Efficient FMM accelerated vortex methods in three dimensions via the Lamb-Helmholtz decomposition*

Nail A. Gumerov[†] and Ramani Duraiswami[‡]
Institute for Advanced Computer Studies
University of Maryland, College Park

January 25, 2012

Abstract

Vortex-element methods are often used to efficiently simulate incompressible flows using Lagrangian techniques. Use of the FMM (Fast Multipole Method) allows considerable speed up of both velocity evaluation and vorticity evolution terms in these methods. Both equations require field evaluation of constrained (divergence free) vector valued quantities (velocity, vorticity) and cross terms from these. These are usually evaluated by performing several FMM accelerated sums of scalar harmonic functions.

We present a formulation of the vortex methods based on the Lamb-Helmholtz decomposition of the velocity in terms of two scalar potentials. In its original form, this decomposition is not invariant with respect to translation, violating a key requirement for the FMM. One of the key contributions of this paper is a theory for translation for this representation. The translation theory is developed by introducing "conversion" operators, which enable the representation to be restored in an arbitrary reference frame. Using this form, extremely efficient vortex element computations can be made, which need evaluation of just two scalar harmonic FMM sums for evaluating the velocity and vorticity evolution terms. Details of the decomposition, translation and conversion formulae, and sample numerical results are presented.

1 Introduction

Vortex methods solve the incompressible Navier Stokes equation in the velocity–vorticity form with Lagrangian discretization. Since vortex particles are initially placed only in the region of finite vorticity and can convect along with the flow, these methods provide an optimized spatial discretization. Consider an incompressible flow generated by a set of N vortex elements, characterized by coordinates of the centers (sources) $\mathbf{x}_i$ and constant strength vector $\boldsymbol{\omega}_i$, $i = 1, ..., N$. Each element centered at location $\mathbf{x}_i$ produces an elementary velocity field $\mathbf{v}_i(\mathbf{y})$ according to the Biot-Savart law, and the total velocity field can be computed as a superposition of such elementary fields (e.g., see [1]):

$$\mathbf{v}(\mathbf{y}) = \sum_{i=1}^{N} \mathbf{v}_i(\mathbf{y}), \quad \mathbf{v}_i(\mathbf{y}) = \frac{\boldsymbol{\omega}_i \times (\mathbf{y} - \mathbf{x}_i)}{|\mathbf{y} - \mathbf{x}_i|^3} = \nabla \times \frac{\boldsymbol{\omega}_i}{|\mathbf{y} - \mathbf{x}_i|}. \tag{1}$$

*University of Maryland Department of Computer Science Technical Report CS-TR-5002; University of Maryland Institute for Advanced Computer Studies Technical Report UMIACS-TR-2012-02

†Also at Fantalgo, LLC., Elkridge, MD 21075; E-mail: gumerov@umiacs.umd.edu, info@fantalgo.com web: http://www.umiacs.umd.edu/users/gumerov

‡Also Department of Computer Science, and at Fantalgo, LLC. E-mail:ramani@umiacs.umd.edu, web: http://www.umiacs.umd.edu/users/ramani

In practice this field needs to evaluated at M evaluation points, $\mathbf{y}_j$, which has $O(MN)$ cost. The Biot-Savart kernel is composed of a vector of dipole solutions of the Laplace equation. It is well known that the Fast Multipole Method (FMM) can be used to evaluate such sums to any specified accuracy ϵ at a $O(N+M)$ reduced cost [6].

The vortex elements move with the flow. This motion also causes an evolution of the vortex field according to the vortex evolution equation. For inviscid flow, the evolution equations for the vortex positions and strengths respectively are

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i; \qquad \frac{d\boldsymbol{\omega}_i}{dt} = \mathbf{s}_i, \quad \mathbf{s}_i = \boldsymbol{\omega}_i \cdot \nabla \mathbf{v}. \tag{2}$$

Here the right hand side for the vortex strength is the so-called vortex stretching term, the evaluation of which requires an evaluation of the gradient of the velocity vector. The evolution equation for the vorticity can be modified to account for liquid viscosity. In the latter case the elementary velocity field in Eq. (1) can be modified using a smoothing kernel, $K(|\mathbf{y}-\mathbf{x}_i|\,;t)$,

$$\mathbf{v}_i\left(\mathbf{y};t\right) = \frac{\boldsymbol{\omega}_i \times (\mathbf{y}-\mathbf{x}_i)}{|\mathbf{y}-\mathbf{x}_i|^3} K(|\mathbf{y}-\mathbf{x}_i|\,;t), \quad K(r;t) = 1 + O(\epsilon), \quad r > a(t), \tag{3}$$

which has an effect only on the near field, $r \leqslant a$, where a is the radius of the vortex core, and $\epsilon \ll 1$ is the tolerance for approximation. This modification does not affect the far field, which is computed with the tolerance $O(\epsilon)$. The computation of the far field can still be accelerated by the FMM. In the sequel we do not specify the core function $K(r;t)$; several choices including the Gaussian and polynomial forms are discussed in the literature (see e.g., [15, 19, 20]), and there are several ways to speed up the local summation as well (e.g., [17]). Extensions to compressible flow are also possible [3].

The evolution equation is integrated using an appropriate time stepping scheme. The right hand side of this equation, also results in an N-body computation for the influence of particles in the far-field, with somewhat more complicated terms. As discussed above, the FMM for the vortex element method is closely related to the scalar FMM for sums of multipoles of the Laplace equation ("harmonic FMM"). In fact, it is possible to start with a program written for performing a harmonic FMM and appropriately modify it to create a fast vortex method software. A key issue in the implementations is the ratio of the amounts of time taken for computing a harmonic FMM, to the amount of time taken to compute the vector sum for the vortex velocities, and the vector sum for computing the right hand side of the vortex evolution equations.

In [20] the computations for evaluating the stretching term is shown to be about six times the evaluation of the velocity alone, while in [14] the cost of evaluating the velocity is shown to be a bit less than twice the cost of evaluating a single harmonic FMM sum. This suggests that in current implementations of FMM accelerated vortex methods the standard cost of evaluating a vortex element method computation (velocity+stretching) is an order of magnitude larger than a single harmonic FMM. For a similar problem, of evaluating sums of Stokeslets and Stresslets (elementary solutions of Stokes' equations) [18] show that the evaluation can be done with a cost of four to six harmonic FMM calls.

In this paper we develop an extremely efficient version of the FMM for vortex element methods, which achieve an evaluation of both the velocity and stretching term sums at a cost of only about two harmonic FMMs. This provides a substantial speed up of vortex element methods. Our basic tool is the use of the Lamb-Helmholtz decomposition [16] which allows the representation of the vector field in the form of two scalar potential fields. This form is however not invariant to translation, and cannot be used as is, within the context of an FMM summation algorithm. We develop conversion operators that allow this form to be translated. In some sense the method which we develop in this paper is similar to the method of translation of solutions of the biharmonic and Maxwell's equations that we developed previously in [11, 12] respectively.

In [11] it was shown that any solution of the biharmonic equation can be expressed as a combination of two solutions of the Laplace equations. However, when the biharmonic solution is expressed in this form, the translations cannot be done independently. Instead the two functions must be translated jointly, which can be handled relatively easily by the introduction of the concept of a sparse "conversion" operator. In fact given a fast-multipole method routine for the Laplace equation, we show there that using the conversion operators, it can be employed as a fast-multipole method routine for the biharmonic and polyharmonic equations. In [12] a translation theory for the Debye form of the solution of Maxwell's equation, in terms of two scalar potentials that satisfy the Helmholtz equation was presented. Here we present an analogous theory for the divergence constrained Helmholtz-Lamb potential decomposition of the velocity field, which are closely related to vorticity formulations, and show how they can be used in vortex element methods.

Section 2 of the paper introduces the problem and notation, and shows that the equations can be considered to be solutions of a divergence constrained vector Laplace equation. Section 3 introduces the translation theory developed for such equations. Section 4 shows how the new translation theory along with a program for performing fast multipole method accelerated sums for multipoles of the Laplace equation, can together be used to create a FMM accelerated vortex element solver. Together Sections 3 and 4 represent the main mathematical results of the paper. Section 5 presents the results of numerical testing, and shows that using the current form of the decomposition, substantial savings can be made with respect to state of the art FMM accelerated vortex element techniques. Section 6 concludes the paper.

2 Statement of the problem

We are given N vortex blobs of strength $\boldsymbol{\omega}_i$, $i = 1, ..., N$ located at points $\mathbf{x}_i$ and moving with the flow. The velocity field can be evaluated using either Eq. (1) or Eq. (3), which both have the same asymptotic far-field form. The evolution of the vortex positions and the vortex strengths is given by Eqs. (2). We note now that at $\mathbf{y} \neq \mathbf{x}_i$, $i = 1, ..., N$, the velocity field $\mathbf{v}(\mathbf{y})$ satisfies divergence constrained vector Laplace equation (DCVLE)

$$\nabla^2 \mathbf{v} = \mathbf{0}, \quad \nabla \cdot \mathbf{v} = 0. \tag{4}$$

In the case when the divergence of the field is not constrained, each Cartesian component of the velocity is an independent harmonic function, and if the FMM for the Laplace equation is available this can be solved at a cost of execution of 3 harmonic FMMs. The divergence constraint, however, reduces the degree of freedom for solutions by 1, and, in fact, only two harmonic scalar potentials, ϕ and χ, are necessary to describe the total field inside or outside a sphere centered at the origin of the reference frame

$$\mathbf{v}(\mathbf{r}) = \nabla \phi(\mathbf{r}) + \nabla \times (\mathbf{r} \chi(\mathbf{r})), \quad \nabla^2 \phi = 0, \quad \nabla^2 \chi = 0. \tag{5}$$

This decomposition can be treated as a general Helmholtz decomposition of an arbitrary vector field, with a particular form of the vector potential. Presumably, this form is due to Lamb [16], who found a general solution for the Stokes equations in spherical coordinates, and we refer to this as the Lamb-Helmholtz decomposition. Indeed, Eq. (4) can be treated as the Stokes equations with zero pressure, and the Lamb solution, where the pressure is zeroed out provides the form (5).

The major problem with the use of this representation in the FMM is providing a form that is invariant to translation. This can be done by using conversion operators, as was presented for the biharmonic and Maxwell equations in our previous publications [11, 12]. In this paper we develop such operators and present a complete translation theory for the DCVLE. We also use this theory with existing harmonic FMM software to compute both the velocity field induced by N vortex elements (1) and for the evaluation of the velocity gradient for an expense of execution of only 2 real-valued harmonic FMMs. In fact, as was shown in [11], is also possible to use just one FMM for a complex valued harmonic functions (for real solutions of Eq. (3)) to solve the problem, which may also have certain computational advantages.

It is not difficult to show that the DCVLE appears as a general equation for reconstruction of an arbitrary vector field from given curl, $\boldsymbol{\omega}(\mathbf{r})$, and divergence, $q(\mathbf{r})$,

$$\nabla \times \mathbf{v} = \omega, \quad \nabla \cdot \mathbf{v} = q. \tag{6}$$

Indeed, solution of these equations in free space can be written in the form (e.g. see [1]):

$$\mathbf{v}(\mathbf{y}) = -\nabla_{\mathbf{y}} \int_V \frac{q(\mathbf{x})}{4\pi |\mathbf{y}-\mathbf{x}|} dV(\mathbf{x}) + \nabla_{\mathbf{y}} \times \int_V \frac{\boldsymbol{\omega}(\mathbf{x})}{4\pi |\mathbf{y}-\mathbf{x}|} dV(\mathbf{x}). \tag{7}$$

Subdividing the space to the vicinity of evaluation point $\mathbf{y}$ (near field) and the domain outside this neighborhood (far field) and discretizing the integrals for the far field using quadratures with weights w_i and nodes $\mathbf{x}_i$, we obtain for the far field contribution

$$\begin{aligned} \mathbf{v}_f(\mathbf{y}) &= \sum_i \mathbf{v}_{fi}(\mathbf{y}), \quad \mathbf{v}_{fi}(\mathbf{y}) = -\nabla \frac{q_i}{|\mathbf{y}-\mathbf{x}_i|} + \nabla \times \frac{\boldsymbol{\omega}_i}{|\mathbf{y}-\mathbf{x}_i|}, \\ q_i &= \frac{w_i q(\mathbf{x}_i)}{4\pi}, \quad \omega_i = \frac{w_i \boldsymbol{\omega}(\mathbf{x}_i)}{4\pi}. \end{aligned} \tag{8}$$

Hence, the far field satisfies Eq. (4) for which decomposition (5) can be used and just an addition to potential ϕ due to a given monopole distribution $q(\mathbf{x})$ provides solution for a general case. As mentioned, in the present paper we do not address computation of the near field, which can be done locally, e.g. using appropriate smoothing kernels. Note that solution (7) of Eq. (6) is unique up to a gradient of a harmonic function Φ, which should be found from the boundary conditions. Such functions for a given boundary can be added naturally to $\phi(\mathbf{r})$ in Eq. (5).

Equations (6) with $q \neq 0$ appear, e.g. in vortex methods for compressible flows. An example of equations (for 2D) can be found in [3], which can be appropriately modified for 3D. In terms of computational complexity, besides the velocity field and stretching term computations, also contraction of the velocity gradient tensor, $\beta = \nabla \mathbf{v} : \nabla \mathbf{v}$ should be computed in this case. This term can be computed simultaneously with computation of the vortex stretching term. Thus the cost in these extended cases should remain the same.

3 Translation theory for DCVLE

3.1 Basic translation and differential operators

Translation operator: A generic translation or shift operator $\mathcal{T}(\mathbf{t})$, where $\mathbf{t}$ is a constant termed the translation vector, acts on some scalar valued function $\phi(\mathbf{r})$, to produce a new function $\widehat{\phi}(\mathbf{r})$ (the translate), whose values coincide with $\phi(\mathbf{r})$ at shifted values of the argument

$$\widehat{\phi}(\mathbf{r}) = \mathcal{T}(\mathbf{t})[\phi(\mathbf{r})] = \mathcal{T}_{\mathbf{t}}\phi, \quad \widehat{\phi}(\mathbf{r}) = \phi(\mathbf{r}+\mathbf{t}), \quad \mathbf{r}, \mathbf{t} \in \mathbb{R}^3. \tag{9}$$

This operator is linear, and if $\phi(\mathbf{r})$ satisfies the Laplace equation, its translate $\widehat{\phi}(\mathbf{r})$ satisfies the same equation in the shifted domain.

Elementary directional differential operators: We introduce the following notation for differential operators which appear in derivations:

$$\mathcal{D}_{\mathbf{r}} = \mathbf{r} \cdot \nabla, \quad \mathcal{D}_{\mathbf{t}} = \mathbf{t} \cdot \nabla, \quad \mathcal{D}_{\mathbf{r}\times\mathbf{t}} = (\mathbf{r} \times \mathbf{t}) \cdot \nabla. \tag{10}$$

It is not difficult to prove that if $\phi(\mathbf{r})$ is a harmonic function in some domain, then $\mathcal{D}_{\mathbf{r}}\phi$, $\mathcal{D}_{\mathbf{t}}\phi$, and $\mathcal{D}_{\mathbf{r}\times\mathbf{t}}\phi$ are also harmonic functions in the same domain. Note also that operators $\mathcal{D}_{\mathbf{t}}$ and $\mathcal{D}_{\mathbf{r}\times\mathbf{t}}$ are related to infinitesimal translation in direction of vector $\mathbf{t}$ and infinitesimal rotation with the rotation axis $\mathbf{t}$.

3.2 Conversion operators for the DCVLE

Consider now translation of the vector function $\mathbf{v}(\mathbf{r})$ given in form (5) using notation (9) for the translated functions. We have

$$\begin{aligned}\widehat{\mathbf{v}}(\mathbf{r}) &= \mathbf{v}(\mathbf{r}+\mathbf{t}) = \nabla\phi(\mathbf{r}+\mathbf{t}) + \nabla\times((\mathbf{r}+\mathbf{t})\,\chi(\mathbf{r}+\mathbf{t})) \\ &= \nabla\widehat{\phi}(\mathbf{r}) + \nabla\times((\mathbf{r}+\mathbf{t})\,\widehat{\chi}(\mathbf{r})) = \nabla\widehat{\phi}(\mathbf{r}) + \nabla\times(\mathbf{r}\widehat{\chi}(\mathbf{r})) + \nabla\times(\mathbf{t}\widehat{\chi}(\mathbf{r}))\,.\end{aligned} \tag{11}$$

It is clear that the obtained form of expansion of the translated function does not coincide with form (5). Our goal is to find harmonic functions $\widetilde{\phi}$ and $\widetilde{\chi}$, which provide a representation $\widehat{\mathbf{v}}$ in the form 5), i.e.

$$\widehat{\mathbf{v}}(\mathbf{r}) = \nabla\widetilde{\phi} + \nabla\times(\mathbf{r}\widetilde{\chi})\,, \tag{12}$$

we introduce "conversion" operators $\mathcal{C}_{ij}$, $i,j=1,2$:

$$\widetilde{\phi} = \mathcal{C}_{11}\widehat{\phi} + \mathcal{C}_{12}\widehat{\chi}, \quad \widetilde{\chi} = \mathcal{C}_{21}\widehat{\phi} + \mathcal{C}_{22}\widehat{\chi}, \tag{13}$$

which are linear, due to the linearity of all transforms considered.

Comparing representations (11) and (12) we can deduce, that $\widehat{\phi}(\mathbf{r})$ contributes only to $\widetilde{\phi}(\mathbf{r})$, which leads to

$$\mathcal{C}_{11} = \mathcal{I}, \quad \mathcal{C}_{21} = 0, \tag{14}$$

where $\mathcal{I}$ is the identity operator. So, we can introduce harmonic functions ϕ' and χ' according to the following relations

$$\begin{aligned}&\widetilde{\phi} = \widehat{\phi} + \mathcal{C}_{12}\widehat{\chi} = \widehat{\phi} + \phi', \\ &\widetilde{\chi} = \mathcal{C}_{22}\widehat{\chi} = \widehat{\chi} + \chi'.\end{aligned} \tag{15}$$

Having two representations of $\widehat{\mathbf{v}}$, (11) and (12), and using Eq. (15), we obtain

$$\nabla\phi' + \nabla\times\left(\mathbf{r}\chi'\right) = \nabla\times(\mathbf{t}\widehat{\chi})\,. \tag{16}$$

Taking scalar product with $\mathbf{r}$ and noticing that $\mathbf{r}\cdot\nabla\times(\mathbf{r}\chi') = 0$

$$\mathbf{r}\cdot\nabla\phi' = \mathbf{r}\cdot\nabla\times(\mathbf{t}\widehat{\chi}) = \mathbf{r}\cdot(\nabla\widehat{\chi}\times\mathbf{t}) = -(\mathbf{r}\times\mathbf{t})\cdot\nabla\widehat{\chi}. \tag{17}$$

Another relation can be obtained if we take the curl of expression (16):

$$\nabla\times\nabla\times\left(\mathbf{r}\chi'\right) = \nabla\times\nabla\times(\mathbf{t}\widehat{\chi})\,. \tag{18}$$

It is not difficult to check that for harmonic functions χ' and $\widehat{\chi}$ the following identities hold:

$$\begin{aligned}\nabla\times\nabla\times\left(\mathbf{r}\chi'\right) &= \nabla\left(\chi' + \mathbf{r}\cdot\nabla\chi'\right), \\ \nabla\times\nabla\times(\mathbf{t}\widehat{\chi}) &= \nabla(\mathbf{t}\cdot\nabla\widehat{\chi})\,.\end{aligned} \tag{19}$$

We note further, that all scalar potentials are defined up to a constant, so for the space integration an arbitrary constant can be added or dropped. Therefore, we obtain from Eqs (18) and (19):

$$\chi' + \mathbf{r}\cdot\nabla\chi' = \mathbf{t}\cdot\nabla\widehat{\chi}. \tag{20}$$

Using notation (10) we can rewrite relations (17) and (20) in the form

$$\mathcal{D}_{\mathbf{r}}\phi' = -\mathcal{D}_{\mathbf{r}\times\mathbf{t}}\widehat{\chi}, \quad (\mathcal{I}+\mathcal{D}_{\mathbf{r}})\,\chi' = \mathcal{D}_{\mathbf{t}}\widehat{\chi}. \tag{21}$$

In the next sections, we consider functions represented via harmonic expansions, and show that operators $\mathcal{D}_{\mathbf{r}}$ and $(\mathcal{I}+\mathcal{D}_{\mathbf{r}})$ are invertible, in which case, we can write solution in the form

$$\begin{aligned} \widetilde{\phi} &= \widehat{\phi}-\mathcal{D}_{\mathbf{r}}^{-1}\mathcal{D}_{\mathbf{r}\times\mathbf{t}}\widehat{\chi}, \\ \widetilde{\chi} &= \widehat{\chi}+(\mathcal{I}+\mathcal{D}_{\mathbf{r}})^{-1}\mathcal{D}_{\mathbf{t}}\widehat{\chi}. \end{aligned} \tag{22}$$

Comparing this with Eq. (15), we obtain the following expressions for the conversion operators

$$\mathcal{C}_{12}(\mathbf{t}) = -\mathcal{D}_{\mathbf{r}}^{-1}\mathcal{D}_{\mathbf{r}\times\mathbf{t}}, \quad \mathcal{C}_{22}(\mathbf{t}) = \mathcal{I}+(\mathcal{I}+\mathcal{D}_{\mathbf{r}})^{-1}\mathcal{D}_{\mathbf{t}}. \tag{23}$$

3.3 Expansions of harmonic functions

In addition to the Cartesian coordinates of points and vectors we will use spherical coordinates (r,θ,φ) in 3D:

$$\mathbf{r}=(x,y,z)=r(\sin\theta\cos\varphi,\sin\theta\sin\varphi,\cos\theta), \quad \mathbf{r}\in\mathbb{R}^3. \tag{24}$$

For expansions of the solutions of the Laplace equation in 3D that are regular inside or outside a sphere centered at the origin of the reference frame we introduce the regular or local functions $R_n^m(\mathbf{r})$ and singular or multipole functions $S_n^m(\mathbf{r})$, that are respectively defined as

$$\begin{aligned} R_n^m(\mathbf{r}) &= \frac{(-1)^n i^{|m|}}{(n+|m|)!} r^n P_n^{|m|}(\mu)e^{im\varphi}, \quad \mu=\cos\theta, \\ S_n^m(\mathbf{r}) &= \frac{i^{-|m|}(n-|m|)!}{r^{n+1}} P_n^{|m|}(\mu)e^{im\varphi}, \quad n=0,1,..., \quad m=-n,...,n, \end{aligned} \tag{25}$$

where $P_n^m(\mu)$ are the associated Legendre functions, defined by Rodrigues' formula

$$P_n^m(\mu)=\frac{(-1)^m\left(1-\mu^2\right)^{m/2}}{2^n n!}\frac{d^{m+n}}{d\mu^{m+n}}\left(\mu^2-1\right)^n, \; m\geqslant 0. \tag{26}$$

These functions are related to each other via

$$S_n^m(\mathbf{r})=(-1)^{n+m}(n-m)!(n+m)!r^{-2n-1}R_n^m(\mathbf{r}). \tag{27}$$

The functions $R_n^m(\mathbf{r})$ and $S_n^m(\mathbf{r})$ defined above coincide with the normalized basis functions $I_n^m(\mathbf{r})$ and $O_n^m(\mathbf{r})$ considered in [4] and normalized spherical basis functions in [11]. The expansion of the Green's function in this basis is

$$|\mathbf{r}-\mathbf{r}_0|^{-1}=\sum_{n=0}^{\infty}\sum_{m=-n}^{n}R_n^{-m}(-\mathbf{r}_0)S_n^m(\mathbf{r}), \quad r>r_0. \tag{28}$$

Further, we represent harmonic functions in terms of sets of expansion coefficients over a certain basis centered at a given point, e.g. the local and multipole expansions centered at the origin are

$$\phi(\mathbf{r})=\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\phi_n^m F_n^m(\mathbf{r}), \quad \chi(\mathbf{r})=\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\chi_n^m F_n^m(\mathbf{r}), \quad F=R,S. \tag{29}$$

Absolute and uniform convergence of these series in the expansion regions is assumed everywhere below. We also extend the definition of the basis functions for arbitrary order m, to shorten some expressions

$$R_n^m(\mathbf{r})=S_n^m(\mathbf{r})=0, \quad |m|>n, \quad n=0,1,... \tag{30}$$

3.4 Matrix representation of operators

Let $\mathcal{L}$ be a linear operator, such that for harmonic function ϕ, $\psi = \mathcal{L}\phi$ be also a harmonic function. Assume further that both ϕ and ψ can be expanded in to series of type (29). There should be a linear relation between the expansion coefficients $\mathbf{\Psi} = \{\psi_n^m\}$ and $\mathbf{\Phi} = \{\phi_n^m\}$, which, generally speaking, will have a form $\mathbf{\Psi} = \mathbf{L}\mathbf{\Phi}$, where $\mathbf{L}$ is a matrix, or representation of $\mathcal{L}$. Of course, for a given $\mathcal{L}$ the matrix $\mathbf{L}$ depends on the bases over which the expansion is taken.

Let $\phi(\mathbf{r})$ be expanded over basis $\{F_n^m(\mathbf{r})\}$, while $\psi(\mathbf{r})$ be expanded over basis $\{G_n^m(\mathbf{r})\}$. Action of operator $\mathcal{L}$ on a basis function $F_n^m(\mathbf{r})$ can be represented as

$$\mathcal{L}F_n^m(\mathbf{r}) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} L_{n'n}^{m'm} G_{n'}^{m'}(\mathbf{r}), \quad n = 0,1,..., \quad m = -n,...,n, \tag{31}$$

where $L_{n'n}^{m'm}$ are the reexpansion coefficients. It is not difficult to show then that the entries of matrix $\mathbf{L}$ are $L_{nn'}^{mm'}$, i.e. $\mathbf{L}$ is the matrix transposed to the matrix of reexpansion coefficients. Indeed

$$\begin{aligned}\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\psi_n^m G_n^m(\mathbf{r}) &= \psi(\mathbf{r}) = \mathcal{L}\phi(\mathbf{r}) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\phi_{n'}^{m'}\mathcal{L}F_{n'}^{m'}(\mathbf{r}) \\ &= \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\phi_{n'}^{m'}\sum_{n=0}^{\infty}\sum_{m=-n}^{n} L_{nn'}^{mm'} G_n^m(\mathbf{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\left[\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} L_{nn'}^{mm'}\phi_{n'}^{m'}\right] G_n^m(\mathbf{r}).\end{aligned} \tag{32}$$

Reexpansion coefficients for the translation operator $\mathcal{T}_{\mathbf{t}}$, Eq. (9), in the local and multipole bases (25) can be simply expressed via the respective basis functions (see [4] and [11]):

$$\begin{aligned}\mathcal{T}_{\mathbf{t}}F_n^m(\mathbf{r}) &= \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}(F|G)_{n'n}^{m'm}(\mathbf{t})\, G_{n'}^{m'}(\mathbf{r}), \quad F,G = S,R, \\ (R|R)_{n'n}^{m'm}(\mathbf{t}) &= R_{n-n'}^{m-m'}(\mathbf{t}), \quad (S|R)_{n'n}^{m'm}(\mathbf{t}) = S_{n+n'}^{m-m'}(\mathbf{t}), \quad (S|S)_{n'n}^{m'm}(\mathbf{t}) = R_{n'-n}^{m-m'}(\mathbf{t}).\end{aligned} \tag{33}$$

To obtain representations of other operators appeared above, we will use differential relations for the basis functions, which also can be found in [4] and [11]:

$$\begin{aligned}\mathcal{D}_z R_n^m(\mathbf{r}) &= -R_{n-1}^m(\mathbf{r}), \quad \mathcal{D}_z S_n^m(\mathbf{r}) = -S_{n+1}^m(\mathbf{r}), \\ \mathcal{D}_{x+iy} R_n^m(\mathbf{r}) &= iR_{n-1}^{m+1}(\mathbf{r}), \quad \mathcal{D}_{x+iy} S_n^m(\mathbf{r}) = iS_{n+1}^{m+1}(\mathbf{r}), \\ \mathcal{D}_{x-iy} R_n^m(\mathbf{r}) &= iR_{n-1}^{m-1}(\mathbf{r}), \quad \mathcal{D}_{x-iy} S_n^m(\mathbf{r}) = iS_{n+1}^{m-1}(\mathbf{r}).\end{aligned} \tag{34}$$

where

$$\mathcal{D}_{x\pm iy} = \frac{\partial}{\partial x} \pm i\frac{\partial}{\partial y}, \quad \mathcal{D}_z = \frac{\partial}{\partial z}. \tag{35}$$

3.4.1 Operator $\mathcal{D}_{\mathbf{r}}$.

To obtain the representation of this operator there is no need to consider differential relations, since according to definition (10) we have $\mathcal{D}_{\mathbf{r}} = r\,(\partial/\partial r)$ and from Eq. (25) we have

$$\mathcal{D}_{\mathbf{r}} R_n^m(\mathbf{r}) = nR_n^m(\mathbf{r}), \quad \mathcal{D}_{\mathbf{r}} S_n^m(\mathbf{r}) = -(n+1)\, S_n^m(\mathbf{r}), \quad n = 0,1,... \tag{36}$$

This shows that matrices $\mathbf{D}_{\mathbf{r}}^{(R)}$ and $\mathbf{D}_{\mathbf{r}}^{(S)}$ representing this operator are diagonal and have entries

$$\left(\mathbf{D}_{\mathbf{r}}^{(R)}\right)_{nn'}^{mm'} = n\delta_{mm'}\delta_{nn'}, \quad \left(\mathbf{D}_{\mathbf{r}}^{(S)}\right)_{nn'}^{mm'} = -(n+1)\,\delta_{mm'}\delta_{nn'}, \quad n = 0,1,... \tag{37}$$

where $\delta_{mm'}$ is the Kronecker symbol.

Conversion operators (23) contain inverse operators $\mathcal{D}_{\mathbf{r}}^{-1}$ and $(\mathcal{I}+\mathcal{D}_{\mathbf{r}})^{-1}$, which also are diagonal. It may be a cause for concern for the inverse operators that zeros appear on the diagonal of matrix $\mathbf{D}_{\mathbf{r}}^{(R)}$ and on the diagonal of matrix $\mathbf{I}+\mathbf{D}_{\mathbf{r}}^{(S)}$ at $n=0$. However, these are easily dispensed with. For the former case we note that harmonic $n=0$ corresponds to a constant basis function $R_0^0(\mathbf{r})$. Eq. (22) shows that this affects only the constant added to potential ϕ, which obviously does not affect the velocity field (5), and can be set to an arbitrary value, e.g. to zero. For the latter case, Eq. (11) shows that harmonic $n=0$ in multipole expansion of function χ also does not affect the velocity field, since

$$\nabla\times\left(\mathbf{r}r^{-1}\right)=-\mathbf{r}\times\nabla\left(r^{-1}\right)=r^{-2}\mathbf{r}\times\mathbf{r}=\mathbf{0}. \tag{38}$$

Since operator $\left(\mathbf{I}+\mathbf{D}_{\mathbf{r}}^{(S)}\right)^{-1}$ is needed only to determine converted function χ (see Eq. (22) this also can be set to zero. In other words, for the purpose of computation of the conversion operators the inverse operators for singular matrices can be defined as follows

$$\left[\left(\mathbf{D}_{\mathbf{r}}^{(R)}\right)^{-1}\right]_{nn'}^{mm'}=\delta_{mm'}\delta_{nn'}\left\{\begin{array}{cc} n^{-1}, & n>0 \\ 0, & n=0\end{array}\right., \quad \left[\left(\mathbf{I}+\mathbf{D}_{\mathbf{r}}^{(S)}\right)^{-1}\right]_{nn'}^{mm'}=\delta_{mm'}\delta_{nn'}\left\{\begin{array}{cc} -n^{-1}, & n>0 \\ 0, & n=0\end{array}\right. . \tag{39}$$

3.4.2 Operator $\mathcal{D}_{\mathbf{t}}$.

From definitions (10) and (35) we have

$$\mathcal{D}_{\mathbf{t}}=t_x\frac{\partial}{\partial x}+t_y\frac{\partial}{\partial y}+t_z\frac{\partial}{\partial z}=\frac{1}{2}\left[\left(t_x-it_y\right)\mathcal{D}_{x+iy}+(t_x+it_y)\mathcal{D}_{x-iy}\right]+t_z\mathcal{D}_z. \tag{40}$$

Using Eq. (34) we obtain

$$\begin{aligned}\mathcal{D}_{\mathbf{t}}R_n^m &= \frac{1}{2}\left[\left(t_y+it_x\right)R_{n-1}^{m+1}-(t_y-it_x)R_{n-1}^{m-1}\right]-t_zR_{n-1}^m, \\ \mathcal{D}_{\mathbf{t}}S_n^m &= \frac{1}{2}\left[\left(t_y+it_x\right)S_{n+1}^{m+1}-(t_y-it_x)S_{n+1}^{m-1}\right]-t_zS_{n+1}^m.\end{aligned} \tag{41}$$

Taking into account that matrices $\mathbf{D}_{\mathbf{t}}^{(R)}$ and $\mathbf{D}_{\mathbf{t}}^{(S)}$ are transposed to the reexpansion matrices, we obtain

$$\begin{aligned}\left(\mathbf{D}_{\mathbf{t}}^{(R)}\right)_{nn'}^{mm'} &= \delta_{n+1,n'}\left[\frac{1}{2}\left(t_y+it_x\right)\delta_{m-1,m'}-\frac{1}{2}(t_y-it_x)\delta_{m+1,m'}-t_z\delta_{mm'}\right], \\ \left(\mathbf{D}_{\mathbf{t}}^{(S)}\right)_{nn'}^{mm'} &= \delta_{n-1,n'}\left[\frac{1}{2}\left(t_y+it_x\right)\delta_{m-1,m'}-\frac{1}{2}(t_y-it_x)\delta_{m+1,m'}-t_z\delta_{mm'}\right].\end{aligned} \tag{42}$$

This also can be rewritten in terms of the expansion coefficients of functions ψ and ϕ, $\psi=\mathcal{D}_{\mathbf{t}}\phi$,

$$\begin{aligned}\psi_n^m &= \sum_{n,m}\left(\mathbf{D}_{\mathbf{t}}^{(R)}\right)_{nn'}^{mm'}\phi_{n'}^{m'}=\frac{1}{2}\left[\left(t_y+it_x\right)\phi_{n+1}^{m-1}-(t_y-it_x)\phi_{n+1}^{m+1}\right]-t_z\phi_{n+1}^m, \\ \psi_n^m &= \sum_{n,m}\left(\mathbf{D}_{\mathbf{t}}^{(S)}\right)_{nn'}^{mm'}\phi_{n'}^{m'}=\frac{1}{2}\left[\left(t_y+it_x\right)\phi_{n-1}^{m-1}-(t_y-it_x)\phi_{n-1}^{m+1}\right]-t_z\phi_{n-1}^m.\end{aligned} \tag{43}$$

3.4.3 Operator $\mathcal{D}_{\mathbf{r}\times\mathbf{t}}$.

From definitions (10) and (35) we have

$$\begin{aligned}
\mathcal{D}_{\mathbf{r}\times\mathbf{t}} &= (yt_z - zt_y)\frac{\partial}{\partial x} + (zt_x - xt_z)\frac{\partial}{\partial y} + (xt_y - yt_x)\frac{\partial}{\partial z} \qquad (44)\\
&= -i(t_x + it_y)\left[\xi_-\mathcal{D}_z - \frac{1}{2}z\mathcal{D}_{x-iy}\right] + i(t_x - it_y)\left[\xi_+\mathcal{D}_z - \frac{1}{2}z\mathcal{D}_{x+iy}\right] + it_z\left[\xi_-\mathcal{D}_{x+iy} - \xi_+\mathcal{D}_{x-iy}\right],\\
\xi_\pm &= \frac{x \pm iy}{2}.
\end{aligned}$$

Consider first action of this operator on basis functions $R_n^m(\mathbf{r})$. The following relations derived in [11] are useful in this case:

$$\begin{aligned}
\xi_+ R_n^m &= -i\frac{n+m+2}{2}R_{n+1}^{m+1} - \frac{i}{2}zR_n^{m+1}, \qquad (45)\\
\xi_- R_n^m &= -i\frac{n-m+2}{2}R_{n+1}^{m-1} - \frac{i}{2}zR_n^{m-1}.
\end{aligned}$$

We have then, using these relations and Eq. (34)

$$\begin{aligned}
\left[\xi_-\mathcal{D}_z - \frac{1}{2}z\mathcal{D}_{x-iy}\right]R_n^m &= -\xi_- R_{n-1}^m - \frac{1}{2}izR_{n-1}^{m-1} = i\frac{n-m+1}{2}R_n^{m-1}, \qquad (46)\\
\left[\xi_+\mathcal{D}_z - \frac{1}{2}z\mathcal{D}_{x+iy}\right]R_n^m &= -\xi_+ R_{n-1}^m - \frac{1}{2}izR_{n-1}^{m+1} = i\frac{n+m+1}{2}R_n^{m+1},\\
\left[\xi_-\mathcal{D}_{x+iy} - \xi_+\mathcal{D}_{x-iy}\right]R_n^m &= i\xi_- R_{n-1}^{m+1} - i\xi_+ R_{n-1}^{m-1} = -mR_n^m.
\end{aligned}$$

Now we obtain from Eq. (44)

$$\mathcal{D}_{\mathbf{r}\times\mathbf{t}}R_n^m = (t_x + it_y)\frac{n-m+1}{2}R_n^{m-1} - (t_x - it_y)\frac{n+m+1}{2}R_n^{m+1} - it_z m R_n^m. \qquad (47)$$

To get a similar relation for basis functions $S_n^m(\mathbf{r})$ we can use relation (27). Using identity $(\mathbf{r}\times\mathbf{t})\cdot\nabla\left[f(r)\,g(\mathbf{r})\right] = f(r)\,(\mathbf{r}\times\mathbf{t})\cdot\nabla g(\mathbf{r})$ and Eq. (47), we obtain

$$\begin{aligned}
\mathcal{D}_{\mathbf{r}\times\mathbf{t}}S_n^m &= (-1)^{n+m}(n-m)!(n+m)!r^{-2n-1}\mathcal{D}_{\mathbf{r}\times\mathbf{t}}R_n^m \qquad (48)\\
&= -(t_x + it_y)\frac{n+m}{2}S_n^{m-1} + (t_x - it_y)\frac{n-m}{2}S_n^{m+1} - it_z m S_n^m.
\end{aligned}$$

Expressions for the representing matrices for the local and multipole bases follow from Eqs (47) and (48):

$$\begin{aligned}
\left(\mathbf{D}_{\mathbf{r}\times\mathbf{t}}^{(R)}\right)_{nn'}^{mm'} &= \delta_{nn'}\left[(t_x + it_y)\frac{n'-m'+1}{2}\delta_{m+1,m'} - (t_x - it_y)\frac{n'+m'+1}{2}\delta_{m-1,m'} - it_z m'\delta_{mm'}\right], \qquad (49)\\
\left(\mathbf{D}_{\mathbf{r}\times\mathbf{t}}^{(S)}\right)_{nn'}^{mm'} &= \delta_{nn'}\left[-(t_x + it_y)\frac{n'+m'}{2}\delta_{m+1,m'} + (t_x - it_y)\frac{n'-m'}{2}\delta_{m-1,m'} - it_z m'\delta_{mm'}\right].
\end{aligned}$$

This also can be rewritten in terms of the expansion coefficients of functions ψ and ϕ, $\psi = \mathcal{D}_{\mathbf{r}\times\mathbf{t}}\phi$,

$$\begin{aligned}
\psi_n^m &= \sum_{n,m}\left(\mathbf{D}_{\mathbf{r}\times\mathbf{t}}^{(R)}\right)_{nn'}^{mm'}\phi_{n'}^{m'} = (t_x + it_y)\frac{n-m}{2}\phi_n^{m+1} - (t_x - it_y)\frac{n+m}{2}\phi_n^{m-1} - it_z m\phi_n^m, \qquad (50)\\
\psi_n^m &= \sum_{n,m}\left(\mathbf{D}_{\mathbf{r}\times\mathbf{t}}^{(S)}\right)_{nn'}^{mm'}\phi_{n'}^{m'} = -(t_x + it_y)\frac{n+m+1}{2}\phi_n^{m+1} + (t_x - it_y)\frac{n-m+1}{2}\phi_n^{m-1} - it_z m\phi_n^m.
\end{aligned}$$

3.4.4 Conversion operators

It is not difficult to obtain matrix representations for the conversion operator from Eqs (14) and (23) and expressions for the differential operators derived above. A more compact form relating the expansion coefficients of functions in Eq. (22) for the R-expansions is

$$\begin{aligned}\widetilde{\phi}_n^m &= \widehat{\phi}_n^m - \frac{1}{n}\left[(t_x+it_y)\frac{n-m}{2}\widehat{\chi}_n^{m+1} - (t_x-it_y)\frac{n+m}{2}\widehat{\chi}_n^{m-1} - it_z m\widehat{\chi}_n^m\right], \quad \left(\widetilde{\phi}_0^0 = \widehat{\phi}_0^0\right) \\ \widetilde{\chi}_n^m &= \widehat{\chi}_n^m + \frac{1}{n+1}\left[\frac{1}{2}(t_y+it_x)\widehat{\chi}_{n+1}^{m-1} - \frac{1}{2}(t_y-it_x)\widehat{\chi}_{n+1}^{m+1} - t_z\widehat{\chi}_{n+1}^m\right].\end{aligned} \tag{51}$$

Similarly, for the S-expansions we have

$$\begin{aligned}\widetilde{\phi}_n^m &= \widehat{\phi}_n^m + \frac{1}{n+1}\left[-(t_x+it_y)\frac{n+m+1}{2}\widehat{\chi}_n^{m+1} + (t_x-it_y)\frac{n-m+1}{2}\widehat{\chi}_n^{m-1} - it_z m\widehat{\chi}_n^m\right] \\ \widetilde{\chi}_n^m &= \widehat{\chi}_n^m - \frac{1}{n}\left[\frac{1}{2}(t_y+it_x)\widehat{\chi}_{n-1}^{m-1} - \frac{1}{2}(t_y-it_x)\widehat{\chi}_{n-1}^{m+1} - t_z\widehat{\chi}_{n-1}^m\right], \quad \left(\widetilde{\chi}_0^0 = \widehat{\chi}_0^0\right).\end{aligned} \tag{52}$$

3.5 RCR-decomposition

In many FMM codes translations are performed using the rotation-coaxial translation-back rotation (RCR) decomposition of the translation operators , which reduces translation cost of expansions of length p^2 to $O(p^3)$, opposed to $O(p^4)$ required for the direct application of the translation matrix (e.g. see [11]). Such a decomposition is also beneficial for faster conversion, since the rotations do not change the form of decomposition of the vector field (5) and there is no need to rotate ϕ and χ in conversion operators. Coaxial translation means translation along the z-direction to distance t, in which case expressions for the conversion operators (51) and (52) become even simpler ($\mathbf{t} = t\mathbf{i}_z$, $t_x = t_y = 0$, $t_z = t$):

$$\begin{aligned}&R\text{-conversion} : \quad \widetilde{\phi}_n^m = \widehat{\phi}_n^m + it\frac{m}{n}\widehat{\chi}_n^m, \quad \left(\widetilde{\phi}_0^0 = \widehat{\phi}_0^0\right), \quad \widetilde{\chi}_n^m = \widehat{\chi}_n^m - \frac{t}{n+1}\widehat{\chi}_{n+1}^m, \\ &S\text{-conversion} : \quad \widetilde{\phi}_n^m = \widehat{\phi}_n^m - it\frac{m}{n+1}\widehat{\chi}_n^m, \quad \widetilde{\chi}_n^m = \widehat{\chi}_n^m + \frac{t}{n}\widehat{\chi}_{n-1}^m, \quad \left(\widetilde{\chi}_0^0 = \widehat{\chi}_0^0\right).\end{aligned} \tag{53}$$

3.6 Other harmonic bases

In the case of the use of other than $\{R_n^m\}$ and $\{S_n^m\}$ bases, i.e. $\{R_n'^m\}$ and $\{S_n'^m\}$, matrices representing the operators should be modified accordingly. Generally speaking, we can introduce basis reexpansion matrices, $\mathbf{B}^{(R)}$ and $\mathbf{B}^{(S)}$, for the local and multipole bases, so

$$F_n'^m(\mathbf{r}) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\left[\mathbf{B}^{(F)}\right]_{n'n}^{m'm} F_{n'}^{m'}(\mathbf{r}), \quad F = S, R. \tag{54}$$

Which results in conversion of expansion coefficients in basis F to expansion coefficients in basis F' with matrix $\mathbf{B}^{(F)}$, $\mathbf{C}' = \mathbf{B}^{(F)}\mathbf{C}$. Assume now that $\mathbf{L}$ is the matrix representing operator $\mathcal{L}$ in basis F, for which we have expressions derived above. Matrix $\mathbf{L}'$, representing the same operator in basis F' then will be $\mathbf{L}' = \mathbf{B}^{(F)}\mathbf{L}\left[\mathbf{B}^{(F)}\right]^{-1}$, where invertibility is provided by the fact that both systems $\{F_n'^m\}$ and $\{F_n^m\}$ are complete (bases). In the case of complex harmonic bases, basis transform matrices are usually diagonal, i.e.

$$\left[\mathbf{B}^{(F)}\right]_{n'n}^{m'm} = \delta_{nn'}\delta_{mm'}B_n^{(F)m}, \quad n = 0, 1, ..., \quad m = -n, ..., n. \tag{55}$$

In this case the above relations can be easily modified. For example, conversion relations for coaxial translations (53) for coefficients in bases R' and S' become

$$R\text{-conversion} \quad : \quad \widetilde{\phi}_n^{\prime m} = \widehat{\phi}_n^{\prime m} + it\frac{m}{n}\widehat{\chi}_n^{\prime m}, \quad \left(\widetilde{\phi}_0^{\prime 0} = \widehat{\phi}_0^{\prime 0}\right), \quad \widetilde{\chi}_n^{\prime m} = \widehat{\chi}_n^{\prime m} - \frac{t}{n+1}\frac{B_n^{(R)m}}{B_{n+1}^{(R)m}}\widehat{\chi}_{n+1}^{\prime m} \tag{56}$$

$$S\text{-conversion} \quad : \quad \widetilde{\phi}_n^{\prime m} = \widehat{\phi}_n^{\prime m} - it\frac{m}{n+1}\widehat{\chi}_n^{\prime m}, \quad \widetilde{\chi}_n^{\prime m} = \widehat{\chi}_n^{\prime m} + \frac{t}{n}\frac{B_n^{(S)m}}{B_{n-1}^{(S)m}}\widehat{\chi}_{n-1}^{\prime m}, \quad \left(\widetilde{\chi}_0^{\prime 0} = \widehat{\chi}_0^{\prime 0}\right).$$

This also can be checked by substitution $\phi_n^{\prime m} = B_n^{(F)m}\phi_n^m$ for all functions with hats and tildes (similarly for χ) in Eq. (53).

4 Fast multipole method

There is an extensive literature on the FMM for the 3D Laplace equation and several implementations have been descripted. Because of this, we do not see any need to describe this algorithm again, and refer the reader to papers, which in details discuss the method [6, 2, 7, 10, 13].

4.1 Mapping of a real vector field to a complex harmonic function

It is proposed in [11] to modify an available FMM routine for complex valued harmonic function to an FMM routine which provides the FMM for real valued biharmonic functions. So just one complex FMM can be executed instead of two FMMs for the real functions. Our tests show that such an approach provides a small advantage compared to the FMMs for real harmonic functions (which modification requires a bit more analytical work, to convert matrix representations of the conversion operators of the present paper given in complex bases into respective expressions in the real bases, e.g. used in GPU implementation [13]). Anyway, this method can be taken and applied directly to the present case, since a complex valued harmonic function $\Psi(\mathbf{r})$ can be composed from two real functions $\phi(\mathbf{r})$ and $\chi(\mathbf{r})$, as

$$\Psi(\mathbf{r}) = \phi(\mathbf{r}) + i\chi(\mathbf{r}). \tag{57}$$

Further the translation algorithm for $\Psi(\mathbf{r})$ will be exactly the same as for the biharmonic functions, described in [11], with the only difference in the conversion operators, where relations (53) should be used instead of conversion relations for the biharmonic functions.

4.2 Expansion of elementary velocities

Consider multipole expansion of $\mathbf{v}_l(\mathbf{y})$ given by Eq. (1) about the center, $\mathbf{x}_*$,of a source box containing $\mathbf{x}_l$. Using Eq. (28), we obtain

$$\begin{aligned}\mathbf{v}_l(\mathbf{y}) &= \nabla\times\frac{\boldsymbol{\omega}_l}{|\mathbf{r}-\mathbf{r}_l|} = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} g_{ln}^m \mathbf{F}_{ln}^m(\mathbf{r}), \quad \mathbf{F}_{ln}^m(\mathbf{r}) = \nabla\times\left[\omega_l S_n^m(\mathbf{r})\right], \\ g_{ln}^m &= R_n^{-m}(-\mathbf{r}_l), \quad \mathbf{r} = \mathbf{y}-\mathbf{x}_*, \quad \mathbf{r}_l = \mathbf{x}_l - \mathbf{x}_*\end{aligned} \tag{58}$$

Note then that $\mathbf{F}_{ln}^m$ is equivalent to the right hand side of Eq. (16), where one should set $\mathbf{t} = \omega_l$, $\widehat{\chi} = S_n^m(\mathbf{r})$. So this function can be represented in the form provided by the left hand side of Eq. (16), where functions ϕ' and χ' can be found from Eqs (21) and (23) i.e.

$$\begin{aligned}\mathbf{F}_{ln}^m(\mathbf{r}) &= \nabla\times\left[\omega_l S_n^m(\mathbf{r})\right] = \nabla\Phi_{ln}^m(\mathbf{r}) + \nabla\times\left(\mathbf{r}X_{ln}^m(\mathbf{r})\right), \\ \Phi_{ln}^m(\mathbf{r}) &= \mathcal{C}_{12}(\omega_l)\, S_n^m(\mathbf{r}), \quad X_{ln}^m(\mathbf{r}) = \left(\mathcal{C}_{22}(\boldsymbol{\omega}_l) - \mathcal{I}\right) S_n^m(\mathbf{r}).\end{aligned} \tag{59}$$

Substituting this into Eq. (58) and using representation of the conversion operators in the space of expansion coefficients, Eq. (52), we obtain

$$
\begin{aligned}
\mathbf{v}_l(\mathbf{y}) &= \nabla\phi_l(\mathbf{r}) + \nabla\times(\mathbf{r}\chi_l(\mathbf{r})), \\
\phi_l(\mathbf{r}) &= \sum_{n=0}^{\infty}\sum_{m=-n}^{n} g_{ln}^m \mathcal{C}_{12}(\boldsymbol{\omega}_l) S_n^m(\mathbf{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} \phi_{ln}^m S_n^m(\mathbf{r}), \\
\chi_l(\mathbf{r}) &= \sum_{n=0}^{\infty}\sum_{m=-n}^{n} g_{ln}^m \left(\mathcal{C}_{22}(\boldsymbol{\omega}_l) - \mathcal{I}\right) S_n^m(\mathbf{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} \chi_{ln}^m S_n^m(\mathbf{r}),
\end{aligned} \tag{60}
$$

where

$$
\begin{aligned}
\phi_{ln}^m &= \sum_{n,m}\left[\mathbf{C}_{12}^{(S)}(\omega_l)\right]_{mm'}^{nn'} g_{ln}^m \\
&= \frac{1}{n+1}\left[(\omega_{lx} - i\omega_{ly})\frac{n-m+1}{2} g_{ln}^{m-1} - (\omega_{lx} + i\omega_{ly})\frac{n+m+1}{2} g_{ln}^{m+1} - i\omega_{lz} m g_{ln}^m\right], \\
\chi_{ln}^m &= \sum_{n,m}\left[\mathbf{C}_{22}^{(S)}(\omega_l) - \mathbf{I}\right]_{mm'}^{nn'} g_{ln}^m \\
&= -\frac{1}{n}\left[\frac{1}{2}(\omega_{ly} + i\omega_{lx}) g_{l,n-1}^{m-1} - \frac{1}{2}(\omega_{ly} - i\omega_{lx}) g_{l,n-1}^{m+1} - \omega_{lz} g_{l,n-1}^m\right], \quad \left(\chi_{l0}^0 = 0\right).
\end{aligned} \tag{61}
$$

In the FMM coefficients g_{ln}^m are generated routinely (as coefficients of expansion of monopoles). So a small modification of an available routine is needed to obtain ϕ_{ln}^m and χ_{ln}^m for a given vector $\boldsymbol{\omega}_l$.

It is also not difficult to show that harmonic functions

$$
\psi_l = \mathbf{r}\cdot\nabla\phi_l = \mathcal{D}_{\mathbf{r}}\phi_l, \quad \varpi_l = \chi_l + \mathbf{r}\cdot\nabla\chi_l = (\mathcal{I} + \mathcal{D}_{\mathbf{r}})\chi_l, \tag{62}
$$

which spectra are simply related to scalar potentials for source l, are nothing but dipoles, and so they can be computed using standard subroutines for dipole expansions. Indeed, taking scalar product of both sides of equation

$$
\nabla\phi_l(\mathbf{r}) + \nabla\times(\mathbf{r}\chi_l(\mathbf{r})) = \nabla\times\frac{\omega_l}{|\mathbf{r}-\mathbf{r}_l|} \tag{63}
$$

with $\mathbf{r}$, we can see that

$$
\psi_l = \mathbf{r}\cdot\nabla\times\frac{\boldsymbol{\omega}_l}{|\mathbf{r}-\mathbf{r}_l|} = (\mathbf{r}_l\times\boldsymbol{\omega}_l)\cdot\left(\frac{\mathbf{r}-\mathbf{r}_l}{|\mathbf{r}-\mathbf{r}_l|^3}\right), \tag{64}
$$

i.e. ψ_l is a dipole with moment $\mathbf{p}_l = \mathbf{r}_l\times\boldsymbol{\omega}_l$. Taking curl of both sides of Eq. (63), we obtain, using identity $\nabla\times\nabla\times = \nabla(\nabla\cdot) - \nabla^2$, and $\nabla^2(\mathbf{r}\chi_l) = 2\nabla\chi_l$, which is valid for any harmonic function χ

$$
\nabla\left[\nabla\cdot(\mathbf{r}\chi_l(\mathbf{r})) - 2\chi_l(\mathbf{r})\right] = \nabla\left(\nabla\cdot\frac{\boldsymbol{\omega}_l}{|\mathbf{r}-\mathbf{r}_l|}\right). \tag{65}
$$

So, up to the integration constant

$$
\varpi_l = \nabla\cdot[\mathbf{r}\chi_l(\mathbf{r})] - 2\chi_l(\mathbf{r}) = \nabla\cdot\frac{\omega_l}{|\mathbf{r}-\mathbf{r}_l|} = \frac{-\boldsymbol{\omega}_l\cdot(\mathbf{r}-\mathbf{r}_l)}{|\mathbf{r}-\mathbf{r}_l|^3}, \tag{66}
$$

i.e. ϖ_l is a dipole with moment $\mathbf{p}_l = -\omega_l$.

Note that for reconstruction of the vector field from the curl and divergence, expansion of a monopole of intensity $-q_l$ should be added to ϕ_l (see Eq. (8)).

4.3 Evaluation of the vector field

Other procedures, which can be accelerated are related to the final step of the FMM for the dense matrix-vector product. As a result of the FMM algorithm R-expansions of scalar potentials are obtained about the center $\mathbf{y}_*$,of an evaluation box containing receiver point $\mathbf{y}_j$

$$\phi(\mathbf{r}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \phi_n^m R_n^m(\mathbf{r}), \quad \chi(\mathbf{r}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \chi_n^m R_n^m(\mathbf{r}), \quad \mathbf{r} = \mathbf{y} - \mathbf{y}_*. \tag{67}$$

Cartesian components of the velocity then can be obtained by projection of Eq. (5) to the basis vectors $\mathbf{i}_x, \mathbf{i}_y$, and $\mathbf{i}_z$ as follows

$$v_k = \mathbf{i}_k \cdot \mathbf{v} = \mathbf{i}_k \cdot \nabla\phi + \mathbf{i}_k \cdot \nabla \times (\mathbf{r}\chi) = \mathcal{D}_{\mathbf{i}_k}\phi + \mathcal{D}_{\mathbf{r}\times\mathbf{i}_k}\chi, \quad k = x, y, z. \tag{68}$$

since $\mathbf{i}_k \cdot \nabla \times (\mathbf{r}\chi) = \mathbf{i}_k \cdot (\nabla\chi \times \mathbf{r}) = (\mathbf{r} \times \mathbf{i}_k) \cdot \nabla\chi$. Using representations of the above operators in the space of expansion coefficients (43) and (50), where $\mathbf{t} = \mathbf{i}_k$, we determine

$$\begin{aligned}
v_k &= \sum_{n=0}^{\infty} \sum_{m=-n}^{n} v_{kn}^m R_n^m(\mathbf{r}), \quad k = x, y, z, \\
v_{xn}^m &= \frac{1}{2} \left[i\phi_{n+1}^{m-1} + i\phi_{n+1}^{m+1} + (n-m)\,\chi_n^{m+1} - (n+m)\,\chi_n^{m-1} \right], \\
v_{yn}^m &= \frac{1}{2} \left[\phi_{n+1}^{m-1} - \phi_{n+1}^{m+1} + i\,(n-m)\,\chi_n^{m+1} + i\,(n+m)\,\chi_n^{m-1} \right], \\
v_{zn}^m &= -\phi_{n+1}^m - im\chi_n^m.
\end{aligned} \tag{69}$$

Furthermore, consider computation of the vortex stretching at evaluation point $\mathbf{y}_j$ $(\mathbf{r}_j = \mathbf{y}_j - \mathbf{y}_*)$, assuming that the vorticity vector at this point is $\boldsymbol{\omega}_j$. The stretching is a vector

$$\mathbf{s}_j = (\boldsymbol{\omega}_j \cdot \nabla)\, \mathbf{v}(\mathbf{r})\big|_{\mathbf{r}=\mathbf{r}_j} = \mathcal{D}_{\boldsymbol{\omega}_j} \mathbf{v}(\mathbf{r})\big|_{\mathbf{r}=\mathbf{r}_j} = \sum_k \mathbf{i}_k\, \mathcal{D}_{\boldsymbol{\omega}_j} v_k(\mathbf{r})\big|_{\mathbf{r}=\mathbf{r}_j}. \tag{70}$$

Hence, the Cartesian components of this vector can be obtained simply from computed coefficients v_{kn}^m, Eq. (69), to which sparse operator $\mathbf{D}_{\boldsymbol{\omega}_j}^{(R)}$ should be applied (see Eq. (43)):

$$\begin{aligned}
s_{jk} &= \sum_{n=0}^{\infty} \sum_{m=-n}^{n} s_{jkn}^m R_n^m(\mathbf{r}_j), \quad k = x, y, z, \\
s_{jkn}^m &= \frac{1}{2} \left[(\omega_{jy} + i\omega_{jx})\, v_{k,n+1}^{m-1} - (\omega_{jy} - i\omega_{jx}) v_{k,n+1}^{m+1} \right] - \omega_{jz} v_{k,n+1}^m.
\end{aligned} \tag{71}$$

This shows that computation of this term can be also performed efficiently, and the data on expansion coefficients of ϕ and χ can be easiliy converted to the parameters required for computation of a vortical flow. In practice, it can be more efficient to compute expansion coefficients u_{lkn}^m for functions $u_{lk} = \mathcal{D}_{\mathbf{i}_l} v_k$, $l, k = 1, 2, 3$, which do not depend on the evaluation point j and then form appropriate coefficients s_{jkn}^m for each point as

$$s_{jkn}^m = \sum_{l=1}^{3} \omega_{jl} u_{lkn}^m. \tag{72}$$

Note that u_{lk} are components of tensor $\nabla\mathbf{v}$. So as these quantities are available, contraction $\beta = \nabla\mathbf{v} : \nabla\mathbf{v}$ can be computed (this is needed for compressible flows, see a remark at the end of section "Statement of the Problem"). Also, computation of $\nabla\mathbf{v}$ provides the strain tensor, which can be used for modeling of complex fluids.

5 Numerical tests

For the numerical tests we used basic FMM code with the RCR-decomposition of translation operators modified for two harmonic functions. Conversion operators in the R- or S- basis were executed after coaxial translation operators, as described in [11]. Additional small modifications were used in the algorithm to compute R-basis functions for real harmonic functions recursively, as presented in [13]. In contrast to [13] no optimizations of the algorithm were used (no GPU, standard 189 M2L translation stencils, no variable truncation number, etc.), as the main purpose of this paper was to provide a basic comparative performance and accuracy test of the method. Nonetheless, to speedup the tests Open MP parallelization was used, which for 4 core PC provided parallelization efficiency close to 100 percent. Wall clock times reported below were measured for Intel QX6780 (2.8 GHz) 4 core PC with 8GB RAM.

5.1 Error tests

The first test we conducted is related to the numerical errors of computation of the velocity and stretching term. Also for comparisons we executed the FMM for a single harmonic function and measured numerical errors in potential and its gradient. There are two basic sources of the errors: first, the FMM errors, due to truncation of the infinite series. These errors are controlled by the truncation number p (so infinite series (29) were replaced by the first p^2 terms, $n = 0, ..., p-1$; $m = -n, ..., n$), which we varied in the tests. Second, the roundoff errors, which in our computations with double precision in the range of tested p were smaller than the truncation errors (the roundoff errors were observed for potential computations at $p \gtrsim 25$). The basic test was performed for N sources/receivers distributed randomly and uniformly inside a cube. The error, ϵ_2, was measured in the L_2 relative norm based on 1000 points randomly selected from the source set (our previous tests with the number of reference points using direct computations up to 100,000 points show that even 100 points provide sufficient confidence for the L_2-norm error, see [11]). For the reference solution the velocity field, stretching term, potential and gradient were computed directly.

Figure 1 illustrates behavior of the computed errors for the velocity and stretching term. For reference dependences of the respective errors in the harmonic potential and in its gradient are shown. It is seen that starting with $p \approx 7$ spectral convergence is observed for all cases. It is also noticeable, that the errors in potential computations are substantially smaller than that for the gradient or higher derivative computations. There are two basic reasons why this happens. First, the effective truncation number for each derivative is smaller by one compared to the potential, and second that in the truncated term for the derivative an additional factor $\sim p$ appears (indeed, one can imagine series over monomials x^n, which being truncated to the first p terms provide error $O(x^p)$; differentiation of this series provides nx^{n-1} monomials and the error bound $O\left(px^{p-1}\right)$ for the same x and p) (we checked also this numerically and found consistency of the orders of the measured errors with this explanation). In any case the difference in the errors between the gradient computations and those for the velocity/stretching is not very large and some increase of the truncation number can compensate such a difference.

5.2 Performance tests

Peculiarity of the FMM is that for different problems different depth of the octrees, $l_{\max}$, should be used to minimize the total execution time. So for all reported test cases we conducted optimization study and profiling of the algorithm. Some results of profiling (wall clock time in seconds) for optimized benchmark cases with random uniform distributions of sources inside a cube and on the surface of a sphere are provided in Tables 1 and 2. In these tables $\mathbf{v}$ and $\mathbf{s}$ mark computations of the velocity and stretching term for vortical flows, while ϕ and $\nabla\phi$ refer to a reference case for the scalar Laplace equation, where the potential or both potential and its gradient should be computed. The total initialization time, which includes setting of

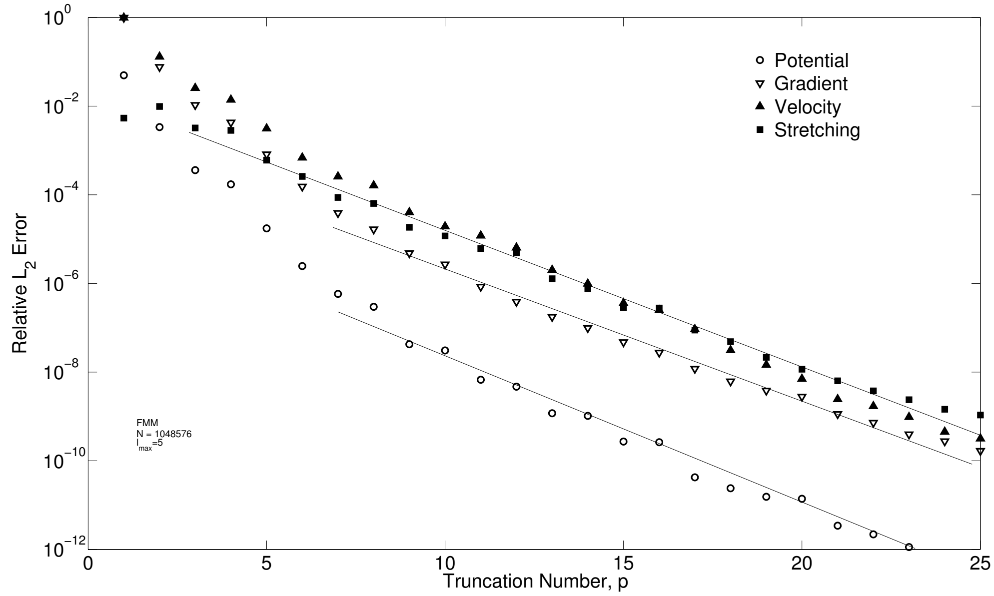

Figure 1: Dependences of the relative FMM errors in the L_2 norm on the truncation number, p. Errors were computed over 1000 random points for $N = 2^{20}$ sources of random intensity distributed uniformly randomly inside a cube. The maximum level of space subdivision $l_{\max} = 5$.

the data structure and precomputations related to translation operators and can be amortized for a constant source/receiver set but for different input vectors is reported separately from the total run time. As one can see this time is relatively small, while for dynamic problems it should be added to the total run time. Tables also separate the time for sparse and dense matrix-vector products, produced by the FMM. The latter is also expanded to show timing of the basic FMM stages, which include generation of the multipole expansions, upward and downward passes involving translations, and evaluation of the local expansions. Truncation number for all cases was $p = 12$, which provides errors $\epsilon_2 \sim 10^{-5}$ for the velocity and stretching term computations, while smaller errors for ϕ and $\nabla\phi$ (see Fig. 1).

Table 1: Profiling of the FMM for random uniform distribution of sources inside a cube, $p = 12$.

Case	$l_{\max}$	Total Init	S-expansion	Upward	Downward	R-evaluation	Sparse MV	Total Run
				$N = 2^{19}$				
$\mathbf{v}$ and $\mathbf{s}$	4	0.55	0.55	0.04	2.65	0.52	25.9	29.7
$\mathbf{v}$ alone	4	0.55	0.55	0.04	2.65	0.34	16.4	20.0
ϕ and $\nabla\phi$	5	1.20	0.20	0.12	10.7	0.36	3.95	15.3
ϕ alone	5	1.20	0.20	0.12	10.7	0.21	1.89	13.1
				$N = 2^{20}$				
$\mathbf{v}$ and $\mathbf{s}$	5	1.71	1.05	0.27	25.3	1.11	14.2	41.9
$\mathbf{v}$ alone	5	1.71	1.05	0.27	25.3	0.59	9.04	36.3
ϕ and $\nabla\phi$	5	1.71	0.39	0.12	10.7	0.71	15.4	27.3
ϕ alone	5	1.71	0.39	0.12	10.7	0.43	7.32	19.0

The tables show that in the cases when the number of translations for single potential ϕ for the scalar Laplace equations and coupled potentials ϕ and χ for the DCVLE is the same (the same $l_{\max}$) the translation

Table 2: Profiling of the FMM for random uniform distribution of sources on a sphere surface, $p = 12$.

Case	$l_{\max}$	Total Init	S-expansion	Upward	Downward	R-evaluation	Sparse MV	Total Run
				$N = 2^{19}$				
$\mathbf{v}$ and $\mathbf{s}$	7	1.23	0.88	0.34	9.48	0.66	3.75	15.1
$\mathbf{v}$ alone	6	0.66	0.63	0.10	2.46	0.34	8.66	12.2
ϕ and $\nabla\phi$	7	1.23	0.23	0.15	3.93	0.41	3.90	8.62
ϕ alone	7	1.23	0.23	0.15	3.93	0.26	1.86	6.43
				$N = 2^{20}$				
$\mathbf{v}$ and $\mathbf{s}$	7	1.81	1.29	0.34	9.48	1.30	14.1	26.5
$\mathbf{v}$ alone	7	1.81	1.29	0.34	9.48	0.70	8.86	20.7
ϕ and $\nabla\phi$	7	1.81	0.46	0.15	4.14	0.84	15.3	20.9
ϕ alone	7	1.81	0.46	0.15	4.14	0.52	7.30	12.6

time for the latter case approximately 2 times larger that is an expected result. The translation time ratio, in fact, is slightly larger than 2, which can be explained by two factors. First, this is due to increase in the size of the arrays representing expansions and more time needed for data access, and, second, by the presence of the conversion operators. The tables also show that the time for sparse matrix-vector products for velocity only computations in DCVLE is slightly larger than for potential only computations in a harmonic FMM, while the time for the same operations for velocity and stretching computations are slightly smaller than for potential and gradient computations. Note, however, that if an additional near-field kernel should be computed, which may involve computation of special functions (exponents, error integrals, etc.) the time for the sparse matrix-vector product would increase, while the translation part would not be affected. Also note that, theoretically, in the optimized algorithm increase of the complexity of the sparse matrix-vector product k times affects the total complexity as $\sqrt{k}$. The ratio of the total time for the velocity and stretching computations to the time of potential and gradient computations depends on the problem size and source/receiver distributions, but in all our numerical experiments this ratio never exceeded 2 (except of one outlier at $N = 1024$, see Fig. 3), while in average it was about 1.5. Finally, for the same maximum level of space subdivision, we can see that both most expensive parts contributing to the total complexity, translations and sparse matrix-vector product, for the velocity and stretching computations approximately 2 times more expensive than that for a single harmonic potential. So, a factor 2 is expected for this ratio. Slightly larger ratios were observed in the numerical tests due to other parts of the algorithm and overheads mentioned above.

Figure 2 illustrates dependence of the wall clock time on the number of sources N, which in all cases was set to be equal to the number of receivers. This is the case for uniform random distributions inside the cube. It is seen that at large N the algorithm scales linearly, and the time for velocity and stretching term computations is always larger than that for scalar potential only computations approximately two times.

Figure 3 illustrates the wall clock FMM run time ratio of velocity and stretching to potential and gradient computations for different p and $N = 2^k$, $k = 10, ..., 20$. It is seen that for $k > 11$ this ratio is larger than 1 and smaller than 2 with the mean value about 1.5. Oscillations between 1 and 2 can be explained by optimization of the octree depth which was independent for each case reported.

6 Conclusion

The main goal of this study was to develop an efficient method for fast summation of elementary vortices. Numerical tests confirm the validity of the theory presented and efficiency of the method. Our results show that one should expect approximately 2 times increase of the computation time for velocity and vorticity stretching term computations compared to a single harmonic function computations for the same

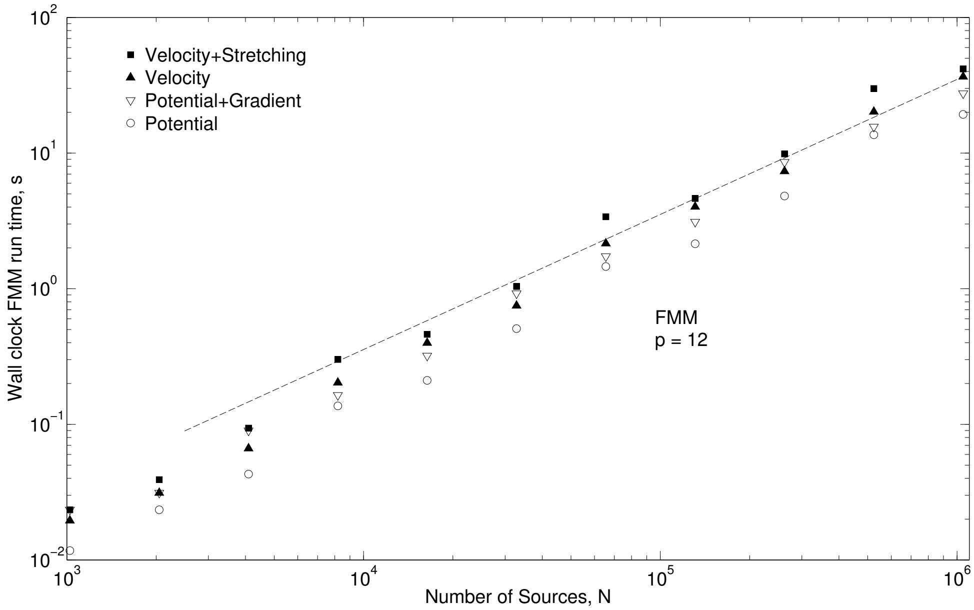

Figure 2: The wall clock time for the FMM for computation of vortical and potential flows (different terms and compinations). The straight line shows linear dependence. For all cases the sources are distributed randomly and uniformly inside a cube; the truncation number is constant, $p = 12$.

FMM octree and the truncation number. Compared to potential and gradient computations for the scalar Laplace equation this increase varies in the range from 1 to 2 times (average 1.5) depending on particular source/receiver distribution, truncation number, etc.

An interesting observation from the study is that a general reconstruction of vector fields from given curl and divergence can be obtained via the present method, which operates only with two scalar potentials. This may have application to many other fields of physics, including plasma physics, electromagnetism, etc. In this sense the DCVLE appears to be a fundamental equation, the solutions of which can be accelerated via the harmonic FMM. As is shown, modifications of standard FMM programs are relatively easy, and require tracking of two harmonic functions, instead of one, and implementation of the conversion operators used in each translation. Such operators are very sparse and simple (especially for the case of the RCR-decomposition) and their execution does not create substantial overheads. In terms of further acceleration of computations it is natural to consider implementations of the method on graphics processors (GPUs) for which the vortex methods are developed and tested (e.g. [20]).

7 Acknowledgement

Work partially supported by AFOSR under MURI Grant W911NF0410176 (PI Dr. J. G. Leishman, monitor Dr. D. Smith); and by Fantalgo, LLC.

References

[1] G.K. Batchelor. *An Introduction to Fluid Dynamics*, Cambridge University Press, Cambridge, 1967.

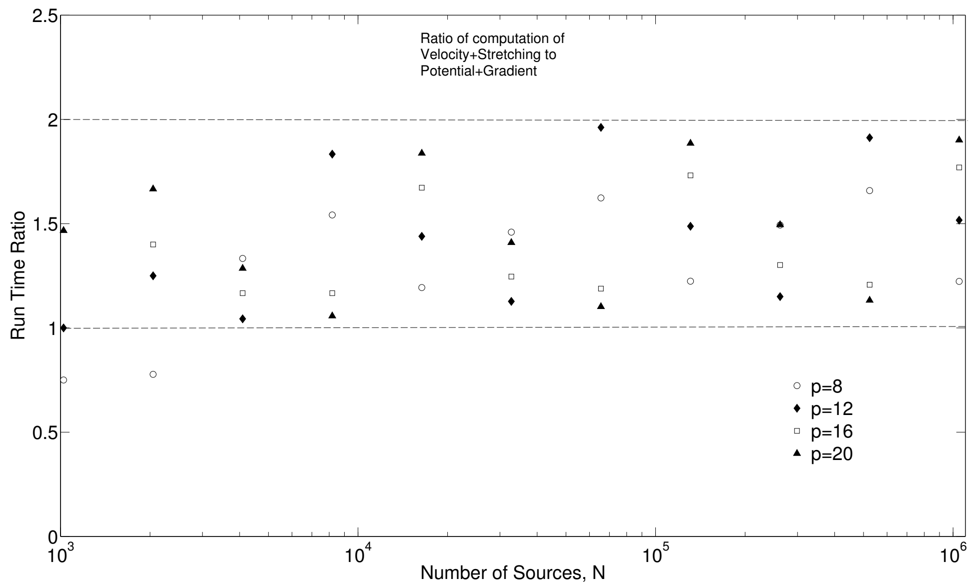

Figure 3: The ratio of the FMM run time for computation of the velocity and stretching term in a vortical flow to the respective time for the potential and gradient computations in a potential flow for different truncation numbers p and number of sources, N, which were randomly and uniformly distributed inside a cube. All data for $N > 2048$ are located between the dashed lines.

[2] H. Cheng, L. Greengard and V. Rokhlin. "A Fast Adaptive Multipole Algorithm in Three Dimensions," Journal of Computational Physics, 155:468–498, 1999.

[3] J.D. Eldredge, T. Colonius, and A. Leonard. "A vortex particle method for two-dimensional compressible flow," J. Comput. Phys., 179:371-399, 2002.

[4] M A. Epton and B. Dembart. "Multipole translation theory for the three-dimensional Laplace and Helmholtz equations," SIAM Journal on Scientific Computing archive, 16:865-897, 1995

[5] Y. Fu and G.J. Rodin, "Fast solution methods for three-dimensional Stokesian many-particle problems," Commun. Numer. Meth. En., 16:145-149, 2000.

[6] L. Greengard and V. Rokhlin. "A fast algorithm for particle simulations," Journal of Computational Physics, 73:325-348, 1987.

[7] L. Greengard and V. Rokhlin. "A new version of the Fast Multipole Method for the Laplace equation in three dimensions," Acta Numerica 6:229-269, 1997.

[8] N. A. Gumerov and R. Duraiswami. "Recursions for the computation of multipole translation and rotation coefficients for the 3-D Helmholtz equation," SIAM Journal on Scientific Computing, 25:1344–1381, 2003.

[9] N. A. Gumerov and R. Duraiswami. *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions.* Elsevier Science, Amsterdam, 2005. ISBN: 0080443710.

[10] N. A. Gumerov and R. Duraiswami. “Comparison of the efficiency of translation operators used in the fast multipole method for the 3D Laplace equation,” Technical Report CS-TR-4701 and UMIACS-TR 2005-09, University of Maryland Department of Computer Science and Institute for Advanced Computer Studies, Nov 2005.

[11] N. A. Gumerov and R. Duraiswami. “Fast Multipole method for the biharmonic equation in three dimensions,” Journal of Computational Physics, 215:363–383, 2006.

[12] N. A. Gumerov and R. Duraiswami. “A scalar potential formulation and translation theory for the time-harmonic Maxwell equations,” Journal of Computational Physics, 225:206–236, 2007.

[13] N. A. Gumerov and R. Duraiswami. “Fast Multipole Methods on Graphics Processors,” Journal of Computational Physics, 227:8290-8313, 2008.

[14] Q. Hu, M. Syal, N. A. Gumerov, R. Duraiswami, and J. G. Leishman. “Toward improved aeromechanics simulations using recent advancements in scientific computing, ” in Proceedings 67th Annual Forum of the American Helicopter Society, May 3-5 2011.

[15] P. Koumoutsakos. “Multiscale flow simulations using particles,” Annu. Rev. Fluid Mech. 2005. 37:457–87

[16] H. Lamb, *Hydrodynamics*, Cambridge University Press, Cambridge, 1932.

[17] V C. Raykar , C. Yang , R. Duraiswami, and N. A. Gumerov. “Fast computation of sums of Gaussians in high dimensions,” Technical Report CS-TR-4767, Department of Computer Science, University of Maryland, College Park, 2005.

[18] A.K. Tornberg and L. Greengard, “A fast multipole method for the three-dimensional Stokes equations,” Journal of Computational Physics, 227:1613-1619, 2008.

[19] G.Winckelmans, R. Cocle, L. Dufresne, and R. Capart. “ Vortex methods and their application to trailing wake vortex simulations,” C.R. Physique 6:467-486, 2005.

[20] R. Yokota, T. Narumi, R. Sakamaki, S. Kameoka, S. Obi, K. Yasuoka. “Fast multipole methods on a cluster of GPUs for the meshless simulation of turbulence,” Computer Physics Communications 180:2066-2078, 2009.