Two New Approaches in Solving the Nonlinear Shallow Water Equations for Tsunamis

Chaofan Zhang [a] Matthew G. Knepley [b] David A. Yuen [c]
Yaolin Shi [a]

[a] *Laboratory of Computational Geodynamics, Graduate University of Chinese Academy of Sciences, Beijing 100049, China*

[b] *Mathematics and Computer Science Division, Argonne National Labratory, Chicago, IL 60439, USA*

[c] *Department of Geology & Geophysics and Minnesota Supercomputing Institute, University of Minnesota, MN 55455, USA*

Abstract

One key component of tsunami research is numerical simulation of tsunamis, which helps us to better understand the fundamental physics and phenomena and leads to better mitigation decisions. However, writing the simulation program itself imposes a large burden on the user. In this survey, we review some of the basic ideas behind the numerical simulation of tsunamis, and introduce two new approaches to construct the simulation using powerful, general-purpose software kits, PETSc and FEPG. PETSc and FEPG support various discretization methods such as finite-difference, finite-element and finite-volume, and provide a stable solution to the numerical problem. Our application uses the nonlinear shallow-water equations in Cartesian coordinates as the governing equations of tsunami wave propagation.

Key words:
Tsunami, Numerical Simulation, Nonlinear Shallow Water Equations, PETSc, FEPG
PACS: *92.10.hl, 07.05.Tp, 02.60.Lj, 91.30.Nw

Email addresses: zhangchf@gmail.com (Chaofan Zhang), knepley@mcs.anl.gov (Matthew G. Knepley), daveyuen@gmail.com (David A. Yuen), shiyl@gucas.ac.cn (Yaolin Shi).

1 Introduction

Earthquakes, combined with resulting tsunami, can cause devastation, as we saw during the Mw 9.1 Sumatra earthquake (Dec. 26, 2004) and the Mw8.1 Solomon Islands earthquakes (April 1, 2007). It is imperative that we understand this nonlinear phenomenon in order to predict likely damage and minimize expected casualties.

The numerical problem of tsunami wave propagation is usually solved by linear or nonlinear shallow water equations. However, writing the simulation program itself can be time consuming and error prone, especially with the added complication of parallelism.

Rapid progess in computer science and computational mathematics make it possible to carry out larger scale and more intricate numerical simulations. Some general-use software kits and numerical libraries provide stable resolutions to the numerical problems. They have flexible extensions and easy-to-use graphical interfaces, which can greatly simplify the application research in this field.

We introduce here two new approaches for the numerical simulation of a tsunami using powerful software platforms, PETSc and FEPG, applied to the nonlinear shallow water equations. PETSc is free software for scientific computing developed and maintained by Argonne National Laboratory. It contains a large suite of linear and nonlinear algebraic equation solvers, and uses high performance computing techniques such as MPI and multigrid. Various discretization methods such as finite difference, finite element and finite volume can be expressed, and it is inherently parallel. FEPG is authorized for distribution by a Chinese company Fegensoft. It has a serial version, a parallel version and a free web-client version. It can help to generate the numerical simulation program with the user-provided files describing the PDEs, finite element formulations, and some other informations.

2 Fundamentical Physics and Mathematical Development behind Numerical Tsunami Simulation

A tsunami excited by an earthquake is basically an initial boundary value problem. As we can see from figure 1, in a Cartesian coordinate system, we select a domain of length L, width W, and depth D. The wave motion inside the domain can be described by the Navier-Stokes equations. The side surface boundary can be set as a wall (total reflection), open ocean (total absorption), or coastal shore along which water can run up or down. The lower surface

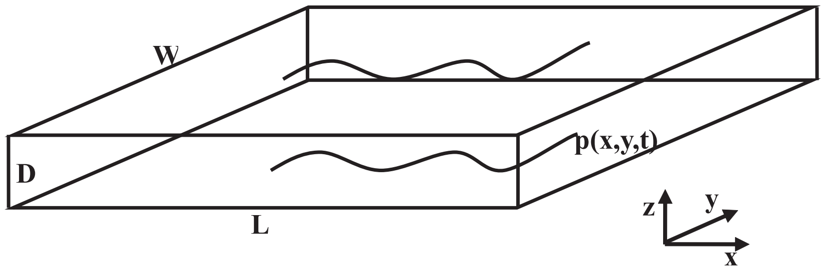

Fig. 1. Description of the calculation domain. Seafloor deformation excited by earthquakes induce the tsunami waves that we see on the sea-surface.

boundary is specified by the bathymetry of water depth, so $D(x, y)$ varies over the plane. The upper surface is a free surface boundary between the water and air. Given the continuous boundary perturbation $p(x, y, t)$ on the lower surface, we can obtain the solution of the motion on the upper free water surface.

However, it remains essentially an unresolved numerical problem to determine and solve the full well-posed boundary value problem (Kirby et al., 1998). The usual method employs the Boussinesq equations (Kirby, 1996; Witting, 1984; Nwogu, 1993), adding terms for wave generation, reflection, absorption, and friction, etc, and specifys the required boundary conditions.

The fully nonlinear Boussinesq equations derived by Wei et al. (1995) can be given as Equation (1),

$$
\begin{aligned}
&\eta_t + \nabla \cdot \Big\{(h+\eta)[\mathbf{u}_\alpha + (z_\alpha + \tfrac{1}{2}(h-\eta))\nabla(\nabla \cdot (h\mathbf{u}_\alpha))] \\
&\quad +(\tfrac{1}{2}z_\alpha^2 - \tfrac{1}{6}(h^2 - h\eta + \eta^2))\nabla(\nabla \cdot \mathbf{u}_\alpha)\Big\} = 0 \\
&\mathbf{u}_{\alpha t} + (\mathbf{u}_\alpha \cdot \nabla)\mathbf{u}_\alpha + g\nabla\eta + z_\alpha \left\{\tfrac{1}{2}z_\alpha \nabla(\nabla \cdot \mathbf{u}_{\alpha t}) + \nabla(\nabla \cdot (h\mathbf{u}_{\alpha t}))\right\} \\
&\quad +\nabla \left\{\tfrac{1}{2}(z_\alpha^2 - \eta^2)(\mathbf{u}_\alpha \cdot \nabla)(\nabla \cdot \mathbf{u}_\alpha) + \tfrac{1}{2}\left[\nabla \cdot (h\mathbf{u}_\alpha) + \eta\nabla \cdot \mathbf{u}_\alpha\right]^2\right\} \\
&\quad +\nabla \left\{(z_\alpha - \eta)(\mathbf{u}_\alpha \cdot \nabla)(\nabla \cdot (h\mathbf{u}_\alpha)) - \eta\left[\tfrac{1}{2}\nabla \cdot \mathbf{u}_{\alpha t} + \nabla \cdot (h\mathbf{u}_{\alpha t})\right]^2\right\} = 0
\end{aligned} \tag{1}
$$

where η is the wave height related to the still water, h is the still water depth, $\mathbf{u}_\alpha$ is the velocity vector in x- and $y-$ direction at water depth $z = z_\alpha$, g is the gravitational acceleration and subscript t means partial derivative with respect to time. This is a general approach to reduce the problem to two dimensions, but we may also use a depth averaged velocity.

If we make three further simplifications (Pedlosky, 1987; Haidvogel and Beckman, 1999; Kirby, 1997):

(1) the fundamental scaling condition for shallow water, $\delta = D/L \ll 1$, is satisfied,

(2) static fluid pressure, $0 \approx -(1/\rho)(\partial p/\partial z) - g$, meaning gravity is balanced with the vertical water pressure gradient,
(3) incompressibility, $\nabla \cdot \mathbf{v} = 0$,

we arrive at the shallow water equations expressed by Pedlosky (1987) in Equation (2). For large-scale ocean wide tsunami simulation, the Coriolis term (Haidvogel and Beckman, 1999) must be included to account for the spherical inertia effect of Earth's rotation. Equation (2) can also be obtained by integrating the Navier-Stokes equations in z- direction under these three assumptions.

$$\begin{aligned} \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + g\frac{\partial h}{\partial x} = 0 \\ \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + g\frac{\partial h}{\partial y} = 0 \\ \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left[(h-h_B)\,u\right] + \frac{\partial}{\partial y}\left[(h-h_B)\,v\right] = 0 \end{aligned} \tag{2}$$

Here u and v are the horizontal velocities in x- and y- direction respectively, h the wave height, g the gravitational acceleration, and h_B the ocean depth as a function of (x, y). We have neglected effects from the ocean bottom friction.

Many researchers have used the shallow water equations to construct numerical models, such as TUNAMI (Goto et al., 1997), MOST (Titov and Gonzalez, 1997) and a finite element method proposed by Hanert et al. (2005). Other models, like FUNWAVE (Kirby et al., 1998) and GeoWave (Watts et al., 2003), use the Boussinesq equations directly. Some perform quite well and after tuning the model parameters, the results are comparable with experimental and field data (Titov and Gonzalez, 1997). However, most of these models are built manually by the researchers, which is a lengthy process. They are usually limited to finite difference discretization methods and some easy-to-implement solution methods. Finite difference methods are easy to implement, as well as solve, and have reasonable memory requirements. Finite element and finite volume methods are more suited to the irregular domain, are capable of higher accuracy, and the boundary conditions can be more naturally formulated. However, they demand more computational power and often run more slowly. In sections 3 and 4 we will outline our applications, based on the nonlinear shallow water equations, with the general-purpose software PETSc and FEPG. We compare their results and the results from TUNAMI model in section 5.

3 Numerical Tsunami Simulation with PETSc

3.1 General Philosophy in PETSc

PETSc, the Portable, Extensible Toolkit for Scientific Computing, is a suite of libraries providing data structures and routines which are the building blocks for implementation of large-scale serial and parallel application codes. PETSc is inherently parallel. MPI is used for all parallel communication (Balay et al., 2006), but its use is hidden under the abstract linear algebra interface so that a user never need to explicitly call MPI methods.

The basic libraries in PETSc include vectors, matrices, distributed arrays, Krylov subspace methods, preconditioners, nonlinear solvers, and timestepping routines, etc. PETSc constructs these modules with the object-oriented programming method. The core of the library is written in C, however interfaces to Fortran 77/90, C++, and Python are also provided. The libraries are organized by functionality and usually have a single main class, such as PC for preconditioners, KSP for Krylov subspace linear solvers, SNES for nonlinear equations solvers, etc. High-level support for multilevel algorithms is also provided by the DMMG module, which is essential in solving large-scale problems, saving a large amount of time and storage while maintaining the required accuracy. Each module has a well-defined API which separates the application code from the specific implementation. In fact, all objects are instantiated dynamically so that implementation types can be specified at runtime on the command line. For instance, if we would like to run the stabilized Bi-Conjugate Gradients solver on a matrix type optimized for vector machines, we need only give the options: -ksp_type bcgs -mat_type csrperm.

PETSc provides the user an abundance of linear solvers and preconditioners that may be suitable for different problems, such as the solver GMRES, CG, QMR, and preconditioners Jacobi, ILU, additive Schwarz, and algebraic and geometric multilgrid. For nonlinear problems, Newton-like methods with line search and trust region techniques are used. Moreover, all individual solvers and preconditioners may be easily composed, either on a subdomain level as in Schwarz-type methods, in hierarchical arrangements with two-level Schwarz or multigrid, or with a Galerkin-type composition. In addition, these compositions may be accomplished with only command line arguments as well.

The ability to easily customize a PETSc simulation from the command line has saved a great amount of time and made the application code much simpler. The user can readily assemble a small script to run with a wide range of different solvers and preconditioners. Using the built-in profiling mechanism, comparing different methods and selecting the best is almost automatic. Debugging the program also becomes easier, as a direct solver can first be used to check results, or a finite difference approximant to the Jacobian to check hand-coded derivatives. There is also integrated support for various debuggers, as well as checks for memory corruption.

3.2 Application on Tsunami Wave propagation Simulation

We used a finite difference discretization method to handle the nonlinear shallow water equations (Equation (2)) with PETSc. By changing the discretization part, the finite element or finite volume method can also be integrated into it. Forward differences are used in the time domain and central differences in the spatial domain, as shown in Equation (3). An implicit scheme is used for the timestepping, and Newton's method for the nonlinear algebraic system. Restarted GMRES with a block Jacobi preconditioner is used for the linear iterative solution.

$$\frac{du}{dt} = \frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{dt}, \frac{du}{dx} = \frac{u_{i+1,j}^{n+1} - u_{i-1,j}^{n+1}}{2dx} \tag{3}$$

We use the elastic deformation during an earthquake as the tsunami wave initial condition, which is common in tsunami community. There are many different approaches to get the surface deformation. Most of them based on dislocation theory and linearly superpose the deformation due to point sources (Okada, 1985, 1992; Wang et al., 2006). Okada's program is used here. Because the earthquake occurs in a very short period, the motion of the ocean floor and the motion of the water can be decoupled (Shuto, 1991). Some typical results from this PETSc model are shown in figure 2. The initial condition of the coseismic deformation caused by an assumed earthquake is shown in sub-figure (a). The tsunami wave simulation result on a uniform grid of 451 by 301 with a uniform bathymetry of -1000m, is shown in sub-figure (b). Sub-figures (c) and (d) show the tsunami wave propagation results using the real bathymetry around Taiwan area (115°-130°E, 20°-30°N). We used the ETOPO2 bathymetry (http://www.ngdc.noaa.gov/mgg). The topography in this area is very complex, with lands, oceans and islands, and the bathymetry changes abruptly near the subduction plate edge of Manila trench and Ryukyu trench. PETSc obtains good results under both conditions.

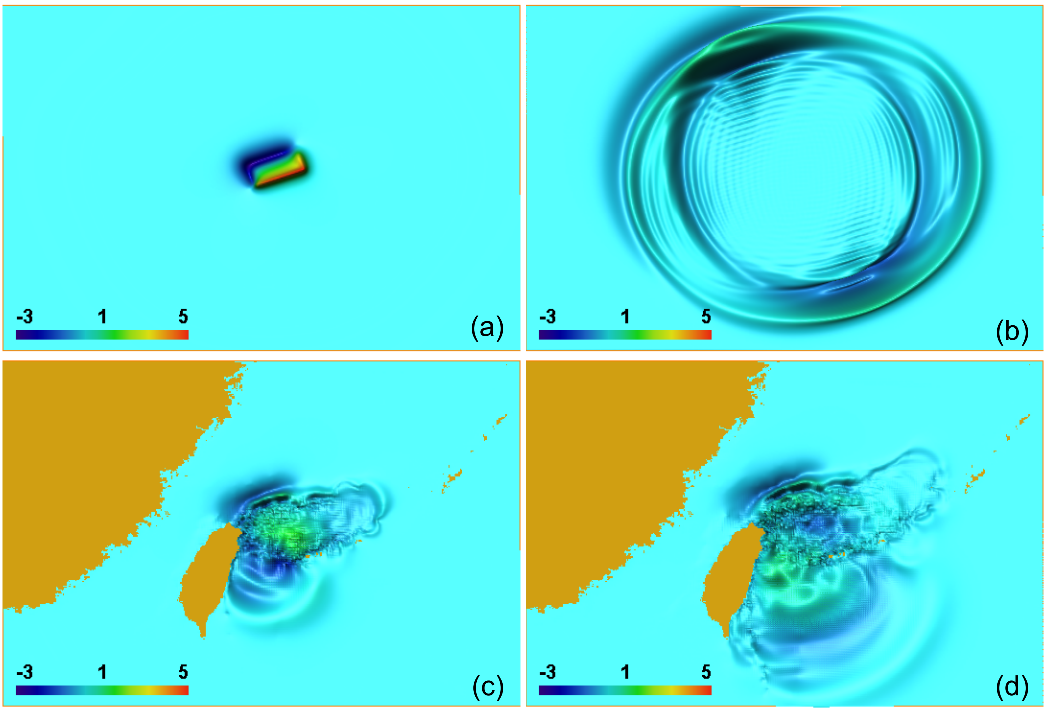

Fig. 2. Simulation results from PETSc application on nonlnear shallow water equations. The wave height is in meter(m). (a) The earthquake-generated initial deformation of the sea water, t=0minute(min). (b) tsunami wave simulation results with uniform bathymetry, at t=75min. (c) and (d) are the simulation results at t=30min and t=50min using the real bathymetry around Taiwan area.

4 Finite Element Modelling of Tsunami Wave Propagation with FEPG

4.1 General Philosophy in FEPG

FEPG stands for Finite Element Program Generator. The basic idea of FEPG is to provide a general-purpose solver with finite element discretizations for Partial Differential Equations (PDEs). Finite volume methods are also used. FEPG has a serial version, a parallel version, and a web client version. The program can be generated for both MS Windows and the Linux Operating system. It uses a domain-specific finite element programming language to describe a system of partial differential equations, and the solution method for the associated algebraic system (Fepgsoft, 2003, 2004). With a specified format for the input and output data, easy to use graphical pre-processing and post-processing software can be integrated with it.

FEPG includes a large variety of libraries for different elements type and numerical integration schemes, and several common solution methods, which can be chosen according to different problems and precision requirements. To describe a PDE with the finite element method, we first construct its weak formulation, that is, to choose the test and shape function spaces and the residual form. Then we choose the solution method, the relations among different physical fields and their solution methods respectively, etc. A description of the computational domain, such as the mesh type, node and element data, is also needed. Based on these information files, FEPG will generate all source code files necessary to carry out the calculation. A great advantage of FEPG is that it provides not only the executables but the FORTRAN source code, so you can check it easily and make modifications as you wish.

Dividing the whole calculation procedure into several components, the FEPG software make it clear which portion is which, such as the calculation of the element stiffness, mass, and damping matrices, the matrix assembly, formation of the right-hand side, iterative procedure for solution of the nonlinear algebraic problem, and the evolution by time stepping. Users can debug the program and find errors easily.

Compared with some other systems, it is not very straightforward unless the user is well acquainted with the finite element method, and considerable time and effort needs to be expended to become familiar with it. However, researchers can benefit much from this process, and it helps to do furthur study. Zhang et al. (2007) gave the procedure and more details in the use of FEPG, and Liu et al. (2007) used it to explore the multi-scale continental deformation in the Western United States.

4.2 Application on Tsunami Wave Propagation Simulation

We get the finite element formulation of the nonlinear shallow water equations (Equation (2)), with both virtual displacement principle and least squares method. Simple linearization is used for the nonlinear terms. The conjugate gradient method(CG) with ILU(0) preconditioner is used for the iterative solution. With virtual displacement principle, the integral weak form can be written as Equation (4),

$$
\begin{aligned}
&(\tfrac{\partial u^{n+1}}{\partial t}, \bar{u}) + u^n(\tfrac{\partial u^{n+1}}{\partial x}, \bar{u}) + v^n(\tfrac{\partial u^{n+1}}{\partial y}, \bar{u}) + g(\tfrac{\partial h^{n+1}}{\partial x}, \bar{u}) \\
&+(\tfrac{\partial v^{n+1}}{\partial t}, \bar{v}) + u^n(\tfrac{\partial v^{n+1}}{\partial x}, \bar{v}) + v^n(\tfrac{\partial v^{n+1}}{\partial y}, \bar{v}) + g(\tfrac{\partial h^{n+1}}{\partial x}, \bar{v}) \\
&+(\tfrac{\partial h^{n+1}}{\partial t}, \bar{h}) + u^n(\tfrac{\partial h^{n+1}}{\partial x} - \tfrac{\partial h_B}{\partial x}, \bar{h}) + v^n(\tfrac{\partial h^{n+1}}{\partial y} - \tfrac{\partial h_B}{\partial y}, \bar{h}) \\
&\qquad +(h^n - h_B)(\tfrac{\partial u^{n+1}}{\partial x} + \tfrac{\partial v^{n+1}}{\partial y}, \bar{h}) = 0
\end{aligned}
\tag{4}
$$

Where superscripts n and $n+1$ denote the time steps, and $\bar{a}$ means the virtual displacement of a. (a, b) represents the inner product of a and b.

With the least-squares method, the integral weak form can be written as Equation (5),

$$\begin{aligned}
(l_x, \bar{l}_x) + (l_y, \bar{l}_y) + (l_h, \bar{l}_h) = u^n * (\bar{l}_x) + v^n * (\bar{l}_y) \\
+(h^n + \tfrac{\partial h_B}{\partial x} * u^n * dt + \tfrac{\partial h_B}{\partial y} * v^n * dt) * (\bar{l}_h) \\
l_x = u^{n+1} + \tfrac{\partial u^{n+1}}{\partial x} * u^n * dt + \tfrac{\partial u^{n+1}}{\partial y} * v^n * dt + \tfrac{\partial h^{n+1}}{\partial x} * g * dt \\
l_y = v^{n+1} + \tfrac{\partial v^{n+1}}{\partial x} * u^n * dt + \tfrac{\partial v^{n+1}}{\partial y} * v^n * dt + \tfrac{\partial h^{n+1}}{\partial v} * g * dt \\
l_h = h^{n+1} + \tfrac{\partial h^{n+1}}{\partial x} * (h^n - h_B) * dt + \tfrac{\partial v^{n+1}}{\partial y} * (h^n - h_B) * dt \\
+\tfrac{\partial h^{n+1}}{\partial x} * u^n * dt + \tfrac{\partial h^{n+1}}{\partial y} * v^n * dt
\end{aligned} \tag{5}$$

In figure 3, we show some typical results of our simulation with virtual displacement formulation. Compared with the least squares formulation, the virtual displacement formulation can resolve more detailed wave information, and can handle the boundary condition more easily. Total absorption along the open ocean boundaries is easily achieved. We use the same initial condition as that in our PETSc application. The simulation results with uniform bathymetry are shown in sub-figure (a) and (b). With the real bathymetry around Taiwan area, the results are shown in sub-figure (c) and (d). FEPG can obtain stable resolution with uniform bathymetry. It gives good results at the beginning of the run with real bathymetry, but the long-time run shows that the solution doesn't converge, possibly due to the complex bathymetry, and needs more investigaton.

5 Results Analysis

We compared the tsunami wave patterns and wave height fields obtained from our PETSc and FEPG simulation. It is shown that, under both conditions of uniform bathymetry and real bathymetry around Taiwan area, the wave propagation patterns from two models coincide very well, as shown in figure 4. The maximum wave height vary slightly, within a range of 5%. The minimum wave height show a bigger difference, about 15%, as shown in table 1. Because PETSc and FEPG use finite difference and finite element discretization method respectively, and they use different solution methods, and PETSc use uniform grid sizes along x- and y- directions while FEPG calculate the coordinates of each node according to their real longitude and latitude, some of the difference in the results can be accounted for.

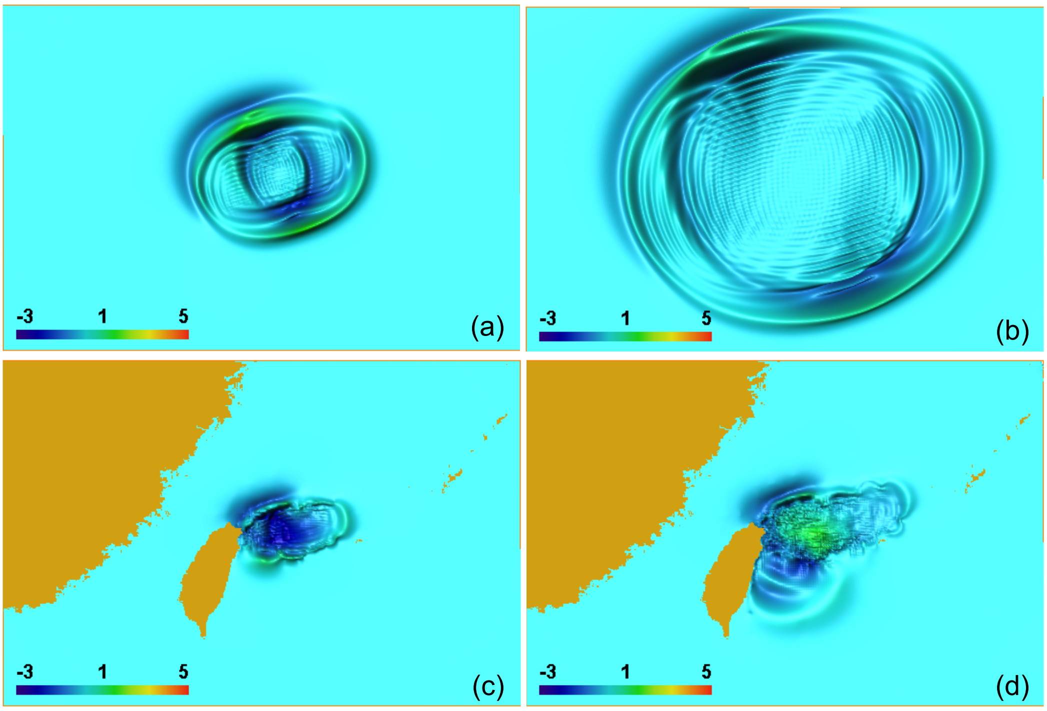

Fig. 3. Tsunami waves induced by an earthquake, modeling with FEPG. (a) and (b) are the simulation results with uniform bathymetry, at t=30min and t=75min respectively. (c) and (d) are the simulation results using the real bathymetry around Taiwan area, at t=15min and t=30min respectively.

We then compared our models with the TUNAMI model from Tohoku university (Goto et al., 1997). In the TUNAMI model, the shallow water equations are integrated along the depth direction, and the water flux in x- and y- directions replace the velocities as two variables. It uses the finite difference discretization method, forward differences in time domain, and central differences with a leap-frog scheme in the spatial domain. The comparison shows that, with uniform bathymetry, the TUNAMI model obtains similiar wave patterns and wave height fields to our models. But with real bathymetry around Taiwan area, the wave patterns show much larger differences, and the height fields are quite different, as shown in figure 4 and table 1. Compared with FEPG, the maximum waveheight difference with uniform bathymetry can reach up to 25%, and the difference is even larger with real bathymetry, reach up to 300%. This is a very interesting phenomena, and worthy of further investigation.

With further analysis, we get the tsunami wave propagation time-distance graph and the maximum wave height distribution graph based on the PETSc model result, as shown 5. And also the tsunami wave on specified locations can be drawn, as figure 6.

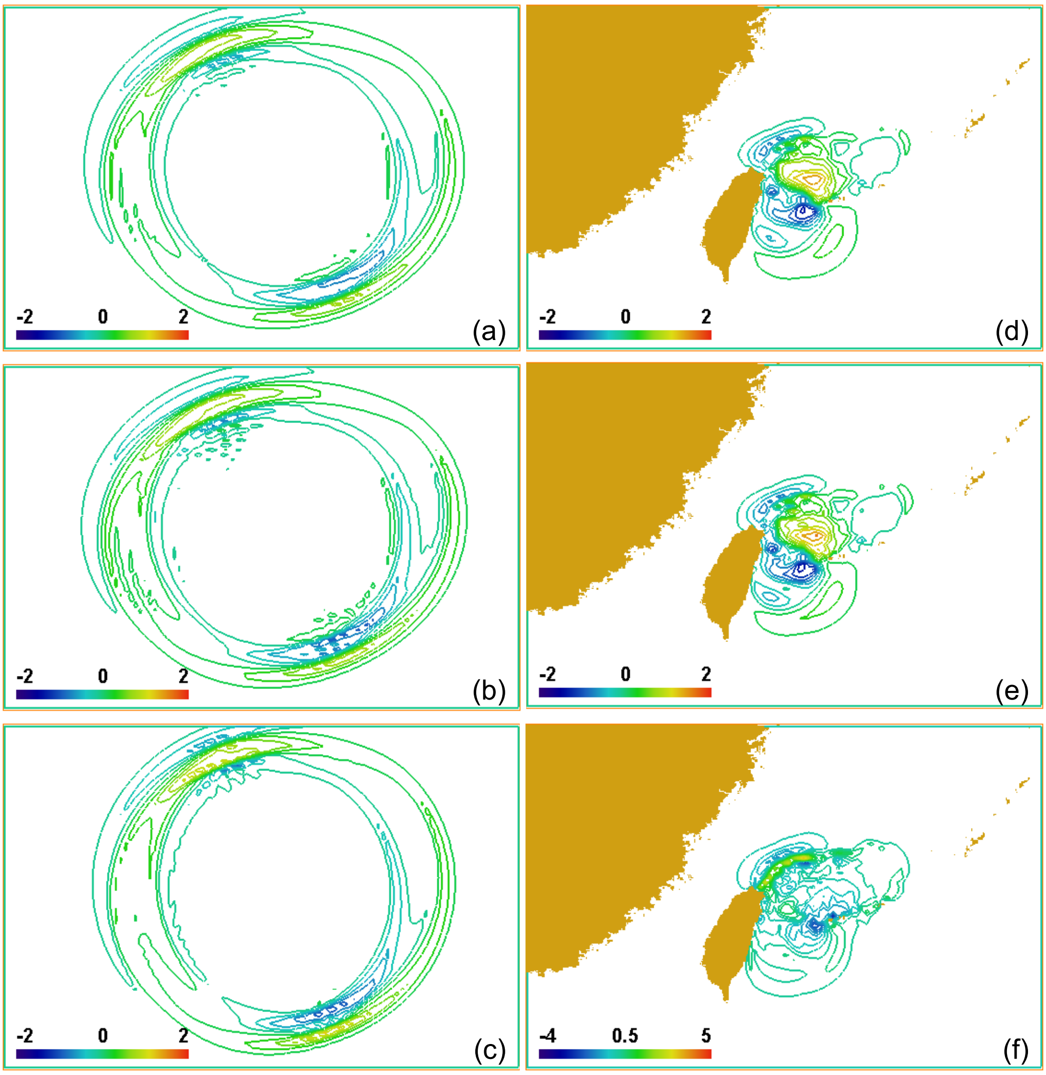

Fig. 4. Isolines of wave propagation patterns, (a) (b) and (c) are isolines of the wave propagation results from PETSc, FEPG and TUNAMI respectively, with uniform bathymetry, at t=75min, (d) (e) and (f) are isolines of the wave propagation results from PETSc, FEPG and TUNAMI, with the real bathymetry around Taiwan area, at t=30min.

Wave Height(m)	Uniform Bathymetry（-1000m）				Real Topography around Taiwan			
	t=30minutes		t=75min		t=15min		t=30min	
	min	max	min	max	min	max	min	max
PETSc	-1.679	1.598	-0.965	0.926	-2.584	1.153	-1.894	1.756
FEPG	-1.737	1.606	-1.133	0.960	-2.743	1.164	-1.967	1.807
TUNAMI	-1.750	1.969	-1.146	1.228	-3.440	5.308	-3.748	4.057

Table 1
Comparisions among wave height fields from PETSc, FEPG and TUNAMI.

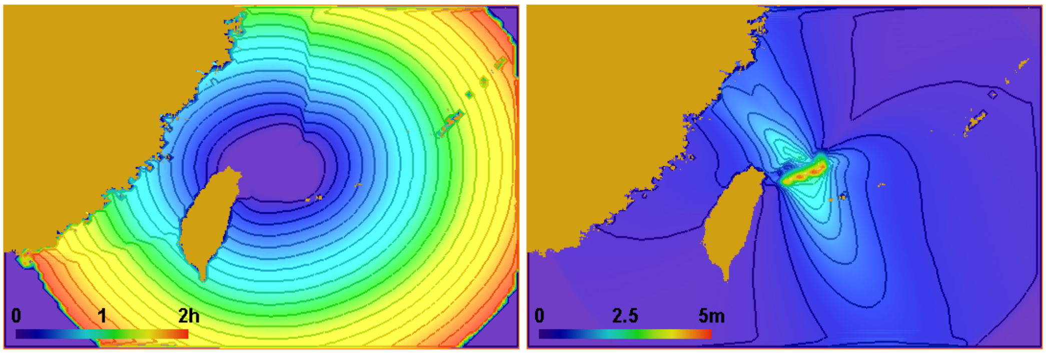

Fig. 5. Tsunami wave propagation time-distance graph (left) and the maximum wave height distribution graph, based on the PETSc model result.

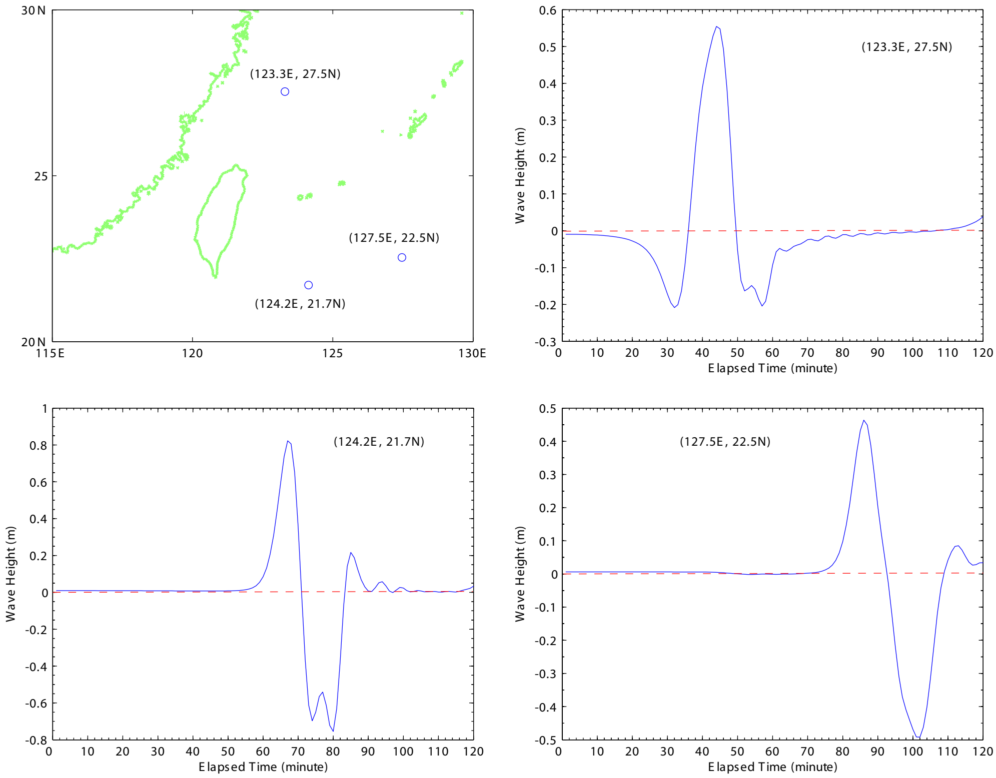

Fig. 6. Tsunami wave graph on three specified locations, based on the PETSc model result.

6 Discussions

With the progress in computational techniques and numerical methods, modern software is enabling people to do research more quickly and efficiently. Armed with these frameworks, we are liberated from the cumbersome task of coding which is also prone to error, to focus on the physical problem. And it gives us an easy way to use the many state of the art technologies. FEPG and PETSc show us the wide variety of techniques in numerical computation, such as the finite element and finite difference methods, and they apply high performance computing techniques, such as parallel computing using MPI and the domain decomposition method. We used the finite difference method and restarted GMRES solver with the PETSc library in our tsunami simulation, and found it quite easy to implement and the code is highly efficient. The code is inherently parallel and fit for large-scale computing on big machines. We tested our PETSc code on both serial and parallel machines. The speedup was excellent on the parallel machines. Through the FEPG web client, we ran the FEPG serial library. The code runs slower and needs more memory because of the intrinsic characteristic of finite element method. FEPG now provides a parallel library for improved performance. With the finite element description files we wrote, we can easily generate the parallel FEPG source codes through the FEPG parallel version, enabling us to run larger problems on parallel machines.

Scientific visualization is also becoming an important part of scientific computing, as it becomes increasingly complex. Good pictures give us more intuitive knowledge and better insight into the physical phenomenon. Both preprocessing and postprocessing need good visualization. Many general-purpose and application-specific visulization packages have been developed for this purpose, e.g. VTK, ParaView, Gid, Amira, and Tecplot. FEPG uses Gid as its default pre- and post- processing visualization software. We used Amira to analyze the simulation result in our application. Amira detects the very small perturbation in the tsunami height field and draws vivid output which was invaluable in testing the code.

In the future, we will develop our model further, adding the effects of wave run-up and run-down, wave breaking, the friction caused by sea floor, etc. These effecets are crucial to tsunami mitigation. Tsunami generation are sensitive to fault rupture process, and further study should include observations like GPS-Shield (Sobolev et al., 2007) to help inverse fault rupture parameters. A more complex 3-D model is also one future research area. Although we have only a few truly useful results at this stage, with the continuous development of our model, we can take full advantage of these software platforms and other technologies to explore both the earthquake and tsunami hazards in much greater detail, and hopefully make better predictions leading to hazard prevention and mitigation.

7 Acknowledgments

Prof. Michael Navon from Florida State University gave good suggestions and made corrections, Prof. Fumihiko Imamura from Tohoku University provided the TUNAMI model, and NingNing Chen from Fepgsoft provided useful discussions. Many thanks to them. This research is supported by the National Natural Science Foundation of China, and in part by the MICS division of the U.S. Dept. of Energy under Contract DE-AC02-06CH11357.

References

Balay, S., Buschelman, K., Eijkhout, V., Gropp, W., Kaushik, D., Knepley, M., McInnes, L., Smith, B., Zhang, H., 2006. Petsc users manual. ANL-95/11-Revision 2.3.2.

Fepgsoft, 2003. Finite element method based on FEPG. http://www.fegensoft.com/.

Fepgsoft, 2004. How to generate the finite element program with the formulation. http://www.fegensoft.com/.

Goto, C., Ogawa, Y., Shuto, N., Imamura, F., 1997. Numerical method of tsunami simulation with the leap-frog scheme (IUGG/IOC Time Project. IOC Manual, UNESCO 35.

Haidvogel, D., Beckman, A., 1999. Numerical Ocean Circulation Theory. World Scientific Publishing.

Hanert, E., Le Roux, D., Legat, V., E., D., 2005. An efficient eulerian finite element method for the shallow water equations. Ocean Modelling 10, 115–136.

Kirby, J., 1996. Nonlinear, dispersive long waves in water of variable depth. Research Report No. CACR-96-02.

Kirby, J., Wei, G., Chen, Q., Kennedy, A., Dalrymple, R., 1998. Funwave 1.0, fully nonlinear boussinesq wave model documentation and users manual. Research Report No. CACR-98-06.

Kirby, J. T., 1997. Gravity waves in water of variable depth. In: Hunt, J. N. (Ed.), Nonlinear, dispersive long waves in water of variable depth. Computational Mechanics Publications, Ch. 3, pp. 55–126.

Liu, M., Yang, Y., Li, Q., Zhang, H., 2007. Parallel computing of multi-scale continental deformation in the western united states: Preliminary results. Physics of Earth and Planeteray interiors 163, 35–51.

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

Okada, Y., 1985. Surface deformation due to shear and tensile faults in a half-space. Bull. Seism. Soc. Am. 75, 1135 –1154.

Okada, Y., 1992. Internal deformation due to shear and tensile faults in a half-space. Bull. Seism. Soc. Am. 82, 1018–1040.

Pedlosky, J., 1987. Geophysical Fluid Dynamics. Springer-verlag, New York.

Shuto, N., 1991. Numerical simulation of tsunamis - its present and near future. Natural Hazards 4, 171–191.

Sobolev, S., Babeyko, A., Wang, R., 2007. Tsunami early warning using gps-shield arrays (to be appeared). Journal of Geophysical Research.

Titov, V., Gonzalez, F., 1997. Implementation and testing of the method of splitting tsunami (most) model. NOAA Technical Memorandum ERL PMEL-112.

Wang, R., Lorenzo-Martin, F., Roth, F., 2006. Psgrn/pscmp-a new code for calculating co- and post-seismic deformation, geoid and gravity changes based on the viscoelastic-gravitational dislocation theory. Computers and Geosciences 32, 527–541.

Watts, P., , Grilli, S., Kirby, J., Fryer, G., Tappin, T., 2003. Landslide tsunami case studies using a boussinesq model and a fully nonlinear tsunami generation model. Natural Hazards and Earth System Sciences 3, 391–402.

Wei, G., Kirby, J., Grilli, S., Subramanya, R., 1995. A fully nonlinear boussinesq model for surface waves. Journal of Fluid Mechanics 294, 71–92.

Witting, J., 1984. A unifed model for the evolution of nonlinear water waves. Journal of Computational Physics 56, 203–236.

Zhang, H., Liu, M., Shi, Y., Yuen, D., Yan, Z., Liang, G., 2007. Toward an automated parallel computing environment for geosciences. Physics of Earth and Planeteray interiors 163, 2–22.