Analysis of an immersed boundary method for three-dimensional flows in vorticity formulation

Philippe Poncet

Analysis of an immersed boundary method for three-dimensional flows in vorticity formulation

Philippe Poncet[*,†]

∗ *Toulouse University, INSA, GMM, 135 avenue de Rangueil, F-31077 Toulouse, France*

† *CNRS, Toulouse Institute of Mathematics, Team MIP, F-31077 Toulouse, France*

Abstract

This article presents numerical analysis and practical considerations for three-dimensional flow computation using an implicit immersed boundary method. The Euler equations, or half a step of the Navier-Stokes equations when using fractional step algorithms, are investigated in their vorticity formulation. The context of flow computation around an arbitrarily shaped body is especially investigated.

In conventional immersed boundary methods using vorticity, singular vortex are dispatched over the body surface. In the present study, one prefers using sources of potential velocity field, dispatched on the body, whose nature is not vorticity. Such a formulation is compatible to the Euler equations. In practice, these sources of potential flow produce a velocity through this surface, aiming in practice at cancelling a flow-through velocity.

This article focuses on the use of the source-to-flow-through linear application, its properties being the key points for fast convergence. Its self-adjointness, or lack thereof, conditioning and preconditioning aspects are investigated. It follows that computing a velocity field with no-flow-through conditions in complex geometry, when using the source-to-flow-through linear application, can be achieved for $4/3$ of the computational cost of standard Poisson equation in a Cartesian box.

The robustness of immersed boundaries is especially interesting when used together with Vortex-in-Cell methods, well known for their robustness in time and their ability to compute accurately convective effects. A few examples, based on real-world geometries, illustrate the method capabilities.

Key words: Immersed boundaries, boundary conditions, three-dimensional flows, complex geometry, Neumann-to-Dirichlet, vorticity, vortex methods, particle methods.
1991 MSC: Primary 65N, 65F, 65D; Secondary 65M, 35K20, 65L10, 35B05, 35C15

Email address: Philippe.Poncet@math.univ-toulouse.fr (Philippe Poncet[*,†]).

Computing fields satisfying a flux condition on an arbitrarily shaped body in three dimensions of space is a challenge in many engineering and research fields, including fluid dynamics, electromagnetism, acoustic. One of the main difficulties in the numerical simulation of these problems is to efficiently and accurately take account of body effects on surrounding environment.

In the present study, we are interested in the analysis of an immersed boundary method for computing a flow satisfying no-flow-through boundary conditions on an arbitrarily shaped body. Such a method concerns the three-dimensional Euler equations with smooth boundaries, or half a step of Navier-Stokes equation, when performing a time-splitting algorithm, possibly with edges on the geometry. Computation of three-dimensional flow dynamics using vortex-in-cell methods will be the main focus of the present article.

Immersed boundary methods consist in using a domain of simple geometry (typically a Cartesian box), in which a surface is set, whose impact on solution is computed either by introducing sources on the body, with strength or distribution to estimate.

One will show that in their velocity-vorticity formulation, the geometrical aspect of such boundary conditions can be reduced to the resolution of a scalar Poisson equation, that is to say an elliptic problem. This is the opposite of usual immersed boundary methods based on velocity-pressure formulations, which involve parabolic or hyperbolic equations, for which time dependent jump relations are generally difficult to handle. Furthermore, to the opposite of conventional immersed boundary methods for vorticity, the present approach does not consider sources of vorticity, but a potential velocity.

In the present article, the following questions are addressed : Are spectrum and singular value decomposition sufficiently clustered to ensure fast convergence when used as iterative method ? Are the properties observed in the cylindrical case also valid for an arbitrarily shaped body, even with sharp edges ? Is it possible to recover this good conditioning when the surface is spread over grid points instead of using an approximated surface, such as methods inspired from [7] ? Should this method be preferred to a boundary integral method ? Is it possible to consider real-world geometries at lowest possible computational cost, using such immersed boundary methods ? In this case, is it possible to exhibit, in practice, the properties obtained theoretically, such as algebraically satisfied no-flow-through and circulation conservation ?

Despite the fact that immersed boundary methods have been intensively used and improved during the last decade, little attention has been paid to the nature and numerical analysis of the source-to-flow-through linear application (and the related linear system), such as its conditioning, spectra spreading and bounds, or self-adjointness.

Section 1 defines the Navier-Stokes and Euler equations in their vorticity formulation, and how Poisson equations are the only equations involves. Immersed boundary technique is set up in section 2, where the source-to-flow-through application is eventually defined. Theoretical numerical analysis is provided in section 3, and practical considerations are developed in section 4, by means of two discretizations and a convergence benchmark. Section 5 provide a coupling between vortex-in-cell method and this immersed boundary technique.

1 Fluid dynamics and vorticity formulation

1.1 Navier-Stokes equations in vorticity-velocity formulation

One considers a fluid domain around a body $\mathcal{B}$, whose boundary is denoted Γ, and the three-dimensional incompressible Navier-Stokes equations in their velocity-pressure formulation, for a fluid of constant density ϱ :

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \Delta \mathbf{u} = \frac{\mathbf{f}}{\varrho} - \frac{\nabla p}{\varrho} \tag{1}$$

where $\mathbf{u}$ is the divergence-free velocity field satisfying the no-slip condition $\mathbf{u} = 0$ on Γ, p the pressure, ν the kinematic viscosity, and $\mathbf{f}$ the external force, assumed to derive from a potential (*ie* is the gradient of a scalar function). Taking the curl of equation (1) and introducing the vorticity as $\boldsymbol{\omega} = \mathrm{curl}\mathbf{u}$, one gets :

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \nabla \mathbf{u} - \nu \Delta \boldsymbol{\omega} = 0 \tag{2}$$

with kinematic boundary conditions $\mathbf{u} = 0$.

When trying to integrate numerically such a dynamical system, it is common to use a time-splitting algorithm [2] over a time step $[t_n,\, t_n + \delta t]$. On the one hand, one considers the transport, or convective, effects :

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \nabla \mathbf{u} = 0 \tag{3}$$

with no-flow-through condition $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ. The final vorticity field at the end of the time step is denoted $\boldsymbol{\omega}^*$. On the other hand, one considers the viscous effects :

$$\frac{\partial \boldsymbol{\omega}}{\partial t} - \nu \Delta \boldsymbol{\omega} = 0 \tag{4}$$

with full no-slip conditions $\mathbf{u} = 0$ on boundaries, and with an initial condition being the final vorticity $\boldsymbol{\omega}^*$ obtained by solving equation (3). The vorticity obtained at the end of the viscous step is consistent with the Navier-Stokes equations (2).

Moreover, in practice, the viscous part can be itself split by linearity in two part, thus without approximation. Basically, equation (4) is split into an equation in the fluid whose solution is $\boldsymbol{\omega}^f$, and an equation "close to boundaries", whose solution is denoted $\boldsymbol{\omega}^b$:

$$\begin{cases} \dfrac{\partial \boldsymbol{\omega}^f}{\partial t} - \nu \Delta \boldsymbol{\omega}^f = 0 \\ \boldsymbol{\omega}^f(t = t_n) = \boldsymbol{\omega}^* \\ \text{Arbirary boundary} \\ \text{conditions} \end{cases} \quad (5) \qquad \begin{cases} \dfrac{\partial \boldsymbol{\omega}^b}{\partial t} - \nu \Delta \boldsymbol{\omega}^b = 0 \\ \boldsymbol{\omega}^b(t = t_n) = 0 \\ \mathbf{u}^b = -\mathbf{u}^f \text{ on boundaries} \end{cases} \quad (6)$$

Consequently, $\boldsymbol{\omega}^f + \boldsymbol{\omega}^b$ is the exact solution of equation (4).

While it is easy to solve equation (5) with any standard scheme, equation (6) is subject to kinematic boundary conditions. Such a boundary condition, involving velocity, translates the problem and the difficulty of pressure estimation (in velocity-pressure formulations of Navier-Stokes equations) in terms of vorticity.

An efficient way to solve equation (6) is to use Chorin's algorithm [4], involving only vorticity. Basically, Chorin's algorithm consists in creating a vorticity flux at boundaries, aiming at cancelling the spurious velocity $\mathbf{u}^f$. Using sources of vorticity with Gaussian spreading (the heat kernel) on the surrounding environment of body has been successfully used in the 90's for two-dimensional problems [18,19], and accurate formulæ in 3D has been recently established and analyzed [29]. Indeed, $\boldsymbol{\omega}^b$ can be written under its fundamental formulation

$$\boldsymbol{\omega}^b(x,t) = \nu \int_0^t \int_\Gamma \mu(\xi,\tau) \exp\left(-\frac{\|x-\xi\|_2^2}{4\nu(t-\tau)}\right) \mathrm{d}\sigma(\xi)\mathrm{d}\tau \quad (7)$$

where σ is the measure on Γ. Then, it has been shown in [29] that

$$\mu(x,t)\cdot\mathbf{k}_i(x) \simeq \frac{-2\mathbf{n}\times\mathbf{u}^f(x,t)\cdot\mathbf{k}_i(x)}{1-2(\bar{\kappa}(x)-\kappa_i(x))\sqrt{\nu t/\pi}} \quad (8)$$

where $\mathbf{k}_1$ and $\mathbf{k}_2$ are the two tangential vector fields of Γ (orthogonal to one another), κ_i is the curvature in the direction $\mathbf{k}_i$ and $\bar{\kappa}$ is the mean arithmetic curvature. Results on three-dimensional control, using such a technique, have followed [27,30]. Some improvements have also been brought to ensure circulation conservation [36].

When using these two equations in the context of Lagrangian methods, one can notice that vorticity flux at boundaries is natural when using particle of fluids, while diffusion stencils for Lagrangian treatment of equation (5) are provided in [28].

1.3 Treatment of convection and Euler equations

Let the body $\mathcal{B} \subset \mathbb{R}^n$ be a sufficiently smooth bounded domain, connex or not. The body and its boundary are included in a domain Ω possibly unbounded, but of simple geometry, which means practically that a fast solver is available for the geometry of Ω. One also considers the domain $\Omega^* = \Omega \setminus \overline{\mathcal{B}}$ complementing the body closure $\overline{\mathcal{B}}$ in Ω, as displayed on figure 2.

As soon as it is possible to build the velocity $\mathbf{u}$ from the vorticity $\boldsymbol{\omega}$, by means of a linear operator $\mathcal{A}$, equation on convective effects (3) can written

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathcal{A}\boldsymbol{\omega} \cdot \nabla \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \nabla \mathcal{A}\boldsymbol{\omega} = 0 \tag{9}$$

with no-flow-through condition $\mathbf{u} \cdot \mathbf{n} = \mathcal{A}\boldsymbol{\omega} \cdot \mathbf{n} = 0$ on $\Gamma = \partial\mathcal{B}$, which is no more than the usual three-dimensional Euler equations.

A Lagrangian formulation of this equation, once the operator $\mathcal{A}$ is available, is known as vortex-in-cell method and consists in discretizing the fluid volume Ω by a set $k = 1..K$ of cells of volume v_k, at location $\boldsymbol{\xi}_k$, and holding a quantity of vorticity $\boldsymbol{\omega}_k$ (see [10,5,8] for instance). Equation (9) is then written by means of cell dynamics :

$$\frac{\mathrm{d}\boldsymbol{\omega}_k}{\mathrm{d}t} = [\boldsymbol{\omega} \cdot \nabla \mathcal{A}\boldsymbol{\omega}]_{\boldsymbol{\xi}_k}, \quad \frac{\mathrm{d}\boldsymbol{\xi}_k}{\mathrm{d}t} = [\mathcal{A}\boldsymbol{\omega}]_{\boldsymbol{\xi}_k}, \quad \frac{\mathrm{d}v_k}{\mathrm{d}t} = v_k [\mathrm{div} \mathcal{A}\boldsymbol{\omega}]_{\boldsymbol{\xi}_k} = 0 \tag{10}$$

No additional boundary conditions are required on this dynamical system since the condition $\mathbf{u} \cdot \mathbf{n} = 0$ can be included in the definition of application $\mathcal{A}$. The evaluation of $[\mathcal{A}\boldsymbol{\omega}]_{\boldsymbol{\xi}_k}$ has been historically performed using Biot-Savart laws (see [16,10] for example), but coupling between particles and grids has been proven to be more efficient (see [8,9,27]).

A classical approach for defining $\mathbf{u} = \mathcal{A}\boldsymbol{\omega}$ is to introduce the stream function $\boldsymbol{\psi}^* : \Omega^* \to \mathbb{R}^3$ in the fluid domain Ω^*, satisfying

$$-\Delta \boldsymbol{\psi}^* = \boldsymbol{\omega} \ \ in \ \ \Omega^*$$

with boundary conditions such as $\mathrm{div} \boldsymbol{\psi}^* = 0$ and $\mathrm{curl} \boldsymbol{\psi}^* \cdot \mathbf{n} = 0$ on $\partial\Omega^*$. One gets the velocity by $\mathbf{u} = \mathrm{curl} \boldsymbol{\psi}^*$.

Unfortunately, these two boundary conditions are coupled. This leads to major difficulties, especially when implementation of rigidity matrix is needed. Some geometries allow to uncouple the stream components, such as the cube (see next section for instance), the cylinder [9] or any conformal mapping of the cylinder (see figure 1).

2 Immersed boundary method for vortical flows

The immersed boundary method consist in two main ideas. On the one hand, one considers a domain Ω of simple geometry, containing both the body $\mathcal{B}$ and the fluid domain Ω^* (see figure 2). The domain Ω is usually a cube in Cartesian coordinates, for which a fast solver for Poisson equations is available. On the other hand, one dispatches over the body surface Γ sources in order to take into account the effect of the body in its surrounding environment, that is to say in order to satisfy the adequate boundary conditions on Γ.

Immersed boundary methods have been dramatically improved and generalized since 1974 [25], involving integral techniques, or joined to finite differences [13,21]. Most of improvements consists in providing more accurate formulations [14,36,15], with a special challenge for three-dimensional fluid dynamic problems, either with explicit [35] or implicit [32] formulations. Time discretization improvements for time dependant problems have also been performed, focusing on the conservation aspects [6]. Some reviews of existing immersed boundary methods are available [26].

Numerical simulation involving Navier-Stokes or Euler equations, and immersed boundaries, have mainly two features : their velocity-pressure formulation involves complex numerical techniques for time and space jump estimations [20,37,22], while velocity-vorticity formulation involves only elliptic problems [9,28] even with non-stationary boundary conditions [27]. Moving bodies are also a challenge, for which accurate schemes are now available, either in vorticity formulation [12] or in pressure formulation [38].

2.1 Immersed boundary technique : problem setup

The vorticity being linked to stream and velocity by means of Poisson equation, time is not involved, thus leading to algorithms easier to handle, compared to immersed boundary methods developed in velocity-pressure context.

Conventional immersed boundary methods using vorticity use sources of vorticity [17]. The present approach consists in using a Helmholtz decomposition of velocity :

$$\mathbf{u} = \mathrm{curl}\boldsymbol{\psi} - \nabla\phi \tag{11}$$

where the potential ϕ contains everything in order to satisfy the boundary conditions. Indeed, the stream components satisfy

$$-\Delta\boldsymbol{\psi}_x = \boldsymbol{\omega}_x, \quad -\Delta\boldsymbol{\psi}_y = \boldsymbol{\omega}_y, \quad -\Delta\boldsymbol{\psi}_z = \boldsymbol{\omega}_z \tag{12}$$

The boundary conditions of these Poisson equations are homogeneous Dirichlet

conditions, except for the following components for which homogeneous Neumann conditions are set :

- on planes for which x is constant, $\partial \boldsymbol{\psi}_x / \partial x = 0$,
- on planes for which y is constant, $\partial \boldsymbol{\psi}_y / \partial y = 0$,
- on planes for which z is constant, $\partial \boldsymbol{\psi}_z / \partial z = 0$,

This way, for any face of the Cartesian box Ω, the stream $\boldsymbol{\psi}$ satisfies $\mathrm{div}\boldsymbol{\psi} = 0$ everywhere on boundary $\partial\Omega$. Since $\boldsymbol{\omega}$ is supposed divergence-free, the quantity $\mathrm{div}\boldsymbol{\psi}$ satisfies the equation $-\Delta \mathrm{div}\boldsymbol{\psi} = \mathrm{div}\boldsymbol{\omega} = 0$ and consequently $\mathrm{div}\boldsymbol{\psi} = 0$ everywhere in Ω.

Moreover, the body effects are contained in the harmonic potential ϕ satisfying

$$\begin{cases} -\Delta\phi = 0 \text{ in } \Omega^* \\ \dfrac{\partial \phi}{\partial \mathbf{n}} = \mathrm{curl}\boldsymbol{\psi} \cdot \mathbf{n} \text{ on } \Gamma = \partial\mathcal{B} \\ \phi = 0 \text{ or } \partial\phi/\partial\mathbf{n} = 0 \text{ on } \partial\Omega \end{cases} \tag{13}$$

Consequently, the velocity field $\mathbf{u}$ satisfies all the required relations :

- $\mathrm{div}\mathbf{u} = -\Delta\phi = 0$,
- $\mathrm{div}\boldsymbol{\psi} = 0$ so $\mathrm{curl}\mathbf{u} = \mathrm{curl}(\mathrm{curl}\boldsymbol{\psi}) = -\Delta\boldsymbol{\psi} + \nabla \mathrm{div}\boldsymbol{\psi} = -\Delta\boldsymbol{\psi} = \boldsymbol{\omega}$
- $\mathbf{u} \cdot \mathbf{n} = \mathrm{curl}\boldsymbol{\psi} \cdot \mathbf{n} - \mathbf{n} \cdot \nabla\phi = 0$

Immersed boundary method is then applied to equation (13) by setting a scalar source T over the body boundary Γ only (so that $-\Delta\phi = 0$ in the fluid domain Ω^*). Equation (13) can then be replaced by

$$-\Delta\phi = T \tag{14}$$

The main advantages of using an immersed boundary technique on this potential are that the potential does not create artificial vorticity, is compatible with the Euler equation (for smooth domains), and is a scalar Poisson equation. The equation (14) can be solved by using grid solvers of Poisson equation or equivalently by using integral methods and Biot-Savart laws [16], possibly improved with multipole expansion.

Using immersed boundary technique on the potential velocity leads to four Poisson problems (12,14) in a Cartesian box Ω, for which fast solvers are available. Among fast solvers, two efficient ones are FISHPACK and MUDPACK, which involve FFT, standard finite difference stencils, and more sophisticated technique such as cyclic reduction [1,33].

Furthermore, the robustness of immersed boundary methods, used for discretization

in space, is especially interesting when used together with vortex methods, renown for their high stability in time [10,5,8].

In the present article, one shows how to compute the potential velocity source T in equation (14), which is the only unknown herein. The source T is equivalently defined by means of a Poisson equation or by means of integral method (evaluation of Green kernels).

The main question addressed in this article is why does it always work ? It will be shown that the matrix of the discretized linear problem defining T is well-conditioned. This is a key point in convergence of immersed boundary method, especially when iterative methods are used to find the correct potential T. Moreover, this will be illustrated by means of singular value distribution for several examples, such as the sphere, a NACA airfoil, and two real-life geometries (aircraft and bridge).

2.2 A harmonic problem for no-flow-through condition

As described there above, Euler equations, or the convective half-step for Navier-Stokes equations, is written as a dynamical system (9) :

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathcal{A}\boldsymbol{\omega} \cdot \nabla \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \nabla \mathcal{A}\boldsymbol{\omega} = 0$$

with no-flow-through condition $\mathbf{u} \cdot \mathbf{n} = 0$ on $\Gamma = \partial\mathcal{B}$, where $\mathcal{A}$ is the vorticity-to-velocity linear application defined by

$$\mathbf{u} = \mathcal{A}\boldsymbol{\omega} = \mathrm{curl}\boldsymbol{\psi} - \nabla\phi \tag{15}$$

The stream function $\boldsymbol{\psi}$ and the potential velocity ϕ are defined by the Poisson equations (12,14) :

$$-\Delta\boldsymbol{\psi} = \boldsymbol{\omega}, \quad -\Delta\phi = T \tag{16}$$

with adequate uncoupled boundary conditions, in order to have four scalar Poisson equations.

The goal of source T is to cancel flow through $g = \mathrm{curl}\boldsymbol{\psi} \cdot \mathbf{n}$, and is defined as a singular source over Γ, thus a generalized function (and not a convolution) given by

$$T_\mu(\xi) = \int_\Gamma \mu(s)\xi(s)\mathrm{d}\sigma(s) \tag{17}$$

where $\mu : \Gamma \to \mathbb{R}$ is the source distribution over Γ, and $\xi : \Omega \to \mathbb{R}$ a test function, that is to say infinitely differentiable and compact supported in Ω.

The immersed boundary technique consists in finding a source distribution μ such

as the solution of

$$\begin{cases} -\Delta\phi = T_\mu \text{ on } \Omega \\ \phi = 0 \text{ or } \partial\phi/\partial\mathbf{n} = 0 \text{ on } \partial\Omega \end{cases} \tag{18}$$

satisfies

$$\left.\frac{\partial\phi}{\partial\mathbf{n}}\right|_\Gamma = g \tag{19}$$

for a given flow-through velocity $g = \mathrm{curl}\boldsymbol{\psi}\cdot\mathbf{n}$, where Ω is the computational domain of simple geometry, typically a cube, for which a fast solver is available. One can notice that the linear application $g \mapsto \phi$ is well defined, at least up to a constant, and its trace $\mathcal{R} : g \mapsto \phi|_\Gamma$ is the usual Neumann-to-Dirichlet operator.

Consequently, the key point is the use and the analysis of the linear application

$$\Lambda : \mu \longmapsto \left.\mathbf{n}\cdot\nabla\phi\right|_\Gamma \tag{20}$$

called from now on the *source-to-flow-through*, with the normal field $\mathbf{n}$ pointing toward the fluid domain Ω^*. It can be summarized by the following diagram :

$$\begin{array}{ccc} \mu : \Gamma \to \mathbb{R} & \xrightarrow{\Lambda} & \left.\mathbf{n}\cdot\nabla\phi\right|_\Gamma \\ \downarrow & & \uparrow \\ T_\mu & \longrightarrow & \phi : \Omega^* \to \mathbb{R} \end{array} \tag{21}$$

In case of invertible application Λ, the source distribution of singularity T_μ is $\mu = \Lambda^{-1}(g)$. Equations (18)-(19), equivalent to equation (13), read then

$$\begin{cases} -\Delta\phi = T_{\Lambda^{-1}g} \text{ on } \Omega \\ \mathcal{L}\phi = 0 \text{ on } \partial\Omega \end{cases} \tag{22}$$

where $\mathcal{L} = \mathrm{Id}$ for far-field Dirichlet conditions, or $\mathcal{L} = \partial/\partial\mathbf{n}$ for far-field Neumann condition. This equation can be solved with high efficiency when one has sufficient knowledge on linear application Λ.

2.3 Potential immersed boundary or boundary integral method ?

The use of Λ is equivalent to evaluations of Green kernels and equation (22) is equivalent to solve the usual boundary integral equation. Indeed, potential ϕ can be expressed into its integral form

$$\phi(x) = \int_\Gamma K(x,y)q(y)\mathrm{d}\sigma(y) \tag{23}$$

where K is a Green function, $(4\pi|x-y|)^{-1}$ in $\mathbb{R}^3$, and q the density defined on domain boundary, solution of the integral equation

$$-\frac{q(x)}{2} + \int_\Gamma \mathbf{n} \cdot \nabla K(x,y)\, q(y)\, \mathrm{d}\sigma(y) = g(x) \tag{24}$$

Solving equation (24) is equivalent to build and invert operator Λ.

This gives satisfactory results when velocity is computed by means of integral formulation, such as Biot-Savart laws, possibly with multipole expansion. Nevertheless, it is commonly admitted that in the context of fluid dynamics, especially vortex methods, hybrid grid-particle methods are much faster than Biot-Savart laws for large three-dimensional flows.

When computing a velocity field on a grid, one may prefer the use of equation (22). Indeed, the application Λ is build to make the numerical solution satisfy algebraically the no-flow-through condition on the body, as shown in section 4.4 and 4.5.

Furthermore, boundary integral method is shown to be very sensitive to staircase effects when the potential is computed on a grid, especially for the Navier-Stokes equations, for which a strong and thin boundary layer is located in the body neighbourhood. This makes the method developped in this paper an interesting alternative to boundary integral when performing flow computation on a grid.

3 Numerical Analysis of Source-to-Flow-Through application

In this section, one focuses on fundamental properties of the linear application Λ described above. Its nature is exhibited below, in section 3.1, to be a jump-of-flux to flux application : the unknown source distribution μ is actually the jump of normal derivatives of the solution through the body Γ. This proves that the original harmonic problem (13) is equivalent to the surface singularity formulation (18,19).

Good conditioning properties are investigated and justified in section 3.2, using theoretical tools of functional analysis. Indeed, considering Λ as an harmonic operator of degree zero allows to exclude large aspect ratio of spectrum.

Furthermore, since the coefficients of the matrix $\mathbf{M}$ encoding application Λ are by scalar product with basis vectors, that is to say $< \mathbf{M}\mathbf{e}_i, \mathbf{e}_j >$, it is interesting to establish normality and self-adjointness properties of Λ is the $\mathbb{L}^2(\Gamma)$ context. The self-adjointness, and even normality, are shown to be unavailable in section 3.3. Moreover, annex A establishes that self-adjointness holds in a fractional Sobolev space.

3.1 Relations between source distribution, flux and jump

We are interested in proving that the source distribution μ of surface singularity is weakly equal to the jump of flux over Γ of the solution harmonic on both sides of Γ. Let $\phi_1 : \Omega^* \to \mathbb{R}$ and $\phi_2 : \mathcal{B} \to \mathbb{R}$ be respectively the solutions of :

$$\begin{cases} -\Delta\phi_1 = 0 \text{ on } \Omega^* \\ \mathcal{L}\phi_1 = 0 \text{ on } \partial\Omega \\ \dfrac{\partial\phi_1}{\partial\mathbf{n}} = g \text{ on } \Gamma = \partial\mathcal{B} \end{cases} \quad \text{and} \quad \begin{cases} -\Delta\phi_2 = 0 \text{ on } \mathcal{B} \\ \phi_2 = \phi_1|_\Gamma \text{ on } \Gamma = \partial\mathcal{B} \end{cases} \tag{25}$$

where the body $\mathcal{B}$ is supposed sufficiently smooth to avoid any drop of regularity due to the normal field $\mathbf{n}$.

If ones denotes by ϕ the extension of ϕ_1 by ϕ_2 in body $\mathcal{B}$, the jump of flux of ϕ is written :

$$\left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma = \left.\frac{\partial\phi_1}{\partial\mathbf{n}_{\Omega^*}}\right|_\Gamma + \left.\frac{\partial\phi_2}{\partial\mathbf{n}_\mathcal{B}}\right|_\Gamma \tag{26}$$

where $\mathbf{n} = \mathbf{n}_{\Omega^*}$ and $\mathbf{n}_\mathcal{B} = -\mathbf{n}_{\Omega^*}$ are respectively the normal fields on Γ directed toward the interior of Ω^* and $\mathcal{B}$. Moreover, there is no jump of ϕ through Γ :

$$[\phi]_\Gamma = \phi_1|_\Gamma - \phi_2|_\Gamma = 0 \tag{27}$$

In the sense of generalized functions, one can write from equations (25) the following Green formulae, in the spirit of boundary element methods :

$$\int_\Gamma \left.\frac{\partial\phi_1}{\partial\mathbf{n}_{\Omega^*}}\right|_\Gamma v \,\mathrm{d}\sigma = \int_{\Omega^*} \Delta\phi_1 v \,\mathrm{d}\lambda + \int_{\Omega^*} \nabla\phi_1 \cdot \nabla v \,\mathrm{d}\lambda \tag{28}$$

and

$$\int_\Gamma \left.\frac{\partial\phi_2}{\partial\mathbf{n}_\mathcal{B}}\right|_\Gamma v \,\mathrm{d}\sigma = \int_\mathcal{B} \Delta\phi_2 v \,\mathrm{d}\lambda + \int_\mathcal{B} \nabla\phi_2 \cdot \nabla v \,\mathrm{d}\lambda \tag{29}$$

for any test function v, where σ is the measure on Γ induced by the Lebesgue measure λ in Ω. From these two equations, one gets

$$\int_\Gamma \left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma v \,\mathrm{d}\sigma = \int_\Omega \nabla\phi \cdot \nabla v \,\mathrm{d}\lambda = -\int_\Omega \Delta\phi \, v \,\mathrm{d}\lambda \tag{30}$$

and, since $-\Delta\phi = T_\mu$, one has finally

$$\int_\Gamma \left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma v \,\mathrm{d}\sigma = T_\mu(v) = \int_\Gamma \mu \, v \,\mathrm{d}\sigma \qquad \forall v \in \mathcal{D}(\Omega) \tag{31}$$

and eventually the following weak equality $\mu = [\partial\phi/\partial\mathbf{n}]_\Gamma$.

Consequently, the inverse source-to-flow-through application Λ^{-1} is the application that takes a Neumann condition $\partial\phi/\partial\mathbf{n}$ on a virtual boundary Γ and associates the jump of flux through Γ of the solution, which coincides with the singularity $-\Delta\phi = T_\mu$. This can be summarized with the following notation :

$$\Lambda^{-1} : \left.\frac{\partial\phi_1}{\partial\mathbf{n}}\right|_\Gamma \longmapsto \left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma \tag{32}$$

in the case of invertible source-to-flow-through linear application Λ.

3.2 Conditioning of source-to-flow-through

The good conditioning of source-to-flow-through represents the ability of its spectrum to be included in an annular domain, that is to say to avoid large aspect ratio between eigenvalues, forbiding them to go to zero or infinity as the surface is refined.

This section uses functional analysis, and this theoretical approach leads to qualitative result on the conditioning of the matrix involved in the discretized immersed boundary method.

In term of harmonic analysis, avoiding large ratio between eigenvalues is interpreted as being close to identity. This means that if $\Lambda : \mu \to \mathbf{n}\cdot\nabla\phi|_\Gamma$ is an integro-differential operator of degree zero, one can expect a good conditioning.

In order to establish this property, one needs to introduce the fractional Sobolev spaces $\mathbb{H}^s$. $\mathbb{H}^1$ denotes the set of functions for which the derivatives are in $\mathbb{L}^2$, functions of integrable square.

One considers equation (13) with a solution $\phi \in \mathbb{H}^1(\Omega^*)$, with $\mathbb{H}^1$ proloungement in body $\mathcal{B}$, with no jump through Γ. It follows that $T_\mu = -\Delta\phi \in \mathbb{H}^{-1}(\Omega)$ and

$$\mathbf{n}\cdot\nabla\phi\Big|_\Gamma \in \mathbb{H}^{-1/2}(\Gamma) \tag{33}$$

Furthermore, the jump of flux is the trace of two gradients of functions in $\mathbb{H}^1$ on both sides of Γ :

$$\left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma = \left.\frac{\partial\phi_1}{\partial\mathbf{n}_{\Omega^*}}\right|_\Gamma + \left.\frac{\partial\phi_2}{\partial\mathbf{n}_{\mathcal{B}}}\right|_\Gamma \in \mathbb{H}^{-1/2}(\Gamma) \tag{34}$$

Since it has been established in last section that the source distribution μ is the jump of flux of ϕ through Γ, one gets

$$\mu = \left[\frac{\partial\phi}{\partial\mathbf{n}}\right]_\Gamma \in \mathbb{H}^{-1/2}(\Gamma) \tag{35}$$

Finally, from equations (33,35), one gets that the source-to-flow-through linear application is an homomorphism of $\mathbb{H}^{-1/2}(\Gamma)$, thus a zeroth degree integro-differential operator, summarized as follow :

$$\begin{array}{ccc} \mu \in \mathcal{V} \subset \mathbb{H}^{-1/2}(\Gamma) & \xrightarrow{\Lambda} & \mathbf{n} \cdot \nabla\phi \in \mathbb{H}^{-1/2}(\Gamma) \\ \downarrow & & \uparrow \\ T_\mu \in \mathbb{H}^{-1}(\Omega) \subset \mathcal{D}'(\Omega) & \longrightarrow & \phi \in \mathbb{H}^1(\Omega^*) \end{array} \tag{36}$$

Furthermore, in case of non-convex bodies, one may have $\phi \in \mathbb{H}^{1-\varepsilon}(\Omega^*)$ instead of $\phi \in \mathbb{H}^1(\Omega^*)$, which does not alter the results presented in the present study.

3.3 Non-normality of source-to-flow-through application

One considers the interval $[-1, 1] \subset \mathbb{R}$ in which the interval $]-\alpha, \beta[$ is immersed. The $\mathbb{L}^2$ scalar product is then defined in a discrete sense, by the Dirac measure $\delta_{-\alpha}^{\otimes 2} + \delta_{\beta}^{\otimes 2}$, or equivalently :

$$\langle f, g \rangle = f(-\alpha)g(-\alpha) + f(\beta)g(\beta) \tag{37}$$

This leads to the following one-dimensional problem with two real parameters A and B :

$$\begin{cases} -u'' = A\delta_{-\alpha} + B\delta_{\beta} \quad \text{in }]-1, 1[\\ u(-1) = u(1) = 0 \end{cases} \tag{38}$$

whose solution can be explicited as piecewise linear function. The application is $\Lambda(A, B) = (-u'(-\alpha^-), u'(\beta^+))$ (sign in front of u' comes from the normal pointing toward the exterior of interval $]-\alpha, \beta[$), and is represented by the matrix

$$\mathbf{M} = \frac{1}{2}\begin{bmatrix} 1-\alpha & \beta-1 \\ \alpha-1 & -(\beta+1) \end{bmatrix} \tag{39}$$

In a spatially symmetric configuration, for example $\alpha = \beta = \varepsilon$, one obtains self-adjointness, but as soon as one considers an arbitrary configuration, self-adjointness and normality are broken. For example, in the case $\alpha = 0$ and $\beta > 0$, the matrix $\mathbf{M}$ is non-normal, that is to say $\mathbf{M}\mathbf{M}^T \neq \mathbf{M}^T\mathbf{M}$.

In order to solve equation (18,19), one can notice that the source-to-flow-through application Λ is linear and consequently its matrix can be encoded. Non-normality implies that the conditioning (in case of invertible source-to-flow-through linear application) cannot be provided by eigenvalues of largest and smallest modules, but only by extreme singular values.

4 Discrete Density-to-Flow-Through application

In this section, one describes properties and improvement of the discretization of the source-to-flow-through application. The generalized function for body sources, defined by equation (17), is naturally discretized as

$$T_\mu = \sum_{p=1}^{K} \mu_p \delta_{\mathbf{X}_p} s_p \tag{40}$$

for a surface described by a set of elementary surfaces s_p pointwise located at $\mathbf{X}_p$ with a normal field denoted $\mathbf{n}_p$. The set of sources values is then μ_p.

Such a discretization is also easily implemented when points $\mathbf{X}_p$ are located on grid points, once again in order to use fast solvers for Poisson equations. Two ways of setting these points to grid points, as displayed on figure 3, are as follows :

- The surface is the graph of a function, and normal field and elements of surfaces are deduced from this function,
- The surface is more tortuous and is described only by a set of points, normals and elementary surfaces, in which case geometry is interpolated on grid points.

In practice the first case (graph-defined surfaces) lead to one layer of points, while the second case involves usually two grid-points layers. It is shown thereafter that two-layer configurations do not lead to conditioning discrepancy. Nevertheless, conditioning can be improved by means of an explicit weighting technique.

4.1 Graph-defined surfaces

One considers a domain $\Omega =]-2\pi, 2\pi[^3 \subset \mathbb{R}^3$ and a spherical body defined by the Euclidean ball of radius $R = 3$ whose boundary is the sphere $\Gamma = \partial\mathcal{B}$, on which the normal field is defined by $\mathbf{n}(X) = X/\|X\|_2$. One wishes to study the properties of source-to-flow-through for the surface Γ, displayed with its normal bundle on figure 4, in the discrete context.

Numerical derivations are provided by standard centred finite difference scheme, except on singular points (on body) where schemes are naturally one-sided. The domain Ω is discretized uniformly in each direction by $N + 1$ points, the spatial step being consequently $h = 4\pi/N$ and the elementary volume $v = h^3$.

In order to define the discrete source-to-flow-through application, the sphere is discretized by its best approximation on grid points. One proceeds as follows : the top half-sphere is defined as the graph of function $z = f(x, y) = (R^2 - x^2 - y^2)^{1/2}$.

The sphere can be consequently approximated by the best approximant $z_{i,j}$ of

$\pm f(x_i, y_j)$ over the two dimensional mesh $x_i = -2\pi + (i-1)h$ and $y_j = -2\pi + (j-1)h$, which satisfies $x_i^2 + y_j^2 \leqslant R^2$, for $i, j = 1 \ldots N+1$. One denotes from now on $\{\mathbf{X}_p\}_{p=1..K}$ the set of points in $\mathbb{R}^3$ approximating the sphere. The generalized function $\tilde{T_\mu}$, surface supported, of source distribution $\{\mu_p\}_{p=1..K}$ is consequently defined in a discrete way as follows :

$$\tilde{T_\mu} = \sum_{p=1}^{K} \mu_p \delta_{\mathbf{X}_p} s_p \tag{41}$$

where s_p are the elements of surface defined thanks to the mapping :

$$s_p = \left\| \frac{\partial F}{\partial x}(\mathbf{X}_p) \wedge \frac{\partial F}{\partial y}(\mathbf{X}_p) \right\|_2 h^2 \quad \text{with} \quad F(x,y) = \left(x, y, \sqrt{R^2 - x^2 - y^2} \right) \tag{42}$$

which also reads $s_p = Rh^2/(\mathbf{X}_p \cdot e_z)$ where e_z is the basis vector in which direction the graph is made.

The potential ϕ, such as $-\Delta\phi = \tilde{T_\mu}$ in Ω, is computed in practice by the Poisson solver FISHPACK. The discrete source-to-flow-through linear application is then written

$$\begin{array}{rccc} \Lambda : & \mathbb{R}^K & \longrightarrow & \mathbb{R}^K \\ & \{\mu_p\}_{p=1..K} & \longmapsto & \mathbf{n}(\mathbf{X}_p) \cdot \nabla\phi(\mathbf{X}_p) \end{array} \tag{43}$$

which is represented by a matrix $\mathbf{M}$ in the canonical basis.

As expected, the source-to-flow-through matrix $\mathbf{M}$ is not normal, as displayed on figure 4. Indeed, normal matrix are in the kernel of the self-commutator $\mathbf{M}^T\mathbf{M} - \mathbf{M}\mathbf{M}^T$, which is not the case presently. Moreover, a normal matrix has its eigenvectors orthogonal two-by-two (except for conjugate pairs of complex eigenvectors). Furthermore, if one denotes by $\{\mathbf{V}_p\}_{p=1..K}$ the set of eigenvectors, $\mathbf{M}$ being normal would imply the matrix of Hermitian product of eigenvectors $A_{i,j} = \mathbf{V}_i \cdot \overline{\mathbf{V}_j}$ to be a three diagonal matrix. This brings a second proof that the source-to-flow-through application is not normal. Consequently, conditioning is accessible by singular value decomposition, and not by of eigenvalues.

4.2 Conditioning and preconditioning

The discrete source-to-flow-through application $\tilde{\Lambda}$ for a spherical body $\mathcal{B}$ or radius $R = 3$ immersed in the box $\Omega =]-2\pi, 2\pi[^3 \subset \mathbb{R}^3$ uniformly discretized, is represented by matrix $\mathbf{M}$. The singular value decomposition of matrix $\mathbf{M}$, for Ω discretized by N^3 intervals with $N = 24, 32, 48, 64, 96, 128$ and 192, are displayed on figure 5, as well as their singular value distribution. One observes that singular value distribution tends to converge toward a law centred on a value close to

$1/2$, while eigenvalue cloud tends to be more dense without spreading, as shown on figure 6.

The densification of singular value distribution around the singular value $1/2$ can be illustrated by the very simplified case of one point immersed in the one-dimensional interval $[-1, 1]$. This means that one considers the Poisson equation with the singular right hand side $A\delta_0$:

$$\begin{cases} -u'' = A\delta_0 \quad \text{in }]-1,1[\\ u(-1) = u(1) = 0 \end{cases} \tag{44}$$

whose solution satisfies $\Lambda(A) = u'(0^+) = -A/2$, which produces a unique singular value of $1/2$. This provides a scaling for a trivial case, to be compared with the more complex case of sphere immersion in $\mathbb{R}^3$.

Furthermore, the conditioning, obtained by the ratio of the largest by the smallest singular values, is plotted on figure 7 with dashed lines. One can see that conditioning value fluctuates close to 20. Oscillations of conditioning come from the presence of unwanted fluctuation in the surface mapping defining the elements of surface s_p. Indeed, since $s_p = R/(\mathbf{X}_p \cdot \mathbf{e}_z)$, this quantity becomes artificially high when point are dispatched close to the symmetry plane $z = 0$.

One can easily get rid of this problem by preconditioning on the source-to-flow-through application, which can be improved by using a scaled source distribution as follows. Indeed, definition of singularity $\tilde{T}_\mu$, defined by equation (41), can be modified into $\overline{T}_\mu$:

$$\overline{T}_\mu = \sum_{p=1}^{K} \mu_p \delta_{\mathbf{X}_p} h^2 \tag{45}$$

where $h = 4\pi/N$ is the step of discretization of the grid. This is equivalent to a preconditioning on source distribution, since

$$\overline{T}_\mu = \tilde{T}_{\bar{\mu}} \quad \text{with} \quad \bar{\mu}_p = \mu_p h^2 / s_p$$

Let this scaling factor be denoted H, so that $\tilde{\Lambda} = \Lambda \circ H$ is the preconditioned application, an iso-weighted source-to-flow-through application (since all points of discrete surface are weighted as h^2 instead of s_p). The conditioning of the iso-weighted application remains at a very low value (see left picture of figure 7), and makes the distribution of singular value converge toward a Gaussian distribution, as shown on figure 8. A lack of eigenvalue spreading is also observed on the eigenvalue clouds displayed on right picture of figure 6.

One considers a non-derivable geometry with the NACA-2412 airfoil (displayed on figure 9), in order to focus the impact of lack of surface regularity on the source-to-flow-through application. This surface is, as well as the spherical geometry, built

as the graph of a function immersed in $\Omega =]-2\pi, 2\pi[^3$. The discretization is the surface best grid approximant in a grid containing $(N+1)^3$ points, uniformly discretized, with a grid step $h = 4\pi/N$. The boundary conditions on box boundaries are homogeneous Neumann condition except on top and bottom boundaries (*ie* $z = \pm 2\pi$) where periodic conditions are set.

The resulting spectra is displayed on figure 9 in the case of iso-weighted source-to-flow-through application in the 96^3 box, and the $\mathbb{L}^2$ conditioning is found to be $\mathrm{Cond}\mathbf{M} = 9.98$. Consequently, the geometrical singularity at the body rear side does not break the well-conditioning of source-to-flow-through application. Furthermore, the grid approximation leads to a slight displacement of the surface, but still defines correctly the angle of trailing edge.

4.3 Density-to-Flow-Through application for interpolated surfaces

The goal of this section is to study the behaviour of the source-to-flow-through application for a body described only by a set of points on its surface, and not by the graph of an explicit function f as described above. The main drawback of this configuration is that spreading surface point among grid points by convolution leads to multiple layers of point in the grid, as shown on figure 3. But the advantages are, on the one hand, that no multiple point can be introduced (the matrix is full rank, so systematically invertible), and on the other hand, that no knowledge of the graph defining the surface is required.

One considers a surface defined by a set of points $\mathbf{X}_p \in \mathbb{R}^3$, indexed by $p = 1..N$, the surface normal vector $n_p \in \mathbb{R}^3$, and the elementary surface $s_p \in \mathbb{R}^+$ related to this point. The global surface measure is then

$$S = \sum_{p=1}^{N} s_p \tag{46}$$

These elements of surface are interpolated on a grid, in the spirit of [7,8], by means of a kernel $\zeta : \mathbb{R} \to \mathbb{R}$ whose support is $[-\varepsilon, \varepsilon]$. This kernel satisfies the moment properties [8], or even better the discrete moment properties such as kernels based on B-splines [23,9]. This allows to transfer the normal field $\mathbf{n}$ a points $\{\mathbf{X}_p\}_{p=1..N}$ on a grid point $\bar{\mathbf{X}}_j$ by means of the convolution $\mathbf{n} * \zeta^{\otimes 3}$, that is to say :

$$\bar{\mathbf{n}}_j = \sum_{p=1}^{N} \mathbf{n}_p \zeta^{\otimes 3}\left(\bar{\mathbf{X}}_j - \mathbf{X}_p\right) \tag{47}$$

where ε is in practice chosen equal to the grid step h.

One can notice that this sum can be restricted to points $\mathbf{X}_p$ located only in the ε-neighbourhood of $\bar{\mathbf{X}}_j$ (*ie* such as $\|\mathbf{X}_p - \bar{\mathbf{X}}_p\|_\infty < \varepsilon$), thanks to the support size of

kernel ζ.

In order to keep the normal orientation, one restricts kernels to functions of positive values (thus limited to second order kernels), in a similar way of hybrid particle methods for probabilistic equations (such as Vlasov or Fokker-Plank equations [31]), and to the opposite of fluid equations for which positivity is less important than accuracy (such as Euler or Navier-Stokes equations [9]).

The impact of multi-layer structured grid points on conditioning is very limited. The non-alteration of singular values is shown by the comparison of their distribution between graph defined and interpolated surfaces, plotted on figure 10 for the spherical body of radius $R = 3$.

This property can be understood on the simplified case of a one-dimensional problem : two points are immersed in the interval at location $\pm\varepsilon$ holding two Dirac functions, similar to equation (38) :

$$\begin{cases} -u'' = A\delta_{-\varepsilon} + B\delta_{\varepsilon} \quad \text{in }]-1,1[\\ u(-1) = u(1) = 0 \end{cases} \tag{48}$$

The application $\Lambda(A,B) = (u'(-\varepsilon^-), u'(\varepsilon^+))$ is then associated to the following non-normal matrix :

$$\mathbf{M} = -\frac{1}{2}\begin{bmatrix} 1-\varepsilon & -(1-\varepsilon) \\ 1-\varepsilon & 1+\varepsilon \end{bmatrix} \tag{49}$$

The two singular values $S_\pm$ are given by

$$2S_\pm^2 = 1 - \varepsilon + \varepsilon^2 \pm \varepsilon \left(2 - 2\,\varepsilon + \varepsilon^2\right)^{1/2} \tag{50}$$

and leads, by means of Taylor expansion, to the conditioning

$$\text{Cond}\mathbf{M} = \frac{S_+}{S_-} = 1 + \sqrt{2}\varepsilon + O\left(\varepsilon^2\right) \tag{51}$$

as long as ε is sufficiently small, and satisfies $\lim_{\varepsilon\to 0} \text{Cond}\mathbf{M} = 1$.

This technique is consequently robust and allows to handle easily very general geometries since only a set of points on the surface and their normal vectors are required to make the algorithm valid.

4.4 Convergence and sphere potential

In order to check convergence of the global algorithm, the harmonic potential $\phi : \mathbb{R}^3 \to \mathbb{R}$ satisfying $n \cdot \nabla\phi = 0$ on the sphere of radius $R = 1$ and

$$\lim_{|\mathbf{X}|\to\infty} \nabla\phi(\mathbf{X}) = \mathbf{K} \in \mathbb{R}^3$$

is computed numerically using the immersed boundary method with interpolated surface. Vector $\mathbf{K} = \mathbf{e}_x$ is consequently the streamwise direction. The well known exact solution of this problem, up to a constant, is

$$\phi_0(\mathbf{X}) = \left(1 - \frac{R^3}{2|\mathbf{X}|^3}\right) \mathbf{K} \cdot \mathbf{X} \tag{52}$$

By setting $\xi(\mathbf{X}) = \phi(\mathbf{X}) - \mathbf{K} \cdot \mathbf{X}$, the potential ϕ is solution of the following partial differential equation :

$$\begin{cases} -\Delta\xi = 0 \text{ on } \Omega^* \\ \dfrac{\partial\xi}{\partial n}(x) = -\mathbf{K} \cdot \mathbf{n} \text{ on } \Gamma = \partial\mathcal{B} \\ \dfrac{\partial\xi}{\partial\mathbf{n}} = 0 \text{ on } \partial\Omega \end{cases} \tag{53}$$

which is a particular case of equation (13), where $\Gamma = \partial\mathcal{B}$ is the sphere.

The immersed boundary formulation of problem (53) can be written under the form of equation (18), which is as a Poisson equation in a Cartesian box :

$$\begin{cases} -\Delta\xi = T_{\Lambda^{-1}(-\mathbf{K}\cdot\mathbf{n})} \text{ in } \Omega \\ \dfrac{\partial\xi}{\partial\mathbf{n}} = 0 \text{ on } \partial\Omega \end{cases} \tag{54}$$

where the singular right hand side T is defined by equation (17).

The function ξ is computed numerically, and the solution ϕ is recovered by $\phi(\mathbf{X}) = \xi(\mathbf{X}) + \mathbf{K} \cdot \mathbf{X}$.

Solving equation (54) numerically is equivalent to solve a 7-diagonals linear system. Such standard linear systems, here a discretized 3D Laplacian in a Cartesian box, can be solved with very efficient methods like cyclic reduction and/or Fourier decomposition, implemented for example in software FISHPACK [33]. The computational time for solving such systems behaves linearly with the number of grid points [9].

The numerical solution is computed in the box $\Omega =]-8, 8[^3$, with value fixed to the exact value on boundaries. In order to check consistency, and thus convergence by

means of stability of source-to-flow-through application, the conventional method of applying the discrete scheme to the exact solution is used. This means that one proceeds as follows :

- One considers the flux of the exact solution over the points of the discrete surface $\mathbf{X}_p$, that is to say $\mathbf{F}_p = \mathbf{n}_p \cdot \nabla\phi_0(\mathbf{X}_p)$, with $p = 1..P$,
- The resulting source distribution μ is obtained by the inverse discrete source-to-flow-through matrix $\mu = \Lambda^{-1}(\mathbf{F})$, as described by equation (43).
- The final potential ϕ is the potential generated by the source over the spherical body, with a source distribution μ : $-\Delta\phi = T_\mu$.

The overall relative error $\|\phi\|_2/\|\phi_0\|_2$ is then computed over the part of the computational box Ω outside the spherical body. The convergence is first order (see figure 11) as expected.

The resulting velocity $\mathbf{U}$, which is the quantity used in practice when computing fluid dynamics, is then compared to the exact velocity $\nabla\phi_0$. It appears that the relative error on velocity in this volume $\|\mathbf{U} - \nabla\phi_0\|_2/\|\nabla\phi_0\|_2$, is also first order, as shown on figure 11. One can also notice that first order accuracy is related to the Euler equations, but second order is reached when the Navier-Stokes equations with full no-slip conditions are considered, due to the velocity that tends to zero in the surface neighbourhood (see [9] for instance).

4.5 No discrepancy in conditioning for more complex geometry

In order to show that the matrix involved in this immersed boundary technique is always well conditioned, one considers the same problem as in the last section 4.4, but using an aircraft geometry for the body surface Γ.

The numerical parameters are as follows : the aircraft is immersed by interpolation in the box $\Omega = [0, 4] \times [-2, 2] \times [0, 1]$ and discretized by $128 \times 128 \times 32$ intervals, leading to a step $h = 2^{-5}$ in all directions. The aircraft is defined by 5617 points with their body normal vectors. Points and normals of the body boundary Γ are dispatched over $P = 3428$ grid points $\mathbf{X}_p$ (and normals n_p) by means of convolution formula (47), using the piecewise linear interpolation kernel $\zeta(x) = [1 - |x/h|]^+/h$ (where $[x]^+$ is x if $x > 0$ and 0 otherwise).

One can notice that the size of $\mathbf{M}$, representing the discretized source-to-flow-through linear application, is the size of the body, which is much less than the size of the computational box (here 3428 compared to 549153). Moreover, this matrix is not normal, as expected, and the distribution of singular values is still grouped, following a bi-Gaussian regression law, as shown on figure 12. This gives the expected good conditioning of the source-to-flow-through application, found to be $\mathrm{Cond}\mathbf{M} = 16.42$.

As shown on figure 12, it is then easy to compute potential fields around the aircraft geometry, here plotted for various pitch and roll angles, that is to say for several directions of vector $\mathbf{K}$.

5 Coupling with Vortex-in-Cell methods and practical considerations

Vortex method is used to descritize in a Lagrangian way the convective sub-step for Navier-Stokes equation (9), equivalent to the Euler equations for smooth geometry. The resulting dynamical system is given by equation (10), for which the velocity field is computed by equations (15,16) solved in practice by FISHPACK [33]. This dynamical system is solved by RK2 method, grid/particle interpolation and particle remeshing are performed by standard interpolation techniques for vortex methods [9,8,23], here with the kernel M_4'. This leads to a globally second order scheme in time and space, without any stability condition depending on time and space steps, due to its Lagrangian feature, thus allowing in practice the use of large time steps.

The initial vorticity field is then given by :

$$\boldsymbol{\omega}(t=0,\boldsymbol{\xi}) = \sum_{i=1,2} \mathrm{curl}(\mathcal{J}_i(\boldsymbol{\xi})\mathbf{e}_z) + \mathcal{B}_i(\boldsymbol{\xi})\mathbf{e}_z - \epsilon\mathcal{P}_i(\boldsymbol{\xi})\mathbf{e}_x \tag{55}$$

where $\boldsymbol{\xi}^{\|} = (\xi_x, \xi_y, 0)$ is the projection on plane x, y. The Batchelor vortices, their core-jets generating swirl, and streamwise pertubation are respectively given by :

$$\mathcal{B}_i(\boldsymbol{\xi}) = \frac{\Gamma_0}{2\pi\sigma^2} e^{|\boldsymbol{\xi}^{\|}-\boldsymbol{\zeta}_i|^2/2\sigma^2},\ \mathcal{J}_i(\boldsymbol{\xi}) = V_i e^{|\boldsymbol{\xi}^{\|}-\boldsymbol{\zeta}_i|^2/2\tau^2},\ \mathcal{P}_i(\boldsymbol{\xi}) = \frac{1}{\tau^2} e^{|\boldsymbol{\xi}^{\|}-\boldsymbol{\zeta}_i|^2/2\tau^2} \tag{56}$$

It is noticeable, as mentioned above, that the immersed boundary technique involves only one more Poisson equation in the velocity computation than standard vortex methods in a uniform Cartesian box without body. The computational cost of a velocity computation in a VIC algorithm using this immersed boundary technique is then $4/3$ of a velocity computation in an empty Cartesian box (that is to say without body), which has already been proved to behave linearly with respect to the number of particles, with a theoretical evolution scaling as $\mathcal{O}(n \log n)$.

On the one hand, interaction between vortex and the aircraft geometry of section 4.5 is considered. Parameters are $\boldsymbol{\zeta}_1 = (1, 0.7, 0)$, $\boldsymbol{\zeta}_2 = (1, -0.7, 0)$, $\Gamma_0 = 1$, $V_1 = V_2 = 0.1$, $\tau = \sigma = 0.1$ and $\epsilon = 0.07$. The large perturbations induce a strong interaction between this vortex and the aircraft, with non trivial dynamics of the vorticity field, as shown on figure 13. The effective residual penetration is the numerical zero at all iterations, as shown on middle picture of figure 12. Less than 12 minutes have been required to perform the global computation on a sequential

Opteron 245 processor, involving 200 time steps and $5.5\,10^5$ grid points, with a second order accuracy in time and space.

On the other hand, one considers a hollow mechanical structure, here a bridge, inside and around which vorticity evolves. Formula (55) is used again as initial condition, with the parameters $V_1 = 0.5$, $V_2 = 0$, $\epsilon = 0$ and $\boldsymbol{\zeta}_1 = (25, 0, 0)$, $\Gamma_0 = 10$, $\tau = \sigma = 4$. The computational box is $\Omega = [-40, 40, -40, 40, -160, 160]$ where z is the axis aligned to the road over the bridge, discretized by $2^7 \times 2^7 \times 2^9$ cells. Solution is computed incrementally up to $T = 300$ by steps of $\delta t = 0.1$. The isovorticity surface at time $t = 28$ is displayed on figure 13, as well as spectrum of the source-to-flow-through application associated to this geometry, which is shown to be still well-grouped. Circulation is well conserved in its three components with a maximum error lower than 0.6%, and 0.22% in the vortex main direction.

6 Conclusion

The technique of immersed boundaries, involving singular sheets of sources, has been revisited, focusing on its implicit formulation in order to satisfy no-through-flow boundary condition on a body of arbitrary geometry.

In the context of vorticity formulation, taking into account body geometry leads to invert the source-to-flow-through linear application, or at least to solve the associated linear system. Essential properties of this application have been given, such as its non-normality with respect to the standard scalar product, its good conditioning and its ability to be used easily, even for complex geometries.

Two kinds of discretizations have been investigated : a description of the immersed surface as the graph of a function on the one hand, and a set on points and normals dispatched on neighbour grid points on the other hand. The second technique is more general and easier to handle at an engineering level, without noticeable discrepancy on the conditioning. An explicit preconditioning technique has also been described, provided at no computational cost.

The size of such linear systems is the number of discretization points of the surface, which means much smaller than the number of grid points. For discretization of moderate size, direct inversion of the source-to-flow-through matrix is possible. This makes the method valid for much refined surfaces, using subgrid multiscale modelling techniques (such as [3]).

Coupling this method to Euler or Navier-Stokes solvers in their vorticity formulation has been shown to be natural, especially the coupling with vortex methods, which allows to manage three-dimensional flow computations in complex geometries at a computational cost close to the cost in simple Cartesian box.

References

[1] J. Adams, *MUDPACK: Multigrid Fortran Software for the Efficient Solution of Linear Elliptic Partial Differential Equations*, Appl. Math. Comp., 34 (1989), 113–146.

[2] J. T. Beale and A. Majda, *Rates of convergence for viscous splitting of the Navier-Stokes equations*, Math. Comp., 37 (1981), 243–259.

[3] M. Bergdorf, G.-H. Cottet and P. Koumoutsakos, *Multilevel Adaptive Particle Methods for Convection-Diffusion Equations*, SIAM Multiscale Modeling and Simulation, 4 (2005), 328–357.

[4] A. J. Chorin, *Numerical study of slightly viscous flow*, J. Fluid Mech., 57 (1973), 785–796.

[5] J. P. Christiansen, *Numerical solution of hydrodynamics by the method of point vortices*, J. Comp. Phys. 13 (1973), 363–379.

[6] P. Collela, D. Graves, B. Keen and D. Modiano, *A Cartesian grid embedded boundary method for hyperbolic conservation laws*, J. Comp. Phys., 211 (2006), 347–366.

[7] R. Cortez, *The Blob Projection Method for Immersed Boundary Problems*, J. Comp. Phys., 161 (2000), 428–453.

[8] G.-H. Cottet and P. D. Koumoutsakos, *Vortex methods, theory and practice*, Cambridge University Press (2000).

[9] G.-H. Cottet and P. Poncet, *Advances in DNS of three-dimensional wall-bounded flows by Particle in Cell methods*, J. Comp. Phys., 193 (2003), 136–158.

[10] B. Couët, O. Buneman and A. Leonard, *Simulation of three-dimensional flows with vortex-in-cell methods*, J. Comp. Phys. 39 (1981), 305.

[11] D. G. Crighton, *The Kutta condition in unsteady flow*, Ann. Rev. Fluid Mech. 17 (1985), 411-445.

[12] J. Eldredge, *Numerical simulation of the fluid dynamics of 2D rigid body motion with the vortex particle method*, J. Comput. Phys., 221(2), 626–648

[13] E. A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, *Combined immmersed-boundary finite-difference methods for three-dimensional complex flow simulations*, J. Comp. Phys., 161 (2000), 35–60.

[14] A. L. Fogelson and J. P. Keener, *Immersed interface methods for Neumann and related problems in two and three Dimensions*, SIAM J. Sci. Comp. 22:5 (2000), 1630-1654.

[15] A. Gilmanov, F. Sotiropoulos and E. Balaras, *A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids*, J. Comp. Phys., 191 (2003), 660—669

[16] L. Greengard and V. Rokhlin, *A fast algorithm for particle simulation*, J. Comp. Phys., 73 (1987), 325–348.

[17] P. Koumoutsakos and A. Leonard, *Improved boundary integral method for inviscid boundary condition applications*, AIAA J. 31:2 (1992), 401–404.

[18] P. Koumoutsakos, A. Leonard, and F. Pepin, *Boundary conditions for viscous vortex methods*, J. Comput. Phys. 113 (1994), 52.

[19] P. Koumoutsakos and A. Leonard, *High-resolution simulations of the flow around an impulsively started cylinder using vortex methods*, J. Fluid Mech. 296 (1995), 1–38.

[20] M. C. Lai and Z. Li, *A remark on jump conditions for the three-dimensional Navier-Stokes equations involving an immersed moving membrane,* Appl. Math. Lett. 14 (2001), 149–154.

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

[22] Z. Li and K. Ito, *The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*, SIAM Frontiers in Applied Mathematics 33 (2006).

[23] J. J. Monaghan, *Extrapolating B-Splines for interpolation*, J. Comp. Phys, 60 (1985), 253–262

[24] J. C. Nédélec, *Acoustic and Electromagnetic Equations. Integral Representations for Harmonic Problems*, Applied Mathematics Science 144, Springer Verlag, New York, 2001.

[25] C. S. Peskin, *Flow patterns around heart valves : A numerical method*, J. Comp. Phys. 10 (1972), 252-271.

[26] C.S. Peskin, *The immersed boundary method*, Acta Numerica (2002), 1–39.

[27] P. Poncet, *Topological aspects of the three-dimensional wake behind rotary oscillating circular cylinders.*, J. Fluid Mech., 517 (2004), 27–53.

[28] P. Poncet, *Finite difference stencils based on particle strength exchange schemes for improvement of vortex methods*, J. Turbulence, 7 (2006), 1–24.

[29] P. Poncet, *Analysis of direct three-dimensional parabolic panel methods*, SIAM J. Numer. Anal., 45:6 (2007), 2259–2297.

[30] P. Poncet, R. Hildebrand, G-H. Cottet and P. Koumoutsakos, *Spatially distributed control for optimal drag reduction in cylinder wakes*, J. Fluid Mech. 599 (2008), 111–120.

[31] E. Sonnendrücker, J. Roche, P. Bertrand and A. Ghizzo, *The semi-lagrangian method for the numerical resolution of Vlasov equations*, J. Comp. Phys., 149:2 (1998), 201–220.

[32] P. Schwartz, M. Barad, P. Colella and T. Ligocki, *A cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions*, J. Comp. Phys., 211 (2006), 531–550.

[33] R. Sweet, *A Parallel and Vector Variant of the Cyclic Reduction Algorithm*, SIAM J. Sci. and Stat. Comp., 9 (1988), 761-766.

[34] Y.-H. Tseng, J. H. Ferziger, *A ghost-cell immersed boundary method for flow in complex geometry*, J. Comp. Phys., 192 (2003), 593–623.

[35] M. Uhlmann, *An immersed boundary method with direct forcing for the simulation of particulate flows*, J. Comp. Phys., 209 (2005), 448-476.

[36] P. Ploumhans and G. S. Winckelmans, *Vortex Methods for High-Resolution Simulations of Viscous Flow Past Bluff Bodies of General Geometry* J. Comp. Phys., 165 (2000), 354–406.

[37] S. Xu, Z. J. Wang, *Systematic derivation of jump conditions for the immersed interface method in three-dimensional flow simulation.*, SIAM J. Sci. Comp., 27 (2006), 1948–1980.

[38] S. Xu, *The immersed interface method for simulating prescribed motion of rigid objects in an incompressible viscous flow*, J. Comp. Phys., 227 (2008), 5045–5071

Figures

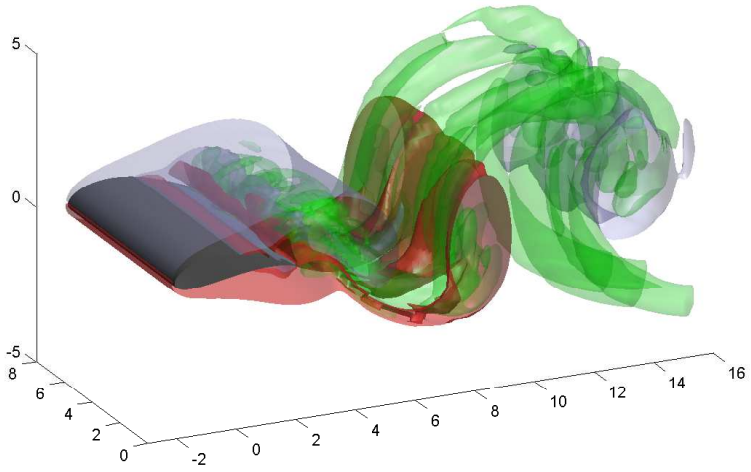


Fig. 1. Case of a viscous wake behind an airfoil, for which adding vorticity at the trailing edge is unneeded when solving the convective part of the Navier-Stokes equations.

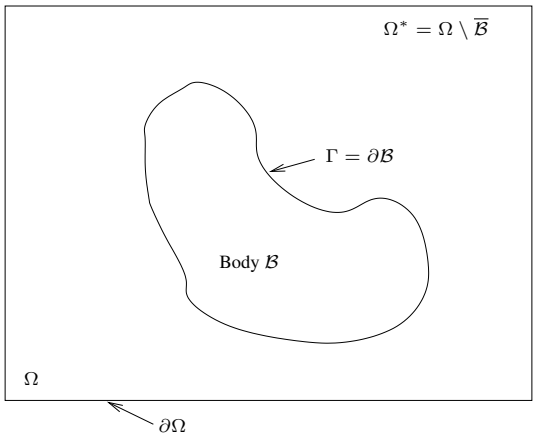


Fig. 2. Different domains and notations involved in definition of immersed boundaries.

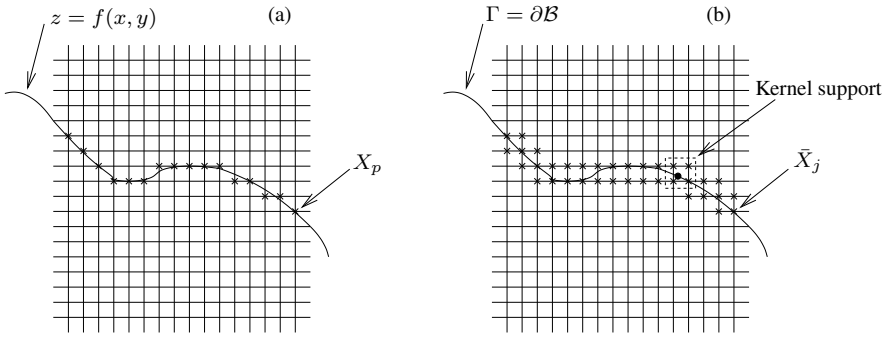


Fig. 3. Two different ways to immerse a surface : Surface Γ is defined as the graph of a function $z = f(x, y)$ (a), or surface is defined by a sequence of points spread over the grid by means of a convolution with a kernel whose support size scale as the grid step (b). Case (b) shows a point on the surface Γ denoted by $\bullet$ and spread over the 4 neighbour points on the grid, involving a kernel which support size is $h \times h$.

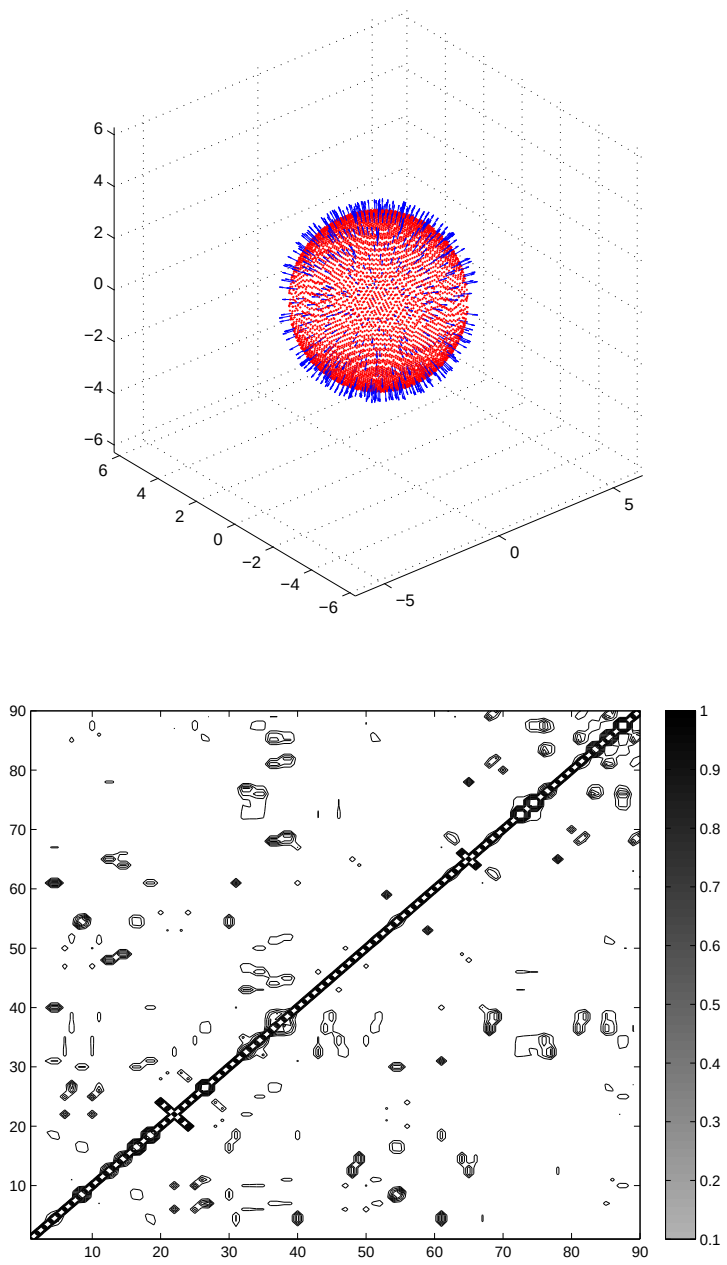


Fig. 4. Non-normality of source-to-flow-through linear application for a spherical body (displayed with its normal bundle on top picture), and Hermitian product of eigenvectors (bottom picture).

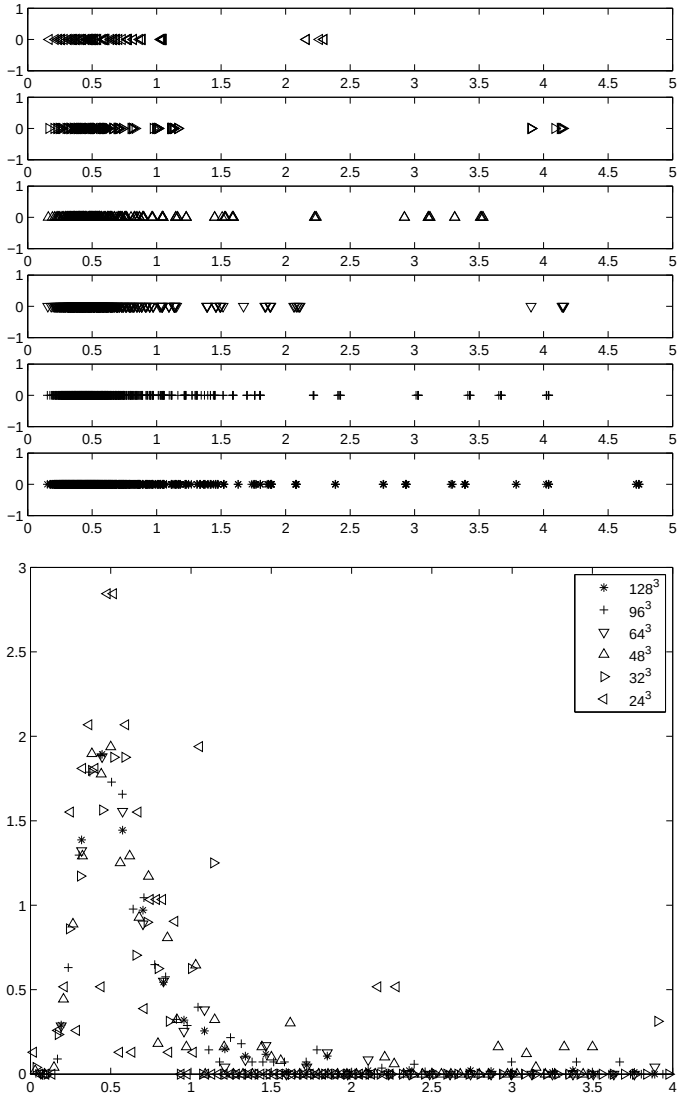


Fig. 5. Singular value decomposition of source-to-flow-through matrix for a spherical geometry and for several grid refinements (top pictures), and singular value distributions (bottom picture).

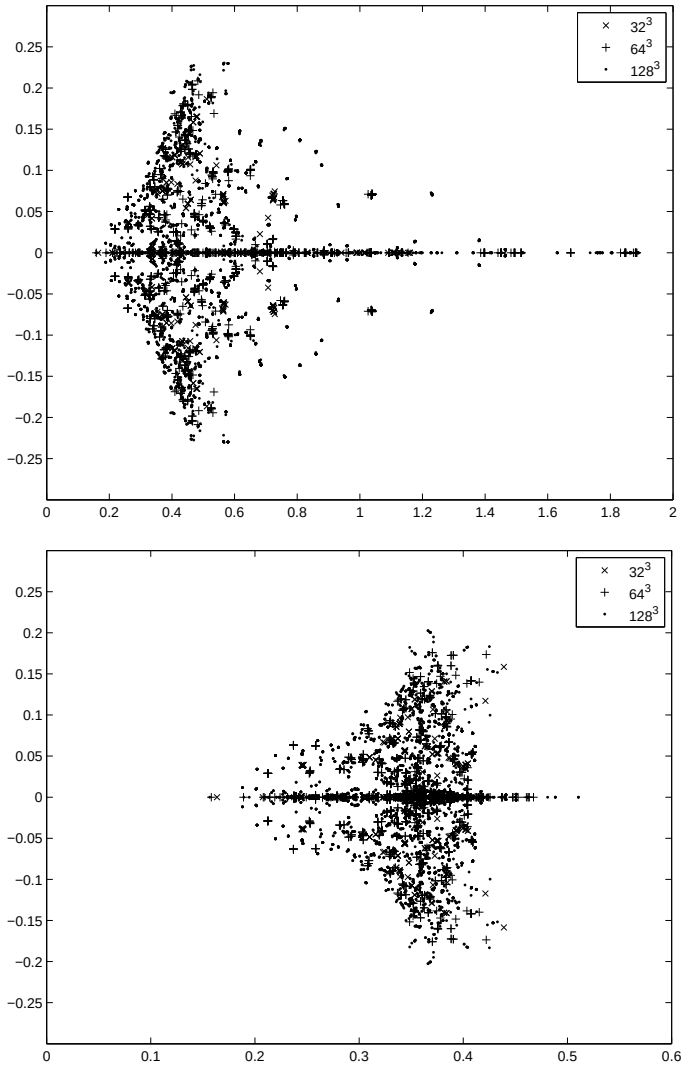


Fig. 6. Spectra of regular (to the left) and iso-weighted (to the right) source-to-flow-through applications for spherical body geometry.

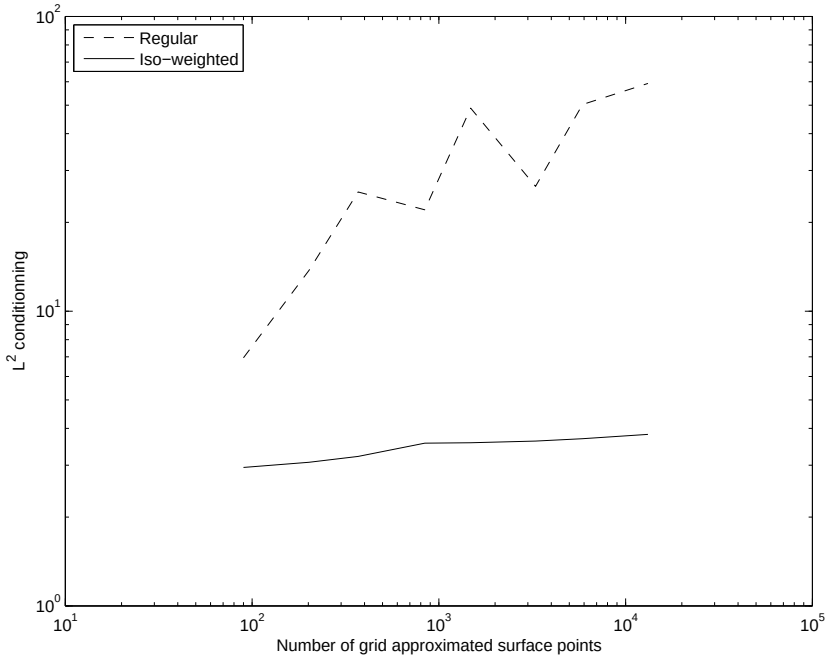


Fig. 7. Conditioning of regular (- -) and iso-weighted (—) source-to-flow-through applications for a spherical body with respect to the number of surface points.

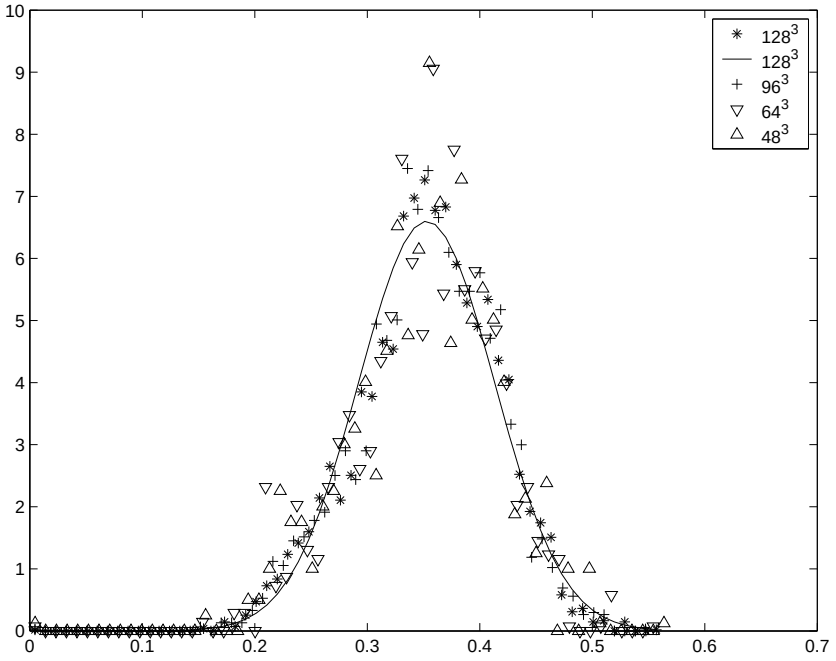


Fig. 8. Distribution of 5882 Singular Values over 60 classes of source-to-flow-through application for a spherical body immersed in a grid with various refinements, and its best Gaussian fit in the grid 128^3.

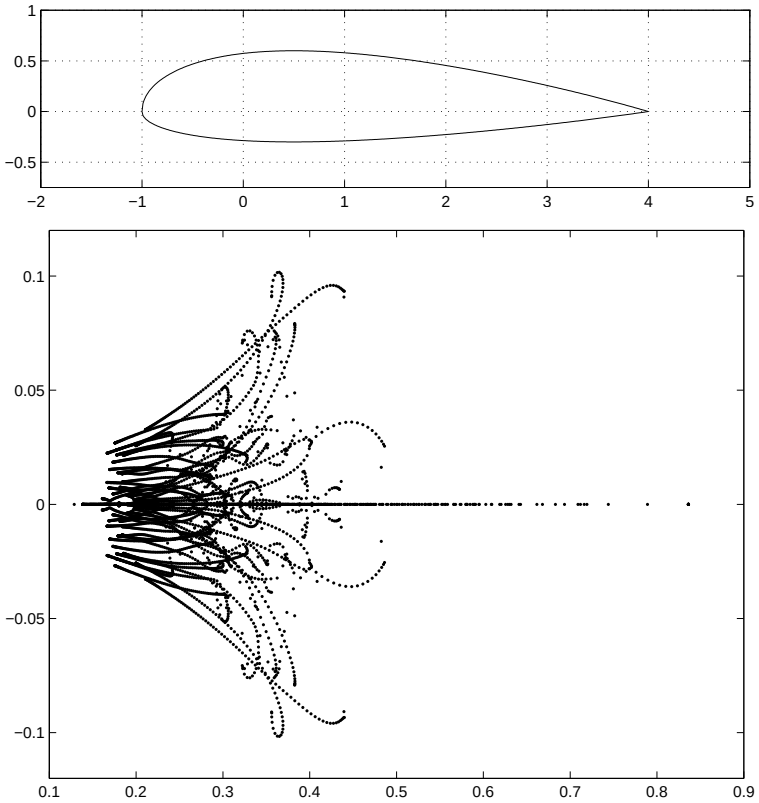


Fig. 9. Complex spectrum of source-to-flow-through application (bottom picture) for a NACA2412 airfoil (top picture) in a 96^3 grid.

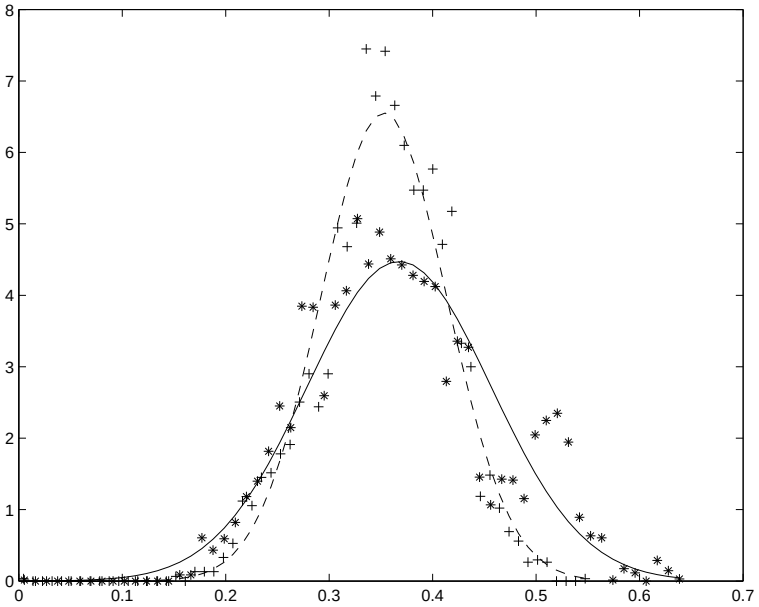


Fig. 10. Singular Values distribution over 60 classes of source-to-flow-through application for a spherical body immersed in a grid 96^3, in the cases of graph defined surface ($+$) and interpolated surfaces ($*$), and their best Gaussian fits (respectively $--$ and —).

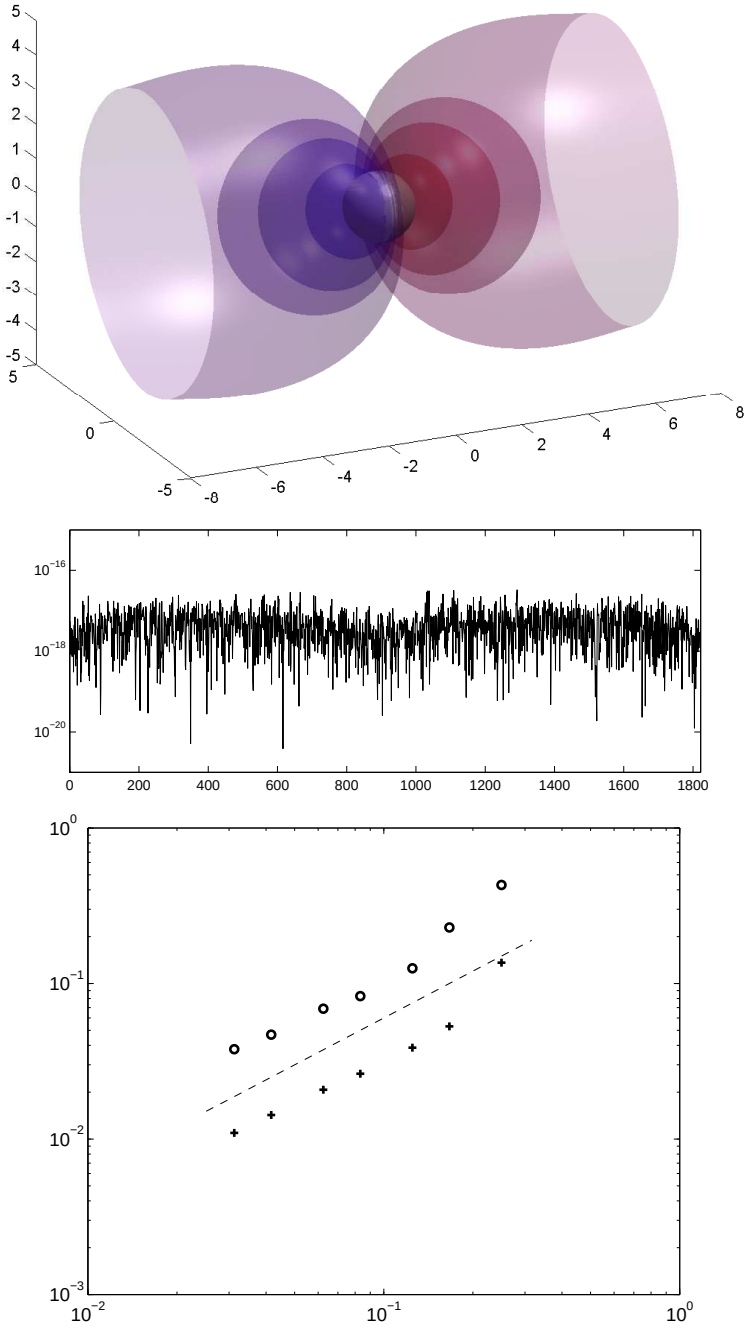


Fig. 11. Potential flow around a sphere with far field velocity (displayed on top picture), and its residual flow-through (middle picture) close to numerical zero. Consistency of the potential computation ($+$, bottom picture), and convergence of resulting velocity ($\circ$, bottom picture), with respect to the number of grid points (first order line is plotted as $--$).

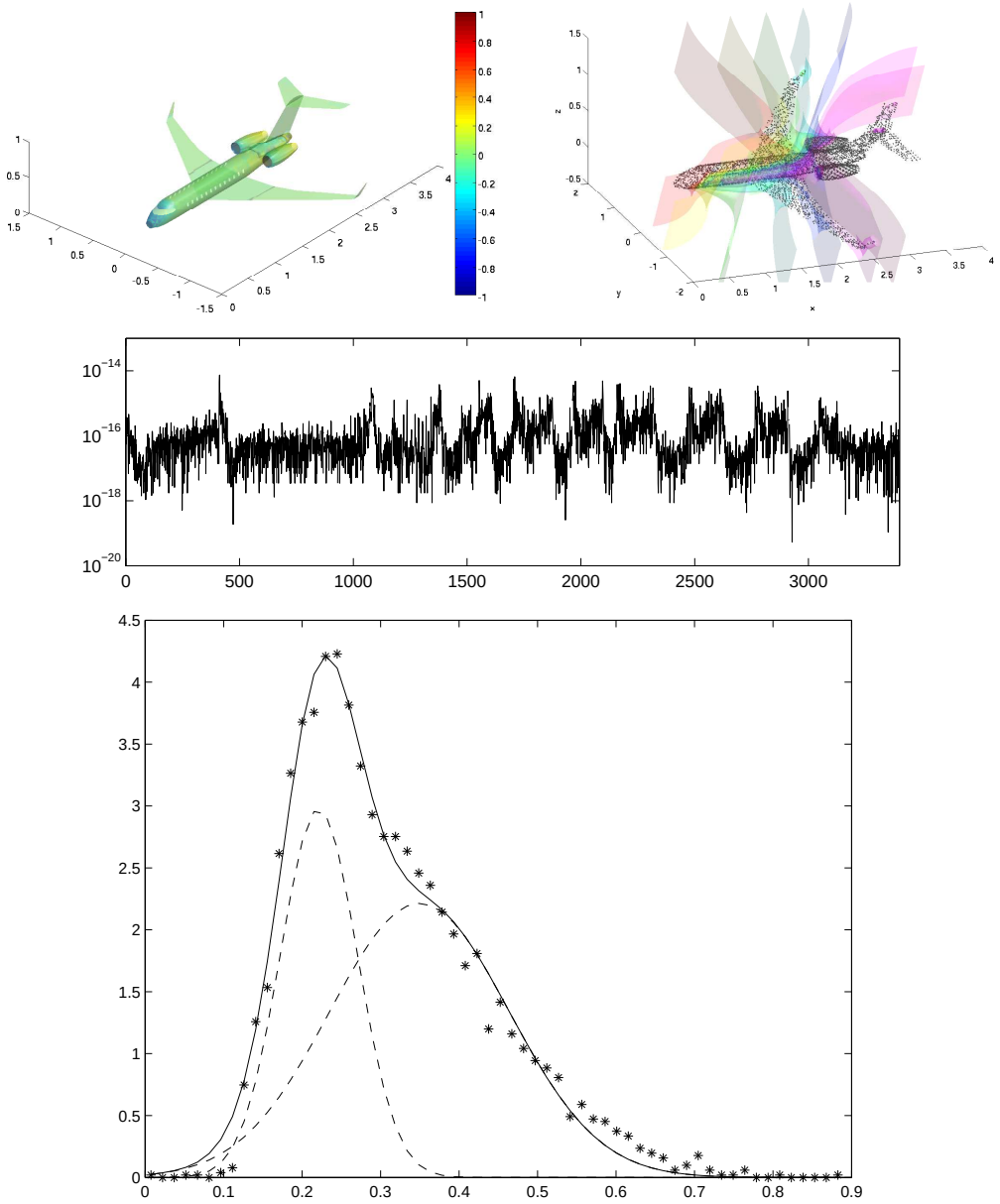
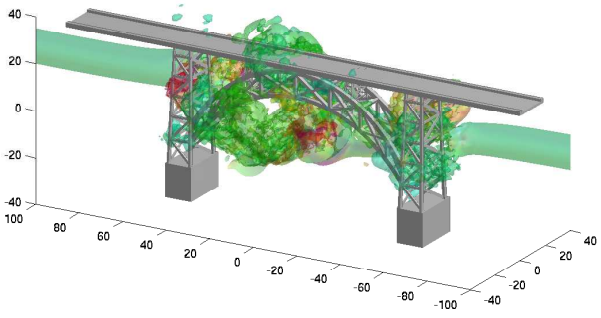


Fig. 12. Aircraft discretized by 5617 points spread over 3428 grid points (top left picture, colored by streamwise component of normal field one aims at cancelling). Top right picture shows isosurfaces of potential ϕ for the aircraft geometry immersed in a $128 \times 128 \times 32$ box for a pitch angle 0^o and roll angle 10^o. The resulting absolute error on penetration condition is displayed on middle picture. Bottom picture shows the distribution of the 3428 singular values of source-to-flow-through application, dispatched over 60 classes ($*$), and its best bi-Gaussian fits (—) with its two Gaussian components ($--$).



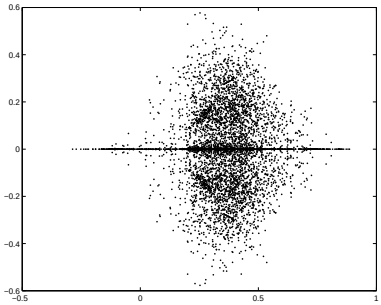


Fig. 13. Vorticity isovalue at level $\|\boldsymbol{\omega}\| = 0.05$ at $t = 17$ for the aircraft (top picture) and $t = 28$ for the hollow bridge geometry (bottom left picture), and spectra of its related iso-weighted source-to-flow-through application (bottom right picture).

A Self-adjointness of source-to-flow-through

The goal of the present annex is to show that the Sobolev space $\mathbb{H}^{-1/2}(\Omega)$ is the natural space where the source-to-flow-through linear application is self-adjoint, which suggests that defining it in other spaces would lead to a lack of self-adjointness and even normality.

One introduces the product of duality as a bilinear form over a topological space, in practice $\mathbb{H}^s(\Gamma) \times \mathbb{H}^{-s}(\Gamma)$, which can be written as follows for regular generalized function :

$$(f, v)_{s,-s} = \int_\Gamma f(x)v(x)\mathrm{d}\sigma(x) \tag{A.1}$$

One also considers the Neumann-to-Dirichlet operator :

$$\begin{array}{rcl} \mathcal{R} : \mathcal{V} \subset \mathbb{H}^s(\Gamma) & \longrightarrow & \mathbb{H}^{s+1}(\Gamma) \\ g & \longmapsto & \phi_1|_\Gamma \end{array} \tag{A.2}$$

such as ϕ_1 is solution of equation (25).

The adjoint takes its sense by means of the scalar product in $\mathbb{H}^{-1/2}(\Omega)$, defined using the product of duality and the Neumann-to-Dirichlet operator for $s = -1/2$:

$$\langle f, g\rangle_{\mathbb{H}^{-1/2}(\Gamma)} = (\mathcal{R}f, g)_{1/2,-1/2} = \int_\Gamma \mathcal{R}f(x)g(x)\mathrm{d}\sigma(x) \tag{A.3}$$

which is commutative (see [24]).

One considers the application that associates to $\mu \in \mathbb{H}^{-1/2}(\Gamma)$ the solution $\phi \in \mathbb{H}^1(\Omega)$ of the following harmonic problem with a no-jump-through condition :

$$\left\{ \begin{array}{l} -\Delta\phi = T_\mu \text{ on } \Omega \\ \quad \mathcal{L}\phi = 0 \text{ on } \partial\Omega \\ \quad [\phi]_\Gamma = 0 \text{ on } \Gamma \end{array} \right. \tag{A.4}$$

with $T_\mu = (\mu, \cdot)_{-1/2,1/2}$. As shown in section 3.1, the definition of Λ using the jump relation (32) allows to write $\mathcal{R}\Lambda$ as a source-to-Dirichlet operator :

$$\begin{array}{rcl} \mathcal{R}\Lambda : \mathcal{V} \subset \mathbb{H}^{-1/2}(\Gamma) & \longrightarrow & \mathbb{H}^{1/2}(\Gamma) \\ \mu & \longmapsto & \mathcal{R}\,\Lambda(\mu) = \mathcal{R}\left(\left.\dfrac{\partial\phi}{\partial\mathbf{n}}\right|_\Gamma\right) = \phi|_\Gamma \end{array} \tag{A.5}$$

Let ϕ and $\boldsymbol{\psi}$ be respectively the solutions of problem (A.4) for right hand side

densities μ and ν. This leads the following calculus of adjoint linear application Λ^* :

$$\langle \Lambda^*(\nu), \mu \rangle_{\mathbb{H}^{-1/2}(\Gamma)} = \langle \Lambda(\mu), \nu \rangle_{\mathbb{H}^{-1/2}(\Gamma)} = T_\nu(\phi) \tag{A.6}$$

with

$$T_\nu(\phi) = \int_\Omega -\Delta \boldsymbol{\psi}(x)\phi(x)\,\mathrm{d}\lambda(x) = \int_\Omega \nabla \boldsymbol{\psi}(x) \cdot \nabla \phi(x)\,\mathrm{d}\lambda(x) - \int_{\partial\Omega} \phi(x) n \cdot \nabla \boldsymbol{\psi}(x)\,\mathrm{d}\sigma(x)$$

Since far-field conditions are homogeneous, the integral at the right of the expression above vanishes and the equation above is commutative in ϕ and $\boldsymbol{\psi}$. Consequently, one gets

$$\langle \Lambda^*(\nu), \mu \rangle_{\mathbb{H}^{-1/2}(\Gamma)} = \langle \Lambda(\mu), \nu \rangle_{\mathbb{H}^{-1/2}(\Gamma)} = T_\nu(\phi) = T_\mu(\boldsymbol{\psi}) = \langle \Lambda(\nu), \mu \rangle_{\mathbb{H}^{-1/2}(\Gamma)}$$

for aln couple of densities $(\mu, \nu) \in \mathbb{H}^{-1/2}(\Gamma)^2$, which finishes the proof that source-to-flow-through application Λ is self-adjoint in $\mathbb{H}^{-1/2}(\Gamma)$. This suggests that $\mathbb{L}^2$ is not the natural space to have the source-to-flow-through application self-adjoint.