

The Application of Domain Decomposition to Time-Domain Computations of Nonlinear Water Waves with a Panel Method

P. C. A. De Haas* and P. J. Zandbergen†

*Delft Hydraulics, P.O. Box 177, 2600 MH Delft, The Netherlands; and †Faculty of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Received September 21, 1995; revised July 2, 1996

In this paper an iterative domain decomposition method for the solution of Laplace's equation is described and its effectiveness in time-domain computations of nonlinear water waves with a panel method is investigated. An important aspect of these computations is the varying shape of the free surface. The convergence of the iterative method is fast and leads to a speedup of the computations in the aforementioned application. The domain decomposition method gives a considerable reduction of memory requirements. Furthermore, it lends itself naturally for parallel computing. © 1996 Academic Press, Inc.

1. INTRODUCTION

For the solution of many large-scale problems, advanced methods are required to solve problems from mathematical physics with the existing computational resources. Domain decomposition is a widely addressed method to tackle problems with elliptic field equations.

For this reason various kinds of domain decomposition methods have received a lot of attention in the past 10 years. In these methods the elliptic equation is solved using subdomains in such a way that the solution in the subdomains corresponds with the sought solution of the original domain. The efficiency of the method strongly depends on the problem and on the applied numerical method. One property that domain decomposition methods all have in common is the suitability for parallel computing. A comprehensive impression of "domain decomposition" is presented in, e.g., [6].

Here we present the application of a domain decomposition method to the computation of propagating nonlinear water waves. A popular method for solving this type of problem is based on a time marching scheme combined with a boundary element method (BEM) for the elliptic field equation (Laplace's equation for the velocity potential). These methods are very suitable for this purpose because they only require a discretization of the boundary of the fluid domain. Compared with field discretization methods, the advantages are a much smaller amount of grid points and a natural description of the position of the free surface which varies in time.

Nevertheless, at present there is still a need to reduce the computational costs of a BEM. Moreover, the straightforward application of a BEM requires a computational effort that depends approximately quadratically on the size of the computational domain. This implies that for large-scale wave problems special numerical techniques are needed. A technique suitable to solve these problems is domain decomposition.

The domain decomposition method that we will describe here consists of a division of the computational domain into subdomains and of an iterative (coupling) procedure which generates a sequence of boundary conditions on the interfaces between the subdomains. This sequence converges to conditions corresponding to the solution of the original problem. The solution of the decomposed problem is equal to the solution of the original problem.

The idea of working with a sequence of boundary conditions originates from Schwarz [16]. For a decomposed Poisson problem he analyzed the convergence behaviour of the solutions generated by a method using overlapping subdomains. Just recently the technique has been applied to Poisson problems with nonoverlapping subdomains. Bourgat *et al.* [3] and Marini and Quarteroni [12] gave a variational formulation of the decomposed problem which could be translated directly into a finite element approach. Funaro *et al.* [7] gave an analysis of an iterative procedure for the Poisson problem on rectangular domains and implemented it in a spectral collocation method.

The contents of this paper is as follows. Section 2 gives a description of the mathematical and our numerical model for nonlinear water waves. In Section 3 the iterative procedure of the domain decomposition method is described and in Section 4 it is analyzed for some typical geometrical configurations related to nonlinear wave problems. Section 5 deals with the implementation of the domain decomposition method in the time-domain computation of propagating nonlinear water waves. Three examples will be given. The first one shows results on efficiency for a problem with an even bottom. The second example shows the application to a problem with an underwater bar and the third example

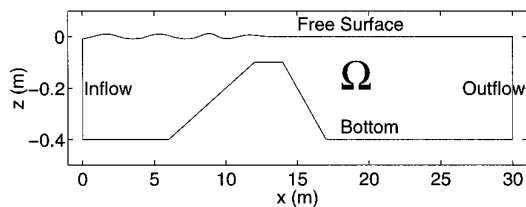


FIG. 1. Definition of the fluid domain and its boundaries.

shows that it is possible to handle large-scale wave problems with domain decomposition. Finally some conclusions will be drawn in Section 6.

2. MATHEMATICAL MODEL AND NUMERICAL METHOD

2.1. Description

In the mathematical model for nonlinear water waves considered here, the motion of the water is described by a potential flow; the water is assumed to be incompressible and inviscid and the flow is irrotational. This so-called classical water wave problem is described by the field equation

$$\Delta \phi = 0, \quad \mathbf{x} \in \Omega, \quad (1)$$

for the velocity potential ϕ and the boundary conditions

$$\left. \begin{aligned} \frac{\partial \phi}{\partial t} + \frac{1}{2} (\nabla \phi)^2 + gz &= -\frac{p}{\rho} \\ \frac{D\mathbf{x}}{Dt} &= \nabla \phi \end{aligned} \right\}, \quad \mathbf{x} \in \partial\Omega_{FS} \quad (2)$$

for the free surface and

$$\frac{\partial \phi}{\partial n} = 0, \quad \mathbf{x} \in \partial\Omega_B \quad (3)$$

for the bottom. The various boundary parts of the fluid domain Ω are indicated in Fig. 1 for a two-dimensional configuration. ρ indicates the density of water and p is the atmospheric pressure at the free surface; p is usually taken equal to zero. Equation (2) contains the dynamic and kinematic conditions at the free surface describing the wave motion and (3) describes the impermeability of the bottom.

In order to describe a finite part of the fluid domain which is of interest, the domain is truncated in the horizontal direction by vertical boundaries and wave generating or wave absorbing boundary conditions are imposed there.

Since the interest is focused mainly on the evolution of

the free surface in time and because the time-dependence comes in through the boundary conditions only, the transformation of Laplace's equation to a boundary integral equation is very useful. This transformation can be made using Green's identity. For points \mathbf{x} on the boundary, we then find

$$\begin{aligned} \frac{\vartheta(\mathbf{x})}{2\pi} \phi(\mathbf{x}) &= \int_{\partial\Omega} \left(\frac{\partial \phi}{\partial n}(\boldsymbol{\xi}) G(\mathbf{x}; \boldsymbol{\xi}) \right. \\ &\quad \left. - \phi(\boldsymbol{\xi}) \frac{\partial G}{\partial n}(\mathbf{x}; \boldsymbol{\xi}) \right) d\Omega_{\boldsymbol{\xi}}, \quad \mathbf{x} \in \partial\Omega, \end{aligned} \quad (4)$$

where G is the Green's function $\ln(|\mathbf{x} - \boldsymbol{\xi}|)/2\pi$ in 2D and $1/(4\pi|\mathbf{x} - \boldsymbol{\xi}|)$ in 3D. \int denotes the finite part of the integral in the sense of Hadamard. $\vartheta(\mathbf{x})$ is the interior angle of $\partial\Omega$ at \mathbf{x} ; if \mathbf{x} is on a smooth part of $\partial\Omega$, $\vartheta(\mathbf{x}) = \pi$. See Jaswon and Symm [10] for an extensive treatment of integral equations and potential problems.

In our numerical method the boundary $\partial\Omega$ is approximated by N panels, each represented by one collocation point \mathbf{x}_i situated near the middle of the panel. The method is split into two parts. In the time-dependent part both the potential in the collocation points on the free surface and the positions of the collocation points itself are integrated in time with a fourth-order Runge–Kutta method using the dynamic and kinematic boundary conditions. In order to use the time derivatives in these equations, spatial derivatives of the potential have to be found from the solution of Laplace's equation.

Laplace's equation is solved by imposing boundary integral equation (4) in every collocation point \mathbf{x}_i . The boundary integral equation is discretized by using higher order approximations of the boundary shape and for the variations in ϕ and $\partial\phi/\partial n$. It can be represented by

$$\frac{1}{2} \phi(\mathbf{x}_i) = \sum_{j=1}^N \left[C_s^{i,j} \frac{\partial \phi}{\partial n}(\mathbf{x}_j) + C_d^{i,j} \phi(\mathbf{x}_j) \right], \quad i = 1, \dots, N,$$

where $C_s^{i,j}$ and $C_d^{i,j}$ are the so-called source and dipole coefficients, respectively. In the Laplace problem, the free surface represents a Dirichlet boundary since ϕ is known from the time integration, the bottom represents a Neumann boundary because $\partial\phi/\partial n = 0$, and the type of the lateral boundaries depends on the imposed conditions there. Substitution of ϕ for Dirichlet boundaries and $\partial\phi/\partial n$ for Neumann boundaries results in a system

$$\mathbf{A}\mathbf{X} = \mathbf{B}$$

of N linear equations in exactly N unknowns, which can be solved using direct methods (e.g., Gaussian elimination) or iterative methods (e.g., conjugate gradient type). The

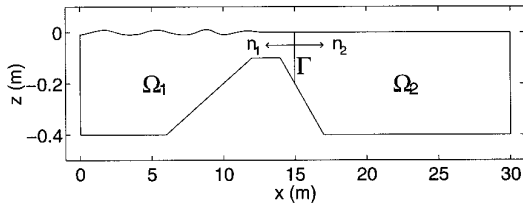


FIG. 2. Decomposed computational domain.

solution yields $\partial\phi/\partial n$ for the Dirichlet boundaries and ϕ for the Neumann boundaries. $\nabla\phi$ can be determined from $\partial\phi/\partial n$ and the tangential variation of ϕ .

The method is based on the work of Romate [14] and Romate and Zandbergen [15] for the simulation of three-dimensional nonlinear water waves. It has been developed further by Broeze (see [4, 5]), who also implemented the two-dimensional method described here. Furthermore, it has led to the development of a two- and a three-dimensional method for nonlinear wave–ship interactions (see Van Daalen [17] and Berkvens [2]).

2.2. Computational Effort

Boundary element methods are widely applied techniques for the computation of propagating water waves. Advantages in comparison with field discretizations are a reduction in the number of discretization points and a straightforward description of the free surface. Computational efforts are, however, shifted towards high computational costs in determining the source and dipole coefficients and solving a system of linear equations whose representing matrix is full. The required computation time is approximately quadratic in the number of collocation points of which a big part is consumed by the determination of the influence coefficients. But also the solution of the system of equations becomes more difficult as the number of collocation points increases. In a time-domain simulation these expensive computations have to be made every time-step. Another problem is the size of required memory which depends quadratically on the number of collocation points.

A domain decomposition method in which the computational area is divided into separate subdomains then seems a natural approach for these computations. The computational effort per subdomain is small. The efficiency of the domain decomposition method is determined by the computational effort which is required to reach equivalence of the decomposed problem with the original problem.

3. DOMAIN DECOMPOSITION

3.1. Description

We consider Laplace's equation on a domain Ω with prescribed boundary conditions. As illustrated in Fig. 2 we

decompose Ω into the subdomains Ω_1 and Ω_2 which are separated by an interface Γ . To get a unique definition of the normal derivative on the interface we choose

$$\mathbf{n} = \mathbf{n}_2 = -\mathbf{n}_1, \quad \mathbf{x} \in \Gamma.$$

We impose an arbitrary initial Dirichlet condition φ on Γ for both subdomains. If φ equals the exact solution of Laplace's equation on the original domain at the interface location ($\varphi = \phi|_{\Gamma}$), then the solutions ϕ_1 and ϕ_2 in the respective subdomains equal the solution ϕ of the original domain. Also the gradients $\partial\phi_1/\partial n$ and $\partial\phi_2/\partial n$ will then be continuous over the interface. Because of the uniqueness of the solution of Laplace's equation any other Dirichlet condition on the interface will lead to a discontinuity in the gradient on the interface. In the same way will any Neumann condition imposed on the interface and different from $\psi = \nabla\phi|_{\Gamma} \cdot \mathbf{n}$, lead to a discontinuity in the potential itself over the interface.

The right boundary conditions are found by generating a sequence of boundary conditions such that the discontinuity in the computed variable ($\partial\phi_1/\partial n|_{\Gamma} - \partial\phi_2/\partial n|_{\Gamma}$ if a Dirichlet condition is imposed and $\phi_1|_{\Gamma} - \phi_2|_{\Gamma}$ for an imposed Neumann condition) converges to zero. There are different ways to construct such a sequence and the procedure which will be used in this paper is the so-called D/D-N/N scheme:

0. Choose an initial Dirichlet condition $\varphi^{(k)}$, $k = 0$.
1. Solve $\Delta\phi_1^{(k)} = 0$ in Ω_1 and $\Delta\phi_2^{(k)} = 0$ in Ω_2 with Dirichlet condition $\phi_1^{(k)} = \phi_2^{(k)} = \varphi^{(k)}$ on Γ . This yields $\partial\phi_1^{(k)}/\partial n$ and $\partial\phi_2^{(k)}/\partial n$ (D/D).
2. Generate a Neumann condition $\psi^{(k)}$ by taking a weighted average of the computed solutions:

$$\psi^{(k)} = \omega_N^{(k)} \frac{\partial\phi_1^{(k)}}{\partial n} + (1 - \omega_N^{(k)}) \frac{\partial\phi_2^{(k)}}{\partial n} \quad \text{on } \Gamma.$$

3. Solve $\Delta\phi_1^{(k)} = 0$ in Ω_1 and $\Delta\phi_2^{(k)} = 0$ in Ω_2 with Neumann condition $\partial\phi_1^{(k)}/\partial n = \partial\phi_2^{(k)}/\partial n = \psi^{(k)}$ on Γ . This yields $\phi_1^{(k)}$ and $\phi_2^{(k)}$ (N/N).
4. Generate a Dirichlet condition $\varphi^{(k+1)}$ by taking a weighted average of the computed solutions:

$$\varphi^{(k+1)} = \omega_D^{(k)} \phi_1^{(k)} + (1 - \omega_D^{(k)}) \phi_2^{(k)} \quad \text{on } \Gamma.$$

5. Repeat procedure steps 1 to 4 ($k := k + 1$) until convergence is reached.

In the case of more subdomains (and thus more interfaces) the scheme can be generalized by performing each step in the scheme on all subdomains respectively all interfaces simultaneously. The exchange of information on every interface concerns only the neighbouring subdomains.

The implications for the numerical method are the following:

- By introducing the interfaces, extra panels are generated in the BEM. However, because the required memory per subdomain depends quadratically on the number of panels, the total required memory is reduced considerably.
- Because the number of panels per subdomain is much smaller, there is also a considerable reduction of required computation time for determining the source and dipole coefficients. They have to be determined only at the beginning of the iterative procedure. Because of the quadratic dependency of the required computation time on the number of collocation points, the total costs over all subdomains still is considerably smaller.
- There is a similar reduction of required computation time for solving the linear system of equations for one iteration step. But because now this has to be done every step of the iterative procedure, the total costs for this part of the method depend on the convergence of the iterative procedure.

The convergence of the iterative procedure is the subject of the following subsections. Results on reductions of computational costs will be given in Section 5.

The D/D-N/N scheme is also known as a Neumann–Neumann preconditioner in the context of domain decomposition methods for field discretization techniques. See, e.g., [11]. In this field there is a wide variety of methods. The explicit formulation used here was also used and analysed by Funaro *et al.* [7] for a different procedure. Here we restrict ourselves to the D/D-N/N scheme and we keep the weight factor constant: $\omega_N^{(k)} = \omega_D^{(k)} = \omega$.

3.2. Convergence Analysis

In the previous subsection we have formulated the scheme for Laplace’s equation with inhomogeneous boundary conditions on the outer boundaries. It is, however, important to realize that we only generate a sequence of boundary conditions on the interface and that the conditions on the outer boundaries are fixed during the iteration process. We can therefore simplify the description in the following way.

Because of the linearity of Laplace’s equation, the convergence of the solution of the Laplace problems with inhomogeneous boundary conditions is equivalent with the convergence of the solution of the Laplace problems for the error functions $\tilde{\phi}_i^{(k)} = \phi|_{\Omega_i} - \phi_i^{(k)}$, $i = 1, 2$, with homogeneous boundary conditions on the outer boundaries:

$$\begin{aligned} \Delta \tilde{\phi}_i^{(k)} &= 0 & \text{in } \Omega_i \\ \tilde{\phi}_i^{(k)} &= 0 & \text{on } \partial\Omega_i \setminus \Gamma, i = 1, 2. \\ \tilde{\phi}_i^{(k)} &= \tilde{\varphi} & \text{on } \Gamma. \end{aligned}$$

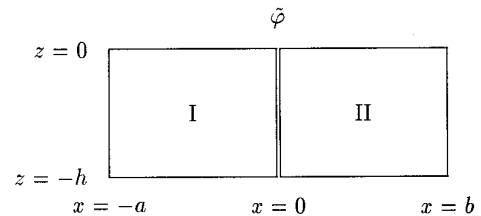


FIG. 3. Definition of the computational domain.

In our analysis we will therefore only consider Laplace problems with homogeneous boundary conditions on the outer boundaries. For these problems the interface boundary conditions have to converge to 0.

4. RESULTS

In this section the convergence of the iterative procedure for Laplace’s equation as it occurs in our wave problems is analyzed for a number of geometrical simplifications.

4.1. Results for Rectangular Domains

As a first simplification of the computational domain, we consider a rectangular domain divided into a number of subdomains in the horizontal direction. In the case of rectangular subdomains we can solve the Laplace problems analytically and the convergence of the iterative process can be analyzed fairly simple.

To simplify notation we will consider only problems with (homogeneous) Dirichlet conditions on the outer boundaries. The Laplace problems originating from water wave problems have a Neumann condition on the bottom. For problems with a horizontal bottom we can, however, reflect the homogeneous Laplace problem in the bottom to obtain a Laplace problem with only Dirichlet conditions. The height of the computational domain of course is twice that of the original one then.

4.1.1. Two Subdomains

We will investigate the convergence of the D/D-N/N scheme for a Dirichlet problem with two subdomains characterized by the dimensions given in Fig. 3.

In the iterative process according to the D/D-N/N-scheme, two types of Laplace problems are defined in the subdomains. One with a Dirichlet condition on the interface (procedure step 1) and one with a Neumann condition on the interface (procedure step 3). The solutions of these two problems can easily be obtained by separation of variables. The convergence of the iterative procedure can then be expressed in terms of the consecutive solutions.

We will first formulate the solutions on both subdomains with a Dirichlet condition on the interface $\Gamma = \{(x, z) \in \Omega | x = 0\}$. The original homogeneous Dirichlet problem

$$\begin{aligned} \Delta \tilde{\phi} &= 0 && \text{in } \Omega \\ \tilde{\phi} &= 0 && \text{on } \partial\Omega \end{aligned}$$

is decomposed into problems with the initial Dirichlet boundary condition $\tilde{\varphi}$ on the interface Γ :

$$\begin{aligned} \Delta \tilde{\phi}_i &= 0 && \text{in } \Omega_i \\ \tilde{\phi}_i &= 0 && \text{on } \partial\Omega_i \setminus \Gamma, i = 1, 2. \\ \tilde{\phi}_i &= \tilde{\varphi} && \text{on } \Gamma. \end{aligned}$$

We represent the boundary condition on the interface by the Fourier series

$$\tilde{\varphi}(z) = \sum_{n=1}^{\infty} c_n \sin\left(\frac{n\pi}{h} z\right).$$

The solution in domain I then equals

$$\tilde{\phi}_1(x, z) = \sum_{n=1}^{\infty} c_n \frac{\sinh\frac{n\pi}{h}(x+a)}{\sinh\frac{n\pi}{h}a} \sin\left(\frac{n\pi}{h} z\right),$$

and we have a similar expression for the solution in domain II. This leads to the expression for $\partial\tilde{\phi}_1/\partial x$ on the interface,

$$\begin{aligned} \frac{\partial\tilde{\phi}_1}{\partial x}\Big|_{\Gamma} &= \sum_{n=1}^{\infty} c_n \frac{n\pi}{h} \left(\tanh\left(\frac{n\pi}{h}a\right)\right)^{-1} \sin\left(\frac{n\pi}{h} z\right) \\ &=: \sum_{n=1}^{\infty} \alpha_{1,n} c_n \sin\left(\frac{n\pi}{h} z\right) \end{aligned} \tag{5}$$

and a similar one for $\partial\tilde{\phi}_2/\partial x$ on the interface. In these expressions each separate Fourier mode $c_n \sin((n\pi/h)z)$ is multiplied by a factor $\alpha_{1,n}$ respectively $\alpha_{2,n}$ which is independent of the Fourier coefficients c_n .

To arrive at a Neumann boundary condition $\tilde{\psi}$ we take a weighted average $\omega(\partial\tilde{\phi}_1/\partial x)|_{\Gamma} + (1 - \omega)(\partial\tilde{\phi}_2/\partial x)|_{\Gamma}$ of both solutions. An important observation is that the averaging formula can be applied to the separate Fourier modes of $(\partial\tilde{\phi}_i/\partial x)|_{\Gamma}$. Therefore, we can represent this step in the iterative procedure for each Fourier mode $\tilde{\varphi}_n$ separately by the diagram

$$\tilde{\varphi}_n := c_n \sin\frac{n\pi}{h} z$$

$$\left| \begin{array}{ccc} & \swarrow & \\ \frac{\partial\tilde{\phi}_1}{\partial x}\Big|_{\Gamma} = \alpha_{1,n}\tilde{\varphi}_n & \parallel & \frac{\partial\tilde{\phi}_2}{\partial x}\Big|_{\Gamma} = \alpha_{2,n}\tilde{\varphi}_n \\ & \searrow & \end{array} \right|$$

$$\begin{aligned} \tilde{\psi}_n &= \omega \frac{\partial\tilde{\phi}_1}{\partial x}\Big|_{\Gamma} + (1 - \omega) \frac{\partial\tilde{\phi}_2}{\partial x}\Big|_{\Gamma} \\ &= (\omega\alpha_{1,n} + (1 - \omega)\alpha_{2,n})\tilde{\varphi}_n. \end{aligned}$$

In each even step of the iteration procedure a Neumann condition is imposed on the interface. The solution process of this step can be represented by a similar diagram as above,

$$\tilde{\psi}_n := c'_n \sin\frac{n\pi}{h} z$$

$$\left| \begin{array}{ccc} & \swarrow & \\ \tilde{\phi}_1|_{\Gamma} = \beta_{1,n}\tilde{\psi}_n & \parallel & \tilde{\phi}_2|_{\Gamma} = \beta_{2,n}\tilde{\psi}_n \\ & \searrow & \end{array} \right|$$

$$\begin{aligned} \tilde{\varphi}_n &= \omega\tilde{\phi}_1|_{\Gamma} + (1 - \omega)\tilde{\phi}_2|_{\Gamma} \\ &= (\omega\beta_{1,n} + (1 - \omega)\beta_{2,n})\tilde{\psi}_n. \end{aligned}$$

The coefficients $\beta_{1,n}$ and $\beta_{2,n}$ are equal to $\alpha_{1,n}^{-1}$ and $\alpha_{2,n}^{-1}$, respectively: calculating $(\partial\tilde{\phi}_i/\partial x)|_{\Gamma}$ for a given $\tilde{\phi}_i|_{\Gamma}$ is the inverse operation from calculating $\tilde{\phi}_i|_{\Gamma}$ for a given $(\partial\tilde{\phi}_i/\partial x)|_{\Gamma}$.

Now that we have formulated the solutions of all the Laplace problems that occur during the iterative process, we can express the rate of convergence in terms of the aforementioned parameters. The D/D-N/N scheme can be represented as

$$\tilde{\varphi}^{(0)} \rightarrow \tilde{\psi}^{(0)} \rightarrow \tilde{\varphi}^{(1)} \rightarrow \tilde{\psi}^{(1)} \rightarrow \tilde{\varphi}^{(2)} \rightarrow \tilde{\psi}^{(2)} \rightarrow \dots$$

All interface values of the potential can be expressed in terms of the starting value $\tilde{\varphi} = \tilde{\varphi}^{(0)}$ per Fourier mode:

$$\tilde{\varphi}_n^{(k+1)} = \varepsilon_n \tilde{\varphi}_n^{(k)} = \varepsilon_n^k \tilde{\varphi}_n^{(0)}.$$

We call ε_n the reduction factor of the n th Fourier mode. It is independent of k . Therefore the convergence process can be characterized by this parameter. If $|\varepsilon_n| < 1$ then the iterative procedure is convergent for the n th Fourier mode.

For the D/D-N/N scheme,

$$\varepsilon_n = (2 + \gamma_n)\omega^2 - (2 + \gamma_n)\omega + 1 = 1 - (2 + \gamma_n)\omega(1 - \omega),$$

where

$$\gamma_n = \frac{\tanh\frac{n\pi}{h}b}{\tanh\frac{n\pi}{h}a} + \frac{\tanh\frac{n\pi}{h}a}{\tanh\frac{n\pi}{h}b}. \tag{6}$$

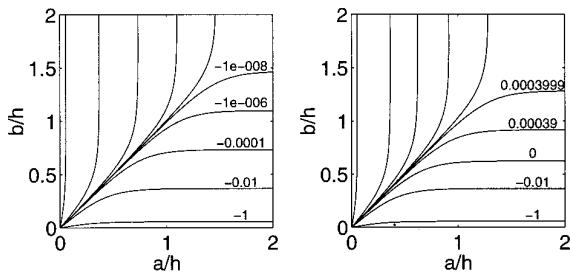


FIG. 4. Contour plots for ε_1 : left graph, $\omega = 0.50$; right graph, $\omega = 0.49$.

In the analysis of this expression we distinguish between the case of subdomains of equal length ($a = b$) and the case of subdomains of unequal length ($a \neq b$).

If $a = b$ we have $\gamma_n = 2$ and $\varepsilon_n = 4(\omega - \frac{1}{2})^2$. So the reduction factor ε_n is independent of n . If we choose $\omega = \frac{1}{2}$ we find $\varepsilon_n = 0$. Thus the exact result is found after one iteration. This can be explained by considering the geometry of the domain; because of the symmetry we find opposed horizontal derivatives on the interface, which by averaging ($\omega = \frac{1}{2}$) directly leads to the correct Neumann boundary condition $\tilde{\psi}^{(0)} = 0$.

If $a \neq b$ then different Fourier modes have different reduction factors. For a given geometry the range of γ_n as function of n is limited. Therefore also the range of the reduction factors ε_n is limited; ε_n is an increasing function of n with lower bound $\varepsilon_1 = 1 - (2 + \gamma_1)\omega(1 - \omega)$ and upper bound $\varepsilon_\infty = 4(\omega - \frac{1}{2})^2$.

More information about the convergence can be obtained when we express γ_n in terms of the length-to-height ratios of the subdomains. We take

$$\hat{a} = \frac{a}{h}, \quad \hat{b} = \frac{b}{h}.$$

Then we can express γ_n as

$$\gamma_n = \frac{\tanh n\pi\hat{b}}{\tanh n\pi\hat{a}} + \frac{\tanh n\pi\hat{a}}{\tanh n\pi\hat{b}}.$$

For a fixed value of ω and varying values of \hat{a} and \hat{b} the contour plots in Fig. 4 indicate the reduction factor ε_1 for different geometrical configurations. Because of the symmetry of the plots, the values of the reduction factors for the contour lines in the upper left parts have not been indicated. Notice that by taking the multiples ($n\hat{a}$, $n\hat{b}$), we can find the reduction factors ε_n of the consecutive Fourier modes.

Figure 4 indicates the following: For a large set of geometrical configurations the iterative procedure is convergent for all Fourier modes if $\omega = 0.50$. For very small

length-to-height ratios in one of the subdomains the D/D-N/N scheme does not converge for the first modes. When the length-to-height ratio is larger than 2 for both subdomains, the iterative procedure converges very fast for all Fourier modes. In one iteration step the difference $|\tilde{\phi}_1 - \tilde{\phi}_2|$ will be reduced by a factor smaller than 10^{-10} .

We can also see that the set of geometrical configurations for which the iterative procedure is convergent for $\omega = 0.49$ is approximately the same as for $\omega = 0.5$. However, for large length-to-height ratios the reduction factors of all Fourier modes are approximately equal to the upper bound $\varepsilon_\infty = 0.0004$. The 0-contour line indicates the set of geometrical configurations for which the reduction factor of the first Fourier mode equals 0. Nevertheless, the reduction factors of the higher modes in the same configuration have a positive value and converge much slower.

These results can be interpreted as follows: For subdomains with equal length, the solutions generated in the iterative procedure are symmetric with respect to the boundary condition on the interface. A symmetric averaging ($\omega = \frac{1}{2}$) annihilates the error made in the initial boundary condition.

For subdomains with unequal lengths, the solutions generated in the iterative procedure are not symmetric. However, this asymmetry is not noticeable when the length-to-height ratios \hat{a} and \hat{b} are large, especially for the higher Fourier modes. Taking $\omega = \frac{1}{2}$ then also leads to very fast convergence.

4.1.2. Three or More Subdomains

The iterative process in the multisubdomain problem can be described similar to the iterative process of the 2-subdomain problem in the previous subsection. We will formulate and analyze it for the 3-subdomain problem first. Consider the domain shown in Fig. 5.

In step 1 Dirichlet boundary conditions are imposed on the interfaces and Neumann conditions are imposed in step 3. We can, again, compute the solution of both types of problems in all subdomains. The difference with the 2-subdomain problem is now that the inner subdomain has two inhomogeneous boundary conditions instead of one. The solution there is the sum of the Laplace problem with $\tilde{\varphi}_1$ (resp. $\tilde{\psi}_1$) as boundary condition and homogenous

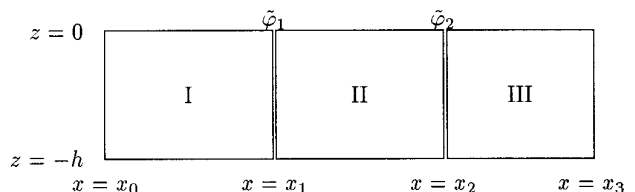


FIG. 5. Definition of the computational domain with three subdomains.

boundary conditions elsewhere plus the Laplace problem with $\tilde{\varphi}_2$ (resp. $\tilde{\psi}_2$) as boundary condition with otherwise homogenous boundary conditions.

Again, we can describe the iterative process for each Fourier mode separately. The transformation of the n th Fourier-mode $\tilde{\varphi}_{1,n}^{(k)} = c_{1,n} \sin((n\pi/h)z)$ on interface Γ_1 and $\tilde{\varphi}_{2,n}^{(k)} = c_{2,n} \sin((n\pi/h)z)$ on interface Γ_2 during the iterative process can be represented by the transformation matrix A_n ,

$$\begin{pmatrix} \tilde{\varphi}_{1,n}^{(k+1)} \\ \tilde{\varphi}_{2,n}^{(k+1)} \end{pmatrix} = \begin{pmatrix} A_{11,n} & A_{12,n} \\ A_{21,n} & A_{22,n} \end{pmatrix} \begin{pmatrix} \tilde{\varphi}_{1,n}^{(k)} \\ \tilde{\varphi}_{2,n}^{(k)} \end{pmatrix} =: A_n \begin{pmatrix} \tilde{\varphi}_{1,n}^{(k)} \\ \tilde{\varphi}_{2,n}^{(k)} \end{pmatrix}.$$

The matrix A_n can be expressed in terms of coefficients similar to $\alpha_{i,n}$ and $\beta_{i,n}$ in the 2-subdomain problem and the parameter ω . The convergence of the iterative process is determined by the two eigenvalues of the transformation. If both eigenvalues $\varepsilon_{1,n}$ and $\varepsilon_{2,n}$ lay inside the unit circle then the iterative process will converge for the n th Fourier mode.

For a general geometrical configuration, the expressions for the eigenvalues are very complicated and involve too many parameters to be analyzed easily. Therefore we consider the case of subdomains of equal length l and $\omega = \frac{1}{2}$. The matrix A_n then has one eigenvalue ε_n of multiplicity 2 which is equal to

$$\varepsilon_n = -\frac{1}{4C(C-1)},$$

with $C = \cosh((n\pi/h)l)$. The first Fourier mode ($n = 1$) exhibits the slowest convergence. The condition $|\varepsilon_1| < 1$ gives the condition on the ratio l/h ,

$$\frac{l}{h} > \pi^{-1} \operatorname{arccosh} \left(\frac{1}{2} + \frac{1}{2} \sqrt{2} \right) \approx 0.2.$$

This condition is similar to the condition for the 2-subdomain problem.

When the subdomains are not of equal length, the eigenvalues can be determined for some given parameters. For large length-to-height ratio of all subdomains the choice of $\omega = \frac{1}{2}$ leads to very fast convergence on both interfaces for all Fourier modes. The symmetry-argument used in the 2-subdomain case can be used here as well; because of the large length-to-height ratio the solutions that are generated during the iterative process are determined mainly by the boundary conditions on the interfaces. The solutions on either side of each interface will be almost symmetric which explains the fast convergence.

The matrix representation can also be used in the case of a decomposition into four or more subdomains. For a

decomposition into M subdomains, $M - 1$ interfaces are present and $M - 1$ independent initial Dirichlet conditions must be formulated. During the iteration process they are transformed and the transformation can be represented by a $(M - 1)$ by $(M - 1)$ matrix A_n for each Fourier mode.

The parameters in this matrix involve the M subdomain lengths and the weight factor ω . To be able to say something about the convergence, we again consider the case of subdomains of equal length l and $\omega = \frac{1}{2}$.

If $M - 1$ is odd (and $M \geq 4$) then the matrix A_n is nilpotent with index 0, implying that all eigenvalues are 0.

If $M - 1$ is even (and $M \geq 5$) then the matrix A_n has one eigenvalue $\varepsilon_{1,n}$ of multiplicity 4 and one eigenvalue $\varepsilon_{2,n}$ of multiplicity $M - 4$ which both are $O(\frac{1}{4}S^{-2})$, with $S = \sinh((n\pi/h)l)$. The exact condition on l/h for convergence is $\frac{1}{4}S^{-2} < 1$ or

$$\frac{l}{h} > \pi^{-1} \operatorname{arcsinh} \left(\frac{1}{2} \right) \approx 0.15.$$

These results again show very fast convergence for a large length-to-height ratio. The symmetric subdivision ($M - 1$ is odd) transforms all initial boundary conditions to 0. The asymmetric subdivision ($M - 1$ is even) shows a condition similar to the one for the 3-subdomain problem.

4.1.3. Three-Dimensional Problems

The analysis and results are also applicable to Laplace problems in three dimensions. If we consider a rectangular three-dimensional domain with height h and width w with interfaces in y - z -planes, then we have to deal with Fourier series of the form

$$\tilde{\varphi}(y, z) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} c_{m,n} \sin \left(\frac{m\pi}{w} y \right) \cdot \sin \left(\frac{n\pi}{h} z \right).$$

The expressions describing the iterative process in three dimensions are the same as those in two dimensions with $n\pi/h$ replaced by

$$\left(\left(\frac{m}{w} \right)^2 + \left(\frac{n}{h} \right)^2 \right)^{1/2} \pi.$$

This applies to, for example, Eq. (6), but also to the expressions for the eigenvalues in the multisubdomain case. Therefore the convergence results for the 2-subdomain problem can be translated directly to three-dimensional problems; fast convergence will occur when *either* the length-to-height *or* the length-to-width ratio is large. The convergence is determined by the smallest ratio.

This last result gives the opportunity to handle large three-dimensional problems. The convergence is not af-

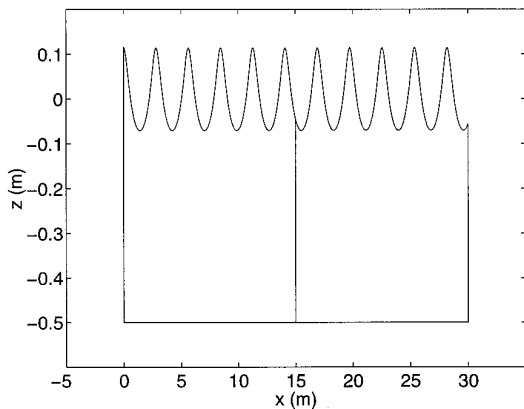


FIG. 6. Computational domain for nonlinear wave with two subdomains.

ected by a small length-to-width ratio if the length-to-height ratio is not too small. If the subdomains in this kind of problems are still very large then the domain decomposition method can be applied again in the separate subdomains in the y -direction. In this way a checkerboard of subdomains is created in which the actual solution of the Laplace problems is generated in the small cells of this checkerboard. The required number of iterations is at most the maximum number for the division in one direction times the maximum number for the other direction.

4.2. Results for Domains with Even Bottoms

In the simulation of nonlinear water waves over an even bottom, the computational domain in which Laplace's equation has to be solved deviates from the rectangular form because of the disturbed free surface boundary. As the next simplification we therefore take a rectangular domain with a local asymmetrical disturbance near the interface. We suppose that l/h is large enough to have no noticeable influence of the lengths of the subdomain on the solutions on the interface.

An analysis in which exact or approximating expressions for solutions are used to characterize the iterative process is very difficult now. It is, however, clear that the asymmetry near the interface leads to an asymmetry in the solutions on both sides of the interfaces. This can be explained by expressing the solutions in terms of Fourier modes again.

Consider a Laplace problem on a domain with a wave-like disturbance of the upper horizontal boundary and divided by a vertical interface. See Fig. 6. Homogeneous boundary conditions are imposed at the outer boundaries and an inhomogeneous one at the interface. If we consider this problem on a domain with the upper boundary replaced by the undisturbed upper horizontal boundary then also an inhomogeneous boundary condition has to be imposed there. For this problem, the solution in both subdo-

main is not only expressed in Fourier modes along the interface, but also in Fourier modes along the upper boundary. The contribution of the latter to the new boundary condition on the interface will be different on both sides of the interface and convergence will deteriorate due to this asymmetry.

4.2.1. Two Subdomains

To have an indication of the sensitivity of the iterative process for this asymmetry we shall present some numerical experiments in this section. For the numerical investigations we define the reduction factors $\varepsilon_D^{(k)}$ and $\varepsilon_N^{(k)}$ as the reduction of the error over the interface in ϕ and $\partial\phi/\partial n$, respectively, in iteration step k . In the numerical method this is translated into

$$\varepsilon_D^{(k)} = \frac{\max_i |\phi_1^{(k+1)}(\mathbf{x}_i) - \phi_2^{(k+1)}(\mathbf{x}_i)|}{\max_i |\phi_1^{(k)}(\mathbf{x}_i) - \phi_2^{(k)}(\mathbf{x}_i)|}, \quad \mathbf{x}_i \in \Gamma,$$

and similarly for $\varepsilon_N^{(k)}$. The subindices 1 and 2 of ϕ now refer to the subdomains. These reduction factors indicate the convergence of the iteration process. For rectangular domains $\varepsilon_D^{(k)}$ and $\varepsilon_N^{(k)}$ are equivalent with the reduction factor defined in Section 4.1 for single Fourier modes.

In the following we will look at Laplace problems originating from one specific nonlinear wave problem and will investigate the relationship between the angle between free surface and interface and an average of the reduction factors for this geometry.

We consider the computational domain of Fig. 6. The free surface shape corresponds to a waveprofile of a nonlinear stationary propagating wave and can be calculated according to the theory of Rienecker and Fenton [13]. The wave conditions for this wave are: depth $h = 0.50$ m, wave period $T = 1.44$ s, wave height $H = 0.185$ m, and wave length $\lambda = 2.82$ m. The wave is moderately high; it is equal to 60% of the maximum wave height.

In the numerical experiments the interface has been placed at different locations, compared with the position of the wavetops, in order to obtain different values of the slope of the free surface at the interface. For these wave conditions the maximum slope is obtained when the wavetop is shifted approximately $\frac{1}{8}\lambda$ from the position of the interface. The starting boundary condition $\varphi^{(1)}$ for the iteration process was taken equal to 2.0 which is very different from the actual solution there for all cases.

Two different grid distributions have been taken on the interface, both with five collocation points. An equidistant grid (I) and a nonequidistant grid clustering near the free surface (II).

In Fig. 7 the results of these computations are plotted. The iterative process in the numerical experiments reached the stop criterion (either ϕ or $\partial\phi/\partial n$ has difference over

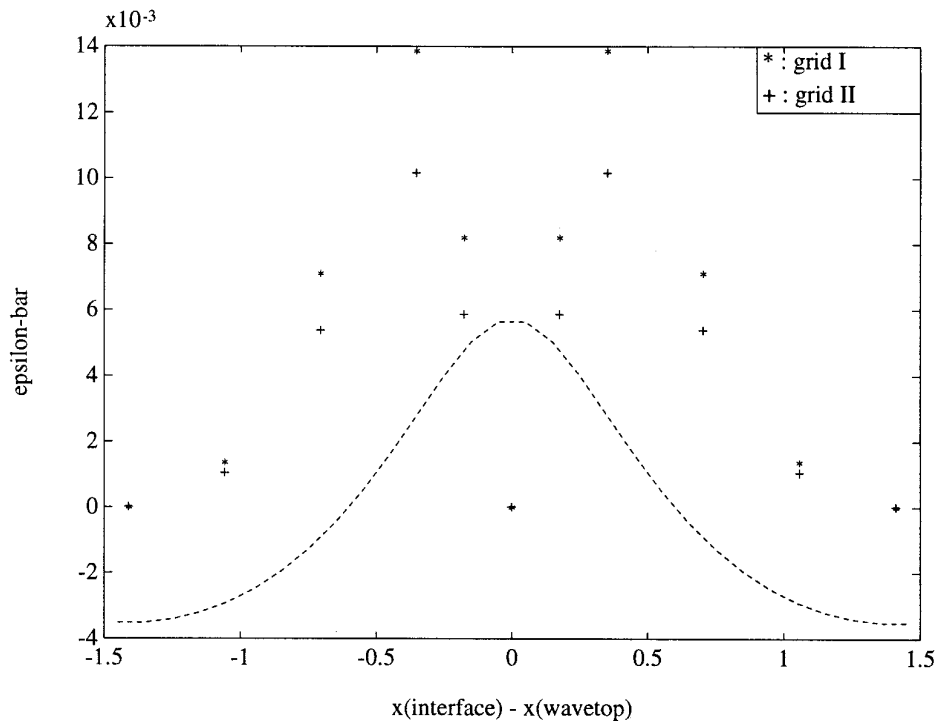


FIG. 7. Average reduction factors for some geometries.

the interface smaller than 10^{-7}) for all cases in less than 4 iterations. The average reduction factor $\bar{\epsilon}$ is defined as $(\epsilon_D^{(1)} + \epsilon_N^{(1)})/2$. The following observations can be made:

- The reduction factor $\bar{\epsilon}$ increases when the slope α of the free surface at the interface increases. In fact the reduction factor $\bar{\epsilon}$ is approximately quadratic in α .
- The reduction factor $\bar{\epsilon}$ is smaller for the nonequidistant grid. Examination of both subdomain solutions indicates that it is determined by the difference in the computed unknown ($\partial\phi/\partial n$, respectively ϕ) in the upper collocation point.
- The reduction factor $\bar{\epsilon}$ and the individual reduction factors $\epsilon_D^{(k)}$ and $\epsilon_N^{(k)}$ are very small. Only a few iterations are needed to achieve the accuracy of the BEM-discretizations.

We conclude that the iterative process in the numerical method converges well. Its convergence rate depends on the discretization. The optimal convergence rate is achieved with the most accurate solution near the local asymmetry of the interface.

For a given computational domain corresponding to a nonlinear wave problem, the best choice of a vertical interface with respect to convergence rate is perpendicular to the free surface.

4.2.2. Three or More Subdomains

If the computational domain of a wave problem is divided into three or more subdomains, then convergence at each interface is influenced by the local asymmetry. For rectangular subdomains of equal size we saw that the convergence rate is independent of the number of subdomains. To check if the same is true for a general wave problem with an even bottom we considered a stationary wave problem as in the previous section. For computations with 12, 20, and 100 subdomains of fixed size, it was seen that with the use of more subdomains, more iterations were required, but that it did not exceed a certain limit. This limit is determined by the iterative process on the interface with the slowest convergence which corresponds with the maximum slope for a given wave problem. Therefore, with the use of many subdomains, the CPU time per subdomain is independent of the number of subdomains and the computational costs depend only linearly on the number of subdomains.

4.2.3. Three-Dimensional Problems

In wave problems in three dimensions the slope of the free surface at the intersection with the interface varies along the interface. To have an indication of the influence on the convergence rate, we have performed some numerical experiments with regular waves and found that the

reduction factor is related to the maximum slope of the free surface in the direction perpendicular to the interface. See [9]. For waves propagating not perpendicular to the interface this slope is smaller than the skewness of the wave itself and convergence of the iterative process is better than for the corresponding two-dimensional problem.

4.3. Results for Domains in General Wave Problems

In wave problems over an arbitrary bottom topography the convergence rate can be studied further by taking into account the slope at the bottom near the interface. We have performed no specific numerical experiments to investigate this. The effect of an uneven bottom has been noticed in time-domain experiments. It leads to a deterioration of the convergence rate. These experiments will be presented next.

5. APPLICATION TO TIME-DOMAIN COMPUTATIONS

5.1. Implementation

So far we have investigated the domain decomposition method for Laplace problems for a fixed geometrical configuration. In the time-domain computation of propagating nonlinear water waves, the computational domain changes in time and Laplace's equation for ϕ has to be solved at every time level. For the efficiency of the domain decomposition method in the complete numerical method this has important consequences.

The free surface shape changes in time according to the kinematic free surface condition. For the solution of Laplace's equation one is free to choose the position of the interfaces. With respect to convergence rates, the analysis of the previous sections implies that for vertical interfaces the optimal choice is perpendicular to the free surface. With respect to equal loads in the subdomains equally spaced interfaces are optimal. However, both demands cannot be satisfied simultaneously. Moreover, the choice of perpendicular connecting interfaces will necessarily lead to a reorganization of data from one subdomain to the other as the time-marching scheme proceeds. We will clarify this in the following.

In an Eulerian description of the movement of the grid of the free surface, the collocation points only move in the vertical direction. To have an organization of data into subdomains which is fixed in time, the interfaces then have to be located at fixed horizontal positions. The propagating water waves will then "move through" the interfaces and the free surface will not be rectangular to the interface but its slope will vary from a minimum value to a maximum value.

We use a Lagrangian description in the present method. This means that the collocation points move along with the water particles. In this case no reorganization of data

TABLE I
Speedup for the Simulation of Regular Waves

No. of subdomains	Aver. no. of iterations	CPU time (10^3 s)	Speedup
1	—	12.7	1
2	4.23	13.3	0.97
4	5.39	5.6	2.3
8	6.09	2.7	4.7
12	6.94	2.2	5.8
16	6.74 [†]	0.98 [†]	—

is required if (and only if) the interfaces drift along with the water particles of the free surface. Still, the waves will propagate through the interfaces because the phase speed of the waves is larger than the velocity of the water particles.

In the present implementation of the domain decomposition method this choice (interfaces that drift along with the water particles) has been made. It implies a loss of efficiency with respect to the convergence rate of the domain decomposition method. But it has two important advantages with respect to parallel computing. First of all it gives equal loads in all subdomains and secondly the transfer of data between different processors from one time-level to the other will be minimal. The efficiency of the domain decomposition method in combination with parallel computing has been studied in [8].

5.2. Efficiency

In [8] we have examined the efficiency of domain decomposition on the most simple problem of regular waves over an even bottom. During the simulation waves enter the computational domain. We shall briefly repeat these results here.

For a division into 1, 2, 4, 8, 12, and 16 subdomains we have measured CPU times and the averaged number of required iterations for a simulation of 200 time steps. These results are given in Table I. The computations were done on a HP workstation.

From Table I it is clear that more iterations are required as the number of subdomains increases. However, the total required CPU time decreases and gives a considerable speedup in the case of 4, 8, and 12 subdomains.

In the course of the simulation the number of required iterations increases due to the wave signal entering the subdomains. The asymmetry near the interfaces increases which slows down the convergence rate. Because of this, in combination with the small length-to-height ratio of the subdomains, the iterative process for 16 subdomains diverges after 197 time steps.

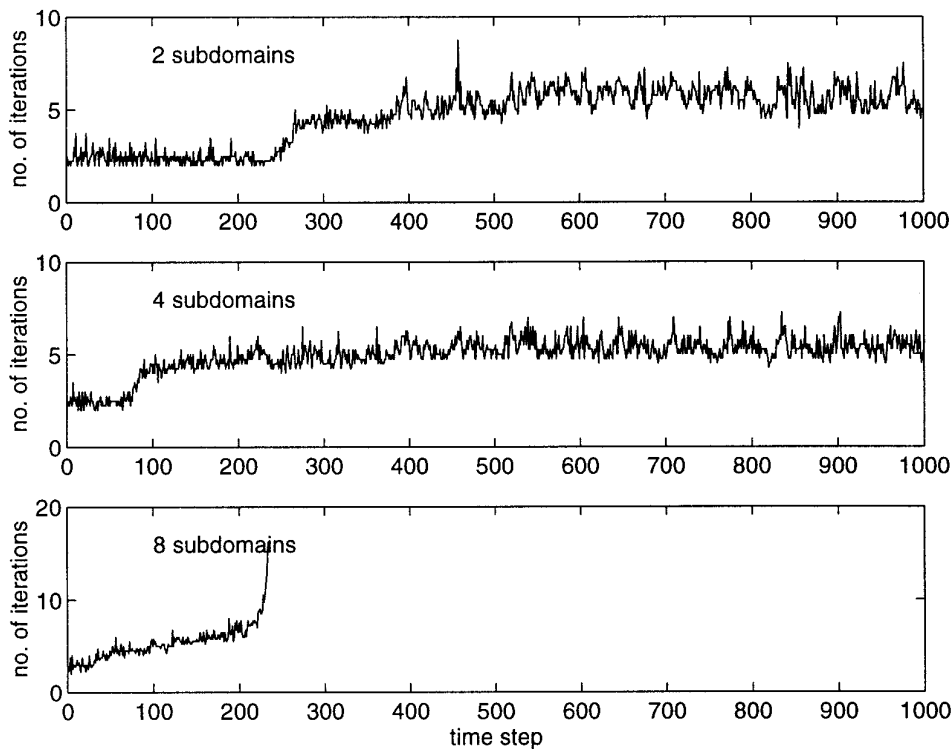


FIG. 8. Number of iterations during a simulation using 2, 4, and 8 subdomains.

5.3. Example: Waves over an Underwater Bar

As a first example of applying domain decomposition in a time-domain computation with an uneven bottom we will consider the simulation represented in Figs. 1 and 2 which includes an underwater bar. The point of interest with respect to wave propagation is the generation of higher frequency components of the incoming wave by the underwater bar. See [1] for a description of physical tests carried out in a wave flume.

We will first consider the number of required iterations. For a division into 2, 4, and 8 subdomains we have plotted the index of the time step against the number of required iterations averaged over the four time levels of the Runge–Kutta method. See Fig. 8.

Again it can be seen that as waves enter the computational domain, the number of iterations increases due to the increase of asymmetry near the interfaces. The increase in the case of eight subdomains becomes so large that it exceeds the imposed maximum of 20 iterations after 234 time steps. Another interesting observation is that the variation in the case of two subdomains is larger than in the case of four subdomains. Averaged over the total simulation remarkably less iterations are required.

These simulations have been performed on a Pentium/

90 PC and on a Cray C98 computer. Results on required CPU time and memory are given in Table II.

A considerable speedup can be noticed when using a PC in the case of four subdomains. For the Cray computer the speedup is not so spectacular. This is connected with the relatively slower performance of the Cray on memory references in two-dimensional arrays representing the source and dipole coefficients. Using domain decomposition these arrays have to be read every iteration step.

5.4. Example: Simulation of the Generation of Long Waves

As an example of the application in a large-scale wave problem, the simulation of irregular waves over an uneven

TABLE II
Computational Requirements for Different
Number of Subdomains

No. of subdomains	CPU time on PC (h)	CPU time on C98 (min)	Required memory (MWord)
1	21.51	22.00	1.729
2	19.51	20.16	1.289
4	7.84	13.60	1.088
8	—	—	1.049

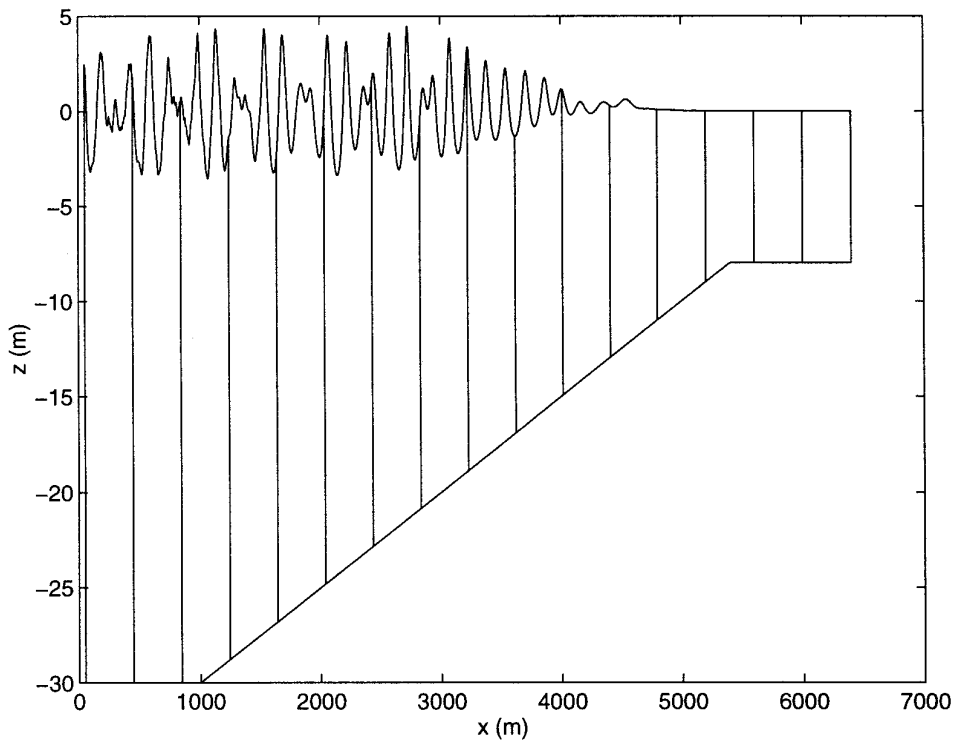


FIG. 9. Computational domain of a large-scale wave problem with 16 subdomains.

bottom, thereby generating long waves, is shown. See Fig. 9.

In this simulation the original computational domain consisting of 1930 panels, was divided into 16 subdomains, giving a problem with 2080 panels. Memory requirements, however, dropped from 11.70 MWords to 2.43 MWords. The computation was done for 1775 time steps simulating a period of 355 s. It took about 2 h on the Cray computer. Because of the large CPU time requirements no simulations with less subdomains or with a larger computational domain were performed.

5.5. Three-Dimensional Problems

In Section 4 it was already mentioned that convergence in 3D problems is at least as good as convergence in 2D for the corresponding problems. In [9] it was shown that, also for time-dependent problems, only a limited number of iterations is required. But because relatively more extra panels are introduced, reductions of computational costs are much smaller in 3D.

Nevertheless, domain decomposition seems to be an appropriate technique to handle large-scale simulations in 3D because of its suitability for parallel computing and the linear dependency of computational requirements on the length of the domain. These aspects are under investigation now and will be reported in the future.

6. CONCLUSION

For the solution of Laplace's problem in geometries related to water wave problems, a domain decomposition method is proposed which is applied in combination with a higher order boundary element method. For a length-to-height ratio of the subdomains which is not too small, the convergence is fast. With the use of a panel method to solve practical wave problems it is possible to achieve considerable reductions of required memory and a speedup of the computations. Besides that, it is possible to increase the length of the computational domain with only a linear increase in the computational costs.

ACKNOWLEDGMENTS

The investigations were supported by the National Computing Facilities Foundation (NCF) for the use of supercomputer facilities, with financial support from the Dutch Technology Foundation (STW), by Delft Hydraulics, and by the University of Twente. The first author further gratefully acknowledges his colleagues at the University of Twente, at Delft Hydraulics, and at Marin for their ideas and feedback.

REFERENCES

1. S. Beji and J. A. Battjes, *Coastal Eng.* **19**, 151 (1993).
2. P. J. F. Berkvens and P. J. Zandbergen, *J. Ship Res.*, to appear. 1996.
3. J. F. Bourgat, R. Glowinski, P. Le Tallec, and M. Vidrascu, in *Proceed-*

- ings, *2nd Int. Conf. on Domain Decomposition Methods for Partial Differential Equations*, edited by T. F. Chan *et al.* (SIAM, Philadelphia, 1989), p. 3.
4. J. Broeze, Ph.D. thesis, University of Twente, Enschede, 1993 (unpublished).
 5. J. Broeze, E. F. G. Van Daalen, and P. J. Zandbergen, *Comput. Mech.* **13**, 12 (1993).
 6. T. F. Chan, R. Glowinsky, J. Périaux, and O. B. Widlund (Eds.), *Proceedings, 2nd Int. Conf. on Domain Decomposition Methods for Partial Differential Equations* (SIAM, Philadelphia, 1989).
 7. D. Funaro, A. Quarteroni, and P. Zanolli, *SIAM J. Numer. Anal.* **25**, 1213 (1988).
 8. P. C. A. de Haas, J. Broeze, and M. Streng, in *Proceedings, 1996 EUROSIM Int. Conf., Delft 1996*, to appear.
 9. P. C. A. de Haas, J. Broeze, and P. J. Zandbergen, in *Proc. 15th Int. Conf. on Numerical Methods in Fluid Dynamics, Monterey, 1996*, to appear.
 10. M. A. Jaswon and G. T. Symm, *Integral Equation Methods in Potential Theory and Elastostatics* (Academic Press, London, 1977).
 11. P. Le Tallec, *Comput. Mech. Adv.* **1**, 121 (1994).
 12. L. D. Marini and A. Quarteroni, *Numer. Math.* **55**, 575 (1989).
 13. M. M. Rienecker and J. D. Fenton, *J. Fluid Mech.* **104**, 119 (1981).
 14. J. E. Romate, Ph.D. thesis, University of Twente, Enschede, 1989 (unpublished).
 15. J. E. Romate and P. J. Zandbergen, *Comput. Mech.* **4**, 276 (1989).
 16. H. A. Schwarz, *Ges. Math. Abh.* **11**, 65 (1869).
 17. E. F. G. Van Daalen, Ph.D. thesis, University of Twente, Enschede, 1993 (unpublished).
 18. P. Wang, Y. Yao, and M. P. Tulin, *Int. J. Offshore Polar Eng.* **4**, 200 (1994).