1 cm, at 1000 m it should be 2 m. In addition, the initial value problem solution of the nonlinear shallow water-wave equations is described and runup/rundown and velocity quantities for shoreline are presented in Appendix A2.1.4 for benchmarking. Any well-benchmarked code should produce results within 5% of the calculated value from the analytical solution.

Solitary wave on composite beach: 1+1 models that perform well with the solitary wave experiments must still be tested with the composite beach geometry, for which an analytical solution (Appendix A2.2) exists, with solitary waves as inputs. Numerical predictions should not differ by more than 5% in predicting maximum runup for non-breaking waves.

Subaerial landslide on simple beach: Inundation computations are exceedingly difficult when the beach is deforming as a landslide is occurring. Numerical predictions of the runup from an idealized landslide of translating Gaussian shaped mass should not differ from the analytical model by more than 10% (Appendix A2.3).

2.3 Laboratory benchmarking

Why is laboratory benchmarking important? It is quite clear from the earlier discussion that numerical methods have evolved only through careful validation accomplished through comparisons with analytic solutions and laboratory measurements. Long before the availability of numerical codes,

physical models at small scale have been used to visualize wave phenomena in the laboratory and then predictions were scaled to the prototype. Even today, when designing harbors, laboratory experiments—scale model tests—are used to confirm different flow details and validate the numerical model used in the analysis.

For the purpose of validating inundation models, the scale differences are not believed to be important. Numerical codes developed in the last decade that consistently produce predictions in excellent agreement with measurements from small-scale laboratory experiments have been shown to model geophysical-scale tsunamis well. For example, a numerical code that adequately models the inundation in a 1 m-deep model is also expected to model the inundation in the 1 km-deep geophysical geometry, as the grid sizes are adjusted accordingly and in relationship to the scale of the problem. Scale models, in general, do not have bottom friction characteristics similar to real ocean floors or sandy beaches, but this has proven not to be a severe limitation. Tsunamis are such long waves that bottom friction tends to be less important than the inertia of the motion. Friction may be important in cases of extreme inundation, as observed during the 2004 Boxing Day tsunami in Banda Aceh with 3 km inundation distances. However, it has been observed that even with numerical codes that use friction factors within reasonable limits, the predictions are not sensitive to the first order.

Experimental results from five laboratory tests are described as laboratory benchmarking:

Single wave on simple beach: Given a small number of 2+1 wave basin laboratory measurements, 1+1 versions of the 2+1 models should be tested with 1+1 directional laboratory models. The solitary wave experiments on the canonical model—waves propagating over a constant-depth region and running up a 1:20 sloping beach—should be used first. Numerical models should calculate the maximum runup of nonbreaking solitary waves within 5% of the measured values in the laboratory. For breaking waves, the models should produce predictions within 10% of the measured values, and they should consistently predict the runup variation described in Appendix A3.1.

Solitary wave on composite beach: 1+1 models that perform well with the solitary wave experiments must still be tested with the composite beach geometry (see Appendix A3.2 for details). This additional test will ensure that the code is stable enough for large waves that are near the breaking limit offshore ($\tilde{H}/\tilde{d} = 0.696$). Numerical predictions should not differ by more than 5% from the experimental values, and the numerical procedure should be capable of predicting the entire runup variation.

Conical island: 2+1 dimensional calculations should be tested with the conical island geometry of 1:4 slope (Appendix A3.3). The numerical method should stably model two wave fronts that split in front of the island and collide behind it. Predictions of the runup on the back of the island where the two fronts collide should not differ by more than 20% from the laboratory measurements.

Monai Valley: 2+1 numerical computations should then be tested with the laboratory model of Monai Valley, Okushiri Island, Japan (Appendix A3.4). The initial condition is a leading depression N-wave, and the entire

simulation shows how well the code performs in a rapid sequence of withdrawal and runup. Comparison of results from different codes has shown that the maximum runup in these experiments can be calculated within 10%, which is thus the standard.

Landslide: As discussed, landslide wave generation remains the frontier in terms of numerical modeling, particularly for aerial slides. These involve not only the rapid change of the seafloor, but also the impact of the slide on the shoreline. Therefore numerical codes that will be used to model landslide-generated tsunamis need to be tested against three-dimensional landslide experiments given in Appendix A3.5. It is expected to estimate maximum runup predictions within 10% of the experiments.

2.4 Field data benchmarking

Why is field data benchmarking important? Verification of a model in a real-world setting is an important part of model validation, especially for operational models. No analytical or laboratory data comparisons (or any limited number of tests, for that matter) can assure robust model performance in the operational environment. Test comparisons with real-world data provide an additional important step in the validation of a model to perform well during operational implementation. The main challenge of testing a model against real-world events is to overcome uncertainties of the tsunami source. While the source of the wave is deterministic in the controlled setting of the laboratory experiment and can be usually reproduced precisely, field data always have uncertainties in the source. For tsunamis, deep ocean measurements are the most unambiguous data quantifying the source of a tsunami.

Only a very limited number of high-quality deep-ocean tsunami measurements exist and these do not include data for Pacific tsunamis that have been destructive or caused inundation (Titov *et al.*, 2005). No DART tsunameters existed in the Indian Ocean at the time of the megatsunami, since DART had only been deployed in the Pacific Ocean. Satellite altimetry measurements of the Indian Ocean tsunami do not provide enough quality and coverage to constrain the tsunami source. Hydrodynamic inversion remains an ill-posed problem and criteria for its regularization are lacking. Deep ocean measurements allow for more defensible inversions, since they are not affected by local coastal effects. Several events have been recorded by both deep-ocean and coastal gages in the Pacific that allow reasonably constrained comparison with models. The expanded DART system array will be providing more tsunami measurements for future events, expanding the library of well-constrained propagation scenarios for testing.

The 10 June 1996 Andreanov Islands (Tanioka and González, 1998) and the 4 October 1994 Kuril Islands (Yeh *et al.*, 1995) events were recorded by several research tsunameters (without real-time data transmission) at similar locations offshore of Alaska and the U.S. West Coast. The offshore model scenario for the Andreanov Island event was obtained from the forecast database by inverting data from research Bottom Pressure Recorders (BPRs) as described in Titov *et al.* (2005). The inversion of the Kuril Islands

data was performed using BPR gages in similar locations. The 17 November 2003 $M_w = 7.8$ Rat Islands tsunami was recorded in real time by the DART tsunameter system and provides a good operational test of field data.

High quality bathymetry/topography data, source definition, and field runup measurements suggest using Okushiri Island, Japan, tsunami data as one of the field benchmarks (Appendix A4.1). In addition, the Rat Islands, Alaska, offshore model scenario constrained by the deep-ocean measurement can be used as input for testing high-resolution inundation models for Hilo Bay, Hawaii, where tide gage recording provides data for model comparison (Appendix A4.2).

Okushiri Island: 2+1 computations of the field measurements from the Hokkaido-Nansei-Oki tsunami around Okushiri Island, Japan (Appendix A4.1). The initial condition to be used is DCRC-17a (Takahashi, 1996), which is a composite fault with three segments. The bathymetry and measurements are provided in Appendix A4.1. Predictions for the maximum runup at Aonae, Okushiri Island, Japan, should not differ by more than 20% from the measurements.

17 November 2003 Rat Islands tsunami: For operational codes, testing should invert the tsunameter signal of the 17 November 2003 Rat Islands tsunami to improve the initial estimate of sea surface displacement derived from a seismic deformation model, then use the results as input to a Hilo, Hawaii, inundation model to hindcast the tide gage record. This is the most difficult but most realistic test for any operational model, for it involves a forecast (now hindcast) and has to be done much faster than real time. Here, at least the first four waves must be simulated and compared with the observations, with amplitudes and periods accurate to within 25%, maximum amplitude accurate to within 10%, and an arrival time error less than 3 min. Details of the problem are given in Appendix A4.2.

2.5 Scientific evaluation

Peer-review documentation: Model validation and verification is a continuing process. Any model used for inundation mapping or operational forecasts needs to be presented in peer-reviewed scientific journals with impact factors greater than 0.5. One or more of these publications should include benchmark comparisons identical or similar to those described above.

Formal scientific evaluation: A formal evaluation process of individual models needs to be established to avoid ad hoc decisions as to the suitability of any given model. This process may include solicitation of additional reviews of the model's veracity by experts, or the requirement that additional testing be performed. This process will set the standard for the best available practice at any given time, and it will hopefully eliminate the liability to NOAA, universities, states, engineers, and geophysicists who collaborate on the development of inundation maps.

2.6 Operational evaluation

The operational evaluation should be done by a test-bed consisting of research and operational parts of NOAA. While the scientific evaluation process may identify models that are realistic and computationally correct, some models may still not be appropriate for operational inundation mapping or forecasting applications. An additional evaluation process must be established to assess a number of model features that bear on important operational factors, such as special implementation hardware/software issues, ease of use, computation time, etc.

Operational evaluation of candidate real-time forecast models should include the following steps:

Step 1—*Meet Operational Forecasting and Inundation Mapping Requirements and Objectives*: Operational requirements include: basic forecasting computation, analysis and visualization tools; integration with operations (vs. separate, stand-alone applications); basic data assimilation techniques; computational resources needed to meet milestones; etc. If a candidate model does not meet specified NOAA forecasting or inundation mapping requirements and objectives, it should be rejected at this point.

Step 2—*Meet Modular Development Requirements*: Various pieces of the forecast model must be developed in parallel based on the overall objectives defined in Step 1.

Step 3—*Meet Test Bed and Model Standards*: In this step, the candidate model is tested against operational standards, with special attention given to the models ability to simulate previous, major tsunamis with the required speed and accuracy. Based on these test results, forecast model development may return to Step 2, proceed, or the candidate model may be rejected for operational use.

Step 4—*Meet Operational Testing Requirements*: The candidate model is integrated into the operational setting for testing. Potential sources are defined and the model is tested in a forecasting mode on an operational platform. Graphical interfaces are developed and forecast models are applied to a few cases to test operational integration and important individual factors such as speed, accuracy, and reliability (see Section 3, below). Operational testing and feedback is provided by the TWCs at this point, and adjustments are made as necessary.

Step 5—*Implement Operationally*: The model is fully integrated into the operational setting and procedures to provide operational products.

3. Criteria for Evaluating Operational Forecasting and Inundation Mapping Models

Given the accumulated experience in the tsunami community in the past 50 years, it is now possible to describe the requirements for an ideal tsunami model. Given an earthquake fault mechanism and tsunameter data, the ideal model should accurately predict tsunami inundation of at-risk coastlines in a sufficiently short time. Sufficiently short is defined as the time interval

between the initiation of the tsunami and the calculation of the inundation forecast that allows for evacuation of the target communities. For example, the State of Hawaii needs about 3 hr for a complete and orderly evacuation. An ideal model would accurately forecast inundation at least 3 hr before the tsunami impact is expected anywhere in Hawaii.

What are the requirements for building such an ideal model, and what constraints need to be imposed for model selection and/or further development? The following sections define and provide rationale for these constraints.

3.1 Model computational time constraints

The forecast speed will always depend on the distance from the source to the forecast community. However, an ideal forecast would provide a real-time forecast which, after being transmitted to local authorities, allows at least a few minutes before the tsunami impacts the nearest target community. This time interval would allow sirens to trigger the evacuation of beaches and coastal residents and give emergency personnel time to mobilize resources and prepare for search and rescue.

Further, an ideal model would correctly predict the duration of the event. Tsunamis are a series of waves. Tsunamis often get trapped in closed bays or on the continental shelf, resulting in sea level oscillations that may persist for several hours. During the 1993 Okushiri Island, Japan, inundation, bay oscillation at Aonae trapped the tsunami for over 30 min, and a large portion of Aonae remained submerged for a large portion of this time. The Crescent City, California, harbor oscillated for more than 4 hr following the 15 November 2006 tsunami. An operational forecast must ideally provide an estimate of the time that it is safe for search and rescue operations to begin without endangering the lives of responders.

Computational speed standards for inundation maps and real-time forecasts can be different. The current best-practice standard in the U.S. is to use a combination of real-time forecast models at coarse resolution and integrate real-time tsunameter measurements to update the initial sea surface forecast (Titov *et al.*, 2005; Whitmore, 2003). Then a fine-resolution inundation model, developed beforehand and placed on stand-by, ready to be run, uses the output of the fast coarse-resolution computation to produce a quick but physically realistic inundation forecast. The latter is referred to as a Stand-by Inundation Model (SIM).

A long-term forecast for inundation map production is produced months or years before a tsunami strikes; computational time is not the limiting step here. When conducting inundation mapping, the effort is concentrated on acquiring the best available bathymetry/topography and initial conditions to produce the most accurate model results. The objective is to provide guidance for evacuation planning and other hazard assessment products. Inundation mapping is performed at the highest resolution the available resources allow. The current standard is 50 m grid resolution, but site-specific features may demand even finer resolution. Successful computations at resolutions down to 5 m have been performed.

3.2 Model accuracy constraints

The accuracy of any given model depends on how well the computational procedure represents the correct solution of the parent equations of motion. When exact solutions exist (as, for example, for certain cases of the linear and nonlinear shallow water-wave equations), the determination of the accuracy of a solution algorithm is straightforward, i.e., through comparisons of the numerical results with the analytical predictions. Determining maximum runup numerically within 5% of the analytical solution is now possible with a handful of models.

For most bathymetries of geophysical interest, analytical solutions do not exist, and it is unlikely that they will ever be determined, due to the complexity of the physical terrain. However, a few laboratory models at smaller scale than the prototype exist: for example, the scale model of Okushiri Island, as described earlier. The Catalina Island, Los Angeles, 2004 model validation workshop of the National Science Foundation identified a handful of models that could predict the laboratory measurements within 10%. While greater compliance with measurements is hoped for in the next decade, 10% accuracy is achievable now and should be considered a standard.

While a numerical model may be validated through comparisons with analytical results and laboratory data, this does not necessarily imply physical realism for tsunamis of geophysical scales. One example is wave breaking. While a numerical model may realistically approximate the solution of the Navier-Stokes equations at laboratory scales, it may not do so at large scales. Calculating the evolution of breaking waves involves calculating turbulent shear terms and invoking turbulence closure constraints which are scale dependent. Therefore a reliability constraint needs to be applied, and this is discussed in the next subsection.

For operational forecast models, propagation accuracy of 5% is now possible (Titov *et al.*, 2005). For inundation models, accuracy of 10% with respect to analytical results and laboratory data is also now possible. Both should be considered as standards.

An associated accuracy constraint is grid resolution. This depends on the complexity of the shoreline. On a fairly plane and very long beach such as those of Southern California, a 100 m grid resolution may be sufficient. The smallest offshore and onshore features likely to affect tsunami impact on a coastal community should be reflected in the numerical grids. If a community is fronted by a sand spit of width 100 m, at least four grid points are needed to provide accurate resolution of the flow over the spit, implying a resolution of less than 25 m. If a coastline is sparsely populated, a 100 m resolution may be sufficient for satisfactory inundation maps, even when the shoreline is complex.

3.3 Model reliability and realism constraints

Model reliability refers to how well a given model predicts inundation consistently and realistically from a geophysical point of view. Linear theory may predict wave evolution consistently, but unfortunately not always in

a realistic manner. Linear theory predicts that waves during shoaling will keep growing in height, whereas in reality waves will eventually break, if large enough and when exceeding threshold height-to-depth and height-to-wavelength ratios.

The issue of reliability is crucial. Several numerical models now widely used include ad hoc friction factors. Given that these factors were developed not to model the physical manifestation of frictional dissipation, but simply to stabilize what is by its very nature a marginally stable computation, it is not possible to know a priori how well a model that has been fairly successful in a small number of cases performs in more general cases. For example, a model developed and calibrated for stability with examples from steep coastlines with less than 200 m inundation distances may not perform equally well when employed to forecast the inundation from a tsunami that penetrates more than 3 km inland, as the 26 December 2004 Boxing Day tsunami did in Banda Aceh.

It is clear that any numerical model for inundation predictions needs to be tested over a variety of scales from the laboratory to prototype to ensure both reliability and realism. Ideally, inundation models should be continuously tested with every new set of laboratory data or tsunami field data that becomes available. This will also allow for their further improvement. On the other hand, operational propagation forecasting is by its nature less dependent on scale. Some of the existing methods to test for model accuracy, reliability, and realism are discussed in Appendix A.

4. Summary of Model Evaluation Procedures

State-of-the-art inundation codes in use today have evolved through a painstaking process of careful validation and verification, while operational codes have been developed through extensive verification with measurements from real tsunamis, to the point that every new event poses a diminishing challenge. Mining this experience, procedures for approval and application of numerical models for operational uses are proposed as follows:

1. Establishment of standards for model validation and verification;
2. Scientific evaluation of individual models;
3. Operational evaluation of individual models;
4. Development of operational applications for forecasting;
5. Procedures for transfer of technology to operations.

Only through parallel testing of models under identical conditions, as when there is a tsunami emergency and an operational forecast is performed, can the community determine the relative merits of different computational formulations, an important step to further improvements in speed, accuracy, and reliability.

Figure 1 illustrates the process of model development and evaluation for operational use. The process of operational model development should be

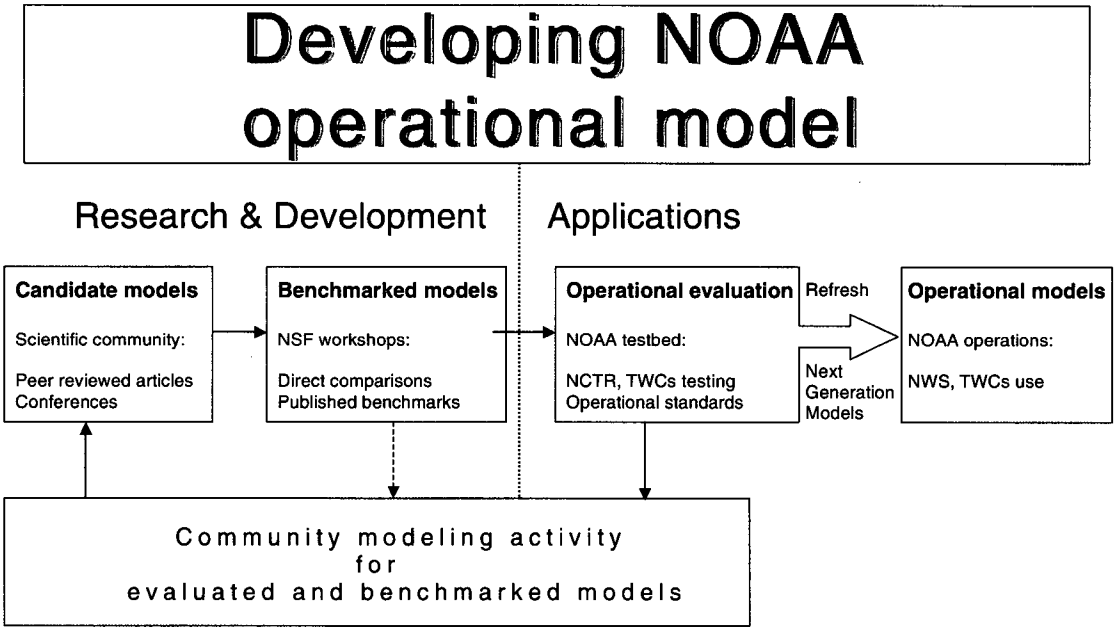

Figure 1: Flowchart illustrating the process of NOAA operational model development.

as inclusive as possible, to ensure the active participation of the geophysical, oceanographic, engineering, and NOAA research communities. Currently, scientific evaluation of a tsunami model takes the form of peer-reviewed publications and NSF tsunami model benchmark workshops. NOAA research and operations will evaluate the tsunami model operationally. The Community Modeling Activity will solidify these evaluations into a sustainable developmental effort that will feed NOAA operations with new and refreshed models. NOAA's responsibility is to test benchmarked models for operational suitability to define if a model fits the NOAA operational standards. This evaluation is done by operational and research components of NOAA as a part of the Research-to-Application implementation plan. The models—those that fit NOAA's operational standards—may be implemented for operations. The models that do not fit the standards (due to accuracy, speed, or robustness deficiencies) will stay in the research community modeling activity for improvement and further development.

It is emphasized again that model testing must remain a continuous process. Operational products produced in real time during an actual event must be thoroughly reviewed, and the operational models systematically tested in hindcast mode after each tsunami strikes. The results must be documented and reported to assist the community in developing and implementing improvements, through the identification and resolution of any serious problems or inadequacies of the models and/or products.

While this process may appear onerous, it reflects our state of knowledge as of December 2006, and is the only defensible methodology when human lives are at stake.

5. Acknowledgments

The authors thank Paul Whitmore (Director, West Coast and Alaska Tsunami Warning Center) for providing a draft of Section 2.6, "Operational evaluation." We also thank Paul and other reviewers of the manuscript for fruitful discussions and many useful comments that shaped the document into its current form. This work was partially funded by the U.S. National Tsunami Hazard Mitigation Program. We thank the National Science Foundation of the United States for supporting some of the early validation exercises through benchmark testing in three individual workshops, and for supporting the analytical studies and laboratory investigations that resulted in the benchmark data sets discussed in this report.

6. References

Borerro, J., M. Ortiz, V.V. Titov, and C.E. Synolakis (1997): Field survey of Mexican tsunami. *Eos Trans. AGU, 78*(8), 85, 87–88 (Eos Cover Article).

Briggs, M.J., C.E. Synolakis, G.S. Harkins, and D. Green (1995): Laboratory experiments of tsunami runup on a circular island. *Pure Appl. Geophys., 144*, 569–593.

Carrier, G.F., and H.P. Greenspan (1958): Water waves of finite amplitude on a sloping beach. *J. Fluid Mech., 17*, 97–110.

Carrier, G.F. (1966): Gravity waves of water of variable depth. *J. Fluid Mech., 24*, 641–659.

Carrier, G.F., T.T. Wu, and H. Yeh (2003): Tsunami runup and drawdown on a sloping beach. *J. Fluid Mech., 475*, 79–99.

Goring, D.G. (1978): Tsunamis—the propagation of long waves onto a shelf. W.M. Keck Laboratory of Hydraulics and Water Resources, California Institute of Technology, Report No. KH-R-38.

Grilli, S.T., I.A. Svenden, and R. Subrayama (1997): Breaking criterion and characteristics of solitary waves on a slope. *J. Waterw. Port Coast. Ocean Eng., 123*(2), 102–112.

Hall, J.V., and J.W. Watts (1953): Laboratory investigation of the vertical rise of solitary waves on impermeable slopes. Tech. Memo. 33, Beach Erosion Board, U.S. Army Corps of Engineers, 14 pp.

Hammack, J.L. (1972): Tsunamis—A model for their generation and propagation. W.M. Keck Laboratory of Hydraulics and Water Resources, California Institute of Technology, Report No. KH-R-28.

Kânoğlu, U. (1998): The runup of long waves around piecewise linear bathymetries. Ph.D. Thesis, University of Southern California, Los Angeles, California, 90089-2531, 273 pp.

Kânoğlu, U. (2004): Nonlinear evolution and runup–rundown of long waves over a sloping beach. *J. Fluid Mech., 513*, 363–372.

Kânoğlu, U., and C.E. Synolakis (1998): Long wave runup on piecewise linear topographies. *J. Fluid Mech., 374*, 1–28.

Kânoğlu, U., and C. Synolakis (2006): Initial value problem solution of nonlinear shallow water-wave equations. *Phys. Rev. Lett., 97*, 148,501–148,504, doi: 10.1103/PhysRevLett.97.148501.

Keller, J.B., and H.B. Keller (1964): Water wave run-up on a beach. ONR Research Report NONR-3828(00), Dept. of the Navy, Wash. DC, 40 pp.

Li, Y., and F. Raichlen (2000): Energy balance model for breaking solitary wave runup. *J. Waterw. Port Coast. Ocean Eng., 129*(2), 47–49.

Li, Y., and F. Raichlen (2001): Solitary wave runup on plane slopes. *J. Waterw. Port Coast. Ocean Eng., 127*(1), 33–44.

Li, Y., and F. Raichlen (2002): Non-breaking and breaking solitary run-up. *J. Fluid Mech., 456*, 295–318.

Liu, P.L.-F., Y.-S. Cho, M.J. Briggs, U. Kânoğlu, and C.E. Synolakis (1995): Runup of solitary waves on a circular island. *J. Fluid Mech., 320*, 259–285.

Liu, P.L.-F., P. Lynett, and C.E. Synolakis (2003): Analytical solutions for forced long waves on a sloping beach. *J. Fluid Mech., 478*, 101–109.

Liu, P.L.-F., C.E. Synolakis, and H.H. Yeh (1991): Report on the International Workshop on Long Wave Runup. *J. Fluid Mech., 229*, 675–688.

Liu, P.L.-F., T.-R. Wu, F. Raichlen, C.E. Synolakis, and J. Borrero (2005): Runup and rundown generated by three-dimensional sliding masses. *J. Fluid Mech., 536*, 107–144.

Liu, P.L.-F., H. Yeh, and C. Synolakis (2007): Advanced Numerical Models for Simulating Tsunami Waves and Runup. *Advances in Coastal and Ocean Engineering, 10*, 250 pp.

National Science and Technology Council (2005): Tsunami Risk Reduction for the United States: A Framework for Action. A Joint Report of the Subcommittee on Disaster Reduction and the United States Group on Earth Observations, 30 pp.

Pedersen, G., and B. Gjevik (1983): Runup of solitary waves. *J. Fluid Mech., 135*, 283–299.

Russel, J.S. (1845): Report on Waves. *Rp. Meet. Brit. Assoc. Adv. Sci. 14th*, 311–390, John Murray, London.

Shuto, N. (1973): Shoaling and deformation of nonlinear waves. *Coastal Eng. Japan, 16*, 1–12.

Synolakis, C.E. (1986): The Runup of Long Waves. Ph.D. Thesis, California Institute of Technology, Pasadena, California, 91125, 228 pp.

Synolakis, C.E. (1987): The runup of solitary waves. *J. Fluid Mech., 185*, 523–545.

Synolakis, C.E., and J.E. Skjelbreia (1993): Evolution of maximum amplitude of solitary waves on plane beaches. *J. Waterw. Port Coast. Ocean Eng., 119*(3) 323–342.

Synolakis, C.E., and E.N. Bernard (2006): Tsunami science before and after Boxing Day 2004. *Phil. Trans. R. Soc. A, 364*(1845), doi: 10.1098/rsta.2006.1824, 2231–2265.

Tadepalli, S., and C.E. Synolakis (1994): The runup of N-waves on sloping beaches. *Proc. R. Soc. Lond. A. 445*, 99–112.

Tadepalli, S., and C.E. Synolakis (1996): Model for the leading waves of tsunamis. *Phys. Rev. Lett., 77*(10), 2141–2144.

Takahashi, T. (1996): Benchmark problem 4; the 1993 Okushiri tsunami—Data, conditions and phenomena. In *Long-Wave Runup Models*, World Scientific, 384–403.

Titov, V.V., and C.E. Synolakis (1995): Modeling of breaking and nonbreaking long-wave evolution and runup using VTCS-2. *J. Waterw. Port Ocean Coast. Eng., 121*(6), 308–316.

Titov, V.V., and C.E. Synolakis (1997): Extreme inundation flows during the Hokkaido–Nansei-Oki tsunami. *Geophys. Res. Lett., 24*(11), 1315–1318.

Titov, V.V., and F.I. González (1997): Implementation and testing of the method of splitting tsunami (MOST). NOAA Technical Memorandum ERL-PMEL-112, PB98-122773, Pacific Marine Environmental Laboratory, Seattle, WA, 11 pp.

Titov, V.V., and C.E. Synolakis (1998): Numerical modeling of tidal wave runup. *J. Waterw. Port Ocean Coast. Eng., 124*(4), 157–171.

Titov, V.V., F.I. González, E.N. Bernard, M.C. Eble, H.O. Mofjeld, J.C. Newman, and A.J. Venturato (2005): Real-time tsunami forecasting: Challenges and solutions. *Nat. Haz., 35*(1), 45–58.

Tuck, E.O., and L.S. Hwang (1972): Long wave generation on a sloping beach. *J. Fluid Mech., 51*, 449–461.

Tanioka, Y., and F.I. González (1998): The Aleutian earthquake of June 10, 1996 (Mw 7.9) ruptured parts of both the Andreanof and Delarof Segments. *Geophys. Res. Lett., 25*(12), 2245–2248.

Whitmore, P.M. (2003): Tsunami amplitude prediction during events: A test based on previous tsunamis. In *Science of Tsunami Hazards, 21*, 135–143.

Yeh H., P.L.-F. Liu, M. Briggs, and C.E. Synolakis (1994): Tsunami catastrophe in Babi Island. *Nature, 372*, 6503–6508.

Yeh, H., P.L.-F. Liu, and C.E. Synolakis (1996): *Long-Wave Runup Models.* World Scientific, 403 pp.

Yeh, H., V.V. Titov, V. Gusyiakov, E. Pelinovsky, V. Khramushin, and V. Kaistrenko (1995): The 1994 Shikotan earthquake tsunami. *Pure Appl. Geophys., 144*(3/4), 569–593.

Zelt, J.A. (1991): The runup of breaking and nonbreaking solitary waves. *Coastal Eng., 125*, 205–246.

Appendix A: Existing Methods for Model Validation and Verification

Benchmarking of numerical models can be classified into four categories: Basic hydrodynamic considerations, analytical benchmarking, experimental benchmarking, and field benchmarking. Here, specific benchmark problems for validating and verifying computational tools for predicting the coastal effect of tsunamis are described in detail. Some of the benchmark problems described here were used as benchmark problems in the 1995 (Yeh *et al.*, 1996) and 2004 (Liu *et al.*, 2007) Long-Wave Runup Models Workshops in Friday Harbor, Washington, and Catalina Island, Los Angeles, California, respectively.

1 Basic hydrodynamic considerations

Two most basic steps are required in ensuring that a numerical model works for predicting evolution and inundation. While the first step is ensuring that the model conserves mass, the second basic step is checking convergence of the numerical code to a certain asymptotic limit.

1.1 Mass conservation

The conservation of mass equation is part of the equations of motion that are solved in any numerical procedure, but cumulative numerical approximations can sometimes produce results that violate mass conservation. This is particularly the case when friction factors are used, or smoothing to stabilize inundation computations for breaking waves.

Conservation of mass can be checked by calculating the water volume at the beginning and at the end of the computation. This should be done by integrating disturbed water depth $\eta(x, y, t)$ over the entire flow domain, i.e., if the flow domain extends from the maximum penetration during inundation $x = X_{max}$ to the outer location of the source region X_S, and $y = Y_{max}$ to Y_S, then total displaced volume $V(t)$ is

$$V(t) = \int_{X_{max}}^{X_S} \int_{Y_{max}}^{Y_S} \eta(x, y, t) dx dy. \tag{A1}$$

The integral of $\eta(x, y, t)$ should be used instead of the integral of the entire flow depth $h(x, y, t) = \eta(x, y, t) + d(x, y, t)$—where $d(x, y, t)$ is the undisturbed water depth—because the latter is likely to conceal errors in the calculation. Typically, $\eta \ll d$ at offshore integrating h will simply produce the entire volume of the flow domain and will mask errors. Note that testing of the conservation of mass as above involves placing a closed domain within reflective boundaries.

Numerical models use absorbing boundaries to allow the wave to radiate outwards. While some loss of mass may in theory occur due to the differencing at the boundary, placing the computational boundaries far enough from the source ensures this loss is minimal. Nonetheless, testing conservation

of mass with open boundaries and monitoring the volume fluxes over the corresponding boundaries at $x = x_b$ and $y = y_b$ is computationally possible. The net flux needs to be added or subtracted to the total displaced volume.

The calculation of the entire volume of the wave motion of the flow domain at the beginning and end of the computation—to ensure that mass is conserved—is very important in code validation. Once the associated mass fluxes at the boundaries are considered, numerical errors can be highly additive, and mass invariably decreases in numerical computations. It is essential that codes self-monitor the entire volume; if the difference between start and finish is greater than 5%, the grid needs to be readjusted. Usually with few changes in Δx, Δy, and Δt, the conservation of mass can be improved.

Calculations of conservation of mass should be such that the total initial displaced volume $V(t = 0)$ should be within 5% of the total displaced volume at the end of the computation $V(t = T)$ where T represents the computation end time. It is assumed that the end of the computation is when the initial wave is entirely reflected and offshore.

1.2 Convergence

The next basic step is checking convergence of the numerical code to a certain asymptotic limit, presumably the actual solution of the equations solved. The grid steps Δx and Δy need to be halved, and the time step Δt reduced appropriately to conform with the Courant–Friedrics–Levy (CFL) criterion. The optimal locations to check convergence are the extreme runup and rundown. A graph needs to be prepared presenting the variation of the calculated runup and rundown (ordinate) with the step size (abscissa). As the step size is reduced, the numerical predictions should be seen to converge to a certain value, and further reductions in step size should not change the results.

2 Analytical benchmarking

The shallow water-wave (SW) equations describe the evolution of the water-surface elevation and of the depth-averaged water particle velocity of waves with wavelengths large compared with the depth of propagation. The equations assume that the pressure distribution is hydrostatic everywhere, i.e., there is no variation with depth of any of the other flow variables. In this section we present several analytic solutions to the 1+1 version of these equations. As stated in Section 2 of this report, 1+1 models are generally unreliable for inundation mapping and entirely inadequate for operational tsunami forecasting, but they are invaluable to the process of testing and validating models.

2.1 Single wave on a simple beach

The so-called canonical problem of the shallow water-wave equations is the calculation of a long wave climbing up a sloping beach of angle β coupled to a constant-depth region (Fig. A1). The origin of the coordinate system is at

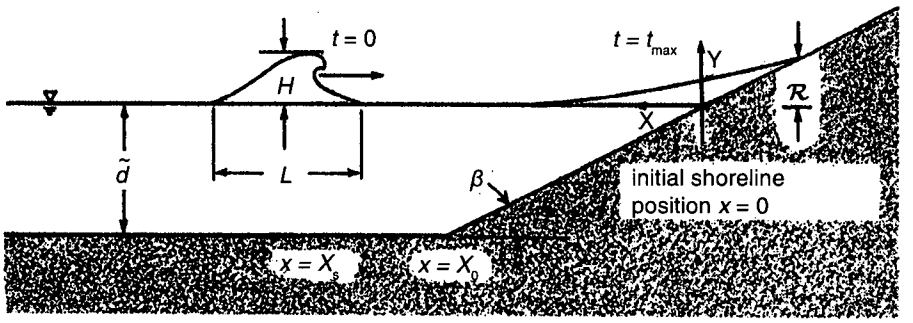

Figure A1: Definition sketch for canonical bathymetry, i.e., sloping beach connected to a constant-depth region.

the initial position of the shoreline and x increases seaward. Dimensionless variables are introduced as follows:

$$x = \frac{\tilde{x}}{\tilde{d}}, \quad (\eta,\ h_0) = \frac{(\tilde{\eta},\ \tilde{h}_0)}{\tilde{d}}, \quad u = \frac{\tilde{u}}{\sqrt{\tilde{g}\tilde{d}}}, \text{ and } t = \frac{\tilde{t}}{\sqrt{\tilde{d}/\tilde{g}}}. \tag{A2}$$

Here quantities with tilde are dimensional and η is the amplitude, u is the depth-averaged horizontal velocity, h_0 is the undisturbed water depth, $\tilde{d}$ is the depth of the constant-depth region, and $\tilde{g}$ is the gravitational acceleration. The topography is described by $h_0(x) = x\tan\beta$ when $x \le X_0 = \cot\beta$ and $h_0(x) = 1$ when $x \ge X_0 = \cot\beta$. Even though dimensionless variables are not preferred in engineering practice, here they have distinct advantages as everything scales simply with an offshore characteristic depth. In numerical solutions, dimensional variables are most often used.

Consider a tsunami evolution problem described by the $1+1$ nonlinear shallow water-wave (NSW) equations:

$$\begin{aligned} h_t + (uh)_x &= 0, \\ u_t + uu_x + \eta_x &= 0, \end{aligned} \tag{A3}$$

with $h(x,t) = \eta(x,t) + h_0(x)$. Through elementary manipulations, neglecting nonlinear terms, (A3) reduces to

$$\eta_{tt} - (\eta_x h_0)_x = 0, \tag{A4}$$

an equation known as the linearized shallow water-wave (LSW) equation. Over constant-depth $h_0 = 1$, then

$$\eta_{tt} - \eta_{xx} = 0, \tag{A5}$$

the classic one-dimensional wave equation.

The solution follows directly from the Fourier transform of the equation when a boundary condition for the wave amplitude $\eta(X_1, t)$ is specified, i.e.,

when the incident wave $\eta(x,t)$ at the constant-depth region is known at some $x = X_1$, and can be described by a Fourier integral of the form

$$\eta(X_1,t) = \int_{-\infty}^{+\infty} \Phi(\omega) e^{-i\omega t} d\omega. \tag{A6}$$

Consider the canonical problem of a constant-depth region of depth $h_0 = 1$ joined to a sloping beach of angle β, with the toe of the beach at $x = X_0$. Synolakis (1986, 1987) matched the linear theory solution at the constant-depth with the solution over the sloping beach as derived by Keller and Keller (1964) to derive the solution over the sloping beach for a wave with a transform as given by (A6),

$$\eta(x,t) = 2\int_{-\infty}^{+\infty} \Phi(\omega) \frac{J_0(2\omega\sqrt{xX_0}) e^{-i\omega(X_0+t)}}{J_0(2X_0\omega) - iJ_1(2X_0\omega)} d\omega, \tag{A7}$$

where $X_0 = \cot\beta$. This solution is only valid when $0 \le x \le X_0$; when $x < 0$, (A4) does not reduce to Bessel's equation. Notice that the integral (A7) can be evaluated with standard numerical methods; however, the advantage of this form is that it allows calculation of the solution for many physically realistic tsunami waveforms simply by plugging in the $\Phi(\omega)$ of the incoming wave, hopefully known at some offshore location X_1.

2.1.1 Solitary wave evolution and runup As discussed in Synolakis (1986, 1987) it is possible to derive exact results for the evolution and runup of solitary waves based on linear theory. Solitary waves have long been used as a model for the leading wave of tsunamis. Solitary waves were first described by Russel (1845) as the great waves of translation, and consist of a single elevation wave. While capturing some of the basic physics of tsunamis, they do not model the physical manifestation of tsunamis in nature, which are invariably N-wave like with a leading-depression wave followed by an elevation wave. A solitary wave centered offshore at $x = X_s$ at $t = 0$ has the following surface profile,

$$\eta(x,0) = H\mathrm{sech}^2\gamma(x - X_s), \tag{A8}$$

where $\gamma = \sqrt{3H/4}$ and H is the dimensionless wave height, i.e., $H = \widetilde{H}/\widetilde{d}$. The function $\Phi(\omega)$ associated with this profile is derived in Synolakis (1986) and it is given by:

$$\Phi(\omega) = \frac{2}{3}\omega\mathrm{cosech}(\alpha\omega) e^{i\omega X_s}, \tag{A9}$$

where $\alpha = \pi/(2\gamma)$.

In the context of water-wave theory, the solitary wave (A8) is an exact solution of the Korteweg-de-Vries (KdV) equation; therefore, a KdV solitary wave propagates over constant-depth without any change in shape. The KdV theory is both dispersive and nonlinear, and solitary waves are the only waves with this unique property of unchanging shape. However, (A8) can be used as an initial condition for other wave theories, without, of course, a priori expectation that the SW model will preserve the classic

soliton properties, which include their ability to go through each other (interact) without any change in shape through nonlinear interactions. This having been said, since the LSW is nondispersive and linear, hence all waves propagate over constant-depth without any change in shape. However, in the range of wave steepness and amplitudes relevant for tsunamis, it is now well established that, at least for the $1+1$ problem far from the shoreline, the LSW theory, which also preserves the wave shape for propagation over constant-depth, is quite adequate (Liu *et al.*, 1991) and useful when the engineering problem has simple geometry.

The derivation of the amplitude evolution for solitary waves is not as straightforward as often assumed. When superposing sinusoids, there is a frequency ω-dependent phase shift. It is therefore not obvious that linear superposition will produce a similar amplitude variation given this frequency-dependent phase shift.

To describe the evolution of a solitary wave up a plane beach, Synolakis (1986, 1987), substituted (A9) into (A7) to obtain

$$\eta(x,t) = \frac{4}{3}\int_{-\infty}^{+\infty} \omega \mathrm{cosech}(\alpha\omega)\frac{J_0(2\omega\sqrt{xX_0})e^{-i\omega(X_0-X_s+t)}}{J_0(2X_0\omega) - iJ_1(2X_0\omega)}d\omega, \qquad \text{(A10)}$$

where, as earlier, $\alpha = \pi/\sqrt{3H}$. This integral can be evaluated directly through contour integration. In the region where the wave evolves on the sloping beach far off the shoreline, x is large, and (A10) becomes

$$\eta(x,t) = \frac{4\pi^2}{3\alpha^2}(\frac{X_0}{x})^{1/4}\sum_{n=1}^{+\infty}(-1)^{n+1}ne^{-(\pi/a)\theta' n}, \qquad \text{(A11)}$$

with $\theta' = X_0 - X_s - t - 2\sqrt{xX_0}$. The maximum of the power series is $1/4$, therefore the maximum local value of the wave amplitude $\eta_{\max}$ is given explicitly by

$$\frac{\eta_{\max}}{H} = (\frac{X_0}{x})^{1/4} = (\frac{1}{h_0})^{1/4}; \qquad \text{(A12)}$$

this is an amplitude variation similar to Green's law.

The region over which (A12) applies is the region of gradual shoaling; the region of rapid shoaling is often identified with the Boussinesq result, i.e., $\eta_{\max} \sim h$. The fact that both evolution laws may coexist was first identified by Shuto (1973). Synolakis and Skjelbreia (1993) also present results which show that Green's law type evolution is valid over a wide range of slopes and for finite-amplitude waves at least in the region of gradual shoaling.

The results (A10) can now be readily applied to derive a result for the maximum runup of a solitary wave climbing up a sloping beach. Writing $R(t) = \eta(0,t)$, i.e., $R(t)$ is the free-surface elevation at the initial shoreline; in the LSW theory, the shoreline does not move beyond $x = 0$. The maximum value of $R(t)$ is the maximum runup $\mathcal{R}$, arguably the most important parameter in the long-wave runup problem, and it is the maximum vertical excursion of the shoreline. Per Synolakis (1986), from equation (A10), it can

be deduced that

$$R(t) = 8H \sum_{n=1}^{+\infty} \frac{(-1)^{n+1} n e^{-2\gamma(X_s - X_0 - t)n}}{I_0(4\gamma X_0 n) + I_1(4\gamma X_0 n)}. \tag{A13}$$

The series can be simplified further by using the asymptotic form for large arguments of the modified Bessel functions. The resulting series is of the form $s = \sum_{n=1}^{\infty}(-1)^{n+1} n^{3/2} \chi^n$; its maximum value $s_{\max}$ occurs at $\chi = 0.481 = e^{-0.732}$ with $s_{\max} = 0.15173$. Then the following expression results for the maximum runup $\mathcal{R}$:

$$\mathcal{R} = 2.831 \sqrt{\cot\beta} H^{5/4}. \tag{A14}$$

This result is formally correct when $\sqrt{H} \gg 0.288 \tan\beta$—the assumption implied when using the asymptotic form of the Bessel functions. Equation (A14) was first derived by Synolakis (1986) and has since been referred to as *the runup law* and shown in Fig. A2. As will be apparent in later sections, this methodology is quite powerful to find the maximum runup and it allows calculation of the runup of other waveforms such as N-waves, not to mention the runup of waves evolving over piecewise-linear bathymetries. Recent results suggest that the dependence of the runup on the slope and on the offshore wave height in a two-dimensional problem of idealized conditions is often quite similar to this one-dimensional power law.

The asymptotic result (A14) is valid for waves that do not break during runup, suggesting that it is appropriate to use the qualifier nonbreaking for waves that do not break during runup but may or may not break during rundown. The real usefulness of any asymptotic result is how well it identifies the scaling, i.e., it can identify the solution dependence on the problem parameters; numerical solutions will invariably produce more accurate specific predictions, but they can rarely provide useful information about the problem scaling. To check if the runup law (A14) provides the correct scaling, Synolakis (1986, 1987) examined the classic laboratory data set of Hall and Watts (1953), Fig. A2. That study includes both breaking and nonbreaking wave data without identifying them as such, clearly because there was no realization of the differences; the empirical runup relationships derived by Hall and Watts (1953) are not directly applicable when determining the runup of nonbreaking waves. To perform a posteriori identification of those data, the breaking criterion $H < 0.479(\cot\beta)^{-10/9}$ was used.

2.1.2 N-wave runup Most tsunami eyewitness accounts suggest that tsunamis are N-wave like, i.e., they are dipolar, which means they appear as a combination of a depression and an elevation wave, and frequently as a series of N-waves, sometimes known as double N-waves. Up until the late 1990s, the solitary wave model was used exclusively to evaluate the runup of tsunamis. The N-wave model was motivated by observations from earthquakes in Nicaragua [1 September 1993], Flores, Indonesia [12 December 1992], Okushiri, Japan [7 July 1993], East Java, Indonesia [6 June 1994], Kuril Islands, Russia [4 October 1994], Mindoro, Philippines [14 November 1994], Manzanillo, Mexico [9 October 1995], Chimbote, Peru [3 March 1996], Papua New Guinea [17 July 1998], Vanuatu [26 November 1999], and

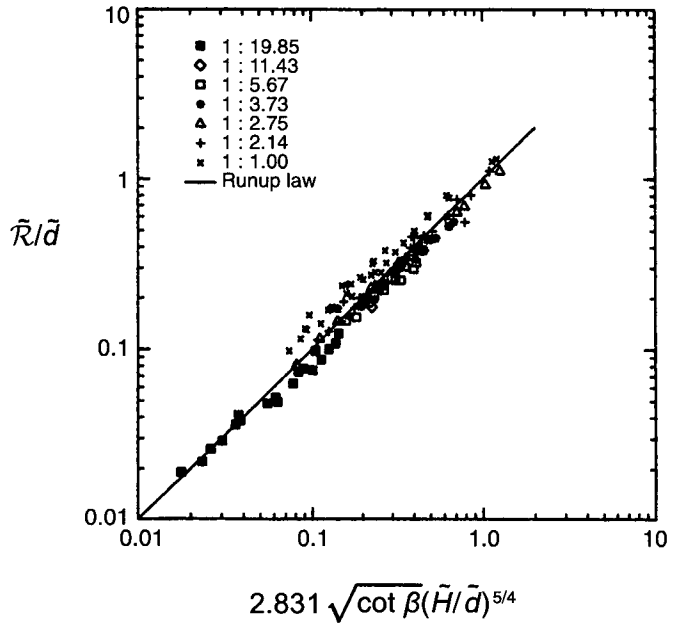

Figure A2: Laboratory data for maximum runup of nonbreaking waves climbing up different beach slopes: ■ 1:19.85 (Synolakis, 1986), ◊ 1:11.43 (Hall and Watts, 1953), □ 1:5.67 (Hall and Watts, 1953), ∗ 1:3.73 (Hall and Watts, 1953), △ 1:2.75 (Pedersen and Gjevik, 1983), + 1:2.14 (Hall and Watts, 1953), × 1:1.00 (Hall and Watts, 1953). Solid line represents the runup law (A14).

Caminade, Peru [20 June 2001], all of which produced tsunami waves which caused nearby shorelines to first recede before advancing. The most specific description was during the 9 October 1995 Manzanillo, Mexico earthquake. One eyewitness saw the shoreline retreat beyond a rock outcrop which was normally submerged in over 5 m depth and at a distance of about 400 m from the shoreline, suggesting a leading-depression wave. Before the megat-sunami of 26 December 2004, this was the only photographic evidence of these leading-depression waves (Borrero *et al.*, 1997). Recall that the megat-sunami manifested itself first with a rapid withdrawal of the water surface in most locales east of the rupture zone.

Modeling tsunamis with solitary waves cannot possibly explain these observations, because a solitary wave is technically a leading-elevation wave. Therefore, and to reflect the fact that tsunamigenic faulting in subduction zones associated with both vertical uplift and subsidence of the sea bottom, Tadepalli and Synolakis (1994) conjectured that all tsunami waves at generation have an N-wave or dipole shape. Tadepalli and Synolakis (1994, 1996) proposed a general function as a unified model for both near-shore and far-field tsunamis as generalized N-waves. When a wave propagates with the trough first it is referred to as a leading depression N-wave or LDN. When the crest arrives first, it is a leading-elevation wave or LEN. Also, Tade-

palli and Synolakis (1994) defined another type of N-wave of this class with leading-elevation and depression waves of the same height and at a constant separation distance and refer to this wave as an isosceles N-wave with a surface profile given by

$$\eta(x,0) = \frac{3\sqrt{3}H}{2}\mathrm{sech}^2[\gamma(x-X_N)]\tanh[\gamma(x-X_N)], \tag{A15}$$

with $\gamma = \frac{3}{2}\sqrt{\sqrt{\frac{3}{4}}H}$. Wave profile (A15) is an LDN and has a maximum wave amplitude H. The function $\Phi(\omega)$ associated with this profile is derived in Tadepalli and Synolakis (1994) using contour integration and it is given by:

$$\Phi(\omega) = \frac{2i}{3\gamma}\omega^2\mathrm{cosech}(\frac{\pi\omega}{2\gamma})e^{i\omega X_N}. \tag{A16}$$

Now, the maximum of $R(t) = \eta(0,t)$ can be evaluated for LEN using the symmetry of the profile given in (A15) and one can find that

$$\mathcal{R}_{\mathrm{N-wave}} = 3.86\sqrt{\cot\beta}H^{5/4}. \tag{A17}$$

Comparing the runup of the Boussinesq solitary wave (A14) with the runup of an isosceles N-wave, $\mathcal{R}_{\mathrm{N-wave}} = 1.364\mathcal{R}_{\mathrm{Solitary}}$. Because of the symmetry of the profile, this is also the minimum rundown of an isosceles leading-depression N-wave. Tadepalli and Synolakis (1994) showed that the normalized maximum runup of nonbreaking isosceles LEN is smaller than the runup of isosceles LDN, and that both are higher than the runup of a solitary wave with the same wave height, and the latter is known as the N-wave effect (Fig. A3).

The two-dimensional character of the generation region limits the direct application of the N-wave and solitary wave models. However, N-wave theory does provide a conceptual framework for analysis and for explaining certain field observations qualitatively.

2.1.3 Boundary value problem To solve the nonlinear set (A3) for the sloping beach case, $h_0(x) = x\tan\beta$, Carrier and Greenspan (1958) introduced the hodograph transformation known as Carrier–Greenspan transformation:

$$u = \frac{\psi_\sigma}{\sigma}, \tag{A18}$$

$$\eta = \frac{\psi_\lambda}{4} - \frac{u^2}{2}, \tag{A19}$$

$$t = \cot\beta(\frac{\psi_\sigma}{\sigma} - \frac{\lambda}{2}), \tag{A20}$$

$$x = \cot\beta(\frac{\sigma^2}{16} - \frac{\psi_\lambda}{4} + \frac{u^2}{2}), \tag{A21}$$

into (A3), and they derived the following second-order ordinary differential equation:

$$(\sigma\psi_\sigma)_\sigma = \sigma\psi_{\lambda\lambda}. \tag{A22}$$

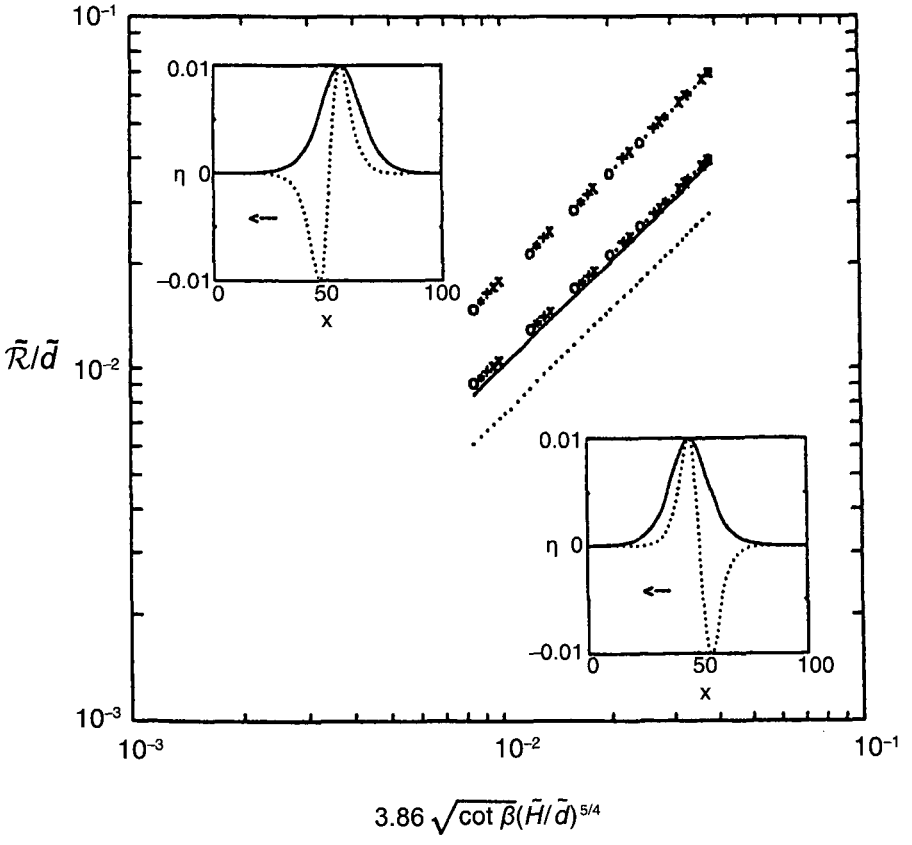

Figure A3: Maximum runup of isosceles N-waves and solitary wave. Top and lower set of points are results for the maximum runup of leading-depression and -elevation isosceles N-waves, respectively. Dotted line represents the runup of solitary wave (A14). Lower and upper insets compare a solitary wave profile to a leading-depression and -elevation isosceles N-waves, respectively.

Note that the original Carrier–Greenspan transformation does not include $\cot\beta$ in (A20) and (A21) because of a different scaling as given in (A25) in section A2.1.4. Note also the similarity with the linear form of the shallow water-wave equation (LSW), $(\eta_x h_0)_x = \eta_{tt}$. Also, notice the conservation of difficulty. Instead of having to solve the coupled nonlinear set (A3), one now has to solve a linear equation, but the transformation equations which relate the transformed variables with the physical variables are nonlinear, coupled, and implicit. Yet, a redeeming feature is that in the hodograph plane, i.e., in the (σ, λ)-space, the shoreline is always at $\sigma = 0$. This allows for direct analytical solutions without the complications of the moving shoreline boundary.

In general, it is quite difficult to specify initial or boundary data for the nonlinear problem in the physical (x, t)-space coordinates without making restrictive assumptions; a boundary condition requires specification of the solution at $(X_0, \forall t)$, and an initial condition specification at $(t_0, \forall x)$, but, in

practice, the wave approaching the beach is only known offshore for ($X_0 \geq \cot\beta, t < t_0$), where t_0 is the time that the wave reaches the x-location X_0. Even when boundary or initial conditions are available in the (x,t)-space, the process of deriving the equivalent conditions in the (σ,λ)-space is not trivial.

These difficulties have restricted the use of this transformation to problems that can be reduced directly to those solved by Carrier and Greenspan (1958). Synolakis (1986) revived the Carrier–Greenspan formalism by developing a method to specify a boundary condition including reflection. Synolakis (1986) used the solution of the equivalent linear problem, as given by (A7), at the seaward boundary of the beach, i.e., at $x = X_0 = \cot\beta$ corresponding to $\sigma = \sigma_0 = 4$. Then, equation (A19) implies that $\eta(X_0,t) \approx \frac{1}{4}\psi_\lambda(4,\lambda)$. Assuming that $\psi(\sigma_0,\lambda) \to 0$ as $\lambda \to \pm\infty$, Synolakis (1986, 1987) showed that the Carrier–Greenspan potential is given by

$$\psi(\sigma,\lambda) = -\frac{16i}{X_0}\int_{-\infty}^{+\infty}\frac{\Phi(\kappa)}{\kappa}\frac{J_0(\sigma\kappa X_0/2)e^{-i\kappa X_0(1-\frac{\lambda}{2})}}{J_0(2\kappa X_0) - iJ_1(2\kappa X_0)}d\kappa. \qquad \text{(A23)}$$

Even though the solution now can be obtained in the (σ, λ)-space using (A18) and (A19) and can be converted to the solution in the (x,t)-space through (A20) and (A21), the problem with this transformation is deriving a solution for a particular-time t^* or at a particular-location x^*. Synolakis (1986) and later Kânoğlu (2004) evaluated the solution either for given t^* or at given x^* using the Newton–Raphson iteration algorithms, respectively;

$$\lambda_{i+1} = \lambda_i - \left[\frac{t^* - t(\lambda)}{\frac{\partial}{\partial\lambda}\left[t^* - t(\lambda)\right]}\right]_{\lambda_i} \quad \text{or} \quad \sigma_{i+1} = \sigma_i - \left[\frac{x^* - x(\sigma)}{\frac{\partial}{\partial\sigma}\left[x^* - x(\sigma)\right]}\right]_{\sigma_i}. \qquad (A24a, b)$$

Here $t(\lambda)$ and $x(\sigma)$ are given with (A20) and (A21), respectively.

An astonishing feature of the NSW is that the predictions for the maximum runup are identical to those of the LSW, when identical boundary conditions are specified at $X_0 = \cot\beta$. The maximum runup according to LSW is the maximum value attained by the wave amplitude at the initial position of the shoreline, while the maximum runup is given by the maximum value of the amplitude at the evolving shoreline $\eta(x_s,\lambda)$, where x_s is the x-coordinate of the shoreline tip and corresponds to $\sigma = 0$. Carrier (1966)—without reflection—and Synolakis (1987)—including reflection—have shown that the linear and nonlinear theory produce mathematically identical predictions. Nonlinear analytical evolution of solitary wave with $H = 0.0185$ over a sloping beach of 1:19.85 is given in Fig. A4.

2.1.4 Initial value problem The nonlinear evolution of a wave over a sloping beach is theoretically and numerically challenging due to the moving boundary singularity. Yet, it is important to have a good estimate of the shoreline velocity and associated runup–rundown motion, since they are crucial for the planning of coastal flooding and of coastal structures. As explained in the previous section, Synolakis (1987) solved this problem as a boundary value problem considering canonical bathymetry. Kânoğlu (2004)

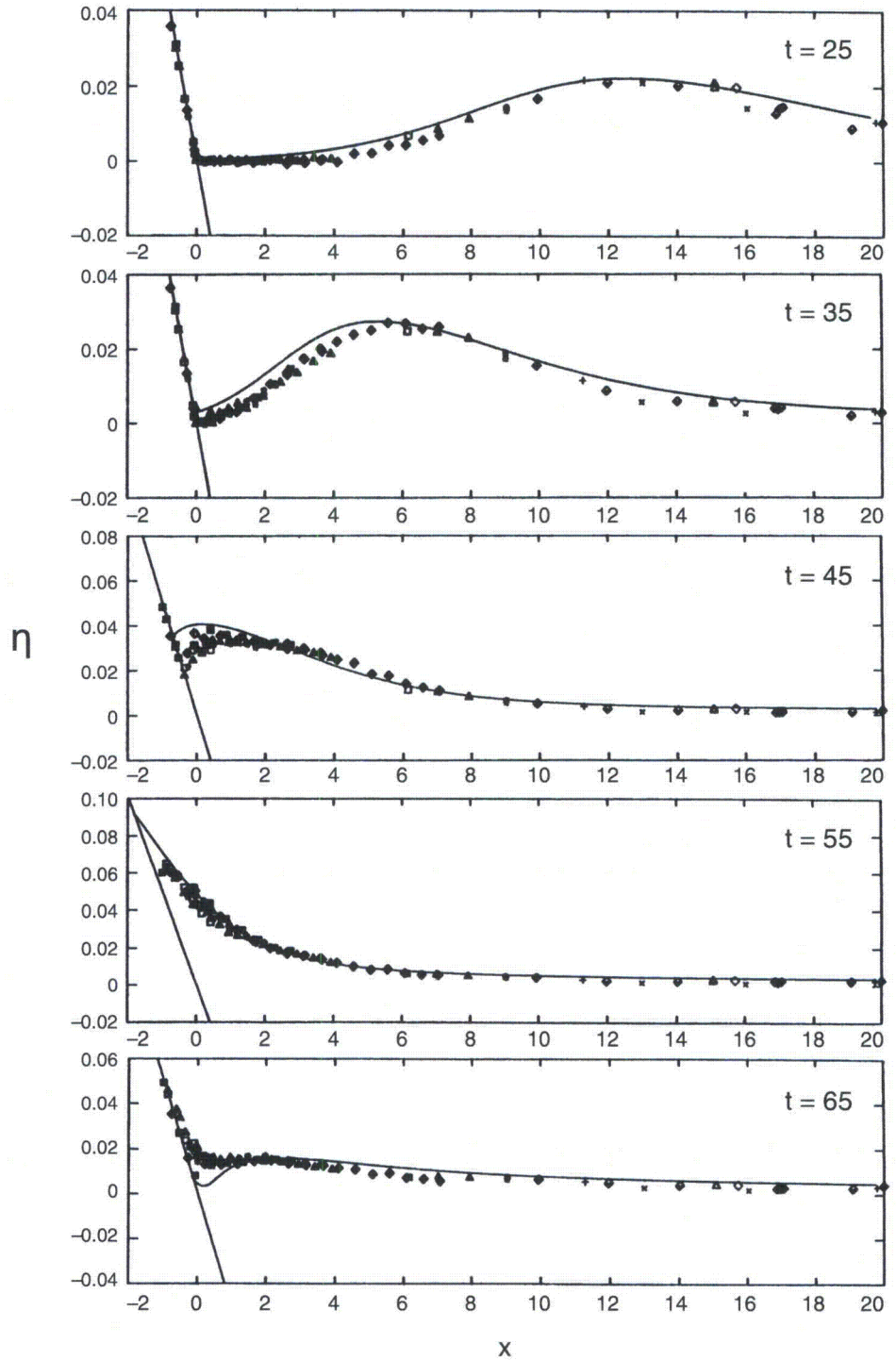

Figure A4: Time evolution of $H = 0.0185$ initial wave over a sloping beach with $\cot\beta = 19.85$ from $t = 25$ to 65 with 10 increments. Constant depth-segment starts at $X_0 = 19.85$. While markers show experimental results of Synolakis (1986, 1987), solid lines show nonlinear analytical solution of Synolakis (1986, 1987).

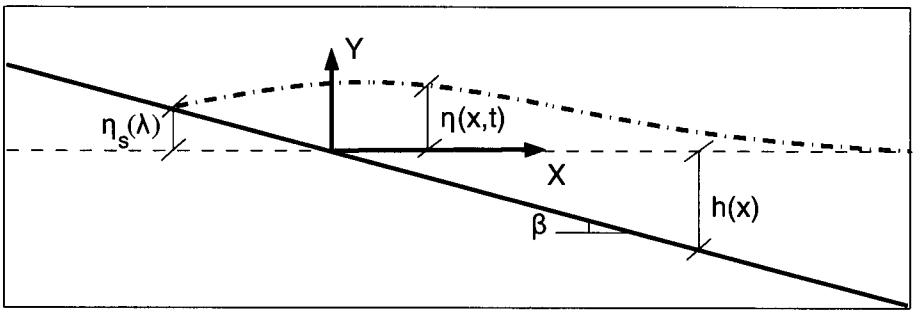

Figure A5: Definition sketch for the initial value problem.

solved nonlinear evolution of any given wave form over a sloping beach as an initial value problem (Fig. A5). It is proposed that any initial waveform can first be represented in the transform space using the linearized form of the Carrier–Greenspan transformation for the spatial variable, then the nonlinear evolutions of these initial waveforms can be directly evaluated. Later, Kânoğlu and Synolakis (2006) solved the similar problem considering a more general initial condition, i.e., initial wave with velocity.

Kânoğlu (2004) considers NSW equations (A3) with slightly different nondimensionalization than (A2), i.e., using the reference length $\widetilde{l}$ as a scaling parameter, the dimensionless variables are introduced as

$$x = \frac{\widetilde{x}}{\widetilde{l}},\ (h,\ \eta) = \frac{(\widetilde{h},\ \widetilde{\eta})}{\widetilde{l}\tan\beta}, u = \frac{\widetilde{u}}{\sqrt{\widetilde{g}\,\widetilde{l}\tan\beta}},\ \text{and}\ t = \frac{\widetilde{t}}{\sqrt{\widetilde{l}/(\widetilde{g}\,\tan\beta)}}\ . \quad \text{(A25)}$$

Using the original Carrier–Greenspan transformation—without $\cot\beta$ coefficient in (A20) and (A21)—it is possible to reduce the NSW equations to the following single second-order linear equation the same as (A22):

$$\sigma\phi_{\lambda\lambda} - (\sigma\phi_\sigma)_\sigma = 0, \quad \text{(A26)}$$

using the Riemann invariants of the hyperbolic system (Carrier and Greenspan, 1958). The Carrier–Greenspan transformation not only reduces the nonlinear shallow water-wave equations into a second-order linear equation, but also fixes the instantaneous shoreline to $\sigma = 0$ in the (σ, λ)-space as explained previously. Furthermore, a bounded solution at the shoreline combined with a given initial condition in terms of a wave profile at $\lambda = 0$ in the $(\sigma,\ \lambda)$-space, $\eta(\sigma, 0)$ implies the following solution in the transform space,

$$\phi(\sigma,\lambda) = -\int_0^\infty \int_0^\infty \frac{1}{\omega}\xi^2\Phi(\xi)J_0(\omega\sigma)J_1(\omega\xi)\sin(\omega\lambda)\mathrm{d}\omega\mathrm{d}\xi, \quad \text{(A27)}$$

where $\Phi(\sigma) = u_\lambda(\sigma, 0) = 4\eta_\sigma(\sigma,\ 0)/\sigma$. Further, given the initial waveform

$\eta(\sigma,0)$, the evolution of the water-surface elevation is now given by

$$\eta(\sigma,\lambda) = \frac{1}{4}\phi_\lambda - \frac{1}{2}u^2$$
$$= -\frac{1}{4}\left\{\int_0^\infty \xi^2\Phi(\xi)\left[\int_0^\infty J_0(\omega\sigma)J_1(\omega\xi)\cos(\omega\lambda)\mathrm{d}\omega\right]\mathrm{d}\xi\right\}$$
$$-\frac{1}{2}\left\{\int_0^\infty \xi^2\Phi(\xi)\left[\int_0^\infty \frac{J_1(\omega\sigma)}{\sigma}J_1(\omega\xi)\sin(\omega\lambda)\mathrm{d}\omega\right]\mathrm{d}\xi\right\}^2 \quad \text{(A28)}$$

where, again, $\Phi(\sigma) = 4\,\eta_\sigma(\sigma,0)/\sigma$.

Since it is important for coastal planning, simple expressions for shoreline runup–rundown motion and velocity are useful. Considering the shoreline corresponds to $\sigma = 0$ in the (σ, λ)-space, (A28) reduces to the following equation for the runup–rundown motion:

$$\eta_s(\lambda) = \eta(0,\lambda) = \frac{1}{4}\phi_\lambda - \frac{1}{2}u_s^2$$
$$= -\frac{1}{4}\left\{\int_0^\infty \xi^2\Phi(\xi)\left[\int_0^\infty J_1(\omega\xi)\cos(\omega\lambda)\mathrm{d}\omega\right]\mathrm{d}\xi\right\}$$
$$-\frac{1}{2}\left\{\int_0^\infty \xi^2\Phi(\xi)\left[\int_0^\infty \frac{1}{2}\omega J_1(\omega\xi)\sin(\omega\lambda)\mathrm{d}\omega\right]\mathrm{d}\xi\right\}^2. \quad \text{(A29)}$$

Here u_s and η_s represent shoreline velocity and motion, respectively. The singularity of the $u = \phi_\sigma/\sigma$ at the shoreline ($\sigma = 0$) is removed with the consideration of the $\lim_{\sigma\to 0}[J_1(\omega\sigma)/\sigma] = \frac{1}{2}\omega$.

The difficulty of deriving an initial condition in the (σ, λ)-space is overcome by simply using the linearized form of the hodograph transformation for a spatial variable in the definition of initial condition. It is proposed that any initial waveform can first be represented in the transform space using the linearized form of the Carrier–Greenspan transformation for the spatial variable ((A21) without $\cot\beta$ coefficient), then the nonlinear evolutions of these initial waveforms can be directly evaluated. Once an initial value problem is specified in the (x,t)-space as $\eta(x,0)$, the linearized hodograph transformation $x \cong \frac{1}{16}\sigma^2$ is used directly to define the initial waveform in the (σ, λ)-space, $\eta(\frac{1}{16}\sigma^2, 0)$. Thus $\Phi(\sigma) = 4\eta_\sigma(\frac{1}{16}\sigma^2,\,0)/\sigma$ is found, and $\phi(\sigma,\lambda)$ follows directly through a simple integration, as in (A28). Then it becomes possible to investigate any realistic initial waveform such as Gaussian and N-wave shapes employed in Carrier *et al.* (2003) and the isosceles and general N-waves defined by Tadepalli and Synolakis (1994). Again, solution in the physical space can be found using the Newton–Raphson algorithm proposed by Synolakis (1987) and later used by Kânoğlu (2004), as presented in (A24a, b).

The shoreline runup-rundown motion and velocity are presented for one of the initial profiles given by Carrier *et al.* (2003):

$$\eta(x,0) = H_1\exp(-c_1(x-x_1)^2) - H_2\exp(-c_2(x-x_2)^2). \quad \text{(A30)}$$

The following initial profile can be obtained in the transform space after using the linearized form of the transformation for the spatial variable:

$$\eta(\sigma,0) \approx H_1\exp(-\frac{1}{256}c_1(\sigma^2-\sigma_1^2)^2) - H_2\exp(-\frac{1}{256}c_2(\sigma^2-\sigma_2^2)^2), \quad \text{(A31)}$$

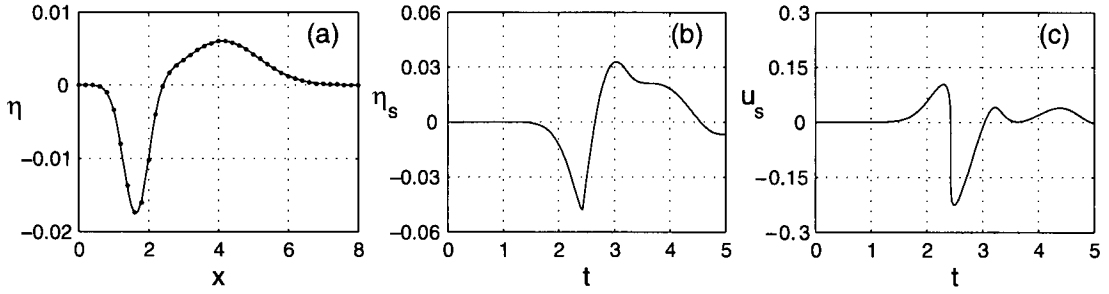

Figure A6: (a) The leading-depression initial waveform (A30) presented by Carrier *et al.* (2003) with $H_1 = 0.006$, $c_1 = 0.4444$, $x_1 = 4.1209$, $H_2 = 0.018$, $c_2 = 4.0$, and $x_2 = 1.6384$ (solid line) compared with the one resulting from approximation (evaluated through (A28)) (dots), (b) shoreline wave height, and (c) shoreline velocity.

which leads to the definition of the initial condition $\Phi(\sigma)$:

$$\Phi(\sigma) = -\frac{1}{16} H_1\, c_1\, (\sigma^2 - \sigma_1^2)\, \exp(-\frac{1}{256} c_1 (\sigma^2 - \sigma_1^2)^2) + \frac{1}{16} H_2\, c_2\, (\sigma^2 - \sigma_2^2)\, \exp(-\frac{1}{256} c_2 (\sigma^2 - \sigma_2^2)^2). \quad \text{(A32)}$$

Figure A6a compares the initial waveforms defined in the physical space as in (A30) with the one resulting from the approximation of it, i.e., (calculated through (A28)). The linearized form of the spatial variable in the definition of the initial waveforms gives satisfactory comparison. Figures A6b and A6c present the shoreline runup–rundown motion and velocity, respectively, calculated from equation (A29) using the corresponding parts. It should be added that the solution presented here cannot be evaluated when the Jacobian of the transformation, $J = x_\sigma\, t_\lambda - x_\lambda\, t_\sigma$, breaks down. Even though the transformation might become singular at certain points, the solution can still be obtained at other points, since local integration can be performed without prior knowledge of the dependent variables, unlike in numerical methods. This feature is discussed in detail in Synolakis (1987) and Carrier *et al.* (2003), and is not explained further in here.

2.2 Solitary wave on composite beach

Most topographies of engineering interest can be approximated by piecewise-linear segments allowing the use of LSW to determine approximate analytical results for the wave runup of more complicated waveforms, in closed form. In principle, fairly complex bathymetry can be represented through a combination of positive and/or negative segments and constant-depth segments. Solutions of the LSW (A4) at each segment can be matched analytically at the transition points between the segments, and then the overall amplification factor and reflected waves can be determined, analytically. We consider here an application to check this assertion with a complex topography consisting of three segments and a vertical wall (Fig. A7). Laboratory data

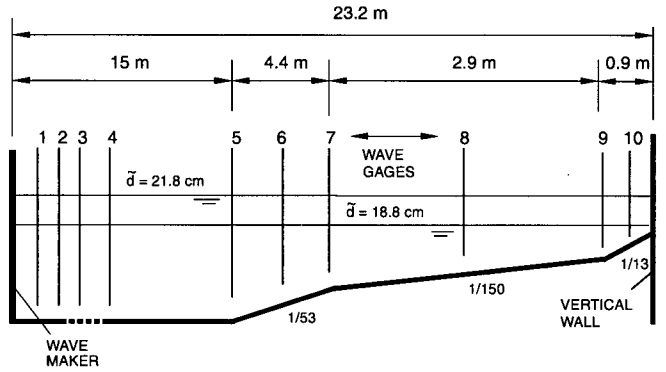

Figure A7: Definition sketch for Revere Beach. Not to scale. Transition points and segments are numbered from shoreward to seaward, i.e., segment numbers 1, 2, 3, and 4 show 1/13, 1/150, 1/53, and constant-depth segments, transition points 1, 2, and 3 show transition points between 1/13 and 1/150, 1/150 and 1/53, and 1/53 and constant-depth segment, respectively.

exist for this topography from a U.S. Army Corps of Engineers, Coastal Engineering Research Center, Vicksburg, Mississippi experiment of wave runup on a model of Revere Beach, Massachusetts. This beach profile and the laboratory data are discussed in greater detail in Yeh *et al.* (1996), Kânoğlu (1998), and Kânoğlu and Synolakis (1998). The profile of the model consists of three piecewise-linear slopes of 1:13 (1st segment), 1:150 (2nd segment), and 1:53 (3rd segment) from shoreward to seaward. At the shoreline there is a vertical wall. In the laboratory experiments, to evaluate the overtopping of the seawall, the wavemaker was located 23.22 m and tests were done at two depths, at 18.8 cm and at 21.8 cm.

Kânoğlu (1998) and Kânoğlu and Synolakis (1998) associated constant-depth segment of depth h_r with the matrix:

$$D_{pr} = \begin{bmatrix} e^{-\frac{i\omega x_p}{\sqrt{h_r}}} & e^{\frac{i\omega x_p}{\sqrt{h_r}}} \\ ie^{-\frac{i\omega x_p}{\sqrt{h_r}}} & -ie^{\frac{i\omega x_p}{\sqrt{h_r}}} \end{bmatrix}, \tag{A33}$$

and the linearly varying depth region with positive slope m_r is associated with the matrix:

$$Q_{pr} = \begin{bmatrix} J_0(2\omega\sqrt{h_p}/m_r) & Y_0(2\omega\sqrt{h_p}/m_r) \\ J_1(2\omega\sqrt{h_p}/m_r) & Y_1(2\omega\sqrt{h_p}/m_r) \end{bmatrix}. \tag{A34}$$

Above, in equations (A33) and (A34), the first subscript p identifies the transition point, and the second subscript r identifies the segment, i.e., if a segment has two transition points, there are two associated 2×2 matrices. Here nondimensional quantities are defined as in (A2) using the depth of the 4th segment $\tilde{h}_4$ as the characteristic length scale.

Using (A33) and (A34), continuities of water surface elevation and its spatial derivative at the transition points lead to the following matrix equa-

tions:

$$Q_{11}V_1 = Q_{12}V_2,\ Q_{22}V_2 = Q_{23}V_3,\ \text{and}\ Q_{33}V_3 = D_{34}V_4, \tag{A35}$$

for each transition point. Here column vectors $V_r = [V_{r1}, V_{r2}]^T$ identify unknowns for each segment. One additional condition must be considered at the vertical wall, i.e., perfect reflection. Boundary condition at the wall requires V_1 to be a column vector considering perfect reflection, i.e., $Q_w = \left[1, -J_1(2\omega\sqrt{h_w}/m_1)/Y_1(2\omega\sqrt{h_w}/m_1)\right]^T$. $V_1 = Q_w V_{11}$ with h_w is the dimensionless water depth at the wall. Combining (A35) with perfect reflection gives

$$Q_w V_{11} = Q_{11}^{-1}Q_{12}Q_{22}^{-1}Q_{23}Q_{33}^{-1}D_{34}V_4. \tag{A36}$$

To evaluate time histories of the surface elevation, the following integral must be evaluated

$$\eta_r(x,t) = \int_{-\infty}^{+\infty} \Phi(\omega)\frac{1}{V_{41}} \left\{ \begin{array}{c} V_{r1}J_0(\frac{2\omega\sqrt{h_r(x)}}{m_r}) + V_{r2}Y_0(\frac{2\omega\sqrt{h_r(x)}}{m_r}) \\ V_{41}e^{-\frac{i\omega x}{\sqrt{h_4}}} + V_{42}e^{\frac{i\omega x}{\sqrt{h_4}}} \end{array} \right\} e^{-i\omega t}d\omega, \tag{A37}$$

where $r = 1, 2, 3$ and V_{41} and V_{42} represents incident and reflected wave amplitudes, respectively. The temporal variation of the water surface elevation can be calculated through (A37) evaluating the unknowns in terms of V_{41} through (A36) and (A35).

More interestingly, following analysis of Synolakis (1986) the maximum runup of solitary waves over Revere Beach can be calculated analytically given $\Phi(\omega)$ as in (A9). The amplitude at the shoreline is

$$R(t) = \eta(0,t) = -(4/3)\frac{m_1}{\pi\sqrt{h_w}}\int_{-\infty}^{+\infty} \frac{\text{cosech}(\alpha\omega)e^{i\omega(X_s - X_3 - t)}}{\varphi(\omega) + i\psi(\omega)}d\omega. \tag{A38}$$

Here $\varphi(\omega) + i\psi(\omega)$ is a complicated but determinable quite easily from expression (A35) and (A36) in terms of Bessel functions of zero and first order, using symbolic mathematical tools. Kânoğlu (1998) conjectured that $\varphi(z) + i\psi(z)$ is an entire function in the upper half plane, and derived the Laurent series expansion and its asymptotic form as

$$R(t) = 8h_w^{-1/4}H\sum_{n=1}^{+\infty}(-1)^{n+1}n\mathcal{X}^n, \tag{A39}$$

where $\mathcal{X} = e^{-\frac{\pi}{\alpha}(X_s - X_3 - t + 2\{\frac{\sqrt{h_1}-\sqrt{h_w}}{m_1} + \frac{\sqrt{h_2}-\sqrt{h_1}}{m_2} + \frac{\sqrt{h_3}-\sqrt{h_2}}{m_3}\})}$. This is a power series of the form $\sum(-1)^{n+1}n\mathcal{X}^n$ and its maximum is equal to $1/4$. Therefore the maximum runup for solitary waves propagating up Revere Beach is given by the runup law,

$$\mathcal{R} = 2h_w^{-1/4}H. \tag{A40}$$

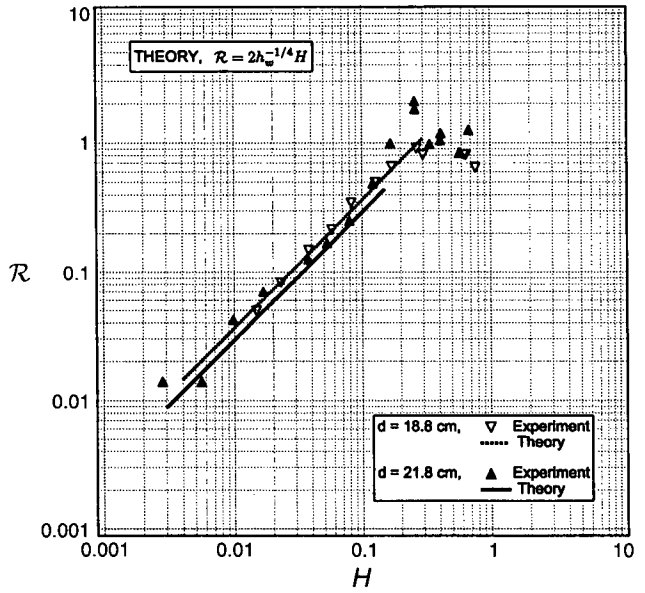

Figure A8: Comparison of maximum runup values for the linear analytical solution (A39) and laboratory results for two different depths, i.e., $\widetilde{d} = 18.8$ cm and $\widetilde{d} = 21.8$ cm.

The runup law above suggests that the maximum runup only depends on the depth at the seawall fronting the beach, and it does not depend on any of the three slopes in front of the seawall. The runup law (A40) predicts the nonbreaking data surprisingly well (Fig. A8).

Note that no Jacobian regularization conditions as yet exist for wave evolution on composite beaches, and this theory can only be applied with caution. For lack of a better method, it is recommended that a limiting height be determined for applying this methodology on a composite beach, by finding the limiting height for the single beach with slope equal to the least steep slope in the composite topography under consideration. This practice will produce adequate validity thresholds for bathymetries such as Revere Beach, but it should not be used when abrupt changes in bathymetry exist. In most cases, for the range of wave-steepness and beach slopes relevant in tsunami studies, this method will give good results with fairly small computational effort.

2.3 Subaerial landslide on simple beach

Liu *et al.* (2003) considered tsunami generation by a moving slide on a uniformly sloping beach, using the forced linear shallow water-wave equation as in Tuck and Hwang (1972);

$$\frac{\partial^2 \eta}{\partial t^2} - \frac{tan\beta}{\mu}\frac{\partial}{\partial x}(x\frac{\partial \eta}{\partial x}) = \frac{\partial^2 h_0}{\partial t^2}, \qquad \text{(A41)}$$

in the dimensionless form. Here $z = \eta(x,t)$ is the free surface elevation and $z = -h(x,t)$ the sea floor. On a sloping beach $h(x,t) = H(x) - h_0(x,t)$ with $H(x) = x\tan\beta/\mu$; $\tan\beta$ is the beach slope, $h_0(x,t)$ is the time-dependent perturbation of the sea floor with respect to the uniformly sloping beach, $\widetilde{\delta}$ and $\widetilde{L}$ are the maximum vertical thickness of the sliding mass and its horizontal length, respectively, and $\mu = \widetilde{\delta}/\widetilde{L}$. Normalization is done as

$$(\eta, h, H, h_0) = \frac{(\widetilde{\eta}, \widetilde{h}, \widetilde{H}, \widetilde{h}_0)}{\widetilde{\delta}}, \quad x = \frac{\widetilde{x}}{\widetilde{L}}, \text{ and } t = \frac{\widetilde{t}}{\sqrt{\widetilde{\delta}/\widetilde{g}/\mu}}. \tag{A42}$$

Here again quantities with tilde represent dimensional quantities. The focus is on thin slides where $\mu = \widetilde{\delta}/\widetilde{L} \ll 1$. Since the shallow water-wave assumption requires that $\tan\beta \ll 1$, then $\tan\beta/\mu \sim O(1)$.

Consider a translating Gaussian shaped mass, initially at the shoreline as in Liu *et al.* (2003), i.e., the seafloor can be described by $h_0(x,t) = \exp\left[-(\xi - t)^2\right]$ with $\xi = 2\sqrt{\mu x/\tan\beta}$. Once in motion, the mass moves at constant acceleration. Solution of (A41) under this translating Gaussian shape is given as

$$\eta(\xi,t) = \int_0^\infty J_0(\rho\xi)\rho\left[a(\rho)\cos(\rho t) + \frac{1}{\rho}b(\rho)\sin(\rho t)\right]\mathrm{d}\rho + \frac{1}{3}(h_0 - \xi\frac{\partial h_0}{\partial\xi}), \tag{A43}$$

in Liu *et al.* (2003) with

$$a(\rho) = \int_0^\infty \varsigma J_0(\rho\varsigma)\left[-\frac{1}{3}(h_0 - \xi\frac{\partial h_0}{\partial\xi})\right|_{t=0,\ \xi\to\varsigma}\mathrm{d}\varsigma, \tag{A44}$$

$$b(\rho) = \int_0^\infty \varsigma J_0(\rho\varsigma)\left[\frac{1}{3}(2\frac{\partial h_0}{\partial t} + \varsigma\frac{\partial^2 h_0}{\partial\xi\partial t})\right|_{t=0,\ \xi\to\varsigma}\mathrm{d}\varsigma. \tag{A45}$$

Two specific examples are presented in Figs. A9 and A10. Maximum runup estimates, maximum wave height at $x = 0$, from the analytical solution (A43) and the numerical solution of Liu *et al.* (2003) are given in Fig. A11.

3 Laboratory benchmarking

As a preamble, and before describing the benchmark laboratory tests, waves are generated in the laboratory by moving vertical paddles. While other generation methods exist, hydraulic paddles allow for the precise and repeatable specification of arbitrary trajectories. In one-dimensional wave channels—also known as wave flumes and sometimes wave tanks—there is usually only one paddle, perpendicular to the water surface. As the paddle displaces horizontally, free-surface waves are generated. In two dimensional wave basins, there are often multiple paddles that can move horizontally. The experiments to be described were conducted in wave basins with multiple paddles. This kind of generator is known as the snake generator. When the paddles move independently, its motion resembles the ophite motion of a snake.

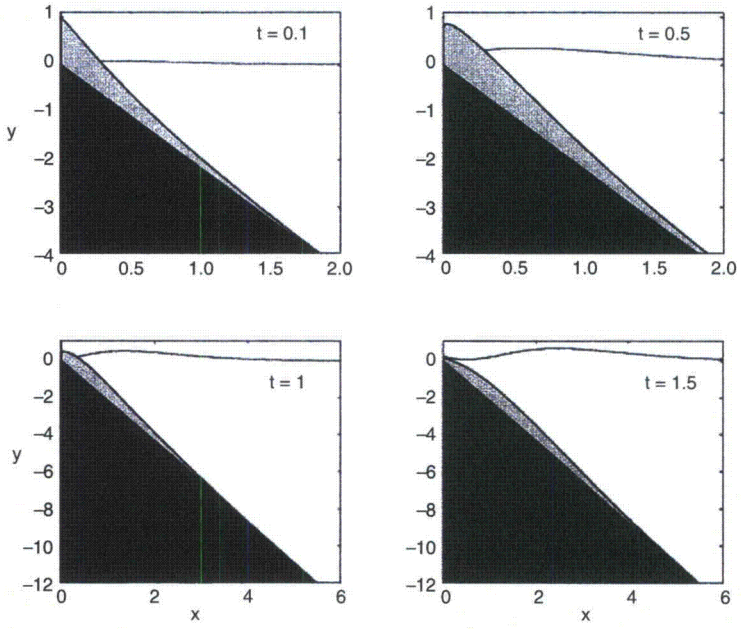

Figure A9: Spatial snapshots of the analytical solution at four different times for a beach slope, $\beta = 5°$, and landslide aspect ratio, $\mu = 0.05$ ($\tan\beta/\mu = 1.75$). The slide mass is indicated by the light shaded area, the solid beach slope by the black region, and η by the solid line (Liu *et al.*, 2003).

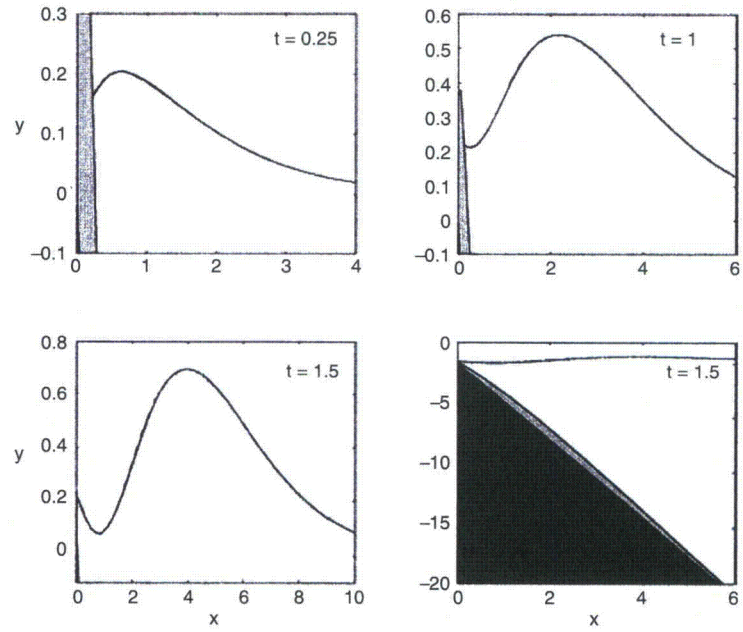

Figure A10: Spatial snapshots of the analytical solution at three different times for a beach slope, $\beta = 10°$, and $\mu = 0.05$ ($\tan\beta/\mu = 3.5$). The slide mass is indicated by the light shaded area, and the solid beach slope by the black region. In the lower right panel, the water depth and landslide profile is shown at $t = 1.5$, the same time as shown in the lower left plot (Liu *et al.*, 2003).

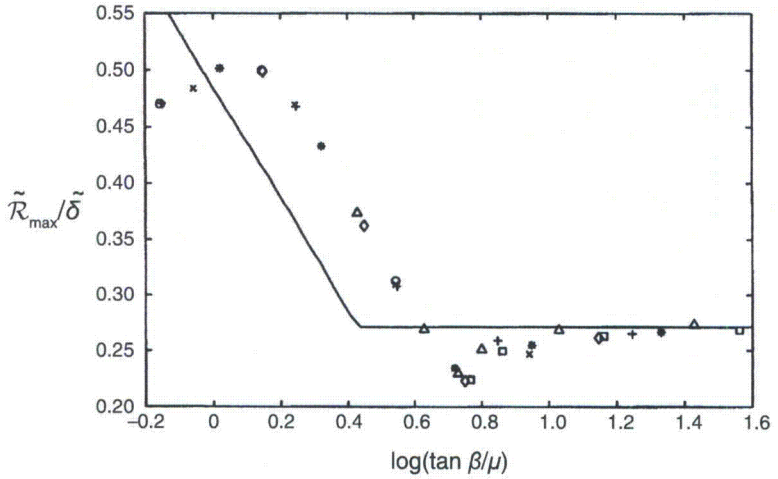

Figure A11: Maximum runup as a function of $\log(\tan\beta/\mu)$. The analytical solutions are shown by the solid line, and the various symbols are from NLSW simulations of Liu *et al.* (2003), corresponding to different slopes ranging from 2° to 20°.

They are also known as Directional Wave Spectrum Generators (DWSGs). In the U.S., such DWSGs exist at the Coastal and Engineering Laboratory of the U.S. Army Corps of Engineers in Vicksburg, Mississippi, at the Texas A&M University in College Station, Texas, and at Oregon State University in Corvallis, Oregon.

Here five sets of experiments will be described in detail for laboratory benchmarking data: solitary wave experiments on a 1:20 sloping beach (Synolakis, 1987), solitary wave runup over a composite beach (Kânoğlu, 1998; Kânoğlu and Synolakis, 1998), conical island experiments (Liu *et al.*, 1995; Kânoğlu, 1998; Kânoğlu and Synolakis, 1998), and tsunami runup onto a complex three-dimensional beach (Takahashi, 1996), and tsunami generation and runup due to three-dimensional landslide (Liu *et al.*, 2005).

3.1 Solitary wave on a simple beach

In this set of experiments, the 31.73 m-long, 60.96 cm-deep, and 39.97 cm-wide California Institute of Technology, Pasadena, California wave tank was used with water at varying depths. The tank is described by Hammack (1972), Goring (1978), and Synolakis (1986, 1987). The bottom of the tank consisted of painted stainless steel plates. Carriage rails run along the whole length of the tank, permitting the arbitrary movement of instrument carriages. A ramp was installed at one end of the tank to model the bathymetry of the canonical problem of a constant-depth region adjoining a sloping beach. The ramp had a slope of 1:19.85. The ramp was sealed to the tank side walls. The toe of the ramp was distant 14.95 m from the rest position of the piston generator used to generate waves.

A total of more than 40 experiments with solitary waves running up the sloping beach were performed, with wave depths ranging from 6.25 cm to 38.32 cm. Solitary waves are uniquely defined by their maximum height $\widetilde{H}$ to depth $\widetilde{d}$ ratio and the depth, i.e., $\widetilde{H}/\widetilde{d}$ and $\widetilde{d}$ are sufficient to specify the wave. $\widetilde{H}/\widetilde{d}$ ranged from 0.021 to 0.626. Breaking occurs when $\widetilde{H}/\widetilde{d} > 0.045$, for this particular beach. This is the same set of experiments used to validate the maximum runup analytical predictions presented in section A2.1.1.

Initial location, X_s in the analytical consideration (A8), changes with different wave heights. The reason X_s distance varies is that solitary waves of different heights have different effective "wavelengths." A measure of the "wavelength" of a solitary wave is the distance between the point x_f on the front and the tail x_t where the local height is 1% of the maximum, i.e., $\eta(x_f, t = 0) = \eta(x_t, t = 0) = (\widetilde{H}/\widetilde{d})/100$. The distance X_s is at an offshore location where only 5% of the solitary wave is already over the beach, so that scaling can work. Therefore, in the laboratory experiments initial wave heights are identified at a point $X_s = X_0 + (L/2)$ where $L/2 = (1/\gamma)\,\text{arccosh}\,\sqrt{20}$ with $\gamma = \sqrt{3(\widetilde{H}/\widetilde{d})/4}$. In the laboratory, even solitary waves can dissipate. If the wave height is measured far offshore and used as an initial condition for non-dissipative numerical models, the comparisons will be less meaningful, as the solitary wave will slightly change as it propagates in the laboratory. By keeping the same relative offshore distance for defining the initial condition, meaningful comparisons are assured.

While only 10 wave gages were used in each experimental run, the generation was extremely repeatable. As experiments were repeated for each wave height, the wave gages were moved to different locations, and the same $\widetilde{H}/\widetilde{d}$ wave was generated again until a sufficient number of data points existed to resolve the entire wave profile. In Synolakis (1987) two different comparisons are presented: one is the amplitude variation at specific x-locations, and the second is the amplitude variation at specific t-times, the latter resembling the image that a photograph from the side with a large depth of field and angular range would show.

This set of laboratory data has been used extensively for code validation: refer to Synolakis (1987), Zelt (1991), Titov and Synolakis (1995, 1997, 1998), Titov and González (1997), Grilli *et al.* (1997), Li and Raichlen (2000, 2001, 2002). In particular, the data sets for the $\widetilde{H}/\widetilde{d} = 0.0185$ (Fig. A4) nonbreaking and $\widetilde{H}/\widetilde{d} = 0.3$ (Fig. A12) breaking solitary waves seem the most often used and most appropriate for code validation.

3.2 Solitary wave on a composite beach

Revere Beach is located approximately 6 miles northeast of Boston in the City of Revere, Massachusetts. To address beach erosion and severe flooding problems, a physical model of the beach was constructed at the Coastal Engineering Laboratory of the U.S. Army Corps of Engineers, Vicksburg, Mississippi facility, earlier known as Coastal Engineering Research Center.

The model consists of three piecewise-linear slopes of 1:53, 1:150, and 1:13 from seaward to shoreward. At the shoreline there is a vertical wall (Fig. A7). In the laboratory, to evaluate the overtopping of the seawall,

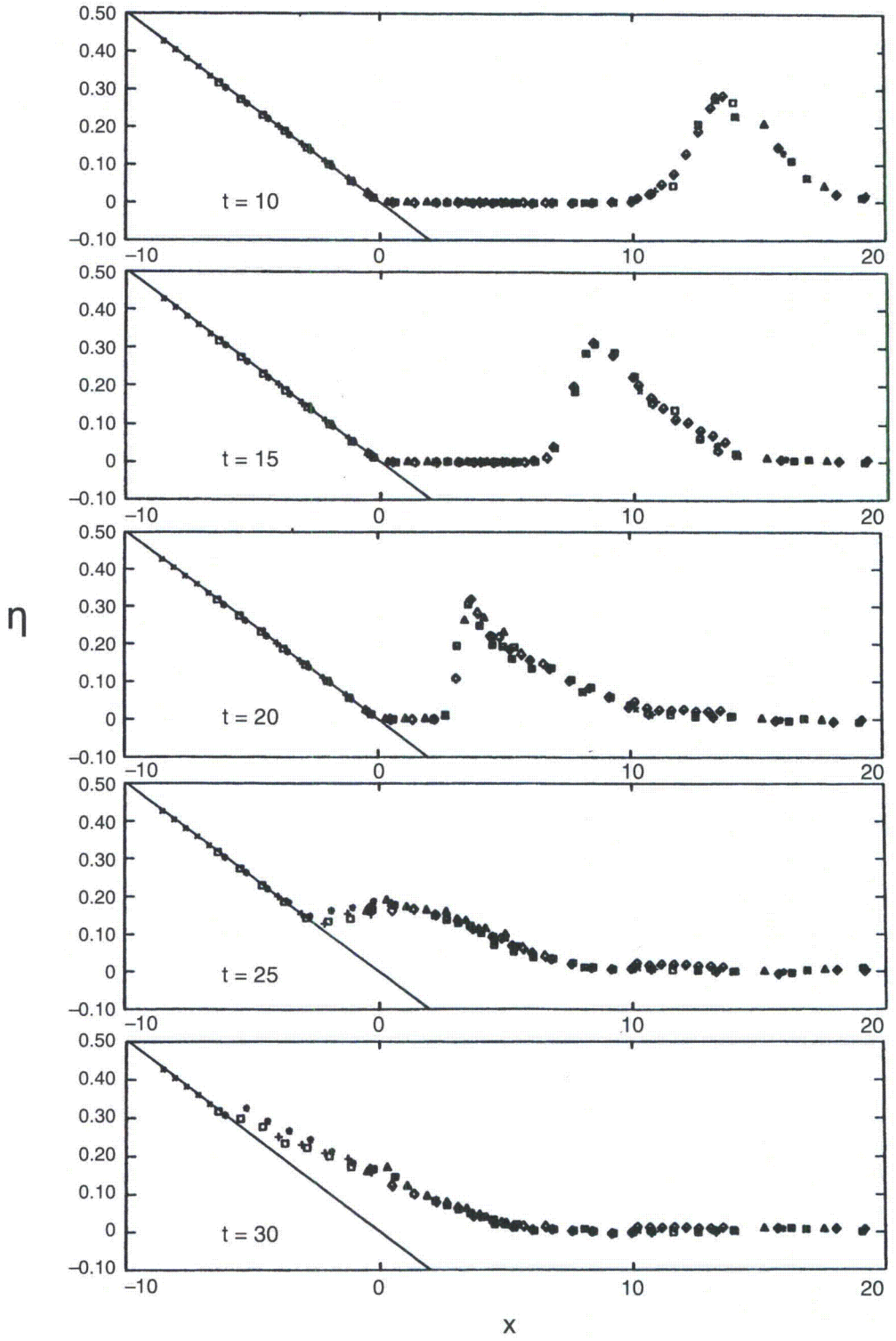

Figure A12: Time evolution of $H = 0.3$ initial wave over a sloping beach with $\cot\beta = 19.85$ from $t = 10$ to 30 with 5 increments. Constant-depth segment starts at $X_0 = 19.85$. Markers show a different realization of experimental results of Synolakis (1987).

the wavemaker was located at 23.22 m and tests were done at two depths, at 18.8 cm and at 21.8 cm. Water surface elevations were measured with 10 wave gages, and LDV measurements were performed at two different locations. The maximum runup on the vertical wall was measured visually.

In the experiments, solitary waves of different heights were generated, and the runup variation for solitary waves striking a vertical wall was determined experimentally. The results were used as benchmark data to validate 1+1 numerical codes in the 1995 Friday Harbor, Seattle, Washington workshop, as discussed in Yeh *et al.* (1996), and by Kânoğlu (1998) and Kânoğlu and Synolakis (1998) in validating their analytical formulation.

In terms of specific measurements, time histories of the water surface elevations exist for $\widetilde{H}/\widetilde{d} = 0.038$ at $\widetilde{x} = 12.22$ m, $\widetilde{H}/\widetilde{d} = 0.259$ at $\widetilde{x} = 13.96$ m, and for $\widetilde{H}/\widetilde{d} = 0.681$ at $\widetilde{x} = 14.37$ m. These are the locations where the solitary waves of the given $\widetilde{H}/\widetilde{d}$ are centered for the same reason explained for the solitary wave experiments over a 1:19.85 beach. In all cases, $\widetilde{d} = 21.8$ cm. $\widetilde{x} = 0$ is at one end of the wave flume, i.e., the initial shoreline where the vertical wall is at $\widetilde{x} = 23.22$ m. Other water-surface elevation time series were measured midway in each sloping segment, i.e., at $\widetilde{x} = 17.22$ m and $\widetilde{d} = 17.7$ cm, at $\widetilde{x} = 20.86$ m and $\widetilde{d} = 12.5$ cm and $\widetilde{x} = 22.80$ m and $\widetilde{d} = 8.1$ cm. Time histories of water surface elevations are given in Figs. A13, A14, and A15 for three different wave heights. The measured maximum runup values are given in Table A1 for the three cases presented here. Maximum runup measurements for the whole experiments are presented in Fig. A8.

Table A1: Maximum runup measurements on the laboratory model of Revere Beach.

$\widetilde{H}/\widetilde{d}$	$\widetilde{\mathcal{R}}$	$\widetilde{\mathcal{R}}/\widetilde{d}$
0.038	2.7 cm	0.13
0.259	45.7 cm	2.10
0.681	27.4 cm	1.26

When modeling these experiments, care is needed in calculations in the near-wall region. As the depth goes to zero, the wave breaks closer to the shoreline. The breaking wave collapses on the wall, and the air void explodes producing a splash which cannot be modeled effectively, except with very high resolution codes.

3.3 Solitary wave on a conical island

Motivated by the catastrophe in Babi Island, Indonesia (Yeh *et al.*, 1994), during the 1992 Flores Island tsunami, large-scale laboratory experiments were performed at Coastal Engineering Research Center, Vicksburg, Mississippi, in a 30 m-wide, 25 m-long, and 60 cm-deep wave basin (Fig. A16). Waves were realistically created in the tank by a horizontal wave generator with 60 different paddles each 46 cm-wide and moving independently. These experiments provided runup observations for validating numerical models and supplemented comparisons with analytical results (Kânoğlu and Synolakis, 1998).

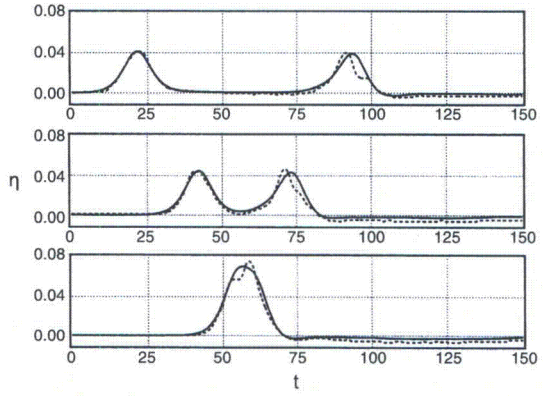

Figure A13: Comparison of the time histories of the free surface elevations midway in each sloping segment for the analytical solution of Kânoğlu and Synolakis (1998) and the laboratory data for $\tilde{H}/\tilde{d} = 0.038$ solitary wave.

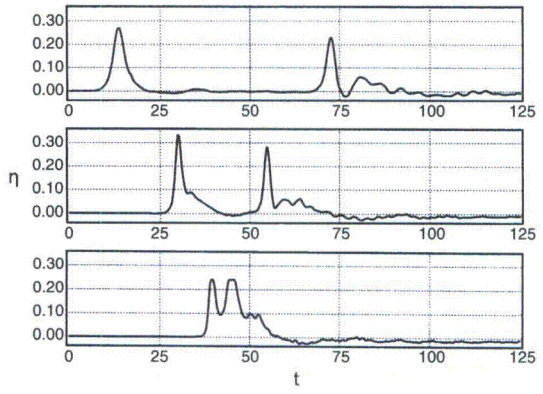

Figure A14: The laboratory data for the time histories of the free surface elevations midway in each sloping segment for $\tilde{H}/\tilde{d} = 0.259$ solitary wave.

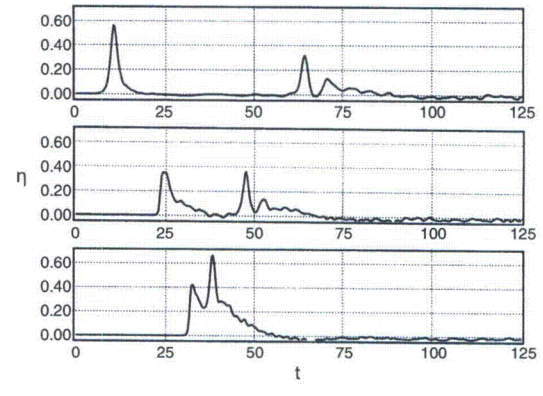

Figure A15: The laboratory data for the time histories of the free surface elevations midway in each sloping segment for $\tilde{H}/\tilde{d} = 0.681$ solitary wave.

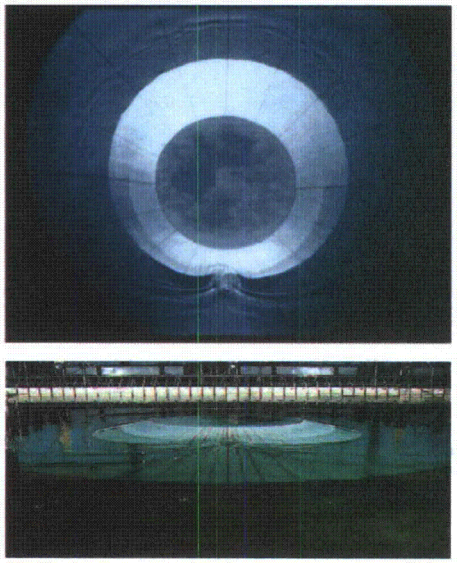

Figure A16: View of conical island (top) and basin (bottom).

The detailed experiments are described elsewhere in greater detail (Liu *et al.,* 1995; Briggs *et al.,* 1995; Kânoğlu, 1998; Kânoğlu and Synolakis, 1998). Briefly, a Directional Spectral Wave Generator (DSWG), located at $\widetilde{x} = 12.96$ m from the island, generated waves with an initial solitary wave-like profile. The 27.42 m-long DSWG consists of sixty 46 cm-wide and 76 cm-high individual paddles, each of which can be driven independently. This allowed performance of experiments with different wave crest lengths. However, the cases presented here were performed using all the paddles. Experimental results for different wave crest lengths are given in Briggs *et al.* (1995) and Kânoğlu (1998).

In the physical model, a 62.5 cm-high, 7.2 m toe-diameter, and 2.2 m crest-diameter circular island with a 1:4 slope was located in the basin (Fig. A17). Experiments were conducted at two different water depths, 32 cm and 42 cm, but presented here with dimensionless solitary wave heights $\widetilde{H}/\widetilde{d}$ equal to 0.045, 0.091, and 0.181 at 32 cm. Each experiment was repeated at least twice and maximum runup heights around the perimeter of the island were measured at 24 locations. Wavemaker signals were presented in Fig. A19 for these cases to allow direct implementation of these solitary waves as a wavemaker motion in the numerical models. Water-surface time histories were measured with 27 wave gages located around the perimeter of the island (Fig. A18). However, here, time histories of the surface elevation around the circular island are given at four locations, i.e., in the front of the island at the toe and gages closest to the shoreline with the numbers 9, 16, and 22 located at the 0°, 90°, and 180° radial lines, respectively (Figs. 20 and 21).

These experiments were used as benchmark tests for validating 2+1 nu-

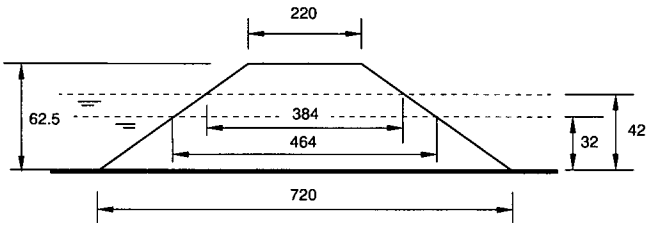

Figure A17: Definition sketch for conical island. All dimensions are in cm. Not to scale.

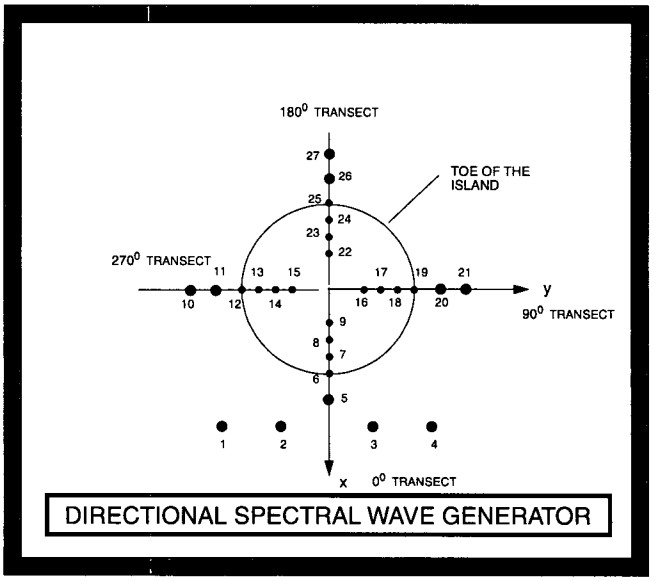

Figure A18: Schematic showing gage locations around the conical island. Not to scale.

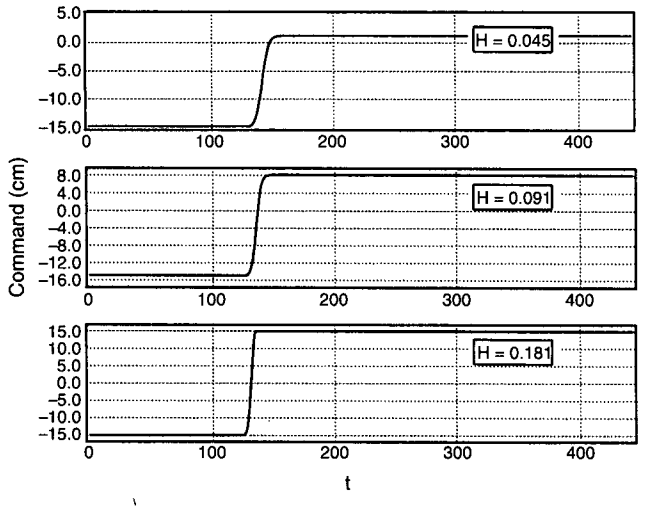

Figure A19: Wavemaker motions for the generation of $\tilde{H}/\tilde{d} = 0.045$, 0.091, and 0.181 solitary waves. Target wave heights are given in the insets.

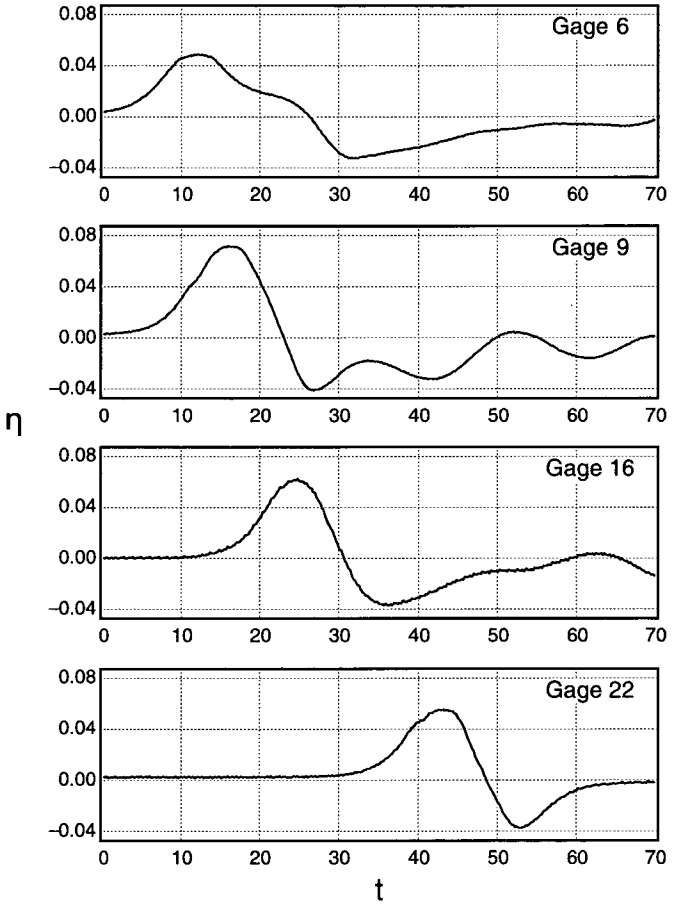

Figure A20: Laboratory data for the time histories of surface elevation for a $\tilde{H}/\tilde{d} = 0.045$ solitary wave at four gages and $\tilde{d} = 32$ cm. Gage 6 is located at the toe of the conical island on 0° radial line. Gages 9, 16, and 22 are the gages closest to the shoreline on the 0°, 90°, and 180° radial lines, respectively. Initial wave is defined half-wavelength (L/2, i.e., gages 1 to 4) away from the toe of the conical island.

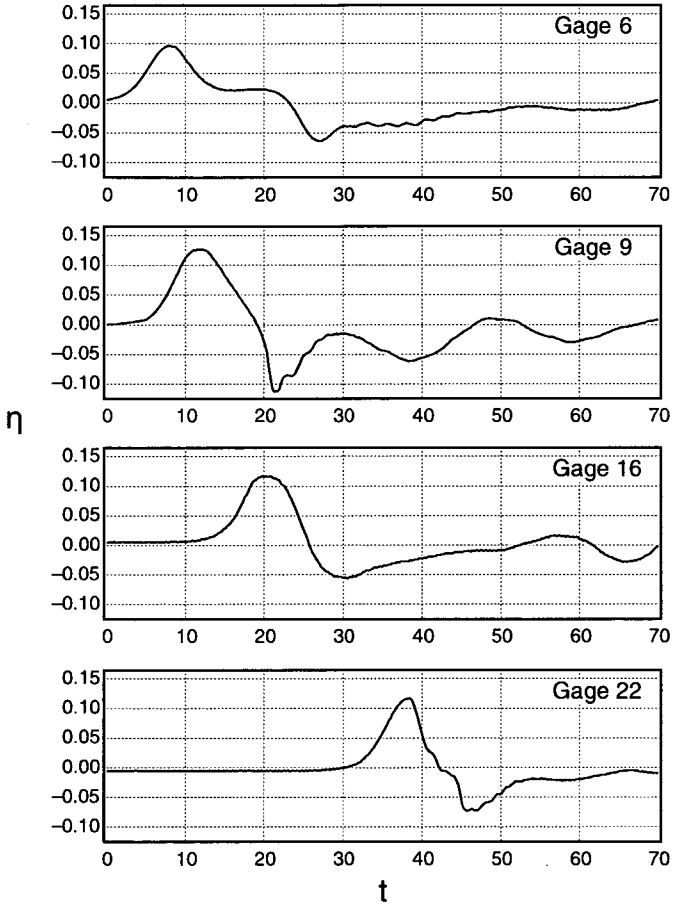

Figure A21: Laboratory data for the time histories of surface elevation for a $\tilde{H}/\tilde{d} = 0.091$ solitary wave at four gages and $\tilde{d} = 32$ cm. Gage 6 is located at the toe of the conical island on 0° radial line. Gages 9, 16, and 22 are the gages closest to the shoreline on the 0°, 90°, and 180° radial lines, respectively. Initial wave is defined half-wavelength (L/2, i.e., gages 1 to 4) away from the toe of the conical island.

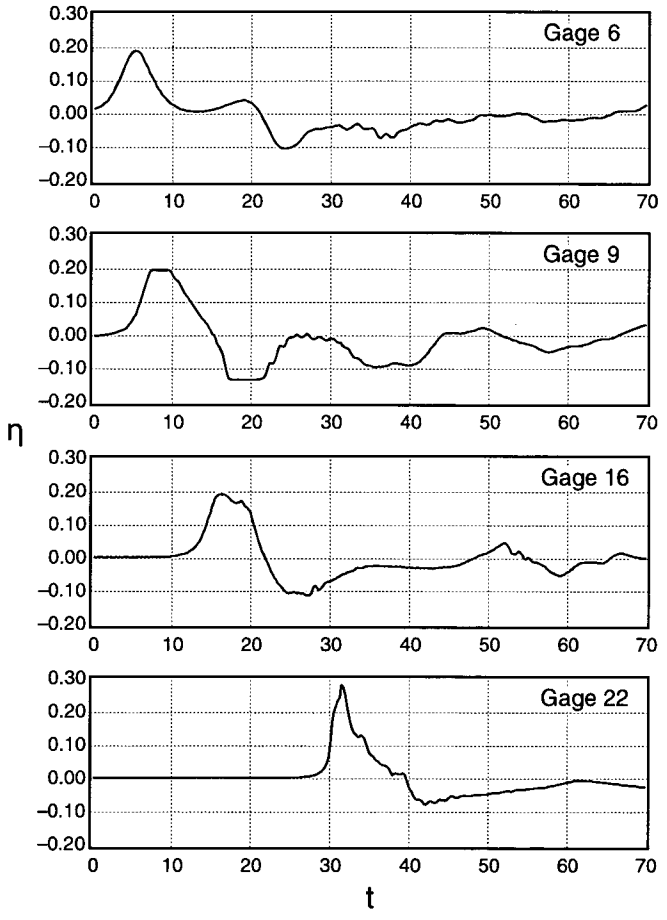

Figure A22: Laboratory data for the time histories of surface elevation for a $\tilde{H}/\tilde{d} = 0.181$ solitary wave at four gages and $\tilde{d} = 32$ cm. Gage 6 is located at the toe of the conical island on 0° radial line. Gages 9, 16, and 22 are the gages closest to the shoreline on the 0°, 90°, and 180° radial lines, respectively. Initial wave is defined half-wavelength (L/2, i.e., gages 1 to 4) away from the toe of the conical island.

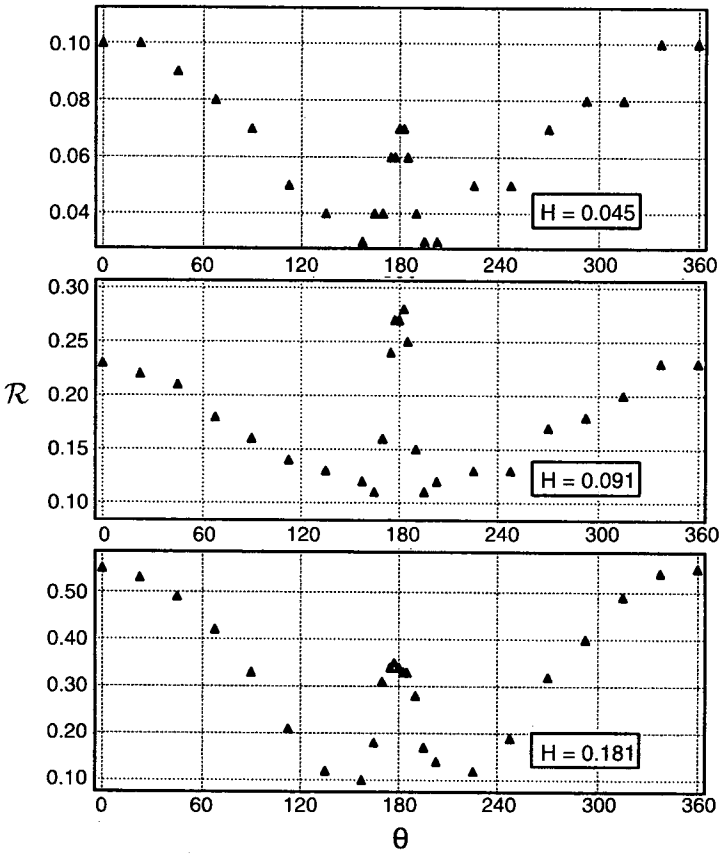

Figure A23: Maximum runup heights from the laboratory data for three solitary waves $\tilde{H}/\tilde{d} = 0.045$, 0.091, and 0.181 and $\tilde{d} = 32$ cm. Initial waves are defined half-wavelength (L/2, i.e., gages 1 to 4) away from the toe of the conical island.

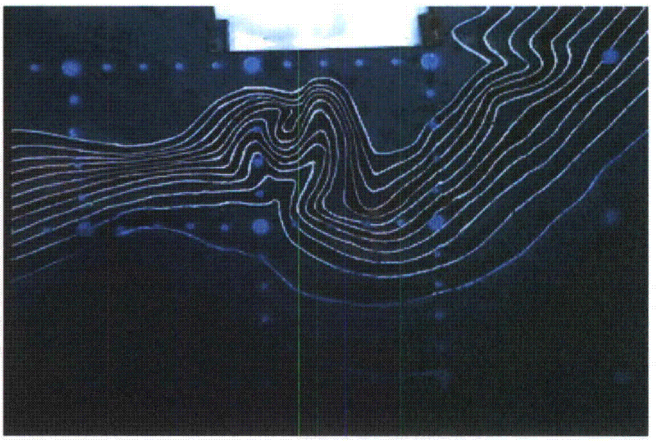

Figure A24: Bathymetric profile for experimental setup for Monai Valley experiment.

merical codes in the 1995 Friday Harbor, Seattle, Washington Long-Wave Runup Models Workshop (Yeh *et al.*, 1996).

3.4 Tsunami runup onto a complex three-dimensional beach; Monai Valley

The Hokkaido-Nansei-Oki tsunami of 1993 that struck Okushiri Island, Japan, with extreme runup height of 30 m and currents of the order of 10–18 m/sec was a disaster, but provided fortuitous high-quality data. The extreme tsunami runup mark was discovered at the tip of a very narrow gulley within a small cove at Monai. High resolution seafloor bathymetry existed before the event and when coupled with bathymetric surveys following the event allowed meaningful identification of the seafloor deformation.

A 1/400 laboratory model of Monai was constructed in a 205 m-long, 6 m-deep, and 3.5 m-wide tank at Central Research Institute for Electric Power Industry (CRIEPI) in Abiko, Japan and partly shown in Fig. A24. The laboratory setup closely resembles the actual bathymetry. The incident wave from offshore, at the water depth $\widetilde{d} = 13.5$ cm is known. There are reflective vertical sidewalls at $\widetilde{y} = 0$ and 3.5 m (Fig. A25). The entire computational area is 5.448 m $\times$ 3.402 m, and the recommended time step and grid sizes for numerical simulations are $\widetilde{\Delta x} = \widetilde{\Delta y} = 1.4$ cm and $\widetilde{\Delta t} = 0.05$ sec.

The input wave is a LDN with a leading-depression height of -2.5 mm with a crest of 1.6 cm behind it (Fig. A26). Waves were measured at thirteen locations and complete time histories are given at three locations, i.e., $(\widetilde{x}, \widetilde{y}) = (4.521,\ 1.196)$, $(4.521,\ 1.696)$, and $(4.521,\ 2.196)$ in meters (Fig. A27). These experiments were used in the 2004 Catalina Island, Los Angeles, California NSF Long-Wave Runup Models Workshop.

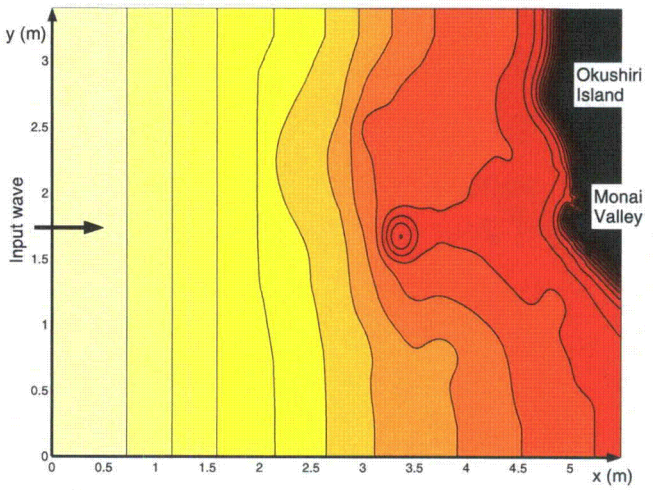

Figure A25: Computational area for Monai Valley experiment.

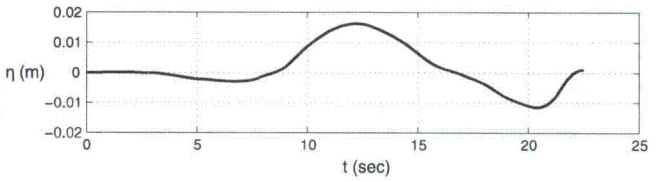

Figure A26: Initial wave profile for Monai Valley experiment.

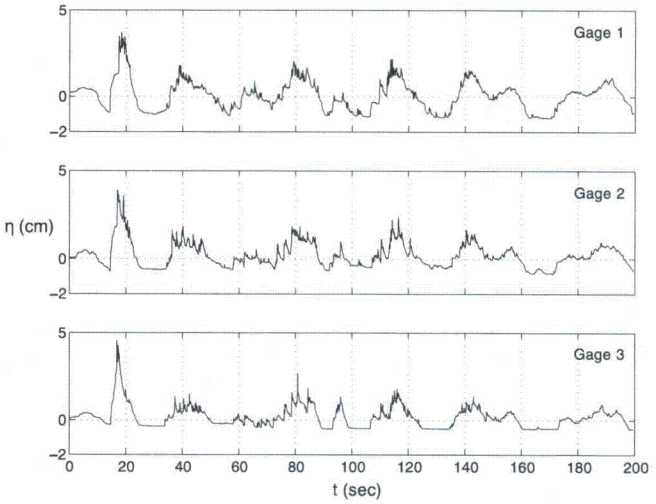

Figure A27: Time series of surface elevation at three different gages. Gages 1, 2, and 3 are located at (4.521, 1.196), (4.521, 1.696), and (4.521, 2.196) meters, respectively.

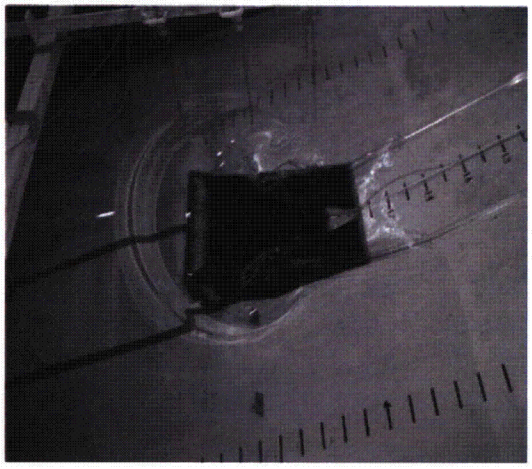

Figure A28: Picture of part of the experimental setup.

3.5 Tsunami generation and runup due to three-dimensional landslide

As discussed, landslide wave generation remains the frontier in terms of numerical modeling, particularly for aerial slides. These involve not only the rapid change of the seafloor, but also the impact of the slide on the shoreline. Details and more experimental results can be found in Liu *et al.* (2005).

Large-scale experiments have been conducted in a wave tank with a 104 m-long, 3.7 m-wide, and 4.6 m-deep wave channel with a plane slope (1:2) located at one end of the tank; part of the experimental setup is shown in Fig. A28. A solid wedge was used to model the landslide. The triangular face had a horizontal length of $b = 91$ cm, a vertical face with a height of $a = 45.5$ cm, and a width of $w = 61$ cm (Fig. A29). The horizontal surface of the wedge was initially positioned either a small distance above or below the still water level to reproduce a subaerial or submarine landslide. The block was released from rest, abruptly moving downslope under gravity, rolling on specially designed wheels (with low friction bearings) riding on aluminum strips with shallow grooves inset into the slope. The wedge was instrumented with an accelerometer to accurately define the acceleration-time history and a position indicator to independently determine the velocity and position time histories. Wedge positions are given in Fig. A30 for the two cases presented here.

A sufficient number of wave gages were used to determine the seaward propagating waves, the waves propagating to either side of the wedge, and for the submerged case, the water surface-time history over the wedge. In addition, the time history of the runup on the slope was accurately measured. Time histories of the surface elevations and runup measurements for two cases are presented in Figs. A31 and A32 for the submerged cases with

Table A2: Fault parameters of DCRC-17a.

Subfault	North	Central	South
Width (km)	25	25	25
Length (km)	90	30	24.5
Strike (deg)	188	175	163
Dip angle (deg)	35	60	60
Slip angle (deg)	80	105	105
Depth (km)	10	5	5
Dislocation (m)	5.71	2.50	12.00
Origin (°)	139.40°E 41.13°N	139.25°E 42.34°N	139.30°E 42.10°N
M_0 ($\times 10^{27}$ dyn cm)	3.85	0.56	2.21

$\Delta = -0.025$ m and $\Delta = -0.1$ m, respectively. While a total of more than 50 experiments with moving wedges, hemispheres, and rectangles were conducted, the wedge experiments were used as benchmark tests in the 2004 Catalina Island, Los Angeles, California workshop.

4 Field benchmarking

Verification of a model in a real-world setting is an important part of an operational model validation. No analytical or laboratory data comparisons (or any limited number of tests, for that matter) can assure robust model performance in the operational environment. Test comparisons with real-world data provide an additional important step in the validation of a model to perform well during operational implementation.

4.1 Okushiri Island

Okushiri Island, Japan, data was the benchmark problem for the 2nd International Long-Wave Runup Models Workshop, Friday Harbor, Washington (Takahashi, 1996). The magnitude $M_s = 7.8$ Hokkaido Nansei-Oki, Japan, earthquake occurred on 12 July 1993 with the hypocenter located at 37 km depth at 42.76°N and 139.32°E off the southwestern coast of Hokkaido, Japan. The Disaster Control Research Center (DCRC), Japan, digitized the bathymetric and topographic data from several sources (Fig. A33).

DCRC constructed initial wave profile DCRC-17a with 4.9 m and 2.2 m uplift and 1.1 m subsidence (depression) as shown in Fig. A34 and given in Table A2. There are several observations which need to be explained by numerical modeling:

- Arrival of the first wave to Aonae 5 min after the earthquake should be estimated with the numerical model. Also, a numerical model should reveal two waves at Aonae approximately 10 min apart; while the first wave came from the west, the second wave came from the east. In addition, two tide gage records at Iwanai and Esashi given in Fig. A35 need to be estimated.

- Maximum runup distribution around Okushiri Island should compare well with the field measurements (Fig. A36). High runup height at

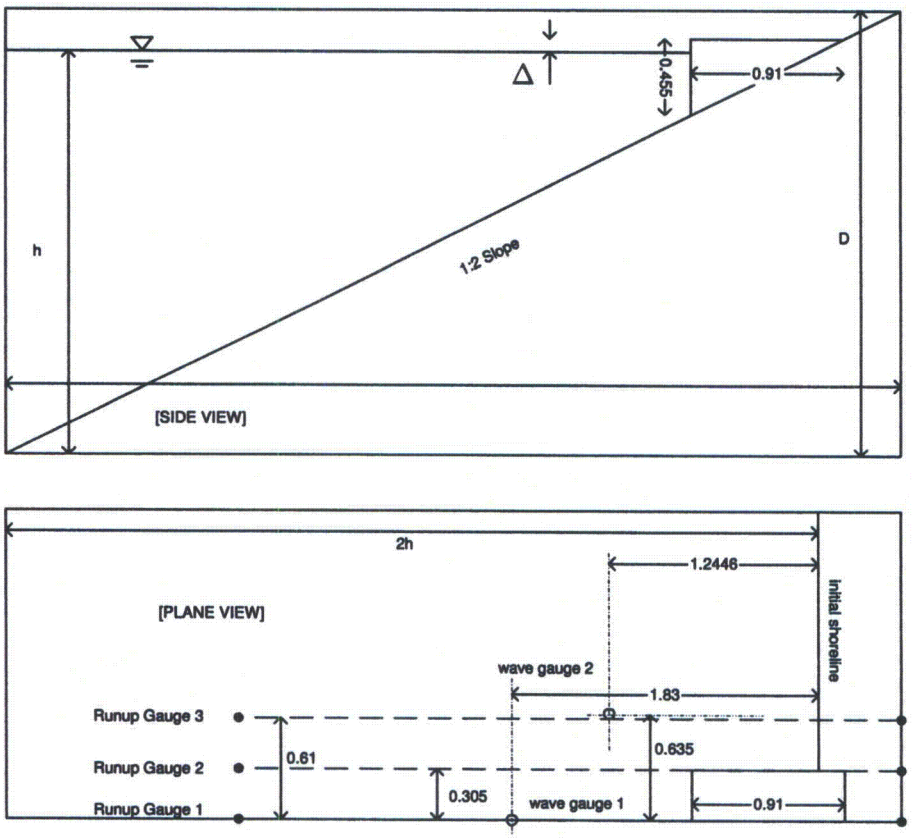

Figure A29: Schematic of the experimental setup.

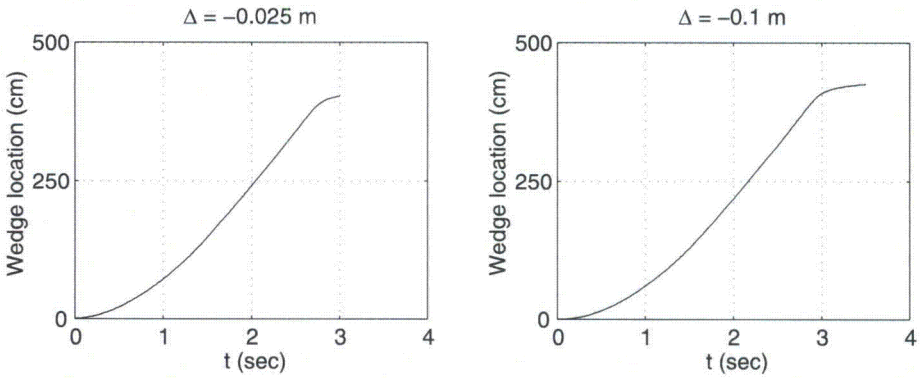

Figure A30: Time histories of the block motion for the submerged case with $\Delta = -0.025$ m and -0.1 m.

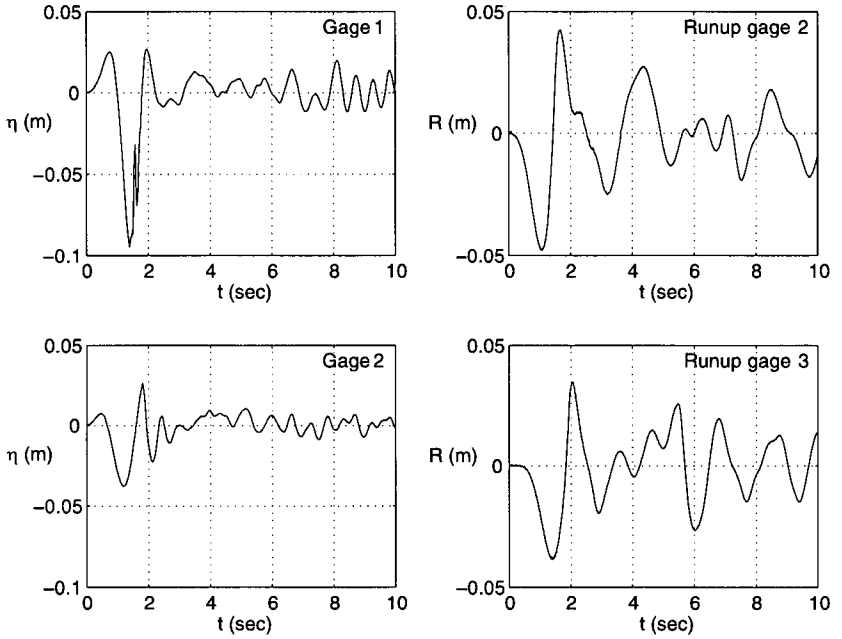

Figure A31: Time histories of the surface elevation and runup measurements for the submerged case with $\Delta = -0.025$ m.

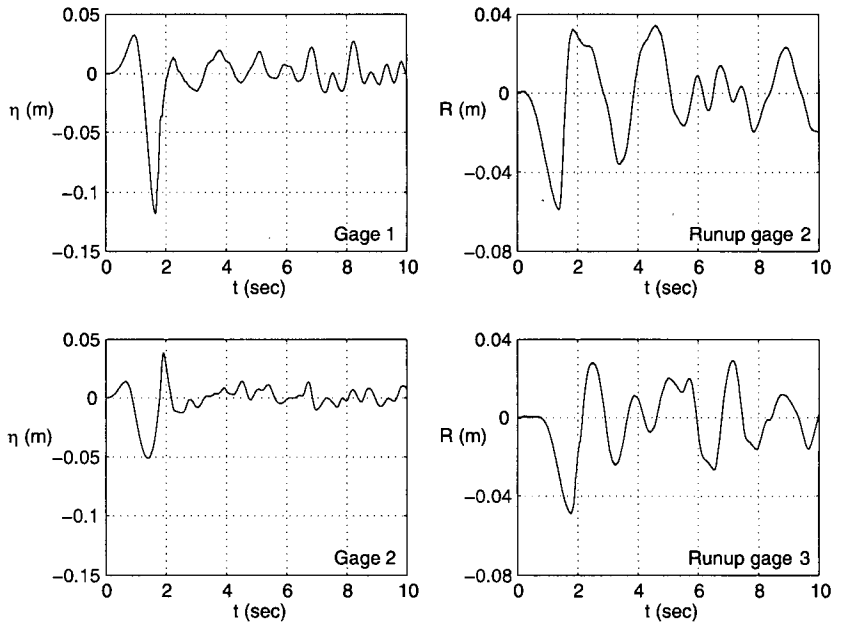

Figure A32: Time histories of the surface elevation and runup measurements for the submerged case with $\Delta = -0.1$ m.

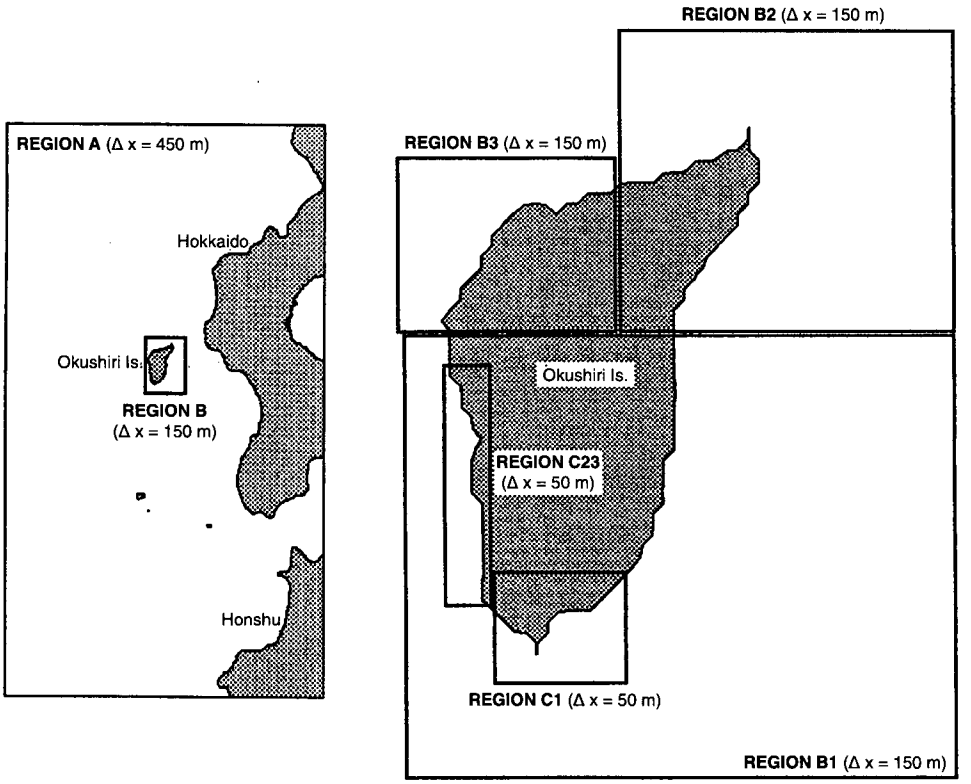

Figure A33: Bathymetric data provided by the Disaster Control Research Center, Japan.

Hamatsumae, located to the east of Aonae, needs to be explained since Hamatsumae is sheltered against the direct attack of the tsunami by the Aonae point. Also, topography does not suggest any tsunami amplification mechanism at this location.

- The highest runup of 31.7 m in a valley north of Monai needs to be approximated with the numerical model (Fig. A37).

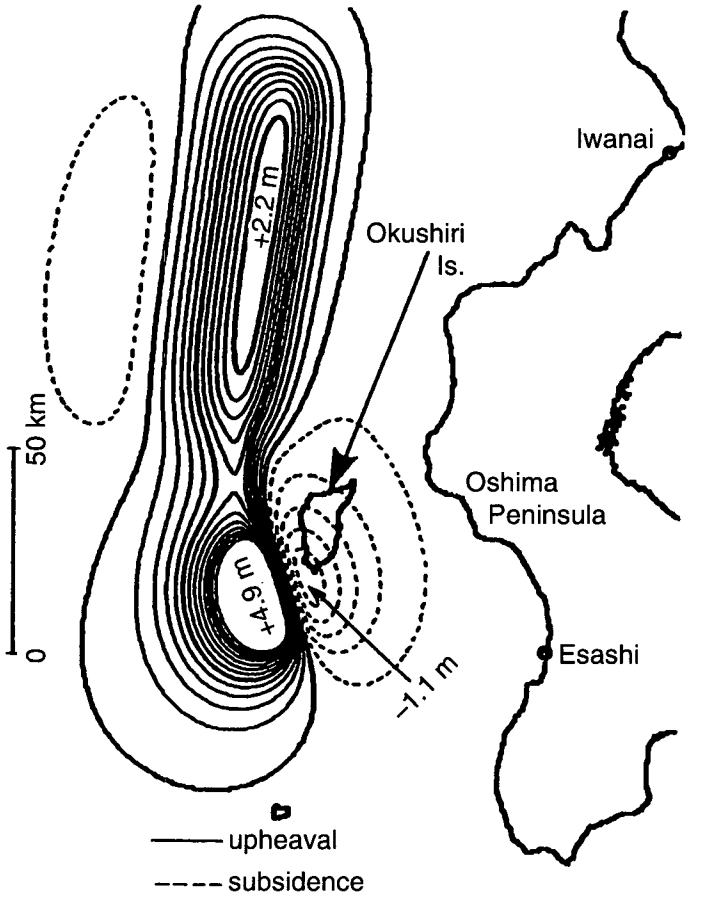

Figure A34: Fault plane DCRC-17a constructed by the Disaster Control Research Center, Japan.

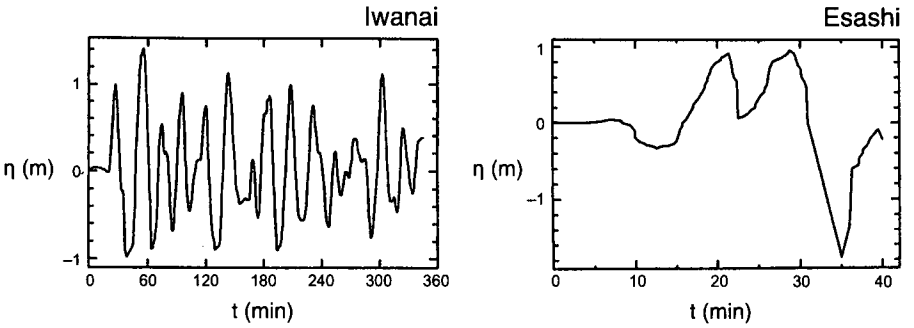

Figure A35: Tide gage data at Iwanai and Esashi.

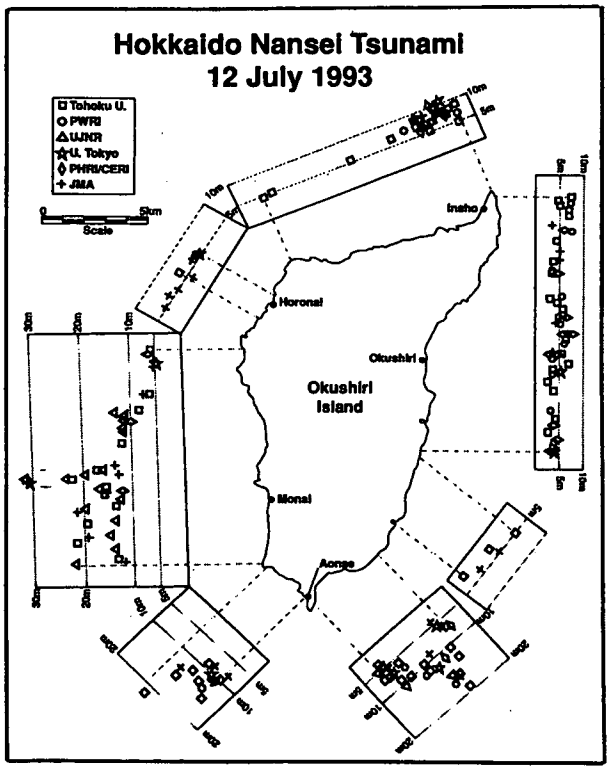

Figure A36: Maximum runup measurements around Okushiri Island.

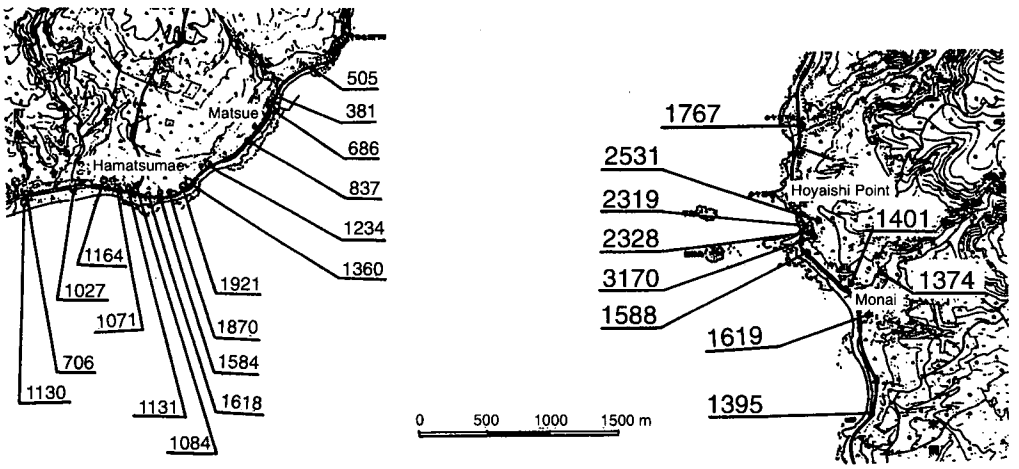

Figure A37: Detailed runup distribution at Hamatsume and north of Monai.

4.2 Rat Islands tsunami

Observation data from the expanding tsunami observation network will continue to provide more data for model verification. NOAA's National Geophysical Data Center (`http://www.ngdc.noaa.gov/seg/hazard/tsu.shtml`), NOAA's Tsunami Warning Centers (`http://wcatwc.arh.noaa.gov/`), and NOAA's Center for Tsunami Research websites provide updated information on the latest tsunami data. Here, we present the data used for the first real-time model forecast test as an example of data use for model verification.

The magnitude $M_w = 7.8$ earthquake was located on the shelf near Rat Islands, Alaska, on 17 November 2003 and generated a tsunami. This tsunami provided good data for testing operational models, since the tsunami was detected by three tsunameters located along the Aleutian Trench and was recorded at many coastal locations (Titov *et al.*, 2005). This was the first real-time tsunami detection by the newly developed tsunameter system (DART). In addition, for the first time, tsunami model predictions were obtained during the tsunami propagation, before the waves had reached many coastlines. Here, the combined use of tsunami propagation and inundation models is required for simulation of tsunami dynamics from generation to inundation. The test requires matching the propagation model data with the DART recording to constrain the tsunami source model (Fig. A38). If a finite-difference method on a structured grid is used, several nested numerical grids would allow "telescoping" from a coarse-resolution propagation model into a high-resolution inundation model with a model grid of at least 50 m resolution. If an unstructured grid method is used, a single grid may include enough resolution near the coast. The data-constrained propagation model should drive the high-resolution inundation model of Hilo Harbor. The inundation model is to reproduce the tide gage record at Hilo (Fig. A39). Since this benchmarking is required for the forecasting models, it is essential to model 4 hr of Hilo Harbor tsunami dynamics in 10 min of computational time.

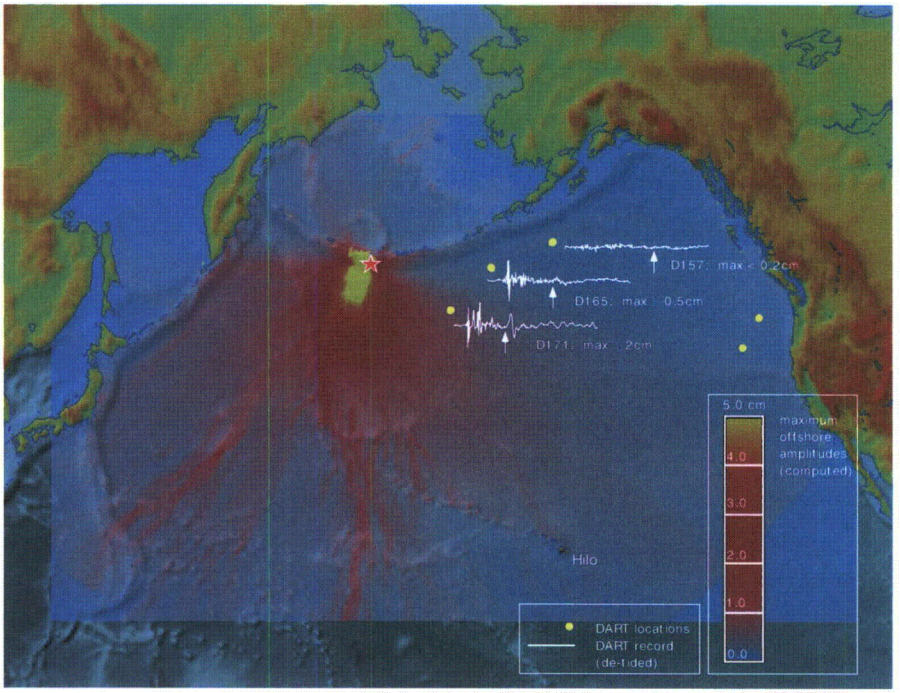

Figure A38: Propagation of the 17 November 2003 Rat Islands tsunami. Star indicates epicenter location of the earthquake. Yellow dots are locations of DART systems. White lines near the DART locations show recorded tsunami signal (detided) at corresponding tsunameter, arrows indicate tsunami arrival on the recordings. Filled colors show example of computed maximum tsunami amplitudes of a model propagation scenario.

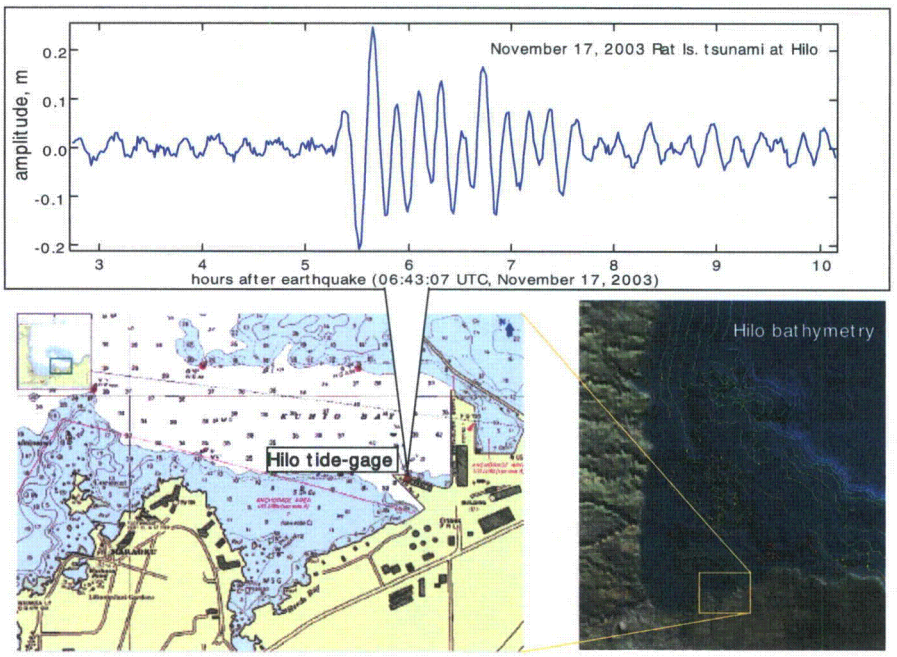

Figure A39: Location of Hilo tide gage and the recording of the 17 November 2003 Rat Islands tsunami.

Tsunami Benchmark Results for Fully Nonlinear Boussinesq Wave Model FUNWAVE-TVD, Version 1.0

Research Report No. CACR-11-02

Babak Tehranirad, Fengyan Shi and James T. Kirby
Center for Applied Coastal Research, University of Delaware, Newark, DE 19716
Jeffrey C. Harris and Stephan Grilli
Department of Ocean Engineering, University of Rhode Island, Narragansett, RI 02882

June 22, 2011

Abstract

This report describes tsunami benchmark testing of the Boussinesq model FUNWAVE-TVD, carried out in conjunction with the National Tsunami Hazard Mitigation Program, MMS Tsunami Inundation Model Validation Workshop, March 28 - April 1, 2011. The results presented here represent testing of Version 1.0 of the code, and will be updated online at http://chinacat.coastal.udel.edu/ kirby/funwave.html with each version change for the publicly distributed code.

This work was supported by the National Tsunami Hazard Mitigation Program.

Contents

1 Introduction 7

2 Model description 7
- 2.1 Governing equations . . . 8
- 2.2 Treatment of the surface gradient term . . . 9
- 2.3 Conservative form of fully nonlinear Boussinesq equations . . . 10
- 2.4 Numerical schemes . . . 11
- 2.5 Wave breaking and wetting-drying schemes for shallow water . . . 12
- 2.6 Boundary conditions and wavemaker . . . 12
- 2.7 Parallelization . . . 12

3 Basic hydrodynamic considerations 12
- 3.1 Mass conservation . . . 13
- 3.2 Convergence . . . 13

4 Analytical benchmarks 13
- 4.1 Solitary wave on a simple beach . . . 14
- 4.2 N-wave runup on a simple beach . . . 16
- 4.3 Solitary wave on composite beach . . . 19

5 Laboratory benchmarks 21
- 5.1 Solitary wave on a simple beach . . . 21
- 5.2 Solitary wave on a composite beach . . . 25
- 5.3 Solitary wave on a conical island . . . 27
- 5.4 Tsunami runup onto a complex three-dimensional beach; Monai Valley . . . 34

List of Figures

1 Definition sketch for simple beach bathymetry. 14
2 Numerical simulation data for maximum runup of nonbreaking waves climbing up different beach slopes. Solid line represents the runup law 16
3 The water level profiles during runup of the non-breaking wave in the case of $H/d = 0.019$ on a 1:19.85 beach. Solid blue line represents the analytical solution in according to Synolakis(1986), and dashed red lines represents the numerical simulation. 17
4 The water level dynamics at two locations $X/d = 0.25$ and $X/d = 9.95$. Solid blue line represents the analytical solution in according to Synolakis(1986), and dashed red lines represents the numerical simulation. 18
5 Numerical simulation data for maximum runup of N-waves climbing up different beach slopes. Solid line represents the runup law (28). 20
6 Definition sketch for Revere Beach (from Synolakis et al (2007, Figure A7). 20
7 Time evolution of nonbreaking $H/d = 0.0378$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution 22
8 Time evolution of breaking $H/d = 0.2578$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution 23
9 Time evolution of breaking $H/d = 0.6404$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution 24
10 Time evolution of nonbreaking $H/d = 0.0185$ initial wave. The solid line shows the numerical solution and dots represent the laboratory data. 25
11 Time evolution of breaking $H/d = 0.3$ initial wave. The solid line shows the numerical solution and dots represent the laboratory data. 26
12 Time evolution of nonbreaking $H/d = 0.0378$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data 27
13 Time evolution of breaking $H/d = 0.2578$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data . 28
14 Time evolution of breaking $H/d = 0.6404$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data . 29
15 View of conical island(top) and basin(bottom) 30
16 Definition sketch for conical island. All dimensions are in cm. 31
17 Schematic gauge locations around the conical island 31
18 Comparison of computed and measured time series of free surface for $H/d = 0.045$.Solid lines: measured, Dashed lines: Computed 32
19 Comparison of computed and measured time series of free surface for $H/d = 0.091$.Solid lines: measured, Dashed lines: Computed 33
20 Comparison of computed and measured time series of free surface for $H/d = 0.181$.Solid lines: measured, Dashed lines: Computed 33

21 Bathymetric profile for experimental setup for Monai Valley experiment 34
22 Initial wave profile for Monai Valley experiment 35
23 Computational area for Monai Valley experiment 35
24 Comparison of computed and measured time series of free surface. Dashed lines: Computed, Solid lines: Measured 36
25 Comparison between extracted movie frames from the overhead movie of the laboratory experiment (left) and numerical simulation (right). 37

List of Tables

1 Maximum runup for gauge 9 for different grid size 14
2 Runup data from numerical calculations compared with runup law values 15
3 Runup data from numerical calculations compared with runup law for N-wave. . . 19
4 Maximum runup of solitary wave on composite beach compared to runup law (29). 21
5 Percent error of predicted maximum runup calculated for each gauge in conical island test. 32

1 Introduction

This report describes the benchmark testing of the Boussinesq model FUNWAVE-TVD which has been carried out as part of the Benchmark Workshop exercise for the National Tsunami Hazard Mitigation Program. The benchmark tests described here are taken from Synolakis et al (2007). The report is organized as follows. Section 2 provides a description of the model equations and numerical scheme. Section 3 provides basic information on hydrodynamic considerations used to judge basic model validity. Sections 4 and 5 describe benchmark tests for analytical and laboratory cases, respectively.

Data for each benchmark are obtained from http://nctr.pmel.noaa.gov/benchmark/

FUNWAVE-TVD is distributed as open source code. General users may obtain the most recent tested version from the web site

http://chinacat.coastal.udel.edu/ kirby/programs/index.html

which provides this code along with other programs developed at the Center for Applied Coastal Research. The code is provided along with a unix/linux makefile, a users manual (Shi et al, 2011b), and input files for executing the tests described in the manual. The present report will also be updated with each major change in program version. Input files and scripts for executing the benchmark tests described here are provided at http://chinacat.coastal.udel.edu/ kirby/programs/index/funwave.html.

Version control for FUNWAVE-TVD is done using Mercurial (O'Sullivan, 2009). Users who would like to become part of the development team should contact Fengyan Shi (fyshi@udel.edu) or Jim Kirby (kirby@udel.edu).

2 Model description

In this section, we describe the set of Boussinesq equations which are used as the basis for FUNWAVE-TVD. FUNWAVE-TVD is formuiated in both Cartesian coordinates and in spherical (lat-long) coordinates for application to ocean basin scale problems. The benchmarks considered here are all treated in Cartesian coordinates, and we therefore omit description of the spherical coordinate model. This may be found in Shi et al (2011b). We retain dimensional forms below but will refer to the apparent $O(\mu^2)$ ordering of terms resulting from deviations from hydrostatic behavior in order to identify these effects as needed. Here. μ is a parameter characterizing the ratio of water depth to wave length, and is assumed to be small in classical Boussinesq theory. The model equations used here follow from the work of Chen (2006). In this and earlier works starting with Nwogu (1993), the horizontal velocity is written as

$$\mathbf{u} = \mathbf{u}_\alpha + \mathbf{u}_2(z) \tag{1}$$

Here, $\mathbf{u}_\alpha$ denotes the velocity at a reference elevation $z = z_\alpha$, and

$$\mathbf{u}_2(z) = (z_\alpha - z)\nabla A + \frac{1}{2}\left(z_\alpha^2 - z^2\right)\nabla B \tag{2}$$

represents the depth-dependent correction at $O(\mu^2)$, with A and B given by

$$\begin{aligned} A &= \nabla \cdot (h\mathbf{u}_\alpha) \\ B &= \nabla \cdot \mathbf{u}_\alpha \end{aligned} \tag{3}$$

The derivation follows Chen (2006) except for the additional effect of letting the reference elevation z_α vary in time according to

$$z_\alpha = \zeta h + \beta\eta \tag{4}$$

where h is local still water depth, η is local surface displacement and ζ and β are constants, as in Kennedy et al (2001). This addition does not alter the details of the derivation, which are omitted below.

2.1 Governing equations

The equations of Chen (2006) extended to incorporate a possible moving reference elevation follow. The depth-integrated volume conservation equation is given by

$$\eta_t + \nabla \cdot \mathbf{M} = 0 \tag{5}$$

where

$$\mathbf{M} = H\{\mathbf{u}_\alpha + \overline{\mathbf{u}}_2\} \tag{6}$$

is the horizontal volume flux. $H = h + \eta$ is the total local water depth and $\overline{\mathbf{u}}_2$ is the depth averaged $O(\mu^2)$ contribution to the horizontal velocity field, given by

$$\overline{\mathbf{u}}_2 = \frac{1}{H}\int_{-h}^{\eta} \mathbf{u}_2(z)dz = \left(\frac{z_\alpha^2}{2} - \frac{1}{6}(h^2 - h\eta + \eta^2)\right)\nabla B + \left(z_\alpha + \frac{1}{2}(h-\eta)\right)\nabla A \tag{7}$$

The depth-averaged horizontal momentum equation can be written as

$$\mathbf{u}_{\alpha,t} + (\mathbf{u}_\alpha \cdot \nabla)\mathbf{u}_\alpha + g\nabla\eta + \mathbf{V}_1 + \mathbf{V}_2 + \mathbf{V}_3 + \mathbf{R} = 0 \tag{8}$$

where g is the gravitational acceleration and $\mathbf{R}$ represents diffusive and dissipative terms including bottom friction and subgrid lateral turbulent mixing. $\mathbf{V}_1$ and $\mathbf{V}_2$ are terms representing the dispersive Boussinesq terms given by

$$\mathbf{V}_1 = \left\{\frac{z_\alpha^2}{2}\nabla B + z_\alpha \nabla A\right\}_t - \nabla\left[\frac{\eta^2}{2}B_t + \eta A_t)\right] \tag{9}$$

$$\mathbf{V}_2 = \nabla\left\{(z_\alpha - \eta)(\mathbf{u}_\alpha \cdot \nabla)A + \frac{1}{2}(z_\alpha^2 - \eta^2)(\mathbf{u}_\alpha \cdot \nabla)B + \frac{1}{2}[A + \eta B]^2\right\} \tag{10}$$

The form of (9) allows for the reference level z_α to be treated as a time-varying elevation, as suggested in Kennedy et al (2001). If this extension is neglected, the term reduced to the form

given originally by Wei et al (1995). The expression (10) for $\mathbf{V}_2$ was also given by Wei et al (1995), and is not altered by the choice of a fixed or moving reference elevation.

The term $\mathbf{V}_3$ in (8) represents the $O(\mu^2)$ contribution to the expression for $\boldsymbol{\omega} \times \mathbf{u} = \omega \mathbf{i}^z \times \mathbf{u}$ (with $\mathbf{i}^z$ the unit vector in the z direction) and may be written as

$$\mathbf{V}_3 = \omega_0 \mathbf{i}^z \times \overline{\mathbf{u}}_2 + \omega_2 \mathbf{i}^z \times \mathbf{u}_\alpha \tag{11}$$

where

$$\omega_0 = (\nabla \times \mathbf{u}_\alpha) \cdot \mathbf{i}^z = v_{\alpha,x} - u_{\alpha,y} \tag{12}$$

$$\omega_2 = (\nabla \times \overline{\mathbf{u}}_2) \cdot \mathbf{i}^z = z_{\alpha,x}(A_y + z_\alpha B_y) - z_{\alpha,y}(A_x + z_\alpha B_x) \tag{13}$$

Following Nwogu (1993), z_α is usually chosen in order to optimize the apparent dispersion relation of the linearized model relative to the full linear dispersion in some sense. In particular, the choice $\alpha = (z_\alpha/h)^2/2 + z_\alpha/h = -2/5$ recovers a Padé approximant form of the dispersion relation, while the choice $\alpha = -0.39$, corresponding to the choice $z_\alpha = -0.53h$, minimizes the maximum error in wave pahse speed occuring over the range $0 \leq kh \leq \pi$. Kennedy et al (2001) showed that allowing z_α to move up and down with the passage of the wave field allowed a greater degree of flexibility in optimizing nonlinear behavior of the resulting model equations. In the examples chosen here, where a great deal of our focus is on the behavior of the model from the break point landward, we adopt Kennedy et al's "datum invariant" form

$$z_\alpha = -h + \beta H = (\beta - 1)h + \beta\eta = \zeta h + (1 + \zeta)\eta \tag{14}$$

with $\zeta = -0.53$ as in Nwogu (1993) and $\beta = 1 + \zeta = 0.47$. This corresponds in essence to a σ coordinate approach which places the reference elevation at a level 53% of the total local depth below the local water surface. This also serves to keep the model reference elevation within the actual water column over the entire wetted extent of the model domain.

2.2 Treatment of the surface gradient term

The hybrid numerical scheme requires a conservative form of continuity equation and momentum equations, thus requiring a modification of the leading order pressure term in the momentum equation. A numerical imbalance problem occurs when the surface gradient term is conventionally split into an artificial flux gradient and a source term that includes the effect of the bed slope for a non-uniform bed. To eliminate errors introduced by the traditional depth gradient method (DGM), a so-called surface gradient method (SGM) proposed by Zhou et al. (2001) was adopted in the TVD based-Boussinesq models in the recent literature. Zhou et al. discussed an example of SGM in 1-D and verified that the slope-source term may be canceled out by part of the numerical flux term associated with water depth, if the bottom elevation at the cell center is constructed using the average of bottom elevations at two cell interfaces. Zhou et al. also showed a 2D application but without explicitly describing 2D numerical schemes. Although this scheme can be extended into

2D following the same procedure as in 1D, it was found that the 2D extension may not be trivial in terms of the bottom construction for a 2D arbitrary bathymetry. Kim et al. (2008) pointed out that the water depth in the slope-source term should be written in a discretized form rather than the value obtained using the bottom construction, implying that their revised SGM is valid for general 2D applications.

For the higher-order schemes such as the fourth-order TVD-MUSCL scheme (Yamamoto and Daiguji, 1993, Yamamoto et al., 1998) used in the recent Boussinesq applications, the original SGM and the revised SGM may not be effective in removing the artificial source. This problem was noticed by some recent authors such as Roeber et al. (2010) who kept a first-order scheme (second-order for normal conditions) for the numerical flux term and the slope-source term in order to ensure well-balanced solution without adding noise for a rapidly varying bathymetry.

In fact, the imbalance problem can be solved by a reformulation of this term in terms of deviations away from an unforced but separately specified equilibrium state (see general derivations in Rogers et al., 2003 and recent application in Liang and Marche, 2009). Using this technique, the surface gradient term may be split into

$$gH\nabla\eta = \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] - g\eta\nabla h \tag{15}$$

which is well-balanced for any numerical order under an unforced stationary condition (still water condition).

2.3 Conservative form of fully nonlinear Boussinesq equations

For Chen's (2006) equations or the minor extension considered here, $H\mathbf{u}_\alpha$ can be used as a conserved variable in the construction of a conservative form of Boussinesq equations, but resulting in a source term in the mass conservation equation such as in Shiach and Mingham (2009) and Roeber et al. (2010). An alternative approach is to use $\mathbf{M}$ as a conserved variable in terms of the physical meaning of mass conservation. In this study, we used $\mathbf{M}$, instead of $H\mathbf{u}_\alpha$, in the following derivations of the conservative form of the fully nonlinear Boussinesq equations.

Using M from (6) together with the vector identity

$$\nabla\cdot(\mathbf{uv}) = \nabla\mathbf{u}\cdot\mathbf{v} + (\nabla\cdot\mathbf{v})\mathbf{u} \tag{16}$$

allows (8) to be rearranged as

$$\begin{aligned}&\mathbf{M}_t + \nabla\cdot\left(\frac{\mathbf{MM}}{H}\right) + gH\nabla\eta \\ &= H\left\{\overline{\mathbf{u}}_{2,t} + \mathbf{u}_\alpha\cdot\nabla\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2\cdot\nabla\mathbf{u}_\alpha - \mathbf{V}_1 - \mathbf{V}_2 - \mathbf{V}_3 - R\right\}\end{aligned} \tag{17}$$

Following Wei et al. (1995), we separate the time derivative dispersion terms in $\mathbf{V}_1$ according to

$$\mathbf{V}_1 = \mathbf{V}'_{1,t} + \mathbf{V}''_1 \tag{18}$$

where

$$\mathbf{V}_1' = \frac{z_\alpha^2}{2}\nabla B + z_\alpha \nabla A - \nabla\left[\frac{\eta^2}{2}B + \eta A\right] \tag{19}$$

and

$$\mathbf{V}_1'' = \nabla\left[\eta_t(A + \eta B)\right] \tag{20}$$

Using (15), (19) and (20), the momentum equation can be rewritten as

$$\begin{aligned}&\mathbf{M}_t + \nabla\cdot\left[\frac{\mathbf{MM}}{H}\right] + \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] = \\ &= H\left\{\overline{\mathbf{u}}_{2,t} + \mathbf{u}_\alpha\cdot\nabla\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2\cdot\nabla\mathbf{u}_\alpha - \mathbf{V}_{1,t}' - \mathbf{V}_1'' - \mathbf{V}_2 - \mathbf{V}_3 - \mathbf{R}\right\} + g\eta\nabla h\end{aligned} \tag{21}$$

A difficulty usually arises in applying the adaptive time-stepping scheme to the time derivative dispersive terms $\bar{u}_{2,t}$ and $\mathbf{V}_{1,t}'$ which are usually calculated using values stored in several time levels in the previous Boussinesq codes such as in Wei et al. (1995) and Shi et al. (2001). The equation can be re-arranged by merging the time derivatives on the right hand side into the time derivative term on the left hand side, giving

$$\begin{aligned}\mathbf{V}_t \quad &+\nabla\cdot\left[\frac{\mathbf{MM}}{H}\right] + \nabla\left[\frac{1}{2}g(\eta^2 + 2h\eta)\right] = \eta_t(\mathbf{V}_1' - \overline{\mathbf{u}}_2) \\ &+H\left(\mathbf{u}_\alpha\cdot\overline{\mathbf{u}}_2 + \overline{\mathbf{u}}_2\cdot\nabla\mathbf{u}_\alpha - \mathbf{V}_1'' - \mathbf{V}_2 - \mathbf{V}_3 - \mathbf{R}\right) + g\eta\nabla h\end{aligned} \tag{22}$$

where

$$\mathbf{V} = H(\mathbf{u}_\alpha + \mathbf{V}_1') \tag{23}$$

In (23), η_t can be calculated explicitly using (5) as in Roeber et al. (2010). Equations (5) and (22) are the governing equations solved in this study. As $\mathbf{V}$ is obtained, the velocity $\mathbf{u}_\alpha$ can be solved by a system of tridiagonal matrix equation formed by (23) in which all cross-derivatives are moved to the right-hand side of the equation.

2.4 Numerical schemes

The governing equations in FUNWAVE-TVD are discretized on a regular grid using a hybrid finite-volume / finite-difference approach. The numerical scheme is described in detail in Shi et al (2011a, 2011b) and is omitted here for conciseness. Examples described below are based on discretizations on regular Cartesian grids with uniform grid spacings in x and y. Terms in conservative form in equations (5) and (22) are written in finite volume form using a MUSCL-TVD scheme, while additional terms are written in cell-centered finite differences. An adaptive third-order Strong Stability Preserving (SSP) Runge-Kutta scheme is used for time stepping (Gottlieb et al, 2001).

2.5 Wave breaking and wetting-drying schemes for shallow water

The wave breaking scheme follows the approach of Tonelli and Petti (2009) who successfully used the ability of NSWE with a TVD scheme to model moving hydraulic jumps. The fully nonlinear Boussinesq equations are switched to NSWE at cells where the Froude number exceeds an certain threshold. Following Tonelli and Petti, the ratio of wave height to total water depth is chosen to the criterion to switch from Boussinesq to NSWE. The threshold value is set to be 0.8 as suggested by Tonelli and Petti.

The wetting-drying scheme for modeling of a moving boundary is straightforward. The normal flux $\mathbf{n} \cdot \mathbf{M}$ at the cell interface of a dry cell is set to zero. A mirror boundary condition is applied to the fourth-order MUSCL-TVD scheme and discretization of dispersive terms in ψ_x, ψ_y at dry cells.

2.6 Boundary conditions and wavemaker

We implemented various boundary conditions including wall boundary condition, absorbing boundary condition following Kirby et al. (1998) and periodic boundary condition following Chen et al. (2003).

Wavemakers implemented in the study include Wei et al's (1999) internal wavemakers for regular waves and irregular waves. For the irregular wavemaker, an extension was made to incorporate the alongshore periodicity into wave generation in order to eliminate a boundary effect on wave simulation. The technique exactly follows the strategy in Chen et al. (2003) who adjusted the distribution of wave directions in each frequency bin to obtain alongshore periodicity. This approach is effective in modeling of breaking wave-induced nearshore circulation such as alongshore currents and rip currents.

2.7 Parallelization

In parallelizing the computational model, we use the domain decomposition technique to subdivide the problem into multiple regions and assign each subdomain to a separate processor core. Each subdomain region contains an overlapping area of ghost cells with three rows deep, as dictated by the fourth order MUSCL-TVD scheme. The Message Passing Interface (MPI) with non-blocking communication is used to exchange the data in the overlapping region between neighboring processors. Velocity components are obtained from Equation (23) by solving tridiagonal matrices using the parallel pipelining tridiagonal solver described in Naik et al. (1993).

3 Basic hydrodynamic considerations

There are two basic states which are required in ensuring that any numerical model works for predicting evolution and inundations. The first step is to ensuring that the model conserves mass; the second basic step is checking convergence of this numerical code to a asymptotic limit.

	Maximum Runup		
Grid Size	$H/d = 0.045$	$H/d = 0.091$	$H/d = 0.181$
0.1	0.02302	0.04061	0.06311
0.05	0.02303	0.04063	0.06315
0.01	0.02303	0.04064	0.06316
0.005	0.02303	0.04064	0.06316

Table 1: Maximum runup for gauge 9 for different grid size.

3.1 Mass conservation

Conservation of mass can be checked by calculating water volume at the beginning and at the end of the computation. This should be dine by integrating disturbed water depth $\eta(x, y, t)$ over the entire flow domain, i.e., if the flow domain extends from the maximum penetration during inundation $x = X_{max}$ to the outer location of the source region X_S, and $y = Y_{max}$ to Y_s, then the total displaced volume $V(t)$ is,

$$V(t) = \int_{X_{max}}^{X_S} \int_{Y_{max}}^{Y_S} \eta(x, y, t) dx dy \tag{24}$$

The integral of $\eta(x, y, t)$ should be used instead of the integral of the entire flow depth $h(x, y, t) = \eta(x, y, t) + d(x, y, t)$ where $d(x, y, t)$ is the undisturbed water depth, because the latter is likely to conceal errors in the calculation. Typically,$\eta << d$ at offshore integrating h will simply produce the entire volume of the flow domain and will mask errors. Note that testing of the conservation of mass as above involves placing a closed domain within reflective boundaries (Synolakis et al., 2007).

Calculations of conservation of mass has been done for all of the benchmark problems reviewed in this report such that the total initial displaced volume $V(t = 0)$ was within less than 1% of the total displaced volume at the end of the computation $V(t = T)$ where T represents the computation end time. It is assumed that the end of the computation is when the initial wave is entirely reflected and reached offshore. However, with few changes in Δx and Δy the conservation of mass can be improved.

3.2 Convergence

Convergence is the another basic hydrodynamic consideration that is checked for all of the benchmarks in this research. Actually this process is made by checking convergence of the numerical code to a certain asymptotic limit, presumably the actual solution of the equations solved. The grid steps Δx and Δy has been halved, and the time step Δt automatically reduced appropriately

to conform to the Courant-Friedrics-Levy (CFL) criterion. As recommended in literature, convergence of the code has been checked through the extreme runup and rundown. Table 1 displays convergence of the code tested during the conical island test problem which is discussed in Section 5.3 below.

4 Analytical benchmarks

In this section, using analytical description of shallow water-wave (SW) equations for evolution of the water surface, numerical results for different test cases is compared analytical solution. Also, in this section we present several analytic solutions to the 1+1 version of shallow water-wave (SW) equations. Although 1+1 models are not completely reliable for estimation of tsunmi behaviour such as inundation mapping, they are first step of testing and validating models (Synolakis et al., 2007).

In this section following benchmark problems has been studied:

1. Solitary wave on a simple beach
2. N-wave on a simple beach
3. Solitary wave on a composite beach

4.1 Solitary wave on a simple beach

The canonical problem of the shallow water-wave equations is covered here which contains the calculation of a long wave climbing up a sloping beach of angle β attached to a constant-depth region (Figure 1). The origin of the coordinate system is at the initial position of the shoreline and x increases seaward.

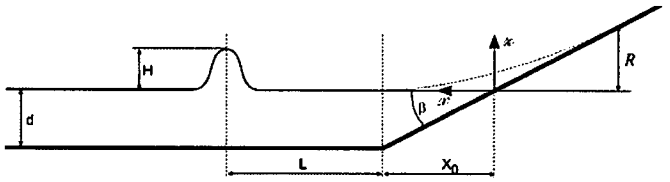

Figure 1: Definition sketch for simple beach bathymetry(from Synolakis et al (2007, Figure A1)).

It is possible to derive exact results for the evolution and runup of solitary waves based on linear theory (Synolakis, 1986, 1987). Solitary waves have long been used as a model for the leading wave of tsunamis. Russell (1845) defined solitary waves as the great waves of translation, and consists of a single elevation wave. While capturing some of the basic physics of tsunamis, solitary waves do not model the physical manifestation of tsunamis in nature, which are invariably N-wave like with

				R/d		
d(m)	$\Delta x(m)$	H/d	$Cot(\beta)$	Runup Law	Numerical Calculations	Error(%)
0.5	0.1	0.03	10.0	0.112	0.110	1.6
0.5	0.1	0.05	10.0	0.212	0.204	3.6
0.5	0.1	0.1	3.333	0.291	0.282	3.0
0.5	0.1	0.48	1.0	1.131	1.109	2.0
0.5	0.1	0.01	20.0	0.040	0.042	4.9
5.0	1.0	0.03	10.0	0.112	0.112	0.4
5.0	1.0	0.05	10.0	0.212	0.212	0.2
5.0	1.0	0.10	3.372	0.308	0.315	2.4
5.0	1.0	0.10	3.372	0.731	0.734	0.3
5.0	1.0	0.294	2.747	1.016	1.058	4.2
5.0	1.0	0.005	20.0	0.017	0.017	1.1
5.0	1.0	0.01	20.0	0.040	0.039	3.0
100	5.0	0.05	10.0	0.212	0.202	4.5
100	5.0	0.03	2.747	0.257	0.266	3.4
100	5.0	0.03	2.747	0.600	0.598	0.4
100	5.0	0.03	20.0	0.040	0.040	0.2

Table 2: Runup data from numerical calculations compared with runup law values.

a leading-depression wave followed by an elevation wave (Synolakis et al., 2007). The following runup law for the maximum runup R is provided based on slope of the beach and wave height of the solitary wave

$$R = 2.831\sqrt{\cot\beta}H^{\frac{5}{4}} \tag{25}$$

Benchmark problems that are studied here have different depths from 50cm to 1000m. Also, for each depth, different slopes and wave heights has been studied. Table 2 provides a list of selected examples that has been modeled including their maximum runup and the grid size for each case. Figure 2 defines a comparison between numerical simulation and runup law.

In addition, the analytical solution for different times is available for a specific case in which $H/d = 0.0019$ and $\beta = \text{arccot}(19.85)$. In order to have the same time with the data it was recommended that $L = \text{arccosh}(\sqrt{20})/\gamma$ in which $\gamma = \sqrt{3H/4d}$; therefore, the distance of the wave from initial shoreline(X_1) can be written as $X_1 = X_0 + L$ (with respect to Figure 1). Figure 3 demonstrates profiles and time series of the water in eight different times. Extreme positions of the shoreline are shown in figure 3 (the maximum runup and rundown occur $t \simeq 55(d/g)^{1/2}$ and $t \simeq 70(d/g)^{1/2}$). Figure 4 shows water level fluctuations at two gauge locations $X/d = 0.25$, $X/d = 9.95$. As it is clear in the figure the point $X/d = 0.25$ which is closer to initial shoreline, becomes temporarily dry during the process but the point $X/d = 9.95$ remains wet throughout the

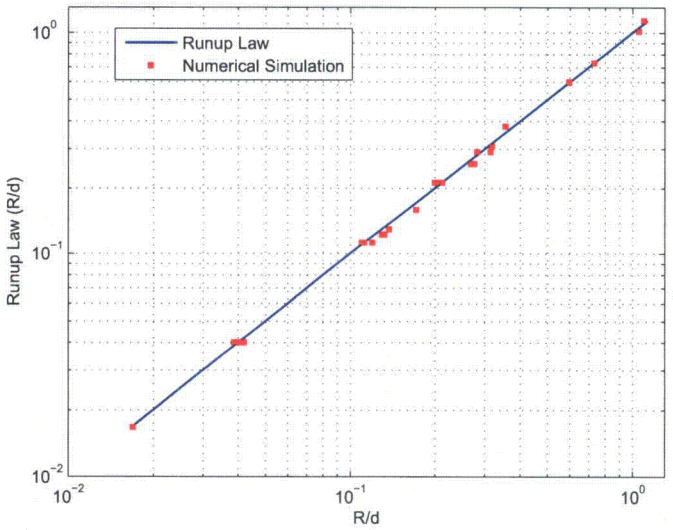

Figure 2: Numerical simulation data for maximum runup of nonbreaking waves climbing up different beach slopes. Solid line represents the runup law (25).

entire length of the numerical simulation.

4.2 N-wave runup on a simple beach

Most tsunami eyewitness accounts suggest that tsunamis are N-wave like, i.e., they are dipolar, which means they appear as a combination of a depression and an elevation wave, and frequently as a series of N-waves, sometimes known as double N-waves (Synolakis et al., 2007).

Tadepalli and Synolakis (1994) described an N-wave with leading-elevation and depression waves of the same height and at a constant separation distance and refer to this wave as an isosceles N-wave with a surface profile given by

$$\eta(x,0) = \frac{3\sqrt{3}H}{2}\mathrm{sech}^2[\gamma(x - X_N)]\tanh[\gamma(x - X_N)] \tag{26}$$

where

$$\gamma = \frac{3}{2}\sqrt{\sqrt{\frac{3}{4}}H} \tag{27}$$

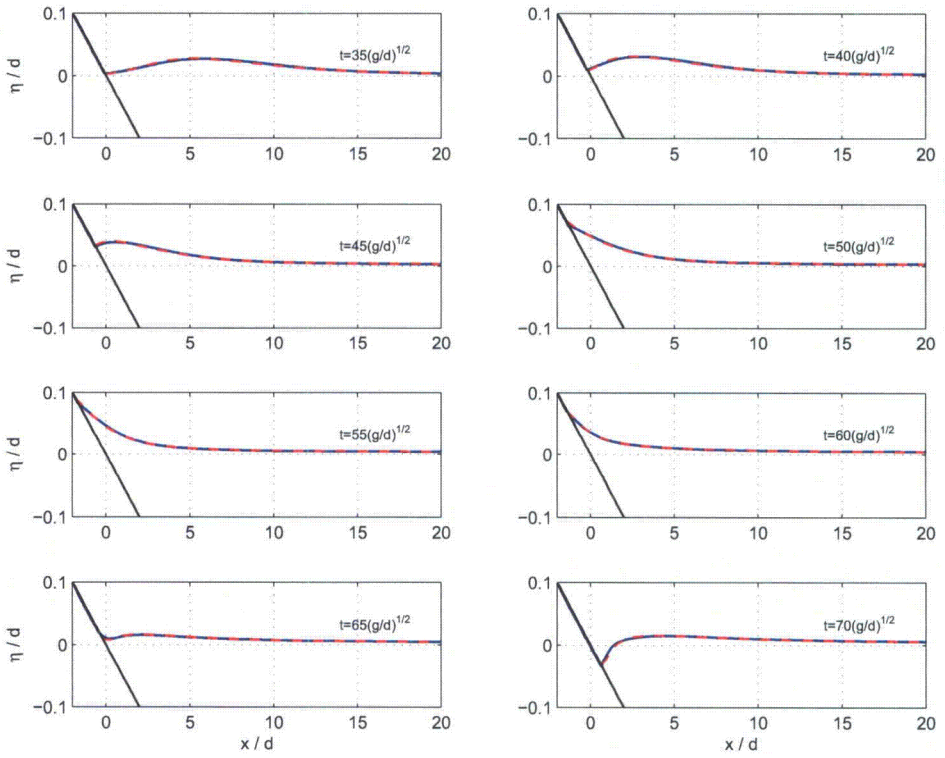

Figure 3: The water level profiles during runup of the non-breaking wave in the case of $H/d = 0.019$ on a 1:19.85 beach. Solid blue line represents the analytical solution in according to Synolakis(1986), and dashed red lines represents the numerical simulation.

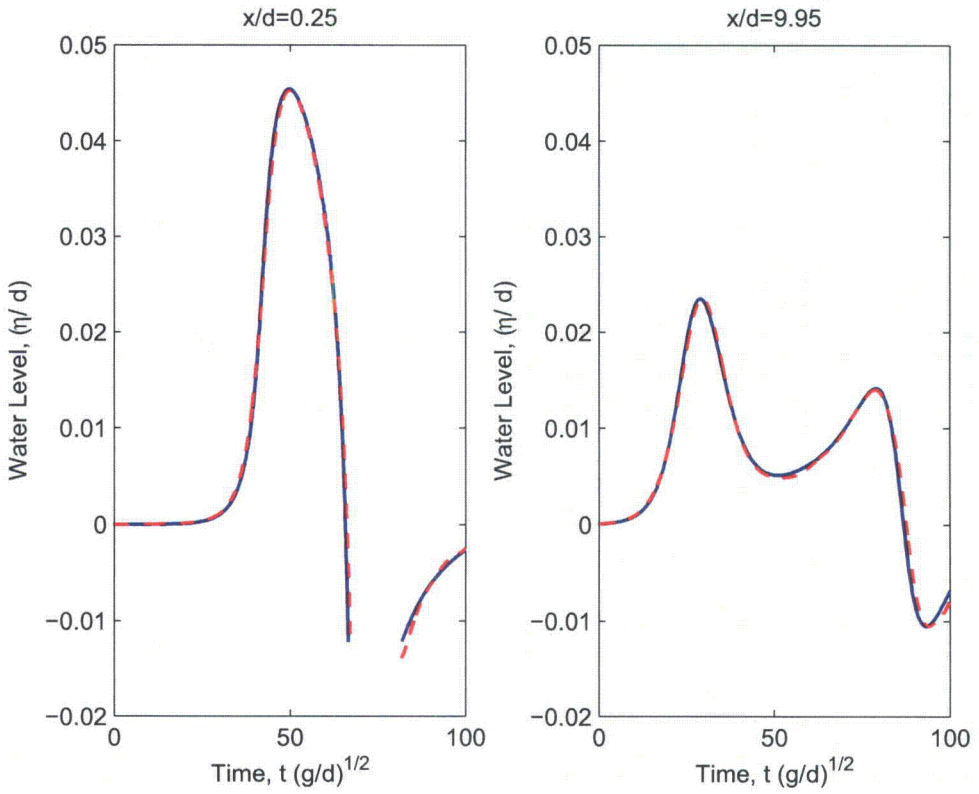

Figure 4: The water level dynamics at two locations $X/d = 0.25$ and $X/d = 9.95$. Solid blue line represents the analytical solution in according to Synolakis(1986), and dashed red line represents the numerical simulation.

					R/d		
d(m)	γ	Δx(m)	H/d	$\cot(\beta)$	Runup Law	Numerical Calculations	Error (%)
0.5	0.22	0.1	0.05	3.333	0.167	0.165	1.0
0.5	0.44	0.1	0.2	1.0	0.516	0.505	2.2
5.0	0.54	1.0	0.03	10.0	0.152	0.157	3.0
5.0	0.70	1.0	0.05	10.0	0.289	0.286	0.9
5.0	0.99	1.0	0.1	3.732	0.419	0.433	3.3
5.0	1.69	1.0	0.294	2.747	1.385	1.388	0.2
5.0	0.22	1.0	0.005	20.0	0.023	0.023	0.2
5.0	0.31	1.0	0.01	20.0	0.055	0.056	2.6
100.0	3.12	5.0	0.05	10.0	0.289	0.277	4.0
100.0	6.13	5.0	0.193	2.747	0.818	0.826	0.9
100.0	1.40	5.0	0.01	20.0	0.055	0.056	2.6
1000.0	9.87	20.0	0.05	10.0	0.289	0.275	4.7
1000.0	9.87	20.0	0.05	3.333	0.167	0.173	3.8

Table 3: Runup data from numerical calculations compared with runup law for N-wave.

Similar to previous part, following expression for the maximum runup of N-waves has been provided based on slope of the beach and wave height of the N-wave similar to solitary wave problem:

$$R = 3.86\sqrt{\cot\beta}H^{\frac{5}{4}} \tag{28}$$

Benchmark problems that are studied here have different depths from 50cm to 1000m. Also, for each depth different slope and wave heights has been studied. Table 3 provides a list of selected different cases that has been modeled including their maximum runup and the grid size for each case. Figure 5 defines a comparison between numerical simulation and runup law for N-wave problem.

4.3 Solitary wave on composite beach

In this benchmark problem a complex topography consisting of three segments and a vertical wall is considered (Figure 6). The benchmark test is described in Appendix A, Section 2.2 of Synolakis et al (2007). Runup of non-breaking solitary waves on the vertical wall is simulated in this case. Results have been compared with the analytical solution. Laboratory data exist for this topography from a U.S. Army Corps of Engineers, Coastal Engineering Research Center, Vicksburg, Mississippi experiment of wave runup on a model of Revere Beach, Massachusetts. However, the maximum runup for solitary waves propagating up Revere Beach (Composite Beach) is given by

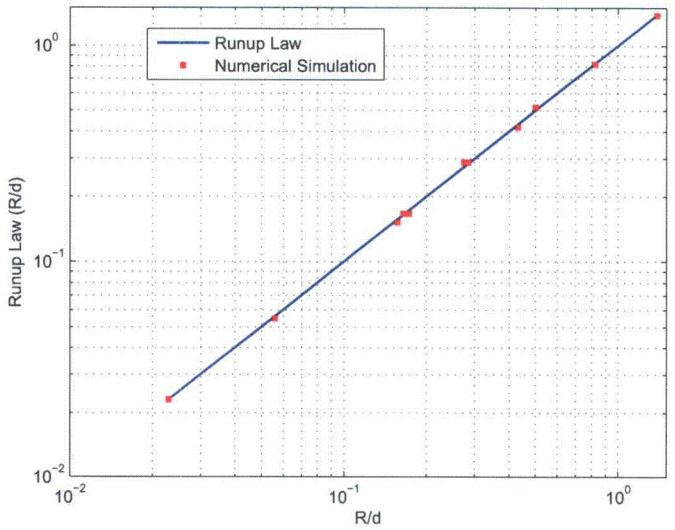

Figure 5: Numerical simulation data for maximum runup of N-waves climbing up different beach slopes. Solid line represents the runup law (28).

the runup law

$$R = 2h_w^{-\frac{1}{4}} H \tag{29}$$

where $h_w$is the initial depth at the right vertical wall and H is the solitary wave height.

Two initial depths have been studied here; $d = 18.8$ cm, and $d = 21.8$ cm. For all cases a grid size of $\Delta x = 0.10$m has been used. Table 4 demonstrates a comparison of runup law with the

$d = 21.8$ cm				$d = 18.8$ cm			
H(m)	R(Runup Law)	R(Numerical)	Error (%)	H(m)	R(Runup Law)	R(Numerical)	Error (%)
0.005	0.0146	0.0141	3.7	0.005	0.0152	0.0148	2.5
0.01	0.0293	0.0288	1.6	0.01	0.0304	0.0310	2.1
0.03	0.0878	0.0850	3.2	0.03	0.0911	0.0871	4.4
0.05	0.1463	0.1390	5.0	0.05	0.1519	0.1440	5.2

Table 4: Maximum runup of solitary wave on composite beach compared to runup law (29).

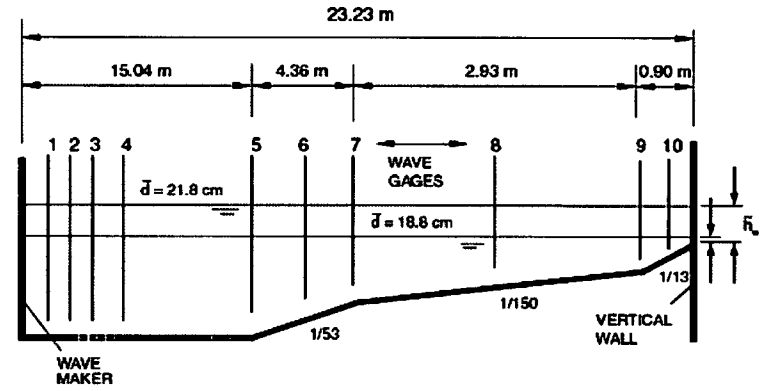

Figure 6: Definition sketch for Revere Beach (from Synolakis et al (2007, Figure A7)).

numerical data for both depths studied here.

In addition, for this benchmark problem three different waves have been modeled ($H/d = 0.0378, 0.2578, 0.6404$ for cases A, B and C). Recorded data is compared with the analytical data for gauge 4 to gauge 10 and also on the wall in Figures 7 - 9 . Grid size for this case is $\Delta x = 0.010$ m. It should be mentioned that for analytical results the model has been used in linear, non-dispersive and no friction mode.

5 Laboratory benchmarks

In this section different laboratory benchmarks are studied and result of numerical calculations is compared with the laboratory data. Following benchmark problems are studied in this section:

1. Solitary wave on a simple beach
2. Solitary wave on a composite beach
3. Solitary wave on a conical island
4. Tsunami runup onto a complex three-dimensional beach; Monai Valley

5.1 Solitary wave on a simple beach

In this laboratory test, the 31.73 m-long, 60.96 cm-deep and 39.97 cm wide California Institute of Technology, Pasadena, California wave tank was used with water at varying depths. The tank is

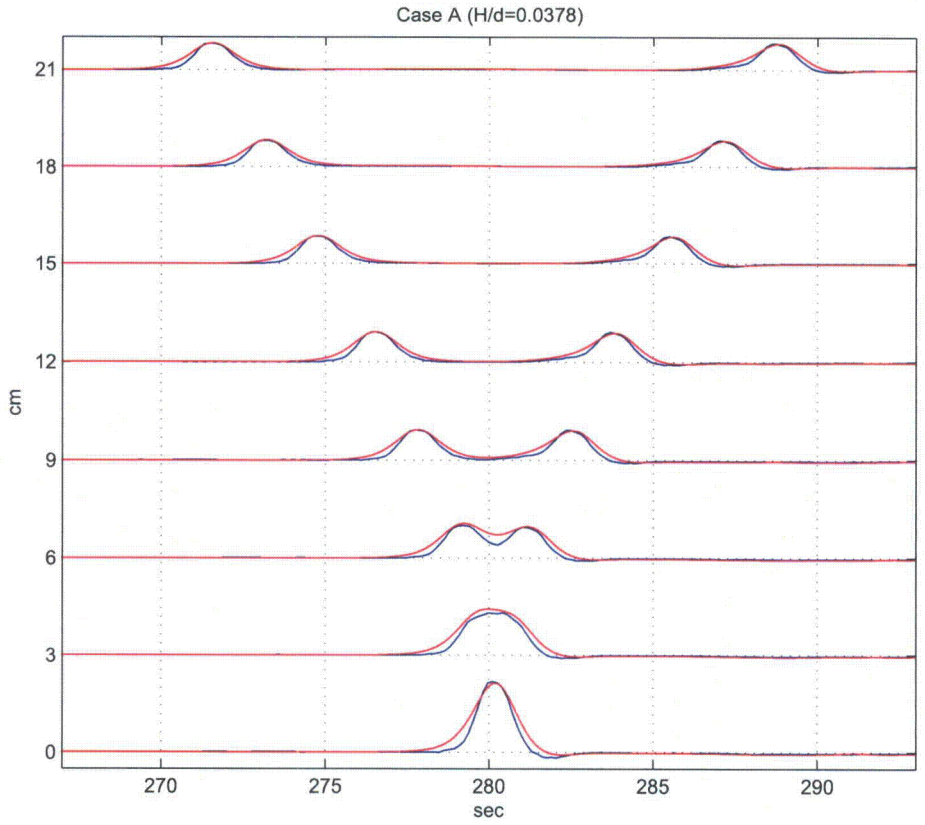

Figure 7: Time evolution of nonbreaking $H/d = 0.0378$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution.

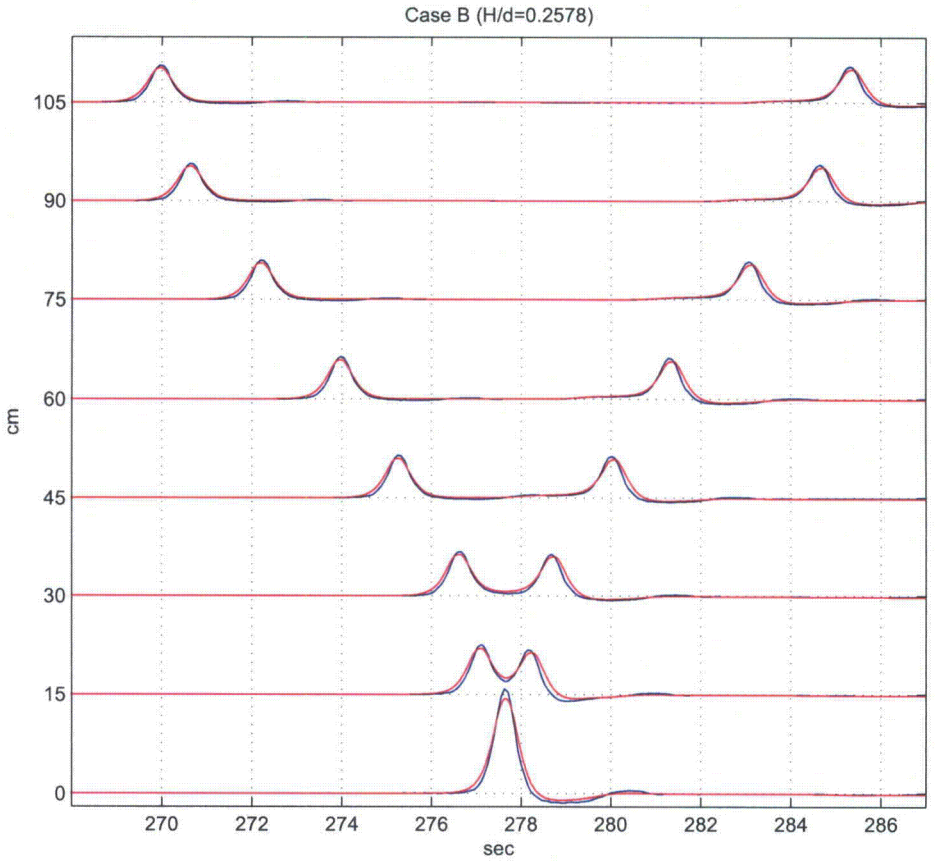

Figure 8: Time evolution of breaking $H/d = 0.2578$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution.

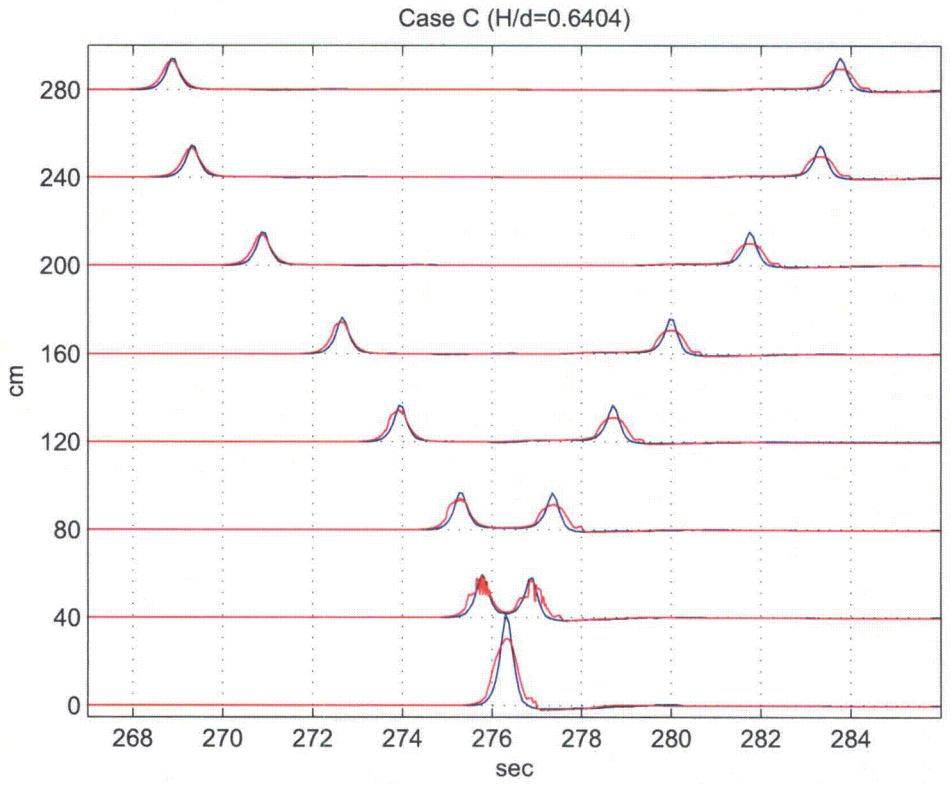

Figure 9: Time evolution of breaking $H/d = 0.6404$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the analytic solution.

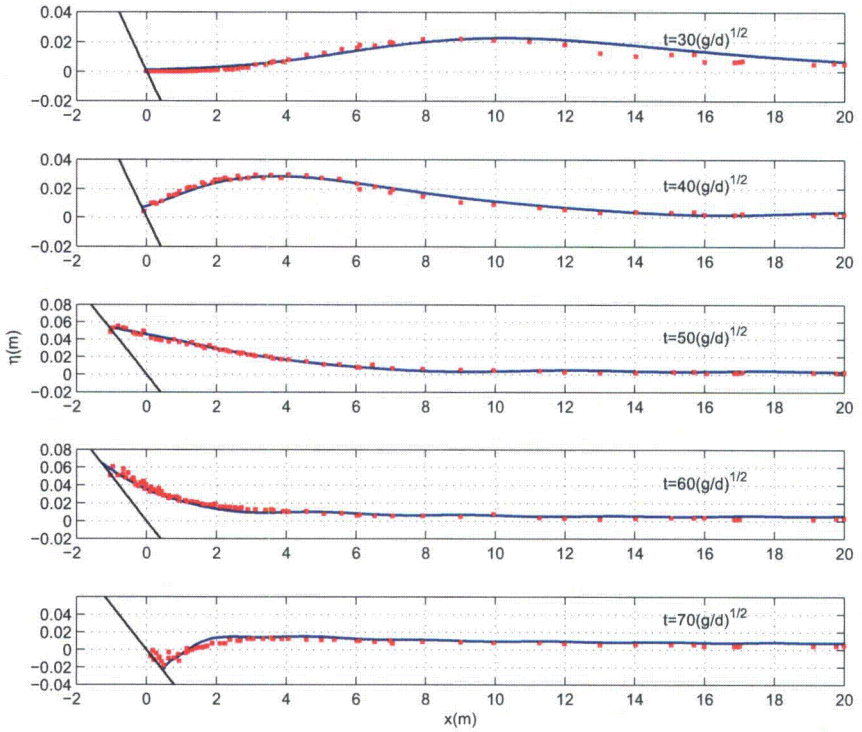

Figure 10: Time evolution of nonbreaking $H/d = 0.0185$ initial wave. The solid line shows the numerical solution and dots represent the laboratory data.

described by Synolakis (1986, 1987). The bottom of the tank consisted of painted stainless steel plates. A ramp was installed at one end of the tank to model the bathymetry of the canonical problem of a constant-depth region adjoining a sloping beach. The ramp had a slope of 1:19.85. The ramp was sealed to the tank side walls. The toe of the ramp was distant 14.95 m from the rest position of the piston generator used to generate waves.
This set of laboratory data has been vastly used for many code validations. In this modeling test, the data sets for the $\tilde{H}/\tilde{d} = 0.0185$ nonbreaking and $\tilde{H}/\tilde{d} = 0.30$ breaking solitary waves which are the most frequently used and most appropriate for code validation.

For these cases a grid size of $\Delta x = 0.10$ m has been used. Figure 10 and Figure 11 displays the accuracy of the model for both nonbreaking and breaking waves. The runup error for the

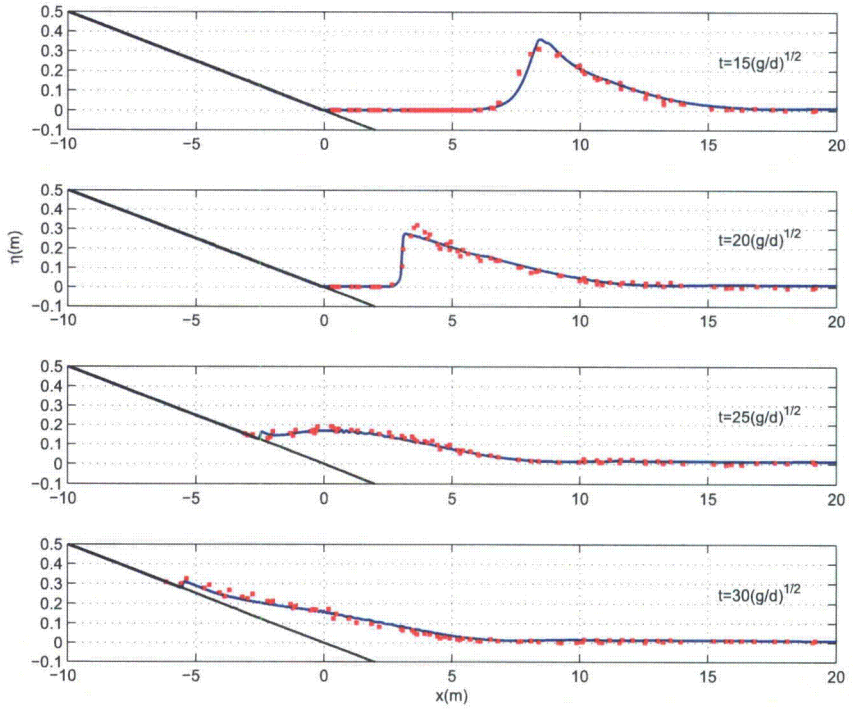

Figure 11: Time evolution of breaking $H/d = 0.3$ initial wave. The solid line shows the numerical solution and dots represent the laboratory data.

nonbreaking wave case was 3.3% and for the breaking wave was 5.8%.

5.2 Solitary wave on a composite beach

Revere Beach is located approximately 6 miles northeast of Boston in the City of Revere, Massachusetts. To address beach erosion and severe flooding problems, a physical model of the beach was constructed at the Coastal Engineering Laboratory of the U.S. Army Corps of Engineers, Vicksburg, Mississippi facility, earlier known as Coastal Engineering Research Center. This benchmark is described in Section 3.2 of Appendix A of Synolakis et al (2007).

The beach characteristics are exactly the same as the composite beach described in section 4.3. In this benchmark problem three different waves are modeled ($H/d = 0.0378, 0.2578, 0.6404$ for

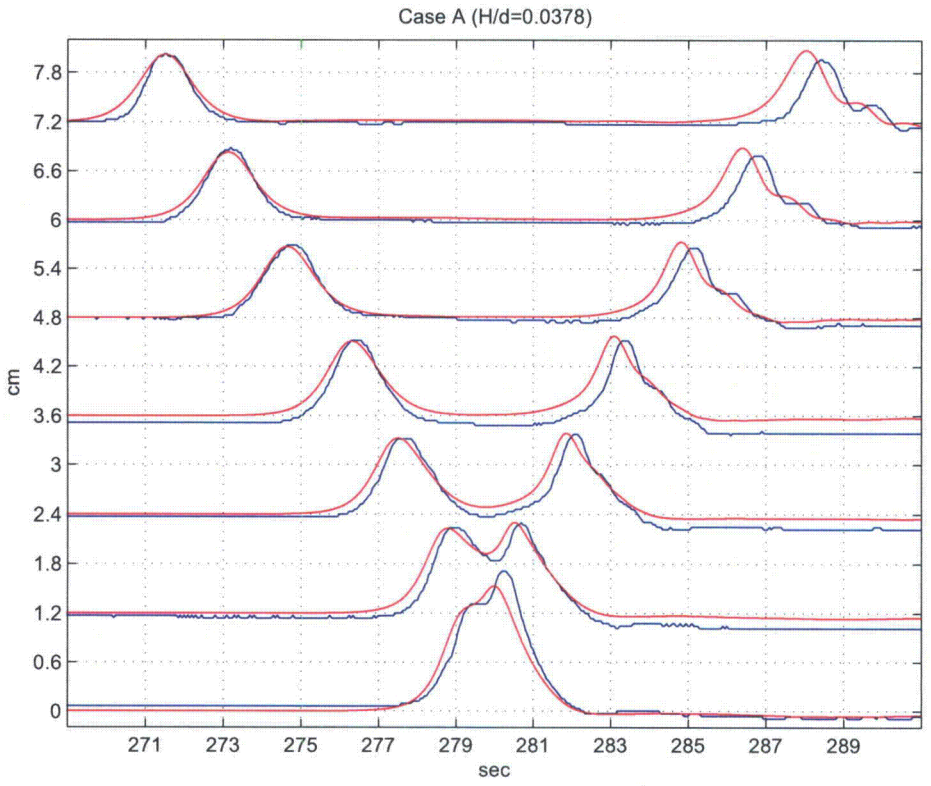

Figure 12: Time evolution of nonbreaking $H/d = 0.0378$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data.

cases A, B and C) and the numerical data is compared with the laboratory data for gauges 4 to 10 in Figures 12 - 14). Grid size for this case is $\Delta x = 0.010$ m.

5.3 Solitary wave on a conical island

Laboratory experiments on the interaction between solitary waves and a conical island were conducted by Briggs et al (1995). The three cases from this test illustrate the important fact that runup and inundation heights on the sheltered back sides of an island can exceed the incident wave height on the exposed front side, due to trapping of wave fronts propagating around the island circumference. These tests have been used in a number of validation studies for a variety of models,

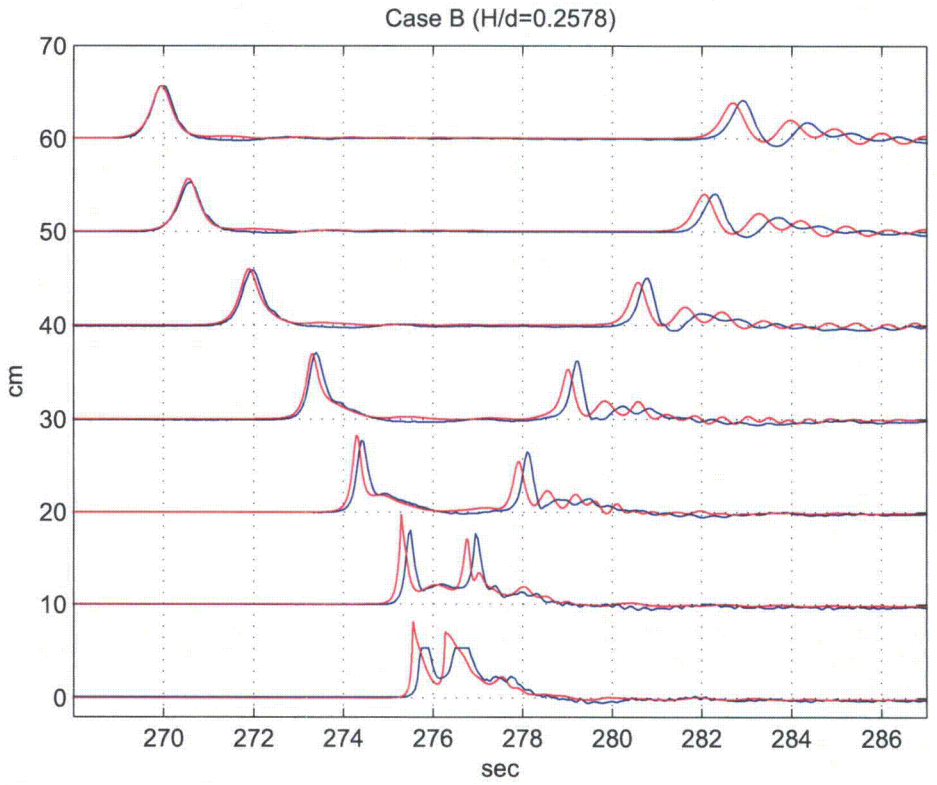

Figure 13: Time evolution of breaking $H/d = 0.2578$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data.

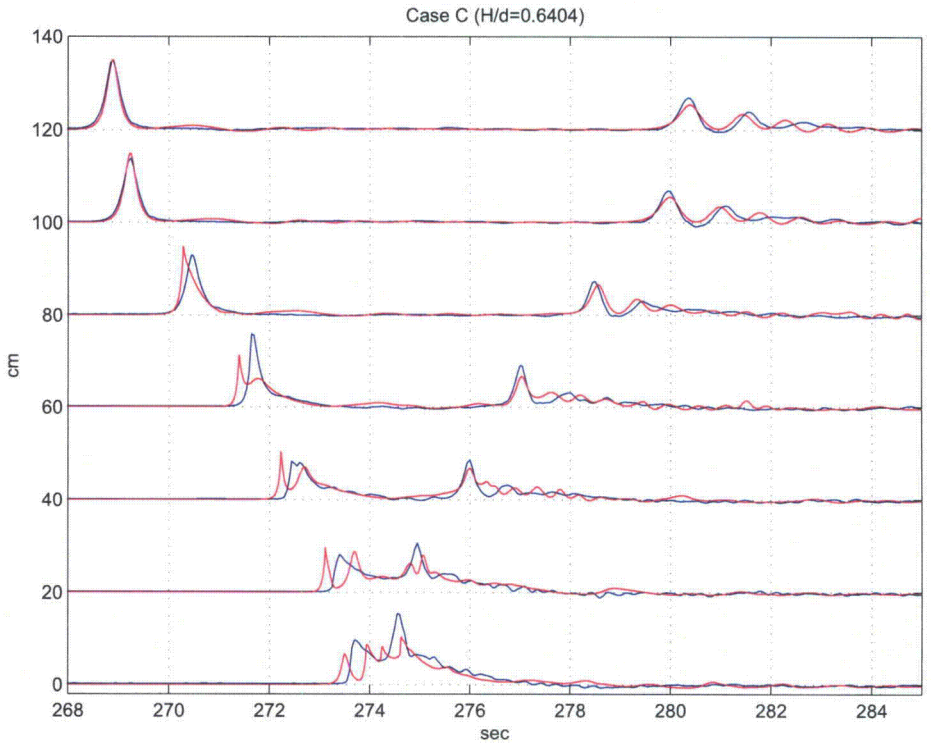

Figure 14: Time evolution of breaking $H/d = 0.6404$ initial wave on composite beach. The red line shows the numerical solution and blue line represents the laboratory data.

including nonlinear shallow water equations (Liu et al 1995) and Boussinesq equations (Chen et al, 2000). The benchmark test is specified in Section 3.3 of Appendix A of Synolakis et al (2007).

Large-scale laboratory experiments were performed at Coastal Engineering Research Center, Vicksburg, Mississippi, in a 30m-wide, 25m-long, and 60cm-deep wave basin (Figure 15). In the physical model, a 62.5cm-high, 7.2m toe-diameter, and 2.2m crest-diameter circular island with a 1:4 slope was located in the basin (Figure 16). Experiments were conducted at depth of 32cm, with three different solitary waves (H/d=0.045, 0.091, 0.181). Water-surface time histories were measured with 27 wave gages located around the perimeter of the island (Figure 17).

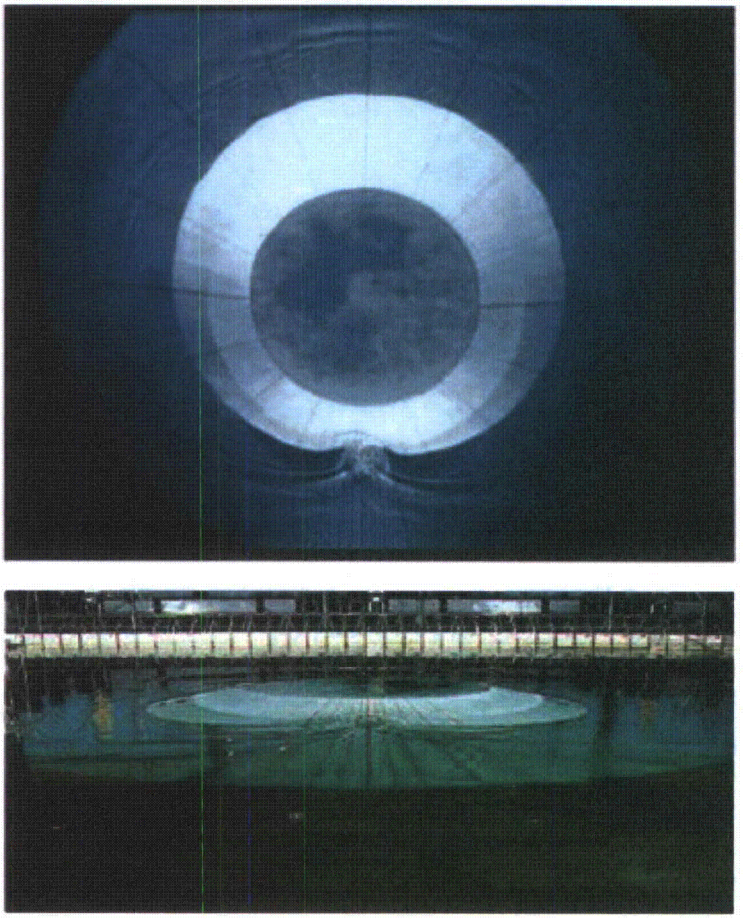

Figure 15: View of conical island(top) and basin(bottom)(from Synolakis et al (2007, Figure A16)).

For this benchmark test, time histories of the surface elevation around the circular island are given at four locations, i.e., in the front of the island at the toe (Gauge 6) and gauges closest to the

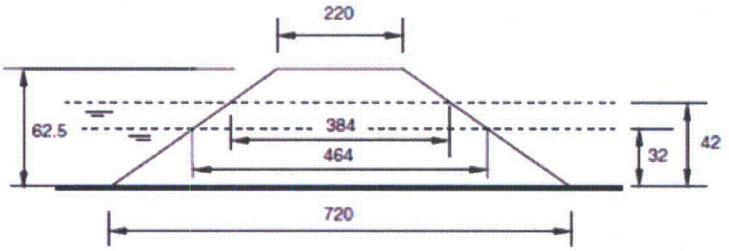

Figure 16: Definition sketch for conical island. All dimensions are in cm (from Synolakis et al (2007, Figure A17)).

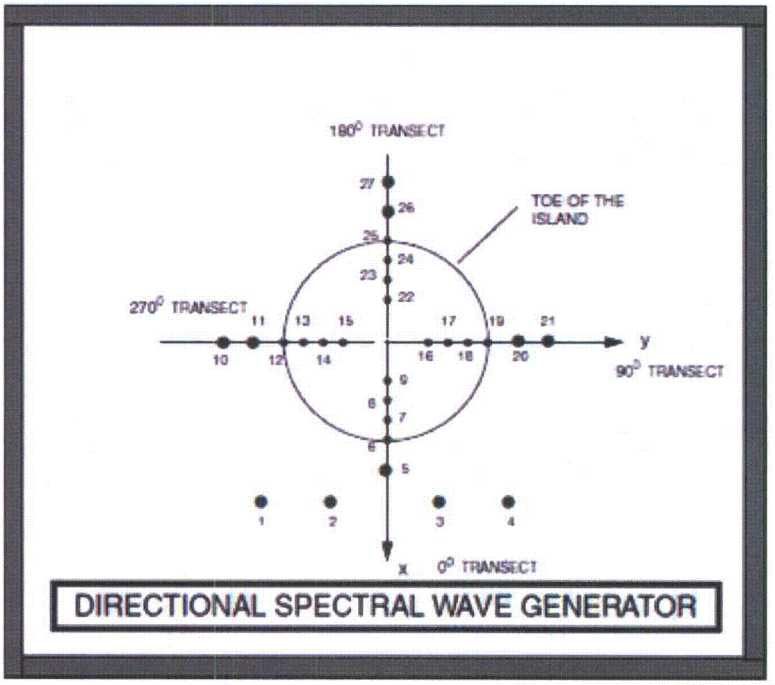

Figure 17: Schematic gauge locations around the conical island(from Synolakis et al (2007, Figure A18)).

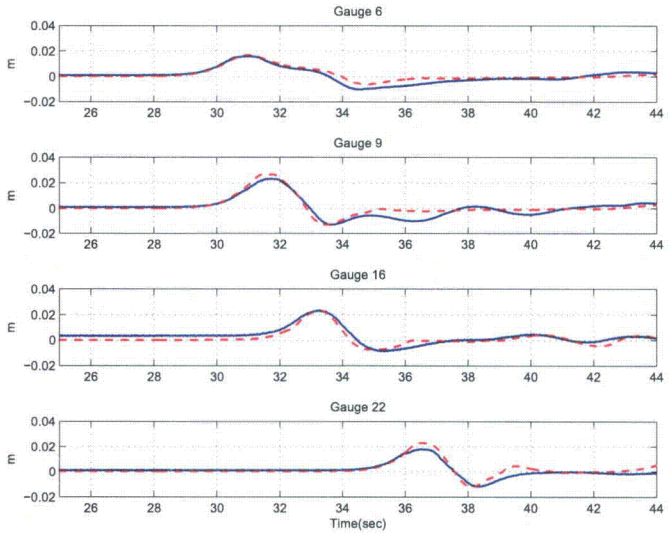

Figure 18: Comparison of computed and measured time series of free surface for $H/d = 0.045$.Solid lines: measured, Dashed lines: Computed.

shoreline with the numbers 9, 16, and 22 located at the $0°$, $90°$, and $180°$ radial lines (Figure 17). A grid size of $\Delta x = 0.10m$ is considered for proper numerical simulation of this benchmark. Figures 18-20 shows the comparison between the laboratory data with numerical calculations. Table 5 represents the error of the maximum runup for each gauge for different wave heights.

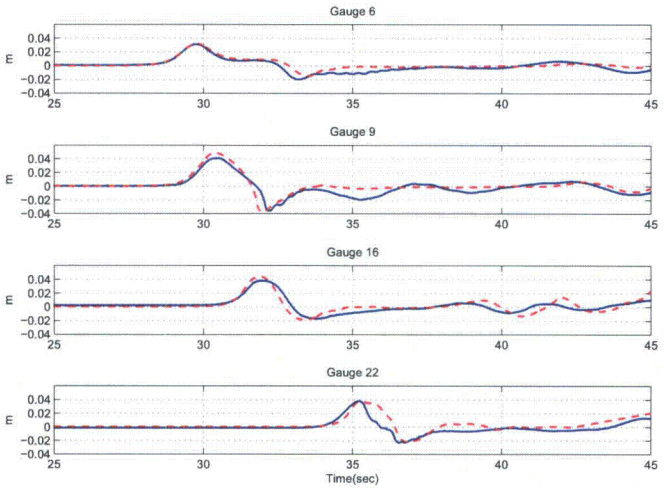

Figure 19: Comparison of computed and measured time series of free surface for $H/d = 0.091$.Solid lines: measured, Dashed lines: Computed.

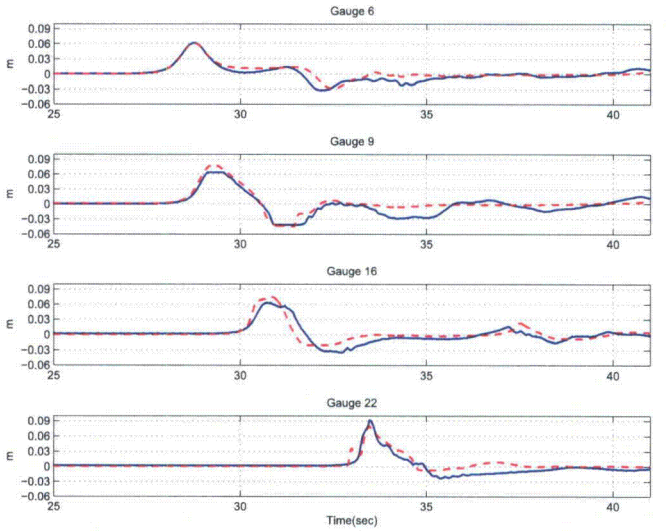

Figure 20: Comparison of computed and measured time series of free surface for $H/d = 0.181$.Solid lines: measured, Dashed lines: Computed.

	Gauge Number			
H/d	6	9	16	22
0.045	6.0	13.2	0.1	18.9
0.091	3.2	16.6	11.6	0.26
0.181	1.6	13.33	13.8	13.3

Table 5: Percent error of predicted maximum runup calculated for each gauge in conical island test.

5.4 Tsunami runup onto a complex three-dimensional beach; Monai Valley

The Hokkaido-Nansei-Oki tsunami of 1993 that struck Okushiri Island, Japan, provided high-quality data for tsunami researchers. Since maximum tsunami runup mark was discovered at the tip of a very narrow gulley within a small cove at Monai, a laboratory benchmark was designed based on Monai valley bathymetry and tsunami wave which struck this area. Based on high resolution seafloor bathymetry existed before the event a 1/400 laboratory model of Monai was constructed in a 205m-long, 6m deep, and 3.5m-wide tank at Central Research Institute for Electric Power Industry (CRIEPI) in Abiko, Japan and partly shown in Figure 21. The incident wave from offshore, at the water depth of d = 13.5 cm is known and it is shown in Figure 22. There are reflective vertical sidewalls at y = 0 and 3.5 m (Figure 23). The entire computational area for the laboratory test is 5.448m $\times$3.402m, and the grid sizesrecommended for numerical simulations are $\Delta x = \Delta y = 1.4$cm. However, due to numerical limitations, computational domain that is used in numerical simulation is longer in order to generate waves without any reflection disturbance from the back wall(12.488m $\times$3.402m)(Figure **??**). The input wave is a LDN with a leading-depression height of 2.5 mm with a crest of 1.6 cm behind it which is produced in the model using FFT analysis and WK TIME SERIES wavemaker option(Figure 22). Data for water surface elevations during the laboratory experiment is given and compared with numerical simulations at three locations (Gauge 5, 7 and 9), i.e., (x, y) = (4.521, 1.196), (4.521, 1.696), and (4.521, 2.196) in meters (Figure 24). Figure 25 represents the comparison between extracted movie frames from the overhead movie of the laboratory experiment and numerical simulation. Finally, maximum runup in the narrow galley in the numerical solution was 7.43cm which is comparable with the laboratory data (maximum runup of 7.5cm in the lab data or 30m in field tsunami data).

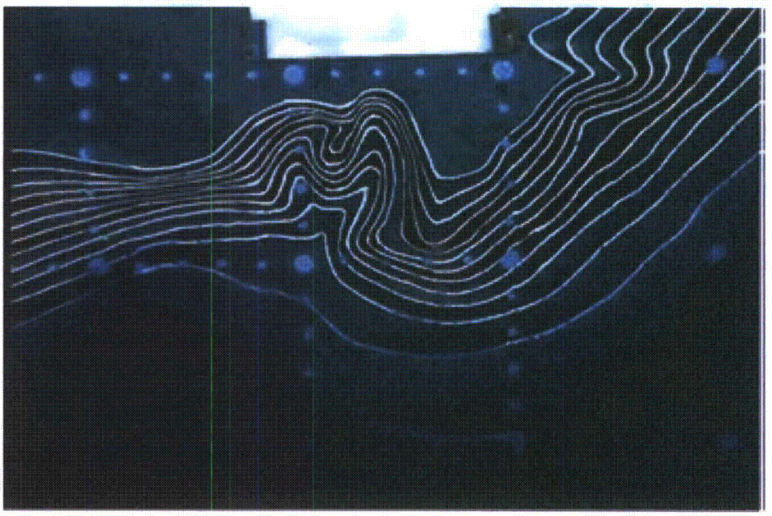

Figure 21: Bathymetric profile for experimental setup for Monai Valley experiment(2007, Figure A24)).

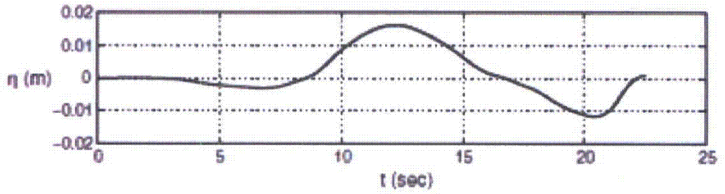

Figure 22: Initial wave profile for Monai Valley experiment (2007, Figure A25)).

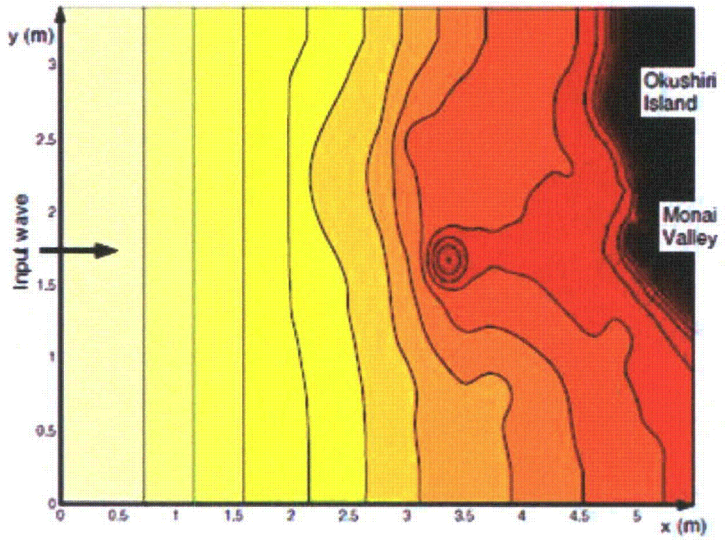

Figure 23: Computational area for Monai Valley experiment(2007, Figure A26)).

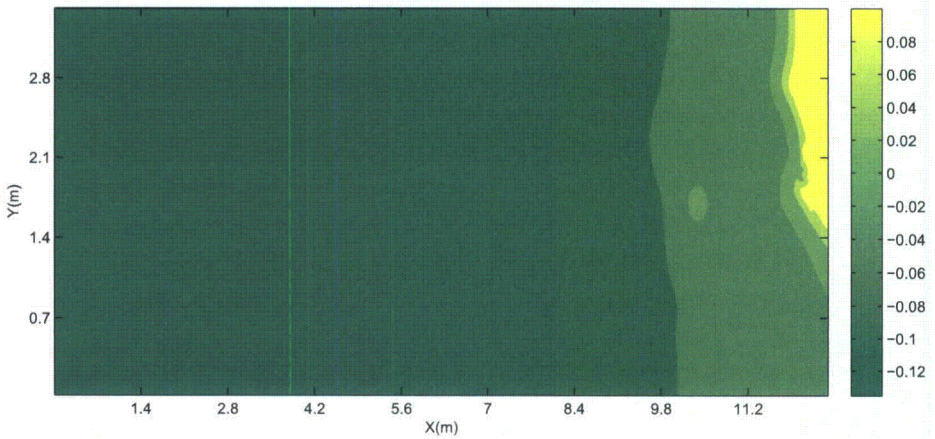

Figure 24: Computational area for Monai Valley numerical simulation.

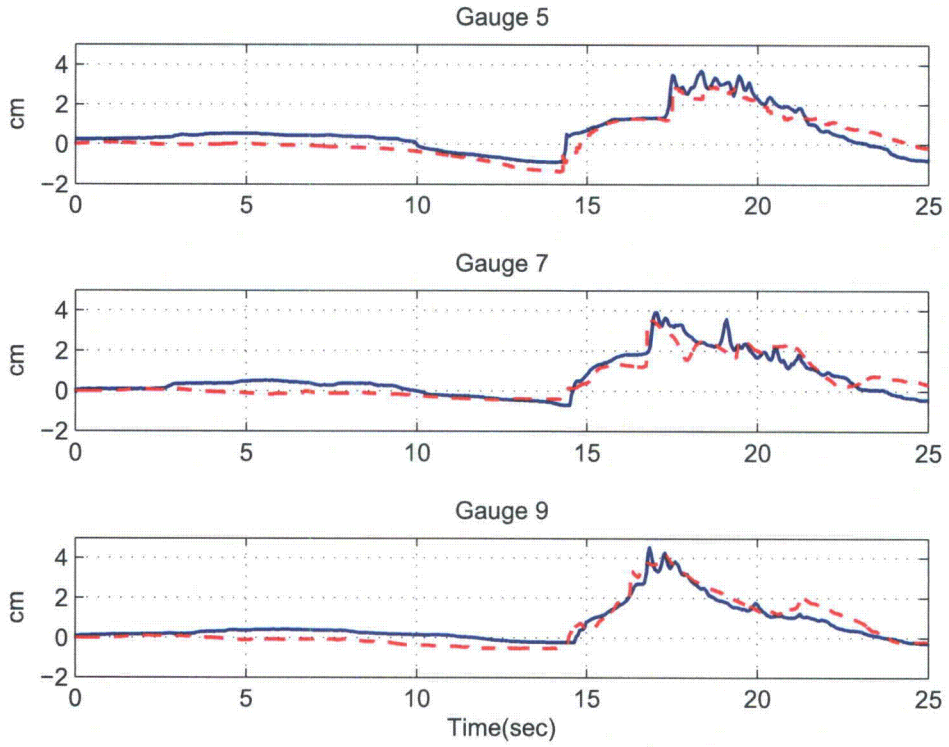

Figure 25: Comparison of computed and measured time series of free surface. Dashed lines: Computed, Solid lines: Measured.

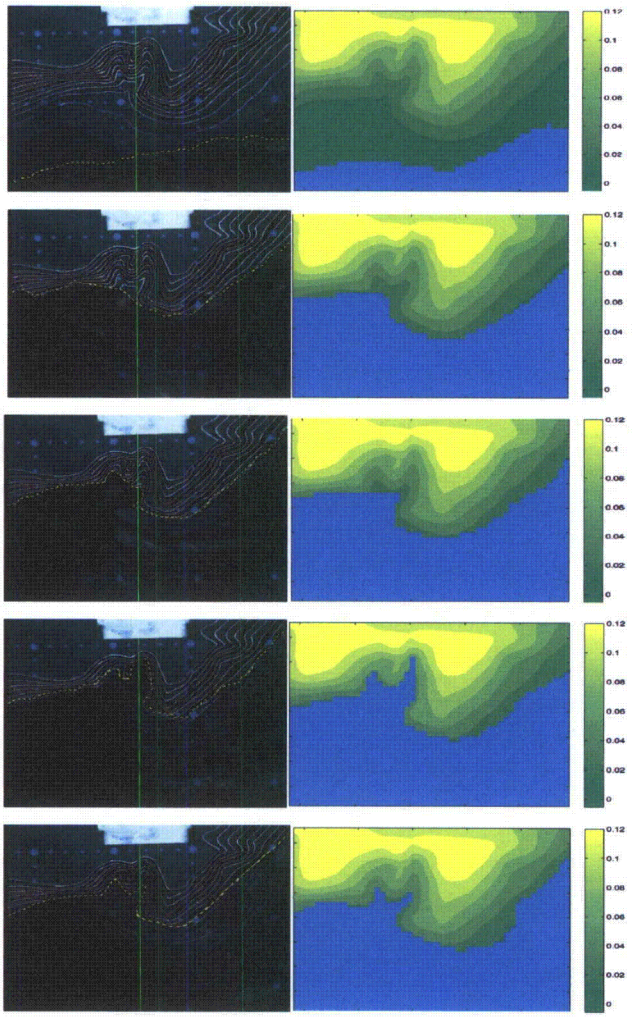

Figure 26: Comparison between extracted movie frames from the overhead movie of the laboratory experiment (left) (from http://burn.giseis.alaska.edu/file_doed/Dmitry/BM7_description.zip) and numerical simulation (right).

References

Briggs, M. J., Synolakis, C. E., Harkins, G. S. and Green, D., 1995, "Laboratory experiments of tsunami runup on a circular island", *Pure Appl. Geophys.*, **144**, 569-593.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Kennedy, A. B. and Chawla, A., 2000, "Boussinesq modelling of wave transformation, breaking and runup. II: Two horizontal dimensions", *J. Waterway, Port, Coast. Ocean Engnrng*, **126**, 48-56.

Chen, Q., Kirby, J. T., Dalrymple, R. A., Shi, F. and Thornton, E. B., 2003, "Boussinesq modelling of longshore currents", *J. Geophys. Res.*, **108**(C11), 3362, doi:10.1029/2002JC001308.

Chen, Q., 2006, "Fully nonlinear Boussinesq-type equations for waves and currents over porous beds", *J. Eng. Mech.*, **132**, 220-230.

Gottlieb, S., Shu C.-W., and Tadmore, E., 2001, "Strong stability-preserving high-order time discretization methods", *SIAM Review*, **43** (1), 89 - 112.

Kennedy, A. B., Kirby, J. T., Chen, Q. and Dalrymple, R. A., 2001, "Boussinesq-type equations with improved nonlinear performance", *Wave Motion*, **33**, 225-243.

Kim, D. H., Cho, Y. S. and Kim, H. J., 2008, "Well balanced scheme between flux and source terms for computation of shallow-water equations over irregular bathymetry", *J. Eng. Mech.*, **134**, 277-290.

Kirby, J. T., Wei, G., Chen, Q., Kennedy, A. B. and Dalrymple, R. A., 1998, " FUNWAVE 1.0, Fully nonlinear Boussinesq wave model. Documentation and user's manual", Research Report No. CACR-98-06, Center for Applied Coastal Research, University of Delaware.

Liang, Q. and Marche, F., 2009, "Numerical resolution of well-balanced shallow water equations with complex source terms", *Adv. Water Res.*, **32**, 873 - 884.

Liu, P. L.-F., Cho, Y. S., Briggs, M. S., Kânoğlu, U. and Synolakis, C. E., 1995, "Runup of solitary waves on a circular island", *J. Fluid Mech.*, **320**, 259-285.

Naik, N. H., Naik, V. K., and Nicoules, M., 1993, "Parallelization of a class of implicit finite difference schemes in computational fluid dynamics", *Int. J.High Speed Computing*, **5**, 1-50.

Nwogu, O., 1993, "An alternative form of the Boussinesq equations for nearshore wave propagation", *J. Waterway, Port, Coastal, and Ocean Engineering*, **119**, 618-638.

O'Sullivan, B., 2009, *Mercurial: The definitive guide*, O'Reilly Media Inc.

Roeber, V., Cheung, K. F., and Kobayashi, M. H., 2010, "Shock-capturing Boussinesq-type model for nearshore wave processes", *Coastal Engineering*, **57**, 407-423.

Rogers, B. D., Borthwick, A. G. L., and Taylor, P. H., 2003, "Mathematical balancing of flux gradient and source terms prior to using Roe's approximate Riemann solver", *J. Comp. Phys.*, **192**, 422-451.

Russel, J.S., 1845, "Report on waves", *Rp. Meet. Brit. Assoc. Adv. Sci. 14th*, 311? 390, John Murray, London.

Shi, F., Dalrymple, R. A., Kirby, J. T., Chen, Q. and Kennedy, A., 2001, "A fully nonlinear Boussinesq model in generalized curvilinear coordinates", *Coastal Engineering*, **42**, 337-358.

Shi, F., Kirby, J. T., Harris, J. C., Geiman, J. D. and Grilli, S. T., 2011a, "A high-order adaptive time-stepping TVD solver for Boussinesq modelling of breaking waves and coastal inundation", In Prepration for Ocean Modelling.

Shi, F., Kirby, J. T., Tehranirad, B., Harris, J. C. and Grilli, S. T., 2011b, "FUNWAVE-TVD Version 1.0. Fully nonlinear Boussinesq wave model with TVD solver. Documentation and user's manual", Research Report No. CACR-11-04, Center for Applied Coastal Research, University of Delaware.

Shiach, J. B. and Mingham, C. G., 2009, "A temporally second-order accurate Godunov-type scheme for solving the extended Boussinesq equations", *Coast. Eng.*, **56**, 32-45.

Synolakis, C.E., 1986, "The runup of long waves", Ph.D. Thesis, California Institute of Technology, Pasadena, California, 91125, 228 pp.

Synolakis, C.E., 1987, "The runup of solitary waves", *J. Fluid Mech.*, **185**, 523?545.

Synolakis, C. E., Bernard, E. N., Titov, V. V., Kânoğlu, U. and González, F. I., 2007, "Standards, criteria, and procedures for NOAA evaluation of tsunami numerical models", NOAA Tech. Memo. OAR PMEL-135, Pacific Marine Env. Lab., Seattle.

Tadepalli, S. and C.E. Synolakis, 1994, "The runup of N-waves on sloping beaches", *Proc. R. Soc. A*, **445**, 99-112.

Tonelli, M. and Petti, M., 2009, "Hybrid finite volume - finite difference scheme for 2DH improved Boussinesq equations", *Coast. Engrng.*, **56**, 609-620.

Wei, G., Kirby, J.T., Grilli, S.T., Subramanya, R., 1995, "A fully nonlinear Boussinesq model for surface waves: Part I. Highly nonlinear unsteady waves", *J. Fluid Mech.*, **294**, 71-92.

Yamamoto, S. and Daiguji, H., 1993, "Higher-order-accurate upwind schemes for solving the compressible Euler and Navier-Stokes equations", *Computers and Fluids*, **22**, 259-270.

Yamamoto, S., Kano, S. and Daiguji, H, 1998, "An efficient CFD approach for simulating unsteady hypersonic shock-shock interference flows", *Computers and Fluids*, **27**, 571-580.

Zhou, J. G., Causon, D. M., Mingham C. G., and Ingram, D. M., 2001, "The surface gradient method for the treatment of source terms in the shallow-water equations", *J. Comp. Phys.*, **168**, 1-25.

Sensitivity analysis of trans-oceanic tsunami propagation to dispersive and Coriolis effects

James T. Kirby[1], Fengyan Shi[1], Jeffrey C. Harris[2] and Stephan T. Grilli[2]

[1]Center for Applied Coastal Research
University of Delaware, Newark, DE 19716 USA
[2]Ocean Engineering Department
University of Rhode Island, Narragansett, RI something USA

Abstract

We derive weakly nonlinear, weakly dispersive model equations for propagation of surface gravity waves in a shallow, homogeneous ocean of variable depth on the surface of a rotating sphere. A numerical model is developed based on a combined finite-volume and finite-difference method with a forth-oder MUSCL-TVD scheme in space and a third-order SSP Runge-Kutta scheme in time. Scaling analysis reveals that the importance of frequency dispersion increases with a decrease of the source width, and the Coriolis force increases with an increase of the source width. A sensitivity analysis to dispersive and Coriolis effects is carried out using the numerical model through a series of numerical experiments in an idealized ocean using Okada sources with different source sizes.

1 Introduction

Conventional models used for simulating global-scale tsunamis are based on the shallow water equations, which neglect frequency dispersion effects on wave propagation. Recent studies, however, revealed that these standard models may not be satisfactory for simulating tsunamis caused by smaller scale or more concentrated non-seismic sources, such as landslides (e.g., Lovholt et al., 2008; Tappin et al., 2008). Moreover, even for very long waves such as found in co-seismic tsunamis, frequency dispersion effects may become significant in the long distance propagation of tsunami fronts. This was evidenced by Kulikov's (2005) wavelet frequency analysis of satellite altimetry data recorded in deep water in the Bay of Bengal during the 2004 Indian Ocean tsunami. Based on these, Kulikov concluded that a dispersive long wave model should be used for this event. Further supporting this conclusion, Glimsdal et al. (2006) and Grue et al. (2008) showed, in their dispersive numerical simulations of the same event, that an undular bore could evolve in shallow water, in accordance with observations (Shuto, 1985). Finally, using the dispersive Boussinesq model FUNWAVE (Kirby et al., 1998) to simulate the same event, Grilli et al. (2007) and Ioualalen et al. (2007) quantified dispersive effects, by performing simulations with and without dispersive terms (thus solving Nonlinear Shallow Water Equations (NSWE) in the latter case). Differences of

up to 20% in surface elevations, between Boussinesq and NSWE simulations, were found in deeper water.

Regarding effects of sphericity and earth rotation on tsunami propagation, even for the large 2004 event, numerical results showed that a Cartesian implementation of the models (neglecting Coriolis effects) is adequate for regional scale tsunami simulations, provided distances are corrected to account for earth's sphericity (e.g., Grilli et al., 2007; Ioualalen et al., 2007); this is particularly so when the main direction of propagation closely follows a great circle. For global tsunami propagation, however, sphericity and Coriolis effects might play a larger role in simulating tsunami wave arrival at far distant tide gauges. While standard NSWE tsunami simulation models have typically included such effects (e.g., Shuto et al., 1990), dispersive Boussinesq models such as FUNWAVE, which were initially developed for modeling ocean wave transformations from intermediate water depths to the coast, have usually been implemented in Cartesian coordinates. Lovholt et al. (2008) recently developed a Boussinesq model in spherical coordinates, including Coriolis effects. Their simulations quantified the effects of earth's rotation and the importance of Coriolis forces, on the far-field propagation across the Atlantic Ocean of a potential tsunami originated in La Palma (Canary Islands).

Based on this recent work, it appears that Boussinesq models may be more accurate tools than standard models, based on shallow water theory, for predicting the wide variety of tsunami wave phenomena that occur in nature, in which dispersive effects may play a more significant role than generally believed. However, the computational demands of such simulations is a concern. As pointed out by Yoon (2002), Boussinesq models require vast computer resources due to the implicit nature of the solution technique used to deal with dispersion terms. Some simulations may involve a wide range of scales of interest, from propagation out of the generation region, through propagation at ocean basin scale, to runup and inundation at affected shorelines (Grilli et al., 2007). Improvement in model efficiency can be achieved by using nested grids (e.g., Ioualalen et al., 2007), unstructured or curvilinear grids (Shi et al., 2001) and the parallelization of computational algorithm (Sitanggang and Lynett, 2005; Pophet, 2008; Pophet et al., 2008).

In this study, we rigorously derive and validate equations for a weakly nonlinear and dispersive Boussinesq model on the surface of a rotating sphere, including Coriolis effects. The numerical scheme is developed following the most recent work of Shi et al. (2011), who applied TVD Riemann solver to the FUNWAVE model. The model is implemented using a domain decomposition technique and optimized for parallel computer clusters, by way of the message passing interface (MPI). The efficiency of the model parallel implementaiton is assessed using various numbers of computer processors. The relative importance of frequency dispersion and Coriolis effects in tsunami propagation is finally evaluated both theoretically and based on numerical simulations of an idealized case.

2 Model Equations in Spherical Polar Coordinates

We consider the motion of a fluid column, of variable still water depth $h'(\phi,\theta)$, on the surface of a sphere of radius r'_0 to the still water level, where coordinates (r',θ,ϕ) denote radial distance from the sphere center, latitude, and longitude, with the local vertical coordinate defined as $z' = r' - r'_0$. The dimensional Euler equations describing the flow of an incompressible, inviscid fluid are given by (Pedlosky 1979, section 6.2),

$$w'_{z'} + \frac{2w'}{r'} + \frac{1}{r'\cos\theta}(v'\cos\theta)_\theta + \frac{1}{r'\cos\theta}u'_\phi = 0 \tag{1}$$

$$\frac{du'}{dt'} + \frac{u'}{r'}(w' - v'\tan\theta) + 2\Omega(w'\cos\theta - v'\sin\theta) = -\frac{1}{\rho r'\cos\theta}p'_\phi \tag{2}$$

$$\frac{dv'}{dt'} + \frac{1}{r'}(v'w' + (u')^2\tan\theta) + 2\Omega\sin\theta\, u' = -\frac{1}{\rho r'}p'_\theta \tag{3}$$

$$\frac{dw'}{dt'} - \frac{(u')^2 + (v')^2}{r'} - 2\Omega\cos\theta\, u' = -\frac{1}{\rho}p'_{z'} - g \tag{4}$$

where Ω is the sphere's angular velocity around the absolute vertical axis, u' and v' are positive velocity components in the Easterly (ϕ) and Northerly (θ) directions (local horizontal directions), respectively, and w' denotes the local vertical velocity. In the selected coordinate system, the total time derivative operator is defined as,

$$\frac{d()}{dt'} = ()_{t'} + \frac{u'}{r'\cos\theta}()_\phi + \frac{v'}{r'}()_\theta + w'()_{z'} \tag{5}$$

Boundary conditions consist in a dynamic condition, specifying pressure p_s on the free surface as,

$$p'_s(\phi,\theta,t') = 0; \qquad z' = \eta' \tag{6}$$

together with kinematic constraints on the velocity field at the surface and bottom boundary. The kinematic surface boundary condition (KSBC) is given by

$$\frac{D\eta'}{Dt'} = w'; \qquad z' = \eta' \tag{7}$$

and the (kinematic) bottom boundary condition (BBC) is given by

$$\frac{D(-h')}{Dt'} = w'; \qquad z' = -h' \tag{8}$$

where,

$$\frac{D()}{Dt'} = \frac{d()}{dt'} - w'()_{z'} = ()_{t'} + \frac{u'}{r'\cos\theta}()_\phi + \frac{v'}{r'}()_\theta \tag{9}$$

Note that (8) allows for an imposed motion of the ocean bottom to be specified.

In Boussinesq or shallow water theory, it is typical to replace the local continuity equation (1) with a depth-integrated conservation equation for horizontal volume fluxes. Thus, integrating (1)

over depth and employing the kinematic boundary conditions (7) and (8) yields,

$$(r')^2|_{\eta'}\eta'_{t'} + (r')^2|_{-h'}h'_{t'} + \frac{1}{\cos\theta}\frac{\partial}{\partial\phi}\left\{\int_{-h'}^{\eta'} u'r'dz'\right\} + \frac{1}{\cos\theta}\frac{\partial}{\partial\theta}\left\{\int_{-h'}^{\eta'} v'r'dz'\right\} = 0 \qquad (10)$$

which, as of yet, is still an exact equation. The latter will be simplified below, based on scaling arguments.

2.1 Scaling

Based on the standard procedure for shallow water scaling, we introduce the length scales h'_0, a'_0 and λ', denoting a characteristic water depth, wave amplitude, and horizontal length; with the latter remaining to be examined. Combining these scales with each other and with r'_0 yields a family of dimensionless parameters: $\epsilon = h'_0/r'_0$ denoting the relative depth or thickness of the ocean layer; $\mu = h'_0/\lambda'$, the usual parameter characterizing frequency dispersion in Boussinesq theory; and $\delta = a'_0/h'_0$, the shallow water nonlinearity parameter. The parameter ϵ takes on values of $O(10^{-3})$ at maximum, and will thus always be taken to indicate vanishingly small effects when it occurs in isolation.

Based on this family of parameters, we define dimensionless variables as,

$$(h, z) = \frac{(h', z')}{h'_0}; \quad \eta = \frac{\eta'}{a'_0} \qquad (11)$$

We take $u'_0 = \delta c'_0 = \delta\sqrt{gh'_0}$ to denote a scale for horizontal velocities, and let w'_0 denote a scale for vertical velocities, so that,

$$(u, v) = \frac{(u', v')}{u'_0}; \quad w = \frac{w'}{w'_0} \qquad (12)$$

Pressure is scaled by the weight of the static reference water column,

$$p = \frac{p'}{\rho g h'_0} \qquad (13)$$

We first introduce a rescaling of the dimensionless lattitude and longitude according to,

$$(\phi^*, \theta^*) = \frac{r'_0}{\lambda'}(\phi, \theta) = \frac{\mu}{\epsilon}(\phi, \theta); \qquad (14)$$

This gives horizontal coordinates, which will change by $O(1)$ amounts over the wavelength of a relatively short wave. The nondimensional form of the continuity equation (1) is then given by (retaining terms to $O(\epsilon)$)

$$\left(\frac{w'_0}{u'_0}\right)[w_z + 2\epsilon w] + \frac{\mu(1-\epsilon z)}{\cos\theta}[u_{\phi^*} + (v\cos\theta)_{\theta^*}] = O(\epsilon^2) \qquad (15)$$

indicating that $w'_0 = \mu u'_0$, as is usual in a Boussinesq model framework. Turning to the depth-integrated mass conservation equation (10), we introduce the total depth,

$$H = h + \delta\eta \qquad (16)$$

and obtain,

$$H_t + \frac{\delta}{\cos\theta}\left[(H\overline{u})_{\phi^*} + (H\overline{v}\cos\theta)_{\theta^*}\right] = O(\epsilon) \tag{17}$$

where,

$$(\overline{u}, \overline{v}) = \frac{1}{H}\int_{-h}^{\delta\eta}(u, v)dz \tag{18}$$

are depth-averaged horizontal velocities, and where time t' is scaled according to

$$t = \omega' t' = \frac{\sqrt{gh'_0}}{\lambda'}t' \tag{19}$$

Equation (17) is the final volume flux conservation equation for the Boussinesq model system, based on $(\overline{u}, \overline{v})$ developed below. In keeping with the notion that waves, which are short relative to the basin scale (λ'/r'_0 or $\epsilon/\mu \ll 1$), may have frequencies, which are high relative to the earth's rotation rate ($\omega'/\Omega >> 1$), we introduce the scaling $\Omega = \frac{\Omega'}{\omega'} = O(\frac{\epsilon}{\mu})$ Turning to the Easterly (ϕ) momentum equation (2), we obtain,

$$u_t - \left(\frac{\epsilon}{\mu}\right) f(v - \mu w\tan\theta) + \delta\left[\frac{u}{\cos\theta}u_{\phi^*} + vu_{\theta^*} + wu_z - \frac{\epsilon}{\mu}uv\tan\theta\right] + \frac{\delta^{-1}}{\cos\theta}p_{\phi^*} = O(\epsilon) \tag{20}$$

where the Coriolis parameter is defined as $f = \Omega\sin\theta$. Similarly, the Northerly (θ) momentum equation (3) becomes,

$$v_t + \frac{\epsilon}{\mu}f\,u + \delta\left[\frac{u}{\cos\theta}v_{\phi^*} + vv_{\theta^*} + wv_z + \frac{\epsilon}{\mu}u^2\tan\theta\right] + \delta^{-1}p_{\theta^*} = O(\epsilon) \tag{21}$$

The dimensionless z (local vertical) momentum equation is finally given by,

$$\delta\mu^2\left[w_t + \delta\left(\frac{u}{\cos\theta}w_{\phi^*} + vw_{\theta^*} + ww_z\right) - \frac{\epsilon}{\mu^2}fu\cot\theta\right] + (p_z + 1) = O(\epsilon) \tag{22}$$

In the following, we consider two relations between ϵ and μ: (1) the regime $\mu = O(\epsilon)$, which recovers the shallow water equations; and (2) the regime $\mu = O(\epsilon^{1/3})$, which yields the Boussinesq approximation.

2.2 Shallow Water Equations

Most theories of transoceanic tsunami propagation are based on shallow water equations (linear or NSWE), in recognition of the vanishing effects of dispersion ($\mu \to 0$) for very long waves. In the present discussion, this limit is obtained when the horizontal length scale of wave motion approaches the horizontal scale of a global-sized ocean basin, or $\lambda' \to r'_0$. This implies that the ratio $\epsilon/\mu = O(1)$, while terms proportional to μ appearing alone are essentially the size of already neglected terms of $O(\epsilon)$. In this combined limit, the local vertical momentum equation (22) reduces to an hydrostatic balance, which may be integrated down from the free surface to yield,

$$p_h = \delta\eta - z \tag{23}$$

where p_h denotes the hydrostatic component of pressure. This expression is used to evaluate pressure gradient terms in the horizontal momentum equations, yielding the final set of shallow water equations,

$$\frac{1}{\delta}H_t + \frac{1}{\cos\theta}\left[(H\overline{u})_\phi + (H\overline{v}\cos\theta)_\theta\right] = 0 \tag{24}$$

$$\overline{u}_t - f\overline{v} + \frac{1}{\cos\theta}\eta_\phi = 0 \tag{25}$$

$$\overline{v}_t + f\overline{u} + \eta_\theta = 0 \tag{26}$$

where in this limit the scaled lattitude and longitude revert to the original values. Note, equation (24) retains the possibility of describing wave generation through a bottom motion h_t, which appears at $O(1/\delta)$ since h' is scaled by h'_0 rather than wave amplitude.

2.3 The Boussinesq Approximation

We now wish to retain dispersive effects to leading order in the description of wave motion. Further, in order to provide a uniformly valid model, which can be used to describe nonlinear wave evolution in shallow coastal margins as well as mainly linear evolution in the deep ocean basin, we will retain the mechanics of the fully nonlinear Boussinesq model framework, following largely the approach of Wei et al. (1995), but working in the framework of rotational flow. We will, for now, retain depth-averaged horizontal velocities $(\overline{u}, \overline{v})$ as dependent variables rather than velocity at a reference elevation (as in Wei et al. and subsequent efforts). In developing the model, we seek to retain terms to $O(\mu^2)$ without any truncation in orders of δ. This is in contrast to the classical Boussinesq approach, which would take $\delta = O(\mu^2)$ and truncate terms of $O(\delta^2, \delta\mu^2, \mu^4)$ and higher.

In the derivation, we will retain the effect of an imposed bottom motion $h(t)$. The approximation is accompanied by the assumption that $\mu = O(\epsilon^{1/3})$. (For $\epsilon = O(10^{-3})$, this implies a dispersion term $\mu = O(10^{-1})$, which would be reasonable for the usual surface wave problems). This choice of scaling then implies that $O(\epsilon/\mu) = O(\mu^2)$, indicating that Coriolis terms and undifferentiated advective acceleration terms are the same size as the leading-order deviation from hydrostatic of the pressure term.

2.3.1 Pressure and vertical momentum

Pressure in the system being considered will deviate from hydrostatic by $O(\mu^2)$ amounts. Denoting this non-hydrostatic component by $\tilde{p}$, we write,

$$\begin{aligned} p(\phi^*, \theta^*, z, t) &= p_h(\phi^*, \theta^*, z, t) + \delta\mu^2\tilde{p}(\phi^*, \theta^*, z, t) \\ &= \delta\eta - z + \delta\mu^2\tilde{p} \end{aligned} \tag{27}$$

Introducing (27) in (22) and integrating up to the free surface (where $\tilde{p} = 0$) gives,

$$\tilde{p}(z) = \int_z^{\delta\eta} w_t dz + \delta\int_z^{\delta\eta}\left[\frac{u}{\cos\theta}w_{\phi^*} + vw_{\theta^*} + ww_z\right]dz + O(\epsilon) \tag{28}$$

The weakly nonlinear approximation with $\delta/\mu^2 = O(1)$ would retain,

$$\tilde{p}(z) = \int_z^0 w_t dz + O(\delta) \tag{29}$$

2.3.2 The vertical structure of velocities

In order to use (28) to evaluate horizontal pressure gradients, we need to establish a relation between w and components (u, v) through the continuity equation (15), which simplifies to,

$$w_z + \frac{\mu}{\cos\theta}\left[u_{\phi^*} + (v\cos\theta)_{\theta^*}\right] = O(\epsilon) \tag{30}$$

Integrating (30) from $-h$ to z and using the bottom boundary condition gives,

$$\begin{aligned} w(z) &= -\frac{1}{\cos\theta}\frac{\partial}{\partial\phi^*}\left(\int_{-h}^{z} u_{\phi^*} dz\right) - \frac{1}{\cos\theta}\frac{\partial}{\partial\theta^*}\left(\int_{-h}^{z} v\cos\theta dz\right) - \frac{1}{\delta}h_t \\ &= -\frac{1}{\cos\theta}\frac{\partial}{\partial\phi^*}\left(\overline{u}(h+z)\right) - \frac{1}{\cos\theta}\frac{\partial}{\partial\theta^*}\left(\overline{v}\cos\theta(h+z)\right) - \frac{1}{\delta}h_t + O(\mu^2) \end{aligned} \tag{31}$$

2.3.3 Weakly Nonlinear Approximation

The standard Boussinesq approximation follows from the assumption that $\delta = O(\mu^2)$. To the required order, the perturbation to hydrostatic pressure is then given by,

$$\tilde{p}(\phi^*, \theta^*, z, t) = \frac{z}{\cos\theta}\left[(h\overline{u}_t)_{\phi^*} + (h\cos\theta\overline{v}_t)_{\theta^*}\right] + \frac{z^2}{2\cos\theta}\left[\overline{u}_{\phi^* t} + (\overline{v}\cos\theta)_{\theta^* t}\right] + BFT \tag{32}$$

where BFT denotes forcing terms resulting from motion of the ocean bottom,

$$BFT = -\frac{1}{\delta}(\delta\eta - z)h_{tt} + \frac{(z-\delta\eta)}{\cos\theta}\left[(\overline{u}h_t)_{\phi^*} + (\overline{v}\cos\theta h_t)_{\theta^*}\right] \tag{33}$$

Using these results, the horizontal momentum equations are integrated over depth and the perturbation pressure is eliminated using the previous results. The final dimensionless equations are given by,

$$\frac{1}{\delta}H_t + \frac{1}{\cos\theta}\left\{(H\overline{u})_{\phi^*} + (H\overline{v}\cos\theta)_{\theta^*}\right\} = 0 \tag{34}$$

$$\begin{aligned} &\overline{u}_t - \mu^2 f\overline{v} + \delta\left\{\frac{\overline{u}}{\cos\theta}\overline{u}_{\phi^*} + \overline{v}\overline{u}_{\theta^*}\right\} + \frac{1}{\cos\theta}\eta_{\phi^*} \\ &+\frac{\mu^2}{\cos^2\theta}\left\{\frac{h^2}{6}\left[\overline{u}_{\phi^*\phi^* t} + (\overline{v}\cos\theta)_{\phi^*\theta^* t}\right] - \frac{h}{2}\left[(h\overline{u}_t)_{\phi^*\phi^*} + (h\cos\theta\overline{v}_t)_{\phi^*\theta^*}\right]\right\} \\ &+\frac{\mu^2}{\cos\theta}(BFT)_{\phi^*} = O(\delta^2, \delta\mu^2, \mu^4) \end{aligned} \tag{35}$$

$$\begin{aligned} &\overline{v}_t + \mu^2 f\overline{u} + \delta\left\{\frac{\overline{u}}{\cos\theta}\overline{v}_{\phi^*} + \overline{v}\overline{v}_{\theta^*}\right\} + \eta_{\theta^*} \\ &+\mu^2\left\{\frac{h^2}{6}\left[\frac{1}{\cos\theta}\left\{\overline{u}_{\phi^* t} + (\overline{v}\cos\theta)_{\theta^* t}\right\}\right]_{\theta^*} - \frac{h}{2}\left[\frac{1}{\cos\theta}\left\{(h\overline{u}_t)_{\phi^*} + (h\cos\theta\overline{v}_t)_{\theta^*}\right\}\right]_{\theta^*}\right\} \\ &+\mu^2(BFT)_{\theta^*} = O(\delta^2, \delta\mu^2, \mu^4) \end{aligned} \tag{36}$$

3 Numerical Approach

3.1 Conservative form of governing equations

The dimensional forms of the model equations will be used subsequently, and are given by (with primes dropped for convenience)

$$H_t + \frac{1}{r_0 \cos\theta}\left\{(H\overline{u})_\phi + (H\overline{v}\cos\theta)_\theta\right\} = 0 \tag{37}$$

$$\begin{aligned}
&\overline{u}_t - f\overline{v} + \frac{1}{r_0\cos\theta}\overline{u}\overline{u}_\phi + \frac{1}{r_0}\overline{v}\overline{u}_\theta + \frac{g}{r_0\cos\theta}\eta_\phi \\
&+\frac{1}{r_0^2\cos^2\theta}\left\{\frac{h^2}{6}\left[\overline{u}_{\phi\theta t} + (\overline{v}\cos\theta)_{\phi\theta t}\right] - \frac{h}{2}\left[(h\overline{u}_t)_{\phi\phi} + (h\cos\theta\overline{v}_t)_{\phi\theta}\right]\right\} \\
&+\frac{1}{r_0\cos\theta}(BFT)_\phi = 0
\end{aligned} \tag{38}$$

$$\begin{aligned}
&\overline{v}_t + f\overline{u} + \frac{1}{r_0\cos\theta}\overline{u}\overline{v}_\phi + \frac{1}{r_0}\overline{v}\overline{v}_\theta + \frac{g}{r_0}\eta_\theta \\
&+\frac{1}{r_0^2}\left\{\frac{h^2}{6}\left[\frac{1}{\cos\theta}\{\overline{u}_{\phi t} + (\overline{v}\cos\theta)_{\theta t}\}\right]_\theta - \frac{h}{2}\left[\frac{1}{\cos\theta}\{(h\overline{u}_t)_\phi + (h\cos\theta\overline{v}_t)_\theta\}\right]_\theta\right\} \\
&+\frac{1}{r_0}(BFT)_\theta = 0
\end{aligned} \tag{39}$$

Recent progress in the development of Boussinesq-type wave models using a combined nite-volume and nite-difference TVD schemes have shown robust performance of the shock- capturing method in simulating breaking waves and coastal inundation (Tonelli and Petti, 2009, Roeber et al., 2010, Shiach and Mingham, 2009, Erduran et al., 2005, and others). In this study, we applied the MUSCL-TVD scheme in space and a high-order Runge-Kutta scheme in time with adaptive time stepping. In order to apply the combined finite-volume and finite-difference schemes, the governing equations (37) - (39) were re-arranged to a conservative form following Shi et al. (2011) for the fully non-linear Boussinesq equations in Cartesian coordinates. We define

$$\begin{cases} \xi_1 = & r_0\cos\theta(\phi - \phi_0) \\ \xi_2 = & r_0(\theta - \theta_0) \end{cases} \tag{40}$$

where (ϕ_0, θ_0) are the reference longitude and latitude, respectively. (ξ_1, ξ_2) represent coordinates in the longitude and latitude directions, respectively. The conservative form of (37) - (39) can be written as

$$\frac{\partial \mathbf{\Psi}}{\partial t} + \nabla\cdot\mathbf{\Theta}(\mathbf{\Psi}) = \mathbf{S} \tag{41}$$

where $\mathbf{\Psi}$ and $\mathbf{\Theta}(\mathbf{\Psi})$ are the vector of conserved variables and the flux vector function, respectively, and are given by

$$\mathbf{\Psi} = \begin{pmatrix} H \\ U \\ V \end{pmatrix}, \qquad \mathbf{\Theta} = \begin{pmatrix} P\mathbf{i} + Q\mathbf{j} \\ \left[\frac{P^2}{H} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{i} + \frac{PQ}{H}\mathbf{j} \\ \frac{PQ}{H}\mathbf{i} + \left[\frac{Q^2}{H} + \frac{1}{2}g(\eta^2 + 2\eta h)\right]\mathbf{j} \end{pmatrix}. \tag{42}$$

where $P = Hu$, $Q = Hv$, and

$$U = H(u + F) \tag{43}$$

$$V = H(v + G) \tag{44}$$

in which

$$F = \frac{h^2}{6}u_{\xi_1\xi_1} + \frac{h^2}{6}(v_{\xi_1\xi_2} - \frac{1}{r_0}\tan\theta v_{\xi_1}) - \frac{h}{2}(hu)_{\xi_1\xi_1} - \frac{h}{2}\left[(hv)_{\xi_1\xi_2} - \frac{1}{r_0}\tan\theta(hv)_{\xi_1}\right] \tag{45}$$

$$\begin{aligned} G = {} & \frac{h^2}{6}\left(u_{\xi_1\xi_2} + \frac{1}{r_0}\tan\theta u_{\xi_1}\right) + \frac{h^2}{6}\left(v_{\xi_2\xi_2} - \frac{1}{r_0}\tan\theta v_{\xi_2} - \frac{1}{r_0^2\cos^2\theta}v\right) \\ & -\frac{h}{2}\left[(hu)_{\xi_1\xi_2} + \frac{1}{r_0}\tan\theta(hu)_{\xi_1}\right] - \frac{h}{2}\left[(hv)_{\xi_2\xi_2} - \frac{1}{r_0}\tan\theta(hv)_{\xi_2} - \frac{1}{r_0^2\cos^2\theta}(hv)\right] \end{aligned} \tag{46}$$

S in (41) represent a source array given by

$$\mathbf{S} = \begin{pmatrix} \frac{1}{r_0}\tan\theta Hv \\ g\eta h_{\xi_1} + fHv + \tau_b^{\xi_1} + \psi_{\xi_1} \\ g\eta h_{\xi_2} - fHu + \tau_b^{\xi_2} + \psi_{\xi_2} \end{pmatrix}, \tag{47}$$

where

$$\psi_{\xi_1} = \eta_t F \tag{48}$$

$$\psi_{\xi_2} = \eta_t G \tag{49}$$

Note that the surface elevation gradient term was split into $\frac{1}{2}g(\eta^2+2\eta h)$ in (42) and $(g\eta h_{\xi 1}, g\eta h_{\xi_2})$ in (47) in order to use a well-ballanced numerical scheme (Shi et al., 2011).

Equation (41) is solved using the MUSCL-TVD scheme and the HLL approximate Riemann solver. A fourth-order TVD-MUSCL scheme in space (Yamamoto and Daiguji, 1993) and a third-order Strong Stability-Preserving (SSP) Runge-Kutta (Gottlieb, et al., 2001) in time were adopted. The elimination method is used in solving the system of tridiagonal matrix equation formed by (43) and (44). Model implementation also includes wave breaking and wetting-drying schemes for shallow water exactly following Shi et al. (2011). Interest readers may refer to Shi et al. for details.

3.2 Parallelization

In parallelizing the computational model, we use the domain decomposition technique to subdivide the problem into multiple regions and assign each subdomain to a separate processor core. Each subdomain region contains an overlapping area of ghost cells three rows deep, as dictated by the 4th order computational stencil for the leading order non-dispersive terms. The Message Passing Interface (MPI) with non-blocking communication is used to exchange the data in the overlapping

region between neighboring processors. Velocity components are obtained from Equation (43) and (44) by solving tridiagonal matrices using parallel pipelining tridiagonal solver described in Naik et al. (1993).

To investigate performance of the parallel program, numerical simulations of the idealized ocean case are tested with different number of processors (16, 24, 36, and 48 processors) of a linux cluster located at University of Delaware. The test case is set up in a numerical grid of 1800 $\times$ 1800. Figure 1 shows the model speedup versus number of processors. It can be seen that a linear speedup is obtained with a slight decrease with more processors due to the communication cost.

4 Tests of Dispersion and Coriolis Effects

The orders of the frequency dispersions terms and Coriolis terms in Equations (35) and (36) are $O(\mu^2)$ and $O(\epsilon/\mu)$, respectively. The relative importance of frequency dispersion and Coriolis force can be evaluated using μ^2 and ϵ/μ values in some specific cases. Figure 2 illustrates variations of μ^2 and ϵ/μ respect to μ or h_0/W, where W represents width of tsunami source, . Typically, for a source width of 100 km (2004 Indian Ocean tsunami), $\mu \approx 0.025$, the Coriolis effect is relatively more important than dispersion as shown in Figure 2. For a narrow source with a width of 25 km, $\mu \approx 0.1$, the dispersive effect is as important as the Coriolis effect, and gets relatively more important as the source width diminishes.

4.1 An Okada source in an idealized ocean

To perform model tests with the spherical Boussinesq model, we generate a model grid in spherical coordinates and specify a flat bottom bathymetry in the ocean basin. The computational domain is in the northern hemisphere and covers a region from $10°N$ to $50°N$ in the south-north direction and from $20°W$ to $20°E$ in the west-east direction. Water depth is $3000m$ over the whole domain. The grid resolution is 1.5 arc minutes. The tsunami source is based on the standard half-plane solution for an elastic dislocation with maximum slip (Okada, 1985). Okada's solution is implemented in "Tsunami open and progressive initial conditions system" (TOPICS) which provides the vertical co-seismic displacements as outputs. We specified a north-south oriented planar fault with centroid located at $30°N$ and $0°W$. As an example, Figure 3 shows the surface elevation calculated by TOPICS which has a width of about $1°$ and a length of about $12°$. The tsunami is transferred and linearly superimposed into the spherical Boussinesq model, as an initial free surface condition.

To examine the effects of wave dispersion and Coriolis corresponding to different source widths, we generated a series of sources with different widths and the same amplitude and source length. The source with the width about $1°$ ($W = 1°$, thereafter) is the widest source in the tests and other source widths are $1/2°, 1/4°$ and $1/8°$, respectively. The corresponding inverse source width

parameter μ at 30° N are $0.0312, 0.0625, 0.125$ and 0.25, respectively. The sources are used in the full model including both dispersion and Coriolis terms and two reduced models, one without dispersion terms (NSWE) and the other without Coriolis terms.

4.2 Source size and wave dispersion effect

Figure 4 demonstrates the evolution of tsunami wave in time. The tsunami wave is generated using an Okada source with a source width of $1°$(96 km at 30° N). The inverse source width parameter μ is 0.0312 which indicates a small dispersive effect according to Figure 2. Wave diffraction is significant at the edges of the wave front as shown in Figure 4. A case with a narrow source is shown in Figure 5. The source width in this case is $1/8°$(12 km at 30° N) and the corresponding inverse source width parameter μ is 0.25. According to Figure 2, the dispersive effect should be important and it is demonstrated in Figure 5. The wave dispersion effect can be recognized by a train of waves near the wave front. Wave diffraction is less obvious compared to the above wider source case.

As indicated by Figure 2, the importance of dispersive effect increases with a decrease of the source width. Figure 6 shows the comparison of wave surface between cases of $W = 1°, 1/2°, 1/4°$ and $1/8°$ at the time $t = 9000s$. It is shown that a narrower source generates waves with a longer dispersive tail and with a weak diffraction.

The same four cases were carried out using the reduced model without dispersion terms. No tailing waves were found in any case. Figure 7 demonstrates results from the narrowest source $W = 1/8°$. The propagating wave basically keeps a constant waveform, which is significantly different from the results of the full model (Figure 5). The difference between the fully model and the reduced model can be clearly seen in a 1-D plot in Figure 8, which shows the comparison of wave surface elevation along the west-east direction at 30° N at $t = 3600s$ and $7200s$. The comparison indicates that the wave dispersive effect is characterized by decreasing the amplitude of the front wave and increasing the amplitude of the wave train behind the wave front. The same comparison for the widest source $W = 1°$ is shown in Figure 9, suggesting that solutions from the full model and the reduced model are basically similar due to less dispersion for a wider source.

To quantify the dispersive effect versus source width, the relative difference, which is the difference of magnitude normalized by the wave height from the full model, was calculated at 30° N and $t = 9000s$. The relative difference varies from 0.075 ($1°$ source) to 0.43 ($1/8°$ source) as shown in Figure 10. Exponential-like decay of the difference can be found with respect to source widths. The trend is consistent with the theoretical analysis.

4.3 Source size and Coriolis effect

The Coriolis effect can be indicated by comparing model results with and without Coriolis force. Figure 11 shows the relative difference calculated in the same way as in the dispersion tests but between models with/without Coriolis force. It indicates that the Coriolis effect increase with an increase of source width. It seems, however, that the difference caused by the Coriolis effect is small in the short distance propagation ($t = 9000s$).

To examine the Coriolis effect on a long distance propagation, we conducted runs in a large domain covering a region from $-90°$ to $0°$ W in the west-east direction and from $0°$ to $50°$ N in the south-north direction. The Okada source is centered at $0°$ N and $-22.5°$ W with a width of about $1°$ and a length of about $48°$. The reason to use the $48°$ long source is to examine the Coriolis effect at the high latitude region and note that only half of the source is located in the computational domain. Figure 12 shows the comparison of surface elevation between the full model (solid line contours) and the reduced model at $t = 12hours$. With the wide source, both models show a significant wave diffraction after a long distance propagation. A large difference between the two contours can be found in the high latitude region. There is no typical monopole radiation pattern found in the result due to the usage of long and unidirectional source.

I really dislike the last case. Actually Lovholt et al have made a conclusion for global test. We got about the same as in Figure 11. Do you think we should remove the last case? The following is their description:

"We quantify the importance of Coriolis forces on the far-field propagation across the Atlantic Ocean, along the two transoceanic transects given in Figure 13. The ampli- tude deviation for the leading crest at 6 h 45 min, for standard Boussinesq models with and without Coriolis terms, give 2.5% along the northern transect, and 1.5% along the southern transect.

5 Conclusions

We derived and validated equations for weakly nonlinear, weakly dispersive Boussinesq model on the surface of a rotating sphere, including Coriolis effects. The numerical scheme is based on the Godunov-type method with a fourth-order MUSCL-TVD scheme in time and a third-order SSP Runge-Kutta scheme. The model is implemented using a domain decomposition technique and optimized for parallel computer clusters, by way of the message passing interface (MPI). Model speedup tests with multi-processors show a linear speedup, suggesting the Boussinesq model is applicable of modeling global wave propagation.

Scaling analysis revealed that the importance of frequency dispersion increases with a decrease of the source width, and Coriolis force increases with an increase of the source width. The theoretical analysis was quantified by the numerical model which was applied to an idealized ocean using

Okada sources with different source sizes.

ACKNOWLEDGMENTS This work was sponsored by the NTHMP project.

References

Erduran, K. S., Ilic, S., and Kutija, V., 2005, Hybrid finite-volume finite-difference scheme for the solution of Boussinesq equations, *Int. J. Numer. Meth. Fluid.*, 49, 1213-1232.

Glimsdal, S., G. Pedersen, K. Atakan, C. B. Harbitz, H. Langtangen, and F. Løvholt , 2006, "Propagation of the Dec. 26, 2004 Indian Ocean tsunami: effects of dispersion and source characteristics", *Int. J. Fluid Mech. Res.*, **33**(1), 15-43.

Gottlieb, S., Shu, G.-W., and Tadmor, Titan, 2001, Strong stability-preserving high-order time discretization methods, *SIAM Review*, 43 (1), pp. 89-112.

Grilli, S. T., Ioualalen, M., Asavanant, J., Shi, F., Kirby, J. T. and Watts, P., 2007, "Source constraints and model simulation of the December 26, 2004 Indian Ocean tsunami", *J. Waterway, Port, Coast. and Ocean Engrng.*, **133**, 414-428.

Grue, J., Pelinovsky, E. N., Fructus, D., Talipova, T. and Kharif, C., 2008, "Formation of undular bores and solitary waves in the Strait of Malacca caused by the 26 December 2004 Indian Ocean tsunami", *J. Geophys. Res.*, **113**, C05008, doi:10.1029/2007JC004343.

Imamura, F., N. Shuto, and C. Goto, 1988, "Numerical simulation of the transoceanic propagation of tsunamis", presented at *Sixth Congress of the Asian and Pacific Regional Division*, Int. Assoc. Hydraul. Res., Kyoto, Japan.

Ioualalen, M., Asavanant, J., Kaewbanjak, N., Grilli, S.T., Kirby, J.T. and P. Watts, 2007, " Modeling the 26th December 2004 Indian Ocean tsunami: Case study of impact in Thailand," *J. Geophysical Res.*, **112**, C07024, doi:10.1029/2006JC003850.

Kulikov, E., 2005, "Dispersion of the Sumatra tsunami waves in the Indian Ocean detected by satellite altimetry." Rep. from P. P. Shirshov Institute of Oceanology, Russian Academy of Sciences, Moscow.

Lovholt, F., Pedersen, G. and Gisler, G., 2008, "Oceanic propagation of a potential tsunami from the La Palma Island", *J. Geophys. Res.*, **113**, C09026, doi:10.1029/2007JC004603.

Naik, N. H., Naik, V. K., and Nicoules, M., 1993, Parallelization of a class of implicit finite difference schemes in computational fluid dynamics, *International Journal of High Speed Computing*, 5: 1-50.

Okada, Y., 1985, Surface deformation due to shear and tensile faults in a half-space, *Bull. Seismol. Soc. Am.*, **75**(4), 1135-1154.

Pedlosky, J., 1979, *Geophysical Fluid Dynamics*, Springer-Verlag, New York, pp.624.

Pophet, N., 2008, "Parallel computation for tsunami", M.S. Thesis, Chulalongkorn University

Pophet, N., Kaewbanjak, N., Asavanant, J. and Ioualalen, M., 2008, "Parallelization of fully non-linear Boussinesq equations for tsunami simulations: new approach on higher grid resolution for tsunami simulation using parallelized fully nonlinear Boussinesq Equations", submitted to *Computer and Fluids*.

Roeber, V., Cheung, K. F., and Kobayashi, M. H., 2010, Shock-capturing Boussinesq-type model for nearshore wave processes, *Coastal Engineering*, 57, 407-423.

Shi, F., Dalrymple, R. A., Kirby, J. T., Chen, Q. and Kennedy, A., 2001, "A fully nonlinear Boussinesq model in generalized curvilinear coordinates", *Coastal Engrng.*, **42**, 337-358.

Shi, F., Kirby, J. T., Tehranirad, B. and Harris, J. C., 2011, FUNWAVE-TVD, users' manual and benchmark tests, Center for Applied Coastal Research Report, CACR 2011-xx, University of Delaware, Newark, Delaware.

Shiach, J. B. and Mingham, C. G., 2009, A temporally second-order accurate Godunov-type scheme for solving the extended Boussinesq equations, *Coastal Engineering*, 56, 32-45.

Sitanggang, K. and Lynett, P., 2005, "Parallel computation of a highly nonlinear Boussinesq equation model through domain decomposition", *Int. J. Numerical Methods in Fluids*, **49**, 57-74.

Tappin, D.R., Watts, P., Grilli, S.T., 2008, "The Papua New Guinea tsunami of 1998: anatomy of a catastrophic event", *Natural Hazards and Earth System Sciences*, **8**, 243-266.

Tonelli, M. and Petti, M., 2009, "Hybrid finite volume - finite difference scheme for 2DH improved Boussinesq equations", *Coast. Engrng.*, **56**, 609-620.

Wei, G., Kirby, J. T., Grilli, S. T. and Subramanya, R., 1995, "A fully nonlinear Boussinesq model for surface waves. I. Highly nonlinear, unsteady waves", *J. Fluid Mech.*, **294**, 71-92.

Yamamoto, S., Daiguji, H., 1993, Higher-order-accurate upwind schemes for solving the compressible Euler and NavierStokes equations, *Computers and Fluids*, 22 (2/3), 259-270.

Yoon, S. B., 2002, "Propagation of distant tsunamis over slowly varying topography", *J. Geophys. Res.*, **107(C10)**, 3140, doi:10.1029/2001JC000791.

Kirby, J. T., G. Wei, Q. Chen, A. B. Kennedy, and R. A. Dalrymple, 1998, "Fully nonlinear Boussinesq wave model documentation and user's manual", Research Rep. CACR-98-06, Center for Appl. Coastal Res., Univ. of Delaware, Newark.

Kulikov, 2005,

Shuto, N., 1985, "The Nihonkai-Chubu earthquake tsunami on the north Akita Coast", *Coastal Eng. Jpn.*, **28**, 255-264.

Shuto,N.,Goto,C.,and Imamura,F.,1990, Numerical simulation as a means of warning for near field tsunamis: Coastal Engineering of Japan, v.33, p.173-193.

Tappin, D. R., Watts, P. and Grilli, S. T., 2008, "The Papua New Guinea tsunami of 17 July 1998: anatomy of a catastrophic event", *Nat. Hazards Earth Syst. Sci.*, **8**, 243-266.

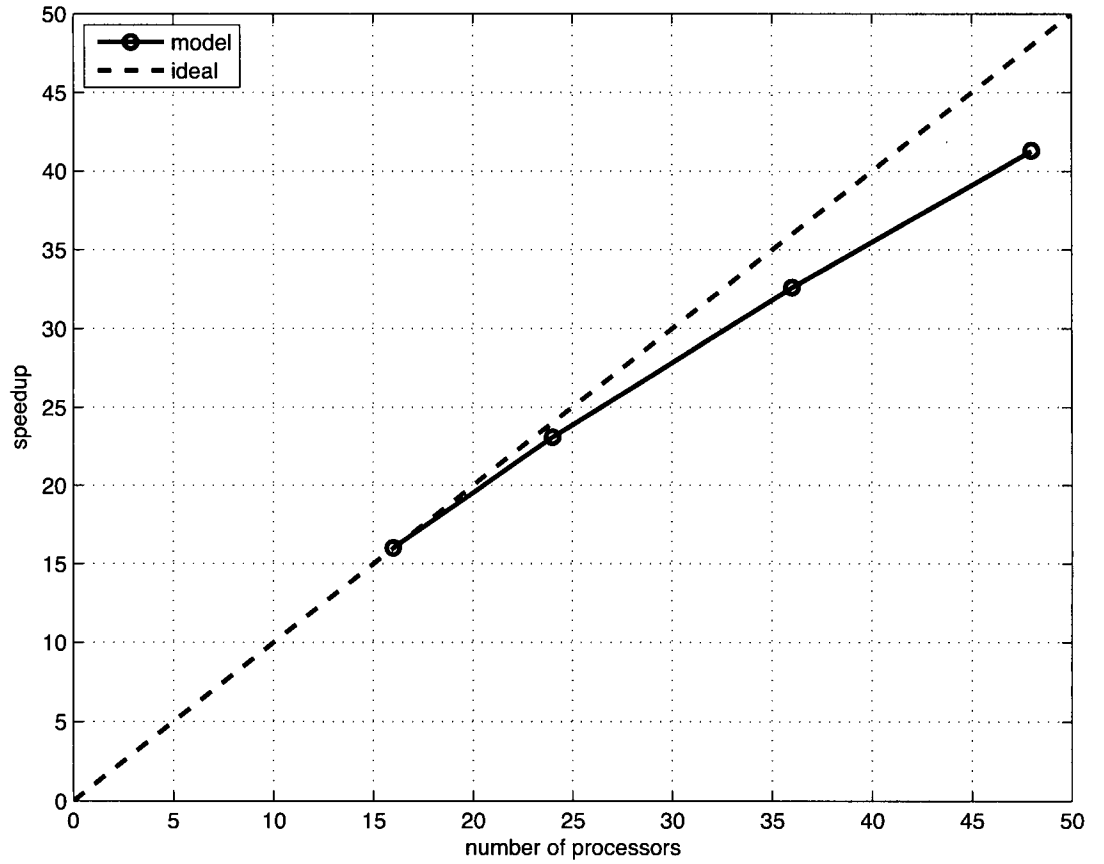

Figure 1: Variation in model performance with number of processors for a 5401 $\times$ 3601 domain. Straight line indicates arithmetic speedup. Actual performance shown by green line.

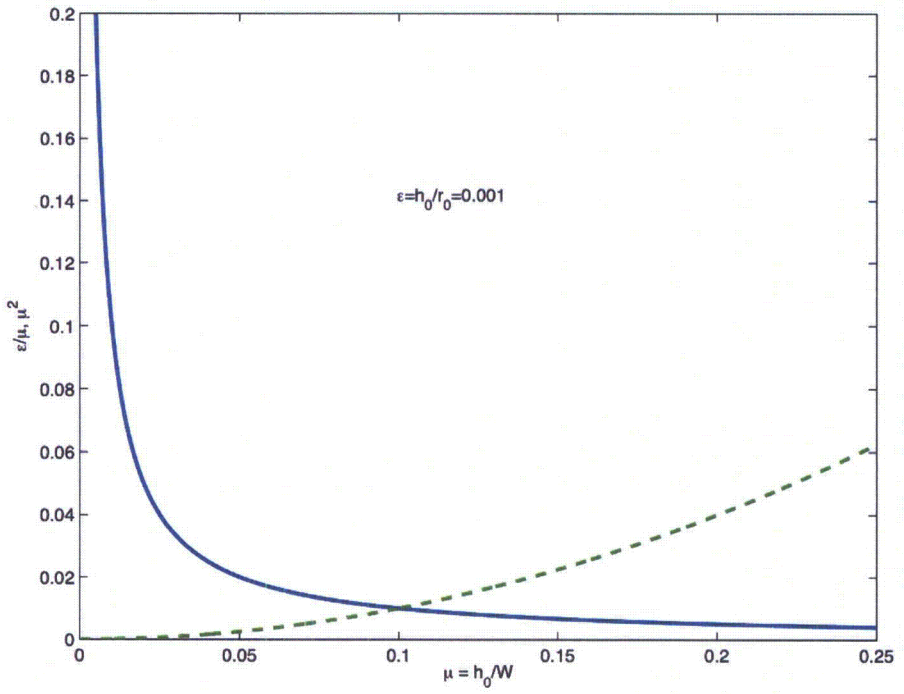

Figure 2: Relative importance of Coriolis force (ϵ/μ) and frequency dispersion (μ^2) with varying inverse source width $\mu = h_0/W$.

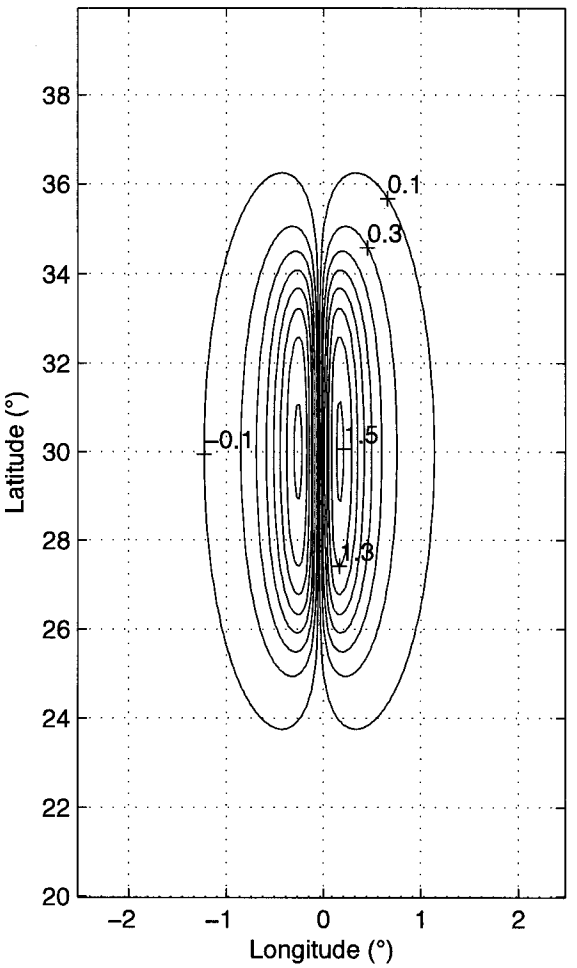

Figure 3: An Okada source with a width of about 1° and length of about 12°

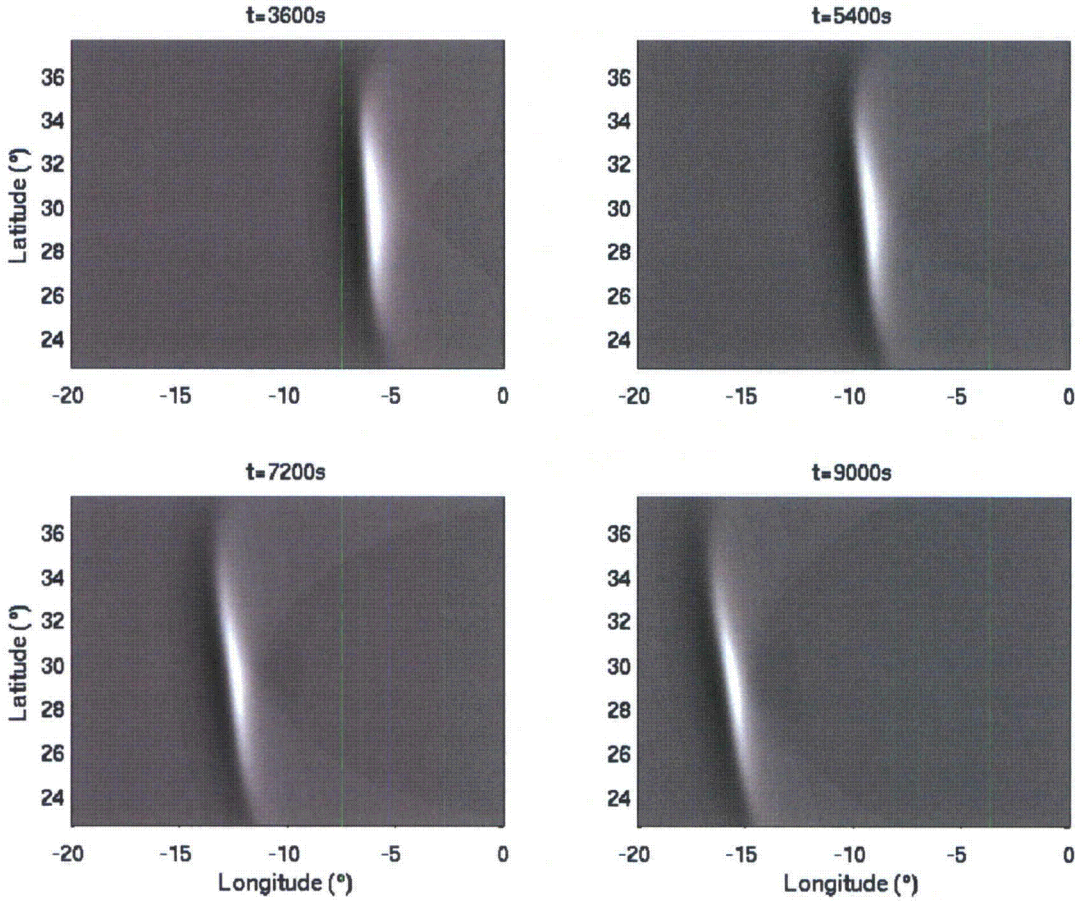

Figure 4: Wave surface calculated from the source $W = 1°$ at t = 3600, 5400, 7200, and 9000 s, respectively

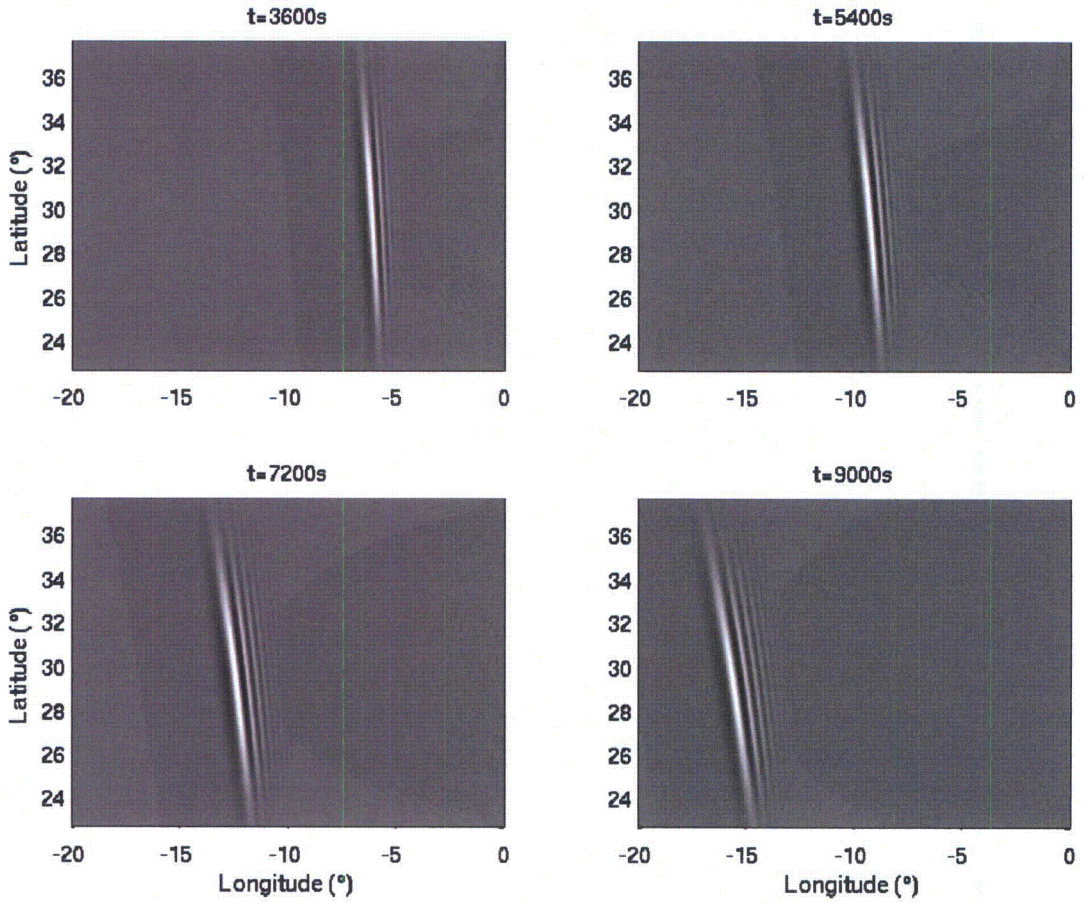

Figure 5: Wave surface calculated from the source $W = 1/8°$ at t = 3600, 5400, 7200, and 9000 s, respectively

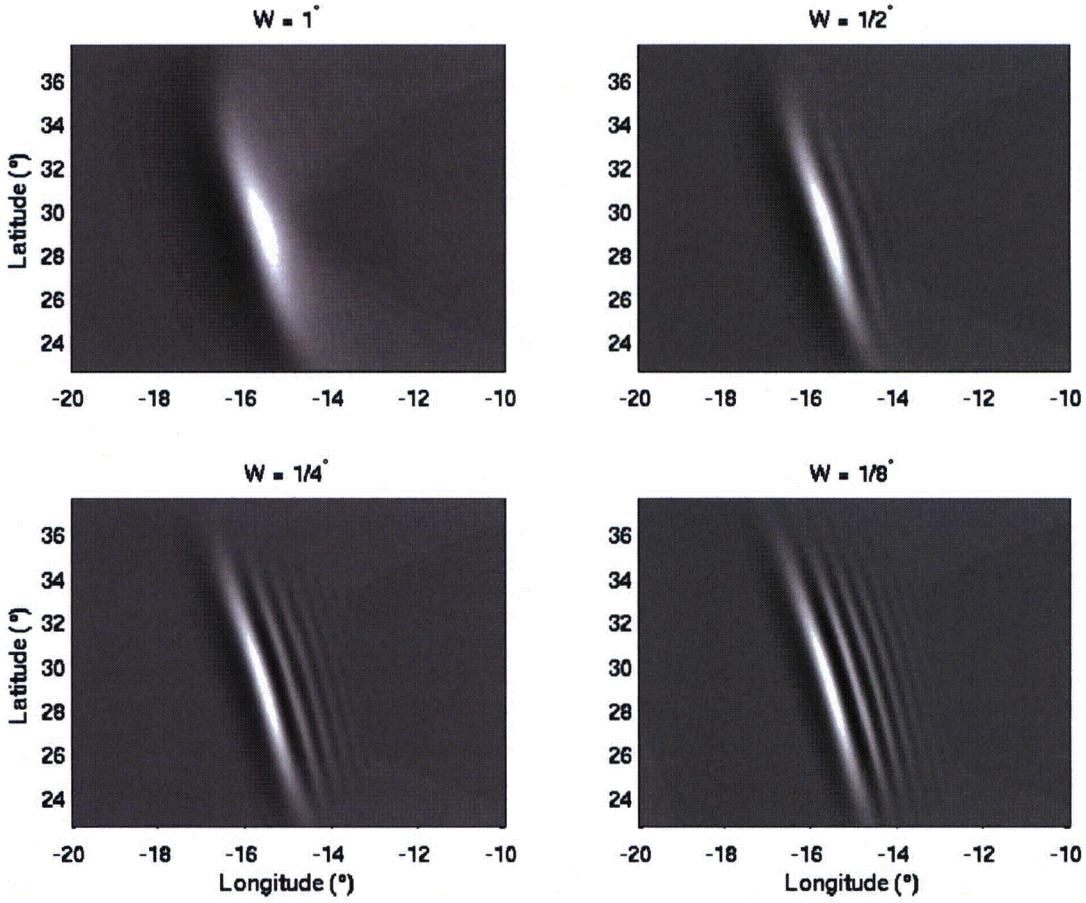

Figure 6: Wave surfaces calculated from sources $W = 1°, 1/2°, 1/4°$ and $1/8°$ at t = 9000 s

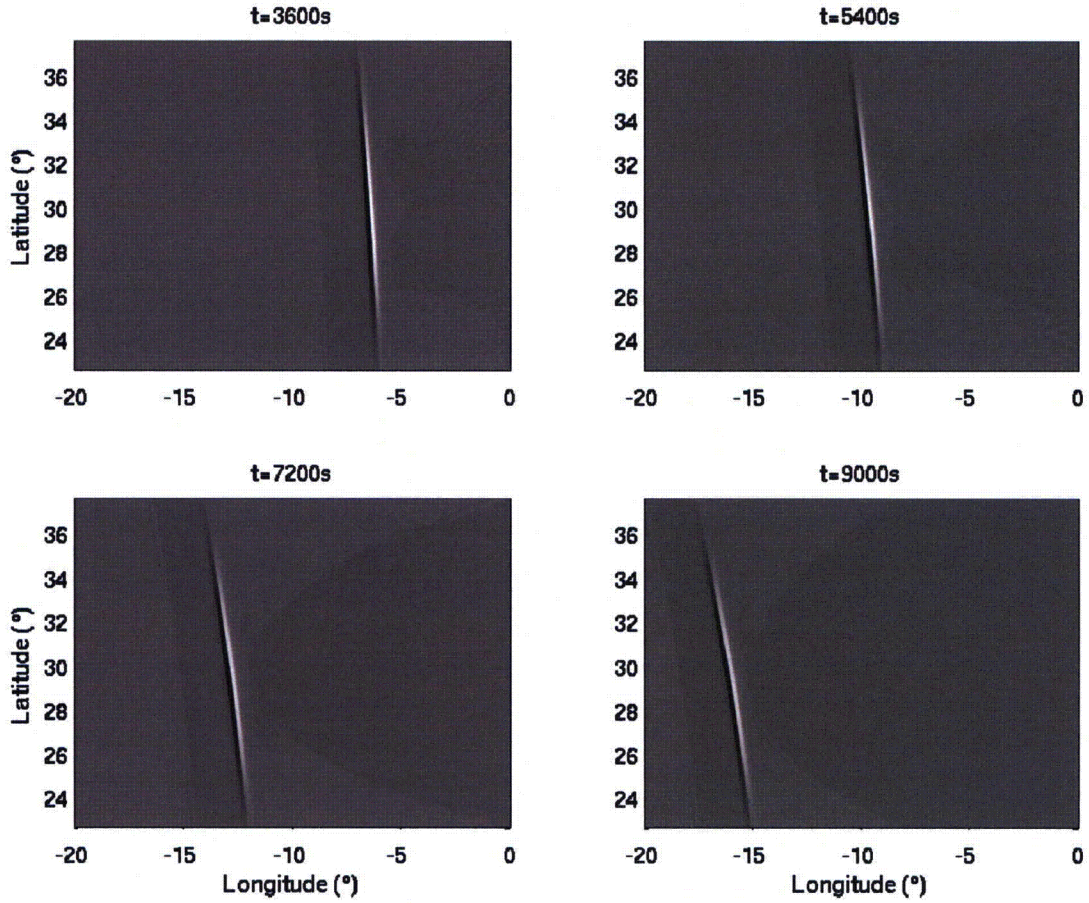

Figure 7: Wave surface calculated from the NSWE model with the source $W = 1/8°$ at t = 3600, 5400, 7200, and 9000 s, respectively

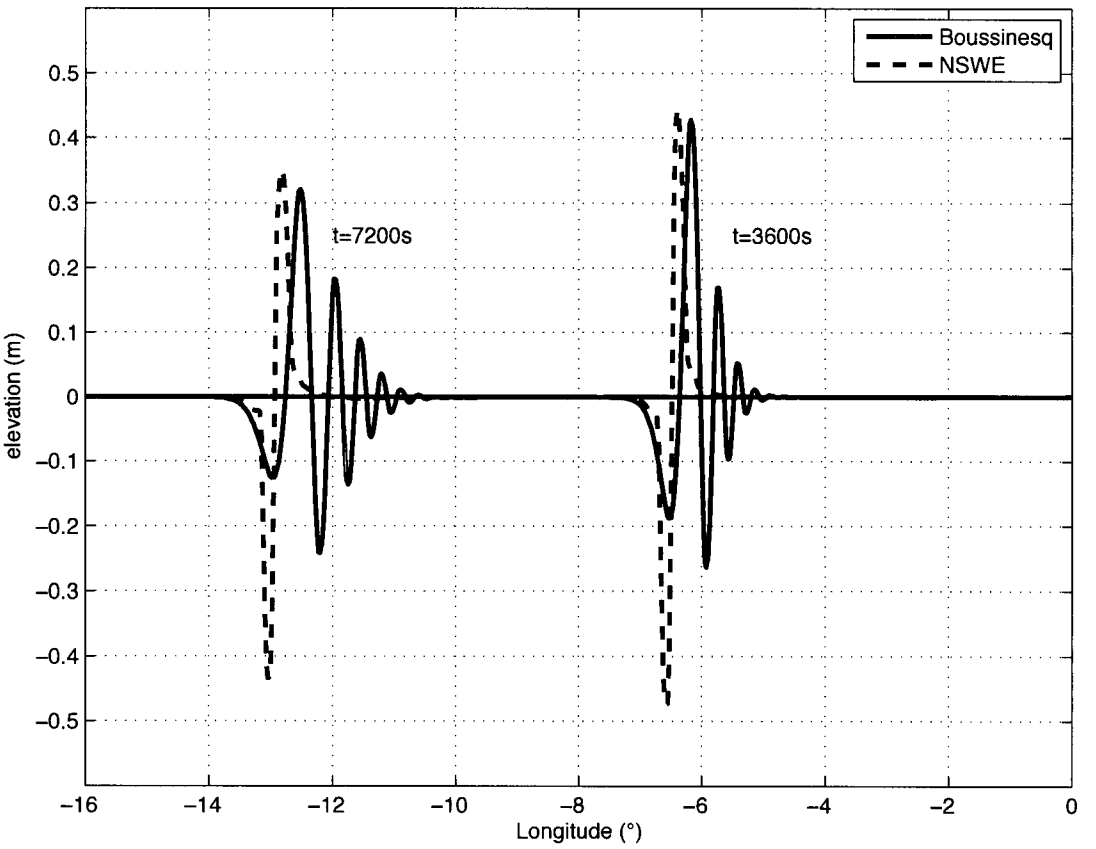

Figure 8: Comparison of wave surface elevation along Latitude= 30° between the Boussinesq model and NSWE model with sources $W = 1/8°$ at t = 3600s and 7200s

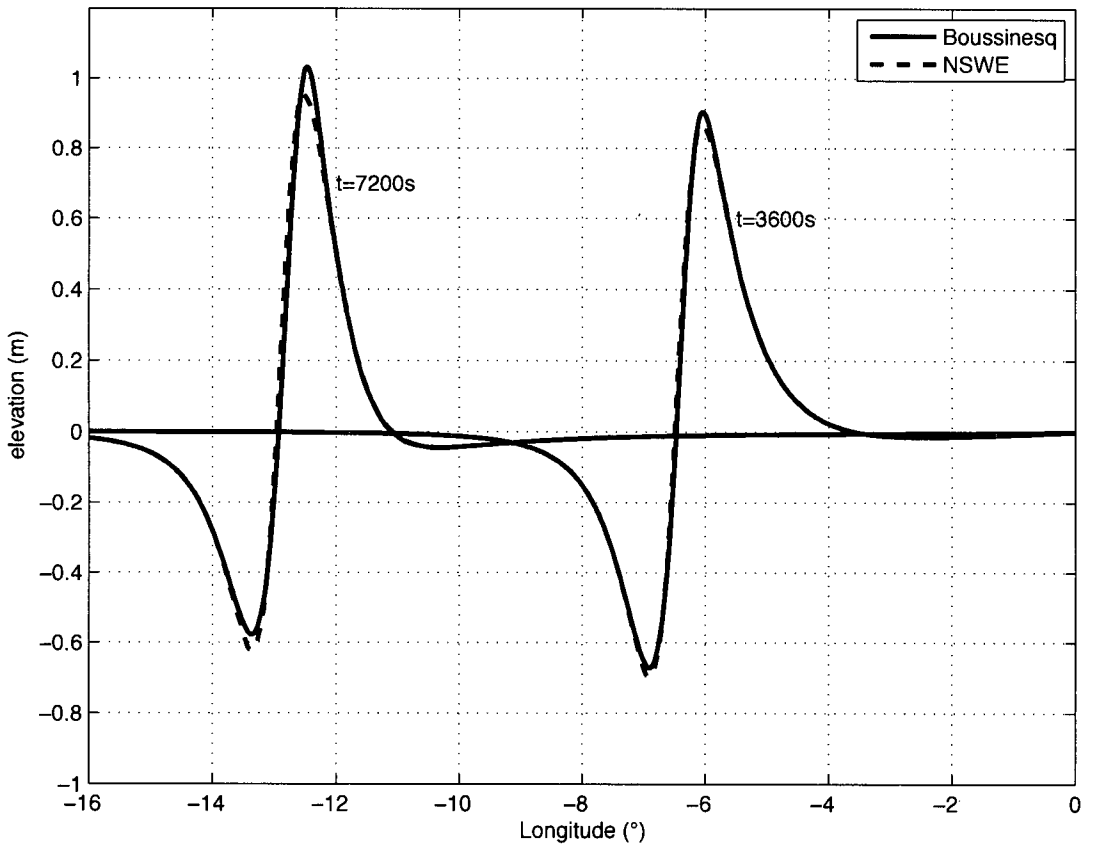

Figure 9: Comparison of wave surface elevation along Latitude= 30° between the Boussinesq model and NSWE model with sources $W = 1°$ at t = 3600s and 7200s

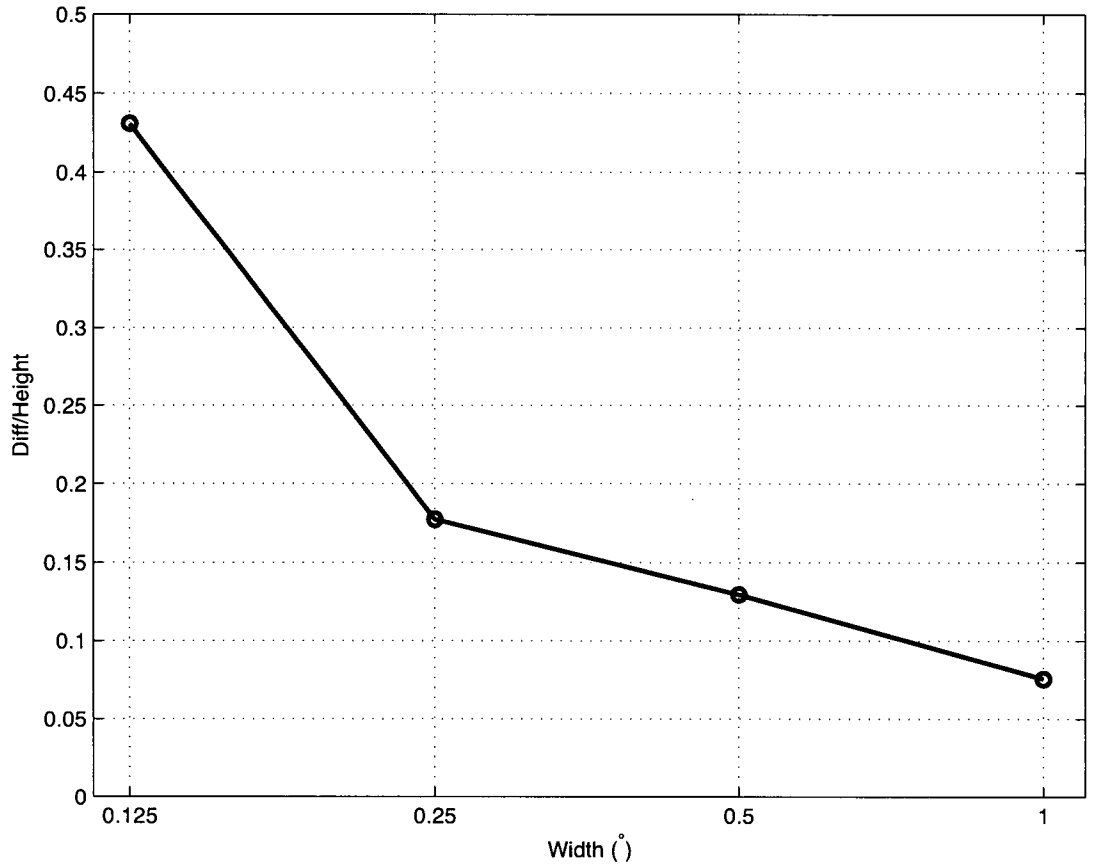

Figure 10: Relative difference of surface elevation between the Boussinesq model and NSWE model versus source width.

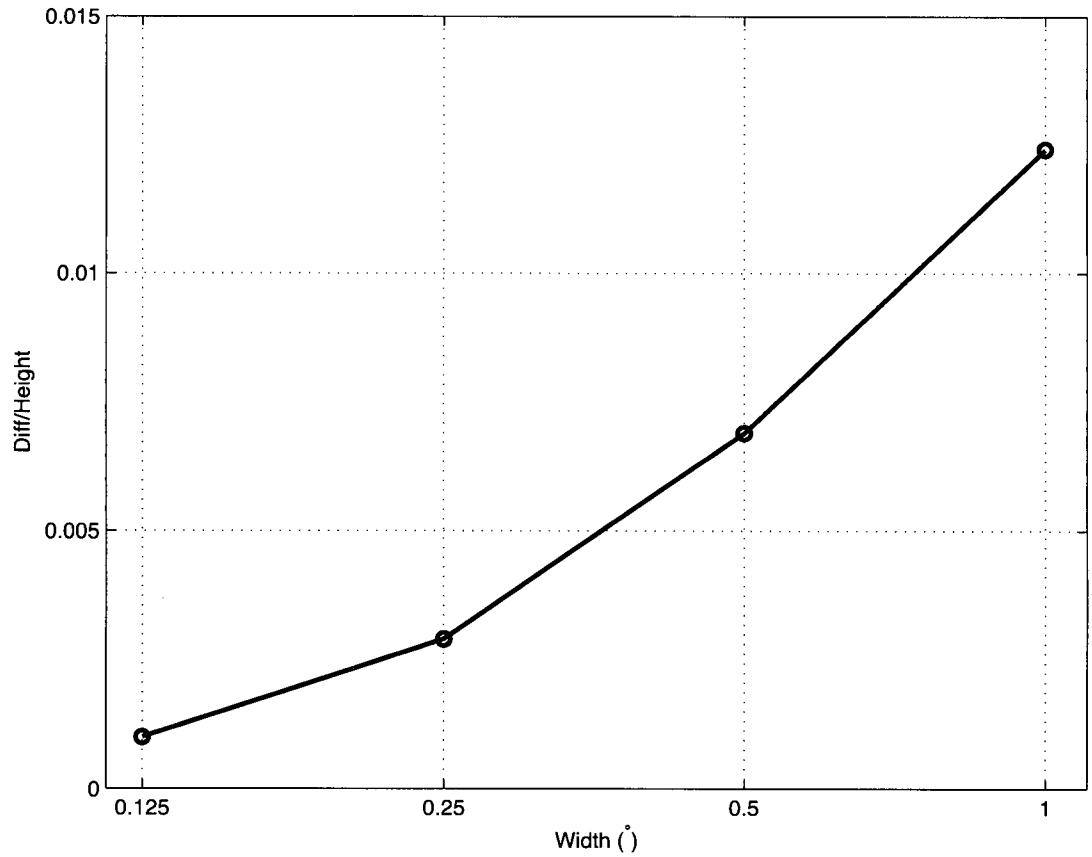

Figure 11: Maximum difference of surface elevation between the Boussinesq model with/without the Coriolis term versus source width.

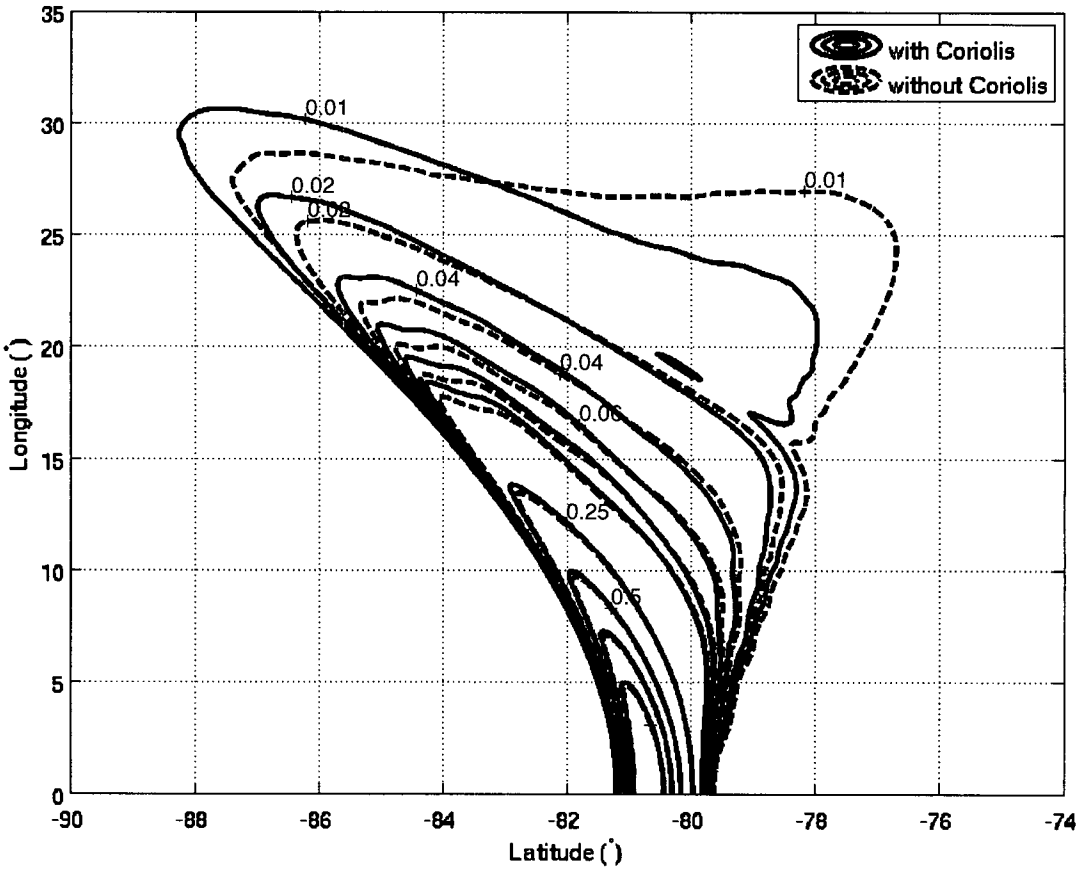

Figure 12: Comparison of surface elevation between models with/without the Coriolis term at t = 12 hours.