

PREDICTION OF EQUILIBRIUM SCOUR HOLES USING OPTIMIZATION PROCEDURES

Xerxes Mandviwalla¹, Jesper Roland Kjærgaard Qvist² and Erik Damgaard Christensen³

This paper presents an optimization procedure that finds the equilibrium scour depth under a pipeline. Even though much knowledge on scour is available for the most typical marine structures such as a vertical circular monopile of a horizontal pipelines the calculation of the scour depth complex and time-consuming as the transient solution is often modelled as well. In this paper we present a optimization procedure that combined with a computational fluid dynamics, and a model of the bed load finds the equilibrium shape of a scour hole. This can potentially speed up the calculation of the shape of the equilibrium scour hole with a factor of 100. However, it comes with a cost as we will not model the transition and the time scale of the scour hole development.

Keywords: Equilibrium scour depth; optimization of shape; bed load transport

INTRODUCTION

Scour around and adjacent to marine and coastal structures is an important design issue. The study of scour and design of scour protection is often based on theoretical considerations and hydraulic experiments. An introduction to the field is given in (Sumer and Fredsøe, 2002). During the last couple of decades computational fluid dynamics (CFD) has been used to model the scour development, see for instance (Roulund et al., 2005), (Baykal et al., 2015), and (Fuhrman et al., 2014). The use of CFD-models has shown promising results. They can predict transition from for instance a flatbed to a fully developed scour hole with good accuracy. A major drawback of such models is the long computational time needed for modelling the whole transition from flatbed to fully developed scour whole. These calculations can take weeks or in some cases even months.

As an alternative, only the depth of the scour whole could be modelled. In (Dixen et al., 2012) CFD was used to calculate scour/deposition rates for prescribed frustum shaped scour holes around a monopile. The comparison with experimental results was encouraging. The main objective is to model the scour depth over long time span as suggested in (Nielsen and Hansen, 2007). In this method, scour and back-filling rates are needed. These can be found from either experiments or computational fluid dynamics as for instance in (Fuhrman et al., 2014). To use the method outlined in (Dixen et al., 2012) an approximate shape of the scour hole is needed. This could be based on the equilibrium scour hole. In this study we used an optimization procedure to assess the shape of the equilibrium scour hole shape for horizontal pipelines above a sandy seabed.

The remaining part of this paper will described the methodology, which three parts the; the optimization procedure, the CFD-model and the bed-load model. This is followed by examples of finding the equilibrium scour depth under a pipe line and under tandem pipeline arrangement. Finally a chapter summarizes and concludes the study.

METHODOLOGY

Optimisation methodology

In order to find a find an optimal value of a continuous function $f(x)$, one can use a numerical technique such as Newton's method to iterate towards a local minimum. The method expresses a change in the function Δx towards a local extrema, with a second order expression. Considering the following Taylor series expansion of the function: Taylors formula gives an approximate solution to a function:

$$f(x + \Delta x) = f(x) + \Delta x^T \nabla f(x) + 0.5 \Delta x^T \nabla^2 f(x) \Delta x + O(\Delta x^3) \quad (1)$$

A function of Δx as $(\Delta x) = f(x + \Delta x)$ is defined, and extrema at $(\partial F(\Delta x) / \partial (\Delta x) = 0)$

¹Present address: DHI, Agern Alle 5, Dk-2970 Hørsholm, Denmark,

² DTU-MEK, Technical University of Denmark, Nils Koppels Alle 403, DK-2800, Kgs. Lyngby, Denmark

³ DTU-MEK, Technical University of Denmark, Nils Koppels Alle 403, DK-2800, Kgs. Lyngby, Denmark

One can equate the left-hand side to a function of $\Delta \mathbf{x}$ as $F(\Delta \mathbf{x}) = f(\mathbf{x} + \Delta \mathbf{x})$, then by taking the derivative of this function with respect to $\Delta \mathbf{x}$ and setting it to zero to find a extrema ($\frac{\partial F(\Delta \mathbf{x})}{\partial (\Delta \mathbf{x})} = \mathbf{0}$) yields:

$$0 = \nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x}) \Delta \mathbf{x} \quad (2)$$

setting $\mathbf{H} = \nabla^2 f(\mathbf{x})$, where H is the Hessian matrix, and by rearranging, results in the following:

$$\Delta \mathbf{x} = -\mathbf{H}^{-1} \nabla f(\mathbf{x}) \quad (3)$$

where $\Delta \mathbf{x}$ is the directional change towards the optima. New iterate $\mathbf{x}_2 = \mathbf{x}_1 + \alpha \Delta \mathbf{x}$, where α is a selected step size.

The following section presents an implementation of the Boyden-Fletcher-Goldfarb-Shanno (BFGS) method (Nocedal and Wright, 2006), (Arora, 2004), which is classified as a ‘quasi-Newton’ method, as the Hessian matrix (\mathbf{H}) is approximated at each iteration rather than directly calculated. The implementation here was performed in MATLAB[®], where the algorithm set-up, ran and analyzed OpenFOAM simulations on a high performance cluster to find a minimized *cost function* pertaining to an equilibrium scour bed.

The optimization problem was formulated such that a cost function or also termed an ‘objective function’ had to be minimized. In the case of the live scour bed, we proposed using the following:

$$f(\mathbf{x}) = \min \left\{ K \int_{-l}^l (|q_\infty - q(\mathbf{x})|) / |q_\infty| dx \right\} \quad (4)$$

Where q_∞ is the free-stream bed-load, $q(\mathbf{x})$ is the bed load along the flow direction, K is an integral gain coefficient, and $[l - l]$ is the extents of the bed. Other functions could also be considered such as minimizing the Exner equation along the bed, i.e. find a bed shape, which minimizes the integrated change of the bed height.

The optimal value of the cost function was found with the Boyden-Fletcher-Goldfarb-Shanno (BFGS) method, (Bonnans et al., 2006).he following outlines the implemented algorithm where:

1. The procedure is initialized with an iteration counter $k = 0$, and an initial shape of the bed with n design variables: $\mathbf{x}^0 = [x_1 \ x_2 \ x_3 \ \dots \ x_n]$. These design variables are input to a shape function that describe the shape of the seabed, which is outlined further later. At the initialization, the Hessian matrix is equated to an $n \times n$ identity matrix ($\mathbf{H}^0 = \mathbf{I}$). The gradient vector $\mathbf{c}^0 = \nabla f(\mathbf{x}^0)$, is then computed linearly by calculating a new value of the cost function by perturbing each of the design variable by ‘ η ’ (a user defined small step size of each design variable) as demonstrated for the design variable x_1 by:

$$\frac{\partial f}{\partial x_1} = \frac{(f([x_1^0 + \eta \ x_2^0 \ x_3^0 \ \dots \ x_n^0]) - f([x_1^0 \ x_2^0 \ x_3^0 \ \dots \ x_n^0]))}{\eta} \quad (5)$$

Calculating gradients requires ‘ n ’ OpenFOAM calculations to evaluate the gradients for each design variable. On a high performance cluster, this can be achieved in parallel.

2. Evaluate the norm of the gradient function, as a stopping criterion of the algorithm. If it is larger than a small tolerance continue ($\|\mathbf{c}^k\| > \epsilon$), else terminate.
3. Calculate the direction of descent by the following $\mathbf{d}^k = -(\mathbf{H}^k)^{-1} \mathbf{c}^k$, where k is the iteration number.
4. Evaluate the step size (α). This is achieved from a user defined array of 10 or so alpha values. After computation for each case, the step size value, which minimizes the cost function, is selected. Armijo’s rule (Arora, 2004) is one of many more formal approaches to finding the step size.
5. Update the design variables $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha \mathbf{d}^k$
6. Calculate the new gradient function $\mathbf{c}^{k+1} = \nabla f(\mathbf{x})$ based on a linear evaluation of the gradients as shown in eq. (5), now for the updated design variables.
7. Update the Hessian matrix using the following:

$$\begin{aligned} \mathbf{s}^k &= \alpha \mathbf{d}^k \\ \mathbf{y}^k &= \mathbf{c}^{k+1} - \mathbf{c}^k \\ \mathbf{D}^k &= \frac{\mathbf{y}^k \mathbf{y}^{(k)T}}{\mathbf{y}^k \cdot \mathbf{s}^k} \\ \mathbf{E}^k &= \mathbf{c}^k \mathbf{c}^{kT} / \mathbf{c}^k \cdot \mathbf{d}^k \end{aligned}$$

$$\mathbf{H}^{k+1} = \mathbf{H}^k + \mathbf{D}^k + \mathbf{E}^k$$

8. Increment the iteration number ($k = k + 1$) and Go to Step 2)

CFD-method

The CFD model was based on the open source CFD library/code OpenFOAM, (“OpenFOAM, The open source CFD toolbox,” 2018). The model solved the Navier-Stokes equations with a RANS turbulence model, and a sediment transport model.

The following outlines the setup for each OpenFOAM simulation, where the hydrodynamics and sediment transport is calculated for each alteration of the seabed. Fig. 1 shows the general geometry for the computational runs.

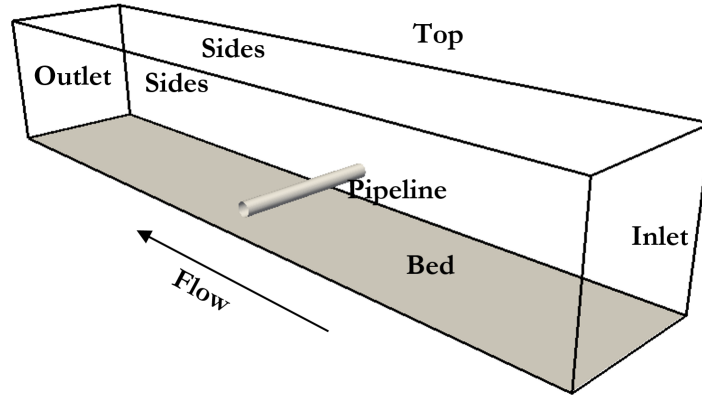


Fig. 1 Computational model domain for analyzing 2D pipeline scour

The boundary conditions applied to the ‘Inlet’ in the figure for the velocity and turbulence quantities was mapped from a separate precursor solution of a fully developed boundary layer, driven by a body force similar to the methodology described in (Petersen et al., 2020), while a zero-gradient condition was applied for the pressure ($\frac{\partial p}{\partial n} = 0$).

Pressure was set to a reference value of zero and zero gradient conditions were applied to all other variables in the ‘Outlet’. The ‘Sides’ were specified as ‘empty’ to enforce a two dimensional solution. The ‘Bed’ and ‘Cylinder’ walls were modelled with no-slip conditions and used the standard wall functions in OpenFOAM 2.2.2 for the turbulent quantities, with the $k-\omega$ SST model (Menter, 2003). A setup using the same turbulence model and wall conditions for forces on a cylinder were presented in (Petersen et al., 2020).

The ‘Top’ boundary was modelled as a friction-less lid with a slip condition. The dimensions of the domain are 20D from the pipeline center to the ‘Inlet’ and 30D to the ‘Outlet’, while the height was 10D tall. The diameter of the pipeline was $D = 0.1m$.

The SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm (Ferziger and Peric, 2002) was used to calculate the hydrodynamics. This iterates to a quasi-steady state, as vortex shedding behind the cylinder is expected, a true steady state cannot be attained. A model simulation is deemed ‘steady’ once an equilibrium of systematic vortex shedding pattern behind the cylinder is achieved. Using a steady-state solver is a practical measure, as model runs were comparably faster than the transient equivalents.

Bed load

The non-linear vectorial bed-load model (Kovacs and Parker, 1994) was implemented in OpenFOAM and used in the following analysis. Properties of the sediment modelled were as follows, sediment diameter, $d = 0.26 \text{ mm}$, specific weight, $s = 2.65$, and coulomb friction factor $\mu_c = 0.73$. This model will estimate the sediment transport.

Bed Shape and Shape Functions

Ideally the shape of the bed is modelled in a simplistic fashion, in order to minimise the number of design variables. Employment of two cubic Hermite style spline functions provides a good general form for a hole in 2D, applied in the following piece-wise fashion (Pedersen, 2016):

$$F(x, B) = 0 \quad \text{for } x < B_2 \quad (6)$$

$$F(x, B) = 2B_1 \left(\frac{x - B_2}{B_3 - B_2} \right)^3 - 3B_1 \left(\frac{x - B_2}{B_3 - B_2} \right)^2 \quad \text{for } B_2 > x > B_3 \quad (7)$$

$$F(x, B) = -2B_1 \left(\frac{x - B_4}{B_3 - B_2} \right)^3 + 3B_1 \left(\frac{x - B_2}{B_3 - B_2} \right)^2 \quad \text{for } B_3 > x > B_4 \quad (8)$$

$$F(x, B) = 0 \quad \text{for } x > B_4 \quad (9)$$

Fig. 2 illustrates the use of eqs. (6) to (9) to represent the shape of the bed under the cylinder. The design variables $[B_1 \ B_2 \ B_3 \ B_4]$ are plotted on the figure, where B_1 and B_3 is the amplitude and horizontal position of the hole respectively. The position of the shoulders of the hole are governed by B_2 and B_4 .

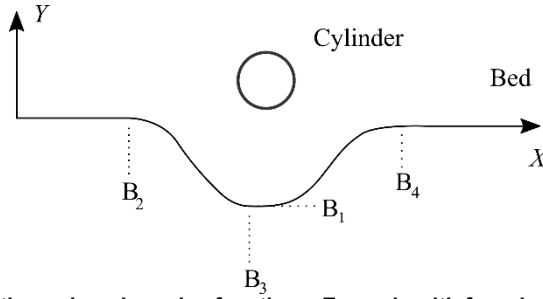


Fig. 2 Sketch of bed interpolation using piecewise functions. Example with four design variables.

The bed could also more generally comprise of piecewise polynomials evaluated as basis splines or 'B-splines'. Fig. 3 demonstrates the general approximation of a B-spline to the experimental scour hole from (Mao, 1986). The red dots in the figure show the control points used to manipulate the shape of the B-spline. Calculation of the B-spline was done via de Boors algorithm (De Boor, 1977), implemented in MATLAB® by (Hunyadi, 2012). For the optimisation procedure, the design variables are the control points, where the shape of B-spline can be manipulated. The five control points in the figure, will result in 10 design variables as each point has a vertical and horizontal coordinate.

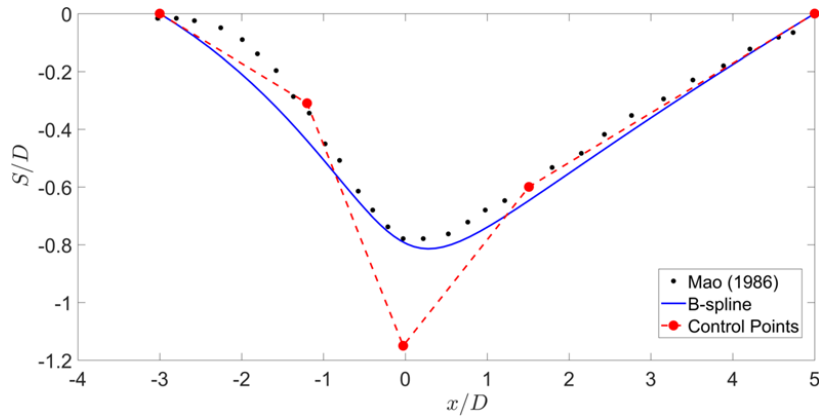


Fig. 3 Illustrates the use of B-splines, with 5 control points to approximate the scour hole from the experiments of (Mao, 1986).

Outline of the final algorithm

The BFGS algorithm as outlined in the previous section was code as a MATLAB[®]-script, for gradient evaluations of the cost function (Step 6) and step size evaluations (Step 4), new OpenFOAM runs needed to be set up, executed and analyzed automatically for the algorithm to proceed. OpenFOAM runs were set up on the high performance computing (HPC) cluster at DTU, such that the runs could be executed in parallel over many computational cores. The folder structure on the cluster was organized in terms of the iteration number from the algorithm (i.e. folder 'Iter1' will contain all the relevant simulations for that iteration). A seed simulation was set up, this contained the correct boundary conditions, solver options, numerical schemes used for each simulation. When the algorithm needed to create a new OpenFOAM run, it simply copied the seed folder and replaced the geometry files (to calculate a new shape of the bed). The shape of bed was interpolated using a selected shape function and converted to a Stereolithographic format (.stl), using a script available from the MathWorks user File Exchange (Holcombe, n.d.). The .stl file was then set in the OpenFOAM run directory and used to create a new computational mesh using the automated meshing tool snappyHexMesh. snappyHexMesh is standard in the OpenFOAM release and creates a predominately hexahedral element mesh, with cut cells used to conform the mesh to the geometry surface described by the .stl file. Layered elements at the boundary were added to the surface for better modelling of the boundary layers. Fig. 4 shows an example of a constructed computational mesh for a bed and pipeline. The .stl surfaces used for the bed and pipeline are shown in red. The graded boundary elements start with a cell center roughly situated at $y^+ = 50$ in the logarithmic boundary layer.

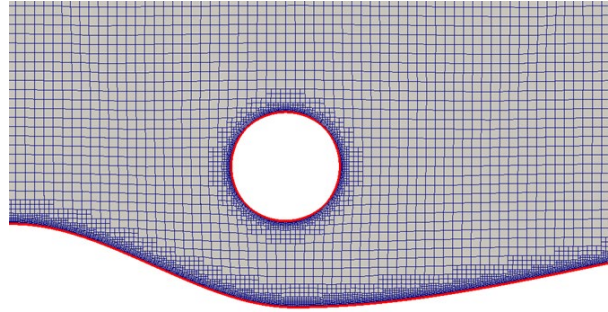


Fig. 4 Computational mesh with .stl surfaces in red for the bed and cylinder.

Fig. 5 shows the evaluation of the gradients of a cost function ((Step 6) at iteration 1 ($k=1$), for five design variables. The new perturbed geometries are created and transferred to the HPC cluster. Four new OpenFOAM simulations are created under the folder 'Iter 1', simply copying the 'SeedRun'. The new .stl geometries are then put into their respective simulation folders. The snappyHexMesh procedure executes on the cluster for each of the runs in order to create new computational meshes. On completion of the mesh creation, the runs are executed in parallel. The MATLAB[®] script continuously queries the cluster to check for completion of the simulations. Once finished, the bed load profiles from each run are transferred back to MATLAB[®] for the algorithm to calculate the gradients as presented in eq. (5) and proceed with the subsequent steps of the BFGS algorithm.

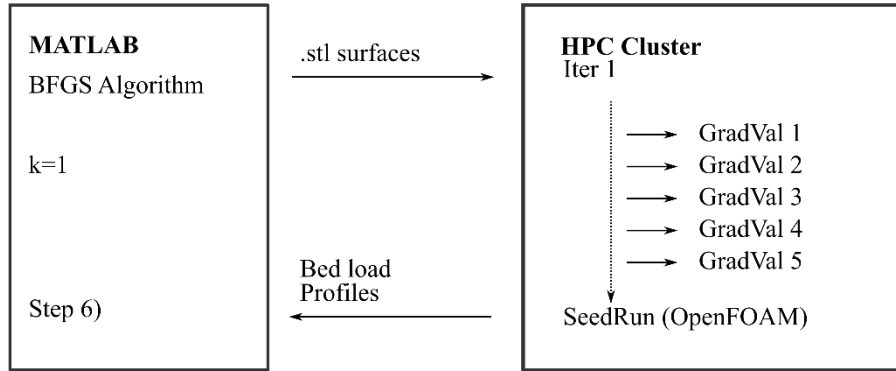


Fig. 5 Illustration of the evaluation of gradients of the cost function (Step 6) Section 3.2.1 for iteration 1 ($k=1$). Four simulations for each gradient evaluation is set up by copying the 'SeedRun', new geometries of the bed are passed by .stl surface from MATLAB®. New meshes are created and the models are run. On completion, the bed load profiles are transferred back to MATLAB® and the algorithm proceeds.

OPTIMISATION OF SCOUR UNDER A PIPELINE

The following presents test cases of the optimization routine. Fig. 6 shows the iteration of the BFGS algorithm for the bed shape represented by the Hermite interpolation as outlined in eqs. (6) to (9). An initial shape of the bed assumes a small hole under the pipeline, this is visible in the figure with the line corresponding to 'Iter 1'. An initial bed scour bed shape allows for a gap between the cylinder and the bed, which makes the meshing of the problem easier. Making an initial guess of a hole around a bottom-mounted structure could be made from viewing the extent of the shear stress amplification around the object on the bed.

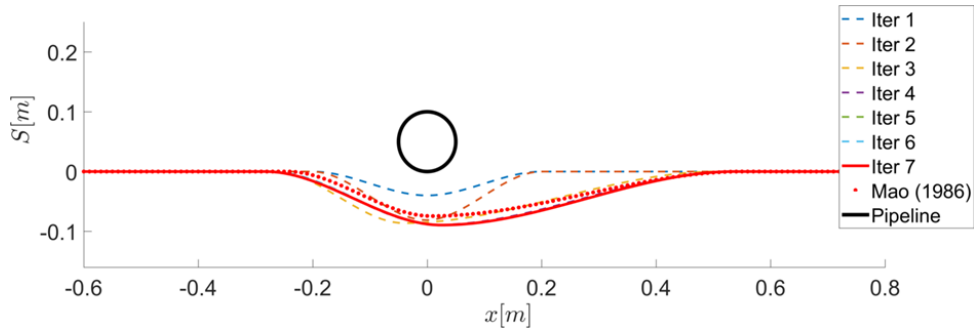


Fig. 6 Iterations towards equilibrium scour bed profile under a pipeline positioned at $e/D=0$ under current live bed conditions. Experiment profile from (Mao, 1986)

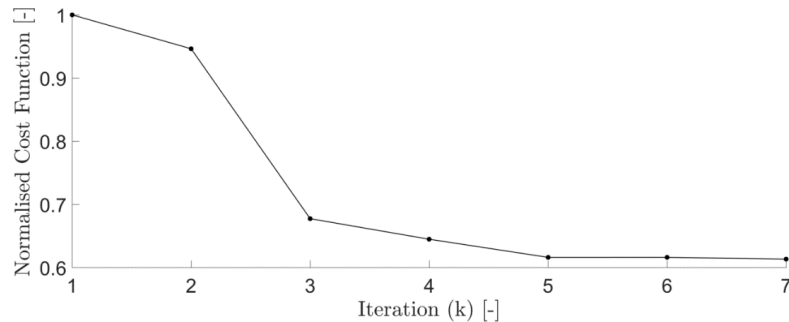


Fig. 7 Decrease of normalized cost function against iteration number for the simulation shown in Fig. 4.

From Fig. 6 it can be seen that the algorithm finds a direction of decent by bringing the shape of the hole downwards. This would be expected due to the constriction between the pipeline and the bed, resulting in high shear stress and consequently enhanced bed load transport. Proceeding iterations push the lee side of the boundary away from the pipeline, this is to reduce the enhanced bedload from the

shear stresses of the shed vortices from the pipeline. Progressive reduction of the cost function with each iteration is shown in Fig. 7.

Comparison of the equilibrium scour depth shows that the optimized depth is larger than the experiment of (Mao, 1986). However, it can be argued that it is close to the higher end of the scatter of equilibrium scour depths as presented in (Sumer and Fredsøe, 1990), where the mean scour depth and standard deviation are derived to be $S/D=0.6$, $\sigma/D=0.2$ (Sumer and Fredsøe, 2002). Generally, the shape of the hole was well captured.

Fig. 8 presents the optimization using B-splines, with five control points as demonstrated in Fig. 3. Here the shape of the final iteration also provides a reasonable comparison against the experiments. As there are more degrees of freedom than the optimization shown in Fig. 6, the front end of the scour hole has a slight convex bend, which would not be physical. Improvement of the algorithm could incorporate a constraint to the bed surface with regard to the angle of repose, by potentially implementing a 'sand-slide' routine as used in morphological simulations (Jacobsen, 2011). The progression of the cost function is shown in Fig. 9.

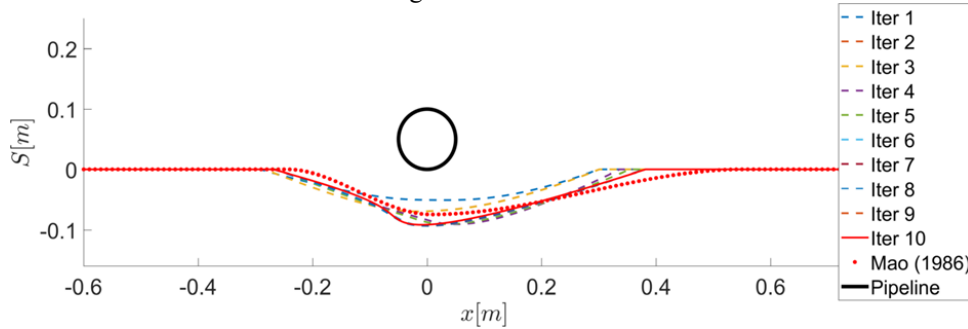


Fig. 8 Iterations towards equilibrium scour bed profile under a pipeline positioned at $e/D=0$ under current live bed conditions. Optimization uses B-splines. Experiment profile from (Mao, 1986)

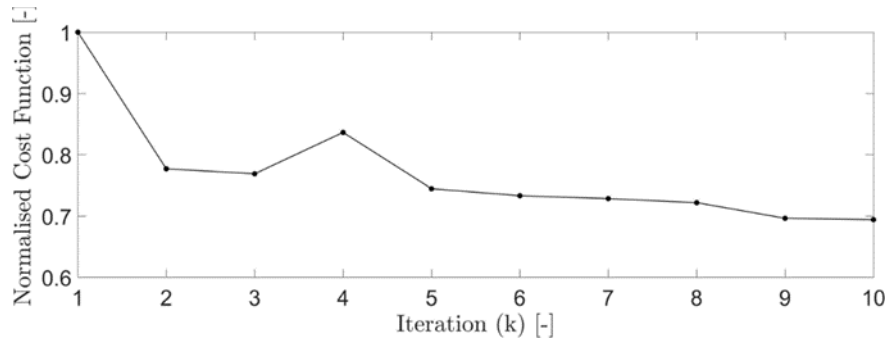


Fig. 9 Decrease of normalized cost function against iteration number for the simulation presented in Fig. 6.

Fig. 10 plots the optimization run for two pipelines in tandem, exposed to current in the live bed regime. The pipelines are positioned such that they are placed on the flatbed ($\frac{e}{D} = 0$) and are ($\frac{G}{D} = 3$), where G is the distance between the pipelines from their respective edges. The diameter of the pipelines was $D = 0.1m$, as used in the previous optimisation runs. The figure compares the optimisation run with experiments and numerical results for the problem presented in (Zhao et al., 2015). The numerical results are from a morphological CFD model, where both suspended and bed load were modelled. It is apparent from the figure that the optimization routine captures the scours depths well in comparison to the experiments. The small rise of the bed between the pipelines within hole observed from the experiments was not resolved by the optimization routine, or very well by the numerical results from the paper. It should be noted that such a bump in the bed could be the result of a transient development of the hole, which may not be possible to resolve at all by an optimization routine which seeks an equilibrium shape of the scour hole, essentially bypassing the realistic development of the hole. Fig. 11 shows the development of the cost function against iterations for the tandem problem.

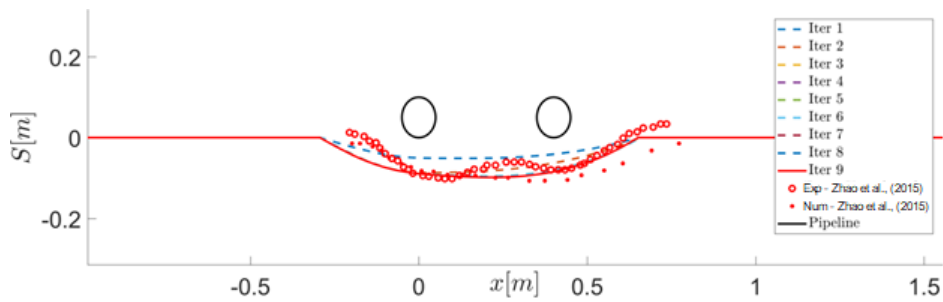


Fig. 10 Iterations towards equilibrium scour bed profile under two tandem pipelines positioned at $e/D=0$ from the bed and $G/D=3$ apart, under current live bed conditions. Optimization uses B-splines. Experiment and numerical work from (Zhao et al., 2015).

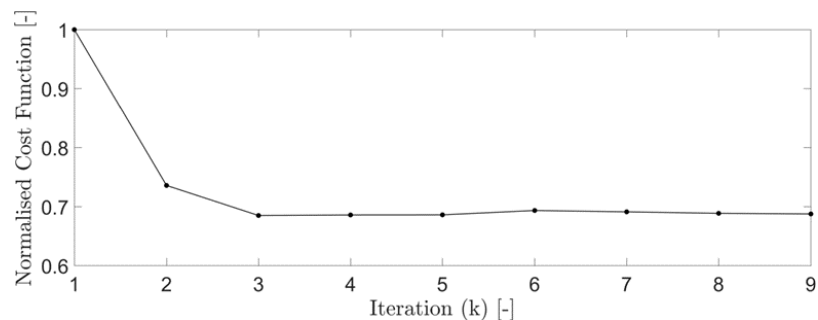


Fig. 11 Decrease of normalized cost function against iteration number for the simulation presented in Fig. 8

The optimization problems presented in this section took approximately 1-1.5 days to compute on the HPC cluster at DTU. To get an idea of the number of runs required, for example, if we were solving a problem with 10 design variables, each iteration would require 10 evaluations (OpenFOAM simulations) for determining of the gradients of the cost function. Furthermore, runs for determining the size of the step size (α), could mean an additional 10-15 simulations per iteration. As the computation ran on a cluster, the runs per iteration can be set off in parallel, but it does become apparent that a practice balance between the number of design variables and representation of the bed geometry must be kept in order to keep the routine efficient.

The routine at present has at times issues with robustness, where it can have difficulty finding search directions for extending the sides of the hole. However, the scour depth of the hole was found relatively effectively by the routine. It was found by the author that changes of the gain coefficient of the cost function ($K = 2$, eq. (4)) and refined step size evaluations appeared to improve the prediction of the hole width.

SUMMARY

To fast estimate the shape of a scour hole, scour development was formulated in terms of an optimization problem. The technique was applied in 2D for the scour around pipelines. Results showed that the method has potential to predict an equilibrium scour hole, in a relatively fast manner in comparison to a fully morphological CFD calculations. A series of B-splines described the bed form. We only incorporated bed-load in the calculations. Future work could include suspended load as well. Furthermore, the analysis could be extended to a transient solution using the PISO algorithm instead of the steady-state SIMPLE algorithm, to model the equilibrium hole shape under waves. This might require a phase averaging of the cost function over the wave periods, in order to find an equilibrium depth.

The shape functions such as the B-splines needs to be extended to 3D to form panels in order to describe a 3D scour problem, such as the scour around a monopile or 3D scour around a pipeline. The use of the BFGS method in 3D a balance must be attained between a reasonable number of design variables and adequate representation of the bed surface.

ACKNOWLEDGEMENT

The authors gratefully acknowledge financial support from Energinet.dk (ForskEL project 2013-1-68012: "Management of seabed and wind farm interaction").

REFERENCES

- Arora, J.S., 2004. Introduction to Optimum Design. *Introd. to Optim. Des.* 1–728. doi:10.1016/B978-0-12-064155-0.X5000-9
- Baykal, C., Sumer, B.M., Fuhrman, D.R., Jacobsen, N.G., Fredsøe, J., 2015. Numerical investigation of flow and scour around a vertical circular cylinder. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* 373, 20140104 1–21. doi:10.1098/rsta.2014.0104
- Bonnans, J.F., Gilbert, J.C., Lemaréchal, C., Sagastizábal, C.A., 2006. Numerical optimization: Theoretical and practical aspects, 2nd Editio. ed. Springer-Verlag, Berlin Heidelberg New York. doi:10.1007/978-3-540-35447-5
- De Boor, C., 1977. Package for calculating with B-splines. *SIAM J. Numer. Anal.* 14, 441–472. doi:10.1137/0714026
- Dixen, M., Lohmann, I.P., Christensen, E.D., 2012. METHOD TO PREDICT LONG TIME SPAN OF SCOUR AROUND OFFSHORE WIND, in: *International Conference on Coastal Engineering (ICCE)*. Santander, Spain, pp. 1–11.
- Ferziger, J.H., Perić, M., 2002. Computational Methods for Fluid Dynamics, 3rd ed. Springer-Verlag, Berlin Heidelberg New York.
- Fuhrman, D.R., Baykal, C., Mutlu Sumer, B., Jacobsen, N.G., Fredsøe, J., 2014. Numerical simulation of wave-induced scour and backfilling processes beneath submarine pipelines. *Coast. Eng.* 94, 10–22. doi:10.1016/j.coastaleng.2014.08.009
- Holcombe, S., n.d. stlwrite - write ASCII or Binary STL files [WWW Document]. [WWW Doc. MathWorks. 2018. URL <https://se.mathworks.com/MATLABcentral/fileexchange/20922-stlwrite-write-ascii-or-binary-stl-files>
- Hunyadi, L., 2012. B-Splines [WWW Document]. MathWorks.
- Jacobsen, N.G., 2011. A Full Hydro- and Morphodynamic Description of Breaker Bar Development (PhD Thesis). Technical University of Denmark.
- Kovacs, A., Parker, G., 1994. A new vectorial bedload formulation and its application to the time evolution of straight river channels. *J. Fluid Mech.* 267, 153–183.
- Mao, Y., 1986. The interaction between a pipeline and an erodible bed. Series Paper 39 (PhD Thesis). Technical University of Denmark.
- Nielsen, A.W., Hansen, E.A., 2007. Time-varying wave and current-induced scour around offshore wind turbines, in: *Proceedings of the ASME 2007 26th International Conference on Ocean, Offshore and Arctic Engineering*. San Diego, California, USA, pp. 399–408.
- Nocedal, J., Wright, S.J., 2006. Numerical Optimisation, Second Edi. ed. Springer.
- OpenFOAM, The open source CFD toolbox [WWW Document], 2018. URL <https://www.openfoam.com/>
- Pedersen, J.R.K., 2016. Optimized estimation of scour holes around marine structures. Technical University of Denmark.
- Petersen, T.U., Mandviwalla, X., Christensen, E.D., Tarp-Johansen, N.J., Rüdinger, F., 2020. Oscillatory loads on circular cylinder with secondary structures. *J. Fluids Struct.* 94, 1–22.
- Roulund, A., Sumer, B.M., Fredsøe, J., Michelsen, J., 2005. Numerical and experimental investigation of flow and scour around a circular pile. *J. Fluid Mech.* 534, 351–401. doi:10.1017/S0022112005004507
- Sumer, B.M., Fredsøe, J., 2002. The Mechanics of Scour in the Marine Environment, Revised ve. ed. World Scientific Publishing Co Pte Ltd.
- Sumer, B.M., Fredsøe, J., 1990. Scour below pipelines in waves. *J. Waterw. Port, Coastal, Ocean Eng.* 116, 307–323.
- Zhao, M., Vaidya, S., Zhang, Q., Cheng, L., 2015. Local scour around two pipelines in tandem in steady current. *Coast. Eng.* 98, 1–15. doi:10.1016/j.coastaleng.2015.01.001