Correcting for mesh size dependency in a regional model's representation of tidal turbines

Simon Waldman
ICIT, Heriot-Watt University
Orkney, UK
smw13@hw.ac.uk

Guillaume Genet
Institut National des
Science Appliquées
Strasbourg, France
guillaume.genet@insa-strasbourg.fr

Susana Bastón
ICIT, Heriot-Watt University
Orkney, UK
s.baston@hw.ac.uk

Jonathan Side
ICIT, Heriot-Watt University
Orkney, UK
j.c.side@hw.ac.uk

***Abstract*—When regional-scale hydrodynamic models use fine mesh sizes, such that the cross-section of a cell approaches that of a turbine, an error emerges in the calculation of turbine thrust. This error can be corrected using a method derived from actuator disk theory. We demonstrate this error, explain its source, and then present and test a new MATLAB package to correct for it. Although some minor mesh dependency remains after the correction, its effect is reduced by an order of magnitude.**

***Keywords*—numerical modelling, marine renewable energy, tidal energy, energy extraction, MIKE 3**

I. INTRODUCTION

Regional-scale hydrodynamic modelling has emerged as an essential tool for predicting both the potential performance of tidal stream energy installations [1], [2], and the environmental impacts that they may have [3]. A number of modelling systems now have the facility to represent tidal turbines and their effect on the flow.

Historically, regional hydrodynamic models have used cell sizes that are much larger than the diameter of a tidal turbine. In recent years, for applications related to tidal energy, the resolution of these models has increased such that the cross-sectional area of a cell in the model approaches that of a turbine rotor (*e.g.* [4], [5]). This increase in resolution is desirable as it permits more accurate representation of a channel and/or turbine array, which can make a significant difference to predicted current speeds [6]. However, at these high resolutions a mesh dependency emerges that can cause the effect of tidal turbines to be underestimated. This effect was recently demonstrated for a two-dimensional MIKE21 model, and a correction proposed, by Kramer *et al.* [7]. In this work we have adapted that correction to operate appropriately with a 3D model using a triangular mesh, and we have developed a MATLAB package to perform the correction for MIKE. Our work uses the flexible mesh (FM) version of MIKE 3 by DHI, but a similar error is likely to exist in other model codes that use the same approach to energy extraction and do not include a correction.

MIKE by DHI is a commercial hydrodynamic modelling suite commonly used in industry. The MIKE 3 Flow Model FM module solves the three-dimensional incompressible Reynolds-averaged Navier-Stokes equations under Boussinesq and hydrostatic pressure assumptions, using a cell-centred finite volume method [8]. Horizontal spatial discretization is on a flexible (unstructured) mesh. Vertical discretization uses σ layers, *i.e.* a constant number of equally-spaced terrain-following layers that change their thickness according to the depth of the water column. The "2012" version of MIKE was used for this work, with the latest available service packs. A later "2014" edition of MIKE has been released, but was unavailable for this project.

The structure of this paper is as follows: Section II explains the reason that the mesh dependency arises, and Section III demonstrates it. Section IV explains a correction for the error, and Section V describes the MATLAB package that was developed to perform this correction for MIKE models. Section VI presents tests of this package, and Section VII discusses conclusions.

II. SOURCE OF THE ERROR

MIKE 3 represents a tidal turbine as a sub-grid object. It calculates the retarding force that is exerted on the flow as a function of the flow velocity, the turbine's thrust coefficient, and the area of the rotor:

$$F = \frac{1}{2}\rho\alpha C_T A u^2 \tag{1}$$

where ρ is the density of the water, C_T is the thrust coefficient, A is the area of the rotor, u is the flow speed and α is a user-defined correction factor that is equal to 1 by default (simplified from [9], assuming that the turbine axis is aligned with the flow*). This force is applied in the horizontal mesh element that contains the turbine centre, and is equally split between vertical elements that intersect the rotor, but cannot be localised any further than that.

As water approaches a tidal turbine, it slows from its upstream speed u_0 (which is the "free stream" speed of the flow before it has begun to feel the effects of the turbine) to the speed at the turbine u_t.

By convention, the thrust coefficient is defined in terms of the free-stream velocity u_0. Therefore, it is u_0 that should be used

---

*MIKE actually allows for turbines that are not oriented into the flow by splitting the thrust coefficient into what the manual describes as drag and lift coefficients, to refer to axial and orthogonal components respectively. For the purposes of this work we make the assumption that the turbine is aligned with the flow, and thus replace C_D with the more commonly used C_T and assume that $C_L = 0$.

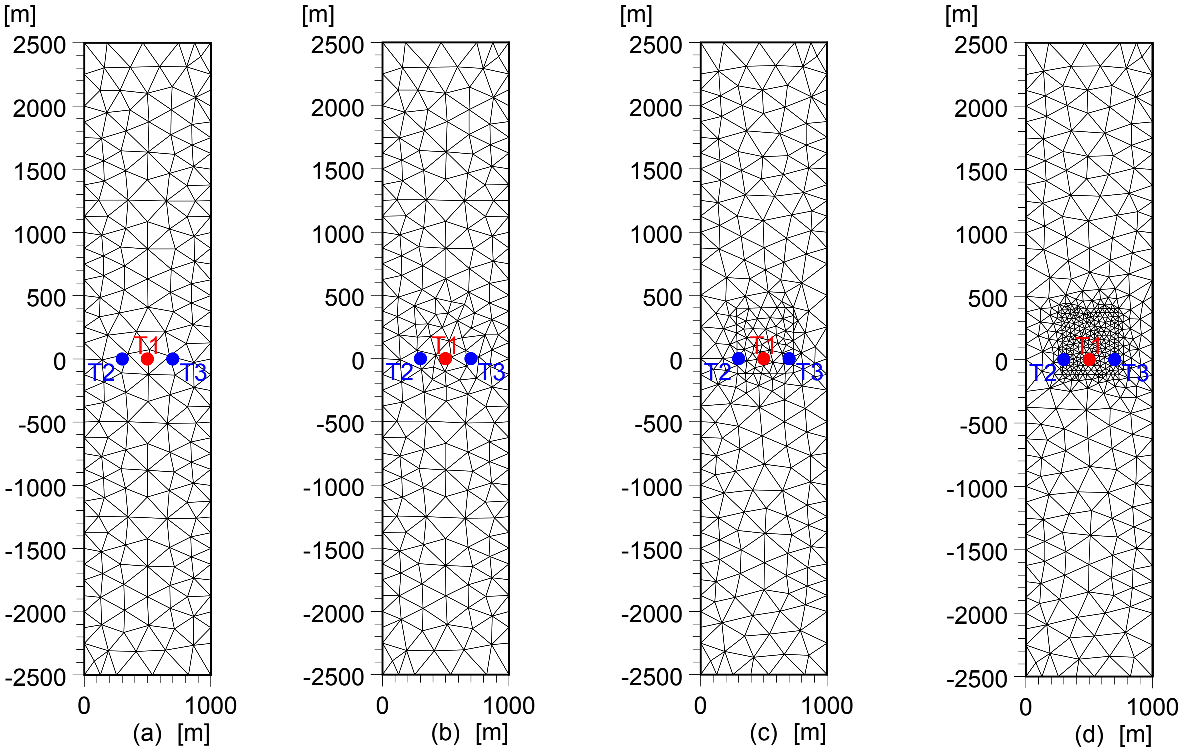


Fig. 1. The four meshes used for testing purposes. Target triangle face lengths were (a) $250\,\mathrm{m}$, (b) $150\,\mathrm{m}$, (c) $100\,\mathrm{m}$, and (d) $50\,\mathrm{m}$. Turbine position T1, described in Section III, is shown in red. Turbine positions T2 & T3, described in Section VI, are shown in blue. The direction of current flow is from the bottom to the top of the figure.

in (1) when calculating the thrust. However, the free-stream velocity is not known in the model.

High-resolution CFD simulations can use actuator disc theory to calculate the free-stream velocity from the velocity at the turbine u_t (see Section IV), but this option is not directly available to a regional-scale model as it does not have sufficient resolution to determine u_t. Instead, it only has access to the velocity u_{cell} that represents the cell as a whole, and this is the value used to calculate the thrust in (1).

When the cell is large compared to the turbine, most of the cell is unobstructed and $u_{cell} \simeq u_0$, so a reasonably accurate result will be obtained. However, as the size of the cell is reduced to approach the scale of the turbine, an increasingly large proportion of its cross-section is occupied by the rotor. This means that the reduction in speed due to the turbine has a significant effect on the cell as a whole, and so $u_{cell} < u_0$. This results in an underestimate of the force applied by the turbine to the flow, which will lead to an error in any prediction of either the energy that can be harvested by the turbine or the environmental effects of its energy extraction.

III. DEMONSTRATION OF THE ERROR

A. Method

To demonstrate this effect, four simple models were built in MIKE 3, identical except for their meshes. The scenario chosen was a channel $5\,\mathrm{km}$ in length and $1\,\mathrm{km}$ in width, with a rectangular cross-section and a constant depth of $50\,\mathrm{m}$. Bed roughness height was set to a constant value of $k_s = 0.05\,\mathrm{m}$ and vertical eddy viscosity was by a simple log law formulation. Two open boundaries were specified at the ends of the channel, and were given clamped elevations so that there was a difference in water level of $0.083\,\mathrm{m}$ (an arbitrary value) from one end of the channel to the other, resulting in a steady undisturbed flow of $2.1\,\mathrm{ms}^{-1}$.

All four meshes used ten equally spaced vertical layers. The MIKE Mesh Generator tool was used to create a computational mesh with a target triangle face length of approximately $250\,\mathrm{m}$. The same base mesh was used for all four models, but for three of them an area of approximately $500\,\mathrm{m}$ square was refined to higher resolution, as shown in Figure 1.

For each mesh the model was run for 25 hours, which was more than sufficient for a steady flow to be reached. Initially they were run without any turbines in place, in order to test for any mesh sensitivity unrelated to the turbine implementation.

A single turbine was then added at the centre of each channel, close to the upstream end of the refined mesh region (position T1 in Figure 1) and oriented to face into the direction of

TABLE I

PREDICTED FLOW SPEEDS IN THE FOUR MESHES WITHOUT AND WITH A TURBINE.

Triangle face length (m)	Speed without turbine (m/s)	Speed with turbine (m/s)	Force on turbine (kN)
250	2.109	1.897	514
150	2.109	1.876	502
100	2.109	1.815	470
50	2.109	1.756	440

flow. The diameter of the turbine was set to 20 m, and its hub elevation to $-37\,\mathrm{m}^*$. A constant thrust coefficient of 0.9 was specified. The force experienced by the turbine on the final time step was recorded and plotted against the width of the mesh element in Figure 2.

B. Result & discussion

In the test without turbines, all four meshes predicted the same flow speed at the planned turbine location to four significant figures (see Table I). Thus, it is reasonable to conclude that the model is insensitive to mesh size when there are no turbines present.

With the turbine included, it can be clearly seen in Table I and Figure 2 that the force experienced by the turbine decreases with the size of the cell, indicating that the result depends on the mesh. The loss of apparent force from the coarsest to the finest mesh trialled was approx. 14%.

This result qualitatively matches those presented by Kramer *et al.* from larger, but two-dimensional, models in both MIKE21 and Fluidity [7].

IV. CORRECTING FOR THE ERROR

A. Princples

The problem facing us is that the value used for u in (1) is u_{cell}, where it should be u_0. MIKE 3 allows for an arbitrary coefficient α in (1), which we can use to perform a correction. What is needed, then, is a way to find the appropriate value for α so that

$$\alpha = \frac{u_0^2}{u_{cell}^2} \tag{2}$$

There are a number of possible approaches to this. The most straightforward, in our simple case of one turbine in a regular channel, would be to take the velocity from some distance upstream of the turbine. However, it is not obvious what upstream point should be used if the turbine is in the second or later row of an array, or otherwise in non-uniform flow, which would limit the practical utility of this method for real-world models. Instead, the approach adopted here is to use the relationship between α and C_T that is given by actuator disc theory. More sophisticated theoretical models of turbine performance (*e.g.* [11]) could be used in a similar manner.

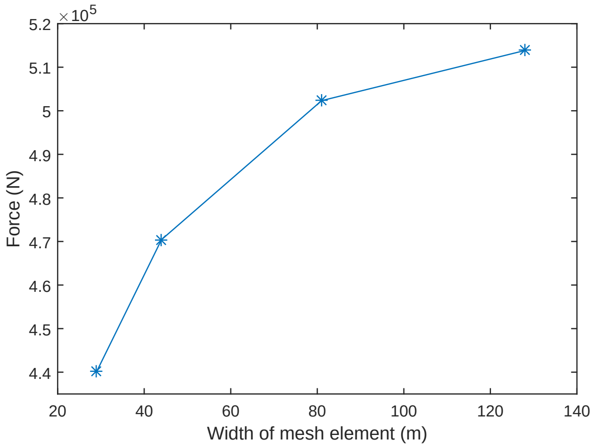


Fig. 2. Initial results, demonstrating the existence of the hypothesised mesh dependency. The definition of 'width' used here is explained in Section IV and shown in Figure 3.

It is a well-known result from actuator disc theory [12] that

$$C_T = 4a(1-a) \tag{3}$$

where a is the axial induction factor $a = 1 - \frac{u_t}{u_0}$. Solving the quadratic for a, substituting in the definition of a, and then squaring both sides gives

$$u_0^2 = \frac{4}{\left(1+\sqrt{1-C_T}\right)^2} u_t^2 \tag{4}$$

In (4), C_T can be thought of as representing the proportion of the momentum passing through the rotor that is removed. If we define an analogous coefficient ν for the whole cell, we can say that,

$$u_0^2 = \frac{4}{\left(1+\sqrt{1-\nu}\right)^2} u_{cell}^2 \tag{5}$$

where ν is equal to the turbine's thrust coefficient, scaled by the proportion of the cross-sectional area of the cell that the turbine occupies. In a three-dimensional model, where the rotor may intersect more than one vertical layer and where (as in MIKE 3) the thrust is split equally between all such layers,

$$\nu = C_T \frac{A_e/n}{\Delta x \Delta z} \tag{6}$$

where A_e is the effective area of the rotor, n is the number of vertical layers intersected, and $\Delta x \Delta z$ is the cross-sectional area of the cell. Note that in the two-dimensional case $\nu = C_T \frac{A_e}{\Delta x H}$, where H is water depth, as given in [7].

In (6), Δx represents the width of the cell or mesh element. In a triangular horizontal mesh, it is not obvious how the 'width' of a triangle sould be defined. The approach that we have adopted is to use the horizontal distance between two faces of the triangle along a horizontal line that passes through the centroid of the triangle and is perpendicular to the direction of current flow (see Figure 3).

---

[*] A 20 m diameter was chosen following feedback from developers that this a probable rotor size for the Round 1 Pentland Firth developments [10, Appendix A]. We note that the vertical position used is unlikely to be practicable for real turbines due to its proximity to the seabed. However, the aim of this work is to examine mesh size dependency and not to make realistic predictions.

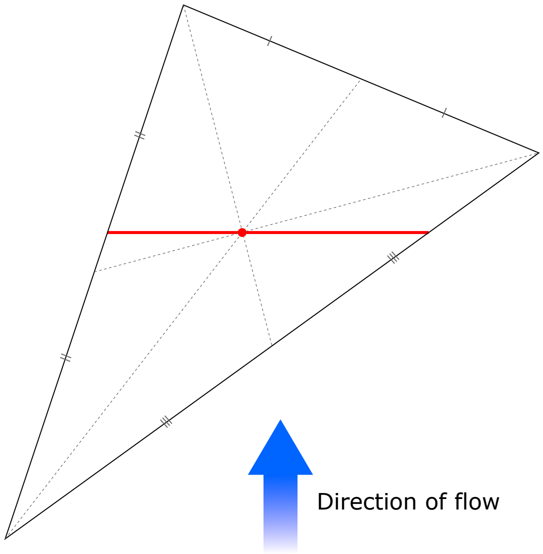


Fig. 3. Illustration of the definition of the width of a triangular mesh element used for this work. The red dot is its centroid, and the length of the heavy red line is the 'width'. Note that the red line is perpendicular to the direction of flow.

B. *Practicalities & implementation*

In the simplest of modelling arrangements, this calculation might be made once and a correction implemented. However, in realistic scenarios the speed and direction of flow in a mesh element may change on each time step, as may the water level. C_T (a function of speed) and Δx (a function of direction) may therefore change from time step to another, as may both the value of Δz and the number of layers that the rotor intersects, due to the use of sigma coordinates in the vertical. Therefore, it is necessary to calculate α once for each turbine, for each time step.

Furthermore, when a realistic turbine is being modelled, the thrust coefficient and the current speed will both depend upon each other. It may therefore be necessary to iterate new correction factors back into the model a number of times until an acceptable covergence is attained.

Clearly, in any but the simplest cases, it is impractical to carry out these corrections by hand, and an automated approach is required. Consequently a MATLAB package was developed to perform the corrections with a minimum of human input.

V. THE MATLAB PACKAGE

A. *Approach, inputs & ouputs*

The MATLAB package built to perform this correction was named 'MTMC', for 'MIKE Turbine Mesh Correction'. The primary design intent was to minimise the amount of human intervention that was required, in order to allow its use as part of a practical workflow. To this end, all of the required input data is parsed from various MIKE model files. Three outputs are produced in the filesystem:

- A new data file containing a time series of correction factors for each turbine.
- Modification of the "Turbines" section of the model definition file to instruct MIKE to apply the correction factors given in the new data file.
- A MATLAB (`.mat`) data file that is used to pass information from one iteration of this function to the next. This makes it unnecessary to repeat time-consuming operations on mesh geometry after the first time that it is run, and makes it easy for the user, by reading this file, to track convergence of force or correction factor over multiple iterations.

The flow of information and calculations is described visually in Figure 4.

B. *Versions and prerequisites*

The MTMC package was developed in MATLAB version 2014b. The following additional packages are required:

- MATLAB Mapping Toolbox.
- DHI MATLAB Toolbox (available from [13]). The 2012 edition was used for this work, but the 2014 version has also been tested. Note that this must be modified, as described below.
- Either MIKE Zero (the MIKE pre/postprocessing suite, included with MIKE 3 licenses), or the MIKE SDK (available without license from DHI). One of these must be present on the system to enable the DHI Toolbox to read MIKE binary files. The 2012 edition of MIKE Zero was used for this work.

If the user is in a country that uses '/' rather than '-' as a date separator, a modification must be made to the DHI MATLAB toolbox to reflect this. In the file `mbin\@dfsTSO\private\parseDatetimeString.m`, hyphens must be changed to slashes in lines 7 and 11 (line numbers from toolbox 2012 edition). This is because MIKE Zero respects locale settings when producing `.dfs0` files, but the Toolbox does not do the same when reading them.

C. *Usage*

The main `MTMC.MakeCorrection` function takes five arguments, all of which are filenames:

- The filename of the model definition file.
- The filename of the MATLAB data file that will be generated to pass information between iterations.
- A cell array containing the filenames of one or more turbine output files, generated from a prior model run.
- The filename of the mesh file.
- The base filename that should be used for the data file that will be created with correction factors.

It provides no returns within MATLAB.

The sequence of usage is as follows:

1) Set up the model, giving the turbines a fixed correction factor of 1 as per default. The model must use Cartesian coordinates; spherical coordinates are not currently supported. The thrust curve should be specified as the 'drag coefficient' identically for all angles using the "Tabulated drag and lift coefficient" option, and the 'lift coefficient'

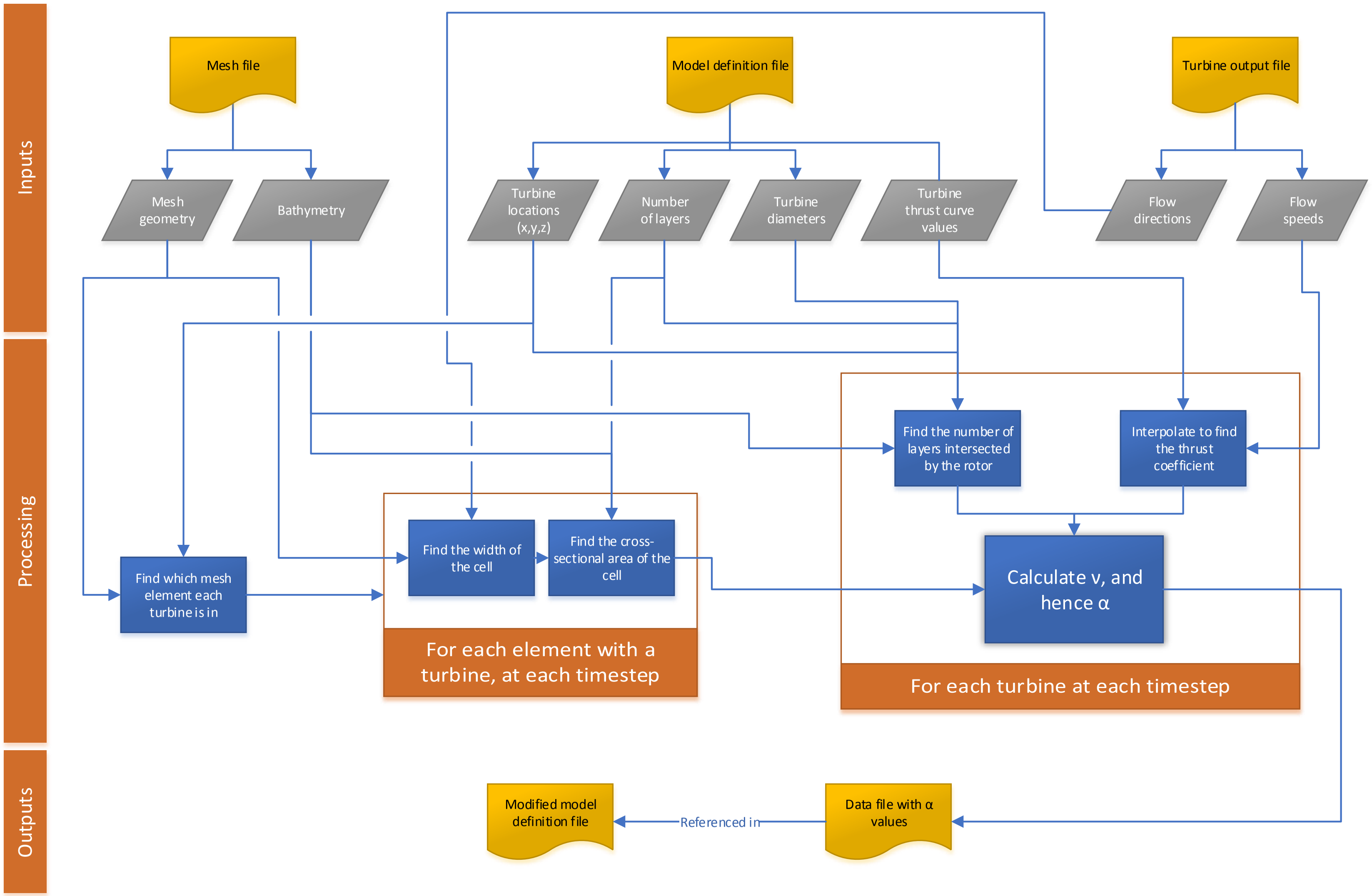


Fig. 4. Flow of information & calculations in the MATLAB package.

set to zero for all angles. This represents a turbine that is always aligned with the oncoming flow. The current version of this script does not support turbines that are not aligned with the flow. Ensure that there is a maximum of one turbine in any mesh element.

2) Run the model. This will provide current speed and direction information for the script.
3) Run `MTMC.MakeCorrection` for the first time. This will automatically modify the model definition file to assign time-varying correction factors to the turbines.
4) Run the model again. If a fixed thrust coefficient is being used, this may provide stable predictions.
5) If a variable thrust coefficient is being used, run the model and then `MTMC.MakeCorrection` as many times as necessary to obtain acceptable convergence. At any time after the first run of the function, the contents of the `.mat` file can be inspected to view the sequence of correction factors and predicted forces until that point.

D. Limitations

The current version of the package has a number of limitations that must be borne in mind:

- No account is currently taken of changing surface elevations; it is assumed that the surface elevation is always equal to mean sea level. This is a reasonable assumption when testing with steady flows, but it may cause significant inaccuracies with real tidal conditions, especially in shallow areas, because a change in surface elevation implies a change in Δz for a model cell.
- Off-axis turbines are not yet supported. This precludes the use of this script for turbines that do not 'weathervane' to face into the flow, unless they are used in a perfectly bidirectional tidal environment.
- There is no support for having multiple turbines in one mesh element. In practice, if the mesh scale is large enough for this to be an issue for well-spaced turbines, then the error that this correction addresses will be small. If the mesh is small and turbines are very tightly packed, then errors due to array effects are likely to be greater than those due to mesh size.

E. Availability

The package is still undergoing testing and improvement, but will be made publicly available at `https://github.com/TeraWatt-EcoWatt2050/MTMC` by the time of this paper's presentation. Improvements are welcomed from the community.

VI. TESTING THE CORRECTION

The proposed correction was initially applied 'by hand' to the simple scenario described in Section III. The region of each mesh around the turbine was printed at a large scale, and the geometry of the mesh element containing the turbine was measured and scaled. The appropriate correction for each version of the mesh was calculated, and was applied as a constant correction factor in a second iteration of the model.

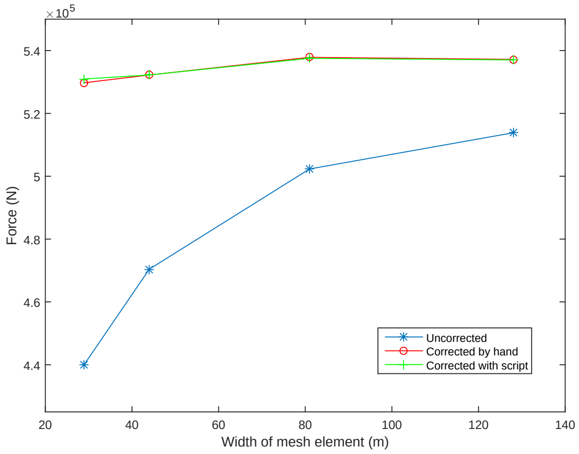


Fig. 5. Force experienced by turbine according width of mesh element. The blue (lower) line is uncorrected; the red line is corrected by hand measurement and calculation, and the green by the MATLAB script. The latter two lines are superimposed and difficult to discern.

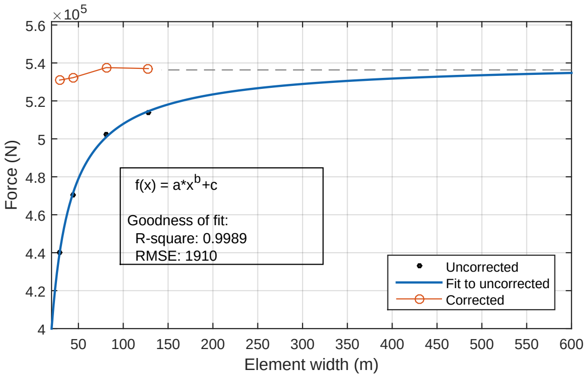
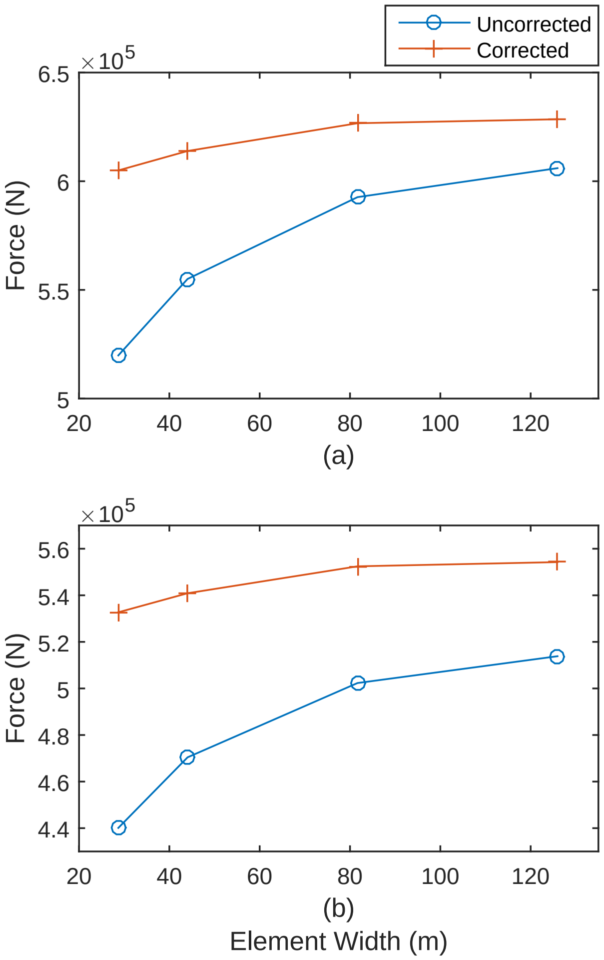


Fig. 6. The corrected and uncorrected values shown in Figure 5 are reproduced here, with a curve fitted to the uncorrected values and the x-axis extended to extrapolate to its assymptotic value. As illustrated by the dashed line, this is close to the corrected values.

Once the MATLAB script for the correction had been completed, this was run against the same simple scenario. As may be seen in Figure 5, the results were very similar, which provides confidence that the intended calculation was correctly implemented. Runtime for the script was less than half a second for this single-turbine scenario.

If a curve is fitted to the uncorrected values using a power law, then its extension to large element width approaches an assymptote (see Figure 6). This is expected, as the assymptote represents the limit where $u_{cell} = u_0$ at large mesh scales. The uncorrected force predicted at these large mesh scales is similar to the corrected predictions at small mesh scales. This is evidence not only that the intended calculation was implemented without error, but that it was the correct calculation to perform.

The result of this correction is not perfect mesh-independence, since there is still some variance in force with mesh size. However, the range in forces amongst the four



(a)

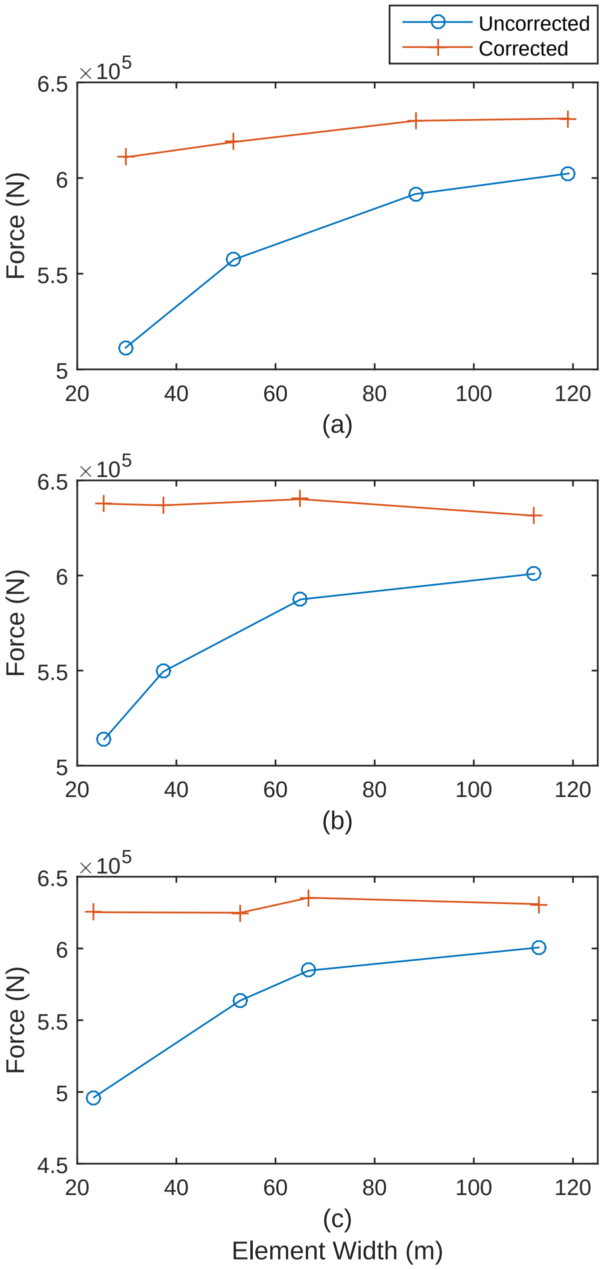
(b)

Fig. 7. Correction applied to a turbine at the same horizontal location (T1), at depths (a) 25m, (b) 37m.

meshes tested was reduced from 14% of the maximum to just 1%, which suggests an improvement.

In order to give further confidence in its correct evaluation of mesh geometry, the script was used to correct a turbine at a number of different depths and horizontal locations (positions T2 & T3 in Figure 1). To avoid any risk of interactions, which might confuse matters at this stage, only one turbine was modelled, and its location changed for each set of simulations.

For different depths in the same horizontal location (Figure 7), while the absolute force values vary with depth due to vertical shear, the change in force with mesh size is near-identical. This is as expected, because the same horizontal triangle on each mesh hosts the turbine at any depth. When the turbine is placed in different horizontal locations (Figure 8), we observe that in each case the corrected result shows a reduction in sensitivity to mesh size when compared to the uncorrected result, but that some mesh dependency clearly remains, and that this varies in magnitude between different locations on the mesh.



(a)

(b)

(c)

Fig. 8. Correction applied to a turbine at the three horizontal locations (a) T1, (b) T2, (c) T3, shown in Figure 1. In each case the turbine was at 26m depth.

VII. CONCLUSIONS

This work is a step towards being able to use high mesh resolutions in regional-scale models, such as MIKE 3, without suffering from the mesh dependency that is illustrated. It is possible that future versions of MIKE may make this correction internally, but in the meantime, the use of this script allows greater accuracy to be attained in predictions of energy resource and of environmental impact.

We have demonstrated that there is no significant mesh dependency in the simulation before turbines are added (Table I). We have shown that a mesh dependency does develop in the presence of a turbine, and that the effects of this vary in a manner consistent with theory, including approaching an asymptote for very coarse meshes (Figure 6). We have presented a correction that reduces this effect by an order of magnitude, automated this correction through MATLAB and performed initial testing of it in a simple model.

The corrections made are not perfect, in that some sensitivity to the mesh still remains. This may be due to limitations in the theory used (*e.g.* the one-dimensional nature of actuator disc theory, or the assumption that all drag due to a turbine happens at the rotor, thus ignoring wake mixing losses), or it may be due to limitations in the current implementation of the correction (*e.g.* in the definition of the width of a triangle).

Further work will focus on testing this script further, improving the correction, and then applying it to more realistic tidal model scenarios. Comparison with measurements and/or high-resolution CFD simulation will provide validation of the results obtained.

ACKNOWLEDGMENTS

This work has been conducted under the auspices of TeraWatt, an EPSRC-funded research project. We thank David Woolf for his comments on a draft of this paper.

REFERENCES

[1] A. Cornett, M. Toupin, S. Baker, S. Piche, and I. Nistor, "Appraisal of IEC standards for wave and tidal energy resource assessment," in *International Conference on Ocean Energy (ICOE) 2014*, Halifax, Canada, Nov. 2014. [Online]. Available: http://www.icoe2014canada.org/wp-content/uploads/2014/11/2-CornettAndrewARTICLE_9-3.pdf

[2] EMEC and C. Legrand, "Assessment of tidal energy resource," 2009. [Online]. Available: http://www.emec.org.uk/assessment-of-tidal-energy-resource/

[3] M. A. Shields, D. K. Woolf, E. P. Grist, S. A. Kerr, A. Jackson, R. E. Harris, M. C. Bell, R. Beharie, A. Want, E. Osalusi, S. W. Gibb, and J. Side, "Marine renewable energy: The ecological implications of altering the hydrodynamics of the marine environment," *Ocean & Coastal Management*, vol. 54, no. 1, pp. 2–9, Jan. 2011.

[4] V. Ramos, R. Carballo, M. lvarez, M. Snchez, and G. Iglesias, "Assessment of the impacts of tidal stream energy through high-resolution numerical modeling," *Energy*, Sep. 2013.

[5] R. Martin-Short, J. Hill, S. Kramer, A. Avdis, P. Allison, and M. Piggott, "Tidal resource extraction in the Pentland Firth, UK: Potential impacts on flow regime and sediment transport in the inner sound of stroma," *Renewable Energy*, vol. 76, pp. 596–607, Apr. 2015.

[6] J. Culina and R. Karsten, "Comparison of different resolution models and observed current profiles in the Bay of Fundy, Canada using turbine-relevant metrics," in *Proceedings of the 9th European Wave and Tidal Energy Conference.*, Southampton, 2011.

[7] S. Kramer, M. Piggott, J. Hill, L. Kregting, D. Pritchard, and B. Elsaesser, "The modelling of tidal turbine farms using multi-scale, unstructured mesh models," in *Proceedings of the 2nd International Conference on Environmental Interactions of Marine Renewable Energy Technologies*, Stornoway, 2014.

[8] DHI, "MIKE 21 & MIKE 3 flow model FM hydrodynamic & transport module scientific documentation," 2012.

[9] DHI, "MIKE 3 Flow Model FM Hydrodynamic Module user guide," 2012.

[10] S. Baston, S. Waldman, and J. Side, "Modelling energy extraction in tidal flows," TeraWatt position paper, May 2015. [Online]. Available: http://www.masts.ac.uk/about/masts-publications/terawatt-publications/

[11] J. Whelan, J. Graham, and J. Peir, "A free-surface and blockage correction for tidal turbines," *Journal of Fluid Mechanics*, vol. 624, pp. 281–291, 2009, 281.

[12] J. F. Manwell, *Wind energy explained: theory, design and application*, 2nd ed. Chichester, U.K: Wiley, 2009.

[13] DHI, "DHI Matlab toolbox." [Online]. Available: http://www.mikepoweredbydhi.com/download/mike-by-dhi-tools/coastandseatools/dhi-matlab-toolbox