# ANALYSIS OF WAVE-BODY INTERACTONS USING ADAPTIVE FINITE ELEMENT MESHES

**D.M. Greaves[1], A.G.L. Borthwick[1], R. Eatock Taylor[1] and G.X. Wu[2]**

[1] Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ

[2] Department of Mechanical Engineering, University College London, Torrington Place, London WC1E 7JE

## SUMMARY

An adaptive finite element grid generator coupled with potential flow theory is used to solve the two-dimensional nonlinear time domain free surface flow problem. The problem is solved by a time marching method, and the mesh is adapted at each time step to follow the moving free surface. The finite element meshes are created by triangularisation of quadtree grids and are refined at the free surface and body boundaries. Triangular elements and linear shape functions are used, and the solution is obtained by the Galerkin method (Wu and Eatock Taylor, 1994). Numerical results have been obtained for wave elevation histories of standing waves in a container and a vertical wavemaker. Results will also be provided for a standing wave interacting with a fixed rectangular body in a container. The adaptive finite element method achieves high resolution where required and is economic in the overall number of elements.

## 1. INTRODUCTION

It is desirable to simulate wave loading on, and interactions with, large volume offshore structures using fully nonlinear theory. Fully nonlinear problems are usually solved by a time marching method, which assumes that the wave profile and position of the structure are known at a particular instant. The problem can then be solved numerically at successive time steps using potential theory. The potential at the free surface is the boundary condition for the next time step calculation, and the new free surface profile is calculated from velocities at the surface. A two dimensional implementation of this procedure based on a finite element analysis has been described by Wu and Eatock Taylor (1994), and its extension to three dimensions is summarised in another abstract at this workshop (Wu, Ma and Eatock Taylor, 1995). The success of the method in the case of complicated geometries is expected to rely on a flexible adaptive finite element grid generator. Preliminary results from the use of such a mesh generator are given here.

This paper describes a new technique for adaptive finite element mesh generation based on quadtree grids. The finite element meshes are produced at each time step by (i) generating a quadtree grid about seeding points which describe the free surface and body geometry, and then (ii) dividing the quadtree grid into a mesh of triangular finite elements. The grid is regenerated at each time step about new positions of the body and free surface boundaries as they move in time. It is intended to extend the two dimensional finite element method to three dimensions at a future date; this can be done using octree grids divided into a mesh of tetrahedral elements. Section 2 outlines the philosophy of the adaptive grid generator and Section 3 gives the finite element equations.

## 2. ADAPTIVE GRID GENERATION

The quadtree grid is generated inside a unit square by recursive subdivision about a set of seeding points. Each division splits the divided cell into four equal sized smaller cells. The resulting grids are refined at the seeding points and coarser elsewhere. Further subdivisions are then performed within the grid to ensure that cell edge ratios of neighbouring cells do not exceed 2:1. The resulting grids have a hierarchical tree structure which enables efficient data storage and fast neighbour finding using tree traversal techniques. Quadtree grids are discussed by Gáspár, Józsa and Simbierowicz (1991) who have applied multigrid-quadtree meshes to solve species transport and linearised shallow flow problems in complex domains.

The numbering system adopted herein is based on that of van Dommelen and Rundensteiner (1989), combined with ideas expressed by Samet (1990). The reference number for each panel contains information about its location within the quadtree stored as a set of x and y translations, and combined into a single integer. This can be summarised as

$$N = \sum_{i=0}^{m-1} N_i \, 5^i \tag{1}$$

where $m$ is the division level of the panel and $N_i$ takes the integer value 1, 2, 3, or 4 depending on the quadrant in which the panel is located within its parent cell. In this scheme, 1 refers to North West, 2 to South West, 3 to North East and 4 to South East. Figure 1 shows the quadtree grid generated about a set of boundary seeding points which describe a sinusoidal standing wave profile in a rectangular tank containing a rectangular body.

Triangular finite elements are produced from the quadtree grid by joining the centres of three neighbouring square cells. This method eliminates hanging midside nodes which occur in quadtree grids. The triangularisation may be extended to three dimensions by joining centres of four neighbouring cubes to form tetrahedral elements. The triangularisation method utilises certain boundary seed points as well as cell centres, and so produces a close approximation to the free surface boundary. Triangles adjacent to the boundary are initially defined using two seed points plus an internal node selected in order to optimise the aspect ratio of the triangle. Where the boundary curvature is higher than a prescribed limit, or when adjacent boundaries move together, such as run-up at the body or jet formation, additional elements are necessary. In these regions, extra seeding points are introduced and removed adaptively, depending on the spatial and angular proximity of adjacent points. Whenever internal nodes are unavailable at the apex of these regions, the apex triangle is divided further. For the quadtree grid shown in Figure 1, the elements on the boundary are given in Figure 2. The interior elements of the mesh are then defined by joining the cell centres of each quadtree cell and its east, southeast and south neighbours if they exist. The presence of any holes in the mesh is prevented by checking whether each edge has exactly two elements connected to it, and performing further triangularisation if any do not. Figure 3 shows the full finite element mesh obtained from the quadtree grid in Figure 1.

## 3. FINITE ELEMENT EQUATIONS

The unknown potentials, $\phi$, at nodes off the free surface $S_F$ are expressed in terms of the potentials, $\phi_F$, on the free surface by means of the matrix equation

$$\boldsymbol{A}\,\boldsymbol{\phi} = \boldsymbol{P} - \boldsymbol{A}_F\,\boldsymbol{\phi}_F \tag{2}$$

$\phi_F$ are obtained by advancing the solution from the previous time step, and the vector $\boldsymbol{P}$ accounts for the known boundary term $f$ on the body surface $S_B$. The finite element discretisation in terms of the shape functions $N_i$ leads to the terms

$$A_{ij} = \int_R \nabla N_i . \nabla N_j \, dR, \;\; i,j \notin S_F \; ; \quad P_i = \int_{S_B} N_i \, f dS, \;\; i \notin S_F \tag{3}$$

The terms of $A_F$ are similar to those of $A$, but for $j \in S_F$.

## 4. RESULTS

In the preliminary phase of testing the procedures, attention has been directed towards standing waves in a container. The test case used for the DNV comparative study will serve here to illustrate the type of mesh which is generated by the quadtree method. The initial wave elevation was taken to be

$$\eta(x,0) = \alpha[1 - (\frac{x}{\beta})^2] \exp(-x^2/\gamma^2) \tag{4}$$

where $\alpha = 0.1714$, $\beta = 0.7571$, $\gamma = 1.0857$, and all lengths have been non-dimensionalised by the depth of the tank. Figures 4 and 5 show the adapting mesh at dimensionless times given by $t(d/g)^{1/2}$ = 0 and 1.72, where $d$ is the depth of the tank, and its width is 2.286$d$. Figure 6 shows the computed time history of surface elevation at $x = 0.8571$.

## 5. CONCLUSIONS

The adaptive grid generator presented herein refines the mesh at the free surface and body, and provides moving boundaries for the fully nonlinear solution. The grid generator is relatively expensive in CPU time for simple geometries, but the results obtained are found to agree well with published data. The method should be extended to more complicated geometries and 3-D cases where its advantages in terms of flexibility and adaptivity will become more compelling.

## 6. ACKNOWLEDGEMENTS

This work forms part of the research programme "Uncertainties in Loads on Offshore Structures" sponsored by EPSRC through MTD Ltd and jointly funded with: Amoco (UK) Exploration Company, BP Exploration Operating Company Ltd, Brown & Root, Exxon Production Research Company, Health and Safety Executive, Norwegian Contractors a.s., Shell UK Exploration and Production, Den Norske Stats Oljeselskap a.s., Texaco Britain Ltd.

## 7. REFERENCES

**Van Dommelen, L. and Rundensteiner, E.A. (1989)** "Fast, Adaptive Summation of Point Forces in the Two-Dimensional Poisson Equation", J. Comp. Physics, Vol. 83, pp. 126-147.

**Gáspár, C. and Józsa, J. and Simbierowicz, P. (1991)** "Lagrangian Modelling of the Convective Diffusion Problem Using Unstructured Grids and Multigrid Technique", Proc. 1st Int. Conf. on Water Pollution (Modelling, Measuring and Prediction), 3-5 September 1991, Southampton, U.K.

**Samet, H. (1990)** "The Design and Analysis of Spatial Data Structures", Addison-Wesley.

**Wu, G.X. and Eatock Taylor, R. (1994)** "Finite Element Analysis of Two-Dimensional Non-Linear Transient Water Waves", Applied Ocean Research, in the press.

**Wu, G.X., Ma, Q.W. and Eatock Taylor, R. (1995)** "Non-Linear Wave Loading on a Floating Body", Proc. 10th International Workshop on Water Waves and Floating Bodies, Oxford.

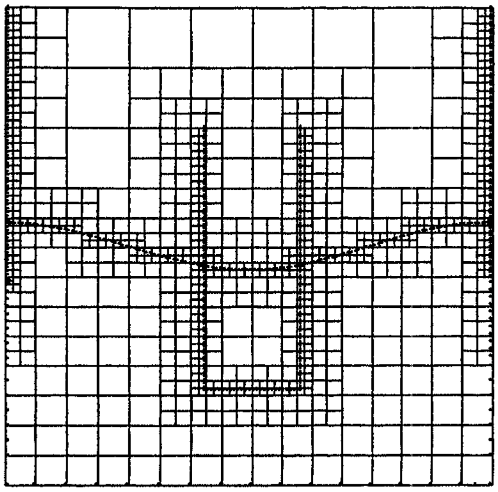

Fig. 1 Quadtree grid

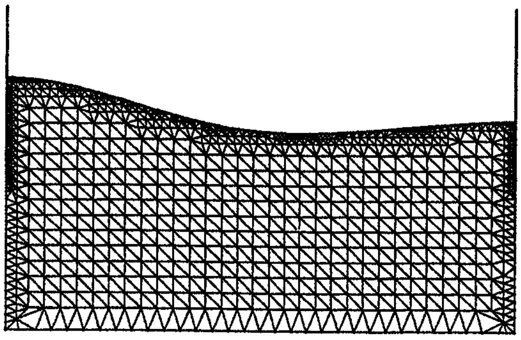

Fig. 4 Mesh at $t(d/g)^{1/2} = 0$

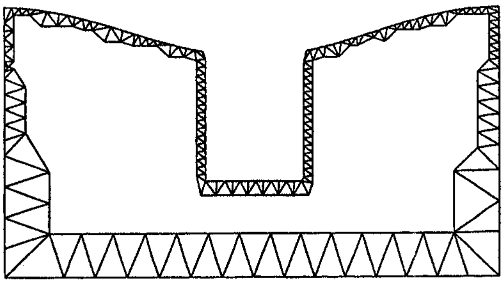

Fig.2 Elements on boundary

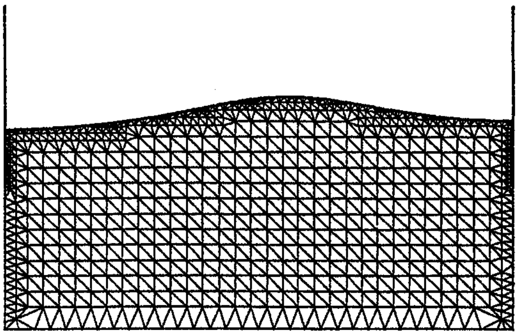

Fig. 5 Mesh at $t(d/g)^{1/2} = 1.72$

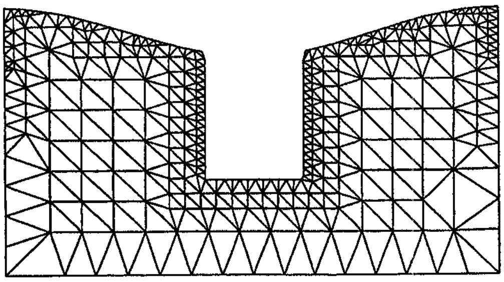

Fig.3 Full finite element mesh

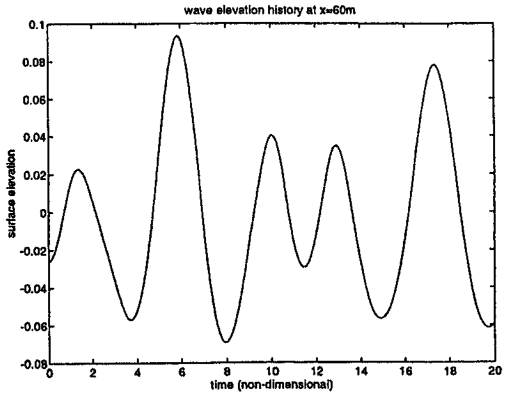


Fig. 6 Time history at $x = 0.857$

## DISCUSSION

**Martin, P. A.:**: Do you have to re-grid the whole domain at each time-step, even though only one part of the boundary has moved? Here, I am thinking about the effects on your tree-structure, in particular.

**Greaves, D. M., Borthwick, A. G. L., Eatock Taylor, R. & Wu, G. X.**: At this stage yes, the whole domain is regridded at each timestep. However, it may be possible to save the base quadtree grid (generated to the minimum division level) and adapt the moving boundary only at each timestep. This would mean saving the base tree and adding and taking away extra "branches" as the grid adapts at each time step.