Accurate and Efficient Solution of 2D Steady Water Flows with Surface Waves and Turbulence

Jeroen Wackers[1] and Barry Koren[2]

[1] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands.
Jeroen.Wackers@cwi.nl

[2] CWI and Delft University of Technology, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands.
Barry.Koren@cwi.nl

Summary. A surface capturing method is developed for steady water–air flow with gravity. Second-order accuracy is obtained with flux limiting and turbulence is modeled with Menter's model. The equations are solved efficiently with a combination of multigrid and defect correction. Results for two test cases confirm the efficiency and accuracy of the method.

1 Introduction

Numerical simulation of steady water flow plays a continuously increasing role in the development of ships and offshore structures. Accurate predictions of a ship's wave pattern, for example, can help in reducing the ship's drag during the design, and thus minimise operating costs [3].

A key choice in the development of a numerical model for water flow is the representation of the water surface. The most widely used approach is surface fitting: the computational mesh is deformed during the computation, such that it coincides with the water surface. Another approach is the surface capturing technique (the volume of fluid and level set methods are well-known examples) where the grid is fixed, but the location of the water surface on the grid can change. These methods are more flexible than surface fitting methods; they allow steeper waves and more complex object geometries. A disadvantage is that, for steady flow problems, the equations resulting from classical surface capturing methods are hard to solve efficiently. Currently, they are usually solved by time-marching the unsteady flow equations to convergence, which is a time-consuming process.

In previous work, we have shown that efficient solution of a surface capturing model is possible [7]. We developed a volume-of-fluid model based on conserva-

H. Deconinck, E. Dick (eds.), *Computational Fluid Dynamics 2006*,
DOI 10.1007/978-3-540-92779-2_103,

tion laws only, which is solved with classical multigrid. Very fast convergence of the solution process was obtained. But it also became clear that the first-order, laminar flow model was not sufficient to accurately compute realistic water flow.

Therefore, the method is improved by making it second-order accurate and by adding a turbulence model. The second-order model is solved with a defect-correction procedure, that uses the efficiency of the first-order multigrid method. Turbulence is modeled with Menter's model [5]: a simple yet accurate one-equation turbulence model that has already been applied in some ship-flow solvers.

This paper briefly describes the first-order accurate discretisation and focuses on the extension to second-order accuracy and on the inclusion of the turbulence model. The flow equations are given in Sect. 2, the discretisation of the different terms in Sect. 3. Section 4 describes the defect correction and the special multigrid method that are used to solve the difficult two-fluid flow equations with turbulence. Finally, the practical application of the method is illustrated in Sect. 5 with a test case: the water flow in a channel with a bottom bump.

2 Flow Equations

The flow equations used here are based on the Reynolds–averaged Navier–Stokes (RANS) equations. The distinction between the water w and air a is made by adding a mass conservation equation for the water. Turbulence is modeled with Menter's model. The resulting system has conservation laws only, which makes it suitable for multigrid solution.

$$
\begin{aligned}
\frac{\partial}{\partial x}\left(p+\varrho u^2\right)+\frac{\partial}{\partial y}\left(\varrho u v\right) &= \frac{\partial}{\partial x}\left(\left(\mu+\mu_T\right)2u_x\right)+ \\
&\quad \frac{\partial}{\partial y}\left(\left(\mu+\mu_T\right)\left(u_y+v_x\right)\right) && (x\text{–mom.}), \\
\frac{\partial}{\partial x}\left(\varrho u v\right)+\frac{\partial}{\partial y}\left(p+\varrho v^2\right) &= \frac{\partial}{\partial x}\left(\left(\mu+\mu_T\right)\left(u_y+v_x\right)\right)+ \\
&\quad \frac{\partial}{\partial y}\left(\left(\mu+\mu_T\right)2v_y\right)-\varrho g && (y\text{–mom.}), \\
\frac{\partial}{\partial x}\left(u\right)+\frac{\partial}{\partial y}\left(v\right) &= 0 && (\text{tot. mass}), \\
\frac{\partial}{\partial x}\left(u\alpha\right)+\frac{\partial}{\partial y}\left(v\alpha\right) &= 0 && (\text{water mass}), \\
\frac{\partial}{\partial x}\left(\tilde{\nu}_T u\right)+\frac{\partial}{\partial y}\left(\tilde{\nu}_T v\right) &= \frac{\partial}{\partial x}\left(\left(\nu+\tilde{\nu}_T\right)\left(\tilde{\nu}_T\right)_x\right)+ \\
&\quad \frac{\partial}{\partial y}\left(\left(\nu+\tilde{\nu}_T\right)\left(\tilde{\nu}_T\right)_y\right)+P-D && (\text{Menter}),
\end{aligned}
\tag{1}
$$

where α is the volume fraction of water, $\nu = \mu/\varrho$ and $\tilde{\nu}_T$ is a scaled version of ν_T [5]. P and D are turbulence production and dissipation terms, based on velocity derivatives.

3 Discretisation

The system (1) is discretised with a cell-centered finite-volume technique on structured, curvilinear grids. The convective and diffusive fluxes and the source terms are discretised separately, to independently control the stability of each part. Both first- and second-order accurate fluxes are used. The derivation of the first-order fluxes is given in [7]. These fluxes are briefly described here, together with a description of the second-order accurate fluxes and the turbulence source terms.

Convective fluxes The convective fluxes are computed by reconstructing the states $\mathbf{q} = [u, v, p, \alpha, \tilde{\nu}_T]^T$ on the left and right side of the cell faces from the states in the cell centres. These states are then put in a flux function, that gives the flux over the cell face. For the first-order accurate fluxes, the state at the cell faces is taken equal to the state in the cell centres. For the second-order accurate fluxes, the states are reconstructed with a slope limiter. We use the $\kappa = \frac{1}{3}$ limiter proposed by Koren [4]. The flux function itself is defined in [7]. It is an approximate Riemann solver, based on artificial compressibility and comparable to the flux used in [2].

Diffusive fluxes The diffusive fluxes in the momentum equations are modeled with central differences. These are both stable and second-order accurate, therefore they are used for both the first- and the second-order accurate discretisation. For non-cartesian grids and for the cross-diffusion terms, the velocity derivatives are computed with a control volume approach.

Source terms Two different source terms appear in the system (1). The gravity force in the y–momentum equation is discretised as $-\varrho_i g\,\Delta x\,\Delta y$. The production and dissipation terms in the turbulence equation contain both first-order and second-order derivatives of the state variables, which are computed with finite differences in a local cartesian coordinate system. When the grid is smooth and locally (close to) orthogonal, then this approach is second-order accurate. It is used for both the first-order and the second-order accurate schemes.

4 Multigrid

For the solution of nonlinear systems, multigrid is one of the most powerful techniques known today. Its application to fluid flow solutions is mature and well studied [6]. The multigrid principle is to apply a simple solution technique or smoother to the solution on the finest grid and to copies of this solution on underlying coarser grids, to eliminate high-frequency and low-frequency errors, respectively.

Choice of smoother When multigrid is used to solve RANS equations, it is usually combined with time stepping. Either a time step smoother is used, or multigrid is used to compute the individual steps in an implicit time stepper. This procedure is robust and stable.

We are not going to use time stepping for two reasons. First, we think that the fastest solution techniques are obtained when the steady flow equations are solved directly with multigrid. And second, time stepping is inefficient as a smoother for flow with gravity waves, because these flows have high Reynolds numbers and transient waves that damp out very slowly in time. Therefore, we solve the discretised flow equations (1) with direct multigrid and alternating line Gauss–Seidel smoothing.

Linear multigrid The usual multigrid technique for nonlinear systems is nonlinear multigrid. Here, discretised flow equations are constructed on the coarser grids in exactly the same way as on the finest grid. Then these flow equations are smoothed, using the residual in the solution on the finer grids as a source term.

For the two-fluid RANS equations, this approach gives two problems. One, the two-fluid flow equations cannot handle large arbitrary source terms [7]. And two, for nonlinear multigrid to work well, the flow equations on the coarser grids must resemble the flow equations on the finest grid. Here, this is not the case, because the volume fraction α is discontinuous and because the Menter source terms are sensitive to the grid resolution.

We overcome these problems by switching to *linear* multigrid. The flow equations on the finest grid are linearised around the current solution and this linearised operator is copied directly to the coarser grids. These linear coarse grid operators can be solved for any source term and they always resemble the fine grid operator reasonably well. Thus, on the coarse grids, we do not compute full solutions $\mathbf{q}$ but (small) corrections. We have chosen this unusual combination of linear multigrid on the coarser grids with nonlinear line smoothing on the finest grids, because the nonlinear smoother is very robust. It can correct small unphysical solutions ($\tilde{\nu}_T < 0$ etc.) that arise from the linear coarse grid corrections.

A disadvantage of the method is that it requires much computer memory. Also, Galerkin operators can reduce the convergence rate of multigrid for convective flows ([6], Sect. 5.4). However, in practice this effect was observed but it caused no problems.

On the other hand, the method can solve flows whose features cannot be resolved accurately on coarse grids. Another advantage is that the coarse grid corrections are cheap. The linear line smoothing is less expensive than nonlinear line smoothing and the computation of the linear operator takes little extra work, as the linearisation is already needed for the line smoothing on the finest grid.

Full multigrid As initial condition, we choose $\tilde{\nu}_T$ very close to zero. Then, when the solution process is started, the turbulence intensity has to grow. Typically, we see the residual in the turbulence equation increase at first, because the turbulence intensity increases. Then, when the boundary layers have more or less developed, it starts to decrease. Experiments show that multigrid relaxation is of little use in the first stage.

Therefore, full multigrid (FMG) is essential to our method: we find the initial condition on the finest grid by computing solutions on the coarser grids, from coarse to fine. Thus, all but the first computations start at the second stage, with developed boundary layers. The time-consuming development of turbulence only occurs on the coarsest grid.

Defect correction To compute second-order accurate solutions, we use defect correction. In an iterative process, the residual of the second-order accurate discretisation is computed and used as a source term in the first-order multigrid smoother. Theoretically, this process converges to the second-order solution. In practice, defect correction does not always converge. But when the process is started from the first-order solution, a great improvement is obtained in the first iterations and the solution is usually second-order accurate after a few steps.

For efficiency, we perform the multigrid step in the defect correction with the linearised operators, even on the finest grid. These operators are not updated during the process, since they are stable and close enough to the converged solution to give good smoothing.

5 Results

As a test case, the flow in a channel with a bottom bump is computed. Experimental results for this test are given by Cahouet [1]. The flow has a Froude number $Fr = 2.05$ and a Reynolds number $Re = 1.9 \times 10^5$ (based on inflow water height). The top wall is modeled as a slip wall, the bottom is a no-slip wall. The bump has a thickness of 44% of the water height. The curvilinear grid has 128×512 cells and is compressed near the boundary layer and the water surface.

In the velocity plot (Fig. 1a), we see low-velocity regions near the leading and trailing edge of the bump and high-velocity regions above the bump and in the air region near the top wall. The turbulent viscosity (Fig. 1b) shows the boundary layer. Figure 1c gives a comparison of the second-order solution with Cahouet's measurements. Overall, the agreement is very good. The slight difference at the back of the wave may be caused by the more developed

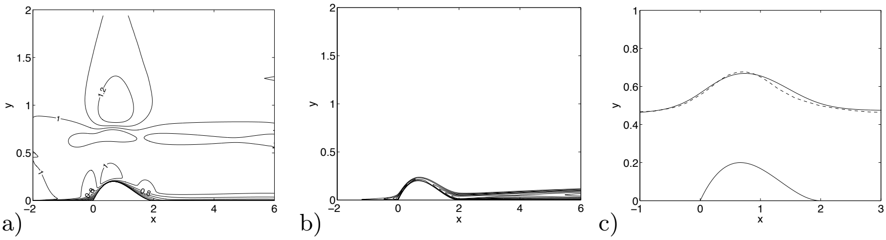

Fig. 1. Cahouet test case, $Fr = 2.05$. Speed (**a**), turbulent viscosity $\tilde{\nu}_T$ (**b**), and a comparison (**c**) of the second-order $\alpha = 0.5$ isolines (—) with Cahouet's experiment (averaged wave height, - -).

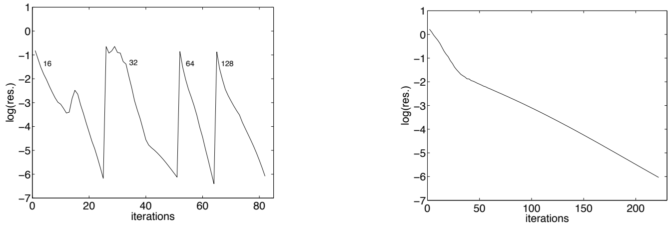

Fig. 2. Cahouet test case, convergence of the sum of the residual for multigrid (FMG on four grids, left) and line Gauss–Seidel smoothing on a single grid (right).

boundary layer in Cahouet's experiment, as Cahouet's tunnel has a longer inflow length than our domain.

The bottom bump solution is computed on four grids, the multigrid convergence is given in Fig. 2. We see bad convergence on the first two grids. Apparently the boundary layer on the second grid is so different from the first grid that boundary layer development is needed on the second grid too. The convergence on the last two grids is acceptable.

Compared with a solution on a single grid, the number of (expensive) fine-grid iterations is reduced by a factor 12. The total computation time is reduced by a factor 8. The current method does not converge so fast in difficult locations, like stagnation points, but we believe this can be solved by simple fixes: therefore, the convergence can likely be improved even further.

6 Conclusion

The computational results for the flow in a channel with a bottom bump, show that the method is efficient. Comparison with experimental results indicates that it is also accurate.

Acknowledgement. This research was supported by the Dutch government through ICT project BRICKS (http://www.bsik-bricks.nl).

References

1. Cahouet, J.: Etude numérique et expérimentale du problème bidimensionnel de la résistance de vagues non-linéaire. Technical report 185, Ecole Nationale Supérieure de Techniques Avancées, Paris (1984)
2. Dick, E., Linden, J.: A multigrid method for steady incompressible Navier-Stokes equations based on flux-difference splitting. Int. J. Num. Meth. Fluids **14**, 1311–1323 (1992)
3. Hino, T. (ed) CFD Workshop Tokyo 2005. National Maritime Research Institute, Tokyo (2005)
4. Koren, B.: A robust upwind discretization method for advection, diffusion and source terms. In: Vreugdenhil, C.B., Koren, B. (eds) Numerical Methods for Advection-Diffusion Problems. Vieweg (1993)
5. Menter, F.R.: Eddy viscosity transport equations and their relation to the $k - \varepsilon$ model. J. Fluids Engineering, **119**, 876–884 (1997)
6. Trottenberg, U., Oosterlee, C.W., Schüller, A.: Multigrid. Academic Press (2001)
7. Wackers, J., Koren, B.: A surface capturing method for the efficient computation of steady water waves. To appear in J. Comp. Appl. Math.