Towards a Fast Fitting Method for 3D Free Surface Flow

Toon Demeester[*], **E. Harald van Brummelen**[†], **Joris Degroote**[*,‡]

[*]Ghent University, Belgium
[†]Eindhoven University of Technology, the Netherlands
[‡]Flanders Make, Belgium
toon.demeester@ugent.be

1   Introduction

An important application where steady free surface flows appear is the calculation of the steady resistance of a ship. Often these flows are solved using a capturing method to represent the free surface, e.g. the volume-of-fluid method (Hirt and Nichols, 1981) or the level-set method (Sussman et al., 1994). A time-stepping scheme is usually employed to reach the steady state solution. This is inefficient, as transient phenomena at the free surface may take a large number of time-steps to disappear (van Brummelen et al., 2001).

The goal of this ongoing research is to develop a fast, steady iterative method to solve the steady free surface problem. The free surface is represented with a fitting technique, meaning that it lies along a deformable domain boundary. This has the additional advantage that the free surface can be accurately represented by a relatively low number of grid points (in contrast to capturing methods). The air phase is not taken into account, as its influence on the water phase is negligible due to its much lower density. For convenience of use, it is required that the new method can be used with a general purpose black-box flow solver.

Currently a 2D version of this method has been completed and its capabilities demonstrated: it gives good results and converges in a low number of iterations (Demeester et al.,). It is based on quasi-Newton iterations, where a surrogate model of the flow solver is used to approximate the Jacobian and a rank-one update is performed in each iteration except the first one. The first step in extending this method to 3D cases, is to construct a surrogate model for these flows. A strategy to construct the surrogate is outlined in this paper, and the performance of the resulting surrogate is tested.

2   2D steady free surface method

A short overview of the 2D steady free surface method is given here, for details see Demeester et al., . The steady free surface problem can be reduced to a root-finding problem: find the free surface height z so that the free surface pressure $p = \mathscr{F}(z) = 0$. $\mathscr{F}$ is a (black-box) flow solver in which the free surface boundary is implemented as a free-slip wall. It takes a vector $z \in \mathbb{R}^{n \times 1}$ containing the discretized free surface height as input, and returns a vector $p \in \mathbb{R}^{n \times 1}$ containing the discretized free surface pressure as output.

This root-finding problem is solved iteratively with a quasi-Newton method:

$$J \, \Delta z^j = -p^j \tag{1}$$

with superscript j the iteration's index and $\Delta z^j = z^{j+1} - z^j$. The approximate Jacobian J of the flow solver $\mathscr{F}$ consists of two parts: a full-rank surrogate model J_{sur} and a low-rank least-squares approximation J_{LS} based on input-output pairs of $\mathscr{F}$ of previous iterations. The least-squares approximation is based on the IQN-ILS algorithm used in partitioned fluid-structure interaction (Degroote et al., 2009). J_{sur} and J_{LS} are combined in such a way that there is no overlap between them.

A surrogate model J_{sur} is required, which gives a relation between perturbations of the free surface height and pressure: $J_{\mathrm{sur}} \Delta z = \Delta p$. For this purpose, a perturbation analysis in the Fourier domain was performed analytically on a basic free surface flow (Demeester et al., 2018). The result was a relation between sinusoidal perturbations of the free surface height and pressure ($z_k(x) \sim p_k(x) \sim \sin(kx + \theta)$):

$$L(k) \cdot z_k(x) = p_k(x) \qquad \text{with} \qquad L(k) = \rho g \left(\mathrm{Fr}^2 \frac{kh}{\tanh kh} - 1 \right) \tag{2}$$
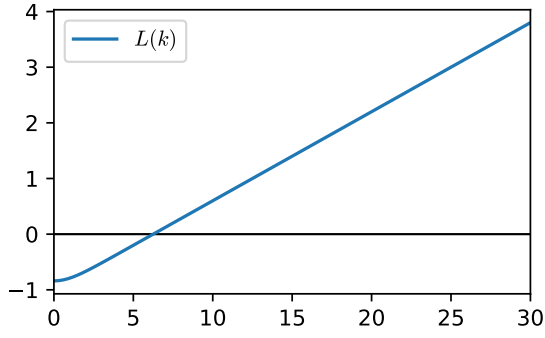
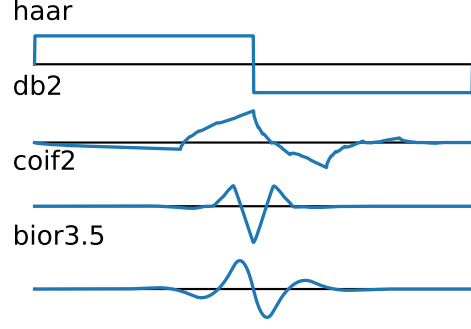Fig. 1: Factor $L(k)$ from Eq. (2) for Fr $= 0.4$, $\rho = g = h = 1$.

Fig. 2: Some examples of mother wavelets, from top to bottom: Haar, Daubechies 2, Coiflets 2, Biorthogonal 3.5 (see also PyWavelets, 2019).

with k the wavenumber, ρ the density, g the gravitational acceleration, h the depth of the flow, Fr $= U/\sqrt{gh}$ and U the average velocity of the flow.

If both z and p were transformed to the frequency domain (with the Discrete Fourier Transform, DFT), a surrogate model based on Eq. 2 would be a diagonal matrix. Getting J_{sur} in the spatial domain takes more effort: two methods based on the Fourier decomposition were proposed. Demeester et al., describes a J_{sur} for uniform free surface grids based on orthogonal projectors while Demeester et al., 2019 describes a J_{sur} for stretched free surface grids based on the convolution theorem.

Fig. 1 plots L from Eq. (2) as a function of k, for subcritical flow conditions (Fr < 1). The curve of L has a zero at a certain wavenumber. This is the steady gravity wave, which has a phase velocity equal (opposite) to the flow velocity and therefore appears to be stationary. This wave can appear with arbitrary amplitude and phase in the solution, so that the steady free surface problem has infinite solutions. Correspondingly, the surrogate model becomes singular. In order to find a unique free surface solution, additional conditions are added which force the free surface to be flat at the flow inlet. These extra equations are solved simultaneously with Eq. (1) for Δz^j using a least-squares solver.

3   Switching to the wavelet domain

For extending the current method to 3D free surface flows, the first requirement is a new surrogate model which describes perturbations of a 3D free surface. This adds two difficulties: new physics (effects perpendicular to the flow direction, see Sec. 3.3) and an additional dimension. The latter is a problem if a surrogate would be constructed in the same way as for 2D flows: due to the much larger number of unknowns at the free surface, it would be very expensive to construct the surrogate model and perform matrix operations on it. This can be seen as follows: imagine a free surface with r grid points in each direction. For a 2D flow, this would give $z \in \mathbb{R}^{r \times 1}$ and $J \in \mathbb{R}^{r \times r}$. For a 3D flow, this would give $z \in \mathbb{R}^{r^2 \times 1}$ and $J \in \mathbb{R}^{r^2 \times r^2}$. Matrix calculations for a 2D case on a fine mesh (e.g. $r = 1000$) pose no problems, but would be infeasible for a 3D case, even for coarser meshes. A different approach is required.

Waves are typically very smooth for steady free surface flows: it should be possible to make good approximations of the free surface height and pressure using m smooth basis functions, with m much smaller than the number of free surface grid points n.

The Fourier basis may seem the obvious choice: it uses sines of varying wavenumber to represent signals, it can be calculated efficiently with the FFT (complexity $O(n \log n)$) and the high wavenumber components can be removed to get a low-dimensional representation. However, the Fourier basis has a serious disadvantage: its base signals correspond to a single wavenumber and are consequently not at all localized in space. As a consequence, local phenomena like a single standing wave are hard to represent and problems occur at the boundaries of the domain. If a discontinuity is present between boundary values, high frequency components appear. When the $n - m$ highest modes are removed, an oscillation appears in the reconstructed low-order signal due to the Gibbs phenomenon. As an alternative to the
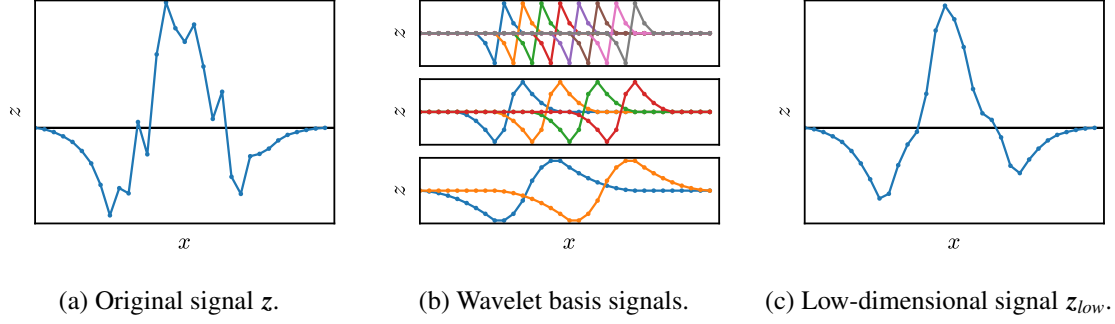
(a) Original signal z. (b) Wavelet basis signals. (c) Low-dimensional signal z_{low}.

Fig. 3: Making a low-dimensional approximation of a signal z by decomposing it into the Biorthogonal 3.1 wavelet basis and removing the smallest scale.

Fourier basis, a wavelet basis is chosen.

3.1 The Discrete Wavelet Transform

For a thorough mathematical description of wavelets, the reader is referred to the reference book by Mallat, 2008, here only the relevant concepts are explained qualitatively. Wavelets are "small waves", which are localized in both the spatial and wavenumber domains. This makes wavelets very suitable for describing local phenomena, e.g. in image compression.[1] The Discrete Wavelet Transform (DWT) can be computed very efficiently using cascaded filter banks, giving a complexity of $O(n)$. An orthogonal wavelet basis can be formed by translating and dilating a single well-chosen *mother wavelet*.[2] The choice of mother wavelet depends on the application, as they can have very different properties. Some examples are given in Fig. 2.

Fig. 3 gives an example of the 1D DWT to explain which basis signals are removed to reduce the number of free surface variables. In Fig. 3a an arbitrary signal z is shown. Calculating the DWT of z means that z is decomposed with respect to the basis signals shown in Fig. 3b. These are discretized versions of the dilated and translated mother wavelet. Three *scales* are used here: the basis signals of scale 0 are shown at the top, those of scale 1 in the center and those of scale 2 at the bottom. By dilating the mother wavelet each time with a factor 2, the basis signals of different scales will correspond to different frequency-bands. The coefficients of z with respect to the basis vectors of scale 0 are collected in a vector $\mathbf{Z}_0 \in \mathbb{R}^{8 \times 1}$. Analogously the coefficients corresponding to scale 1 and scale 2 are collected in respectively $\mathbf{Z}_1 \in \mathbb{R}^{4 \times 1}$ and $\mathbf{Z}_2 \in \mathbb{R}^{2 \times 1}$. The vector $\mathbf{Z} = [\mathbf{Z}_0 \quad \mathbf{Z}_1 \quad \mathbf{Z}_2]^T$ is then the DWT of z. The coefficients of the smallest scales can be set to zero to get a low-dimensional approximation of z. In this example $\mathbf{Z}_{low} = [\mathbf{0} \quad \mathbf{Z}_1 \quad \mathbf{Z}_2]^T$ for which the inverse DWT is z_{low}, shown in Fig. 3c. Note that only three scales are used here and that boundary effects are not taken into account.

3.2 A wavelet-based surrogate

The DWT is introduced in the quasi-Newton iterations to solve the steady free surface problem. For a non-uniform free surface grid, the pressure $\boldsymbol{p}^j$ is first interpolated to a uniform grid. The DWT is then calculated, giving the array of coefficients $\boldsymbol{P}^j = [\boldsymbol{P}_0^j \; \boldsymbol{P}_1^j \; \cdots]^T$. The s smallest scales are then removed, reducing the number of variables from n to m. This also eliminates most interpolation errors, as these are mainly present in the small scales (high wavenumbers). The low-dimensional pressure in the wavelet domain $[\boldsymbol{P}_s^j \; \boldsymbol{P}_{s+1}^j \; \cdots]^T$ is then used in the quasi-Newton equation of Eq. (1), which is solved for the new

---

[1]The well-known JPEG format uses the Discrete Cosine Transform (closely related to the Discrete Fourier Transform) for lossy image compression. The newer JPEG 2000 format instead uses the Discrete Wavelet Transform. It was developed to improve compression performance, but never became widely adopted. https://blog.ansi.org/2018/07/why-jpeg-2000-never-used-standard-iso-iec/

[2]In fact, wavelets cannot form a complete (bi)orthogonal basis: a wavelet always has average value 0, i.e. it corresponds to a frequency band which does not include $k = 0$. The smallest wavelet scale corresponds to the grid size, the largest wavelet scale depends on the number of scales used in the DWT. To capture all wavenumbers between 0 and the largest wavelet scale, a *scaling function* is introduced. These are very similar to wavelets, but have a non-zero average. A dilated and translated mother wavelet in combination with a translated scaling function, can give a complete (bi)orthogonal basis.
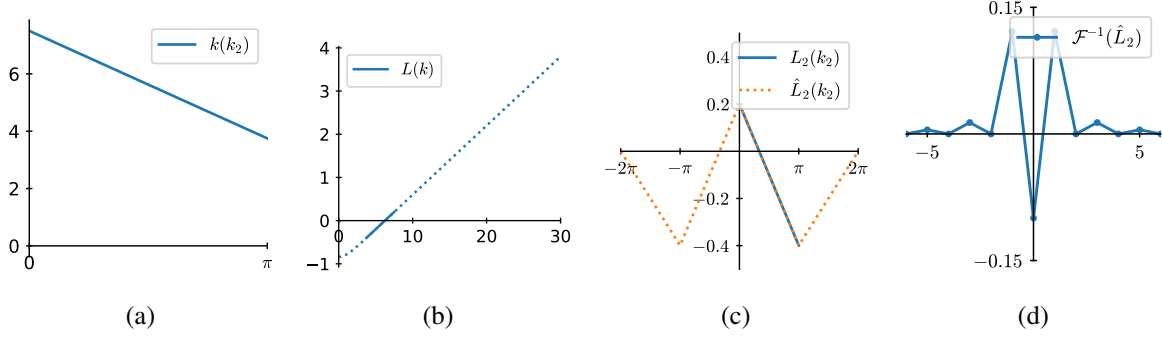
Fig. 4: Example of steps to construct $\boldsymbol{J}_{\text{sur}}$ for scale $s = 2$.

free surface height $[\boldsymbol{Z}_s^{j+1} \ \boldsymbol{Z}_{s+1}^{j+1} \ \cdots]^T$. The resulting array is padded with zeros, transformed back with the inverse DWT and interpolated to the original grid to get z^{j+1}.

As the quasi-Newton equation is now solved in the wavelet domain for a 3D surface, as opposed to in the spatial domain for 2D, the surrogate model $\boldsymbol{J}_{\text{sur}}$ must be adapted to this low-dimensional basis (no adaption is required for the least-squares approximation $\boldsymbol{J}_{\text{LS}}$). The detailed construction of the surrogate model does not fit in the scope of this paper, but an overview of the required steps is given, accompanied by an example in Fig 4. Construction of the 3D surrogate follows the same procedure, but is even more cumbersome to describe.

1. The Biorthogonal 3.5 wavelet basis is used, see Fig. 2. This wavelet is smooth, odd, compact in the spatial domain and well localized in the wavenumber domain. The basis signals are not orthogonal, but using the DWT and inverse DWT, signals have a unique decomposition and reconstruction in this basis.

2. Eq. (2) describes the behavior of sinusoidal perturbations of z with wavenumber k. On the other hand, in the wavelet domain there may be sinusoidal perturbations of $\boldsymbol{Z}_s$ with wavenumber k_s. It has been observed that each sinusoid with wavenumber k_s in the wavelet domain, corresponds approximately to a sinusoid with wavenumber k in the spatial domain. As a consequence, for each scale a function $L_s(k_s)$ can be derived from $L(k)$. This $L_s(k_s)$ relates sinusoidal perturbations $\boldsymbol{Z}_s$ and $\boldsymbol{P}_s$. In Fig. 4a the relation $k(k_s)$ is shown for scale $s = 2$ of a case corresponding to Fig. 4b. In that picture, $L(k)$ is shown with the range of wavenumbers corresponding to scale 2 emphasized. From the curves in Figs. 4a and 4b follows $L_2(k_2)$ in Fig. 4c.

3. In theory, with $\mathcal{F}$ the DFT operator, $\boldsymbol{P}_s$ can be calculated for an arbitrary $\boldsymbol{Z}_s$ by making a detour through the Fourier domain: $\boldsymbol{P}_s = \mathcal{F}^{-1}(L_s \cdot \mathcal{F}(\boldsymbol{Z}_s))$. However, going through the Fourier domain is not very practical. Instead, it is possible to replace the multiplication in the Fourier domain by a convolution in the wavelet domain, using the convolution theorem: $\mathcal{F}^{-1}(L_s \cdot \mathcal{F}(\boldsymbol{Z}_s)) \sim \mathcal{F}^{-1}(L_s) * \boldsymbol{Z}_s$. The left and right hand sides are equal up to a constant factor which depends on the definition of the applied DFT.

4. The inverse transform of L_s is not known. However, each wavelet scale corresponds to only a small band of wavenumbers k, so that all L_s are close to linear functions of k_s. Replacing L_s by a linear approximation $\hat{L}_s$, the inverse Fourier transform can be calculated analytically. $\mathcal{F}^{-1}(\hat{L}_s)$ damps out with $1/x^2$, so a good approximation can be made using only the 7 or 11 central points. Fig. 4c shows the linear approximation $\hat{L}_2(k_2)$ and Fig. 4d its inverse DFT, restricted to 11 points.

5. The discrete convolution $\mathcal{F}^{-1}(\hat{L}_s) * \boldsymbol{Z}_s$ can be seen as a window $\mathcal{F}^{-1}(\hat{L}_s)$ moving over the signal $\boldsymbol{Z}_s$. This process can be described by a matrix product $\boldsymbol{J}_s \cdot \boldsymbol{Z}_s$ where the window is inserted in each row of $\boldsymbol{J}_s$, centered around the diagonal.

6. This process must be done for each wavelet scale which was not set to zero. The individual matrices $\boldsymbol{J}_s$ are put in one big matrix $\boldsymbol{J}_{\text{sur}} \in \mathbb{R}^{m \times m}$.

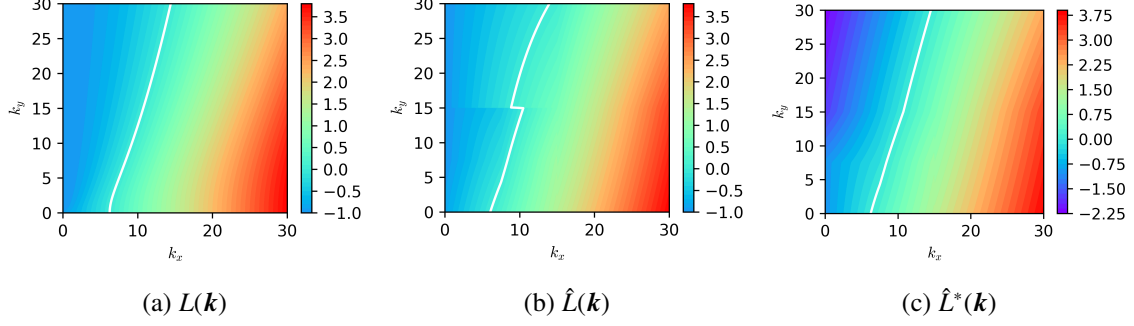(a) $L(\boldsymbol{k})$ (b) $\hat{L}(\boldsymbol{k})$ (c) $\hat{L}^*(\boldsymbol{k})$

Fig. 5: The function $L(\boldsymbol{k})$ as defined in Eq. (3) for Fr $= 0.4, \rho = g = h = 1$. Two approximations used in constructing $\boldsymbol{J}_{\text{sur}}$ are also shown. The zero-contours are shown in white.

3.3 Testing the new surrogate model

In the last step before the surrogate model can be constructed, the relation of Eq. (2) between perturbations of the free surface height and pressure must be extended to 3D flows. Sinusoidal perturbations now have components in the flow direction (k_x) and perpendicular to the flow direction (k_y). Two notations are introduced to make the equations more compact: $\boldsymbol{k} = (k_x, k_y)$ and $k^2 = k_x^2 + k_y^2$. For a sinusoidal height perturbation $z_{\boldsymbol{k}}(x, y) \sim \sin(k_x x + k_y y + \theta)$, the linearized Euler equations can be solved (with suitable boundary conditions), yielding the relation:

$$L(\boldsymbol{k}) \cdot z_{\boldsymbol{k}}(x, y) = p_{\boldsymbol{k}}(x, y) \qquad \text{with} \qquad L(\boldsymbol{k}) = \rho g \left(\text{Fr}^2 \frac{k_x^2}{k^2} \frac{kh}{\tanh kh} - 1 \right) \qquad (3)$$

Notice that for $k_y \to 0$, Eq. (3) reduces to Eq. (2). An example of this relation is plotted in Fig. 5a.

The 3D surrogate model is tested as follows: a height perturbation $z = \sin(k_x x + k_y y + \theta)$ is transformed to the wavelet domain, the small scales are omitted, then it is converted to a pressure using $\boldsymbol{J}_{\text{sur}}$ and finally transformed back to the spatial domain. The obtained pressure $\boldsymbol{p}$ can be compared to the theoretically correct pressure $L(\boldsymbol{k}) \cdot z$. This process is illustrated in Fig. 6 for $\boldsymbol{k} = (3, 5)$. The original domain has $n = 250\,000$ free surface grid points, which is reduced to $m = 7747$ coefficients in the wavelet domain by removing the 3 smallest scales.

From Fig. 6c, the effective factor $L_{sur}(\boldsymbol{k})$ obtained using the surrogate model can be estimated. This should be as close as possible to the correct factor $L(\boldsymbol{k})$. An interesting way of comparing L and L_{sur} for a given test, is by looking at the *amplification factor* $\mu = z^{j+1}/z^j$. This factor expresses how an error component z will converge/diverge, if only the surrogate model $\boldsymbol{J}_{\text{sur}}$ is used as approximate Jacobian in Eq. (1). It can be shown that $\mu = 1 - L/L_{sur}$. In theory, iterations will converge for a certain error mode z when $|\mu| < 1$ for that mode. The amplification factor is calculated for a whole range of values of $\boldsymbol{k}$ and the result is shown in Fig. 7a.

In the region where L is close to zero, the result is not good. The reason can be found in Fig. 5b: the approximation $\hat{L}$ which is used for constructing $\boldsymbol{J}_{\text{sur}}$ is not everywhere a good approximation of L, due to the important non-linearity in L at small k_x. In order to get a better amplification factor, a different approximation of L is made. The original $\hat{L}$ is a bi-linear interpolation from a number of sampled points of L. A new approximation $\hat{L}^*$ (see Fig. 5c) is obtained by adapting the values in some of these points so that there is a better correspondence between the zero-lines in L and $\hat{L}^*$. As a consequence, the difference between L and $\hat{L}^*$ may be larger than between L and $\hat{L}$ in places, but overall μ is much better as can be seen in Fig. 7b.

4 Conclusion

In this paper, wavelets were used to make a low-dimensional approximation of smooth signals. Furthermore, for biorthogonal wavelets a relation was found between the frequency content of a signal in the spatial domain and in the wavelet domain. Using this relation, it is possible to construct surrogate models to approximate the behavior of 2D and 3D steady free surface flows. In 2D the surrogate model
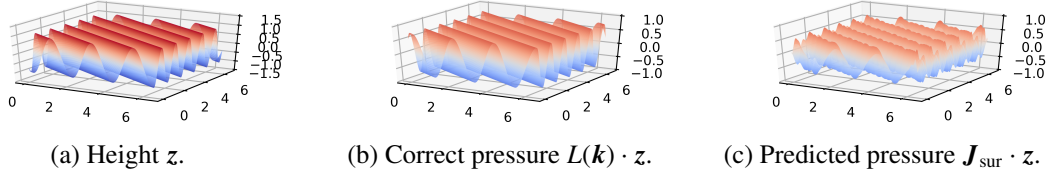
(a) Height z.   (b) Correct pressure $L(\boldsymbol{k}) \cdot \boldsymbol{z}$.   (c) Predicted pressure $\boldsymbol{J}_{\text{sur}} \cdot \boldsymbol{z}$.

Fig. 6: Testing the surrogate model for a wave with $k = (3, 5)$.

(a) Based on original $\hat{L}(\boldsymbol{k})$.   (b) Based on adapted $\hat{L}^*(\boldsymbol{k})$.
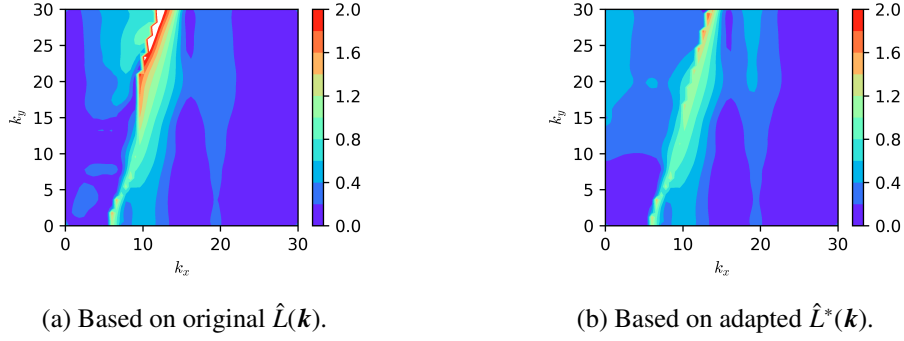
Fig. 7: $|\mu(\boldsymbol{k})|$, the absolute value of the amplification factor.

performed very well when solving steady free surface flow with a quasi-Newton method (not discussed in the paper). For 3D, some first tests were done to check the validity of the surrogate model.

In the near future, more extensive testing of the 3D surrogate model will be done, by applying it to non-sinusoidal signals and comparing the result to CFD simulations. When the surrogate model performs satisfactorily, it will be used to extend the 2D steady free surface method to 3D.

References

Degroote, J., Bathe, K.-J., and Vierendeels, J. (2009). Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Computers & Structures*, 87(11):793–801.

Demeester, T., Degroote, J., and Vierendeels, J. (2018). Stability analysis of a partitioned iterative method for steady free surface flow. *Journal of Computational Physics*, 354:387–392.

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

Demeester, T., van Brummelen, E. H., and Degroote, J. (2019). Extension of a fast method for 2D steady free surface flow to stretched surface grids. In *Proceedings of the 8th International Conference on Computational Methods in Marine Engineering*.

Hirt, C. W. and Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225.

Mallat, S. (2008). *A Wavelet Tour of Signal Processing: The Sparse Way*. Academic Press.

PyWavelets (2019). Wavelet browser. http://wavelets.pybytes.com/. Accessed: 05-09-2019.

Sussman, M., Smereka, P., and Osher, S. (1994). A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146–159.

van Brummelen, E. H., Raven, H. C., and Koren, B. (2001). Efficient numerical solution of steady free-surface Navier–Stokes flow. *Journal of Computational Physics*, 174(1):120–137.