# COMPUTATION OF FULLY NONLINEAR FREE SURFACE FLOWS IN THREE DIMENSIONS

Pierre FERRANT
Sirehna SA
1, rue de la Noë 44071 Nantes Cedex, FRANCE

During the 6th workshop of this series, we presented the basic principles and some applications of a computer code dedicated to the simulation of linearized unsteady free surface flows in three dimensions (1). The code was based on a boundary integral equation method using linear isoparametric triangular elements, with a fourth order Runge-Kutta scheme for the time-stepping. In the present abstract, we report on the extension of the method to the simulation of fully nonlinear flows. The mixed Eulerian-Lagrangian approach initiated by Longuet-Higgins and Cokelet and now commonly used in two dimensions is adopted. Apart some specific difficulties of the 3D case such as the tracking of intersecting surfaces, the primary obstacle lies in the amount of CPU time necessary to simulate 3D flows with sufficient resolution and accuracy. We thus first improved the numerical efficiency of the initial integral equation solver in order to obtain acceptable CPU times for nonlinear simulations. These improvements are detailed in the first section of the abstract. Some results of the fully nonlinear version of the code, which is in its early stage of development, are then discussed. These results concern the forced motion of a submerged body, the problem of intersecting surfaces being avoided. The code behaves correctly in body nonlinear configuration, but problems of stability still occur in fully nonlinear simulations.

## NUMERICAL PROCEDURE

The major part of the computing time is devoted to the solution of the boundary integral problem at each time step, i.e. computation of the set of influence coefficients and solution of the linear system of equations. In the present code, triangular flat panels with linear variation of sources and dipoles, and control points at panel vertices are used. The corresponding influence coefficients are computed analytically for the near field, and using very simple asymptotic formulas for the far field. This task is $O(N^2)$ where N is the number of unknowns.

For the solution of linear systems of equations, an $O(N^3)$ direct method (LU) was initially used in the linear version of the code where the time-invariant coefficient matrix was inverted once for all at the beginning of the simulation. For nonlinear computations, the coefficient matrix changes each time the geometry is updated, that is at least at each time step. An $O(N^3)$ solver is thus absolutely unsuitable. Various existing $O(N^2)$ iterative methods for nonsymmetric systems have been considered: CGS (2), CGSTAB (3), MGRAD (4) and GMRES (5). They have been compared on a test case consisting of a uniformly panelled cubic domain with mixed Neumann and Dirichlet boundary conditions given by the influence of a set of point sources situated outside the domain. Given the potential on Dirichlet boundaries (free surface) and its normal derivative on Neumann boundaries, the solver computes the corresponding normal derivative and potential which are compared to exact analytical values. Figure 1 gives the convergence of CGS, CGSTAB and MGRAD on this test case with 1106 unknowns. The rate of convergence of GMRES on this case is the same as MGRAD. Each method is used here with diagonal preconditioning. The better behaviour is obtained with MGRAD and GMRES, with a monotonic convergence. Furthermore, these two methods are faster, with only one matrix-vector product per iteration. Finally GMRES has been preferred. Convergence problems occurred with MGRAD during tests with more than 3000 unknowns, while GMRES never failed to converge. Furthermore, the CPU time per iteration is slightly lower than for MGRAD. The test has been repeated with GMRES with increasing numbers of unknowns, see figure 2. We observe that for a given accuracy criterion, the number of iterations is not much affected by the number of unknowns, which guarantee that the solution time remains about $O(N^2)$. The global error of the numerical solution is given in table 1. The rate of convergence is $O(h^2)$, were h is the typical panel size.

As stated above, the time-stepping procedure is based on a 4th order Runge-Kutta scheme with dynamic control of the time step size. The method is used with "frozen" coefficients, that is the geometry is updated only once at each time step, while four solutions of the boundary value problem are necessary. Even with only one evaluation per time step, the CPU time for the computation of influence coefficients remains predominant. GMRES is systematically used with diagonal preconditioning, and iterations are stopped when Norm(b-Ax)/Norm(b) <1.E-6. An initial guess of the solution is given by quadratic extrapolation from previous time steps, which allows a moderate, but appreciable reduction of the necessary number of iterations.

## SOME NUMERICAL RESULTS

We describe here some of the first results obtained with the nonlinear version of the code. The test problem consists in a submerged sphere of unit radius R, with a mean depth of submergence Zo/R=-2., submitted to forced motions. This problem has been previously thoroughly investigated using a body-nonlinear time domain Kelvin code (6). As an intermediate step, a body-nonlinear version of the present Rankine panel code has been compared to the Kelvin code. The water depth is infinite, and only the body and a circular portion of the free surface are discretized, with no particular absorbing condition. The results of the present Rankine panel code for the vertical force on a periodically heaving sphere with amplitude A/R=0.7 and wavenumbers $K=\omega^2=0.5$, 2.0, 3.0 are presented respectively in figures 3, 4 and 5. These results agree with those of the Kelvin code within graphical accuracy, except at the end of the simulation where the present results are affected by reflections on the outer boundary. A perspective view of the wave field computed with the present panel code is given by figure 6, for K=0.5, and clearly exhibits large amplitude higher harmonics, as predicted by the Kelvin computations. Of course, in the absence of any absorbing condition for the radiated waves, the computation is valid only until the reflection of the leading wave on the outer boundary.

We are now testing the method in fully nonlinear conditions. Problems of stability still occur at the free surface. Figure 7 gives radial cuts of the free surface in the case of a heaving sphere with Zo/R=2.0, A/R=0.2, and $\omega=1.0$. The plot corresponds to the first period of motion, after which the computation breaks down. This failure is probably due an insufficient spatial resolution, and/or the absence of numerical smoothing at the free surface. This problem is being investigated, and we do hope to be able to present extended results at the Workshop.

*This study is supported by the French Ministry of Defence, under contract Sirehna/DRET 91/11.*

## REFERENCES

(1) Ferrant P. ,"Threedimensional unsteady wave-body interactions in a bounded domain". 6th Workshop on Water Waves and Floating Bodies, Woods Hole, Mass., 1991.

(2) Sonnenveld P., "CGS, a fast Lanczos-type solver for nonsymmetric systems of linear equations". SIAM J. Sci. Stat. Comput. Vol.10, pp 35-52, 1989.

(3) Van der Vorst M.A., Sonnenveld P., "CGSTAB: A more smoothly converging variant of CGS", 1990.

(4) Ryan J., Le T.H., Morchoisne Y., "Panel code solvers", 7th GAMM Conference on Numerical Methods in Fluid Dynamics, 1988.

(5) Saad Y., Schultz M.H., "GMRES: A Generalized Minimal RESidual algorithm for solving nonsymmetric linear systems", Research Report Yale University RR254, 1983.

(6) Ferrant P. ,"A coupled time & frequency domain approach for nonlinear wave radiation". 18th ONR Symposium on Naval Hydrodynamics, Ann Arbor, Michigan, 1990.

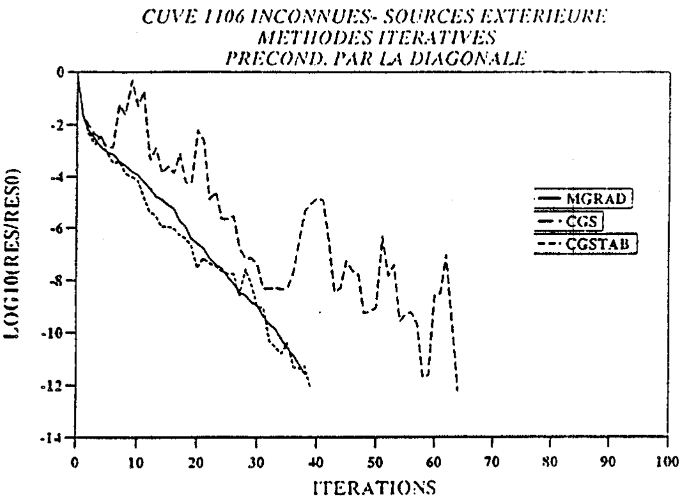


FIGURE 1

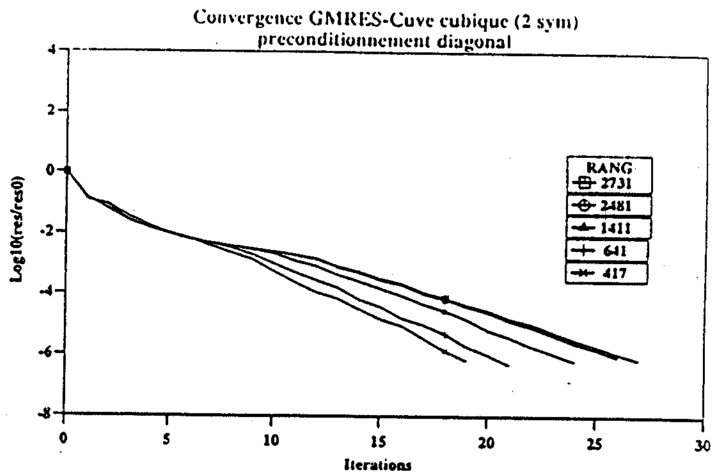


FIGURE 2

| Panels | Points | Panel size h | Mean relative error | | $E_N/h^2$ | $E_D/h^2$ |
|---|---|---|---|---|---|---|
| | | | Neumann $E_N$ | Dirichlet $E_D$ | | |
| 768 | 417 | 0.0625 | $5\ 10^{-4}$ | $8.4\ 10^{-3}$ | 0.128 | 2.15 |
| 1200 | 641 | 0.05 | $3.3\ 10^{-4}$ | $5.5\ 10^{-3}$ | 0.132 | 2.2 |
| 2700 | 1411 | 0.033 | $1.5\ 10^{-4}$ | $2.6\ 10^{-3}$ | 0.135 | 2.39 |
| 4800 | 2481 | 0.025 | $8.4\ 10^{-5}$ | $1.8\ 10^{-3}$ | 0.134 | 2.88 |
| 5292 | 2731 | 0.0238 | $7.6\ 10^{-5}$ | $1.7\ 10^{-3}$ | 0.134 | 3.00 |

Table 1 - Convergence of the solver
Mixed Dirichlet-Neumann boundary conditions

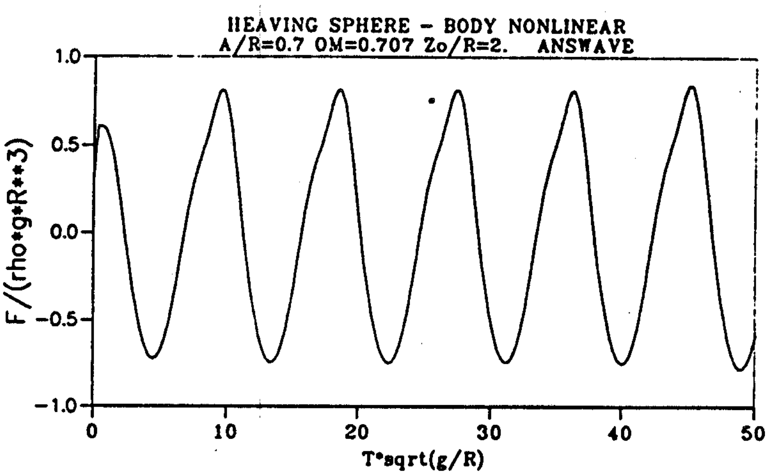


FIGURE 3

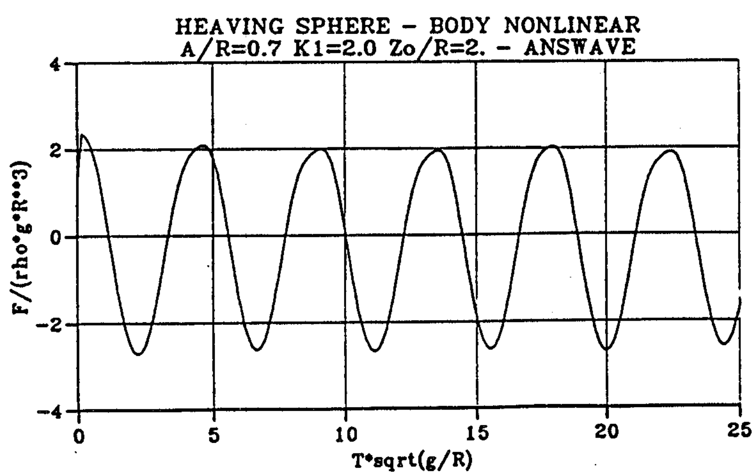


FIGURE 4

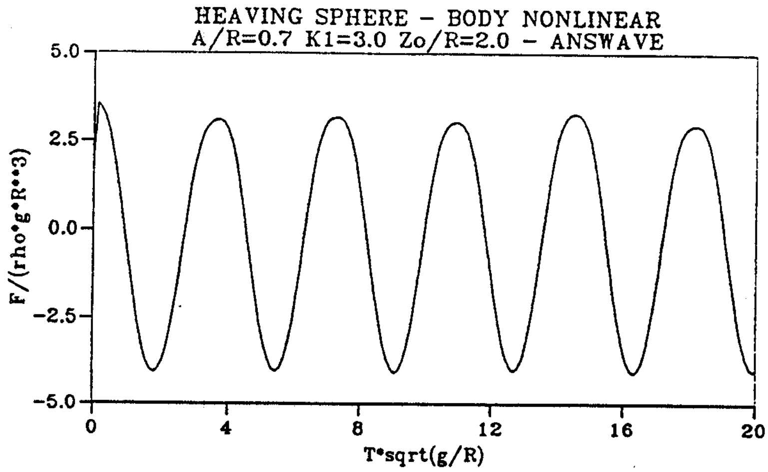


FIGURE 5

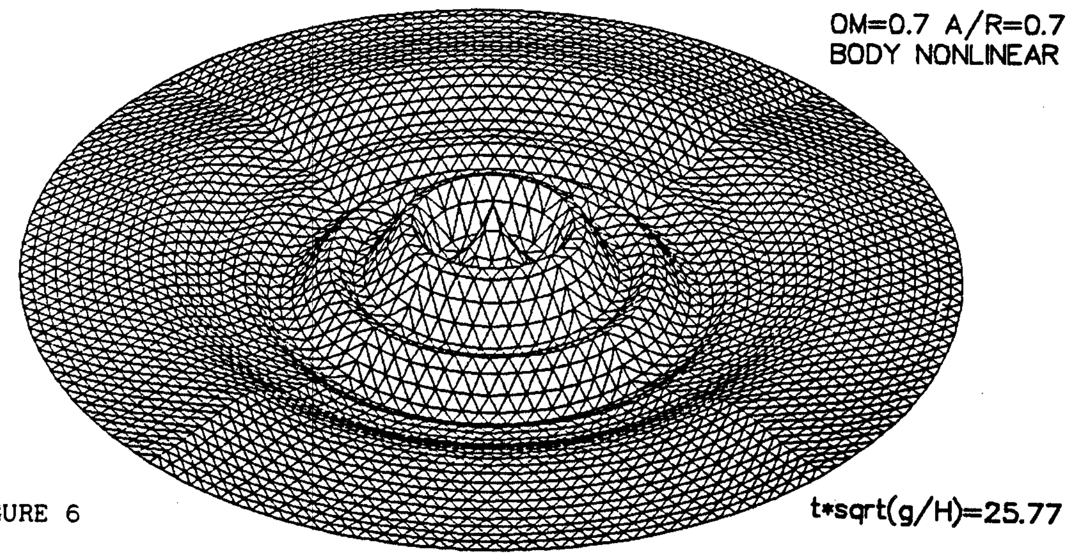


FIGURE 6

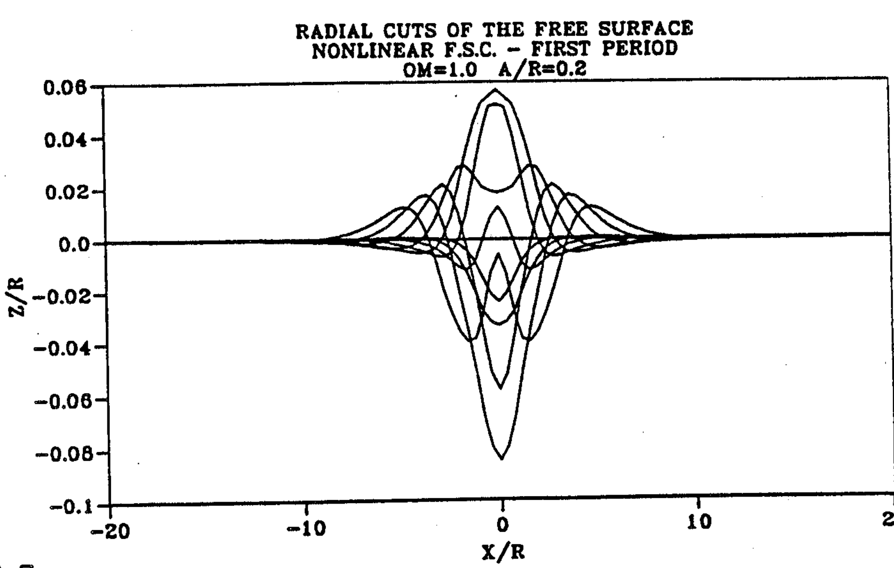


FIGURE 7