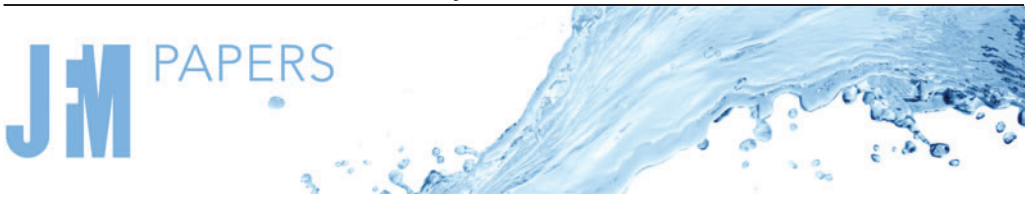
J. Fluid Mech. (2022), *vol*. 941, A19, doi:10.1017/jfm.2022.238



Planar potential flow on Cartesian grids

Diederik Beckers[1] **and Jeff D. Eldredge**[1],†

[1]Mechanical and Aerospace Engineering, University of California, Los Angeles, CA 90095-1597, USA

(Received 19 February 2021; revised 17 January 2022; accepted 11 March 2022)

Potential flow has many applications, including the modelling of unsteady flows in aerodynamics. For these models to work efficiently, it is best to avoid Biot–Savart interactions. This work presents a grid-based treatment of potential flows in two dimensions and its use in a vortex model for simulating unsteady aerodynamic flows. For flows consisting of vortex elements, the treatment follows the vortex-in-cell approach and solves the streamfunction–vorticity Poisson equation on a Cartesian grid after transferring the circulation from the vortices onto the grid. For sources and sinks, an analogous approach can be followed using the scalar potential. The combined velocity field due to vortices, sinks and sources can then be obtained using the Helmholtz decomposition. In this work, we use several key tools that ensure the approach works on arbitrary geometries, with and without sharp edges. First, the immersed boundary projection method is used to account for bodies in the flow and the resulting body-forcing Lagrange multiplier is identified as the bound vortex sheet strength. Second, sharp edges are treated by decomposing the vortex sheet strength into a singular and non-singular part. To enforce the Kutta condition, the non-singular part can then be constrained to remove the singularity introduced by the sharp edge. These constraints and the Poisson equation are formulated as a saddle-point system and solved using the Schur complement method. The lattice Green's function is used to efficiently solve the discrete Poisson equation with unbounded boundary conditions. The method and its accuracy are demonstrated for several problems.

Key words: computational methods, vortex dynamics, separated flows

1. Introduction

Potential flow plays an important role in aerodynamic modelling, but also appears in other areas such as the modelling of water waves or wind farms and the calculation of added mass. Besides its prominent use for steady flow around airfoils at high Reynolds numbers, potential flow theory has long provided the tools for vortex methods to simulate unsteady flows around airfoils and bluff bodies. These vortex methods discretize the vorticity in the flow with singular elements such as point vortices, vortex sheets or a combination of both.

† Email address for correspondence: jdeldre@ucla.edu

© The Author(s), 2022. Published by Cambridge University Press

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

In the case of an inviscid and incompressible model, the irrotational flow outside of these singular vortex elements is a potential flow. Singular vortex elements that represent the free vorticity in an inviscid vortex model are advected by the local flow velocity according to Helmholtz's second theorem and can be tracked as Lagrangian points. In case the vortex method inserts new vortex elements in the flow behind a bluff body or at sharp edges, Kelvin's circulation theorem dictates that the circulation should be conserved. Singular potential flow elements can also serve to enforce the no-penetration condition, with the most common choice in vortex methods being a distribution of singular vorticity on the body, denoted as the bound vortex sheet. Besides their choice for the type of singular elements, potential flow solvers for vortex methods differ in their way of calculating the flow velocity. This can be done by either using direct interaction between the potential flow elements or by calculating the velocity on a grid over the entire domain and the eventual choice dictates the treatment of boundary and edge conditions.

In the first approach, the Green's function of the Laplacian is applied to the Poisson equation in the velocity–vorticity formulation to give the Biot–Savart integral, which provides the exact solution for a velocity field that satisfies unbounded boundary conditions. Biot–Savart vortex methods often smooth the Biot–Savart kernel, equivalent to replacing point vortices by vortex blobs (Chorin & Bernard 1973), to suppress Kelvin–Helmholtz instabilities below a certain wavelength resulting from the interactions between closely spaced vortex elements. The solution generally requires $O(N^2)$ operations, with N the number of vortex elements, to sum the influences of each discretized vortex element on every other element. With fast multipole methods, it scales optimally as $O(N)$ but with a large prefactor and overhead cost. Inviscid vortex methods of this kind can straightforwardly use the potential flow tools to enforce the no penetration, such as conformal mapping or solving the integral equation for a surface singularity distribution through analytical inversion, panel discretization or with Fourier expansions. For a detailed review of this subject, the reader is referred to Cottet & Koumoutsakos (2000) and Eldredge (2019).

A second approach to calculate the flow velocity follows from the discretization of the Poisson equation in the velocity–vorticity formulation or streamfunction–vorticity formulation on an Eulerian grid over the domain of interest and is called a vortex-in-cell (VIC) approach, first developed by Christiansen (1973). The procedure requires us first to transfer the circulation from Lagrangian vortex elements onto the grid, then to solve the discrete Poisson equation and finally to interpolate the velocity (or the curl of the streamfunction) back to the vortex elements. It introduces discretization errors but only requires $O(M \log M)$ operations, with M the number of grid points, to solve the Poisson equation with current numerical techniques and $O(N)$ operations to perform the regularization and interpolation. Similar to the regularized Biot–Savart kernel, the grid spacing together with the vorticity regularization scheme determine the cutoff wavelength below which the Kelvin–Helmholtz instabilities get suppressed.

The early work on VIC methods focused on inviscid vortex dynamics using Fourier-based Poisson solvers with Dirichlet or periodic boundary conditions (Meng & Thomson 1978; Baker 1979; Couët, Buneman & Leonard 1981) and the analysis of different interpolation kernels (Ebiana & Bartholomew 1996). After viscous schemes for vortex methods were introduced, VIC methods increasingly replaced Biot–Savart methods in an effort to speed up vortex methods for viscous flows, leading to methods with over a billion vortex particles (Chatelain *et al.* 2008). This also stimulated the development of VIC methods for external flows over bodies, mostly for viscous flows. The most straightforward way to include a body in the flow is to use a body-fitted mesh as in

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

Cottet & Poncet (2004), who apply the Helmholtz decomposition on the flow in their VIC method and place a Neumann boundary condition for the scalar potential on the body to account for its presence and employ an analytic boundary condition for the far field. However, a body-fitted mesh is case specific and, therefore, the same work (and later also Poncet 2009), develops an immersed boundary method by introducing a singular distribution of sources that represents the influence of the body and is smeared onto a Cartesian grid using a discrete approximation to the Dirac delta function. The result is a source term that is inserted in the Poisson equation for the scalar potential which is solved on the grid. Similar to the immersed boundary methods, Brinkman penalization methods do not require body-fitted meshes. In vortex methods, the Brinkman penalization method (Coquerelle & Cottet 2008; Rossinelli *et al.* 2010; Gazzola *et al.* 2011; Rasmussen, Cottet & Walther 2011; Chatelin & Poncet 2014) adds a volume forcing term to the vorticity transport equation that includes a penalization parameter, equivalent to the porosity of the body. However, the method suffers from a strong time step restriction, which motivated Hejlesen *et al.* (2015) to use an iterative Brinkman penalization method, which Spietz, Hejlesen & Walther (2017) extended to three dimensions. Gillis, Winckelmans & Chatelain (2017) formulate this method as a linear system and uses a recycling iterative solver to obtain the solution more efficiently.

The immersed boundary method and Brinkman penalization method both smear out the influence of the interface onto nearby grid points. LeVeque & Li (1994) developed the immersed interface method (IIM) to overcome this issue and to obtain a higher spatial order of accuracy than the immersed boundary method. The premise of this method is to discretize the jump conditions caused by the interface with finite differences instead of discretizing the Dirac delta function, and the result is a sharp representation of the interface with a second or higher-order accuracy. Marichal, Chatelain & Winckelmans (2014) apply the explicit-jump IIM (Wiegmann & Bube 2000) in his potential flow method. The influence of the interface was condensed into an extra source term in the streamfunction Poisson equation and it was recognized that the term is equivalent to a bound vortex sheet strength regularized to the grid. The bound vortex sheet strength, streamfunction field and outer boundary condition on the streamfunction are then computed iteratively. The work presents results from the flow over a cylinder and an airfoil, for which the Kutta condition enforced through discretization of the streamfunction normal derivatives at the trailing edge. Gillis, Winckelmans & Chatelain (2018) extends this method, but applies the IIM on the scalar potential instead. An explicit formula for the singular distribution of sources on the interface is then obtained by applying the Sherman–Morrison–Woodbury decomposition formula to the Poisson equation, similar to Poncet (2009). Furthermore, by solving the Poisson equation using the lattice Green's function (Katsura & Inawashiro 1971), which automatically satisfies far-field boundary conditions, the method is no longer iterative and the cost is greatly reduced. Gillis *et al.* (2019) apply this method again to the streamfunction in two dimensions and employs it in a viscous VIC method.

In this work we present a grid-based treatment for planar potential flow. The focus of this work is on the flow around point vortices, but the treatment can easily be extended to account for sources and sinks. In the case of the flow around vortices, the treatment follows a VIC approach. In our implementation, the streamfunction–vorticity Poisson equation is solved for the streamfunction on the grid using the lattice Green's function, such that unbounded boundary conditions are accounted for. To enforce the no-penetration condition on surfaces in the flow, the treatment is presented by using the immersed boundary projection method and our implementation is therefore approximately first-order accurate in space (Colonius & Taira 2008). This approach consists of adding an extra

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

singular vorticity source term to the streamfunction–vorticity Poisson equation that is distributed over the discrete surface points and is smeared onto the nearby grid nodes. This extra vorticity term represents the bound vortex sheet strength and assumes the role of a Lagrange multiplier in this method. The modified Poisson equation combined with the no-penetration constraint then forms a saddle-point system that can be solved with Schur's complement method. Note that the IIM introduces a similar modification to the Poisson equation (Marichal *et al.* 2014) and the Sherman–Morrison–Woodbury decomposition can produce an expression for the discrete vortex sheet strength (Gillis *et al.* 2019) that is equivalent to the formula we obtain by using the Schur complement method. Consequently, one could use the IIM to obtain a second-order method instead. Drawing inspiration from the analytical treatment of the Kutta condition in Biot–Savart methods, this work then introduces a new way of enforcing the Kutta condition in a discrete potential flow treatment by decomposing the discrete vortex sheet strength into a singular and non-singular part and constraining the non-singular part. This amounts to algebraically constraining the system arising from the immersed boundary projection method to make it well behaved.

This paper is structured as follows. We first focus on discretizing the unbounded potential flow problem with point vortices in § 2.1 and then discuss the no-penetration condition in § 2.2. We describe the role of circulation in § 2.3 and introduce the treatment for enforcing the Kutta condition in steady and unsteady flows in § 2.4. We extend the treatment to generalized edge conditions in § 2.5 and introduce methods for computing pressure, impulse and added mass in § 2.6. The extension to multiple bodies is discussed in § 2.7.

2. Methodology

2.1. *The basic two-dimensional potential flow problem*

A two-dimensional, differentiable velocity field $\boldsymbol{u}$ on the unbounded domain $\Omega = \{\boldsymbol{x} = (x, y)\}$ can be decomposed according to the Helmholtz decomposition

$$\boldsymbol{u} = \boldsymbol{\nabla}\phi + \boldsymbol{\nabla} \times \psi \boldsymbol{e}_z, \tag{2.1}$$

where ϕ is the scalar potential, ψ is the streamfunction and $\boldsymbol{e}_z$ is the unit vector out of the plane. In a potential flow without sources or sinks, this problem can be solved by solving exclusively the governing equation for the streamfunction

$$\nabla^2 \psi = -\omega, \tag{2.2}$$

where ω is the vorticity field, consisting solely of N_v singular point vortices

$$\omega = \sum_{q=1}^{N_v} \Gamma_{v,q} \boldsymbol{\delta}(\boldsymbol{x} - \boldsymbol{X}_q), \tag{2.3}$$

where $\boldsymbol{\delta}$ is the two-dimensional Dirac delta function, and $\boldsymbol{X}_q = (X_q, Y_q)$ and $\Gamma_{v,q}$ are the position and strength of the qth point vortex.

Because (2.2) does not account for sources and sinks, one would have to solve the auxiliary Poisson problem for the scalar potential

$$\nabla^2 \phi = \Theta, \tag{2.4}$$

where Θ is the rate of dilatation, consisting of singular sources and sinks, similar to (2.3). The overall velocity field due to vortices, sinks and sources can then be obtained using the Helmholtz decomposition (2.1).

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

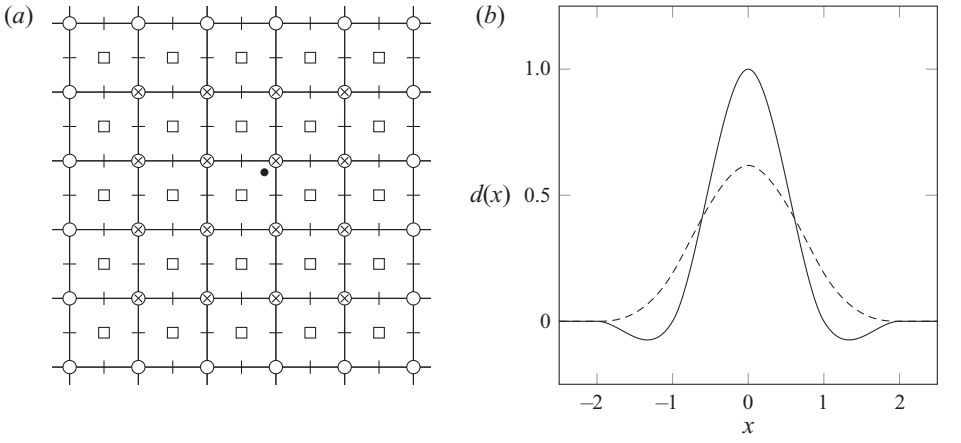


Figure 1. (*a*) Schematic of the grid and the regularization of data from a Lagrangian point (•) onto the grid with a discrete Dirac delta function of radius two. Symbols (○), (□), (|) and (-) denote the locations holding the components of the nodes $\mathcal{N}$, cell centres $\mathcal{C}$, horizontal faces $\mathcal{F}_x$ and vertical faces $\mathcal{F}_y$, respectively. Symbols (×) denote the nodes that are affected by the regularization. (*b*) Two examples of discrete Dirac delta functions of radius two: the M_4' function (——) from Monaghan (1985) and the smoothed three-point function (- - - -) from Yang *et al.* (2009).

We now introduce a discrete treatment of this basic potential flow problem and focus on the discrete streamfunction. We consider here a staggered, Cartesian grid with uniform cell size Δx and of infinite extent. The space corresponding to data at cell vertices (nodes) on this grid is denoted by $\mathcal{N}$, and the physical coordinates of these nodes by $\boldsymbol{x}_{\mathcal{N}}$ and $\boldsymbol{y}_{\mathcal{N}}$. Furthermore, we consider a finite number N_v of Lagrangian, singular point vortices. The space of scalar data on these points is denoted by $\mathcal{W}^{N_v}$ and we define $\Gamma_v \in \mathcal{W}^{N_v}$ as the vector containing the strengths of the point vortices in our grid-based treatment. The basic (unbounded) potential flow problem is expressed as

$$\boldsymbol{L}\boldsymbol{s} = -\boldsymbol{w}, \tag{2.5}$$

where $\boldsymbol{L}$ is the discrete 5-point Laplacian operator, $\boldsymbol{s} \in \mathcal{N}$ is the discrete streamfunction and $\boldsymbol{w} \in \mathcal{N}$ the discrete vorticity. The discrete velocity field $\boldsymbol{v}$, whose components lie on the faces of the cells with the corresponding normals, is computed from $\boldsymbol{s}$ by the discrete curl operation,

$$\boldsymbol{v} = \boldsymbol{C}\boldsymbol{s}. \tag{2.6}$$

The operator $\boldsymbol{C}$ applies centred differences between the nodes to obtain the velocity components at the intermediate centres of cell faces. We denote the space of data that lie on cell faces by $\mathcal{F}$, so $\boldsymbol{C} : \mathcal{N} \mapsto \mathcal{F}$. Figure 1(*a*) shows the staggered grid structure.

The grid differencing operators $\boldsymbol{L}$ and $\boldsymbol{C}$ and others to be defined are scaled by the grid spacing, and thus represent second-order approximations of the corresponding continuous operators, and the discrete streamfunction $\boldsymbol{s}$, vorticity $\boldsymbol{w}$ and velocity $\boldsymbol{v}$ are each approximations of their continuous counterparts. The total flow circulation is equal to the sum of grid vorticity multiplied by the cell area Δx^2. To support the work that follows, we define an inner product on the grid nodes

$$\langle \boldsymbol{w}_1, \boldsymbol{w}_2 \rangle_{\mathcal{N}} = \Delta x^2 \boldsymbol{w}_1^T \boldsymbol{w}_2, \tag{2.7}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

for $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathcal{N}$. The total circulation can then be written compactly as

$$\Gamma_w = \langle \boldsymbol{1}_{\mathcal{N}}, \boldsymbol{w} \rangle_{\mathcal{N}}, \tag{2.8}$$

where $\boldsymbol{1}_{\mathcal{N}} \in \mathcal{N}$ is a grid vector of ones.

The particular solution of (2.5) can be written down immediately with the help of the lattice Green's function for $\boldsymbol{L}$ (Katsura & Inawashiro 1971; Cserti 2000; Liska & Colonius 2014). We denote this simply by the inverse operator

$$\boldsymbol{s} = -\boldsymbol{L}^{-1}\boldsymbol{w}. \tag{2.9}$$

It can be shown that, with a suitable truncation of the grid, both $\boldsymbol{L}$ and its inverse are symmetric operators. However, $\boldsymbol{L}$ is only positive semi-definite, and an additional homogeneous solution, $\boldsymbol{s}_\infty \in \mathcal{N}$ – for example, corresponding to a uniform flow – can be added to the particular solution of this equation $-\boldsymbol{L}^{-1}\boldsymbol{w} + \boldsymbol{s}_\infty$. This homogeneous solution allows us to satisfy boundary conditions at (discrete) infinity. The size of the domain in our simulations is therefore not relevant, as long as it includes the features that are of interest.

As in the VIC approach (Christiansen 1973), the discrete vorticity is obtained by immersing the vortex elements into the grid and transferring their circulation to the nearby nodes using a tensor product of two one-dimensional, discrete Dirac delta functions

$$\boldsymbol{w} = \frac{1}{\Delta x^2} \sum_{q=1}^{N_v} \Gamma_{v,q} d\left(\frac{\boldsymbol{x}_{\mathcal{N}} - X_q}{\Delta x}\right) d\left(\frac{\boldsymbol{y}_{\mathcal{N}} - Y_q}{\Delta x}\right), \tag{2.10}$$

where d is a discrete Dirac delta function. (In particular, $d(x/\Delta x)/\Delta x$ represents a Dirac sequence as $\Delta x \to 0$.) By the properties imposed on d

$$\langle \boldsymbol{1}_{\mathcal{N}}, \boldsymbol{w} \rangle_{\mathcal{N}} = \sum_{q=1}^{N_v} \Gamma_{v,q}. \tag{2.11}$$

In this work, d is the M_4' function from Monaghan (1985), depicted in figure 1(*b*); panel (*a*) depicts how data from a Lagrangian point are regularized onto the grid nodes. For compact notation, we define $\boldsymbol{d}_q \in \mathcal{N}$, the grid vorticity field generated when a vortex of unit strength is immersed into the grid, so that we can write (2.10) as

$$\boldsymbol{w} = \sum_{q=1}^{N_v} \Gamma_{v,q} \boldsymbol{d}_q. \tag{2.12}$$

Even more compactly, we define the regularization operator ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{W}} : \mathcal{W}^{N_v} \mapsto \mathcal{N}$, whose N_v columns are $\boldsymbol{d}_q$, $q = 1, \ldots, N_v$, which allows us to write (2.12) as

$$\boldsymbol{w} = {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{W}} \Gamma_v, \tag{2.13}$$

the discrete streamfunction field as $\boldsymbol{s} = -\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{W}}\Gamma_v + \boldsymbol{s}_\infty$ and the velocity field as

$$\boldsymbol{v} = -\boldsymbol{C}\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{W}}\Gamma_v + \boldsymbol{C}\boldsymbol{s}_\infty. \tag{2.14}$$

Figure 2 shows a spatial grid refinement analysis for a flow consisting of point vortices of random strength that are randomly positioned in the lower-left quadrant of the domain. The analysis verifies that the discretization technique of the Poisson equation is second-order accurate in Δx. We compute the error as $\epsilon_s = \|\boldsymbol{s} - \psi(\boldsymbol{x}_{\mathcal{N}}, \boldsymbol{y}_{\mathcal{N}})\|_2 / \|\psi(\boldsymbol{x}_{\mathcal{N}}, \boldsymbol{y}_{\mathcal{N}})\|_2$, where ψ is the exact solution for the streamfunction. We only consider the values in the

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

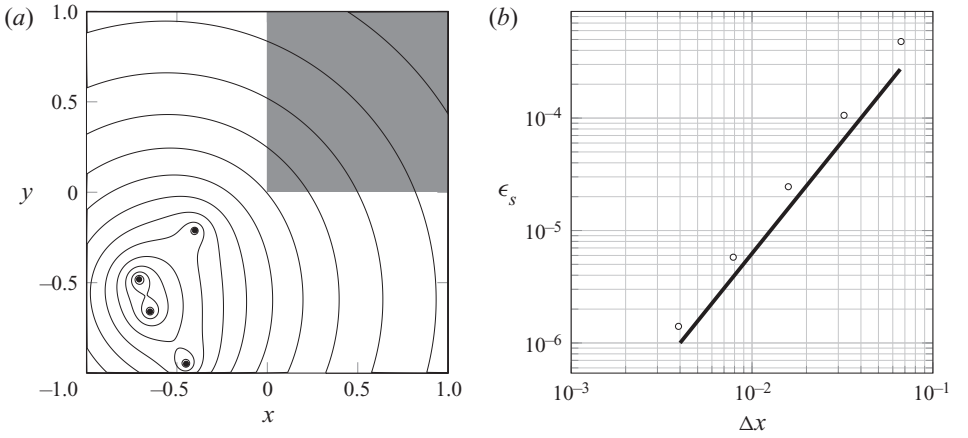


Figure 2. (*a*) Contours (——) of the discrete streamfunction for randomly positioned point vortices (•) of random strengths, and (*b*) its error (○) over the shaded area for different grid spacings. Overlaid is an error (——) that scales as Δx^2.

upper-right quadrant of the domain to exclude the positions of the point vortices, because the exact singularities at these positions are not comparable to the regularized, discrete version.

To obtain the velocity at the locations of the point vortices, the discrete velocity field $\boldsymbol{v}$ should first be interpolated from cell faces to the nodes (using simple averages). Then, the velocity can be interpolated onto the point vortices with the interpolation operator ${}^{\mathcal{W}}\boldsymbol{E}_{\mathcal{N}} : \mathcal{N} \mapsto \mathcal{W}^{N_v}$, which is the transpose of the regularization operator, ${}^{\mathcal{W}}\boldsymbol{E}_{\mathcal{N}} = {}^{\mathcal{N}}\boldsymbol{R}^T_{\mathcal{W}}$, to obtain an overall interpolation scheme that is consistent with (2.13).

It is worth noting here that, if we wish to include sources and sinks, then we can make additional use of a discrete scalar potential, lying at cell centres, $\boldsymbol{\phi} \in \mathcal{C}$, whose governing equation is the discretized version of (2.4), analogous to (2.5). The sources and sinks can be immersed into a rate of dilatation field at cell centres in a similar manner to vortices to cell nodes (2.10). The velocity field due to the combination of vortices, sinks and sources can then be composed using the discrete version of the Helmholtz decomposition (2.1)

$$\boldsymbol{v} = \boldsymbol{G}\boldsymbol{\phi} + \boldsymbol{C}\boldsymbol{s}. \tag{2.15}$$

Analogous to the discrete curl operator, the discrete gradient operator $\boldsymbol{G}$ applies centred differences to scalar data at cell centres $\mathcal{C}$ to obtain the velocity components at the cell faces, so $\boldsymbol{G} : \mathcal{C} \mapsto \mathcal{F}$.

2.2. *Potential flow with an impenetrable surface*

Now, let us suppose we have a rigid impenetrable surface $\mathcal{S}$, on which we seek to enforce the no-penetration condition for the streamfunction. The no-penetration condition asserts that the normal components of the fluid velocity and this surface velocity must be equal. For rigid bodies, this surface motion can be alternatively described by a streamfunction, and the no-penetration condition in the absence of sources and sinks can be imposed equivalently (in two dimensions) by setting the fluid streamfunction equal to that of the surface ψ_b up to a uniform value. In continuous form, this is described by the Dirichlet problem

$$\nabla^2\psi = -\omega \tag{2.16}$$

$$\psi\,(\boldsymbol{x}) = \psi_b\,(\boldsymbol{x})\,, \quad \boldsymbol{x} \in \mathcal{S}. \tag{2.17}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

In the presence of sources and sinks, one can again solve the auxiliary problem for the scalar potential and enforce the no-penetration condition only for the flow due to the sources and sinks. The overall velocity field (2.1) then satisfies the no-penetration condition. Note that the boundary value problem for the scalar potential is a Neumann problem and requires a slightly different numerical treatment (Poncet 2009; Gillis *et al.* 2018).

The continuous Dirichlet problem (2.16)–(2.17) can be solved via Green's theorem with boundary integrals. We will now solve the discrete version of this equation using the immersed boundary projection method, resulting in completely analogous operations.

2.2.1. *Discrete surface and its immersion in the grid*

We enforce the no-penetration condition at a finite number N_s of discrete surface forcing points; the space of scalar data on these Lagrangian points is denoted by $\mathcal{S}^{N_s}$. In particular, let us define $\boldsymbol{r}_x, \boldsymbol{r}_y \in \mathcal{S}^{N_s}$ as the vectors of x and y coordinates of the surface points. Each surface point p is associated with a small straight segment of length ΔS_p. Some of the calculations will require information about the local surface orientation. For this purpose, we define vectors $\boldsymbol{n}_x, \boldsymbol{n}_y \in \mathcal{S}^{N_s}$ of components of the discrete surface unit normals. We also define the space $\mathcal{V}^{N_s}$ to hold vector-valued data, such as velocity, on the immersed surface points. For convenience, let us also define unit vectors $\boldsymbol{e}_p$ on this space, equal to 1 at surface point p ($1 \leqslant p \leqslant N_s$) and zero at every other point. For example, the x coordinate of point p is picked out of the vector $\boldsymbol{r}_x$ by projection onto the pth unit vector

$$\boldsymbol{e}_p^T \boldsymbol{r}_x. \tag{2.18}$$

Each of the surface point spaces has an associated inner product that includes the surface length, e.g.

$$\langle \boldsymbol{g}_1, \boldsymbol{g}_2 \rangle_{\mathcal{S}^{N_s}} = \sum_p \Delta S_p g_{1,p} g_{2,p}, \tag{2.19}$$

for any $\boldsymbol{g}_1, \boldsymbol{g}_2 \in \mathcal{S}^{N_s}$, so that the inner product approximates a surface integral. Another vector we will make substantial use of in this paper is $\boldsymbol{1}_{\mathcal{S}} \in \mathcal{S}^{N_s}$, a vector of ones on all surface points.

From any vector $\boldsymbol{g} \in \mathcal{S}^{N_s}$, we can also form a diagonal $N_s \times N_s$ operator $\boldsymbol{D}_{\boldsymbol{g}}$ with the entries of the vector along the diagonal. When this operator acts upon another vector $\boldsymbol{h} \in \mathcal{S}^{N_s}$, it represents the Hadamard (i.e. element-by-element) product of the two vectors, $\boldsymbol{D}_{\boldsymbol{g}}\boldsymbol{h} = \boldsymbol{g} \circ \boldsymbol{h} \in \mathcal{S}^{N_s}$. Note that $\boldsymbol{D}_{\boldsymbol{g}}\boldsymbol{h} = \boldsymbol{D}_{\boldsymbol{h}}\boldsymbol{g}$, and that $\boldsymbol{D}_{\boldsymbol{g}}\boldsymbol{1}_{\mathcal{S}} = \boldsymbol{g}$.

Similar to (2.13), surface data are immersed into the grid with the regularization operator ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}} : \mathcal{S}^{N_s} \mapsto \mathcal{N}$. Grid data are interpolated onto the surface points with the interpolation operator ${}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} : \mathcal{N} \mapsto \mathcal{S}^{N_s}$; ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}$ can be constructed (and we will assume it has) so that it is the transpose of the interpolation operator, ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}} = {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}{}^T$, with respect to the grid and surface inner products defined in this paper. Furthermore, note that ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}$ and ${}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}$ can be constructed with a different choice for the discrete Dirac delta function than the one used for the vortex regularization. In this work, we use the smoothed three-point function from Yang *et al.* (2009) (figure 1) and we use a uniform spacing between the surface points.

2.2.2. *The immersed surface potential flow problem*

The surface's motion is specified by a velocity distribution $\boldsymbol{v}_b$, represented discretely by components $\boldsymbol{v}_{b,x}, \boldsymbol{v}_{b,y} \in \mathcal{S}^{N_s}$. For rigid bodies, this surface motion can be described by

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

a streamfunction. Specifically, translation at velocity (U, V) and rotation at angular velocity Ω would be described equivalently by velocity components

$$\boldsymbol{v}_{b,x} = U\boldsymbol{1}_{\mathcal{S}} - \Omega \boldsymbol{r}_y, \quad \boldsymbol{v}_{b,y} = V\boldsymbol{1}_{\mathcal{S}} + \Omega \boldsymbol{r}_x \tag{2.20a,b}$$

or by a surface streamfunction $\boldsymbol{s}_b \in \mathcal{S}^{N_s}$

$$\boldsymbol{s}_b = U\boldsymbol{r}_y - V\boldsymbol{r}_x - \tfrac{1}{2}\Omega(\boldsymbol{D}_{\boldsymbol{r}_x}\boldsymbol{r}_x + \boldsymbol{D}_{\boldsymbol{r}_y}\boldsymbol{r}_y). \tag{2.21}$$

The no-penetration condition can be imposed by setting the discrete streamfunction equal to that of the surface, up to a uniform value, $\boldsymbol{s}_0 \in \mathcal{S}^{N_s}$

$${}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{s} = \boldsymbol{s}_b - {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{s}_\infty - \boldsymbol{s}_0. \tag{2.22}$$

For later shorthand, we will denote the difference between the body motion streamfunction and interpolated uniform flow streamfunction by $\boldsymbol{s}'_b \equiv \boldsymbol{s}_b - {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{s}_\infty$. This modified streamfunction simply consists of subtracting the components (U_∞, V_∞) of the uniform flow from (U, V) in (2.21). The uniform value $\boldsymbol{s}_0$ is left unspecified and will later serve the role of enforcing a constraint on circulation. For now, we will suppose that it can be set arbitrarily.

The no-penetration constraint is enforced in the basic potential flow problem (2.5) with the help of a vector of Lagrange multipliers, $\boldsymbol{f} \in \mathcal{S}^{N_s}$, on the surface points. The modified potential flow problem is thus

$$\boldsymbol{L}\boldsymbol{s} = -(\boldsymbol{w} + {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f}). \tag{2.23}$$

In fact, by simple comparison with the vorticity $\boldsymbol{w}$, it is clear that the vector $\boldsymbol{f}$ represents the strength of the discrete bound vortex sheet on the surface and serves as another source term of the Poisson equation. Suppose we consider the bound vortex sheet $\gamma(u)$ that emerges from the analogous continuous problem on the undiscretized surface, where u is the arc-length parameter along the surface. At each point p, the discrete solution $\boldsymbol{f}$ is approximately equal to this continuous solution

$$\boldsymbol{e}_p^T\boldsymbol{f} \simeq \gamma(u_p). \tag{2.24}$$

The vector of Lagrange multipliers $\boldsymbol{f}$ is initially unknown. Thus, the potential flow problem in the presence of the impenetrable surface is

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}} \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \boldsymbol{f} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{s}'_b - \boldsymbol{s}_0 \end{pmatrix}. \tag{2.25}$$

This problem (2.25) has the structure of a generic saddle-point problem (Benzi, Golub & Liesen 2005). We will encounter many such systems in this work, so in Appendix A we summarize a solution approach, based on block-LU (LU, lower-upper) decomposition. The generated solution algorithm of (2.25) is

$$\boldsymbol{L}\boldsymbol{s}^* = -\boldsymbol{w}, \tag{2.26}$$

$$\boldsymbol{S}\boldsymbol{f} = \boldsymbol{s}'_b - \boldsymbol{s}_0 - {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{s}^*, \tag{2.27}$$

$$\boldsymbol{s} = \boldsymbol{s}^* - \boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f}, \tag{2.28}$$

where the Schur complement $\boldsymbol{S}$ is

$$\boldsymbol{S} = -{}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}. \tag{2.29}$$

Based on the properties of the matrices comprising $\boldsymbol{S}$, this operator is symmetric and negative definite, and therefore invertible. Its inverse $\boldsymbol{S}^{-1}$, also symmetric, maps a surface

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

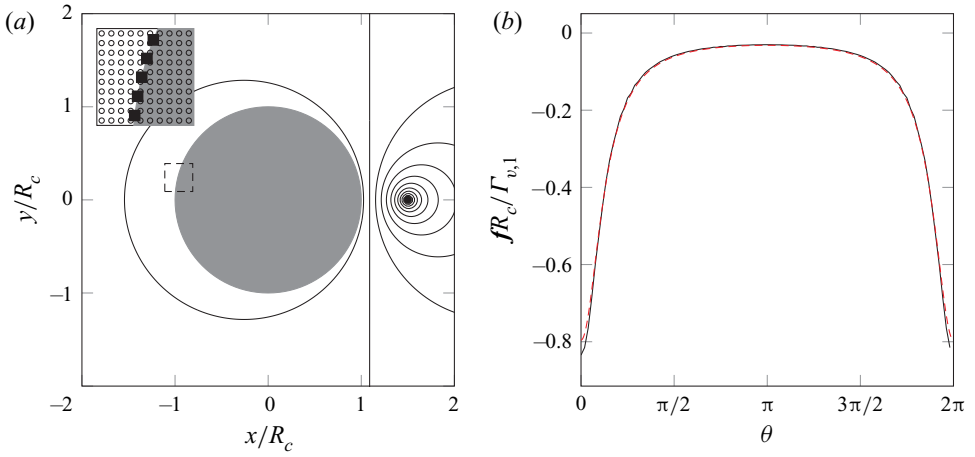


Figure 3. (*a*) Contours (——) of the discrete streamfunction for a point vortex (•) with strength $\Gamma_{v,1}$ at $(R_v, 0)$ near a circular cylinder of radius R_c and with a bound circulation $-\Gamma_{v,1}$. The inset shows a closeup of the nodes (○) and surface points (■) in the boxed area (- - - -). (*b*) The scaled discrete vortex sheet strength (——) as a function of the angle θ measured counter-clockwise from the positive x-axis. Here, $k = 1$ corresponds to the right-most point on the surface and increases counter-clockwise. Overlaid is the exact continuous solution (- - - -, red dashed). The simulation is performed with $R_v/R_c = 3/2$, $\Delta x/R_c = 0.03$ and $\Delta S/\Delta x = 2$.

distribution of streamfunction to a corresponding bound vortex sheet strength. Note that the computation of $\boldsymbol{f}$ through $\boldsymbol{S}^{-1}$ is sensitive to both the ratio of discrete surface spacing to the grid spacing $\Delta S/\Delta x$ and the choice of discrete Dirac delta function. On the one hand, small values for $\Delta S/\Delta x$ and discrete Dirac delta functions with small support generally both lead to more high-frequency noise in $\boldsymbol{f}$. This is because the underlying continuous problem for $\boldsymbol{S}^{-1}$ is a Fredholm integral equation of the first kind, which is ill posed. As a result, the discrete analogue problem is poorly conditioned (Goza *et al.* 2016). On the other hand, values for $\Delta S/\Delta x$ that are too high can lead to streamlines penetrating a surface. We found that values between one to four can provide a good balance between the smoothness of $\boldsymbol{f}$ and the accuracy of the streamlines near a surface.

We can describe this algorithm in words: first, solve for the intermediate streamfunction field, associated with vorticity in the fluid, but without regard for the presence of the surface. Second, find the bound vortex sheet whose associated streamfunction cancels the difference between the specified streamfunction on the surface and the intermediate streamfunction evaluated on the surface. Finally, correct the intermediate streamfunction field for the influence of the bound vortex sheet. A version of the Julia code that implements this algorithm, as well as the algorithms in the following sections, are available in the authors' Github repository (Beckers & Eldredge 2021).

We give two examples of the streamfunction with a body present and show the associated vortex sheet strength. Figure 3 shows a vortex near a circular cylinder and figure 4 shows a circular cylinder that translates horizontally. In both cases, the vortex sheet strength is in good agreement with the analytical solution from potential flow theory.

For later use, we note that the solution of (2.25) can also be written in inverse form using (A7)

$$\begin{pmatrix} \boldsymbol{s} \\ \boldsymbol{f} \end{pmatrix} = \begin{bmatrix} \boldsymbol{L}^{-1} + \boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{S}^{-1}\,{}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1} & -\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{S}^{-1} \\ -\boldsymbol{S}^{-1}\,{}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1} & \boldsymbol{S}^{-1} \end{bmatrix} \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{s}_b' - \boldsymbol{s}_0 \end{pmatrix}. \tag{2.30}$$

The matrix operator in (2.30) is the inverse of the basic saddle-point system.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

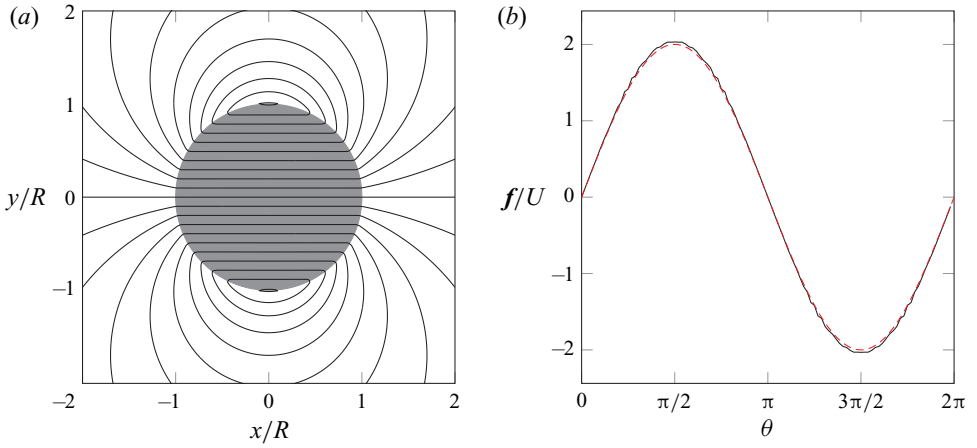


Figure 4. (*a*) Contours (——) of the discrete streamfunction for a horizontally translating circular cylinder with radius R, and (*b*) its scaled discrete vortex sheet strength (——). Overlaid is the exact continuous solution (- - - -, red dashed). The simulation is performed with $\Delta x/R = 0.03$ and $\Delta S/\Delta x = 2$.

2.3. *Non-uniqueness and discrete circulation*

In two-dimensional potential flows, there is no unique solution to problem (2.25), since one can choose any value for the uniform value $\boldsymbol{s}_0$ and still enforce the no-penetration condition. Equivalently, we can specify any circulation about the body and still enforce this condition. Let us determine the relationship between $\boldsymbol{s}_0$ and circulation. For later use, let us write this uniform surface streamfunction as $\boldsymbol{s}_0 = s_0 \boldsymbol{1}_{\mathcal{S}}$, where s_0 is a single scalar value. The discrete circulation Γ_b about the body is given by the sum of the bound vortex sheet data and can be written compactly as

$$\Gamma_b = \langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{f} \rangle_{\mathcal{S}^{Ns}}. \tag{2.31}$$

The discrete circulation of the vortex sheet in the solution (2.27) is

$$\Gamma_b = \langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{S}^{-1}(\boldsymbol{s}_b' + {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} \boldsymbol{L}^{-1} \boldsymbol{w}) \rangle_{\mathcal{S}^{Ns}} - s_0 \langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{S}^{-1} \boldsymbol{1}_{\mathcal{S}} \rangle_{\mathcal{S}^{Ns}}. \tag{2.32}$$

Note that we can obtain the same expression if we would use s_0 as a Lagrange multiplier to enforce the constraint (2.31) as

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}} & 0 \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 & \boldsymbol{1}_{\mathcal{S}} \\ 0 & \boldsymbol{1}_{\mathcal{S}}^T \boldsymbol{D}_{\Delta S} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \boldsymbol{f} \\ s_0 \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{s}_b' \\ \Gamma_b \end{pmatrix}, \tag{2.33}$$

where we use the fact that the inner product $\langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{f} \rangle_{\mathcal{S}^{Ns}}$ can, by its definition, be rewritten as $\boldsymbol{1}_{\mathcal{S}}^T \boldsymbol{D}_{\Delta S} \boldsymbol{f}$. Here, $\boldsymbol{D}_{\Delta S}$ is a diagonal matrix containing the surface element arc lengths. We will use s_0 in a similar way in the next section to enforce the Kutta condition and demonstrate how to solve the associated saddle-point system.

The scalar factor $\langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{S}^{-1} \boldsymbol{1}_{\mathcal{S}} \rangle_{\mathcal{S}^{Ns}}$ in expression (2.32) is a property of the set of points and their immersion into the Cartesian grid. Part of this factor, $\boldsymbol{S}^{-1} \boldsymbol{1}_{\mathcal{S}}$, represents the bound vortex sheet strength associated with a uniform, unit-strength streamfunction on the surface. This sheet has a particularly important role in some of the discussion to follow, so

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

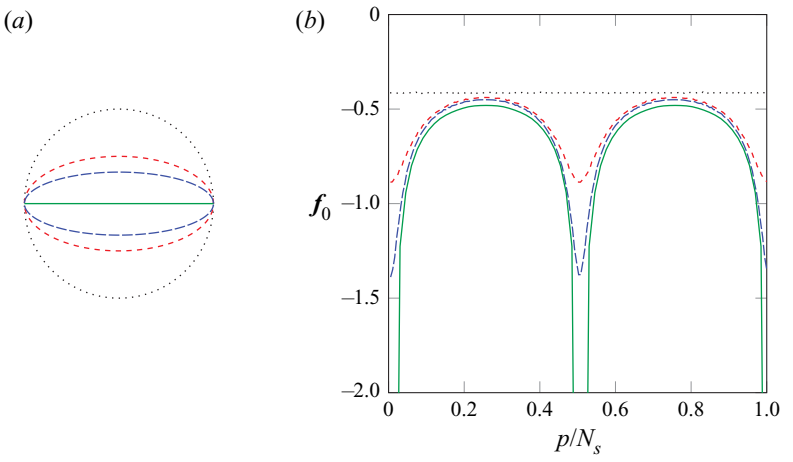


Figure 5. (*a*) Geometry and (*b*) the scaled bound vortex sheet strength $\boldsymbol{f}_0$, associated with a uniform, unit-strength streamfunction, as a function of the surface point index p divided by the number of surface points N_s for elliptical cylinders with different aspect ratios (AR): · · · · · ·, $AR = 1$; - - - - (red dashed), $AR = 2$; – – – (blue broken dashed), $AR = 3$; —— (green solid line), $AR = \infty$ (flat plate). The simulations are performed with $\Delta x/R = 0.004$ and $\Delta S/\Delta x = 3$.

we will denote its strength by $\boldsymbol{f}_0$:

$$\boldsymbol{f}_0 \equiv \boldsymbol{S}^{-1} 1_{\mathcal{S}}. \tag{2.34}$$

The transpose of $\boldsymbol{f}_0$, equal to $1_{\mathcal{S}}^T \boldsymbol{S}^{-1}$, calculates the circulation of the associated bound vortex sheet when it acts upon a surface streamfunction. Thus, the factor $\langle 1_{\mathcal{S}}, \boldsymbol{S}^{-1} 1_{\mathcal{S}} \rangle_{\mathcal{S}^{N_s}}$ is the circulation associated with a uniform, unit-strength surface streamfunction. We will refer to this as Γ_0:

$$\Gamma_0 \equiv \langle 1_{\mathcal{S}}, \boldsymbol{S}^{-1} 1_{\mathcal{S}} \rangle_{\mathcal{S}^{N_s}} \equiv \langle 1_{\mathcal{S}}, \boldsymbol{f}_0 \rangle_{\mathcal{S}^{N_s}} \equiv \langle \boldsymbol{f}_0, 1_{\mathcal{S}} \rangle_{\mathcal{S}^{N_s}}. \tag{2.35}$$

We can rewrite the inner product $\langle \boldsymbol{f}_0, 1_{\mathcal{S}} \rangle_{\mathcal{S}^{N_s}}$ as $\boldsymbol{f}_0^T \boldsymbol{D}_{\Delta S} 1_{\mathcal{S}}$, and we will define $\hat{\boldsymbol{f}}_0 = \boldsymbol{D}_{\Delta S} \boldsymbol{f}_0$, for shorthand in what follows. By this notation, $\Gamma_0 = \hat{\boldsymbol{f}}_0^T 1_{\mathcal{S}}$, and $\hat{\boldsymbol{f}}_0^T$ applied to any surface streamfunction yields the corresponding circulation.

Figure 5 shows the distribution of $\boldsymbol{f}_0$ for elliptical cylinders with different aspect ratios. For a circular cylinder, the flow due to a uniform streamfunction on the body corresponds to the flow when the cylinder is replaced by a point vortex at its centre. The streamlines are concentric circles and the tangential velocity is constant at a given radius. The resulting vortex sheet strength $\boldsymbol{f}_0$ therefore assumes a uniform distribution. The figure also demonstrates that, when the aspect ratio increases, the distribution gradually shows stronger variations near the edges of the major axis, corresponding to an acceleration and deceleration of the flow when it passes those edges. The distribution eventually turns singular at the edges of a flat plate as the flow now has to navigate around a sharp corner, which will be discussed in more detail in the next section. To clearly illustrate the emergence of the singularities, the simulations in this figure are performed with an extremely fine grid ($\Delta x/R = 0.004$). Note that such fine grids are not needed in general, as is demonstrated by the other figures in this work.

The last term in (2.32) illustrates the direct relationship between the scalar value s_0 and the bound circulation Γ_b, and we identified s_0 as a means of setting the circulation. Before we use it in the next section to enforce the Kutta condition, we will use it here for

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

an immediate purpose. The prescribed surface streamfunction $\boldsymbol{s}'_b$ (given by (2.21), with the uniform flow accounted for) may have some associated bound circulation, and it is desirable to adjust it by adding or subtracting a uniform value so that it has none. Equation (2.32) suggests that this circulation can be removed by setting s_0 to $\langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{S}^{-1}\boldsymbol{s}'_b\rangle_{\mathcal{S}^{N_s}}/\Gamma_0 \equiv \hat{\boldsymbol{f}}_0{}^T\boldsymbol{s}'_b/\Gamma_0$ and then subtracting this value (multiplied by the uniform vector $\boldsymbol{1}_{\mathcal{S}}$) from $\boldsymbol{s}'_b$. Overall, this process can be encapsulated in a circulation removal operator, $\boldsymbol{P}^{\Gamma}$, that acts upon a surface streamfunction, $\boldsymbol{q} \in \mathcal{S}^{N_s}$

$$\boldsymbol{P}^{\Gamma} \equiv \boldsymbol{I} - \frac{\boldsymbol{1}_{\mathcal{S}}\hat{\boldsymbol{f}}_0{}^T}{\Gamma_0}. \tag{2.36}$$

It is easy to verify that $\langle \boldsymbol{S}^{-1}\boldsymbol{1}_{\mathcal{S}}, \boldsymbol{P}^{\Gamma}\boldsymbol{q}\rangle_{\mathcal{S}^{N_s}} = \hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}^{\Gamma}\boldsymbol{q} = 0$ for any $\boldsymbol{q} \in \mathcal{S}^{N_s}$, so that the circulation of any surface streamfunction acted upon by $\boldsymbol{P}^{\Gamma}$ is indeed zero. It is important to observe, also, that $\boldsymbol{s}'_b$ can be replaced by $\boldsymbol{P}^{\Gamma}\boldsymbol{s}'_b$ without affecting the nature of the no-penetration condition. We also note that the composite operator $\boldsymbol{S}^{-1}\boldsymbol{P}^{\Gamma}$ is symmetric, just as $\boldsymbol{S}^{-1}$ is, since

$$\boldsymbol{S}^{-1}\boldsymbol{P}^{\Gamma} = \boldsymbol{S}^{-1} - \hat{\boldsymbol{f}}_0\hat{\boldsymbol{f}}_0{}^T. \tag{2.37}$$

2.4. *The Kutta condition*

For surfaces that contain convex edges, the vortex sheet strength assumes a singular behaviour in the vicinity of these edges, with a strength that depends on the interior angle of the edge: sharper edges have more singular behaviour. In the discrete representation of the surface, edges are only approximately represented by the sudden disruptions of positions in clusters of adjacent points, without any information about the surface normals. The behaviour in this discrete form is not quite singular, but the solution of (2.27) nonetheless exhibits a large and rapid change of amplitude.

If we seek to eliminate this behaviour, we must first have some means of exposing it. In fact, for any discretized surface, the essence of this nearly singular behaviour lies in the vector $\boldsymbol{f}_0$, and all other bound vortex sheets associated with the same surface share the same nearly singular behaviour. Thus, we will use a multiplicative decomposition of the vortex sheet strength

$$\boldsymbol{f} = \boldsymbol{f}_0 \circ \tilde{\boldsymbol{f}}, \tag{2.38}$$

where $\circ$ is the Hadamard product. This decomposed form isolates the singular behaviour into $\boldsymbol{f}_0$, and $\tilde{\boldsymbol{f}}$ is a relatively smoother vector of surface point data. In the regularization operation on $\boldsymbol{f}$, we can absorb $\boldsymbol{f}_0$ into ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}$, first noting that the Hadamard product can alternatively be written with the help of a diagonal matrix

$$\boldsymbol{f} = \boldsymbol{f}_0 \circ \tilde{\boldsymbol{f}} = \boldsymbol{D}_{\boldsymbol{f}_0}\tilde{\boldsymbol{f}}. \tag{2.39}$$

Then, we can define a re-scaled regularization operator

$${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f} = {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{D}_{\boldsymbol{f}_0}\tilde{\boldsymbol{f}} = {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}}\tilde{\boldsymbol{f}}. \tag{2.40}$$

The re-scaled operator ${}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} = {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{D}_{\boldsymbol{f}_0}$ can, in turn, be absorbed into the Schur complement, defining $\tilde{\boldsymbol{S}} = -{}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1}{}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} = \boldsymbol{S}\boldsymbol{D}_{\boldsymbol{f}_0}$. A useful property of $\tilde{\boldsymbol{S}}$ is that it

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

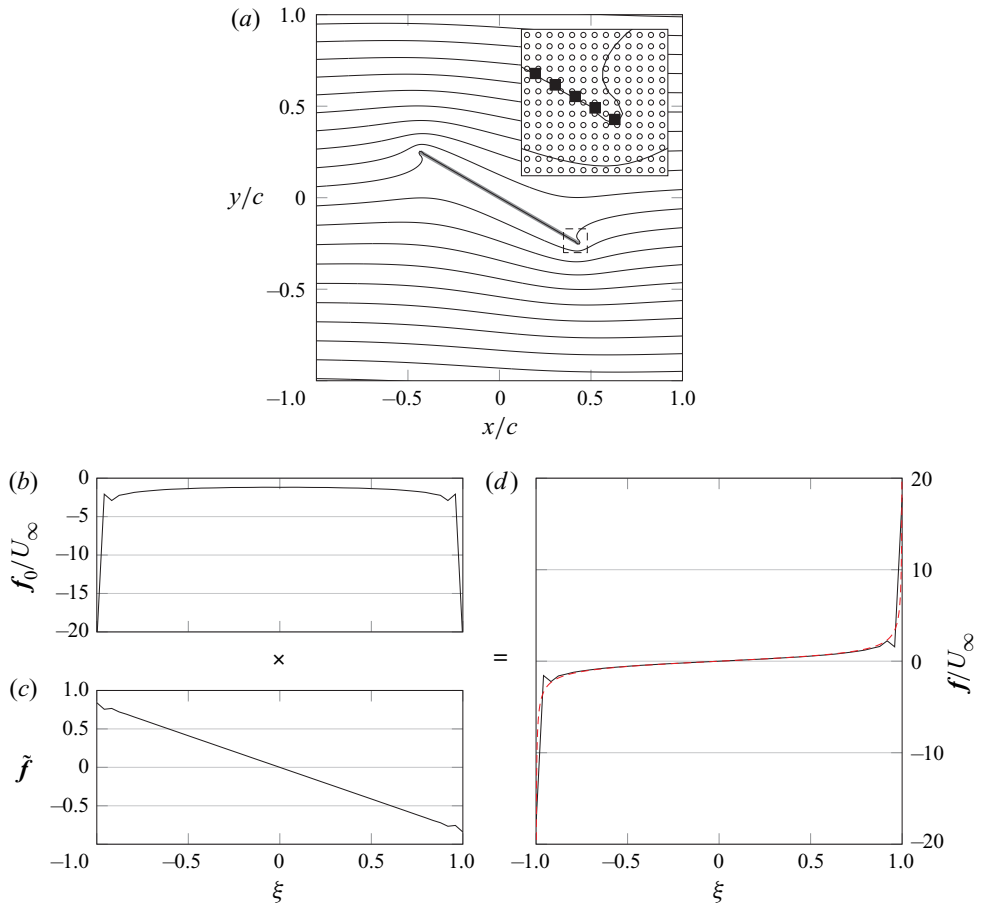


Figure 6. (*a*) Contours (——) of the discrete streamfunction for a flat plate of length c, at 30° in a uniform flow U_∞ without enforcement of the Kutta condition. The inset shows a closeup of the nodes (○) and surface points (■) in the boxed area (- - - -). The point-wise product of (*b*) the discrete vortex sheet strength associated with a uniform, unit-strength streamfunction on the body, and (*c*) the non-singular vector that results from re-scaling the regularization operator, composes (*d*) the discrete vortex sheet strength. Overlaid is the exact continuous solution (- - - -, red dashed). The simulation is performed with $\Delta x/c = 0.01$ and $\Delta S/\Delta x = 2$.

preserves uniform vectors

$$\tilde{\boldsymbol{S}}\boldsymbol{1}_{\mathcal{S}} = \boldsymbol{S}\boldsymbol{D}_{\boldsymbol{f}_0}\boldsymbol{1}_{\mathcal{S}} = \boldsymbol{S}\boldsymbol{f}_0 = \boldsymbol{1}_{\mathcal{S}}, \quad \tilde{\boldsymbol{S}}^{-1}\boldsymbol{1}_{\mathcal{S}} = \boldsymbol{1}_{\mathcal{S}}. \tag{2.41a,b}$$

The decomposition of the vortex sheet strength is demonstrated in figure 6 for a flat plate in a uniform flow. We plot the vortex sheet strength against the scaled coordinate ξ, which varies along the plate from -1 at the leading edge to 1 at the trailing edge. As expected, the vortex sheet strength $\boldsymbol{f}$ shows large amplitude variations at the sharp leading and trailing edges, corresponding to the singularities in the distribution of the continuous vortex sheet strength. By use of decomposition (2.38), these discrete singularities are retained in $\boldsymbol{f}_0$ and we are left with a non-singular $\tilde{\boldsymbol{f}}$, which varies almost linearly with ξ.

The Kutta condition corresponds to annihilating the nearly singular behaviour at a surface point. At such points, we will set the corresponding value of $\tilde{\boldsymbol{f}}$ to zero. Suppose we wish to enforce the Kutta condition at an edge corresponding to surface point k.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

The condition is

$$\boldsymbol{e}_k^T \tilde{\boldsymbol{f}} = 0. \tag{2.42}$$

2.4.1. *Using the Kutta condition in a steady-state problem*

We will first take the steady-state approach to enforce the Kutta condition: allow the bound circulation to be set appropriately, with the implicit understanding that there is a starting vortex of equal and opposite circulation at infinity that preserves the Kelvin circulation theorem. The Lagrange multiplier for this constraint will not be Γ_b, but s_0, similar to (2.33). We also use the circulation removal operator to adjust the imposed surface streamfunction

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} & 0 \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 & 1_{\mathcal{S}} \\ 0 & \boldsymbol{e}_k^T & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \\ s_0 \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{P}^{\Gamma} s_b' \\ 0 \end{pmatrix}. \tag{2.43}$$

This block system, like the earlier one in (2.25), has a saddle-point form, and we can reduce it by the same block-LU decomposition to develop a solution algorithm. We will interpret it in the general form (A3), with the upper-left 2×2 block taking the role of $\mathcal{A}$, the solution vector $\boldsymbol{x}$ and constraint force $\boldsymbol{y}$ set, respectively, to

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \end{pmatrix}, \quad \boldsymbol{y} = s_0, \tag{2.44a,b}$$

the remaining operators set to

$$\mathcal{B}_2 = \begin{bmatrix} 0 & \boldsymbol{e}_k^T \end{bmatrix}, \quad \mathcal{B}_1^T = \begin{bmatrix} 0 \\ 1_{\mathcal{S}} \end{bmatrix}, \quad \mathcal{C} = 0, \tag{2.45a–c}$$

and the right-hand side vectors set to

$$\boldsymbol{r}_{1_{\mathcal{N}}} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{P}^{\Gamma} s_b' \end{pmatrix}, \quad \boldsymbol{r}_{\boldsymbol{2}} = 0. \tag{2.46a,b}$$

We note that block $\mathcal{A}$ has the original form of the system before the Kutta constraint (2.25), although with the slight modification of a re-scaled regularization operator, and we already have the inverse of $\mathcal{A}$ available from (2.30). The solution of this original system forms the intermediate solution of the full system endowed with the Kutta condition

$$\tilde{\boldsymbol{f}}^* = \tilde{\boldsymbol{S}}^{-1}\left(\boldsymbol{P}^{\Gamma} s_b' + {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1}\boldsymbol{w}\right), \quad \boldsymbol{s}^* = -\boldsymbol{L}^{-1}\left(\boldsymbol{w} + {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}}\tilde{\boldsymbol{f}}^*\right). \tag{2.47a,b}$$

Then, using the general procedure outlined in Appendix A, the solution of the full system (2.43) is easy to develop; its Schur complement is simply

$$\mathcal{S} = -1. \tag{2.48}$$

Applying the general solution equations, and using the property (2.41*a*,*b*) to simplify the resulting operators, it can be shown that the solution is

$$s_0 = \boldsymbol{e}_k^T \tilde{\boldsymbol{f}}^*, \tag{2.49}$$

$$\begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{s}^* \\ \tilde{\boldsymbol{f}}^* \end{pmatrix} - \begin{bmatrix} -\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} 1_{\mathcal{S}} \\ 1_{\mathcal{S}} \end{bmatrix} \boldsymbol{e}_k^T \tilde{\boldsymbol{f}}^*. \tag{2.50}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

The entire solution can be written more compactly as

$$\tilde{\boldsymbol{f}} = \boldsymbol{P}_k^K \tilde{\boldsymbol{S}}^{-1} \left(\boldsymbol{P}^{\Gamma} \boldsymbol{s}_b' + {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} \boldsymbol{L}^{-1} \boldsymbol{w} \right), \tag{2.51}$$

$$\boldsymbol{s} = -\boldsymbol{L}^{-1} \left(\boldsymbol{w} + {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} \tilde{\boldsymbol{f}} \right), \tag{2.52}$$

where we have defined the Kutta projection operator,

$$\boldsymbol{P}_k^K \equiv \boldsymbol{I} - \boldsymbol{1}_{\mathcal{S}} \boldsymbol{e}_k^T, \tag{2.53}$$

which acts upon the (non-singular part of the) bound vortex sheet vector, subtracting the value at point k from every point, including at k itself.

Note that the Lagrange multiplier for the Kutta condition takes the simple value given by (2.49), revealing that the additional streamfunction on the surface is exactly the value of the intermediate bound vortex sheet at the Kutta point k.

The application of the Kutta condition to a steady-state problem is demonstrated in figure 7 on the flat plate problem that was introduced in the previous section. By constraining the trailing edge point of $\tilde{\boldsymbol{f}}$, its whole distribution is shifted upward such that the trailing edge value equals zero. The resulting streamfunction indicates that the flow then indeed leaves the trailing edge smoothly.

Figure 7 also shows a spatial grid refinement analysis of the non-singular part of the vortex sheet strength. For this analysis, we can use a multiplicative decomposition for the continuous vortex sheet strength, $\gamma(\xi) = \gamma_0(\xi)\tilde{\gamma}(\xi)$, analogous to (2.38). The continuous counterpart of $\boldsymbol{f}_0$ can be found as the bound vortex sheet strength of a circular cylinder with a point vortex of strength Γ_0 at its centre, conformally mapped to a flat plate. Due to the definition of $\boldsymbol{f}_0$, the circulation Γ_0 depends on the grid spacing. Therefore, if we define γ_0 as the continuous counterpart of $\boldsymbol{f}_0/\Gamma_0$, we obtain the grid-independent solution $\tilde{\gamma}(\xi) = -\pi c U_\infty \sin(\alpha)(1-\xi)$ as the continuous counterpart of $\Gamma_0\tilde{\boldsymbol{f}}$ for a flat plate with the Kutta condition enforced. We can then define the error for the non-singular part of the vortex sheet strength as $\epsilon_{\Gamma_0\tilde{f}} = \|\Gamma_0\tilde{\boldsymbol{f}} - \tilde{\gamma}(\xi(\boldsymbol{r}_x, \boldsymbol{r}_y)\|_2 / \|\tilde{\gamma}(\xi(\boldsymbol{r}_x, \boldsymbol{r}_y)\|_2$. The refinement analysis verifies that the immersed boundary projection method using the vortex sheet strength decomposition is approximately first-order accurate in Δx. Furthermore, the figure shows the streamlines near the plate, the vortex sheet strength and the error for four different values of $\Delta S/\Delta x$. These confirm that a lower value increases the noise in the vortex sheet strength, but a higher value can lead to the streamlines penetrating the surface, as discussed previously.

2.4.2. *Using the Kutta condition to set a new vortex element*

In the previous section, we used the Kutta condition to set the bound circulation but did not explicitly create a new vortex element. This vortex element was assumed to lie at infinity so that its effect was negligible except insofar as it left equal but opposite circulation about the body.

In this section, we will create a new vortex element in the vicinity of the edge at which we are applying the Kutta condition. We will thus seek to establish the strength of this new element and to do so in such a manner that the overall circulation of the flow is conserved. Once the element is created, it will be allowed to advect with the local fluid velocity.

Let us assume that the new vortex element (which we label with the subscript 1) is introduced at some point in physical space, and that its immersion into the Cartesian grid

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

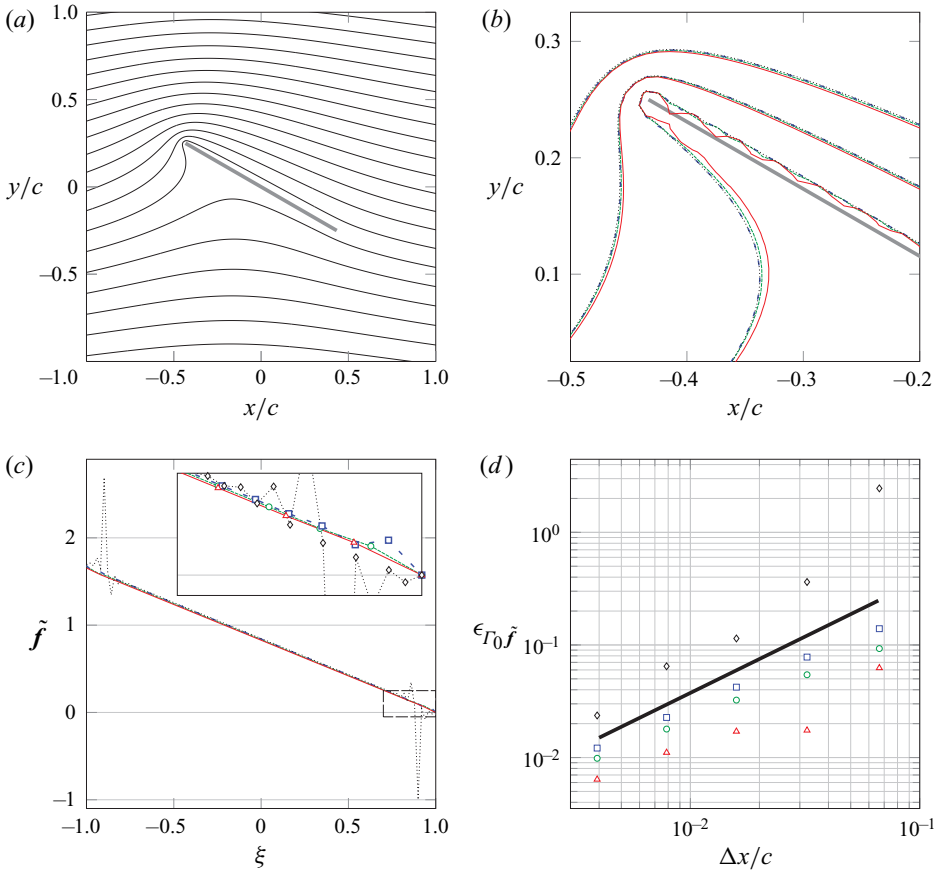


Figure 7. (*a*) Contours (——) of the steady, discrete streamfunction for a flat plate of length c at 30° in a uniform flow with enforcement of the Kutta condition at the trailing edge. (*b*) Zoomed-in image of the leading edge showing the streamlines for three values of the streamfunction, (*c*) the non-singular part of the associated discrete vortex sheet strength for $\Delta x/c = 0.01$ with an inset enlarging the boxed area (— — —) and (*d*) the variation of its error with grid spacing for different values for $\Delta S/\Delta x$: · · · · · ·, and ◇, $\Delta S/\Delta x = 1$; – – – (broken dashed blue) and □ (blue), $\Delta S/\Delta x = 2$; - - - - (green dashed) and ○ (green), $\Delta S/\Delta x = 3$; —— (red solid line) and △ (red), $\Delta S/\Delta x = 4$. Overlaid is an error (——) that scales as Δx.

is described by $\boldsymbol{d}_1$ and that its strength (i.e. its circulation) is $\Gamma_{v,1}$. Thus, the fluid vorticity after this new element's introduction can be written as

$$\boldsymbol{w} + \Gamma_{v,1}\boldsymbol{d}_1. \tag{2.54}$$

The Kutta condition (2.42) is still to be enforced. We also seek to ensure that the total circulation is zero to satisfy Kelvin's circulation theorem. (We are assuming that the flow has started from rest.) Let us denote the circulation of the existing fluid vorticity $\boldsymbol{w}$ by

$$\Gamma_{\boldsymbol{w}} = \langle \boldsymbol{1}_{\mathcal{N}}, \boldsymbol{w} \rangle_{\mathcal{N}}. \tag{2.55}$$

Then, the circulation constraint is

$$\langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{f} \rangle_{\mathcal{S}^{N_s}} + \Gamma_{v,1} + \Gamma_{\boldsymbol{w}} = 0. \tag{2.56}$$

The circulation of the bound vortex sheet $\boldsymbol{f}$ can be re-written in terms of the non-singular part of the sheet as $\langle \boldsymbol{1}_{\mathcal{S}}, \boldsymbol{f} \rangle_{\mathcal{S}^{N_s}} = \langle \boldsymbol{f}_0, \tilde{\boldsymbol{f}} \rangle_{\mathcal{S}^{N_s}} = \hat{\boldsymbol{f}}_0^{\,T} \tilde{\boldsymbol{f}}$.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

With these two constraints, the overall saddle-point system of equations is

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} & 0 & \boldsymbol{d}_1 \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 & \boldsymbol{1}_{\mathcal{S}} & 0 \\ 0 & \boldsymbol{e}_k^T & 0 & 0 \\ 0 & \hat{\boldsymbol{f}}_0{}^T & 0 & 1 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \\ s_0 \\ \Gamma_{v,1} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{P}^{\Gamma}\boldsymbol{s}'_b \\ 0 \\ -\Gamma_w \end{pmatrix}. \tag{2.57}$$

Again, the basic saddle-point matrix constitutes the upper left 2×2 block $\mathcal{A}$ and the solution vector $\boldsymbol{x}_{\mathcal{N}}$ is as before. The constraint force vector is

$$\boldsymbol{y} = \begin{pmatrix} s_0 \\ \Gamma_{v,1} \end{pmatrix}, \tag{2.58}$$

and the remaining vectors and operators are now

$$\boldsymbol{r_2} = \begin{pmatrix} 0 \\ -\Gamma_w \end{pmatrix}, \quad \mathcal{B}_2 = \begin{bmatrix} 0 & \boldsymbol{e}_k^T \\ 0 & \hat{\boldsymbol{f}}_0{}^T \end{bmatrix}, \quad \mathcal{B}_1^T = \begin{bmatrix} 0 & \boldsymbol{d}_1 \\ \boldsymbol{1}_{\mathcal{S}} & 0 \end{bmatrix}, \quad \mathcal{C} = -\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \tag{2.59a–d}$$

The solution algorithm follows, once again, from the equations in Appendix A. After carrying out the block matrix multiplications, it can be shown that the Schur complement (A2) is the 2×2 matrix

$$\mathcal{S} = \begin{bmatrix} -1 & \boldsymbol{e}_k^T\tilde{\boldsymbol{f}}_1 \\ -\hat{\boldsymbol{f}}_0{}^T\boldsymbol{1}_{\mathcal{S}} & 1 + \hat{\boldsymbol{f}}_0{}^T\tilde{\boldsymbol{f}}_1, \end{bmatrix} \tag{2.60}$$

where, for convenience, we have defined

$$\tilde{\boldsymbol{f}}_1 = \tilde{\boldsymbol{S}}^{-1\,\mathcal{S}}\boldsymbol{E}_{\mathcal{N}}\boldsymbol{L}^{-1}\boldsymbol{d}_1, \tag{2.61}$$

which represents the (non-singular part of the) strength of the vortex sheet that 'reacts' to the presence of a unit-strength vortex $\boldsymbol{d}_1$ immersed into the grid, cancelling that vortex's induced velocity on the surface. The term $\hat{\boldsymbol{f}}_0{}^T\tilde{\boldsymbol{f}}_1$ represents this sheet's bound circulation and $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}_1$ is its contribution to the Kutta condition at point k. The problem (A6) for the constraint forces s_0 and $\Gamma_{v,1}$ is then

$$\begin{bmatrix} -1 & \boldsymbol{e}_k^T\tilde{\boldsymbol{f}}_1 \\ -\hat{\boldsymbol{f}}_0{}^T\boldsymbol{1}_{\mathcal{S}} & 1 + \hat{\boldsymbol{f}}_0{}^T\tilde{\boldsymbol{f}}_1, \end{bmatrix} \begin{pmatrix} s_0 \\ \Gamma_{v,1} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^* \\ -\Gamma_w - \hat{\boldsymbol{f}}_0{}^T\tilde{\boldsymbol{f}}^* \end{pmatrix}, \tag{2.62}$$

where the intermediate solution $\tilde{\boldsymbol{f}}^*$ is available from (2.47*a*,*b*).

The determinant of this Schur complement matrix is $-1 - \hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_k^K\tilde{\boldsymbol{f}}_1$, which represents the negative of the circulation of the unit vortex and its associated vortex sheet, after the Kutta condition has been enforced on this sheet. It is straightforward then to calculate the strength of the new vortex $\Gamma_{v,1}$ and the additional uniform surface streamfunction, s_0

$$\Gamma_{v,1} = \frac{-\Gamma_w - \hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_k^K\tilde{\boldsymbol{f}}^*}{1 + \hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_k^K\tilde{\boldsymbol{f}}_1}, \quad s_0 = \boldsymbol{e}_k^T(\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1). \tag{2.63a,b}$$

From these, we can then obtain the vortex sheet strength and the fluid streamfunction,

$$\tilde{\boldsymbol{f}} = \boldsymbol{P}_k^K(\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1), \quad \boldsymbol{s} = -\boldsymbol{L}^{-1}(\boldsymbol{w} + \Gamma_{v,1}\boldsymbol{d}_1 + {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}}\tilde{\boldsymbol{f}}). \tag{2.64a,b}$$

The intermediate solution, which corresponds to the flow associated with existing vorticity, is corrected here with the new vortex 1 to enforce the Kutta condition at the point k.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

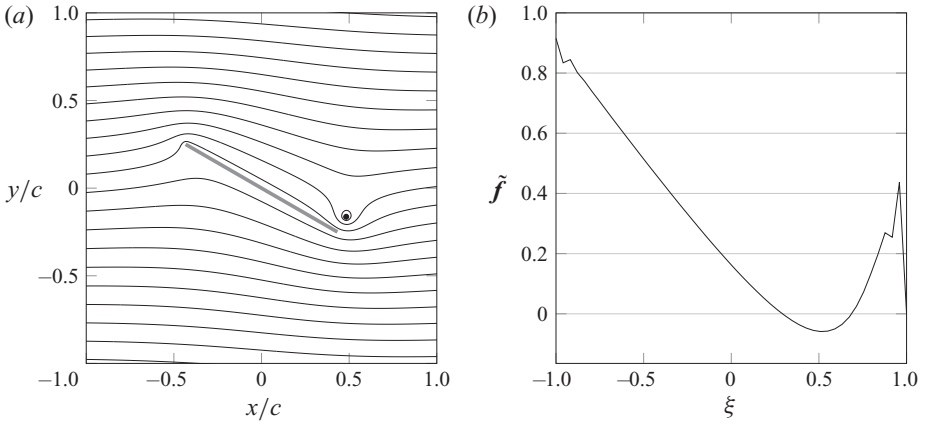


Figure 8. (*a*) Contours (——) of the unsteady, discrete streamfunction for a flat plate of length c at 30° in a uniform flow with release of vorticity into a point vortex (•) for enforcement of the Kutta condition at the trailing edge. (*b*) The non-singular part of the associated discrete vortex sheet strength. The simulation is performed with $\Delta x/c = 0.01$ and $\Delta S/\Delta x = 2$.

We now first demonstrate the enforcement of the Kutta condition in unsteady flow on the flat plate problem with a point vortex near the trailing edge to enforce the Kutta condition at that edge. We position the point vortex at a distance $10\Delta t U_\infty$ from the edge in the direction of the free stream, perpendicular to the plate. Figure 8 demonstrates that, because of the proximity of the point vortex to the flat plate, $\tilde{\boldsymbol{f}}$ exhibits a quick variation at the surface points that lie closest to the point vortex. The value at the trailing edge point itself is still constrained to zero and the flow again leaves the edge smoothly. This situation corresponds to the flow right after impulsively starting a uniform flow around a flat plate and the point vortex now represents the starting vortex. The second demonstration of the method (figure 9) is the repeated enforcement of the Kutta condition on a NACA0012 airfoil while advancing the positions of the point vortices in time after impulsively starting the flow. This simulation, and all following simulations, use forward Euler time stepping, unless noted otherwise, and new point vortices are inserted at one third of the way from the edge to the last released vortex from that edge. The streamlines in the figure show that a strong initial vortex and weaker subsequent vortices were created and convected downstream. In turn, at each time step, the airfoil obtained a circulation that enforces the Kutta condition.

2.4.3. *Applying more than one Kutta condition on a body*

Suppose we wish to enforce the Kutta condition at two edges of the body – at points k_1 and k_2 – instead of one. Each such point has a constraint

$$\boldsymbol{e}_{k_j}^T \tilde{\boldsymbol{f}} = 0, \quad j = 1, 2. \tag{2.65}$$

For two such constraints, we need two Lagrange multipliers: the strengths of two new vortices, $\Gamma_{v,1}$ and $\Gamma_{v,2}$, immersed into the grid with $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$, respectively; and we still need the Lagrange multiplier s_0 to ensure that Kelvin's circulation theorem is also enforced. The system in the previous section is thus easily generalized to the

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

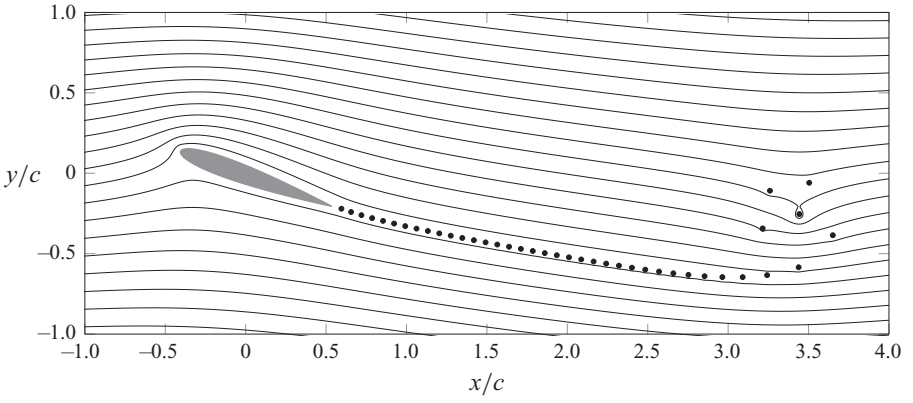


Figure 9. Contours (——) of the unsteady, discrete streamfunction for a NACA0012 airfoil with chord length c at 20°, three convective times after impulsively starting a uniform flow U_∞. Vorticity is released into a stream of point vortices (•) for enforcement of the Kutta condition at the trailing edge. The simulation is performed with $\Delta x/c = 0.01$, $\Delta S/\Delta x = 1.5$ and $\Delta t U_\infty/c = 0.075$.

following:

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} & 0 & \boldsymbol{d}_1 & \boldsymbol{d}_2 \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 & \boldsymbol{1}_{\mathcal{S}} & 0 & 0 \\ 0 & \boldsymbol{e}_{k_1}^T & 0 & 0 & 0 \\ 0 & \boldsymbol{e}_{k_2}^T & 0 & 0 & 0 \\ 0 & \hat{\boldsymbol{f}}_0{}^T & 0 & 1 & 1 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \\ s_0 \\ \Gamma_{v,1} \\ \Gamma_{v,2} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{P}^{\Gamma} \boldsymbol{s}_b' \\ 0 \\ 0 \\ -\Gamma_{\boldsymbol{w}} \end{pmatrix}. \tag{2.66}$$

The system is reduced in the same manner as before, with the same intermediate solution obtained from the basic system (2.25). Now, the Schur complement problem for the constraint forces takes the form

$$\begin{bmatrix} -1 & \boldsymbol{e}_{k_1}^T \tilde{\boldsymbol{f}}_1 & \boldsymbol{e}_{k_1}^T \tilde{\boldsymbol{f}}_2 \\ -1 & \boldsymbol{e}_{k_2}^T \tilde{\boldsymbol{f}}_1 & \boldsymbol{e}_{k_2}^T \tilde{\boldsymbol{f}}_2 \\ -\hat{\boldsymbol{f}}_0{}^T \boldsymbol{1}_{\mathcal{S}} & 1 + \hat{\boldsymbol{f}}_0{}^T \tilde{\boldsymbol{f}}_1 & 1 + \hat{\boldsymbol{f}}_0{}^T \tilde{\boldsymbol{f}}_2, \end{bmatrix} \begin{pmatrix} s_0 \\ \Gamma_{v,1} \\ \Gamma_{v,2} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{e}_{k_1}^T \tilde{\boldsymbol{f}}^* \\ -\boldsymbol{e}_{k_2}^T \tilde{\boldsymbol{f}}^* \\ -\Gamma_{\boldsymbol{w}} - \hat{\boldsymbol{f}}_0{}^T \tilde{\boldsymbol{f}}^* \end{pmatrix}, \tag{2.67}$$

where we have now defined bound vortex sheets associated with each of the two new vortices (with unit strengths)

$$\tilde{\boldsymbol{f}}_j = \tilde{\boldsymbol{S}}^{-1\,\mathcal{S}} \boldsymbol{E}_{\mathcal{N}} \boldsymbol{L}^{-1} \boldsymbol{d}_j, \tag{2.68}$$

for $j = 1, 2$. It is interesting to note that, if we take the difference between the two Kutta constraints, we obtain

$$\left(\boldsymbol{e}_{k_1}^T - \boldsymbol{e}_{k_2}^T\right)(\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1 + \Gamma_{v,2}\tilde{\boldsymbol{f}}_2) = 0. \tag{2.69}$$

It can be shown that this Schur complement problem can be split into

$$\begin{bmatrix} 1 + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_1}^K \tilde{\boldsymbol{f}}_1 & 1 + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_1}^K \tilde{\boldsymbol{f}}_2 \\ 1 + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_2}^K \tilde{\boldsymbol{f}}_1 & 1 + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_2}^K \tilde{\boldsymbol{f}}_2 \end{bmatrix} \begin{pmatrix} \Gamma_{v,1} \\ \Gamma_{v,2} \end{pmatrix} = - \begin{pmatrix} \Gamma_{\boldsymbol{w}} + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_1}^K \tilde{\boldsymbol{f}}^* \\ \Gamma_{\boldsymbol{w}} + \hat{\boldsymbol{f}}_0{}^T \boldsymbol{P}_{k_2}^K \tilde{\boldsymbol{f}}^* \end{pmatrix} \tag{2.70}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

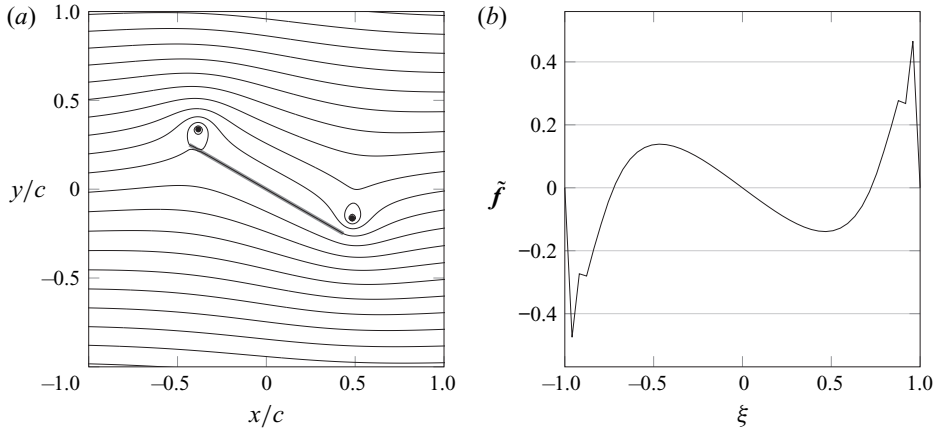


Figure 10. (*a*) Contours (——) of the unsteady, discrete streamfunction for a flat plate of length c at 30° in a uniform flow with release of vorticity into two point vortices (•) for enforcement of the Kutta condition at both edges. (*b*) The non-singular part of the associated discrete vortex sheet strength. The simulation is performed with $\Delta x/c = 0.01$ and $\Delta S/\Delta x = 2$.

and

$$s_0 = \tfrac{1}{2}\left(\boldsymbol{e}_{k_1}^T + \boldsymbol{e}_{k_2}^T\right)(\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1 + \Gamma_{v,2}\tilde{\boldsymbol{f}}_2). \tag{2.71}$$

The latter equation, when combined with (2.69), reveals that the value of the vortex sheet strength $\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1 + \Gamma_{v,2}\tilde{\boldsymbol{f}}_2$ is the same at both Kutta points and equal to s_0.

Equation (2.70) can be solved easily for the strengths of the two new point vortices. Then, the solution for the vortex sheet strength and streamfunction are

$$\tilde{\boldsymbol{f}} = \tfrac{1}{2}\left(\boldsymbol{P}_{k_1}^K + \boldsymbol{P}_{k_2}^K\right)(\tilde{\boldsymbol{f}}^* + \Gamma_{v,1}\tilde{\boldsymbol{f}}_1 + \Gamma_{v,2}\tilde{\boldsymbol{f}}_2), \quad \boldsymbol{s} = -\boldsymbol{L}^{-1}(\boldsymbol{w} + \Gamma_{v,1}\boldsymbol{d}_1 + \Gamma_{v,2}\boldsymbol{d}_2 + {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}}\tilde{\boldsymbol{f}}). \tag{2.72a,b}$$

We now apply this method in figure 10 to enforce the Kutta condition at the leading and trailing edge of our flat plate problem. We position a point vortex close to each edge and observe again that $\tilde{\boldsymbol{f}}$ shows strong variation at the surface points closest to the two point vortices. The contours of the streamfunction indicate that the flow indeed leaves the edges smoothly. Like the previous case, this solution corresponds to the flow right after impulsively starting a uniform flow around a flat plate, but unlike the previous case, the flow now separates at the leading edge.

It should be observed that these solutions are posed in a manner easily extensible to an arbitrary number of edges.

2.5. *Generalized edge condition*

In the previous section, we demonstrated the means of annihilating the (nearly) singular behaviour at edges on a discretized surface. In some cases, our desire is not to annihilate this behaviour, but simply to keep it within some bounds. In the analytical treatment of potential flow problems, this objective is served by placing an inequality constraint on the edge-suction parameter (Ramesh *et al.* 2014; Darakananda & Eldredge 2019; Eldredge 2019). That parameter is proportional to the coefficient on the bound vortex sheet strength's singularity (Eldredge 2019), so in this discrete setting, in which we have

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

extracted the singular part of $\boldsymbol{f}$ in the form of $\boldsymbol{f}_0$, we expect the suction parameter to be related to the value of $\tilde{\boldsymbol{f}}$ at the edge. In fact, by simple comparison, it can be shown that

$$\boldsymbol{e}_k^T\tilde{\boldsymbol{f}} = -\frac{2\pi c}{\Gamma_0}\sigma_k \tag{2.73}$$

for a flat plate of length c, where σ_k is the suction parameter at the edge corresponding to point k.

Let σ_k^{min} and σ_k^{max} denote the minimum and maximum tolerable values of σ_k at edge k. We then seek to confine the suction parameter to the range $\sigma_k^{min} \leqslant \sigma_k \leqslant \sigma_k^{max}$. This generalized edge constraint is placed on the suction parameter of the intermediate sheet $\tilde{\boldsymbol{f}}^*$. To avoid confusion, we will redefine the bounds based on this non-singular part of the vortex sheet rather than σ_k itself; for this, we define $\tilde{f}_k^{min} = -2\pi c\sigma_k^{max}/\Gamma_0$ and $\tilde{f}_k^{max} = -2\pi c\sigma_k^{min}/\Gamma_0$ if Γ_0 is positive or $\tilde{f}_k^{max} = -2\pi c\sigma_k^{max}/\Gamma_0$ and $\tilde{f}_k^{min} = -2\pi c\sigma_k^{min}/\Gamma_0$ if Γ_0 is negative. Thus, we inspect whether the value $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^*$ lies in the range

$$\tilde{f}_k^{min} \leqslant \boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^* \leqslant \tilde{f}_k^{max}. \tag{2.74}$$

If $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^*$ lies within this range, then no new vortex is created near the edge (or equivalently, a new vortex of zero strength is created); if $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^* > \tilde{f}_k^{max}$, then we create a new vortex so that $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}} = \tilde{f}_k^{max}$; and if $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}}^* < \tilde{f}_k^{min}$, then we do the same, but now so that $\boldsymbol{e}_k^T\tilde{\boldsymbol{f}} = \tilde{f}_k^{min}$. Note that the Kutta condition simply corresponds to setting $\tilde{f}_k^{min} = \tilde{f}_k^{max} = 0$.

We can easily accommodate these constraints into our solution approach for enforcing the Kutta condition from the previous section: in the case of two edges, by modifying the right-hand side vector of (2.70) (if the edge suction lies outside of its bounds) or setting the vortex strength corresponding to that edge to zero. For example, suppose that $\boldsymbol{e}_{k_1}^T\tilde{\boldsymbol{f}}^* < \tilde{f}_{k_1}^{min}$ and $\boldsymbol{e}_{k_2}^T\tilde{\boldsymbol{f}}^* > \tilde{f}_{k_2}^{max}$; then we solve the system

$$\begin{bmatrix} 1+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_1}^K\tilde{\boldsymbol{f}}_1 & 1+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_1}^K\tilde{\boldsymbol{f}}_2 \\ 1+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_2}^K\tilde{\boldsymbol{f}}_1 & 1+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_2}^K\tilde{\boldsymbol{f}}_2 \end{bmatrix}\begin{pmatrix}\Gamma_{v,1}\\ \Gamma_{v,2}\end{pmatrix} = -\begin{pmatrix}\Gamma_w+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_1}^K\tilde{\boldsymbol{f}}^*+\Gamma_0\tilde{f}_{k_1}^{min}\\ \Gamma_w+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_2}^K\tilde{\boldsymbol{f}}^*+\Gamma_0\tilde{f}_{k_2}^{max}\end{pmatrix}. \tag{2.75}$$

But if, say, $\tilde{f}_{k_2}^{min} \leqslant \boldsymbol{e}_{k_2}^T\tilde{\boldsymbol{f}}^* \leqslant \tilde{f}_{k_2}^{max}$, then we set $\Gamma_{v,2} = 0$ and this system reduces to

$$\Gamma_{v,1} = -\frac{\Gamma_w+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_1}^K\tilde{\boldsymbol{f}}^*+\Gamma_0\tilde{f}_{k_1}^{min}}{1+\hat{\boldsymbol{f}}_0{}^T\boldsymbol{P}_{k_1}^K\tilde{\boldsymbol{f}}_1}. \tag{2.76}$$

The effect of applying these generalized edge conditions to the leading edge of a flat plate is shown in figure 11 for the first instants after impulsively starting a uniform flow. The positions of the point vortices emanating from the leading edge in the figure indicate that as σ_k^{max} increases, the stream of point vortices is swept back from the edge. At the trailing edge, the Kutta condition is enforced in each case and the positions of the point vortices overlap, as they are not yet influenced by the different situations at the leading edge in these first instants.

2.6. *Pressure, force and the added mass*

In this section, we present the means for calculating pressure and force (and moment) in the grid-based treatment.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

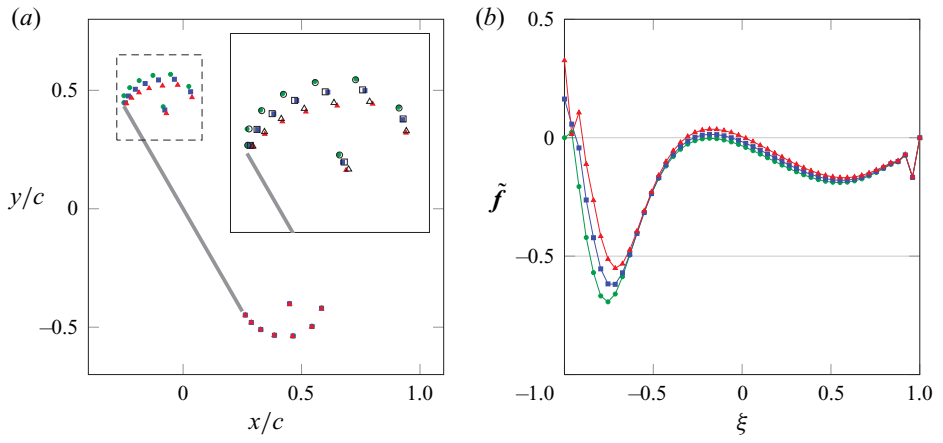


Figure 11. Effect of increasing $\sigma_{LE}^{max}/U_\infty$ from 0 (•, green) to 0.05 (■, blue) and 0.1 (▲, red) on (*a*) the positions of shedded point vortices and (*b*) the non-singular part of the associated discrete vortex sheet strength for a flat plate of length c at 60°, 0.15 convective times after impulsively starting a uniform flow U_∞. The inset enlarges the boxed area (— — —) and overlays the positions of the vortices (○, □ and △) obtained using the Biot–Savart method of Darakananda & Eldredge (2019). At the trailing edge, the Kutta condition is enforced. The simulation is performed with $\Delta x/c = 0.01$, $\Delta S/\Delta x = 2$ and $\Delta t U_\infty/c = 0.025$.

2.6.1. *Pressure*

Here, we devise a means of computing the pressure, both throughout the flow field, $\boldsymbol{p} \in \mathcal{C}$, as well as on the surface of a body. As we will show below, our immersed boundary treatment naturally gives rise to the jump in pressure across this surface, $\Delta\boldsymbol{p} = \boldsymbol{p}^+ - \boldsymbol{p}^- \in \mathcal{S}^{N_s}$ (with + in the direction of the surface normal, $\boldsymbol{n} \in \mathcal{V}^{N_s}$). Thus, to distinguish the pressures on either side of the surface from one another, we use the fact that the interpolation of $\boldsymbol{p}$ onto the surface produces the average of surface values, ${}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{C}}\boldsymbol{p} = (\boldsymbol{p}^+ + \boldsymbol{p}^-)/2 \in \mathcal{S}$, where ${}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{C}}$ interpolates data from cell centres to the surface. It is thus easy to see that the pressure on either side is

$$\boldsymbol{p}^{\pm} = {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{C}}\boldsymbol{p} \pm \tfrac{1}{2}\Delta\boldsymbol{p}. \tag{2.77}$$

Thus, we seek $\Delta\boldsymbol{p}$ and $\boldsymbol{p}$. It should not be a surprise that our starting point for these quantities is the Euler equations. However, our approach exploits the fact that the methodology we have presented thus far already solves the Euler equations in the fluid – in vorticity form, via transport of vortex elements – and provides us with the instantaneous velocity field $\boldsymbol{v}$ and strength of the bound vortex sheet $\boldsymbol{f}$ on any bodies. This approach, which satisfies the incompressibility constraint by expressing velocity in the null space of the divergence operator (i.e. as curl of a streamfunction), obviates the need for computing pressure, the Lagrange multiplier for this constraint. However, now that we seek pressure, we use the Euler equations in their velocity form to provide it.

It is important to note that the immersed boundary treatment enriches the Euler equations' standard form with surface terms (Eldredge 2022). Written in their spatially discrete form, these immersed boundary Euler equations are

$$\rho\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} + \rho(\boldsymbol{w} + {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f}) \times \boldsymbol{v} = -\boldsymbol{G}\left(\boldsymbol{p} + \frac{1}{2}\rho|\boldsymbol{v}|^2\right) + {}^{\mathcal{F}}\boldsymbol{R}_{\mathcal{V}}[\boldsymbol{n} \circ \Delta\boldsymbol{p} + \rho\Delta\boldsymbol{v}({}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}\boldsymbol{v} - \boldsymbol{v}_b) \cdot \boldsymbol{n}], \tag{2.78}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

where ρ is the fluid density and ${}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}$ interpolates grid data from $\mathcal{F}$ to the space of vector-valued surface data $\mathcal{V}^{N_s}$. Note the appearance of a few terms that are not typically seen in the Euler equations. First, we note the final bracketed pair of surface terms on the right-hand side, containing the jumps in pressure and velocity ($\Delta\boldsymbol{v}$) across the surface, as well as the difference between the normal components of the fluid velocity (interpolated onto the surface) and the surface velocity itself, $\boldsymbol{v}_b \in \mathcal{V}^{N_s}$. (This difference is zero by virtue of the no-flow-through condition, but we keep it here since it combines with other terms in later manipulations.) These surface terms are immersed into the grid by the operator ${}^{\mathcal{F}}\boldsymbol{R}_{\mathcal{V}}$. In fact, if we had chosen to solve the Euler equations in velocity – instead of by streamfunction–vorticity – form, then we would have used the pressure jump in this term as a Lagrange multiplier for enforcing the no-flow-through condition on $\boldsymbol{v}$. The second new term is the bound vortex sheet strength, appearing alongside the fluid vorticity on the left-hand side. This term emerges because the curl of the velocity field in the immersed boundary method generates both of these: $\boldsymbol{C}^T\boldsymbol{v} = \boldsymbol{w} + {}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f}$, where $\boldsymbol{C}^T : \mathcal{F} \mapsto \mathcal{N}$. (Our notation for this operator is consistent with that of previous authors, such as Colonius & Taira 2008.)

These quantities, $\Delta\boldsymbol{p}$ and $\boldsymbol{p}$, can be solved for simultaneously from the Schur complement system that arises from solving the Euler equations (2.78) and the associated constraints of divergence-free velocity and no-flow-through condition. Indeed, the approach we outline here is the natural outcome of that system. However, rather than present a detailed derivation, we present the equations with an intuitive explanation.

First, we develop an equation for $\Delta\boldsymbol{p}$ by taking the discrete curl $\boldsymbol{C}^T$ of (2.78) to eliminate the gradient term – since $\boldsymbol{C}^T\boldsymbol{G}$ is identically zero – and obtain a vorticity form of the immersed boundary Euler equations. With some manipulation to account for the motion of the surface (embodied in the time variation of ${}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}$), these equations can be written as

$$\rho\left(\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} - \boldsymbol{C}^T(\boldsymbol{v}\times\boldsymbol{w})\right) + \rho\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}t} = \boldsymbol{C}^T\,{}^{\mathcal{F}}\boldsymbol{R}_{\mathcal{V}}\boldsymbol{n}\circ\left[\Delta\boldsymbol{p} + \rho({}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}\boldsymbol{v} - \boldsymbol{v}_b)\cdot(\boldsymbol{f}\times\boldsymbol{n})\right], \tag{2.79}$$

where $\mathrm{d}\boldsymbol{f}/\mathrm{d}t$ represents the time derivative of each element of $\boldsymbol{f}$ while following a point moving with velocity $\boldsymbol{v}_b$. We assume that each immersed point moves with this local surface velocity, so this is simply the time derivative of the vector $\boldsymbol{f}$. We have written the equation in this intermediate form on purpose in order to make a few key points. First, it is important to note that we have already satisfied the vorticity equation in the fluid – the first terms in parentheses on the left-hand side – by advecting the point vortices. Thus, we can set these terms to zero, leaving only those terms associated with the surface. Aside from $\Delta\boldsymbol{p}$, these remaining terms involve only known quantities, and we could solve them in the current form for $\Delta\boldsymbol{p}$ using similar techniques to the ones we will describe below. However, we will first write the equation in a more familiar form, and define some helpful operators and quantities to enable this.

In (2.79) we see a composite of the curl of the regularization operator; let us write this more compactly as $\boldsymbol{C}_{\mathcal{S}}^T : \mathcal{S}^{N_s} \mapsto \mathcal{N}$,

$$\boldsymbol{C}_{\mathcal{S}}^T\boldsymbol{\sigma} = \boldsymbol{C}^T\,{}^{\mathcal{F}}\boldsymbol{R}_{\mathcal{V}}\,(\boldsymbol{n}\circ\boldsymbol{\sigma})\,, \tag{2.80}$$

for some surface scalar data, $\boldsymbol{\sigma} \in \mathcal{S}^{N_s}$. We refer to this as a surface curl operator. Its transpose, $\boldsymbol{C}_{\mathcal{S}} : \mathcal{N} \mapsto \mathcal{S}^{N_s}$, also arises in what follows, and can also be described as a

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

surface curl operator. It is defined as

$$\boldsymbol{C}_{\mathcal{S}}\boldsymbol{s} = \boldsymbol{n} \cdot {}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}\boldsymbol{C}\boldsymbol{s}, \tag{2.81}$$

for $\boldsymbol{s} \in \mathcal{N}$. The operator $\boldsymbol{C}_{\mathcal{S}}$ obtains the normal component of velocity on an immersed surface for a given streamfunction $\boldsymbol{s}$. Before we explain the role of its transpose, it is useful to remember that any potential flow generated by (or about) an impenetrable surface can be equivalently described by either a distribution of vortices (a vortex sheet, with strength $\boldsymbol{f}$) or a distribution of dipoles (a double layer) on the surface. In the latter case, the strength of the double layer is given by the negative of the jump in scalar potential across the surface, $-\boldsymbol{\Delta\phi} \in \mathcal{S}^{N_s}$. In fact, the two distributions' strengths can be related to each other, either by using Stokes' theorem or by the properties of the generalized functions that underpin the immersed boundary method (Eldredge 2022), leading to

$$^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f} = -\boldsymbol{C}_{\mathcal{S}}^{T}\boldsymbol{\Delta\phi}. \tag{2.82}$$

Thus, $\boldsymbol{C}_{\mathcal{S}}^{T}$ produces the equivalent bound vorticity distribution (immersed into the grid) associated with a given jump in scalar potential on the surface. To calculate $\boldsymbol{\Delta\phi}$ in terms of $\boldsymbol{f}$, we apply $\boldsymbol{C}_{\mathcal{S}}\boldsymbol{L}^{-1}$ to both sides of (2.82) to equate the normal velocity induced on the surface by each distribution. The composite operator $-\boldsymbol{C}_{\mathcal{S}}\boldsymbol{L}^{-1}\boldsymbol{C}_{\mathcal{S}}^{T}$ is positive semi-definite, and its null space can be shown to consist only of uniform values on the surface (i.e. the null space has an equivalent bound vorticity equal to zero). Thus, the jump in scalar potential associated with a vortex sheet of strength $\boldsymbol{f}$ is

$$\boldsymbol{\Delta\phi} = -(\boldsymbol{C}_{\mathcal{S}}\boldsymbol{L}^{-1}\boldsymbol{C}_{\mathcal{S}}^{T})^{-1}\boldsymbol{C}_{\mathcal{S}}\boldsymbol{L}^{-1}\,{}^{\mathcal{N}}\boldsymbol{R}_{\mathcal{S}}\boldsymbol{f}, \tag{2.83}$$

to which we can add any constant value without affecting the result.

Now, armed with this insight, we can return to (2.79), and rewrite the surface terms on the left-hand side in terms of $\boldsymbol{\Delta\phi}$

$$\boldsymbol{C}_{\mathcal{S}}^{T}\left[\Delta\boldsymbol{p} + \rho({}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}\boldsymbol{v} - \boldsymbol{v}_b)\cdot(\boldsymbol{f}\times\boldsymbol{n}) + \rho\frac{\mathrm{d}\boldsymbol{\Delta\phi}}{\mathrm{d}t}\right] = 0, \tag{2.84}$$

where $\mathrm{d}\boldsymbol{\Delta\phi}/\mathrm{d}t$ denotes the time derivative of $\boldsymbol{\Delta\phi}$ associated with a particular immersed point, assumed to be moving with velocity $\boldsymbol{v}_b$. This equation implies that the expression in brackets must be equal to a uniform value, which we can take to be zero without loss of generality. We can immediately write an immersed boundary Bernoulli equation

$$\Delta\boldsymbol{p} + \rho({}^{\mathcal{V}}\boldsymbol{E}_{\mathcal{F}}\boldsymbol{v} - \boldsymbol{v}_b)\cdot(\boldsymbol{f}\times\boldsymbol{n}) + \rho\frac{\mathrm{d}\boldsymbol{\Delta\phi}}{\mathrm{d}t} = 0. \tag{2.85}$$

This equation is the discrete equivalent of a continuous version that appears in previous works, e.g. Jones (2003) and Eldredge (2019). At each time step, we use (2.84) to compute the instantaneous jump in scalar potential associated with the vortex sheet $\boldsymbol{f}$, and then use (2.85) to find $\Delta\boldsymbol{p}$.

Now we can substitute $\Delta\boldsymbol{p}$ from (2.85) into the Euler equations (2.78) and operate on these equations with the discrete divergence operator, so that $\boldsymbol{L} = \boldsymbol{DG}$ acts on $\boldsymbol{p}$. We solve the resulting equation, obtaining

$$\boldsymbol{p} = -\frac{1}{2}\rho|\boldsymbol{v}|^2 - \rho\boldsymbol{L}^{-1}\left[\boldsymbol{D}(\boldsymbol{w}\times\boldsymbol{v}) + {}^{\mathcal{F}}\boldsymbol{R}_{\mathcal{V}}\left(\boldsymbol{n}\circ\frac{\mathrm{d}\boldsymbol{\Delta\phi}}{\mathrm{d}t} - \boldsymbol{v}_b\times\boldsymbol{f}\right)\right]. \tag{2.86}$$

It should be noted that any uniform value can be arbitrarily added to this expression. Also, we note in passing that the final set of terms (with the inverse Laplacian acting on the

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

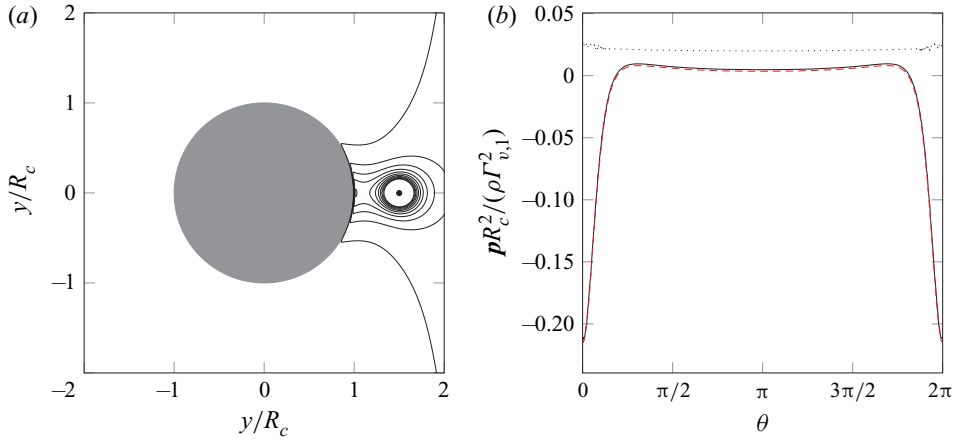


Figure 12. (*a*) Contours (——) of the discrete pressure for a point vortex (•) with strength $\Gamma_{v,1}$ at $(R_v, 0)$ near a cylinder consisting of N_s points with radius R_c and a bound circulation $-\Gamma_{v,1}$. (*b*) The scaled discrete pressure at the exterior (——) and interior (· · · · · ·) of the cylinder. Overlaid is the exact continuous solution (- - - -, red dashed) for the exterior pressure. The simulation is performed with $R_v/R_c = 3/2$, $\Delta x/R_c = 0.05$, $\Delta S/\Delta x = 1.4$ and $\Delta t\Gamma_{v,1}/R_c^2 = 0.01$.

term in brackets) is equivalent to the time derivative of the scalar potential field, rendering the overall equation equivalent to a Bernoulli equation on the grid. However, there is no particular advantage in writing the equation in that form. In the current form, the first term in brackets (the divergence of the Lamb vector) represents the direct force exerted on the fluid by the fluid vorticity. The remaining terms in brackets collectively constitute the effects of surface motion and of the surface's modification of the flow induced by fluid elements (e.g. vorticity, free stream).

We now demonstrate the pressure calculations with two examples that have an analytical solution for the pressure distribution on the surface. Figures 12 and 13 depict the pressure field and surface distribution for a vortex near a cylinder and a plate, respectively. The pressure inside the cylinder is close to the exact solution of a uniform value, except for some noise at the side near the vortex, which is visible on the interior surface distribution of the pressure. The exterior surface pressure distribution on the cylinder shows good agreement with the analytical solution. In the example of the flat plate, shown in figure 13, the bottom and top surface pressure distributions again show good agreement with the analytical solution except for at the edges, which is expected. Note that, in these examples, we used a smaller value for $\Delta S/\Delta x$ than before, which was necessary to prevent the low pressure from the vortex from leaking through the surface. As a result, high-frequency components of the surface pressure are incorrectly amplified through the regularization and interpolation operators (Goza *et al.* 2016), especially at the edges.

2.6.2. *Impulse-based calculations of force and moment*

The integral of the pressure distribution over the surface (plus any edge-suction parameters in the case of sharp edges) will be equal to the force on the surface. However, in this section we provide an alternative means of calculating the force and moment on the body through the negative rate of change of impulse in the fluid. The continuous expressions for linear and angular impulse (about the origin) are, in two dimensions (Saffman 1993;

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

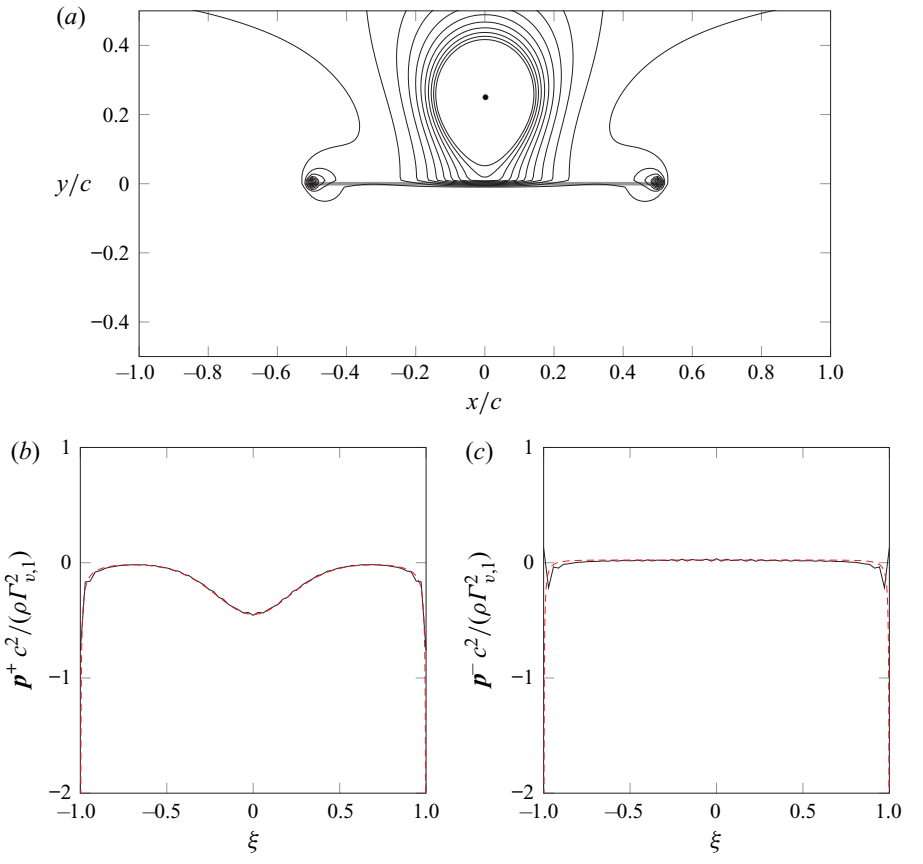


Figure 13. (*a*) Contours (——) of the discrete pressure for a point vortex (•) with strength $\Gamma_{v,1}$ at $(0, R_v)$ near a flat plate consisting of N_s points with chord length c and a bound circulation $-\Gamma_{v,1}$. The scaled discrete pressure (——) at (*b*) the top side and (*c*) the bottom side of the plate. Overlaid is the exact continuous solution (- - - -, red dashed). The simulation is performed with $R_v/c = 0.25$, $\Delta x/c = 0.01$, $\Delta S/\Delta x = 1.4$ and $\Delta t\Gamma_{v,1}/c^2 = 0.005$.

Eldredge 2019)

$$\boldsymbol{P} = \int_{\Omega} \boldsymbol{x} \times \boldsymbol{\omega}\, \mathrm{d}V + \int_{\mathcal{S}} \boldsymbol{x} \times (\boldsymbol{n} \times \boldsymbol{u})\, \mathrm{d}S, \tag{2.87}$$

$$\boldsymbol{\Pi}_O = \tfrac{1}{2} \int_{\Omega} \boldsymbol{x} \times (\boldsymbol{x} \times \boldsymbol{\omega})\, \mathrm{d}V + \tfrac{1}{2} \int_{\mathcal{S}} \boldsymbol{x} \times [\boldsymbol{x} \times (\boldsymbol{n} \times \boldsymbol{u})]\, \mathrm{d}S. \tag{2.88}$$

If there is only a single body, then the force and moment (about the origin) exerted by the fluid on that body are given by

$$\boldsymbol{F} = -\rho \frac{\mathrm{d}\boldsymbol{P}}{\mathrm{d}t}, \quad \boldsymbol{M}_O = -\rho \frac{\mathrm{d}\boldsymbol{\Pi}_O}{\mathrm{d}t}, \tag{2.89a,b}$$

where ρ is the fluid density. In the two-dimensional applications of this paper, the angular impulse and the moment have only a single component, e.g. $\boldsymbol{\Pi}_{\mathrm{O}} = \Pi_O \boldsymbol{e}_z$, where $\boldsymbol{e}_z$ is the unit vector out of the plane.

It should be observed that, by definition, the bound vortex sheet strength γ is equal to the jump in tangential velocity between the fluid and the surface, $\boldsymbol{n} \times \boldsymbol{u} = \gamma \boldsymbol{e}_z + \boldsymbol{n} \times \boldsymbol{u}_b$, where $\boldsymbol{n}$ is the unit surface normal vector directed into the fluid, $\boldsymbol{u}$ is the fluid velocity, and

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

$\boldsymbol{u}_b$ is the velocity of the surface. Thus, the surface integrals in (2.87) and (2.88) can be re-written in terms of the vortex sheet strength and the body motion.

We can easily develop discrete forms of the integrals (2.87) and (2.88) with the solutions and notation described in this paper. For the volume integrals, let us denote diagonal matrices containing the coordinates of the grid nodes by $\boldsymbol{D}_{\boldsymbol{x}_\mathcal{N}}$ and $\boldsymbol{D}_{\boldsymbol{y}_\mathcal{N}}$. Thus, the expressions in (2.87) and (2.88) can be written in discrete form as

$$P_x = \langle \boldsymbol{y}_\mathcal{N}, \boldsymbol{w} \rangle_\mathcal{N} + \langle \boldsymbol{r}_y, \boldsymbol{f} + \boldsymbol{D}_{\boldsymbol{n}_x} \boldsymbol{v}_{b,y} - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{v}_{b,x} \rangle_{\mathcal{S}^{N_s}}, \tag{2.90}$$

$$P_y = -\langle \boldsymbol{x}_\mathcal{N}, \boldsymbol{w} \rangle_\mathcal{N} - \langle \boldsymbol{r}_x, \boldsymbol{f} + \boldsymbol{D}_{\boldsymbol{n}_x} \boldsymbol{v}_{b,y} - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{v}_{b,x} \rangle_{\mathcal{S}^{N_s}}, \tag{2.91}$$

and

$$\Pi_O = -\tfrac{1}{2} \langle \boldsymbol{D}_{\boldsymbol{x}_\mathcal{N}} \boldsymbol{x}_\mathcal{N} + \boldsymbol{D}_{\boldsymbol{y}_\mathcal{N}} \boldsymbol{y}_\mathcal{N}, \boldsymbol{w} \rangle_\mathcal{N} - \tfrac{1}{2} \langle \boldsymbol{D}_{\boldsymbol{r}_x} \boldsymbol{r}_x + \boldsymbol{D}_{\boldsymbol{r}_y} \boldsymbol{r}_y, \boldsymbol{f} + \boldsymbol{D}_{\boldsymbol{n}_x} \boldsymbol{v}_{b,y} - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{v}_{b,x} \rangle_{\mathcal{S}^{N_s}}. \tag{2.92}$$

The overall force and moment exerted on the body are obtained from calculating these impulses and computing their rates of change in (2.89*a*,*b*). Part of this force and moment is attributable to the dynamics of vorticity in the fluid. The remaining part is due to surface motion relative to the fluid, and we will discuss this in the next section.

To illustrate the accuracy of the impulse-based calculation of force, we apply the method to two examples. In the first example, we simulate the trajectories of two point vortices of opposite strength, in which case each vortex is convected past a cylinder due to the presence of the other vortex. The time stepping in this example is carried out using a fourth-order Runge–Kutta scheme. Figure 14 shows the trajectories and the x component of the impulse together with their exact solutions, which show good agreement with the exact solution. In the second example, we compare our simulation of the first instants of the unsteady, fully separated flow around a flat plate after impulsively starting a uniform flow with the Biot–Savart method from Darakananda & Eldredge (2019), using the same positioning rules to insert point vortices and the same time step. The vortex positions and the corresponding impulse and lift are compared in figure 15 and show good agreement as well.

2.6.3. *Added mass*

The added mass tensor provides a measure of the inertial influence of the fluid on the body in response to changes in the body's translational or rotational motion. The coefficients of the added mass tensor of a body are obtained by computing the impulse components associated with a unit-valued component of motion. The motion's influence is both direct, via the surface velocity, and indirect, in the bound vortex sheet that develops on the surface.

For example, suppose that we consider translation at unit velocity in the x direction, for which the motion is described by $\boldsymbol{v}_{b,x} = \boldsymbol{1}_\mathcal{S}$, $\boldsymbol{v}_{b,y} = 0$ and $\boldsymbol{s}_b' = \boldsymbol{r}_y$, and the associated bound vortex sheet – obtained without the Kutta condition by solving the basic problem (2.25) – is $\boldsymbol{f} = \boldsymbol{S}^{-1} \boldsymbol{P}^\Gamma \boldsymbol{s}_b' = \boldsymbol{S}^{-1} \boldsymbol{P}^\Gamma \boldsymbol{r}_y$. The added mass coefficients corresponding to this motion are derived by substituting these into the impulse formulas (2.90)–(2.92)

$$P_x^{(x)} = \langle \boldsymbol{r}_y, \boldsymbol{S}^{-1} \boldsymbol{P}^\Gamma \boldsymbol{r}_y - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{1}_\mathcal{S} \rangle_{\mathcal{S}^{N_s}}, \tag{2.93}$$

$$P_y^{(x)} = -\langle \boldsymbol{r}_x, \boldsymbol{S}^{-1} \boldsymbol{P}^\Gamma \boldsymbol{r}_y - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{1}_\mathcal{S} \rangle_{\mathcal{S}^{N_s}}, \tag{2.94}$$

$$\Pi_O^{(x)} = -\tfrac{1}{2} \langle \boldsymbol{D}_{\boldsymbol{r}_x} \boldsymbol{r}_x + \boldsymbol{D}_{\boldsymbol{r}_y} \boldsymbol{r}_y, \boldsymbol{S}^{-1} \boldsymbol{P}^\Gamma \boldsymbol{r}_y - \boldsymbol{D}_{\boldsymbol{n}_y} \boldsymbol{1}_\mathcal{S} \rangle_{\mathcal{S}^{N_s}}. \tag{2.95}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

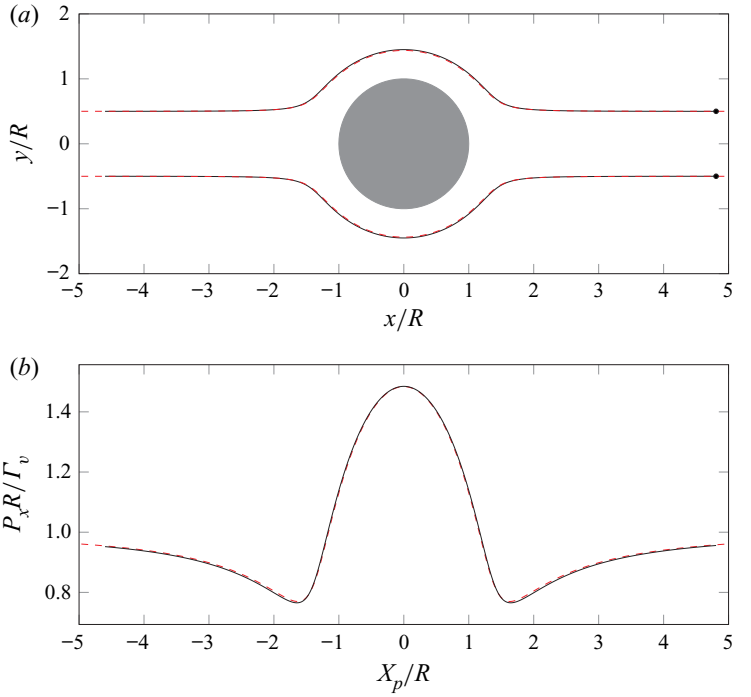


Figure 14. (*a*) Numerically simulated trajectories (——) of two point vortices (•) of opposite strengths $\Gamma_{v,1}$ and $\Gamma_{v,1} = -\Gamma_{v,}$[2] being convected past a circular cylinder with radius R, and (*b*) the x component of the associated, numerically simulated impulse (——) in the fluid. Overlaid are the exact continuous trajectories and impulse (- - - -, red dashed). The simulation is performed with $\Delta x/R = 0.04$, $\Delta S/\Delta x = 2$ and $\Delta t\Gamma_{v,1}/R^2 = 0.1$.

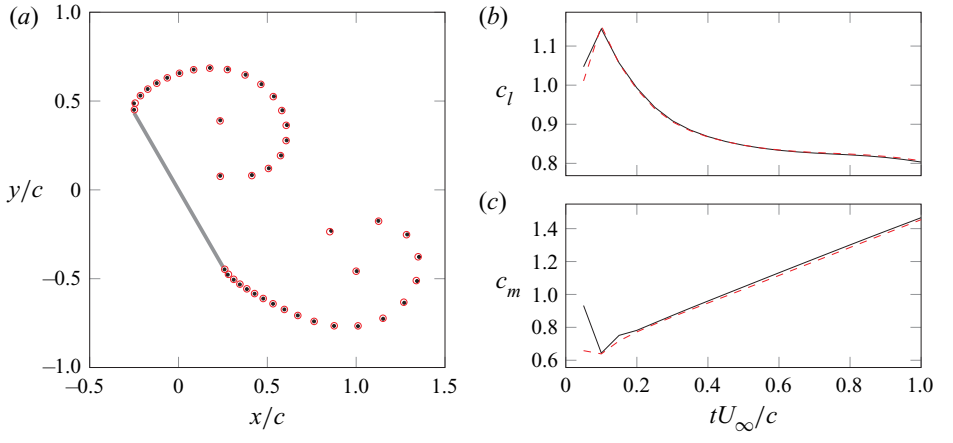


Figure 15. Comparison of the simulated vortex shedding behind a flat plate of length c at 60° in a uniform flow using the method in this paper (• and ——) and using the Biot–Savart method of Darakananda & Eldredge (2019) (○, red open circle and - - - -, red dashed). (*a*) The positions of the shedded point vortices, (*b*) the lift coefficient and (*c*) the moment coefficient, one convective time after impulsively starting the uniform flow. At both edges, the Kutta condition is enforced. The simulation is performed with $\Delta x/c = 0.01$, $\Delta S/\Delta x = 2$ and $\Delta tU_\infty/c = 0.05$.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

Thus, the components of the added mass coefficients tensor associated with translation in the x direction are

$$m^F_{xx} = \rho P^{(x)}_x, \tag{2.96}$$

$$m^F_{xy} = \rho P^{(x)}_y, \tag{2.97}$$

$$m^M_x = \rho \left(\Pi^{(x)}_O - \boldsymbol{X}_c \times \boldsymbol{P}^{(x)} \right), \tag{2.98}$$

where $\boldsymbol{X}_c$ is the centroid of the body, which can be calculated using (B8), and the superscripts F and M are used to denote the coefficient for the force and moment, respectively.

A similar approach can be used to obtain the added mass coefficients due to unit translation in the y direction, for which $\boldsymbol{v}_{b,x} = 0$, $\boldsymbol{v}_{b,y} = \boldsymbol{1}_{\mathcal{S}}$ and $\boldsymbol{s}'_b = -\boldsymbol{r}_x$. The coefficients due to unit rotation follow from taking $\boldsymbol{v}_{b,x} = -\boldsymbol{r}_y$, $\boldsymbol{v}_{b,y} = \boldsymbol{r}_x$ and $\boldsymbol{s}'_b = -\frac{1}{2}(\boldsymbol{D}_{\boldsymbol{r}_x}\boldsymbol{r}_x + \boldsymbol{D}_{\boldsymbol{r}_y}\boldsymbol{r}_y)$.

2.7. *Multiple bodies*

The previous sections provided the formulations for potential flow with the presence of a body. The extension of these expressions to multiple bodies is straightforward and consists of allocating partitions of $\boldsymbol{f}$ to the different bodies. The surface streamfunction has to be partitioned accordingly, with the body motion streamfunction $\boldsymbol{s}_b$ containing the values for the discrete surface points from all the bodies and $\boldsymbol{s}_0 = \sum_j \boldsymbol{1}_{\mathcal{S}j} s_{0,j}$ allocating a uniform value $s_{0,j}$ to the jth body, where $\boldsymbol{1}_{\mathcal{S}j} \in \mathcal{S}^{N_s}$ is a vector whose ith component is one if it belongs to the jth body and zero otherwise. The system (2.25) can then be solved for the streamfunction field without modification.

As in the single-body case, if we want to enforce an edge condition on the jth body, we treat its uniform streamfunction value $s_{0,j}$ as a Lagrange multiplier and add a constraint on $\boldsymbol{f}$ to the saddle-point system. In the unsteady case, we add a circulation constraint for each body and consider each new point vortex to be released from a specified body. For example, let us consider two bodies, with each body having one sharp edge. We assume $\Gamma_{v,1}$ and $\Gamma_{v,2}$ are the strengths from the vortices that were released from the first and second body, respectively. We can compute these strengths by enforcing the Kutta condition for both bodies using the saddle-point system

$$\begin{bmatrix} \boldsymbol{L} & {}^{\mathcal{N}}\tilde{\boldsymbol{R}}_{\mathcal{S}} & 0 & 0 & \boldsymbol{d}_1 & \boldsymbol{d}_2 \\ {}^{\mathcal{S}}\boldsymbol{E}_{\mathcal{N}} & 0 & \boldsymbol{1}_{\mathcal{S}1} & \boldsymbol{1}_{\mathcal{S}2} & 0 & 0 \\ 0 & \boldsymbol{e}^T_{k_1} & 0 & 0 & 0 & 0 \\ 0 & \boldsymbol{e}^T_{k_2} & 0 & 0 & 0 & 0 \\ 0 & \hat{\boldsymbol{f}}^T_{0,1} & 0 & 0 & 1 & 0 \\ 0 & \hat{\boldsymbol{f}}^T_{0,2} & 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} \boldsymbol{s} \\ \tilde{\boldsymbol{f}} \\ s_{0,1} \\ s_{0,2} \\ \Gamma_{v,1} \\ \Gamma_{v,2} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{w} \\ \boldsymbol{P}^{\Gamma}\boldsymbol{s}'_b \\ 0 \\ 0 \\ -\Gamma_{\boldsymbol{w},1} \\ -\Gamma_{\boldsymbol{w},2} \end{pmatrix}, \tag{2.99}$$

where we defined $\Gamma_{\boldsymbol{w},j}$ as the circulation of the existing vorticity in the flow that has previously been released from the jth body and $\boldsymbol{f}_{0,j} = \boldsymbol{D}_{\boldsymbol{f}_0}\boldsymbol{1}_{\mathcal{S}j}$. Again, these solutions are easily extensible to an arbitrary number of edges per body. For example, figure 16 demonstrates the method for two flat plates in a uniform flow where the leading edge and trailing edge are regularized for both plates by releasing four point vortices during each time step.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

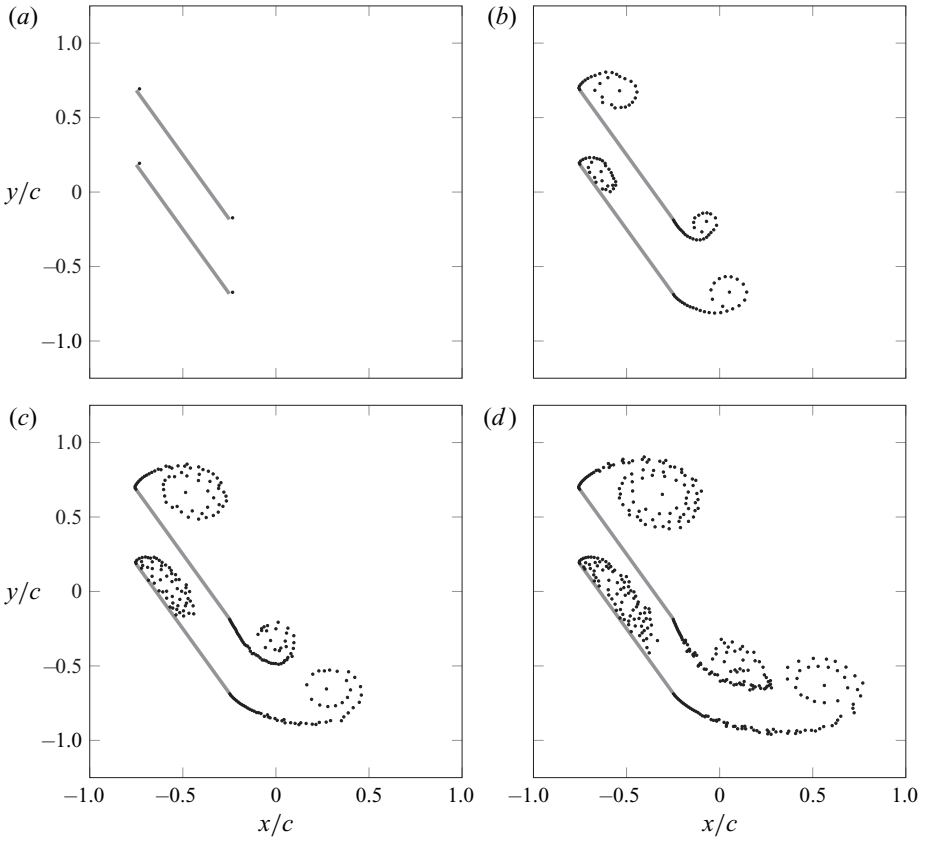


Figure 16. Evolution of the flow around two flat plates of length c at 60°, vertically separated by a distance of $c/2$, at tU_∞/c equal to (*a*) 0, (*b*) 0.3, (*c*) 0.6 and (*d*) 0.9 after impulsively starting a uniform flow. The simulation is performed with $\Delta x/c = 0.01$, $\Delta S/\Delta x = 2$ and $\Delta t U_\infty/c = 0.01$.

The formulas for pressure, force and added mass also generalize to systems with multiple bodies. For example, figure 17 demonstrates an example of a potential flow model with an array of nine circular cylinders and compares the ratio of the largest eigenvalue of added mass coefficient tensor and the largest self-added mass coefficient of the system with the results of Chen (1975).

3. Conclusion

A treatment of potential flow on Cartesian grids was presented. The main body of this work is based on the computation of the discrete streamfunction through the streamfunction–vorticity Poisson equation with singular vortex elements as vorticity sources. The potential flow in the presence of sinks and sources requires the computation of the discrete scalar potential, which is completely analogous except for the scalar potential Neumann boundary condition instead of the streamfunction Dirichlet boundary condition to enforce no penetration on surfaces in the flow. The Helmholtz decomposition then shows how the velocity fields associated with the scalar potential and streamfunction can be superposed to obtain the combined flow due to sources, sinks and vortices.

In our potential flow treatment, we used two algebraic techniques that allowed us to mimic the analytical treatment of potential flows around sharp-edged bodies with

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

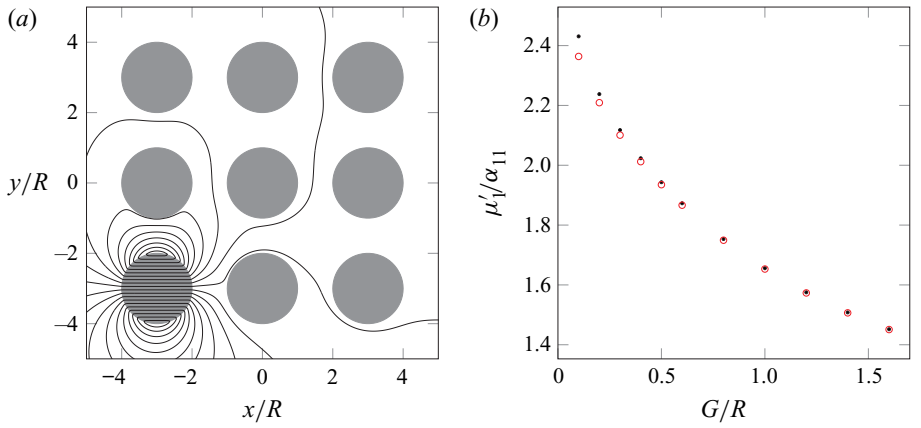


Figure 17. (*a*) Contours of the discrete streamfunction for an array of nine circular cylinders with radius R, spaced with a gap distance G between each cylinder, of which the bottom left cylinder translates horizontally. (*b*) Numerically simulated variation (•) of the ratio between the largest eigenvalue of the added mass coefficient tensor and the largest self-added mass coefficient with the gap-to-radius ratio G/R. Overlaid are the values (○, red) obtained by Chen (1975) from solving a system of truncated analytical expressions. The simulations are performed with $\Delta x/R = 0.05$ and $\Delta S/\Delta x = 2$.

bound vortex sheets. The first technique is to account for surfaces in the flow by using the immersed boundary projection method. We introduce a Lagrange multiplier for the no-penetration constraint in the streamfunction–vorticity Poisson equation and identify it as a discrete version of the continuous strength distribution of a bound vortex sheet on the body. The discrete equations that solve the associated saddle-point system are then completely analogous to the continuous boundary integral equations. It should be noted that this underlying continuous problem is a Fredholm integral equation of the first kind and is ill posed, leading to poorly conditioned discrete operators in the solution method. We have not attempted to address this issue here, but the work of Goza *et al.* (2016) did and its methods are straightforward to apply. The second algebraic technique is to decompose the discrete bound vortex sheet strength for sharp-edged bodies into a singular and non-singular part. One can then add constraints on the elements of the non-singular part that are located at the edges to make the recomposed bound vortex sheet well behaved. This way, we enforced the Kutta condition in a way that is similar to analytical treatments of the Kutta condition and that allows for generalized edge conditions as well. Furthermore, we leveraged the concept of the discrete bound vortex sheet strength to create expressions for the pressure in the flow and on surfaces, the impulse in the flow around surfaces and the added mass matrix for a system of arbitrarily shaped bodies. Finally, it is important to note that the treatment of potential flow that we presented is not restricted to the specific finite-difference discretization tools that we used in our implementation to provide the examples in this work. Also, one is not restricted to use the immersed boundary projection method to obtain an expression for the discrete vortex sheet strength and can use, for example, the immersed interface method instead, as in Gillis *et al.* (2019).

We found that the method can accurately replicate the results of a Biot–Savart method for the unsteady flow around a flat plate. This motivates the goal of implementing a three-dimensional version of this method using vortex particles (with a vector representing their strengths) or filaments (Cottet & Koumoutsakos 2000), since the concepts of the immersed boundary projection method and the enforcement of edge conditions through the multiplicative decomposition of the vortex sheet strength generalize to three dimensions.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

Such a three-dimensional, grid-based solver would rely on the vector potential and a vector treatment of the vorticity field, while properly accounting for vortex stretching. It could potentially make significant cost improvements over unsteady three-dimensional panel methods, which generally scale poorly as the number of panels and vortex elements in the flow increase.

Funding. The support for this work by the US Air Force Office of Scientific Research (grant number FA9550-18-1-0440) with programme manager G. Abate is gratefully acknowledged.

Declaration of interests. The authors report no conflict of interest.

Data availability. All the computational results can be reproduced with the code available at: https://github.com/JuliaIBPM/GridPotentialFlow.jl.

Author ORCIDs.
Diederik Beckers https://orcid.org/0000-0002-0242-1525;
Jeff D. Eldredge https://orcid.org/0000-0002-2672-706X.

Appendix A. Solution of general saddle-point systems

A general block system (with positive semi-definite matrix $\mathcal{A}$) can be decomposed as follows:

$$\begin{bmatrix} \mathcal{A} & \mathcal{B}_1^T \\ \mathcal{B}_2 & -\mathcal{C} \end{bmatrix} = \begin{bmatrix} \mathcal{A} & 0 \\ \mathcal{B}_2 & \mathcal{S} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathcal{A}^{-1}\mathcal{B}_1^T \\ 0 & \boldsymbol{I} \end{bmatrix}, \tag{A1}$$

where

$$\mathcal{S} \equiv -\mathcal{C} - \mathcal{B}_2\mathcal{A}^{-1}\mathcal{B}_1^T \tag{A2}$$

is the Schur complement of the matrix system and $\boldsymbol{I}$ is the identity. By this decomposition, we can develop an algorithm for the solution of the block system

$$\begin{bmatrix} \mathcal{A} & \mathcal{B}_1^T \\ \mathcal{B}_2 & -\mathcal{C} \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \boldsymbol{r_1} \\ \boldsymbol{r_2} \end{pmatrix}. \tag{A3}$$

We will refer to $\boldsymbol{x}$ as the solution vector and $\boldsymbol{y}$ as the constraint force. We define the intermediate solution vector $(\boldsymbol{x}^*, \boldsymbol{y}^*)^{\mathrm{T}}$ as the solution of the lower-triangular system

$$\begin{bmatrix} \mathcal{A} & 0 \\ \mathcal{B}_2 & \mathcal{S} \end{bmatrix} \begin{pmatrix} \boldsymbol{x}^* \\ \boldsymbol{y}^* \end{pmatrix} = \begin{pmatrix} \boldsymbol{r_1} \\ \boldsymbol{r_2} \end{pmatrix} \tag{A4}$$

and then the solution we seek can be found by back substitution of

$$\begin{bmatrix} \boldsymbol{I} & \mathcal{A}^{-1}\mathcal{B}_1^T \\ 0 & \boldsymbol{I} \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}^* \\ \boldsymbol{y}^* \end{pmatrix}. \tag{A5}$$

The algorithm we derive from this is

$$\left.\begin{aligned} \mathcal{A}\boldsymbol{x}^* &= \boldsymbol{r_1}, \\ \mathcal{S}\boldsymbol{y}^* &= \boldsymbol{r_2} - \mathcal{B}_2\boldsymbol{x}^*, \\ \boldsymbol{y} &= \boldsymbol{y}^*, \\ \boldsymbol{x} &= \boldsymbol{x}^* - \mathcal{A}^{-1}\mathcal{B}_1^T\boldsymbol{y}. \end{aligned}\right\} \tag{A6}$$

It is also useful to have an inverse representation of the block matrix system

$$\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{bmatrix} \mathcal{A}^{-1} + \mathcal{A}^{-1}\mathcal{B}_1^T\mathcal{S}^{-1}\mathcal{B}_2\mathcal{A}^{-1} & -\mathcal{A}^{-1}\mathcal{B}_1^T\mathcal{S}^{-1} \\ -\mathcal{S}^{-1}\mathcal{B}_2\mathcal{A}^{-1} & \mathcal{S}^{-1} \end{bmatrix} \begin{pmatrix} \boldsymbol{r_1} \\ \boldsymbol{r_2} \end{pmatrix}. \tag{A7}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

Appendix B. Some geometric relations for discrete surfaces

Consider a closed surface $\mathcal{S}_b$ with unit normal $\boldsymbol{n}$. We will recall some basic geometric relations here, and then provide some discrete versions of these relations based on the set of points with coordinates $\boldsymbol{r}_x$, $\boldsymbol{r}_y$, normal components $\boldsymbol{n}_x$ and $\boldsymbol{n}_y$ (which, the reader will recall, contain the surface length or area of each segment or panel associated with the points).

The volume $\mathcal{V}_b$ of the region enclosed by $\mathcal{S}_b$ can be computed from the integral

$$\mathcal{V}_b = \frac{1}{n_d}\int_{\mathcal{S}_b} \boldsymbol{x}\cdot\boldsymbol{n}\,\mathrm{d}S, \tag{B1}$$

where n_d is the number of spatial dimensions (2 or 3). Using the notation above, the approximate form of this expression is

$$\mathcal{V}_b \approx \frac{1}{n_d}\sum_j \boldsymbol{r}_j^T\boldsymbol{n}_j, \tag{B2}$$

where the sum is taken over the n_d components.

An alternative formula for the volume is

$$\mathcal{V}_b\boldsymbol{e}_j = -\frac{1}{n_d-1}\int_{\mathcal{S}_b} \boldsymbol{x}\times(\boldsymbol{n}\times\boldsymbol{e}_j)\,\mathrm{d}S. \tag{B3}$$

The components of this integral can be written discretely as

$$\mathcal{V}_b\boldsymbol{e}_j \approx \frac{\boldsymbol{e}_j}{n_d-1}\sum_{k\neq j}\boldsymbol{r}_k^T\boldsymbol{n}_k. \tag{B4}$$

And finally, a third alternative is

$$\mathcal{V}_b\boldsymbol{I} = \int_{\mathcal{S}_b}\boldsymbol{x}\boldsymbol{n}\,\mathrm{d}S, \tag{B5}$$

where $\boldsymbol{I}$ is the identity. The discrete form of this is a diagonal matrix with $\boldsymbol{r}_x^T\boldsymbol{n}_x$, $\boldsymbol{r}_y^T\boldsymbol{n}_y$ and $\boldsymbol{r}_z^T\boldsymbol{n}_z$ along the diagonal.

Thus, we can conclude that the volume of the body is approximately

$$\mathcal{V}_b \approx \langle\boldsymbol{r}_x,\boldsymbol{n}_x\rangle_{\mathcal{S}^{N_s}} \approx \langle\boldsymbol{r}_y,\boldsymbol{n}_y\rangle_{\mathcal{S}^{N_s}} \approx \langle\boldsymbol{r}_z,\boldsymbol{n}_z\rangle_{\mathcal{S}^{N_s}}, \tag{B6}$$

or any average of some combination of these.

The centroid of the body can be derived from the equation

$$\boldsymbol{X}_c\mathcal{V}_b = \tfrac{1}{2}\int_{\mathcal{S}_b}\boldsymbol{x}\cdot\boldsymbol{x}\boldsymbol{n}\,\mathrm{d}S, \tag{B7}$$

or, in discrete form,

$$X_c \approx \frac{1}{2\mathcal{V}_b}\langle\boldsymbol{D}_{\boldsymbol{r}_x}\boldsymbol{r}_x+\boldsymbol{D}_{\boldsymbol{r}_y}\boldsymbol{r}_y,\boldsymbol{n}_x\rangle_{\mathcal{S}^{N_s}}, \quad Y_c \approx \frac{1}{2\mathcal{V}_b}\langle\boldsymbol{D}_{\boldsymbol{r}_x}\boldsymbol{r}_x+\boldsymbol{D}_{\boldsymbol{r}_y}\boldsymbol{r}_y,\boldsymbol{n}_y\rangle_{\mathcal{S}^{N_s}}. \tag{B8a,b}$$

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

References

Baker, G.R. 1979 The 'cloud in cell' technique applied to the roll up of vortex sheets. *J. Comput. Phys.* **31** (1), 76–95.

Beckers, D. & Eldredge, J.D. 2021 JuliaIBPM/GridPotentialFlow.jl v0.3.2. Available at: https://github.com/JuliaIBPM/GridPotentialFlow.jl/tree/v0.3.2.

Benzi, M., Golub, G.H. & Liesen, J. 2005 Numerical solution of saddle point problems. *Acta Numerica* **1**, 1–137.

Chatelain, P., Curioni, A., Bergdorf, M., Rossinelli, D., Andreoni, W. & Koumoutsakos, P. 2008 Billion vortex particle direct numerical simulations of aircraft wakes. *Comput. Meth. Appl. Mech. Engng* **197** (13–16), 1296–1304.

Chatelin, R. & Poncet, P. 2014 Hybrid grid-particle methods and penalization: a Sherman–Morrison–Woodbury approach to compute 3D viscous flows using FFT. *J. Comput. Phys.* **269**, 314–328.

Chen, S.S. 1975 Vibration of nuclear fuel bundles. *Nucl. Engng Des.* **35** (3), 399–422.

Chorin, A.J. & Bernard, P.S. 1973 Discretization of a vortex sheet, with an example of roll-up. *J. Comput. Phys.* **13** (3), 423–429.

Christiansen, J.P. 1973 Numerical simulation of hydrodynamics by the method of point vortices. *J. Comput. Phys.* **13** (3), 363–379.

Colonius, T. & Taira, K. 2008 A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Comput. Meth. Appl. Mech. Engng* **197** (25–28), 2131–2146.

Coquerelle, M. & Cottet, G.-H. 2008 A vortex level set method for the two-way coupling of an incompressible fluid with colliding rigid bodies. *J. Comput. Phys.* **227** (21), 9121–9137.

Cottet, G.-H. & Koumoutsakos, P. 2000 *Vortex Methods: Theory and Practice*. Cambridge University Press.

Cottet, G.-H. & Poncet, P. 2004 Advances in direct numerical simulations of 3D wall-bounded flows by vortex-in-cell methods. *J. Comput. Phys.* **193** (1), 136–158.

Couët, B., Buneman, O. & Leonard, A. 1981 Simulation of three-dimensional incompressible flows with a vortex-in-cell method. *J. Comput. Phys.* **39** (2), 305–328.

Cserti, J. 2000 Application of the lattice green's function for calculating the resistance of an infinite network of resistors. *Am. J. Phys.* **68**, 896–906.

Darakananda, D. & Eldredge, J.D. 2019 A versatile taxonomy of low-dimensional vortex models for unsteady aerodynamics. *J. Fluid Mech.* **858**, 917–948.

Ebiana, A.B. & Bartholomew, R.W. 1996 Design considerations for numerical filters used in vortex-in-cell algorithms. *Comput. Fluids* **25** (1), 61–75.

Eldredge, J.D. 2019 *Mathematical Modeling of Unsteady Inviscid Flows*, Interdisciplinary Applied Mathematics, vol. 50. Springer.

Eldredge, J.D. 2022 A method of immersed layers on Cartesian grids, with application to incompressible flows. *J. Comput. Phys.* **448**, 110716.

Gazzola, M., Chatelain, P., van Rees, W.M. & Koumoutsakos, P. 2011 Simulations of single and multiple swimmers with non-divergence free deforming geometries. *J. Comput. Phys.* **230** (19), 7093–7114.

Gillis, T., Marichal, Y., Winckelmans, G. & Chatelain, P. 2019 A 2D immersed interface vortex particle-mesh method. *J. Comput. Phys.* **394**, 700–718.

Gillis, T., Winckelmans, G. & Chatelain, P. 2017 An efficient iterative penalization method using recycled Krylov subspaces and its application to impulsively started flows. *J. Comput. Phys.* **347**, 490–505.

Gillis, T., Winckelmans, G. & Chatelain, P. 2018 Fast immersed interface Poisson solver for 3D unbounded problems around arbitrary geometries. *J. Comput. Phys.* **354**, 403–416.

Goza, A., Liska, S., Morley, B. & Colonius, T. 2016 Accurate computation of surface stresses and forces with immersed boundary methods. *J. Comput. Phys.* **321**, 860–873.

Hejlesen, M.M., Koumoutsakos, P., Leonard, A. & Walther, J.H. 2015 Iterative Brinkman penalization for remeshed vortex methods. *J. Comput. Phys.* **280**, 547–562.

Jones, M.A. 2003 The separated flow of an inviscid fluid around a moving flat plate. *J. Fluid Mech.* **496** (496), 405–441.

Katsura, S. & Inawashiro, S. 1971 Lattice Green's functions for the rectangular and the square lattices at arbitrary points. *J. Math. Phys. (Journal of Mathematical Physics)* **12**, 1622–1630.

LeVeque, R.J. & Li, Z. 1994 The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM. J. Numer. Anal.* **31** (4), 1019–1044.

Liska, S. & Colonius, T. 2014 A parallel fast multipole method for elliptic difference equations. *J. Comput. Phys.* **278**, 76–91.

Marichal, Y., Chatelain, P. & Winckelmans, G. 2014 An immersed interface solver for the 2-D unbounded Poisson equation and its application to potential flow. *Comput. Fluids* **96**, 76–86.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press

MENG, J.C.S. & THOMSON, J.A.L. 1978 Numerical studies of some nonlinear hydrodynamic problems by discrete vortex element methods. *J. Fluid Mech.* **84** (3), 433–453.

MONAGHAN, J.J. 1985 Extrapolating B splines for interpolation. *J. Comput. Phys.* **60** (2), 253–262.

PONCET, P. 2009 Analysis of an immersed boundary method for three-dimensional flows in vorticity formulation. *J. Comput. Phys.* **228** (19), 7268–7288.

RAMESH, K., GOPALARATHNAM, A., GRANLUND, K., OL, M.V. & EDWARDS, J.R. 2014 Discrete-vortex method with novel shedding criterion for unsteady aerofoil flows with intermittent leading-edge vortex shedding. *J. Fluid Mech.* **751**, 500–548.

RASMUSSEN, J.T., COTTET, G.-H. & WALTHER, J.H. 2011 A multiresolution remeshed vortex-in-cell algorithm using patches. *J. Comput. Phys.* **230** (17), 6742–6755.

ROSSINELLI, D., BERGDORF, M., COTTET, G.-H. & KOUMOUTSAKOS, P. 2010 GPU accelerated simulations of bluff body flows using vortex particle methods. *J. Comput. Phys.* **229** (9), 3316–3333.

SAFFMAN, P.G. 1993 *Vortex Dynamics*, Cambridge Monographs on Mechanics, vol. 87. Cambridge University Press.

SPIETZ, H.J., HEJLESEN, M.M. & WALTHER, J.H. 2017 Iterative Brinkman penalization for simulation of impulsively started flow past a sphere and a circular disc. *J. Comput. Phys.* **336**, 261–274.

WIEGMANN, A. & BUBE, K.P. 2000 The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions. *SIAM J. Numer. Anal.* **37** (3), 827–862.

YANG, X., ZHANG, X., LI, Z. & HE, G.W. 2009 A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations. *J. Comput. Phys.* **228** (20), 7821–7836.

https://doi.org/10.1017/jfm.2022.238 Published online by Cambridge University Press