Paper C

An Efficient Method for the Numerical Calculation of Viscous Effects on Transient Long-Waves *

* Accepted for publication in *Coastal Engineering*.

An efficient method for the numerical calculation of viscous effects on transient long-waves

Tomas Torsvik [a,*] Philip L.-F. Liu [b]

[a] *University of Bergen, Department of Mathematics. Johannes Brunsgate 12, NO-5008 Bergen, Norway*

[b] *School of Civil and Environmental Engineering, Cornell University. Ithaca, NY, 14853, USA*

Abstract

Previous studies have shown that the Boussinesq equations can be used to calculate the instantaneous bottom shear stress induced by transient or periodic waves. The bottom friction term occurs as a convolution integral in time in the continuity equation. The exact numerical integration of a convolution integral demands large computational resources, which makes the method less useful for large scale computations. In this paper we explore how the value of the convolution integral can be estimated if we only use the values of the variables in a limited number of time steps, and discuss the accuracy and computational cost of this method.

Key words: Boussinesq equation, boundary layer, bottom stress, convolution integral, computational efficiency

1 Introduction

A correct estimate of the bottom friction effect is important when calculating the propagation of long waves in shallow water over large distances. The bottom friction effect contributes to the dissipation of wave energy, but also influences the shape and speed of the waves. The shear stress at the bottom induced by waves is important when modelling sediment transport and, in a

* Corresponding author.
Email addresses: Tomas.Torsvik@mi.uib.no (Tomas Torsvik), pll3@cornell.edu (Philip L.-F. Liu).

longer time perspective, the morphology of the coast line and the bathymetry. Examples where transient waves are important for sediment transport include tsunamis (Kobayashi and Lawrence (2004)) and long waves from high-speed vessels in confined waters (Soomere (2005)).

Traditional methods for modeling the bottom shear stress take the form of $C_f \rho |u_b| u_b$, which captures the long term average energy dissipation, and can give good results for the damping rate of periodic wave trains. This formulation does not reproduce the correct phase of the bottom shear stress, nor does it take into account the historic development of the velocity field. Because of these deficiencies, the traditional formulation is not well suited for studies of transient wave phenomena.

Recently Liu and Orfila (2004) (LO) suggested a new formulation for the bottom shear stress, and showed how this could be included in the Boussinesq equations. The performance of the model was discussed for idealized cases in LO and later in Liu et al. (2006) where a numerical model was compared with data from laboratory experiments for solitary wave damping and shoaling. These studies have shown that the new formulation is well suited for modelling the correct strength and time variation of the bottom friction effect.

The formulation of the bottom friction found by LO includes a convolution integral in time, which makes it difficult to implement efficiently in a numerical code. In order to calculate the integral directly, we need to store the entire time history of the velocity everywhere in the computational domain, and, for each time step, integrate from start to present over all spatial nodes. As the numerical model is integrated forward in time, we find that more and more effort is needed to calculate the bottom friction term, and it soon becomes the dominating load in terms of CPU time and memory for the computation. In order to use this formulation of the bottom friction term for large scale simulations, we need a method for finding an approximate value for the bottom friction term which can be calculated without retaining the entire time history of the velocity. The development of such a method is the purpose of this paper.

2 Numerical calculation of the bottom friction term

As shown in Liu et al. (2006), the Boussinesq equations can be expressed in terms of the free surface displacement, η, and the horizontal velocity, u_α,

evaluated at $z = z_\alpha$, as

$$\frac{\partial \eta}{\partial t} + \nabla \cdot ((h + \epsilon\eta)u_\alpha) + \mu^2 \nabla \cdot \left(\left(\frac{3z_\alpha^2 h + 6z_\alpha h^2 + 2h^3}{6} \right) \nabla(\nabla \cdot u_\alpha) \right) - \\ - \frac{\alpha}{\mu\sqrt{\pi}} \int_0^t \frac{\nabla \cdot u_\alpha}{\sqrt{t-\tau}}\, d\tau = \mathcal{O}(\mu^4)\,, \tag{1}$$

$$\frac{\partial u_\alpha}{\partial t} + \epsilon(u_\alpha \cdot \nabla)u_\alpha + \nabla\eta + \nabla p + \mu^2 \frac{z_\alpha^2 + 2z_\alpha h}{2} \nabla \left(\nabla \cdot \frac{\partial u_\alpha}{\partial t} \right) = \mathcal{O}(\mu^4)\,. \tag{2}$$

The dimensionless depth h is assumed constant, $\nabla = (\partial/\partial x, \partial/\partial y)$ is the two-dimensional gradient operator, and p is the pressure at the free surface. The parameters which occur in equations (1) and (2), are defined as

$$\mu = \frac{h_0'}{l_0'}\,, \qquad \epsilon = \frac{a_0'}{h_0'}\,, \qquad \text{and} \qquad \alpha^2 = \frac{\nu'}{l_0'\sqrt{g'h_0'}}\,,$$

where h_0' is the characteristic depth, a_0' is a typical wave amplitude, l_0' is a characteristic length scale and ν' is the viscosity of water. As stated in LO, the relation $\mathcal{O}(\alpha) \approx \mathcal{O}(\mu^4) \approx \mathcal{O}(\epsilon^2)$ is required for the relative magnitude of the parameters. The convolution integral in the continuity equation (1) is due to the vertical component of the rotational velocity in the bottom boundary layer. The bottom shear stress

$$\tau_b(x,t) = \frac{u_\alpha(x,0)}{\sqrt{\pi t}} + \frac{1}{\sqrt{\pi}} \int_0^t \frac{u_{\alpha,\tau}(x,\tau)}{\sqrt{t-\tau}}\, d\tau + \mathcal{O}(\mu^2) \tag{3}$$

is defined as the vertical gradient of the horizontal rotational velocity component.

The value of the convolution integral

$$A = \int_0^t \frac{\nabla \cdot u_\alpha}{\sqrt{t-\tau}}\, d\tau$$

can in principle be calculated numerically by storing the values of $\nabla \cdot u_\alpha$ for all time steps, and, for each time step, to integrate A over time for all spatial nodes. It is readily seen that this approach is costly in terms of storage and computation time, which in practice prohibits the use of this method for simulations with a large number of spatial nodes and for integration over long times. In order to use this method for large scale simulations, it is necessary to estimate the value of A based on values of $\nabla \cdot u_\alpha$ for a truncated number of time steps.

For convenience of discussion we shall focus our discussion on one-dimensional problems. Let's assume that we have obtained the values of $\nabla \cdot u_\alpha$ for the last N time steps. On an equidistant grid, discretized by Δx in space and Δt in time,

an estimate for the convolution integral can be calculated by the weighted sum

$$\widetilde{A}_i^{(k)} = \sum_{j=0}^{N-1} C_j \, \delta u_i^{(k-j)} \,, \tag{4}$$

where $\delta u_i^{(k)} = \nabla \cdot u_\alpha(i\,\Delta x, k\,\Delta t)$, and $\widetilde{A}_i^{(k)} \approx A(i\,\Delta x, k\,\Delta t)$. The weights C_j are determined by

$$\begin{aligned} C_0 &= \int_{t-\frac{1}{2}\Delta t}^{t} \frac{1}{\sqrt{t-\tau}}\, d\tau = 2\sqrt{\tfrac{1}{2}\Delta t}\,, && j = 0\,, \\ C_j &= \int_{t-(j+\frac{1}{2})\Delta t}^{t-(j-\frac{1}{2})\Delta t} \frac{1}{\sqrt{t-\tau}}\, d\tau = 2\sqrt{(j+\tfrac{1}{2})\Delta t} - 2\sqrt{(j-\tfrac{1}{2})\Delta t}\,, && j \geq 1\,. \end{aligned}$$

When we integrate forward in time, the first $N-1$ values of $\delta u_i^{(k)}$ in the sum (4) are retained in memory and used to calculate $\widetilde{A}_i^{(k+1)}$. We may retain the discarded part of (4) in a residual term

$$R_i^{(k)} = \widetilde{A}_i^{(k)} - \sum_{j=0}^{N-2} C_j \, \delta u_i^{(k-j)} \,,$$

and use this value to improve our estimate of $\widetilde{A}_i^{(k+1)}$ by including a correction term

$$\widetilde{A}_i^{(k+1)} = \sum_{j=0}^{N-1} C_j \, \delta u_i^{(k+1-j)} + C_R \, R_i^{(k)} \,, \tag{5}$$

where the value of the residual coefficient C_R is as yet undetermined.

The discarded values of $\widetilde{A}$ accumulate in the residual term R as we integrate forward in time. Starting from time step k, the accumulated residual term after $k+m$ time steps is

$$R_i^{(k+m)} = \sum_{j=0}^{m} C_R^j \, C_{N-1} \, \delta u_i^{(k+m-(N-1)-j)} \,, \tag{6}$$

which is a power series in C_R. By choosing a suitable value for the residual coefficient C_R, we can to some extent compensate for the loss in memory caused by replacing A with the truncated sum $\widetilde{A}$. The value of C_R should be within the range

$$\frac{C_N}{C_{N-1}} = \frac{\sqrt{N+\frac{1}{2}} - \sqrt{N-\frac{1}{2}}}{\sqrt{N-\frac{1}{2}} - \sqrt{N-\frac{3}{2}}} \leq C_R < 1\,. \tag{7}$$

We note that the lower bound on C_R depends only on N, and is independent of the time discretization Δt. If we use

$$C_R = \frac{C_N}{C_{N-1}} \,,$$

and substitute the power series (6) for R in (5), we see that the first term in the power series will have the correct weight C_N, while the weights on any subsequent terms will be smaller than the correct value since

$$\left(\frac{C_N}{C_{N-1}}\right)^m C_{N-1} < C_{(N-1)+m}$$

for all $m \geq 2$. We may want to use a larger value for C_R, but we must always ensure that the accumulated residual terms diminish faster than the exact solution for large times. It can be shown that

$$\lim_{m\to\infty} \frac{C_R^m\, C_{N-1}}{C_{(N-1)+m}} = \lim_{m\to\infty} C_R^m \frac{\sqrt{N-\frac{1}{2}} - \sqrt{N-\frac{3}{2}}}{\sqrt{m+N-\frac{1}{2}} - \sqrt{m+N-\frac{3}{2}}} = 0\,, \qquad C_R < 1\,,$$

but in practice we do not integrate over infinite times, so values of C_R close to 1 should be avoided. Instead of choosing an arbitrary value of C_R within the appropriate range, we may determine this value based on the average reduction of the weights over a given number of time steps, according to

$$C_R^{(s)} = \sum_{j=1}^{s} \frac{1}{s} \frac{C_{N-1+j}}{C_{N-2+j}}\,. \tag{8}$$

The value of $C_R^{(s)}$ increases with increasing values of s, but the sum always remains within the valid range.

m	$C_R^{(1)}$	$C_R^{(5)}$	$C_R^{(10)}$	$C_R^{(20)}$
	0.8647	0.9064	0.9295	0.9507
0	0.0000	0.0482	0.0749	0.0994
10	0.2709	0.1041	0.1057	0.2287
20	0.4519	0.2659	0.1538	0.2088
30	0.5497	0.3823	0.2525	0.2045

Table 1
Error of the weights, E_w, for $N = 4$, and for different residual coefficients C_R and time steps m.

Errors of the weights may be calculated from

$$E_w^{(s)} = \frac{\left(\sum_{j=0}^{m} (C_{m+N} - C_R^{(s)\,m}\, C_{N-1})^2\right)^{\frac{1}{2}}}{\left(\sum_{j=0}^{m} C_{m+N}{}^2\right)^{\frac{1}{2}}}\,.$$

Results for $N = 4$ and for different values of C_R and m are shown in Table 1 and Figure 1. As expected, $C_R^{(1)} = 0.8647$ provides the best estimate for small m, but the error $E_w^{(1)}$ increases rapidly with increasing m. With the residual

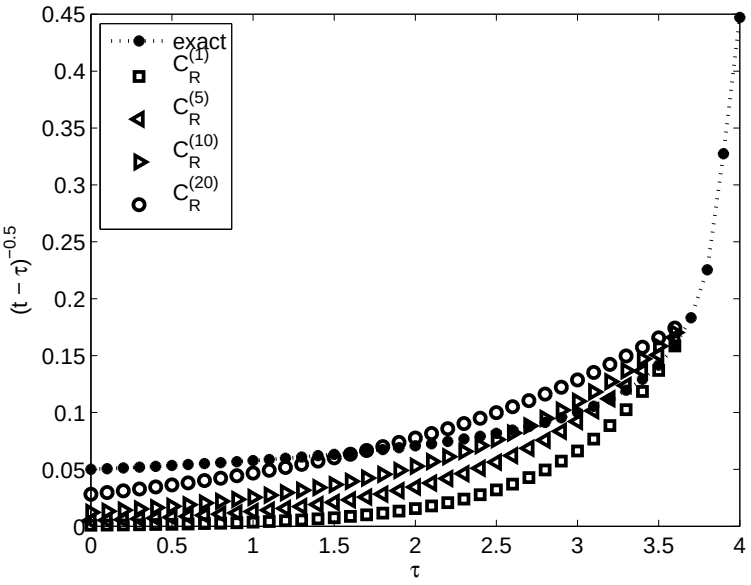

Fig. 1. Weights used to calculate the value of the convolution integral, comparing weights for the exact formulation A with weights for the truncated sum $\widetilde{A}$, for $N = 4$, with different values of C_R.

coefficient $C_R^{(20)} = 0.9507$, we get a large error for small values of m, but the error decreases from a maximum value of $E_w^{(20)} = 0.2287$ as m increases beyond $m = 10$. The error introduced by replacing A with $\widetilde{A}$ involves not only the error of the weights, but also the history of δu. Hence we can not expect to find an optimal value of C_R independently of the problem at hand. In the next section we use results from simulations to determine optimal values of C_R for two test cases.

3 Performance of the algorithm on test cases

The governing equations (1) and (2) have been implemented in the high order, depth integrated COULWAVE model (Lynett (2002), Lynett et al. (2002)). This model utilizes a predictor-corrector time-stepping scheme which is accurate to $O(\Delta t^4)$, and which stores values for the physical variables for the 4 most recent time steps. In the COULWAVE model it is natural to choose $N = 4$, and this value is used in the following examples unless a different value is specified.

Since the value of C_R mainly influence the memory of the history of the velocity field, it is natural to let the assignment of a value to the residual coefficient

be based on a typical time scale for the waves, such as the wave period T. We use the same averaging procedure introduced in eq. (8), but instead of specifying the number of time steps directly, we specify the time for averaging as a fraction of T, and let the program decide how many time steps to include based on the time discretization Δt.

3.1 Idealized cases

Soliton solution

The soliton solution has been tested for five test cases (see Table 2). The soliton does not have a well defined wave length or period. A length scale L_0 was defined as the distance between points where $\eta = 0.01\, a_0$, upstream and downstream of the wave crest. The speed of the wave is $c = \sqrt{g(h + a_0)}$, and this provides a time scale

$$T = \frac{L_0}{c}\,.$$

For each of the test cases we calculated the propagating wave using both the full convolution integral formulation and estimates $C_R^{(0.10T)}$, $C_R^{(0.15T)}$, $C_R^{(0.20T)}$, and $C_R^{(0.25T)}$. The results for the best estimates are shown in Table 3. The error in the amplitude relative to the true value is of $\mathcal{O}(10^{-3})$. The optimal time over which to average in order to determine C_R is within the range $0.15T - 0.20T$, depending on the value of the parameter a_0/h_0.

	h_0	a_0	L_0	c	T
A1	1.00 m	0.0995 m	22.2923 m	3.2842 m/s	6.7877 s
A2	1.00 m	0.0499 m	32.2137 m	3.2093 m/s	10.0377 s
A3	0.15 m	0.0404 m	2.1978 m	1.3667 m/s	1.6081 s
A4	0.15 m	0.0136 m	3.6261 m	1.2670 m/s	2.8619 s
A5	0.15 m	0.0070 m	4.9101 m	1.2410 m/s	3.9565 s

Table 2
Parameters used in the test cases for a solitary wave.

Figure 2 shows a solitary (A4 in Table 3) wave after it has propagated a distance $\Delta x \approx 420 h_0$. The estimate for $C_R = 0.9566$ is close to the exact solution at the location of maximum wave amplitude, whereas the wave amplitude is overestimated for smaller C_R and slightly underestimated for larger C_R. All estimates fail to reproduce the correct surface displacement immediately downstream of the solitary wave. Figure 3 shows the bottom friction term from (1), but with dimensional variables. The bottom friction term computed with the truncated sum (5) overestimates extremal values, for the minimum because the residual coefficient is larger than for the exact computation, and

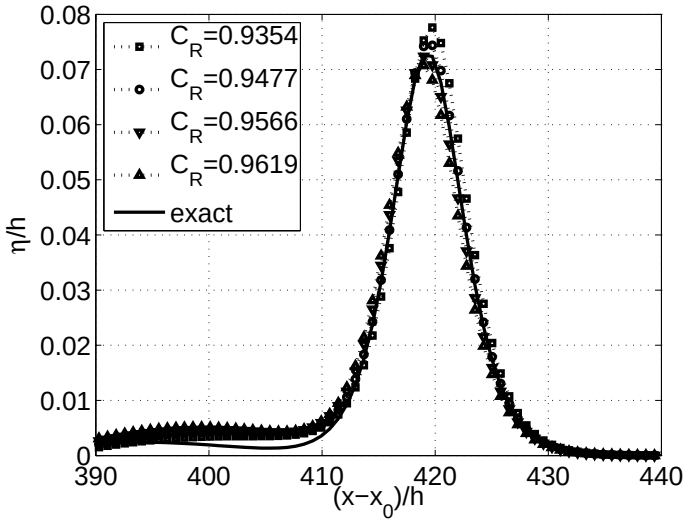

Fig. 2. Free surface displacement for the solitary wave simulation at $t = 50.0$ s. Numbers on the x-axis indicates the distance relative to the position of the soliton crest at $t = 0.0s$.

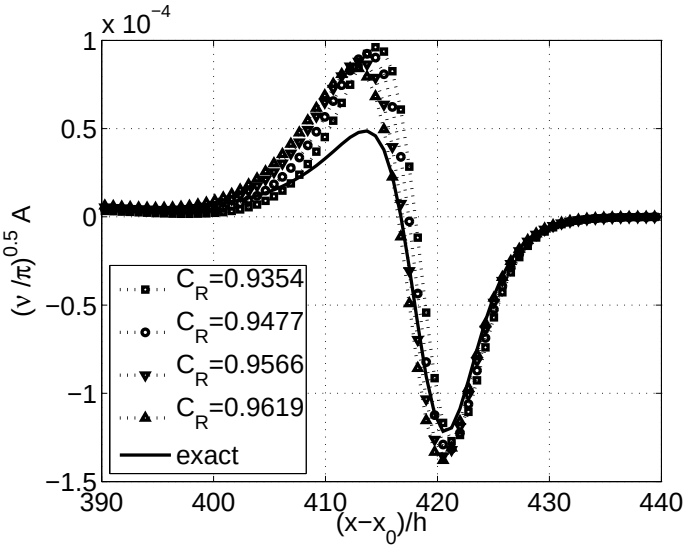

Fig. 3. Bottom friction induced by the solitary wave.

for the maximum because of the loss of memory. The best result in Figure 2 is obtained for $C_R = 0.9566$, and we see in Figure 3 that for this value the estimate reproduces the phase of the bottom friction correctly.

	a_0/h_0	C_R	T_R	error (amp)
A1	0.0955	0.9545	1.3575 s	$-1.64 \cdot 10^{-3}$
A2	0.0499	0.9647	1.5056 s	$1.40 \cdot 10^{-3}$
A3	0.2693	0.9379	0.3216 s	$4.94 \cdot 10^{-4}$
A4	0.0907	0.9566	0.5724 s	$-7.79 \cdot 10^{-3}$
A5	0.0467	0.9533	0.5935 s	$9.02 \cdot 10^{-3}$

Table 3
Best estimates for the solitary wave.

	h_0	a_0	λ	c	T
B1	1.00 m	0.0200 m	32.21 m	3.1321 m/s	10.3487 s
B2	1.00 m	0.0100 m	32.21 m	3.1321 m/s	10.3487 s
B3	1.00 m	0.0100 m	20.00 m	3.1321 m/s	6.4889 s
B4	0.15 m	0.0020 m	5.00 m	1.2131 m/s	4.1461 s

Table 4
Parameters used in the test cases for periodic waves.

Periodic waves

Four test cases with periodic waves have been examined (see Table 4). A periodic wave train with period T is generated by a wave maker at a particular location, and propagate into a region where the initial surface displacement is zero. Simulations of the same test cases were performed using both the full convolution integral and estimates $C_R^{(0.05T)}$, $C_R^{(0.10T)}$, and $C_R^{(0.15T)}$. The results for the best estimates are shown in Table 5. In all these test cases the parameter a_0/h_0 is less than for the soliton test cases, and the optimal time average is near $0.10T$.

Figure 4 shows a periodic wave train (B1 in Table 5) in the region $350h_0-450h_0$ downstream of the wave maker, and Figure 5 shows details of the crest at $\Delta x = 440h_0$. All results are within a reasonable range of the exact solution when we use the truncated sum (5) to calculate the bottom friction term. The bottom friction term is shown in Figure 6. Again we see that the truncated sum overestimates the extremal values, but the phase of the wave train is reproduced correctly.

Increasing N for improved estimate $\widetilde{A}$

The accuracy of the estimate $\widetilde{A}$ can be improved by increasing the value of N, at the cost of storing the velocity field for more time steps. In order to quantify

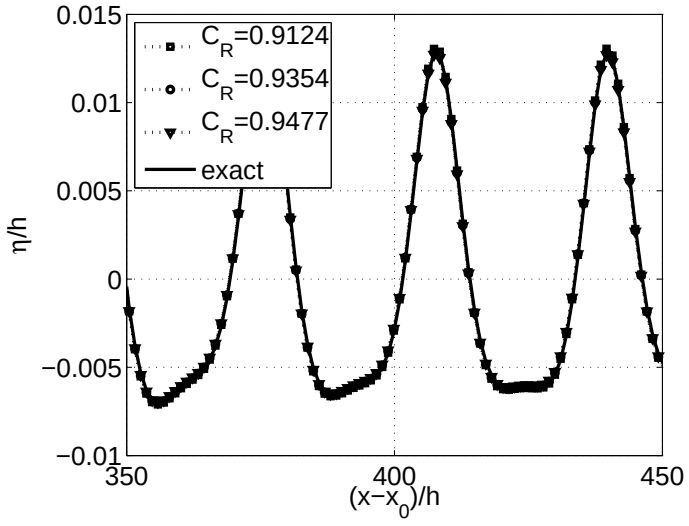

Fig. 4. Free surface displacement for a periodic wave train at $t = 149.1$ s. Numbers on the x-axis indicates the distance relative to the position of the wave maker.

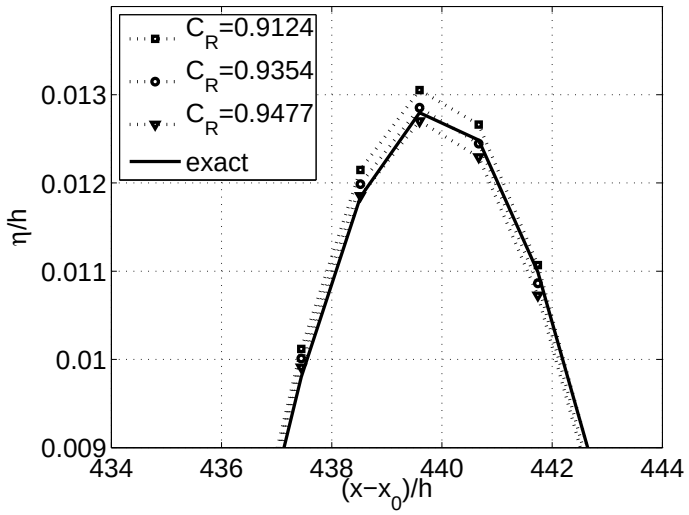

Fig. 5. Periodic waves. Detail near the crest of a wave.

the improvement, we repeat the simulations for the solitary wave and periodic wave test cases, using $N = 8$ and $N = 16$. The lower bound for C_R increases with N according to (7). For $N = 4$ the lower bound is $C_R^{(1)} = 0.8647$, whereas for $N = 8$ and $N = 16$, the lower bounds are $C_R^{(1)} = 0.9353$ and $C_R^{(1)} = 0.9682$, respectively. In each of the simulations, C_R is decided according to the optimal time average found in the in the previous sections. The parameters for the

	a_0/h_0	C_R	T_R	error (amp)
B1	0.0200	0.9354	1.0349 s	$1.48 \cdot 10^{-3}$
B2	0.0100	0.9354	1.0349 s	$1.10 \cdot 10^{-3}$
B3	0.0100	0.9354	0.6489 s	$1.10 \cdot 10^{-3}$
B4	0.0133	0.9295	0.4146 s	$-1.16 \cdot 10^{-3}$

Table 5
Best estimates for periodic waves.

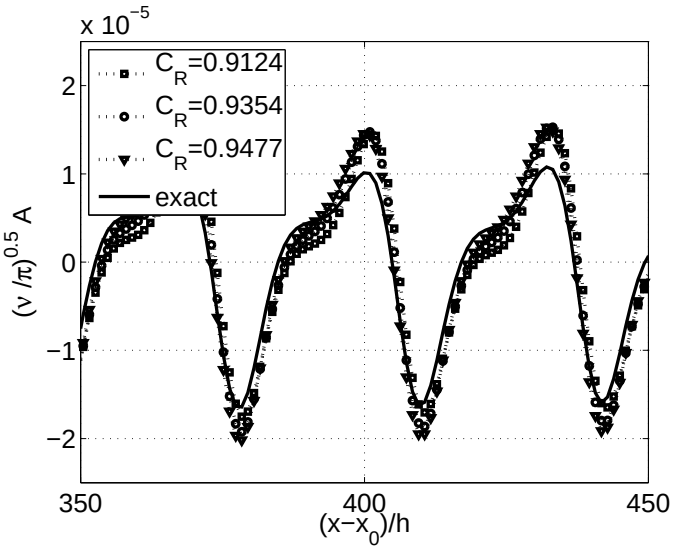

Fig. 6. Bottom friction induced by the periodic wave train.

simulations, and the deviation in L^2 norm relative to the exact solution, are tabulated in Table 6.

Results for both test cases are improved by increasing N, but the rate of convergence is larger for the periodic wave simulation than for the solitary wave simulation. Figures 7 and 8 show the surface displacement and bottom friction for the solitary wave. Increasing the value of N clearly improves the estimate $\widetilde{A}$, but the error in the trailing wave diminish only slightly. These results suggest that the method provides accurate results for the bottom friction in the vicinity of the largest amplitude waves, but a high value of N is required for the accurate computation of waves with smaller amplitudes.

Optimal choice of C_R

The optimal choice of C_R, for a given value of N, depends on the wave celerity, amplitude, and period. In practical applications waves of different amplitudes

	solitary wave		periodic wave	
N	C_R	Error (L^2)	C_R	Error (L^2)
4	0.9566	$1.6924 \cdot 10^{-2}$	0.9354	$2.2784 \cdot 10^{-2}$
8	0.9701	$1.1100 \cdot 10^{-2}$	0.9593	$1.3392 \cdot 10^{-2}$
16	0.9806	$8.1246 \cdot 10^{-3}$	0.9758	$5.9794 \cdot 10^{-3}$

Table 6
Parameters and errors for the test cases, using different values for the number of time steps N retained for the velocity field.

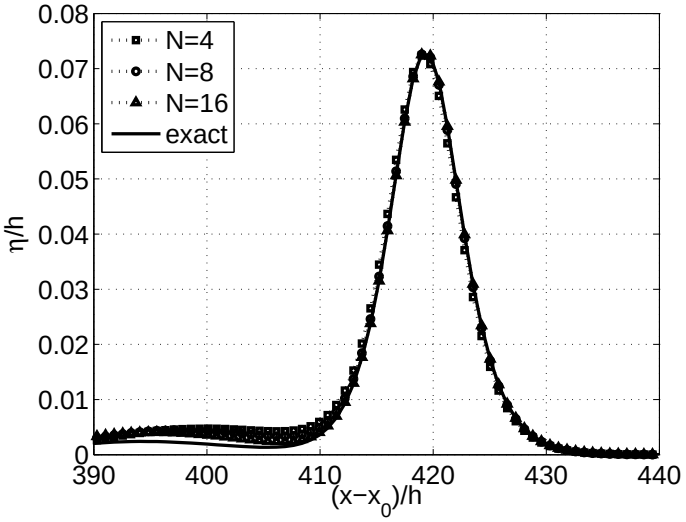

Fig. 7. Free surface displacement for the solitary wave simulation at $t = 50.0$ s. Numbers on the x-axis indicates the distance relative to the position of the soliton crest at $t = 0.0s$. Results for $N = 4$, 8 and 16.

and periods may occur simultaneously at the same location, and the optimal choice of C_R may not be equal for all wave groups. One strategy for deciding C_R would be to determine a time scale T for the largest amplitude wave group, and calculate C_R according to (8) with s as the number of time steps required to cover a certain fraction of T. In the examples above, this fraction is $0.10T$ for the periodic waves and $0.15T - 0.20T$ for the solitary wave.

The value of C_R does not have to be a single constant value throughout the computation. The accumulated residual term (6) at a given point is independent of the accumulated residual term in neighbouring points, which allows us to specify C_R independently for all computational points. This may be useful for computations including a variable bathymetry, where the wave amplitude and celerity is often altered due to depth variation.

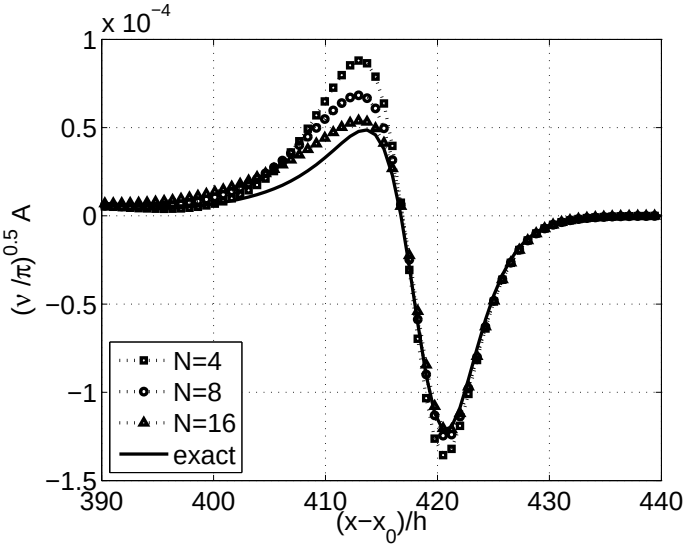

Fig. 8. Bottom friction induced by the solitary wave. Results for $N = 4$, 8 and 16.

Analysis of profile for different simulations

Profiling of the simulations showed that the average time spent calculating the bottom friction amounted to 89% of the total execution time, for both the soliton and sinusoidal waves, when the complete convolution integral was computed for each time step. Compared to simulations without bottom friction, the (wall-clock) execution time increased by 1105% and 970% for the soliton and sinusoidal simulations, respectively. Using the estimated value of the convolution integral increased the execution time by about 20%, when compared with simulations without bottom friction.

3.2 Pressure disturbance propagating in a narrow channel

The estimate for the bottom friction term can be used in large scale simulations. As an example, we have reproduced one of the test cases examined by Ertekin et al. (1986). The implementation of the pressure disturbance in the COULWAVE model follows Liu and Wu (2004). A pressure disturbance is moving to the left at constant speed in a rectangular channel with depth $h_0 = 1.0m$ and width $2b = 8h_0$. The side walls at $y = \pm 4h_0$ are closed, and lateral boundaries at $x_0 + 50h_0$ and $x_0 - 250h_0$, where x_0 is the initial position of the disturbance, have sponge layers which damp waves as they approach the boundary. A wave gauge is included at the position $(x_0 - 200h_0, 0)$. The pressure disturbance $p(x + Ft, y)$ is defined inside a rectangle $0 \leq x + Ft \leq L/2$, $0 \leq y \leq B/2$, by

$$p(x + Ft, y) = P_m f(x + Ft) q(y)$$

with

$$f(x+Ft) = \begin{cases} 1 & ,0 \le x+Ft < \frac{1}{2}\alpha L\,, \\ \cos^2\left[\frac{\pi(x+Ft-1/2\alpha L)}{(1-\alpha)L}\right] & ,\frac{1}{2}\alpha L \le x+Ft \le \frac{1}{2}L\,. \end{cases}$$

$$q(y) = \begin{cases} 1 & ,0 \le y < \frac{1}{2}\beta B\,, \\ \cos^2\left[\frac{\pi(y-1/2\beta B)}{(1-\beta)B}\right] & ,\frac{1}{2}\beta B \le y \le \frac{1}{2}B\,. \end{cases}$$

using parameters

$$\frac{B}{2b} = \frac{1}{2}\,, \quad \frac{L}{B} = 2\,, \quad P_m = 0.1\,, \quad \alpha = 0.7\,, \quad \text{and} \quad \beta = 0.4$$

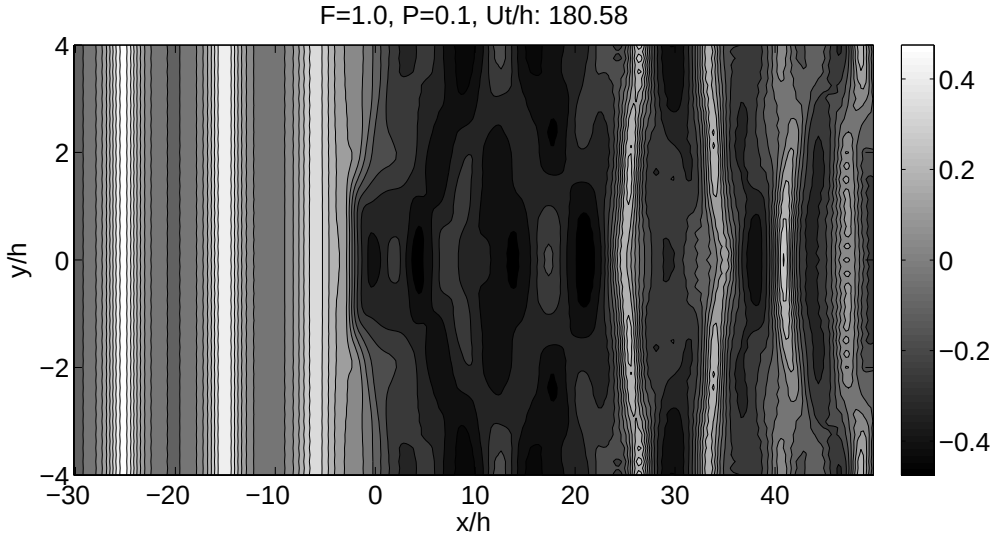

Fig. 9. Wave pattern due to a moving pressure disturbance, located with centre at $(x, y) = (0, 0)$. Contour lines are drawn at vertical increments of $\Delta z = 0.05\,h$.

Two simulations were performed for Froude number $F = 1.0$, using the bottom friction term with $C_R = 0.9827$ in one simulation. The resulting wave pattern, with bottom friction, is shown in Fig.9. The bottom shear stress calculated according to eq. (3), are shown in Figs. 10 and 11 for the lateral ($\tau_b \mathbf{i}$) and transverse ($\tau_b \mathbf{j}$) component, respectively. The bottom friction clearly has an effect on the wave pattern, as seen for the comparison in Fig. 12.

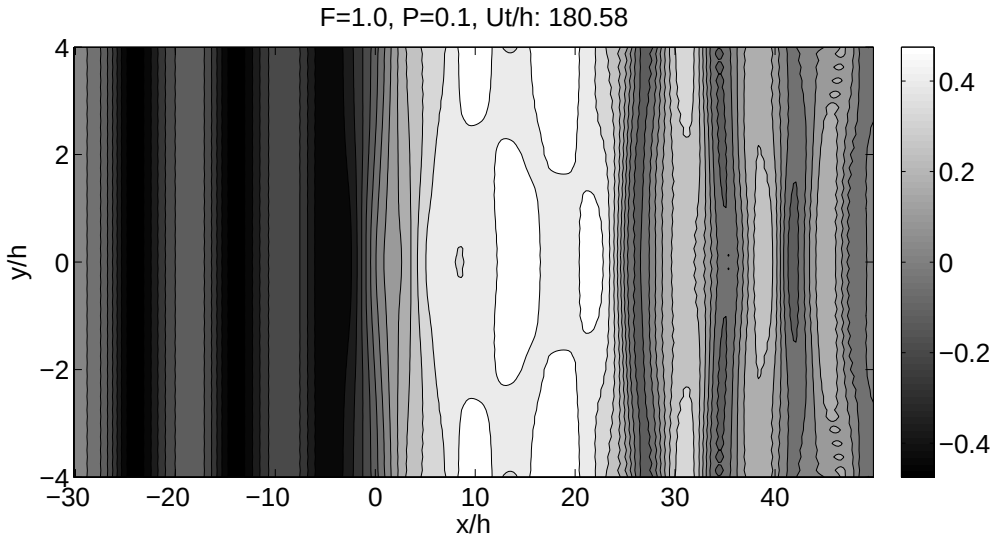

Fig. 10. Bottom shear stress (x-component). Contour lines are drawn at vertical increments of $\Delta\tau = 0.05$.

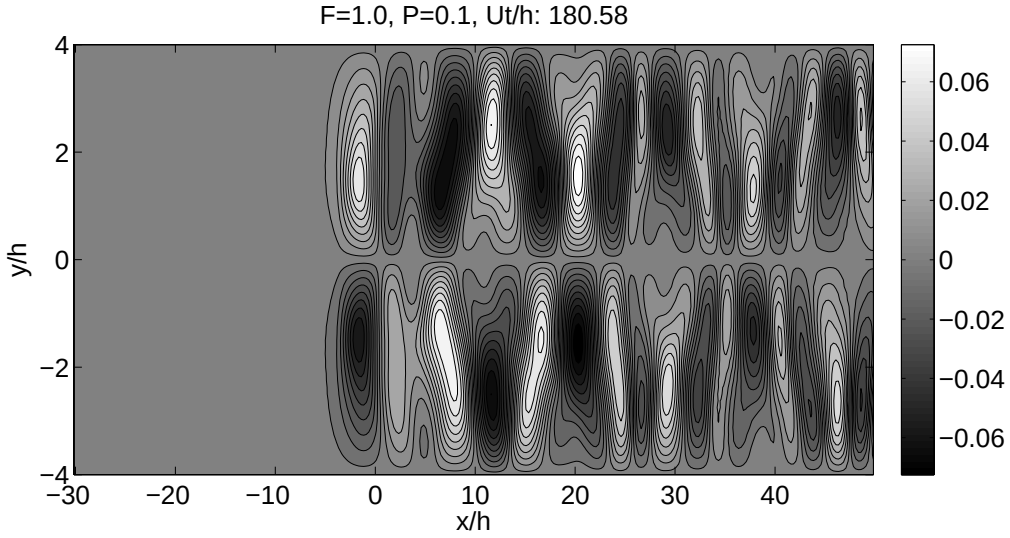

Fig. 11. Bottom shear stress (y-component). Contour lines are drawn at vertical increments of $\Delta\tau = 0.005$.

4 Concluding remarks

In this article we have demonstrated how the convolution integral which occurs in the bottom friction term, can be estimated based on a limited number of time steps. The method is accurate in the vicinity of large amplitude waves, provided some care is taken when deciding on a value for the residual coeffi-

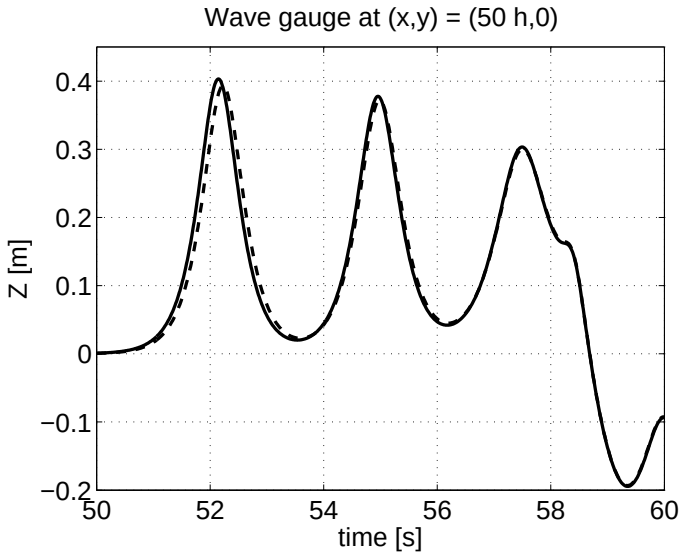

Fig. 12. Comparison of results with bottom friction (dash) and without bottom friction (solid).

cient C_R. Errors occurring in the vicinity of smaller amplitude waves can be reduced by storing the velocity field for a large number of time steps. The computational effort required to calculate the estimate of the integral is a fraction of the effort required for calculating the convolution integral directly. Using this method, it is possible to run large scale simulations with an accurate model for the bottom shear stress.

5 Acknowledgments

This work has been supported by NSF research grant (ITR program, Fluid Dynamics and Physical Oceanography programs) to Cornell University. Tomas Torsvik receives support from the Research Council of Norway through a grant to the program "Modelling of currents and waves for sea structures". The authors are grateful to the reviewers for their helpful comments and suggestions.

References

Ertekin, R. C., Webster, W. C., Wehausen, J. V., 1986. Waves caused by a moving disturbance in a shallow channel of finite width. J. Fluid Mech. 169, 275–292.

Kobayashi, N., Lawrence, A. R., 2004. Cross-shore sediment transport under

breaking solitary waves. Journal of Geophysical Research 109 (C3), Art. No. C03047.

Liu, P. L.-F., Orfila, A., 2004. Viscous effects on transient long-wave propagation. J. Fluid Mech. 520, 83–92.

Liu, P. L.-F., Simorra, G., Vandever, J., Orfila, A., 2006. Experimental and numerical investigation of viscous effects on solitary waves propagation ain a wave tank. Coastal Engineering 53 (2-3), 181–190.

Liu, P. L.-F., Wu, T.-R., 2004. Waves generated by moving pressure disturbance in rectangular and trapezoidal channels. Journal of Hydraulic Research 42 (2), 163–171.

Lynett, P., 2002. A multilayer approach to modeling generation, propagation and interaction of water waves. Ph.D. thesis, Cornell University.

Lynett, P., Wu, T.-R., Liu, P. L.-F., 2002. Modeling wave runup with depth-integrated equations. Coastal Engineering 46 (2), 89–107.

Soomere, T., 2005. Fast ferry traffic as a quantitively new forcing factor of environmental processes in non-tidal sea areas: A case study in tallinn bay, baltic sea. Environmental Fluid Mechanics 5, 293–323.