VIII International Conference on Computational Methods in Marine Engineering
MARINE 2019
R. Bensow and J. Ringsberg (Eds)

EXTENSION OF A FAST METHOD FOR 2D STEADY FREE SURFACE FLOW TO STRETCHED SURFACE GRIDS

Toon Demeester[1,*], E. Harald van Brummelen[2] and Joris Degroote[1,3]

[1] Department of Flow, Heat and Combustion Mechanics, Ghent University
Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium, toon.demeester@ugent.be

[2] Department of Multiscale Engineering Fluid Dynamics, Eindhoven University of Technology
PO Box 513-5600 MB Eindhoven, the Netherlands

[3] Flanders Make, Belgium

Key words: free surface flow, fitting method, surrogate model, quasi-Newton, convolution theorem

Abstract. Steady free surface flow is often encountered in marine engineering, e.g. for calculating ship hull resistance. When these flows are solved with CFD, the water-air interface can be represented using a surface fitting approach. The resulting free boundary problem requires an iterative technique to solve the flow and at the same time determine the free surface position. Most such methods use a time-stepping scheme, which is inefficient for solving steady flows. There is one steady technique which uses a special boundary condition at the free surface, but that method needs a dedicated coupled flow solver. To overcome these disadvantages an efficient free surface method was developed recently, in which the flow solver can be a black-box. It is based on quasi-Newton iterations which use a surrogate model in combination with flow solver inputs and outputs from previous iterations to approximate the Jacobian. As the original method was limited to uniform free surface grids, it is extended in this paper to stretched free surface grids. For this purpose, a different surrogate model is constructed by transforming a relation between perturbations of the free surface height and pressure from the wavenumber domain to the spatial domain using the convolution theorem. The method is tested on the 2D flow over an object. The quasi-Newton iterations converge exponentially and in a low number of iterations.

1 INTRODUCTION

Steady free surface flows of incompressible, viscid fluids are often encountered in the field of marine engineering. The most common application is the ship hull resistance problem. These problems tend to be solved using CFD methods, where the free surface poses an additional difficulty as its position is unknown a priori. Methods to solve these flows deal with the free surface in different ways. Two approaches to represent the free surface can be distinguished:

surface capturing and surface fitting. In capturing approaches the free surface intersects the mesh in an arbitrary manner, which makes them more versatile. Examples are the volume-of-fluid [1] and level-set [2] methods. In surface fitting approaches, the mesh is aligned with the free surface so that its position is known more accurately. These methods are very suitable for the envisioned applications and thus the topic of this paper.

For the envisaged applications in marine engineering, the air-phase may be neglected due to the large density difference with water. The free surface then becomes a free boundary of the domain and the interface conditions must be enforced as kinematic and dynamic boundary conditions. The kinematic boundary condition (KBC) requires that the free surface is impermeable: for steady flows the velocity at the free surface must be parallel to it. The dynamic boundary condition (DBC) requires continuity of the stresses. As the air phase is neglected, this condition can be simplified to the requirement of zero shear stress (tangential DBC) and a constant (atmospheric) pressure (normal DBC) at the free surface.

As there are more free surface boundary conditions than can be applied when solving the Navier-Stokes equations with fixed free surface position, the free boundary problem must inherently be solved iteratively for the flow field and free surface position. Existing methods do this by distributing the boundary conditions over two steps: in one step the flow is solved with a fixed free surface position, in the other step the free surface position is calculated and the mesh deformed accordingly. Most of the methods found in literature use the DBC in the flow solver and the KBC to update the free surface position [3, 4]. This leads to a time-stepping scheme, which is inefficient for steady free surface flows due to the slow decay of transient phenomena. The *steady iterative method* by van Brummelen et al. [5] uses a combined boundary condition (KBC + DBC) in the flow solver and the normal DBC for the surface update. This method converges in a low number of iterations, but needs a dedicated coupled solver to deal with the difficult combined boundary condition.

In [6] a new method was presented which avoids the disadvantages of existing methods. By using an efficient quasi-Newton method based on the normal DBC, it converges in a low number of iterations. The other boundary conditions are easily enforced in a general-purpose (black-box) flow solver by using a free-slip wall. The original method was developed for uniform free surface grids and is in this paper extended to stretched free surface grids.

In Section 2 a surrogate model is developed for a black-box flow solver. This surrogate is used in Section 3 to approximate the Jacobian which is needed to solve the free boundary problem with quasi-Newton iterations. In Section 4 the method is tested by solving the 2D free surface flow over an object.

2 FLOW SOLVER SURROGATE MODEL

In Section 3 a quasi-Newton method is outlined to solve the steady free surface flow as described above. The approximate Jacobian which is used is based (partially) on a surrogate model for the black-box flow solver. The surrogate model proposed originally in [6] can only deal with a uniform free surface discretization, as it is based on a Fourier series decomposition. A more versatile surrogate model which can deal with stretched free surface grids is constructed in the following sections. In Section 2.1 a relation between free surface height and pressure is obtained from a 2D potential flow perturbation analysis. As this relation is only valid in the wavenumber

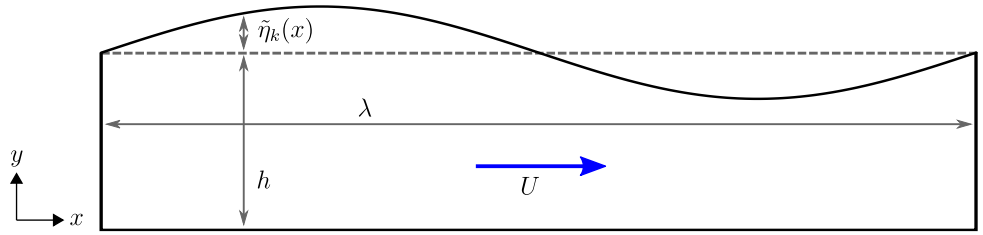

Figure 1: Flow over a horizontal plate with sinusoidal perturbation $\tilde{\eta}$ of the free surface.

domain, it is approximated with linear shape functions and consequently transformed to the spatial domain using the convolution theorem in Section 2.2. The spatial domain relation is then discretized in Section 2.3 to construct the surrogate model. In Section 2.4 the surrogate model is modified to deal with the problematic discretization of high wavenumbers.

2.1 Wavenumber domain relation between free surface pressure and height

A 2D free surface is considered, whose position is determined by a height function $\eta(x)$. The inviscid steady free surface flow over a horizontal plate as shown in Fig. 1 has a flat free surface as solution, as this fulfills all free surface boundary conditions: the velocity is parallel to the free surface and the pressure is constant. A relation between sinusoidal perturbations of free surface height and pressure has been derived by Demeester et al. [7] and is summarized here. The free surface height is perturbed with

$$\tilde{\eta}_k(x) = a \sin(kx + \theta) \tag{1}$$

with $k = 2\pi/\lambda$ the wavenumber, θ the phase angle and a the amplitude which must be small relative to the wavelength λ and flow depth h. The corresponding pressure perturbation $\tilde{p}_k(x)$ is given by a proportional relation:

$$\tilde{p}_k(x) = L(k)\, \tilde{\eta}_k(x) \qquad \text{with} \qquad L(k) = \rho g \left(\mathrm{Fr}^2 \frac{kh}{\tanh kh} - 1 \right). \tag{2}$$

ρ is the density, g the gravitational acceleration and $\mathrm{Fr} = U/\sqrt{gh}$ the flow Froude number based on the average velocity U. For subcritical flow ($\mathrm{Fr} < 1$) L can become zero, so that there is no one-to-one relation between pressure and height perturbations. This complicates the steady free surface problem, requiring additional conditions to be added in order to get a unique free surface solution. The original free surface method [6] handles this, but in this paper only the case of supercritical flow ($\mathrm{Fr} > 1$) is considered.

Functions in the spatial domain such as $p(x)$ can be transformed to the wavenumber domain using the Fourier transform $\mathscr{F}$, and transformed back using the inverse transform $\mathscr{F}^{-1}$. These are defined as

$$\mathscr{F}\left\{g(x)\right\}(k) = \int_{-\infty}^{\infty} g(x)\; e^{-ikx} dx \tag{3}$$

$$\mathscr{F}^{-1}\left\{G(k)\right\}(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(k)\; e^{ikx} dk \tag{4}$$

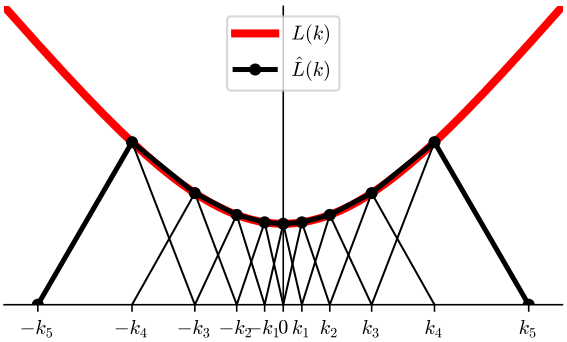

Figure 2: Example of original L and approximation $\hat{L}$.

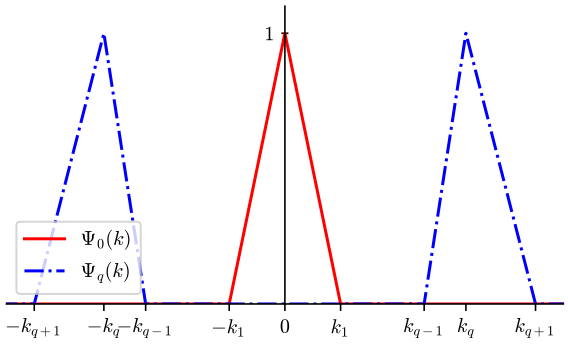

Figure 3: Definition of linear shape functions Ψ_0 and Ψ_q.

Arbitrary –non-sinusoidal– perturbations in the spatial domain then have a wavenumber domain representation

$$\tilde{H}(k) = \mathscr{F}\left\{\tilde{\eta}(x)\right\}(k), \tag{5}$$

$$\tilde{P}(k) = \mathscr{F}\left\{\tilde{p}(x)\right\}(k). \tag{6}$$

In the wavenumber domain the relation between height and pressure perturbations is

$$\tilde{P}(k) = L(k)\,\tilde{H}(k). \tag{7}$$

This relation is based on a perturbation analysis with potential flow. For more general free surface flows (large perturbations, viscid fluid, different geometry) this relation is not valid, but it can be used to construct a surrogate model which approximates the flow behavior.

2.2 Spatial domain relation based on convolution theorem

The relation between pressure and height perturbations in the wavenumber domain (Eq. 7) is not very practical, as it requires that signals are transformed to the wavenumber domain and back. However, the convolution theorem can be used to transform the relation itself to a spatial domain form. With the convolution defined as

$$(f * g)(x) = \int_{-\infty}^{\infty} f(\tau)g(x-\tau)\,d\tau, \tag{8}$$

the convolution theorem states that the Fourier transform of a convolution is the product of the Fourier transforms, i.e.

$$\mathscr{F}\left\{(f * g)(x)\right\} = \mathscr{F}\left\{f(x)\right\} \cdot \mathscr{F}\left\{g(x)\right\}. \tag{9}$$

Applying the convolution theorem to the wavenumber domain relation from Eq. (7) leads to the spatial domain relation

$$\tilde{p}(x) = (l * \tilde{\eta})(x) \qquad \text{with} \qquad l(x) = \mathscr{F}^{-1}\left\{L(k)\right\}(x). \tag{10}$$

The inverse Fourier transform of $L(k)$ is not trivial. First of all, $L(k)$ must be extended to negative wavenumbers in an even way, see Fig. 2 where $L(-k) = L(k)$. This ensures that its

inverse Fourier transform $l(x)$ is real. Furthermore L must be bandwidth limited, i.e. it must go to zero at a certain maximum wavenumber. If this is not done, L is infinitely wide in the wavenumber domain and will therefore have an asymptote in the spatial domain.

The inverse Fourier transform of the bandwidth limited L is not known, but it can be approximated by linear shape functions $\Psi_0(k)$ and $\Psi_q(k)$ as illustrated in Fig. 2. The approximate signal is denoted with a hat, so that

$$\hat{L}(k) = L(0)\,\Psi_0(k) + \sum_{q=1}^{q_{\max}} L(k_q)\,\Psi_q(k) \tag{11}$$

with $q_{\max}$ depending on the free surface discretization. The linear shape funtions Ψ_0 and Ψ_q are defined in Fig. 3. Ψ_0 is a triangle centered around $k = 0$ and depends only on k_1. Ψ_q consists of two triangles which constitute an even function and depends on k_q, k_{q-1} and k_{q+1}. Writing out Ψ_q with $q = 0$ and the condition to be even, gives $2\Psi_0$ and not Ψ_0. That is the reason why the central shape function Ψ_0 is considered separately. To shorten notation, the difference between two neighboring wavenumbers is denoted as $\Delta k_q = k_{q+1} - k_q$. The inverse transforms of $\Psi_0(k)$ and $\Psi_q(k)$ are respectively $\psi_0(x)$ and $\psi_q(x)$ and are known:

$$\psi_0(x) = \frac{1 - \cos k_1 x}{\pi k_1 x^2} \tag{12}$$

$$\psi_q(x) = \frac{(\Delta k_q + \Delta k_{q-1})\cos k_q x - \Delta k_q \cos k_{q-1} x - \Delta k_{q-1} \cos k_{q+1} x}{\pi \Delta k_q \Delta k_{q-1} x^2} \tag{13}$$

Thanks to the linearity of the Fourier transform, the inverse transformation of $\hat{L}$ is given by

$$\hat{l}(x) = \mathscr{F}^{-1}\left\{\hat{L}(k)\right\}(x) = L(0)\,\psi_0(x) + \sum_{q=1}^{q_{max}} L(k_q)\,\psi_q(x). \tag{14}$$

This results in an approximate spatial domain relation between height and pressure perturbations:

$$\tilde{p}(x) \approx \left(\hat{l} * \tilde{\eta}\right)(x) \tag{15}$$

Two important choices remain regarding the construction of $\hat{l}$. Firstly the approximation of L is determined by the choice of the wavenumbers k_q with $q \in [0, q_{\max}]$ in Eq. 11. Secondly $\hat{l}$ will need to be limited to a certain domain when it is discretized. More specifically, a cut-off value x_{co} must be chosen for each term in Eq. (14) so that it can be set to zero for $|x| > x_{\text{co}}$. In the remainder of this section it is shown that these two choices are related and how they can be made.

A closer look is taken at ψ_0 and ψ_q to see where the cut-off is best made for each function. The inverse of the central shape function Ψ_0 can be rewritten as

$$\psi_0(x) = \frac{k_1}{2\pi}\,\text{sinc}^2\,\frac{k_1 x}{2}. \tag{16}$$

A sinc^2 function is plotted in Fig. 4: it damps out with $1/x^2$, and has zeroes at locations $i\pi/\alpha$ with i a non-zero integer. Note that the derivative is also zero in these points, so the zeroes

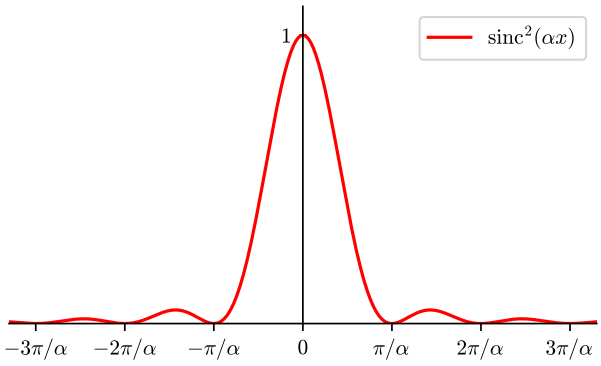

Figure 4: The sinc^2 function.

seem like good cut-off locations for ψ_0. The inverse transform of a general shape function Ψ_q can be rewritten as

$$\begin{aligned}\psi_q(x) =& \frac{\cos k_q x}{2\pi}\Delta k_{q-1}\,\text{sinc}^2\,\frac{\Delta k_{q-1}x}{2}\\ &+\frac{\cos k_q x}{2\pi}\Delta k_q\,\text{sinc}^2\,\frac{\Delta k_q x}{2}\\ &-\frac{k_q}{\pi}\,\text{sinc}\,k_q x\,(\text{sinc}\,\Delta k_{q-1}x - \text{sinc}\,\Delta k_q x)\,.\end{aligned} \tag{17}$$

The first and second term in this expression consist of a sinc^2 function with a low wavenumber multiplied with a cosine with higher wavenumber (namely k_q). The sinc^2 factor acts as an envelope function as it were. A good location for the cut-off could be where the zeroes of the sinc^2 factors coincide, namely locations $x_{\text{co}} = i_1 2\pi/\Delta k_{q-1} = i_2 2\pi/\Delta k_q$. From this expression it follows that the zeroes only coincide when $i_2/i_1 = \Delta k_q/\Delta k_{q-1}$ with i_1 and i_2 non-zero integers. This means that $\Delta k_q/\Delta k_{q-1}$ must be rational in order for the first two terms to have a common zero. Moreover, it can be shown that at these points the third term in Eq. (17) is also zero and has a zero derivative.

Choosing the ratio $\Delta k_q/\Delta k_{q-1}$ to be 1/1 seems the obvious choice, but it is not the best one. With this choice the envelope of ψ_q stays the same, i.e. for high wavenumber phenomena –which are typically more localized– the signal stays very wide in the spatial domain. It is more logical to have a narrower ψ_q for higher wavenumbers, which corresponds to a wider Ψ_q in the wavenumber domain. Good choices for the ratio $\Delta k_q/\Delta k_{q-1}$ are accordingly 2/1 and 3/2. For these values, ψ_q is plotted in Fig. 5 with $q = 4$, together with its three terms (Eq. (17)). Note that x_{co} is located at the first location where ψ_q and its derivative are both zero.

2.3 Discretization of spatial domain relation to construct surrogate model

In this section discrete variables will be introduced, which requires different notation: all matrices will be denoted by bold symbols and their elements with subscript indices.

Eq. (10) gives a relation between continuous functions $\tilde{\eta}(x)$ and $\tilde{p}(x)$. For use as a surrogate model, this relation must be discretized. Height and pressure perturbations have discrete counterparts which are denoted by the column vectors $\tilde{\boldsymbol{\eta}}, \tilde{\boldsymbol{p}} \in \mathbb{R}^{n\times 1}$ which contain the values

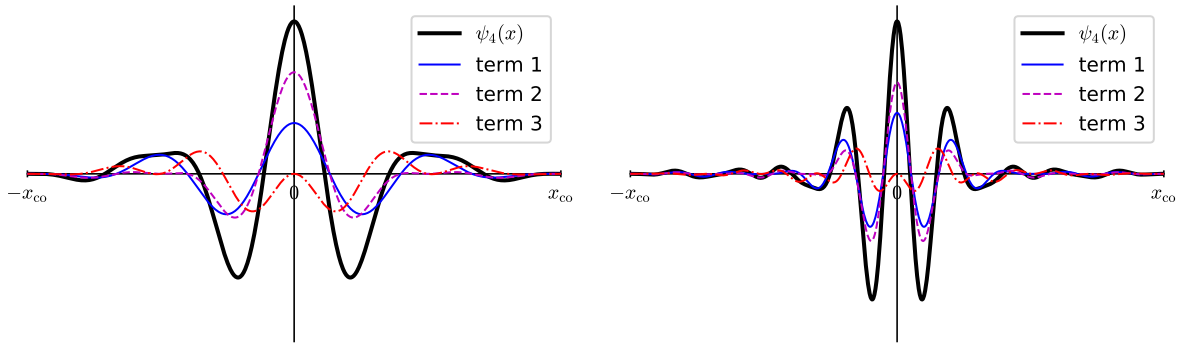

Figure 5: ψ_q for $q = 4$, using ratios $\Delta k_q/\Delta k_{q-1}$ of 2/1 (left) and 3/2 (right). The three terms of ψ_q from Eq. (17) are also plotted separately.

that correspond to positions $\boldsymbol{x} \in \mathbb{R}^{n\times 1}$. Discretization in the x-direction can be non-uniform, but it is assumed that the points are ordered from inlet to outlet, with $\boldsymbol{x}_0$ the inlet position. A surrogate model $\boldsymbol{F} \in \mathbb{R}^{n\times n}$ is now constructed such that

$$\tilde{\boldsymbol{p}} = \boldsymbol{F}\,\tilde{\boldsymbol{\eta}}. \tag{18}$$

The pressure $\tilde{\boldsymbol{p}}_i$ in point $\boldsymbol{x}_i$ is found by discretizing Eq. (15). Using the trapezoidal rule, the convolution becomes

$$\tilde{\boldsymbol{p}}_i = \sum_{j=-\infty}^{+\infty} \tilde{\boldsymbol{\eta}}_j \cdot \hat{l}\,(\boldsymbol{x}_i - \boldsymbol{x}_j) \cdot \frac{\boldsymbol{x}_{j+1} - \boldsymbol{x}_{j-1}}{2}. \tag{19}$$

Although the sum is taken from $-\infty$ to $+\infty$, this does not pose a problem as the region where $\hat{l}$ is non-zero is limited and the sum reduces to a limited number of terms. By comparing the expressions in Eqs. (18) and (19), the elements of $\boldsymbol{F}$ can be identified:

$$\boldsymbol{F}_{i,j} = \hat{l}\,(\boldsymbol{x}_i - \boldsymbol{x}_j) \cdot \frac{\boldsymbol{x}_{j+1} - \boldsymbol{x}_{j-1}}{2} \tag{20}$$

A problem is present near the boundaries of the domain: it is possible that j falls outside the allowed range $[0, n[$. The solution is to extend the height $\tilde{\boldsymbol{\eta}}$ past the boundary. This extension can be even or odd; the best choice is case dependent, as will be explained later. The result is that some elements of F get additional contributions of a form similar to Eq. (20).

2.4 Adaption of surrogate model for high wavenumbers

The highest wavenumber which can be represented on the free surface grid depends on the local discretization of that grid. The highest wavenumber in each node is collected in the grid wavenumber vector $\boldsymbol{k}_{\text{grid}} \in \mathbb{R}^{n\times 1}$, defined as

$$\boldsymbol{k}_{\text{grid}_i} = \frac{2\pi}{\boldsymbol{x}_{i+1} - \boldsymbol{x}_{i-1}}. \tag{21}$$

The discretization of $\hat{l}$ becomes problematic for these wavenumbers. On the one hand, it is important that these wavenumbers are taken into account, so $k_{q_{\max}}$ –which is determined locally–

must be at least as high as the grid wavenumber. On the other hand, it is possible to get aliasing in the discretization of $\hat{l}$, especially for irregular spacing of the points on the free surface. This leads to a wrong result of the convolution.

The solution is to use a filter matrix $\boldsymbol{W} \in \mathbb{R}^{n \times n}$ in the spatial domain to filter out the wavenumbers close to $\boldsymbol{k}_{\text{grid}}$. For the lower wavenumbers the convolution matrix $\boldsymbol{F}$ is still used, but now $k_{q_{\max}}$ can be chosen lower than $\boldsymbol{k}_{\text{grid}}$ to avoid aliasing. The high wavenumbers which were filtered out are treated with a factor L corresponding to $\boldsymbol{k}_{\text{grid}}$. This is written as a diagonal matrix $\boldsymbol{L}_{\text{grid}}$ defined as

$$\boldsymbol{L}_{\text{grid}_{i,i}} = L(\boldsymbol{k}_{\text{grid}_i}). \tag{22}$$

For the wavenumbers that are slightly lower than $\boldsymbol{k}_{\text{grid}}$, $\boldsymbol{L}_{\text{grid}}$ is an overestimation. This reduces the accuracy of the surrogate model, but will still give a stable update in the quasi-Newton iterations (i.e. the change in free surface height will be underestimated, not overestimated). The adapted surrogate model $\boldsymbol{F}^*$ is constructed from the original model $\boldsymbol{F}$ as

$$\boldsymbol{F}^* = \boldsymbol{F}\,\boldsymbol{W} + \boldsymbol{L}_{\text{grid}}(\boldsymbol{I}_n - \boldsymbol{W}) \tag{23}$$

with $\boldsymbol{I}_n$ the identity matrix.

The filter $\boldsymbol{W}$ must remove wavenumbers which are high with respect to the grid resolution, i.e. its cut-off wavenumber is chosen with respect to $\boldsymbol{k}_{\text{grid}}$ and not a reference which is physically meaningful to the flow. This means a filter kernel must be calculated only once and then put on every row of the matrix $\boldsymbol{W}$. At the boundaries, an even or odd extension must be made in the same way as was done for constructing $\boldsymbol{F}$. For the kernel, a windowed-sinc filter is used as described by Smith [8]. A Blackman window is used as this provides low passband and stopband ripples for reasonable rol-off. The kernel is based on two parameters: the cut-off wavenumber and the kernel length.

3 QUASI-NEWTON SOLUTION METHOD FOR STEADY FREE SURFACE FLOW

The free boundary problem which presents itself when the water-air interface is represented by surface fitting, was introduced in Section 1. Using the distribution of free surface boundary conditions proposed in Section 1, the free surface discretization introduced in Section 2.3, and a non-linear black-box flow solver $\boldsymbol{\mathcal{F}}$, the problem may be stated as:

> given the flow solver $\boldsymbol{\mathcal{F}}(\boldsymbol{\eta}) = \boldsymbol{p}$ which fulfills the KBC and tangential DBC, find $\boldsymbol{\eta}$ so that the normal DBC $\boldsymbol{p} = p_{\text{cst}}\mathbf{1}$ is fulfilled.

$\mathbf{1}$ denotes the all-ones vector. The KBC and tangential DBC are applied at the free surface by modeling it as a free-slip wall in the flow solver.

Starting from an initial guess $\boldsymbol{\eta}^0$ (a superscript denotes the iteration index), this problem can be solved iteratively with a quasi-Newton method: an approximate Jacobian $\widehat{\boldsymbol{\mathcal{F}}'}$ of the flow solver $\boldsymbol{\mathcal{F}}$ is used to calculate a new free surface height $\boldsymbol{\eta}$:

$$\widehat{\boldsymbol{\mathcal{F}}'}\,\Delta\boldsymbol{\eta}^m = p_{\text{cst}}\mathbf{1} - \boldsymbol{p}^m \tag{24}$$

with $\Delta\boldsymbol{\eta}^m = \boldsymbol{\eta}^{m+1} - \boldsymbol{\eta}^m$ and $\boldsymbol{p}^m = \boldsymbol{\mathcal{F}}(\boldsymbol{\eta}^m)$. The pressure p_{cst} is unknown and not of interest, as only the gradient of the pressure appears in the incompressible Navier-Stokes equations. To remove it from the system, Eq. (24) is split in two parts

$$\begin{cases} \widehat{\boldsymbol{\mathcal{F}}'}\,\Delta\boldsymbol{\eta}_{\text{a}}^m = -\boldsymbol{p}^m \\ \widehat{\boldsymbol{\mathcal{F}}'}\,\Delta\boldsymbol{\eta}_{\text{b}}^m = p_{\text{cst}}\mathbf{1} \end{cases} \tag{25}$$

so that $\Delta\boldsymbol{\eta}^m = \Delta\boldsymbol{\eta}_{\text{a}}^m + \Delta\boldsymbol{\eta}_{\text{b}}^m$. The first part can be solved once an expression for the approximate Jacobian is known (see further). The second part cannot be solved as p_{cst} is not known. However, the perturbation analysis from Section 2.1 predicts that $\Delta\boldsymbol{\eta}_{\text{b}}^m$ should also be constant, which implies that $\Delta\boldsymbol{\eta}_{\text{b}}^m$ determines the (average) flow height. In a practical case however, the flow height will usually be imposed at the inlet (or alternatively the outlet) by requiring that $\boldsymbol{\eta}_0 = h$. Using this condition instead of p_{cst} to determine $\Delta\boldsymbol{\eta}_{\text{b}}^m$ gives

$$\Delta\boldsymbol{\eta}_{\text{b}}^m = \left(h - \Delta\boldsymbol{\eta}_{\text{a},0}^m\right)\mathbf{1} \tag{26}$$

The subscript notations a and b will not be used any further as the contribution by $\Delta\boldsymbol{\eta}_{\text{b}}^m$ can simply be written as a correction based on $\Delta\boldsymbol{\eta}_{\text{a}}^m$.

The Jacobian of the flow solver can be approximated using the surrogate model $\boldsymbol{F}^*$ developed in Section 2, and is accordingly denoted

$$\widehat{\boldsymbol{\mathcal{F}}'}_{\text{sur}} = \boldsymbol{F}^*. \tag{27}$$

To stabilize the iterations and accelerate convergence, a second approximation of the Jacobian is constructed using the IQN-ILS algorithm by Degroote et al. [9], originally developed to improve convergence in partitioned fluid-structure interaction simulations. Flow solver inputs and outputs from previous iterations are collected in the matrices

$$\boldsymbol{V}^m = \begin{bmatrix} \Delta\boldsymbol{\eta}^{m-1} & \cdots & \Delta\boldsymbol{\eta}^0 \end{bmatrix}, \tag{28}$$

$$\boldsymbol{W}^m = \begin{bmatrix} \Delta\boldsymbol{p}^{m-1} & \cdots & \Delta\boldsymbol{p}^0 \end{bmatrix}. \tag{29}$$

These are used to construct a low-order approximate Jacobian with a least-squares technique as

$$\widehat{\boldsymbol{\mathcal{F}}'}_{\text{IQN}}^m = \boldsymbol{W}^m {\boldsymbol{R}^m}^{-1} {\boldsymbol{Q}^m}^T \qquad \text{with} \qquad \boldsymbol{V}^m = \boldsymbol{Q}^m \boldsymbol{R}^m \tag{30}$$

the economy-size QR-decomposition of $\boldsymbol{V}^m$. This second approximate Jacobian improves while the iterations progress and more information is stored in $\boldsymbol{V}^m$ and $\boldsymbol{W}^m$. The two approximations now have to be combined. While $\widehat{\boldsymbol{\mathcal{F}}'}_{\text{sur}}$ is a full rank Jacobian approximation, $\widehat{\boldsymbol{\mathcal{F}}'}_{\text{IQN}}^m$ only affects the part of $\Delta\boldsymbol{\eta}^m \in \text{range}(\boldsymbol{V})$, which is equal to $\boldsymbol{Q}^m {\boldsymbol{Q}^m}^T \Delta\boldsymbol{\eta}^m$. $\widehat{\boldsymbol{\mathcal{F}}'}_{\text{sur}}$ is then used for the remaining part of $\Delta\boldsymbol{\eta}^m$. The full expression for the approximate Jacobian is

$$\widehat{\boldsymbol{\mathcal{F}}'}^m = \widehat{\boldsymbol{\mathcal{F}}'}_{\text{IQN}}^m \boldsymbol{Q}^m {\boldsymbol{Q}^m}^T + \widehat{\boldsymbol{\mathcal{F}}'}_{\text{sur}} \left(\boldsymbol{I}_n - \boldsymbol{Q}^m {\boldsymbol{Q}^m}^T\right) \tag{31}$$

$$= \boldsymbol{W}^m {\boldsymbol{R}^m}^{-1} {\boldsymbol{Q}^m}^T + \boldsymbol{F}\left(\boldsymbol{I}_n - \boldsymbol{Q}^m {\boldsymbol{Q}^m}^T\right). \tag{32}$$

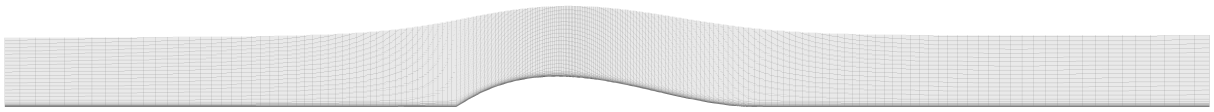

Figure 6: Final solution for flow over object. The free surface grid has a ratio between largest and smallest cell of 10.

Algorithm 1 presents the complete iterative solution method. The convergence criterion is based on the pressure residual r_p. With $d = \boldsymbol{x}_{n-1} - \boldsymbol{x}_0$ (assumed positive), the average pressure $\bar{p}$ is computed as

$$\bar{p} = \frac{1}{d} \sum_{i=1}^{n-1} (\boldsymbol{x}_i - \boldsymbol{x}_{i-1}) \left(\frac{\boldsymbol{p}_i + \boldsymbol{p}_{i-1}}{2} \right). \tag{33}$$

The pressure residual is then defined as the root mean square value of $(\boldsymbol{p} - \bar{p})$:

$$r_p = \frac{1}{d} \sum_{i=1}^{n-1} (\boldsymbol{x}_i - \boldsymbol{x}_{i-1}) \left(\frac{\boldsymbol{p}_i + \boldsymbol{p}_{i-1}}{2} - \bar{p} \right)^2 \tag{34}$$

Algorithm 1 Quasi-Newton method for 2D supercritical steady free surface flow.

$m = 0$
$\boldsymbol{p}^0 = \boldsymbol{\mathcal{F}}\left(\boldsymbol{\eta}^0\right)$
while $r_p^m > \varepsilon$ **do** ▷ Eq. (34)
  if $m > 0$ **then**
    construct $\boldsymbol{V}^m$, $\boldsymbol{W}^m$
    QR-decomposition $\boldsymbol{V}^m = \boldsymbol{Q}^m \boldsymbol{R}^m$
  end if
  construct $\widehat{\boldsymbol{\mathcal{F}}'}^m$ ▷ Eq. (32)
  solve $\widehat{\boldsymbol{\mathcal{F}}'}^m \Delta\boldsymbol{\eta}^m = -\boldsymbol{p}^m$
  $\Delta\boldsymbol{\eta}^m \mathrel{+}= (h - \Delta\boldsymbol{\eta}_0^m)\,\mathbf{1}$
  $\boldsymbol{\eta}^{m+1} = \boldsymbol{\eta}^m + \Delta\boldsymbol{\eta}^m$
  $m = m + 1$
  $\boldsymbol{p}^m = \boldsymbol{\mathcal{F}}\left(\boldsymbol{\eta}^m\right)$
end while

4 NUMERICAL RESULTS

The 2D flow over an obstacle as shown in Fig. 6 is used as test case. Experimental data was collected by Cahouet [10] and the case has been used to evaluate several free surface methods [3, 11, 12] for steady flow. The shape of the obstacle is described by

$$y_\mathrm{b} = \frac{27}{4} \frac{H_\mathrm{b}}{L_\mathrm{b}^3} x\,(x - L_\mathrm{b})^2 \quad \text{for} \quad 0 \le x \le L_\mathrm{b} \tag{35}$$

with $L_\mathrm{b} = 0.42\,\mathrm{m}$ the length and $H_\mathrm{b} = 0.042\,\mathrm{m}$ the height of the obstacle. The inlet water depth $h = 0.09545\,\mathrm{m}$, the Froude number $\mathrm{Fr} = 2.05$. The boundary conditions are: a velocity profile

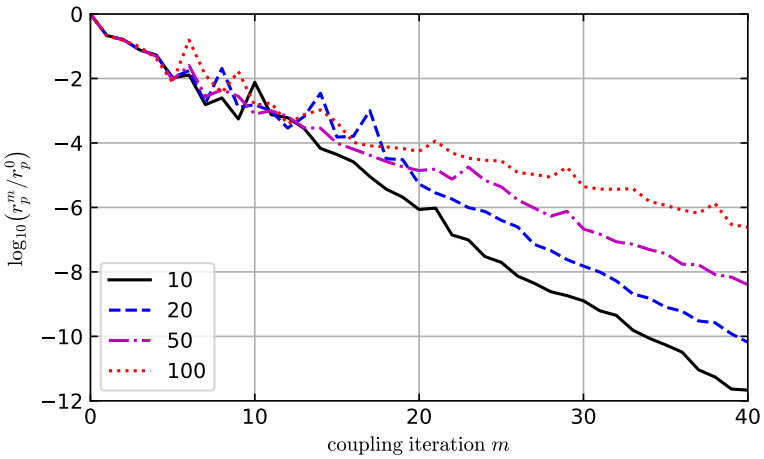

Figure 7: Normalized pressure residual of free surface method on different meshes. The legend shows the ratio between the largest and smallest free surface cells.

[10] at the inlet, a hydrostatic pressure outlet, a no-slip wall at the bottom and a free-slip wall at the free surface. A second order upwind scheme is used for all convection terms. Turbulence modeling is done with the $k\omega$-SST model. The initial height of the free surface is $\boldsymbol{\eta}^0 = h\,\mathbf{1}$.

The mesh which is used is structured, but not equally spaced. In the y-direction, the bottom boundary layer is resolved by the mesh, resulting in $1 < y^+ < 5$. In the x-direction mesh stretching is used to have coarse cells near the inlet and outlet of the domain and fine cells at the wave crest as can be seen in Fig. 6. In all simulations, the coarsest cell has length $L_\mathrm{b}/20$. The ratio between coarsest and finest cell is varied: ratios 10, 20, 50 and 100 are tested.

For the surrogate model a filter $\boldsymbol{W}$ with cut-off frequency 0.15 (which corresponds to $k_\mathrm{co}/k_\mathrm{grid} = 0.3$) and with kernel length 41 is used. For approximating L, a ratio $\Delta k_q/\Delta k_{q-1} = 1.5$ is used with $k_1 = 2.5\pi/L_\mathrm{b}$. To avoid aliasing q_max is chosen in each point $\boldsymbol{x}_i$ as the largest integer so that $3\,k_{q_\mathrm{max}} \leq \boldsymbol{k}_{\mathrm{grid}_i}$. For both $\boldsymbol{F}$ and $\boldsymbol{W}$, the inlet domain is extended in an odd way and the outlet domain in an even way, as these choices gave the best convergence.

Fig. 7 shows the residual r_p as defined in Eq. (34) for simulations with the four different meshes. In all simulations r_p decreases exponentially and in a low number of iterations. For larger ratios (i.e. finer cells), convergence becomes a bit slower.

5 CONCLUSIONS

A quasi-Newton method for efficiently solving 2D steady free surface flow has been developed recently. It offers the advantage of being compatible with black-box flow solvers. The original method uses an approximate Jacobian based on the one hand on a surrogate model of the flow solver, and on the other hand on flow solver inputs and outputs from previous iterations. Due to the formulation of the surrogate model, it can only be applied when the free surface grid is uniform. In this paper a new surrogate model is constructed, which allows the method to be extended to meshes with stretched free surface grids. This surrogate model is obtained by transforming a relation between perturbations of the free surface height and pressure from the

wavenumber domain to the spatial domain using the convolution theorem. The method is tested by solving the free surface flow over an object. It converges exponentially and in a low number of iterations. When the free surface grid is refined (i.e. ratio of largest to smallest cell increases), convergence slows down a bit.

REFERENCES

[1] C. W. Hirt and B. D. Nichols, "Volume of fluid (VOF) method for the dynamics of free boundaries," *Journal of Computational Physics*, vol. 39, no. 1, pp. 201–225, 1981.

[2] M. Sussman, P. Smereka, and S. Osher, "A level set approach for computing solutions to incompressible two-phase flow," *Journal of Computational Physics*, vol. 114, no. 1, pp. 146–159, 1994.

[3] G. Tzabiras, "A numerical investigation of 2D, steady free surface flows," *International Journal for Numerical Methods in Fluids*, vol. 25, no. 5, pp. 567–598, 1997.

[4] S. Muzaferija and M. Perić, "Computation of free-surface flows using the finite-volume method and moving grids," *Numerical Heat Transfer*, vol. 32, no. 4, pp. 369–384, 1997.

[5] E. H. van Brummelen, H. C. Raven, and B. Koren, "Efficient numerical solution of steady free-surface Navier–Stokes flow," *Journal of Computational Physics*, vol. 174, no. 1, pp. 120–137, 2001.

[6] T. Demeester, E. H. van Brummelen, and J. Degroote, "An efficient quasi-Newton method for 2D steady free surface flow." Manuscript submitted for publication.

[7] T. Demeester, J. Degroote, and J. Vierendeels, "Stability analysis of a partitioned iterative method for steady free surface flow," *Journal of Computational Physics*, vol. 354, pp. 387–392, 2018.

[8] S. W. Smith, *The scientist and engineer's guide to digital signal processing*. California Technical Pub., 1999.

[9] J. Degroote, K.-J. Bathe, and J. Vierendeels, "Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction," *Computers & Structures*, vol. 87, no. 11, pp. 793–801, 2009.

[10] J. Cahouet, *Etude numérique et expérimentale du problème bidimensionnel de la résistance de vagues non-linéaire*. Ecole Nationale Supérieure de Techniques Avançées, 1984.

[11] E. H. van Brummelen and A. Segal, "Adjoint shape optimization for steady free-surface flows," *International Journal for Numerical Methods in Fluids*, vol. 40, no. 3-4, pp. 605–614, 2002.

[12] J. Wackers and B. Koren, "A surface capturing method for the efficient computation of steady water waves," *Journal of Computational and Applied Mathematics*, vol. 215, no. 2, pp. 618–625, 2008.