Cauchy Fast Multipole Method for General Analytic Kernels

Pierre-David Létourneau, [a] Cris Cecka, [b] and Eric Darve [a,c]

[a] *Institute for Computational and Mathematical Engineering, Stanford University*
[b] *Institute for Applied Computational Science, Harvard University*
[c] *Mechanical Engineering Department, Stanford University*

Abstract

The fast multipole method (FMM) is a technique allowing the fast calculation of long-range interactions between N points in $\mathcal{O}(N)$ or $\mathcal{O}(N \log N)$ steps with some prescribed error tolerance. The FMM has found many applications in the field of integral equations and boundary element methods, in particular by accelerating the solution of dense linear systems arising from such formulations. Standard FMMs are derived from analytical expansions of the kernel, for example using spherical harmonics or Taylor expansions. In recent years, the range of applicability and the ease of use of FMMs has been extended by the introduction of black box and kernel independent techniques. In these approaches, the user only provides a subroutine to numerically calculate the interaction kernel. This allows changing the definition of the kernel with minimal changes to the computer program. This paper presents a novel kernel independent FMM, which leads to diagonal multipole-to-local operators. The result is a significant reduction in the computational cost, in particular when high accuracy is needed. The approach is based on Cauchy's integral formula and the Laplace transform. We will present a numerical analysis of the convergence and numerical results in the case of a multilevel level one-dimensional FMM.

Key words: fast multipole method, integral equations, boundary element method, dense matrix-vector product

1. Introduction

The fast multipole method (FMM) is a general class of methods used to reduce the cost of computing sums of the form:

Email addresses: pletou1@stanford.edu (Pierre-David Létourneau,), ccecka@seas.harvard.edu (Cris Cecka,), darve@stanford.edu (Eric Darve).

$$\phi_i = \sum_{j=1}^{N} K(x_i, y_j)\, \sigma_j, \quad 1 \leq i \leq N \tag{1}$$

when N is large. The original FMM was first introduced by Greengard and Rokhlin [1] and subsequently improved over the years. The basic approximation strategy in the FMM is a low-rank approximation of the kernel of the type:

$$K(x, y) = \sum_{m=1}^{N} \sum_{n=1}^{N} u_m(x)\, T_{mn}\, v_n(y) + \varepsilon \tag{2}$$

With such an approximation, one can construct an $\mathcal{O}(N)$ or $\mathcal{O}(N \log N)$ method to calculate the sum in Eq. (1).

Fast multipole methods have been derived for many different types of kernels including the Laplace kernel $1/r$ [1–3], the Helmholtz kernel $e^{\imath\kappa r}/r$ [4–9], and the Stokes kernel [10], among others. Efforts have also been made to extend the method to more general kernels in a black-box fashion [11–17].

For instance in [12], Legendre polynomials are used to represent the action of the kernel $K(x, y)$ in various regions of the computational domain. The singular value decomposition (SVD) is then used to obtain an optimal low-rank representation of the latter, which is applied to construct a fast algorithm. The computation of the SVD is crucial in obtaining optimal running time, but can result in a costly pre-computation step especially when large 3D domains are considered.

Fong et al. [16] use Chebyshev polynomials to represent the action of the kernel $K(x, y)$ within the computational domain. However, instead of the full SVD, they use techniques based on Chebyshev interpolation and SVD compression to transfer information along the FMM tree. This results in a significantly less expensive pre-computation step and still exhibits running times comparable to the previous method. A similar scheme for 1D problems is also discussed [15].

Another kind of black-box fast multipole method is presented in Ying et al. [13]. Such method relies on the fact that the solution of an elliptic partial differential equation in a sufficiently well-behaved closed region is characterized by the value of the solution on the boundary of that region. This implies that effects of source points within a region can be represented within the complement of this region through an equivalent surface density. Such equivalent densities can then be used to transfer information along the FMM tree efficiently and create a fast method. Each transfer of information involves the numerical solution of a discretized Fredholm equation of the first kind (normal equations with Tikhonov regularization), which can be made fast through the use of various techniques.

In all the aforementioned methods, the action of the kernel between two separate regions is expressed through a *small but dense* matrix of size r, and the cost of transferring information along the FMM tree scales like r^2. Such method are fast but tend to slow down when high accuracy is desired, i.e., as r increases. One way to alleviate this problem is to transfer information through the use of *diagonal* operators.

This has been attempted in [17] for instance. In this paper, the authors extend the domain of the kernel in such a way that it becomes smooth at the boundary. Then they use a Fourier basis (complex exponentials) and an adaptive sampling scheme combined with a weighted least-square to obtain a low-rank representation of the kernel on this extended domain. The use of complex exponentials as basis functions results in operators

that are diagonal.

In this paper, we consider the creation of an FMM with two goals in mind:

(i) The multipole-to-local (M2L) operator T_{mn} should be *diagonal*.

(ii) The method should be applicable to a wide class of kernels.

The method we are proposing derives low-rank expressions from the Cauchy's integral formula and Laplace tranform. It is applicable to all functions analytic in the domain of interest. We will also obtain explicit bounds for the error incurred in our approximation. This in turn implies that a low-rank approximation can be obtained in a computationally efficient manner.

Some of the applications we have in mind include the use of radial basis functions such as r, r^n (n odd), $r^n \log r$ (n even), $\exp(-cr^2)$, $\sqrt{r^2+c^2}$, $1/\sqrt{r^2+c^2}$, $1/(r^2+c^2)$, ..., for interpolation schemes. These are popular schemes for mesh deformation [18] and graphics applications [19]. Interpolation using these functions requires generalized FMMs capable of handling a wide class of functions. Our technique is also applicable to Green's functions associated with partial differential equations such as $1/r$, e^{ikr}/r, or kernels associated with Stokes' equations or elasticity equations in solid mechanics. In fact, the technique is applicable to any kernel of the form $g(x)e^{\imath k|x|}$ where $g(\cdot)$ is smooth by following a procedure similar to that described in [20].

The expansion used in the paper relies on exponential functions in the complex plane and as such is related to the plane wave acceleration of Greengard et al. ("A new version of the Fast Multipole Method for the Laplace equation in three dimensions," 1997 [21]) and Beylkin et al. [22,23]. In some sense our method proposes an explicit construction of the complex exponentials whereas Beylkin's approach relies on exponentials that minimize the L^2 error, and it extends the result of Greengard to more general kernels. The proposed method proceeds in this way:

(i) The analytic kernel $K(x-y)$ is written as a Cauchy integral that is accurately discretized using a trapezoid rule to yield an approximation of the form [Eq. (6)]

$$K(x-y) = \frac{1}{N\imath} \sum_{n=0}^{N-1} \frac{K(z_n)\, z'(\theta_n)}{z(\theta_n) - (x-y)} + \varepsilon_N$$

where $z(\theta)$ is a parameterization of the smooth contour chosen for the Cauchy integral, $\theta_n = 2\pi n/N$.

(ii) Each term $1/(z(\theta_n)-(x-y))$ is represented using an appropriate Laplace integral.

(iii) After a change of variables, the Laplace integrals are accurately discretized using an appropriate quadrature.

These steps yield a kernel $K(x-y)$ as a sum of separable functions which, with proper parameters, is displayed in Eq. (17).

Some of the key points in the paper are as follows:

- We derive the method using the Cauchy integral formula, the Laplace transform of $1/z$ and appropriate quadratures to discretize integrals.
- We discuss how to choose various parameters in the method such as the contour in the complex plane and the number of quadrature points needed.
- We perform an error analysis that allows bounding the error as a function of the chosen parameters.
- We include numerical benchmarks and a comparison with the black-box FMM of [16].

The organization of the paper is as follows: Section 2 introduces the method and the underlying theoretical ideas. Section 3 presents details regarding the choice of contour in the complex plane. Section 4 introduces the multi-level scheme and a numerical analysis is presented in Section 5. Numerical results demonstrating the performance of the method on various kernels are presented in Section 6.

2. Cauchy's integral formula and low-rank approximations

For simplicity, let us assume that $K(x, y)$ is translationally invariant so it can be expressed as $K(x-y)$. This assumption is not necessary to reach the $\mathcal{O}(N \log N)$ complexity however it will affect the pre-computation time. Since the case of kernels that are not translationally invariant is not very common in FMM studies, we will not explore it in this paper. We will also focus on the one dimensional (1D) case. At the end of this section, we briefly explain how extensions to 2D and 3D can be achieved.

We consider a kernel $K(x - y)$, with $x - y \in \mathbb{R}$, and assume that, in some region $\Omega \subset \mathbb{C}$ that contains the point $x - y$, it is an analytic (holomorphic) function. That is, the function $K(z)$, $z \in \mathbb{C}$, is complex differentiable at every point $z \in \Omega$. By Cauchy's integral formula,

$$K(x - y) = \frac{1}{2\pi\imath} \oint_\Gamma \frac{K(z)}{z - (x - y)} \, \mathrm{d}z \tag{3}$$

where the curve $\Gamma \subset \Omega \subset \mathbb{C}$ is simple, closed and contains $x - y$.

When $\mathrm{Re}(z - (x - y)) > 0$, the Laplace transform is well-defined and one can write,

$$\frac{1}{z - (x - y)} = \int_0^\infty e^{-s(z-(x-y))} \, \mathrm{d}s \tag{4}$$

substitution into (3) would allow for the algebraic separation of x and y. On approximating these integrals using an appropriate numerical quadrature, a low-rank approximation of the form (2) is obtained and may be used to develop a fast method for computing (1) with any analytic kernel, K.

Since Γ encloses $x - y$, it cannot be the case that $\mathrm{Re}(z - (x - y)) > 0$ for all $z \in \Gamma$. However, this can be remedied by partitioning the contour into connected segments and rotating each segment in the complex plane. Define $\{\Gamma_k\}$ to be a set of open, connected segments such that $\Gamma = \cup_k \Gamma_k$, $\emptyset = \Gamma_i \cap \Gamma_j \, (i \neq j)$ and for each k there exists $\lambda_k \in \mathbb{C}$ such that for all $z \in \Gamma_k$,

$$\mathrm{Re}(\lambda_k(z - (x - y))) > 0 \tag{5}$$

This can be achieved in different ways. Consider for example an elliptic contour partitioned into four segments as illustrated in Fig. 1. Each segment satisfies

$$\begin{aligned}
&\mathrm{Re}(z - (x - y)) > 0 \qquad \forall\, z \in \Gamma_1 \\
&\mathrm{Im}(z - (x - y)) > 0 \qquad \forall\, z \in \Gamma_2 \\
&\mathrm{Re}(z - (x - y)) < 0 \qquad \forall\, z \in \Gamma_3 \\
&\mathrm{Im}(z - (x - y)) < 0 \qquad \forall\, z \in \Gamma_4
\end{aligned}$$

and the Laplace transform may be applied piece-wise to the rotated segments ($\lambda_k = \{1, -\imath, -1, \imath\}$, respectively) to yield

$$\frac{1}{z-(x-y)} = \begin{cases} \int_0^\infty e^{-s(z-(x-y))}\,\mathrm{d}s & \mathrm{Re}(z-(x-y)) > 0 \\ -\imath \int_0^\infty e^{\imath s(z-(x-y))}\,\mathrm{d}s & \mathrm{Im}(z-(x-y)) > 0 \\ -\int_0^\infty e^{s(z-(x-y))}\,\mathrm{d}s & \mathrm{Re}(z-(x-y)) < 0 \\ \imath \int_0^\infty e^{-\imath s(z-(x-y))}\,\mathrm{d}s & \mathrm{Im}(z-(x-y)) < 0 \end{cases}$$

Note that any simple contour composed of three or more segments could potentially be used. In this paper, we continue with the four segment ellipse above for convenience and simplicity.

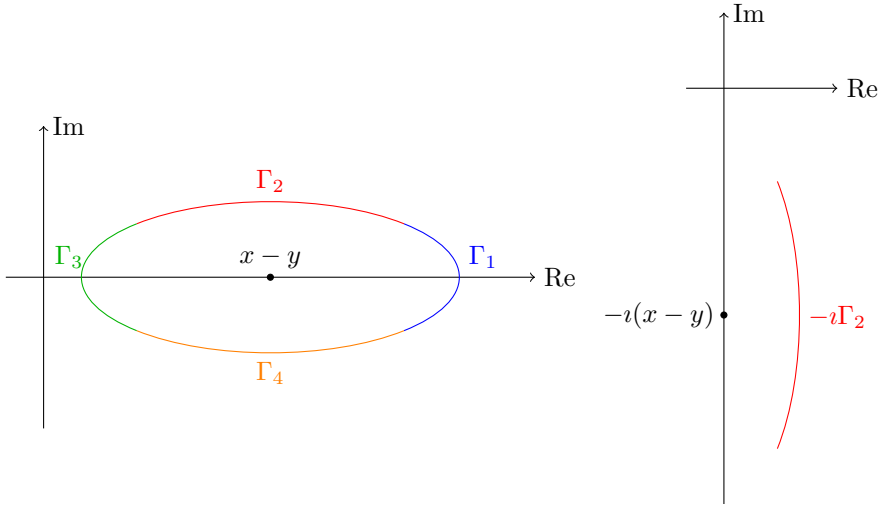

Fig. 1. Schematic of an elliptic contour around $x-y$. Left: elliptic contour decomposed into four paths. Right: example of a rotation (multiplication by $\lambda_2 = -\imath$) of path Γ_2. Notice that $\mathrm{Re}(-\imath(z-(x-y))) > 0$, $\forall\, z \in \Gamma_2$.

2.1. *Construction of the method*

2.1.1. *Discretization of the Cauchy integral*

Consider the Cauchy integral (3) parameterized by θ:

$$K(x-y) = \frac{1}{2\pi\imath} \int_0^{2\pi} \frac{K(z(\theta))\, z'(\theta)}{z(\theta)-(x-y)}\,\mathrm{d}\theta.$$

When the contour is smooth and the kernel analytic, the integrand is *smooth* and *periodic* in θ. A trapezoid rule can then be used to efficiently discretize this integral. Letting $\theta_n = \frac{2\pi n}{N}$, $z_n = z(\theta_n)$,

$$K(x-y) = \frac{1}{N\imath} \sum_{n=0}^{N-1} \frac{K(z_n)\, z'(\theta_n)}{z_n-(x-y)} + \varepsilon_N \tag{6}$$

where ε_N is the error incurred by using a finite uniform quadrature. The error ε_N can be shown to decay exponentially as a function of N (see Section 5) and is bounded by an expression of the form

$$|\varepsilon_N| \le \mathcal{A} e^{-\mathcal{B}N} \tag{7}$$

where $\mathcal{A}, \mathcal{B} > 0$ depend on the contour Γ and the kernel K.

2.1.2. *Discretization of the Laplace integral*

Next, consider the Laplace integral for a given point $z_n = z_n \in \Gamma_k$:

$$\frac{1}{z_n - (x-y)} = \lambda_k \int_0^\infty e^{-s\lambda_k(z_n-(x-y))}\,\mathrm{d}s \tag{8}$$

Making the change of variables $s \to s^2$ simplifies the error analysis and allows to construct simple numerical quadratures,

$$= \lambda_k \int_{-\infty}^\infty |s|\, e^{-s^2\lambda_k(z_n-(x-y))}\,\mathrm{d}s$$

To construct a fast method, we build a tree of boxes (see Section 4) which accumulate fields and interact. Each box stores a field representing the outgoing/incoming influence of all particles inside/outside of it. See Figure 2. Using the center of boxes in the 1D case, we make the substitution $x - y = (x - c_x) + (c_x - c_y) + (c_y - y) \equiv r_x + r_0 + r_y$,

$$\frac{1}{z_n - (x-y)} = \lambda_k \int_{-\infty}^\infty |s|\, e^{-s^2\lambda_k(z_n-r_0)} e^{s^2\lambda_k r_x} e^{s^2\lambda_k r_y}\,\mathrm{d}s \tag{9}$$

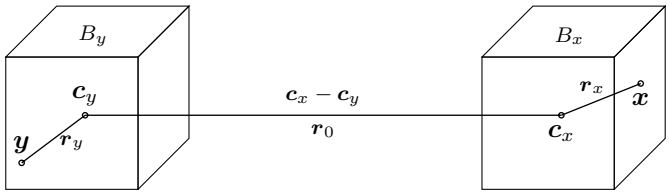

Fig. 2. Schematic of transfer between two particles belonging to two boxes in 3D. Recall that this paper focuses primarily on the 1D case.

Although the integrand is bounded due to the condition (5), the individual terms of the integrand may not be bounded. In order to ensure numerical stability, we introduce a set of real-valued numerical stabilization parameters $\{\ell_k\}$ chosen suitably so that for all $z_n \in \Gamma_k$, $x \in B_x$, $y \in B_y$, each term in the integrand is bounded. That is,

$$\begin{aligned} \mathrm{Re}(\lambda_k(z_n - r_0) - 2\ell_k) &> 0 \\ \mathrm{Re}(\ell_k - \lambda_k r_x) &> 0 \\ \mathrm{Re}(\ell_k - \lambda_k r_y) &> 0 \end{aligned} \tag{10}$$

In order for such ℓ_k to exist, the condition

$$\mathrm{Re}(\lambda_k(z_n - r_0)) > \mathrm{Re}(\lambda_k(r_x + r_y))$$

must be satisfied for all $x \in B_x$ and $y \in B_y$. This is guaranteed when (5) is modified to read

$$\min_{x\in B_x,\, y\in B_y} \mathrm{Re}(\lambda_k(z - (x-y))) > 0 \tag{11}$$

for all $z \in \Gamma_k$. This condition is natural since it can be satisfied (with an appropriate choice of Γ_k) whenever the contour encloses the segment containing all $x - y$, $x \in B_x$, $y \in B_y$. In all cases, the following choice is found to be adequate:

$$\ell_k = \frac{1}{2} \min_{z_n \in \Gamma_k, r_0} \mathrm{Re}(\lambda_k(z_n - r_0)). \tag{12}$$

With these parameters, Eq. (9) can be re-written as

$$\int_{-\infty}^{\infty} |s|\, e^{-s^2(\lambda_k(z_n - r_0) - 2\ell_k)}\; e^{-s^2(\ell_k - \lambda_k r_x)}\; e^{-s^2(\ell_k - \lambda_k r_y)}\, \mathrm{d}s$$

Noting that the integrand is dominated by complex-Gaussians that decay rapidly, we truncate the infinite domain of integration to a finite interval to yield

$$= \int_{-L_k}^{L_k} |s|\, e^{-s^2(\lambda_k(z_n - r_0) - 2\ell_k)} e^{-s^2(\ell_k - \lambda_k r_x)} e^{-s^2(\ell_k - \lambda_k r_y)}\, \mathrm{d}s + \varepsilon_L^k \tag{13}$$

The error ε_L^k incurred through the truncation can be shown to be bounded by an expression of the form,

$$\left|\varepsilon_L^k\right| \leq \mathcal{C} e^{-\mathcal{D} L_k^2} \tag{14}$$

where $\mathcal{C}, \mathcal{D} > 0$ are numerical constants (see Section 5). This is essentially a direct consequence of the decay of the complex-Gaussian functions. This is now in a form suitable for discretization.

Grouping the terms that do not depend on x and y, we define the *transfer function* as

$$T_{k,n}(s) = |s|\, e^{-s^2(\lambda_k(z_n - r_0) - 2\ell_k)}$$

The key observation to determine an appropriate quadrature is that the decay of the integrand is largely controlled by the two complex-Gaussian terms $e^{-s^2(\ell_k - \lambda_k r_x)}$ and $e^{-s^2(\ell_k - \lambda_k r_y)}$, which rapidly decay in Fourier space. Given some accuracy ε, they may be approximated by bandlimited functions, with only modes in the interval $[-P_k, P_k]$ contributing. Frequencies of $T_{k,n}(s)$ that fall outside of this interval do not make significant contributions to the integral by Parseval's theorem (with an error on the order of ε). Consequently, such frequencies of $T_{k,n}(s)$ can be removed. This is possible because the Fourier series coefficients of $T_{k,n}(s)$ can be computed analytically. In this sense, if we let,

$$f_k(s) = e^{-s^2(\ell_k - \lambda_k r_x)}\, e^{-s^2(\ell_k - \lambda_k r_y)}$$

then Eq. (13) can be written as

$$\int_{-L_k}^{L_k} T_{k,n}(s) f_k(s)\, \mathrm{d}s = \int_{-L_k}^{L_k} \left(\sum_{p=-\infty}^{\infty} \hat{T}_{k,n}[p] e^{\pi \imath p s / L_k} \right) \left(\sum_{p=-\infty}^{\infty} \hat{f}_k[p] e^{\pi \imath p s / L_k} \right) \mathrm{d}s$$

where $\hat{T}_{k,n}[p]$ and $\hat{f}_k[p]$ are the pth Fourier coefficient of $T_{k,n}(s)$ and $f_k(s)$ respectively. Truncating the transfer function in Fourier space incurs the error ε_P^k,

$$\int_{-L_k}^{L_k} T_{k,n}(s) f_k(s)\,\mathrm{d}s =$$
$$= \int_{-L_k}^{L_k} \left(\sum_{p=-P_k}^{P_k} \hat{T}_{k,n}[p] e^{\pi\imath ps/L_k} \right) \left(\sum_{p=-\infty}^{\infty} \hat{f}_k[p] e^{\pi\imath ps/L_k} \right) \mathrm{d}s + \varepsilon_P^k$$

by Parseval's theorem this is equivalent to

$$= \int_{-L_k}^{L_k} \left(\sum_{p=-P_k}^{P_k} \hat{T}_{k,n}[p] e^{\pi\imath ps/L_k} \right) \left(\sum_{p=-P_k}^{P_k} \hat{f}_k[p] e^{\pi\imath ps/L_k} \right) \mathrm{d}s + \varepsilon_P^k \tag{15}$$

The spectrum of the integrand is now bandlimited to the interval $[-2P_k; 2P_k]$. Thus, a $(2P_k + 1)$-point trapezoid rule is sufficient to compute this integral exactly. Using symmetries of the integrand, we can reduce this number to P_k.

The numerical analysis in Section 5 shows that by using the low-pass transfer function $T_{k,n}^{lb}(s)$ with a $(2P_k + 1)$-point trapezoidal quadrature rule along each path Γ_k, the error ε_P^k is bounded by,

$$\left|\varepsilon_P^k\right| \leq \mathcal{E} e^{-\mathcal{F}\frac{P_k^2}{L_k^2}} \tag{16}$$

where $\mathcal{E}, \mathcal{F} > 0$ are numerical constants. The sum of the integers P_k, $k = 1, \ldots, 4$ is the total number of quadrature points or rank of the approximation in the FMM.

We also note that since the analysis leads to explicitly computable quantities, it is possible to perform numerical benchmarks to determine accurately which values of P_k and L_k one should use. This analysis is therefore not simply of theoretical interest but can be used in practice when implementing the method to control the error.

2.1.3. *Low-rank method*

Combining the approximations obtained in Eq. (6) and Eq. (15) leads to the formula

$$K(x-y) \approx \sum_k \left(\sum_{p=0}^{P_k} e^{-s_{k,p}^2(\ell_k - \lambda_k r_x)}\ \mathcal{T}_{k,p}\ e^{-s_{k,p}^2(\ell_k - \lambda_k r_y)} \right) \tag{17}$$

analogous to (2) where the nodes and weights are given by,

$$s_{k,p} = p\frac{L_k}{P_k}$$

$$\mathcal{T}_{k,p} = [2 - (\delta_0(p) + \delta_P(p))] \sum_{\{n\,|\,z_n \in \Gamma_k\}} \frac{\lambda_k}{N\imath}\ K(z_n)\ z'(\theta_n)\ T_{k,n}^{lb}(s_{k,p})$$

$$T_{k,n}^{lb}(s_{k,p}) = \sum_{p=-P_k}^{P_k} \hat{T}_{k,n}[p] e^{\pi\imath ps/L_k}$$

and $\delta_k(p)$ is equal to 1 if $k = p$ and 0 otherwise. The Fourier coefficients of the transfer function can be computed analytically and the values $T_{k,n}^{lb}(s_{k,p})$ may be computed rapidly through a fast Fourier transform (FFT).

With this, the total error, ε_T, can be bounded by

$$|\varepsilon_T| \leq \sum_k \left(\left[\sum_{\{n \,|\, z_n \in \Gamma_k\}} \frac{1}{2\pi N} \left| K(z(\theta_n))\, z'(\theta_n) \right| \right] \left(\left|\varepsilon_P^k\right| + \left|\varepsilon_L^k\right| \right) \right) + |\varepsilon_N| \tag{18}$$

From this bound and the estimates presented in Eq. (7), Eq. (14) and Eq. (16), we conclude that for a target accuracy of ε, the parameters N, L_k and P_k should be chosen such that,

$$N = \mathcal{O}\left(\log\left(1/\varepsilon\right)\right) \tag{19}$$

$$L_k = \mathcal{O}\left(\log^{1/2}\left(\max_{z\in\Gamma_k} \left|K(z)\right| / \varepsilon\right)\right) \tag{20}$$

$$P_k = \mathcal{O}\left(\log\left(\max_{z\in\Gamma_k} \left|K(z)\right| / \varepsilon\right)\right) \tag{21}$$

Thus, since the rank of the approximation depends only on P_k, it grows like

$$\log\left(\max_{z\in\Gamma_k} \left|K(z)\right| / \varepsilon\right)$$

Although the rank is often quite small, it remains in general suboptimal. The algorithm presented in [22,23] allows computing an optimal expansion in exponentials, with complex exponents, in the one-dimensional case (minimal number of exponential functions for a given L^2 norm error). On the other hand, the computational effort required to reach such an approximation, which involves solving a con-eigenvalue problem, finding the roots of a high-degree polynomial, and inverting a Vandermonde matrix, is non-negligible.

Efforts have been made to generalize the latter method to multiple dimensions [24,25], but the cost of current algorithms increases drastically with dimensions. Indeed, the algorithm introduced in [24] for the computation of the nodes associated with two-dimensional approximations is significantly more expensive than in the one-dimensional case for it involves computing two con-eigenvectors for two different Hankel matrices of size $(m_1 m_2) \times (m_1 m_2)$, m_1 and m_2 being the number of samples in each dimensions. In addition, even though Hankel matrices can be applied in low complexity, finding con-eigenvalues and con-eigenvectors remains an expensive procedure that does not scale linearly with dimension.

An application to certain multi-dimensional problems is also presented in [22,23] for the case of radially symmetric kernels. In that case a 1D exponential expansion is obtained as a function of the radial variable r, therefore extending the 1D method to a multi-dimensional setting.

In short, the situation with the current method is somewhat opposite to the aforementioned methods in the sense that the approximation used is suboptimal but known explicitly and easy to construct. This is particularly desirable in some instances such as the interpolation on 2D or 3D manifolds through radial basis methods where a whole family of kernel is to be considered, e.g., $\left\{ \frac{1}{1+(\sigma\, r)^2} \right\}_\sigma$. Finally, the approach in this paper allows for a precise a priori error analysis, whereas in [22,23] the error analysis is more difficult and one cannot generally guarantee an L^∞ error. In addition, in this paper, we have a precise analytical relation between P_k and the behavior of K in the complex plane.

2.2. *Extension to two and three dimensions*

Extension of this method to two and three dimensions relies on a tensor product construction. For example, in 3D, the Cauchy formula reads:

$$K(u,v,w) = \frac{1}{(2\pi\imath)^3} \iiint_{\Gamma} \frac{K(\boldsymbol{z})}{(z_1-u)(z_2-v)(z_3-w)}\,\mathrm{d}z_1\,\mathrm{d}z_2\,\mathrm{d}z_3$$

where Γ is a three dimensional poly-disk in $\mathbb{C}^3$. Upon creating an elliptical contour in $\mathbb{C}^3$ and noting that the integral is periodic in each variable along each individual contour, it is possible to accurately discretize the above integral through a trapezoid in each variable (upon considering its 3D Fourier series). Each term in the resulting sum will take the form,

$$K(z_1(\theta_n^1), z_2(\theta_m^2), z_3(\theta_p^3))\,\frac{z_1'(\theta_n^1)}{z_1(\theta_n^1)-u}\,\frac{z_2'(\theta_m^2)}{z_2(\theta_m^2)-v}\,\frac{z_3'(\theta_p^3)}{z_3(\theta_p^3)-w}$$

where n, m, p are the indices of the nodes associated with the trapezoidal rules just described. Then, it suffices to note that each term of the form $\frac{z'(\theta)}{z(\theta)-\cdot}$ can be *independently* approximated through the method described above. By substituting the approximation and exchanging the summations, one obtains an approximation for $K(\boldsymbol{z}) = K(\boldsymbol{x}-\boldsymbol{y})$ of the form:

$$K(\boldsymbol{x}-\boldsymbol{y}) \approx \sum_{\mathbf{k}}\sum_{\mathbf{p}} e^{\boldsymbol{s}_{\mathbf{k},\mathbf{p}}\cdot\lambda_{\mathbf{k}}(\boldsymbol{x}-\boldsymbol{y})} \left(\sum_{\mathbf{n}} \frac{\lambda_{\mathbf{k}}}{(N\imath)^3} K(\boldsymbol{z}_{\mathbf{n}})\;\partial\cdot\mathbf{z}_{\mathbf{n}}\; e^{-\boldsymbol{s}_{\mathbf{k},\mathbf{p}}\cdot\lambda_{\mathbf{k}}\boldsymbol{z}_{\mathbf{n}}} \right)$$

where $\mathbf{k}$, $\mathbf{p}$, and $\mathbf{n}$ are multi-indices, analogous to Eq. (17). The 1D construction can thus be extended to the 2D and 3D cases.

Because the cost of the method ultimately depends *solely* on the number of terms used in approximating $\frac{z'(\theta)}{z(\theta)-\cdot}$, and since this quantity is *independent* of the dimension, the computational cost in dimension d scales like $\mathcal{O}((\log(1/\varepsilon))^d)$.

The extension to non-translation invariant kernels requires recomputing the transfer function $\mathcal{T}_{k,p}$ for each box pair in the interaction list of the FMM. Otherwise the scheme is unchanged.

In the rest of the paper, we will focus on the analysis of the one dimensional case with translation invariant kernels.

2.3. *Cauchy integral formula as generalized Laplace and Fourier transform.*

It is interesting to note that the expression introduced in Eq. (3), (8) to represent the kernel in terms of an integral transform involving complex exponentials can be seen as a generalization of the Laplace and Fourier transforms; these can be recovered whenever the kernel satisfies appropriate conditions at infinity (see [26,27] for details).

Indeed, assume the kernel to be such that we can use one of the rectangular contours shown in Fig. 3. In the Fourier case (top), assume further that the contributions from the vertical paths go to zero as we let these paths go to infinity. This is the case for example when the kernel goes to zero uniformly along these vertical segments. Then, the integrals over z and s become

$$K(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{\imath s x} \left(\int_{-\infty}^{\infty} e^{-\imath s z} K(z) \, \mathrm{d}z \right) \mathrm{d}s \tag{22}$$

which we recognize as the Fourier transform and its inverse. Similarly, in the Laplace case (bottom) if the contributions from the horizontal paths and right vertical path go to zero, then we are left with

$$K(x) = \frac{1}{2\pi\imath} \int_{0}^{\infty} e^{-s x} \left(\int_{\Gamma_3^{\mathrm{Re}} - \imath\infty}^{\Gamma_3^{\mathrm{Re}} + \imath\infty} e^{s z} K(z) \, \mathrm{d}z \right) \mathrm{d}s \tag{23}$$

This is the Laplace transform and its inverse, the Bromwich integral, but written "backward," that is the inner integral is the Bromwich integral and the outer one is the Laplace transform where x is the variable in the Laplace transform space and s plays the role of the usual t variable in a textbook Laplace transform. As per our analysis Γ_3^{Re} (the location of the intersection of the vertical green line with the x axis in Fig. 3) should be to the right of all poles of K, which is also the required assumption in the Bromwich integral.

Regarding the Fourier case, we note that a sufficient criterion for the aforementioned condition to be satisfied is that $K(t \pm a\imath) \in L^1(\mathbb{R})$ in t, for some $a > 0$. Since most kernel of interest grow at a rate that is slower than exponential along the real axis, it is seen that the function,

$$\tilde{K}(z) = e^{-\varepsilon z^2} K(z)$$

where $0 < \varepsilon \ll 1$ satisfies this condition as long as $K(z)$ is analytic in the vicinity of the real axis. If $K(z)$ is only analytic around some interval, a polynomial approximation followed by a multiplication by $e^{-\varepsilon z^2}$ would produce a similar result. Furthermore, if ε is small enough we have $\tilde{K}(z) \approx K(z)$ in the vicinity of the origin, which is also the region where we are interested in approximating $K(z)$. In this case, using $\tilde{K}(z)$ and discretizing the s-integral in Eq.(22) with a trapezoidal rule would lead to an algorithm very similar to the FFT, whereas a nonuniform quadrature would lead to an algorithm similar to the USFFT [28,29]. This shows that, since in our method we have the freedom to vary the contour in the complex plane, we in general can produce low-rank approximations that contain fewer terms than an expansion in plane waves.

Finally, we note in passing that if the inverse Laplace transform of our kernel is known, an FMM can be derived from (23) and that such a formulation would be simpler than the one presented in this manuscript. In particular the integral over z is then known (it is the inverse transform) and we only need to discretize a single integral over s instead of along $\Gamma_1, \ldots, \Gamma_4$.

3. Choice of contour

The choice of contour must satisfy constraints (3) and (11) and influences the integral domain truncations, L_k, and the discretization size, P_k. The choice also influences how the tree is built and used in Section 4. In this section, we review possible choices, as well as guidelines and simplifications made for this introductory paper.

First, in order for the Cauchy integral formula (3) to apply, the contour must avoid the poles of the kernel and contain $x - y \in [-R, R] \subset \mathbb{R}$. For example, the poles of $1/\sqrt{z^2+1}$ are located at $z = \pm\imath$. Consider points that are contained in the interval $[-D, D]$. When

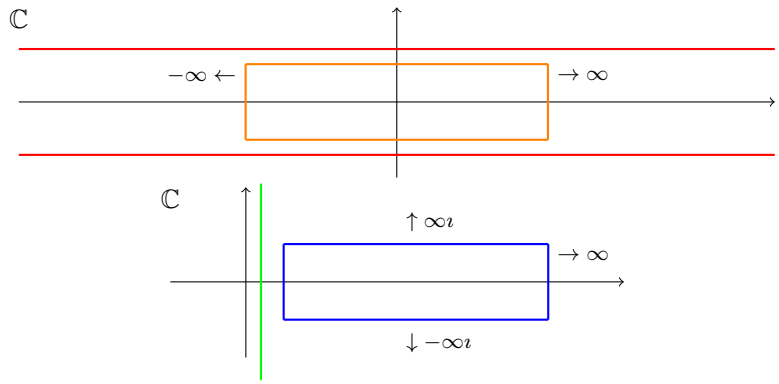

Fig. 3. Top: letting the vertical paths of a rectangular contour go to infinity produces the Fourier transform. Bottom: letting the horizontal paths and right vertical paths of a rectangular contour go to infinity produces the Laplace transform

D is small, it is possible to find an ellipse that surrounds $x - y$, avoids the poles $\pm\imath$, and with an aspect ratio reasonably close to 1. In this case, the tree structure of the multipole method is not required. Using well-separated boxes is no longer important and the matrix is simply low-rank for all entries. This is a degenerate case. When D is large, although it is still theoretically possible to use a single contour, but the efficiency of the method degrades. In that case, it is preferable to use two contours for $x > y$ and $x < y$ as shown in Figure 4.

The case with two contours corresponds to the *well-separated condition* or *multipole acceptance criteria* (MAC) of tree codes and the FMM. The well-separated condition can be enforced with the single-buffer box transfer condition common to FMM. See Section 4.1. This condition ensures that with boxes of side length D,

$$0 \leq |r_x|, |r_y| \leq D/2 \qquad |r_0| \in \{2D, 3D\} \tag{24}$$

See Figure 2. Because this choice is common to fast multipole methods and can be applied to most kernels of interest, we adopt the double contour and single-buffer box transfer formulation unless stated otherwise.

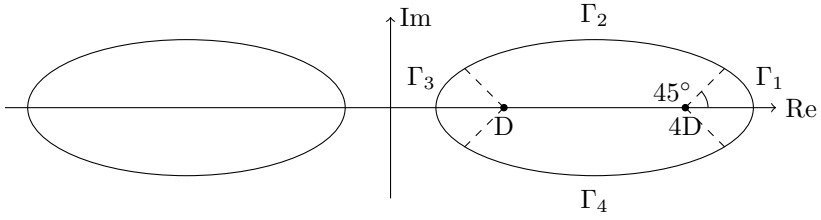

Fig. 4. Schematic for choice of paths. The extremities of the path correspond to the intersection of a 45° line emanating from a focus of the ellipse and the ellipse itself.

Next, to satisfy Eq. (11) given (24), each path of the contour must be sufficiently far away from the interval $[r_0 - D, r_0 + D]$. This allows Eq. (10) to be satisfied and provides an appropriate ℓ_k. In particular, for Γ_1 in Figure 1 which has $\lambda_1 = 1$, Eq. (11) and Eq. (12) read

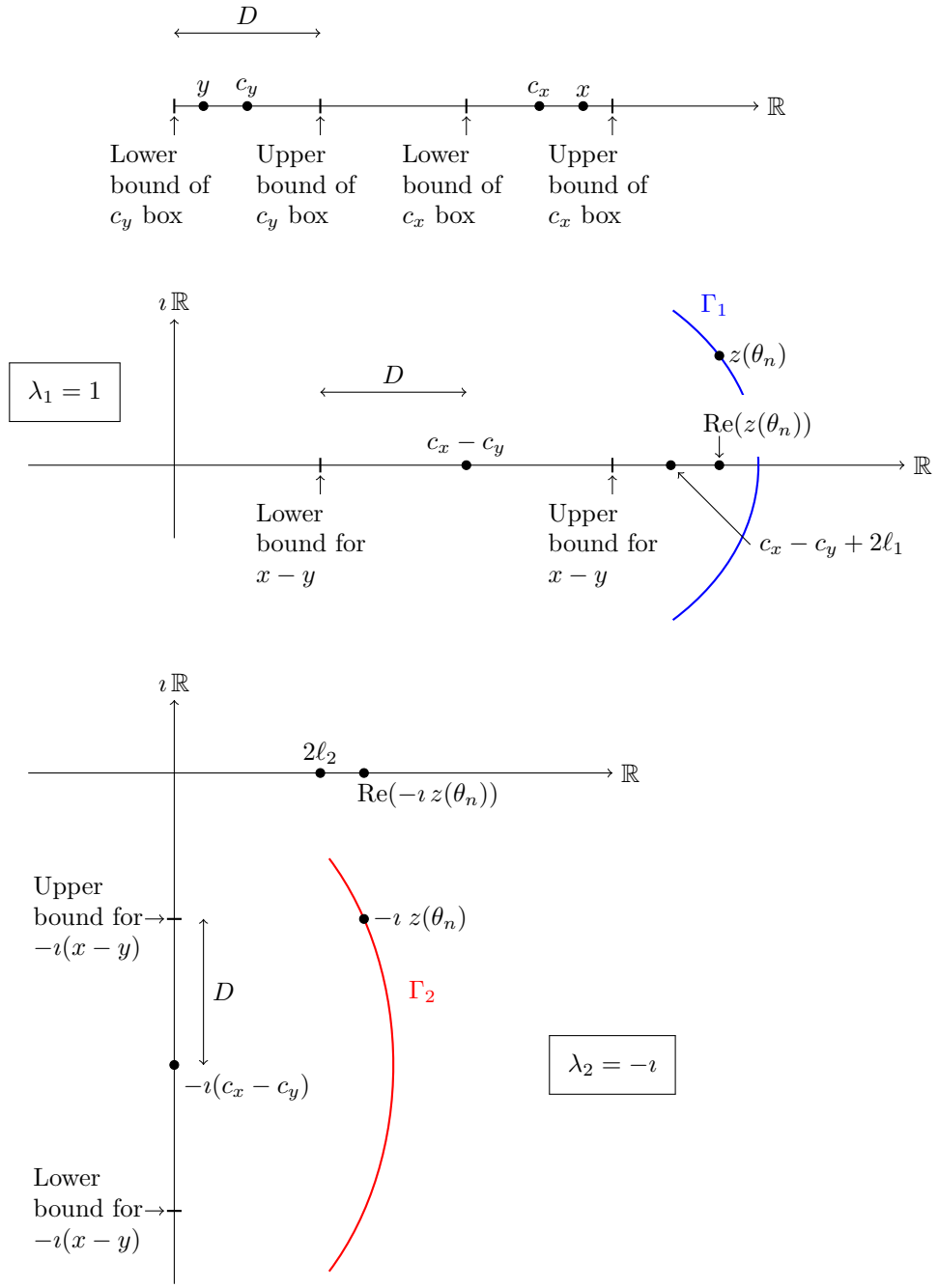

Fig. 5. Schematic representation of some of the variables used in the text. This figure illustrates the geometrical meaning of the variables, the paths Γ_k, and the parameters ℓ_k.

$$\min_{z_n \in \Gamma_1} \mathrm{Re}(z_n) > 4D \qquad\qquad \ell_1 = \frac{1}{2} \min_{z_n \in \Gamma_1} \mathrm{Re}(z_n) - \frac{3D}{2}$$

and for Γ_2 which has $\lambda_2 = -\imath$, these read

$$\min_{z_n \in \Gamma_2} \mathrm{Im}(z_n) > 0 \qquad\qquad \ell_2 = \frac{1}{2} \min_{z_n \in \Gamma_2} \mathrm{Im}(z_n)$$

In Figure 4, the contour goes around D and $4D$. This allows us to use a single contour for both $c_x - c_y = 2D$ and $3D$. This simplified the implementation a bit. However, the optimal choice is to consider two contours, one for $c_x - c_y = 2D$ and $c_x - c_y = 3D$. A unique choice for ℓ_k and the discretization of s needs to be obtained. Each contour will give rise to two different M2L operators. This in general leads to a smaller quadrature for a given tolerance.

Finally, we are interested in the contour(s) that results in the approximation with the lowest rank. In other words, $P = \sum_k P_k$ should be as small as possible while still satisfying appropriate error bounds. We limit our attention to elliptic contours for simplicity,

$$z(\theta) = c + a\cos(\theta) + \imath b \sin(\theta) \tag{25}$$

where $c = 2.5D$, and $a, b > 0$. As in Section 2, we partition each contour into four continuous, disjoint paths $\{\Gamma_k\}_{k=1}^4$ which have been chosen as in Fig. 4. The extremities of the paths correspond to the intersection of a 45° line emanating from the boundary of the interval $[D, 4D]$ and the ellipse itself. Each path has a corresponding integral that must be discretized, Eq. (13).

A full optimization of the rank functional could be carried out, but Section 6.1 implies that the rank remains more or less constant in a large portion of the parameter space. The following rules appear to be sufficient to get an adequate contour:

- The contour Γ should remain as far as possible from the interval containing $(x - y)$ (choose a large a and b).
- Poles or branches in the kernel must be avoided (in order for the Cauchy formula to apply). Contours can, if necessary, be chosen close to poles.
- The vertical axis b should remain relatively close to the real axis for an oscillatory kernel with large box size. This avoids the exponential growth of the kernel when moving away from the real axis.

We previously noted that it is possible to deform the contour either along the x axis to obtain a Fourier-like transform, or stretch the contour along the imaginary axis (Bromwich integral). These contours are in general more efficient since they allow replacing the four arcs Γ_1, Γ_2, Γ_3, and Γ_4 by only two lines (Fourier case) or a single line (Bromwich integral). However we did not pursue these options here. Note that such deformations of the contour to extend it along x or y to infinity require appropriate decay of the kernel.

4. Multi-level algorithm

In this section, we review the full algorithm. This version does not make use of interpolation/anterpolation between different levels in the M2M and L2L stages as is the case with standard multipole algorithms [4]. Rather, each of the $\log(N)$ levels in the tree requires $\mathcal{O}(N)$ steps resulting in $\mathcal{O}(N \log N)$ complexity. The formulation is similar to tree-methods. See for example [30,31].

It is possible to extend the Cauchy formulation and include interpolation and anterpolation steps between levels. This can be done using Fourier analysis to construct Fourier interpolation and filtering steps. Although we have shown independently that such a method can be created, it is complicated. A much simpler extension can use the Chebyshev interpolation technique of [16]. In this way, we could follow the same scheme [16] but replace the SVD and M2L operators by an M2L constructed with the Cauchy formula described in this paper. This would lead to an $\mathcal{O}(N)$ method.

4.1. *The tree*

The 1D multilevel scheme uses a binary tree structure defined over the computational domain, here assumed to be $[0, 1]$. The tree consists of $\mathcal{O}(\log N)$ levels so that each leaf contains $\mathcal{O}(1)$ computational points. This is illustrated in Fig. 6.

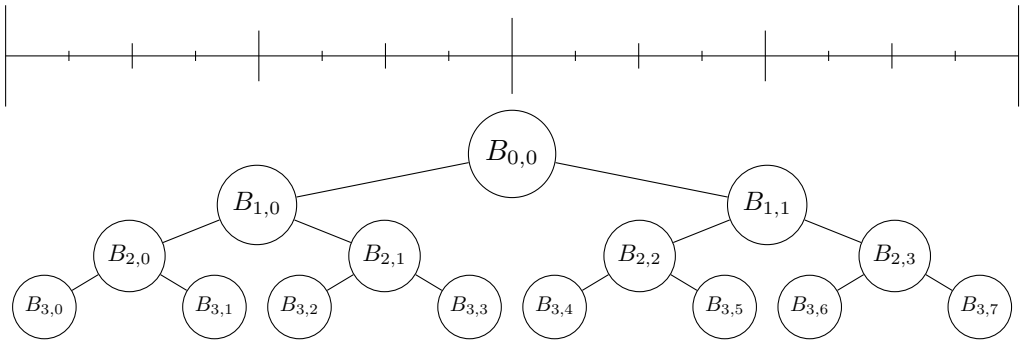

Fig. 6. Binary tree structure. $B_{l,m} = [m2^{-l}, (m+1)2^{-l}]$ represents the mth subinterval at level l.

We then make the following definitions:

(i) Given a box $B_{l,m}$, we define $\mathcal{N}(B_{l,m})$ to be the *neighbor* set of boxes at level l which are adjacent to $B_{l,m}$ or $B_{l,m}$ itself. For instance, in Fig. 6, $\mathcal{N}(B_{3,3}) = \{B_{3,2}, B_{3,3}, B_{3,4}\}$.

(ii) Given a box $B_{l,m}$, we define $\mathcal{I}(B_{l,m})$ to be the *interaction list* of boxes at level l which are not the neighbors of $B_{l,m}$ but whose parent is a neighbor of $B_{l,m}$'s parent. For instance, in Fig. 6, $\mathcal{I}(B_{3,3}) = \{B_{3,0}, B_{3,1}, B_{3,5}\}$. Two boxes that are in each other's interaction list will be called *well-separated*.

At each level, a low-rank approximation of the form (17) is used to compute the interactions between well-separated boxes, i.e., the *far-field interactions*, in $\mathcal{O}(N)$ steps. More precisely, consider any two well-separated boxes B_x with center c_x and B_y with center c_y at level l such that $\{x_i\}_{i=1}^{N_x} \in B_x$ and $\{y_j\}_{j=1}^{N_y} \in B_y$. Using Eq. (17), we want to compute

$$\sum_{y_j \in B_y} \sigma_j K(x_i - y_j) \approx \sum_k \sum_{p=0}^{P_k} e^{-s_{k,p}^2(\ell_k - \lambda_k(x_i - c_x))}\, \mathcal{T}_{k,p} \left(\sum_{y_j \in B_y} \sigma_j\, e^{-s_{k,p}^2(\ell_k + \lambda_k(c_y - y_j))} \right)$$

for each $x_i \in B_x$. Performing the computation in the order prescribed on the right-hand side of the above expression leads to a number of operations linear in N_x and N_y.

4.2. *Method summary*

To compute *all* pairs of interactions, perform the following steps:

(i) **Gather**.
For each box $B_{l,m}$ in the tree, compute,

$$\phi_{l,m}[k,p] = \sum_{y_j \in B_{l,m}} \sigma_j e^{-s_{k,p}^2(\ell_k + \lambda_k(c_y - y_j))}$$

for $k = 1, \ldots, 4$ and $p = 0, \ldots, P_k$.

(ii) **Transfer.**
For each box $B_{l,m}$ in the tree, transfer and accumulate the expansion from its far neighbors by computing,

$$\psi_{l,m}[k,p] = \sum_{B_{l,n} \in \mathcal{I}(B_{l,m})} \mathcal{T}_{k,p}\, \phi_{l,m}[k,p]$$

for $k = 1, \ldots, 4$ and $p = 0, \ldots, P_k$.

(iii) **Scatter.**
For each $x_i \in B_{leaf,m}$, the total field is given by the sum of the close-interactions and the evaluation of the far-field interactions at each level of the tree,

$$\begin{aligned}\sum_{y_j} \sigma_j\, K(x_i - y_j) \approx & \sum_{y_j \in \mathcal{N}(B_{leaf,m})} \sigma_j\, K(x_i - y_j) \\ & + \sum_{\substack{B_{l,m} \\ x_i \in B_{l,m}}} \sum_k \sum_{p=0}^{P_k} e^{s_{k,p}^2(\ell_k - \lambda_k(x_i - c_x))} \psi_{l,m}[k,p]\end{aligned}$$

where $\mathcal{N}(x_i)$ is the neighbor set of the leaf box x_i is contained in.

Let $P = \sum_k P_k$. Then, for each level l, the gather step takes $\mathcal{O}(PN)$ operations (where P here would be the maximum over all levels), the transfer step takes $\mathcal{O}(P2^l) = \mathcal{O}(PN)$ operations, and the scatter step takes $\mathcal{O}(PN)$. This results in a total of $\mathcal{O}(PN)$ operations at every level. Since there are $\mathcal{O}(\log N)$ levels, the algorithmic complexity for computing the far-field interactions is $\mathcal{O}(N \log N)$.

By construction every leaf box contains $\mathcal{O}(1)$ charges and in 1D the set $\mathcal{N}$ has maximum size three. Therefore, computation of the close-term interactions is an $\mathcal{O}(N)$ operation. Thus, the computational time is dominated by the transfer stage and the described tree code requires $\mathcal{O}(N \log N)$ operations.

5. Error analysis

In this section, we provide additional detail leading to the error bounds introduced in Section 2 through Eq. (7), Eq. (14) and Eq. (16). Where possible, the analysis is applied to each path of the contour independently and we drop the k subscripts for clarity.

5.1. *Outer integral along contour* Γ

The first step involves discretizing the complex integral over Γ, that is, finding an appropriate quadrature for the integral

$$\frac{1}{2\pi\imath}\int_0^{2\pi}\frac{K(z(\theta))\,z'(\theta)}{z(\theta)-(x-y)}\,\mathrm{d}\theta$$

Since the integrand is a smooth and periodic function of θ, its Fourier series,

$$\frac{K(z(\theta))\,z'(\theta)}{z(\theta)-(x-y)}=\sum_{n=-\infty}^{\infty}c_n e^{\imath n\theta}$$

has Fourier coefficients c_n that decay *exponentially* fast with n.

The exact integral is

$$I=\frac{1}{2\pi\imath}\int_0^{2\pi}\frac{K(z(\theta))\,z'(\theta)}{z(\theta)-(x-y)}\,\mathrm{d}\theta=\frac{1}{2\pi\imath}\int_0^{2\pi}\sum_{n=-\infty}^{\infty}c_n e^{\imath n\theta}\,\mathrm{d}\theta=\frac{1}{\imath}c_0$$

Considering a discretization through an N-point trapezoidal rule,

$$I_N=\frac{1}{2\pi\imath}\sum_{j=0}^{N-1}\frac{2\pi}{N}\sum_{n=-\infty}^{\infty}c_n e^{\imath n\frac{2\pi j}{N}}=\frac{1}{\imath}\sum_{n=-\infty}^{\infty}c_n\left(\sum_{j=0}^{N-1}\frac{e^{\imath n\frac{2\pi j}{N}}}{N}\right)$$

the term in parenthesis is the Dirichlet kernel, which is 0 unless j is an integer multiple of N, in which case it is 1,

$$=\frac{1}{\imath}\sum_{n=-\infty}^{\infty}c_{nN}$$

This implies that the error in using the N-point trapezoidal rule can be computed as

$$|\varepsilon_N|=|I-I_N|\le\sum_{n\neq 0}|c_{nN}| \tag{26}$$

Since the $|c_n|$ decays exponentially fast in n, so does the error. That is, there exist $\mathcal{A}(K,\Gamma)$, $\mathcal{B}(K,\Gamma)>0$ independent of N such that

$$|\varepsilon_N|\le\mathcal{A}e^{-\mathcal{B}N}$$

5.2. *Inner integral - Domain truncation*

On each path, the error produced by the integral truncation

$$\int_{-\infty}^{\infty}|s|\,e^{-\zeta s^2}\,\mathrm{d}s=\int_{-L}^{L}|s|\,e^{-\zeta s^2}\,\mathrm{d}s+\varepsilon_L$$

when $\mathrm{Re}(\zeta)>0$ is bounded by

$$
\begin{aligned}
|\varepsilon_L| &= 2\left|\int_L^\infty |s|\, e^{-\zeta s^2}\,\mathrm{d}s\right| \\
&\le 2\int_L^\infty s\, e^{-s^2\mathrm{Re}(\zeta)}\,\mathrm{d}s \\
&= \frac{e^{-\mathrm{Re}(\zeta)L^2}}{\mathrm{Re}(\zeta)}
\end{aligned}
$$

where $\zeta = \zeta(z,x,y)$. The total contribution to the error is then bounded by

$$
\left(\sum_{z_n\in\Gamma_k} \frac{|K(z_n)|\,|z_n'|}{2\pi N}\right)\left|\varepsilon_L^k\right| \le \max_{z_n\in\Gamma_k,x,y}\left\{|K(z_n)|\,|z_n'|\,\frac{e^{-\mathrm{Re}(\zeta(z_n,x,y))L_k^2}}{2\pi\mathrm{Re}(\zeta(z_n,x,y))}\right\} \le \mathcal{C}e^{-\mathcal{D}L_k^2} \tag{27}
$$

which is the desired result.

5.3. *Inner integral - Quadrature for proper integral*

As was argued in Section 2, the trapezoid rule is an efficient quadrature for integrals of the form

$$
J = \int_{-L}^{L} |s|\, e^{-\mu s^2} e^{-s^2\alpha(x)} e^{-s^2\beta(y)}\,\mathrm{d}s \tag{28}
$$

when $\mathrm{Re}(\mu)$, $\mathrm{Re}(\alpha(x))$, $\mathrm{Re}(\beta(y)) > 0$. In this section, we derive explicit bounds for the error of this approximation. First, the error produced from using a low-band transfer function is bounded. Then, we study the effects of the discretization itself.

Define the Fourier series (where we assume that $s \in [-L, L]$):

$$
\begin{aligned}
f(s) = e^{-s^2\alpha(x)}e^{-s^2\beta(y)} &= \sum_{n=-\infty}^{\infty} \hat{f}_n e^{\imath\pi ns/L} \\
T(s) = |s|\, e^{-\mu s^2} &= \sum_{m=-\infty}^{\infty} \hat{T}_m e^{\imath\pi ms/L} \\
T^{lb}(s) &= \sum_{m=-P}^{P} \hat{T}_m e^{\imath\pi ms/L}
\end{aligned}
$$

We note that the functions are not smooth at $\pm L$ and therefore the Fourier spectra decay slowly. However the jump in derivative is of size ε and we will show that the impact on accuracy is small.

Substituting the Fourier series into the original integral yields

$$
J = \int_{-L}^{L} \left(\sum_{n=-\infty}^{\infty} \hat{f}_n e^{\imath\pi ns/L}\right)\left(T^{lb}(s) + \sum_{|m|>P} \hat{T}_m e^{\imath\pi ms/L}\right)\,\mathrm{d}s
$$

The integral to be discretized for the scheme is,

$$
J^{lb} = \int_{-L}^{L} e^{-s^2\alpha(x)} e^{-s^2\beta(y)}\, T^{lb}(s)\,\mathrm{d}s \tag{29}
$$

The error due using a low-band approximation of T is bounded by,

$$\left|J - J^{lb}\right| \leq \left| \sum_{n=-\infty}^{\infty} \sum_{|m|>P} \hat{T}_m \hat{f}_n \int_{-L_k}^{L_k} e^{\imath\pi(m+n)s/L_k} \, \mathrm{d}s \right|$$

$$\leq 2L \sum_{|n|>P} \left|\hat{T}_{-n}\right| \left|\hat{f}_n\right| \tag{30}$$

It remains to study the error incurred by the discretization of Eq. (29) through a trapezoid rule. First, we note that

$$J^{lb} = \int_{-L}^{L} \left(\sum_{n=-\infty}^{\infty} \hat{f}_n \, e^{\imath\pi n s/L} \right) \left(\sum_{m=-P}^{P} \hat{T}_m \, e^{\imath\pi m s/L} \right) \mathrm{d}s = 2L \sum_{m=-P}^{P} \hat{T}_{-m} \hat{f}_m \tag{31}$$

After discretization,

$$J_P^{lb} = \frac{L}{P} \sum_{p=-P}^{P} \beta_p \, e^{-s_p^2 \alpha(x)} e^{-s_p^2 \beta(y)} \, T^{lb}(s_p)$$

$$= 2L \sum_{q=-\infty}^{\infty} \sum_{m=-P}^{P} \hat{T}_{-m} \hat{f}_{m+q(2P+1)}$$

where $\beta_p = 1/2$ if $p = \pm P$ and 1 otherwise, and where $s_p = p\frac{L}{P}$. The discretization error is then

$$\left|J^{lb} - J_P^{lb}\right| \leq 2L \sum_{q\neq 0} \sum_{m=-P}^{P} \left|\hat{T}_{-m}\right| \left|\hat{f}_{m+q(2P+1)}\right| \tag{32}$$

From Eq. (30) and Eq. (32), the total error is bounded by,

$$|\varepsilon_P| \leq 2L \sum_{|n|>P} \left|\hat{T}_{-n}\right| \left|\hat{f}_n\right| + 2L \sum_{q\neq 0} \sum_{m=-P}^{P} \left|\hat{T}_{-m}\right| \left|\hat{f}_{m+q(2P_k+1)}\right| \tag{33}$$

$$\leq 2\,(2L)^2 \max_{s\in[-L,L]} |T^{lb}(s)| \sum_{|n|>P} \left|\hat{f}_n\right| \tag{34}$$

since $T^{lb}(\cdot) \in L^2([-L, L])$ and,

$$\left|\hat{T}_m\right| \leq \left\|T^{lb}\right\|_2 \leq 2L \max_{s\in[-L,L]} |T^{lb}(s)|$$

by Plancherel's theorem and Hölder's inequality.

These bounds are useful and the use of the low-pass transfer function is feasible because the Fourier coefficients $\hat{f}_n$ and $\hat{T}_m$ can be computed analytically. The explicit form for the coefficients $\hat{f}_n$ is given by,

$$\hat{f}_n = \frac{\sqrt{\pi}}{2L\sqrt{\gamma}} \Big[e^{-\frac{\pi^2 n^2}{4\gamma L^2}} - \frac{(-1)^n}{2} e^{-\gamma L^2} \Big(W\left(\frac{2\imath L^2\gamma - \pi n}{2L\sqrt{\gamma}}\right) + W\left(\frac{2\imath L^2\gamma + \pi n}{2L\sqrt{\gamma}}\right) \Big) \Big] \tag{35}$$

where $\gamma = \alpha + \beta$ and $W(z) = e^{-z^2}\mathrm{erfc}(-\imath z)$ is the Faddeeva function. Although this expression looks complicated, we recognize two dominant terms:

(i) $e^{-\frac{\pi^2 n^2}{4\gamma L^2}}$: this term corresponds to the Fourier transform of a Gaussian (which is

itself a Gaussian function) evaluated at n.

(ii) $e^{-\gamma L^2}$: this term corresponds to the truncature of the interval at $\pm L$, which leads to a discontinuity in the derivative and therefore a slow decay of the spectrum (in $1/n^2$, see below). However since this slowly decaying term is multiplied by $e^{-\gamma L^2}$, the error is controlled.

The asymptotic form of the Faddeeva function as $|z| \to \infty$ is given by ([32], 7.1.23),

$$W(z) = \frac{1}{-\imath\sqrt{\pi}z}\left(1 + \frac{1}{2z^2} + \mathcal{O}\left(\frac{1}{z^4}\right)\right)$$

for $-\frac{\pi}{4} \leq \arg(z) \leq \frac{5\pi}{4}$. Therefore, for n large enough we have,

$$W\left(\frac{2\imath L^2\gamma \pm \pi n}{2L\sqrt{\gamma}}\right) = \frac{2\imath L\sqrt{\gamma}}{\sqrt{\pi}(2\imath L^2\gamma \pm \pi n)} + \mathcal{O}\left(\frac{1}{n^3}\right)$$

Substituting into the expression for $\hat{f}_n$ gives,

$$\hat{f}_n = \frac{\sqrt{\pi}}{2L\sqrt{\gamma}} e^{-\frac{\pi^2 n^2}{4\gamma L^2}} - \frac{(-1)^n}{2} e^{-\gamma L^2} \frac{4L^2\gamma}{4L^4\gamma^2 - \pi^2 n^2} + \mathcal{O}\left(\frac{1}{n^3}\right)$$

First, note that $\hat{f}_n$ decays asymptotically like $\mathcal{O}(1/n^2)$, as expected from the discontinuity in the derivative at $\pm L$. In addition, the first term dominates as long as,

$$e^{-\frac{\pi^2 n^2}{4\gamma L^2}} \gtrsim e^{-\gamma L^2}$$

Given $\varepsilon > 0$ and choosing L such that $e^{-\gamma L^2} \leq \varepsilon$, the coefficients $\left|\hat{f}_n\right|$ will exhibit an *exponential* decay up to ε-terms, i.e.,

$$\left|\hat{f}_n\right| \leq \left|\frac{\sqrt{\pi}}{2L\sqrt{\gamma}}\right| \left(\left|\frac{e^{-\pi^2 P^2/(4\gamma L^2)}}{n^2}\right| + \frac{\varepsilon}{2}\left|\frac{4L^2\gamma}{4L^4\gamma^2 - \pi^2 n^2}\right|\right)$$

for all $|n| \geq P$. Substituting these estimates in Eq. (34) yields

$$\begin{aligned} |\varepsilon_P| \leq 4L\sqrt{\frac{\pi}{|\gamma|}} \max_{s\in[-L,L]} \left|T^{lb}(s)\right| \Big(&\left|e^{-\pi^2 P^2/(4\gamma L^2)}\right| \sum_{|n|>P} \frac{1}{n^2} \\ &+ \frac{\varepsilon}{2} 4L^2 |\gamma| \sum_{|n|>P} \frac{1}{|4L^4\gamma^2 - \pi^2 n^2|}\Big) \end{aligned} \tag{36}$$

Therefore, there exists $\mathcal{E}, \mathcal{F} > 0$ such that,

$$|\varepsilon_P| \leq \mathcal{E}\left(e^{-\mathcal{F} P^2/L^2} + \varepsilon\right)$$

This is the desired result.

5.4. *Error bound for full approximation*

Eq. (18) provides an explicit bound for the error incurred by the complete approximation. From this expression it follows that given an error threshold $\varepsilon > 0$, it is sufficient to choose N, $\{L_k\}_{k=1}^4$ and $\{P_k\}_{k=1}^4$ such that,

$$
\begin{aligned}
|\varepsilon_N| &\leq \frac{\varepsilon}{9} \\
\left|\varepsilon_L^k\right| &\leq \frac{2\pi N}{\max_{z_n \in \Gamma_k} |K(z_n)|\,|z_n'|}\,\frac{\varepsilon}{9} \\
\left|\varepsilon_P^k\right| &\leq \frac{2\pi N}{\max_{z_n \in \Gamma_k} |K(z_n)|\,|z_n'|}\,\frac{\varepsilon}{9}
\end{aligned}
$$

to obtain an approximation with error bounded by ε in $\|\cdot\|_\infty$-norm. Additionally, Eqns. (26), (27) and (36) can be used to choose the parameters in such a way that the above conditions are satisfied. This can be done by following the steps in Algorithm 1. Note that given the nature of the expressions, the order in which the parameters are computed is important and should follow that of Algorithm 1.

Finally, note that step 3 and 4 in Algorithm 1 can be achieved simply by taking suffi-

1 Choose N s.t. $|\varepsilon_N| < \varepsilon/9$, (Eq. (26))
2 **for** $k = 1, \ldots, 4$ **do**
3 Choose L_k s.t. $\left|\varepsilon_L^k\right| < \frac{2\pi N}{\max_{z_n \in \Gamma_k} |K(z_n)||z_n'|}\,\frac{\varepsilon}{9}$, [Eq. (27)]
4 Choose P_k s.t. $\left|\varepsilon_P^k\right| < \frac{2\pi N}{\max_{z_n \in \Gamma_k} |K(z_n)||z_n'|}\,\frac{\varepsilon}{9}$, [Eq. (36)]
5 Construct approximation (Eq. (17))

Algorithm 1: Construction of the CFMM approximation

ciently large L_k and P_k in Eq. (27) and (36) respectively.

6. Results

In this section, we present numerical results demonstrating the versatility and the performance of the method. Three kernels are studied with independent and representative properties:

- Inverse Multiquadric, $\frac{1}{\sqrt{z^2+1}}$. Monotonically decaying kernel with no pole at the origin.
- Multiquadric, $\sqrt{z^2+1}$. Monotonically increasing kernel with no pole at the origin.
- 3D Helmholtz, $\frac{e^{\imath z}}{z}$. Physical, oscillatory kernel with a pole at the origin.

Note that although we chose the 3D Helmholtz kernel, we study the 1D version of the algorithm. This corresponds to computation points *lying on a line*. The results presented here should be considered a study of the behavior of the approximation for oscillatory kernels, rather than an attempt at a comparison with existing fast multipole methods and fast algorithms specific to this problem (e.g., [33–35]). Indeed, as will be shown below, it is generally not a good idea to use the present scheme to represent such kernels in the high-frequency regime. The method can, however, be used in conjunction with the directional FMM scheme in [36], where approximations for kernels of the form $g(x)e^{\imath k|x|}$ (g smooth) are proposed.

We will call the *effective* number of quadrature points the total number of quadrature points (P_k) needed when symmetry of the paths is taken into account. For instance, the two nonoscillatory kernels above exhibit conjugate symmetry about the real axis in the complex plane. Therefore, computations corresponding to path Γ_2 (Fig. 1) can be used to treat path Γ_4 without incurring additional computational work.

The singularities for each kernel lie on the imaginary axis (at $\pm\imath$ for the nonoscillatory kernels and at 0 for the Helmholtz kernel), and the contour Γ must avoid such singularities. As discussed in Section 3, we use two elliptic contours (two low-rank approximations) on either side of the imaginary axis. This is shown schematically in Figure 4. This does not change the computational cost of the method and results simply in different transfer functions $T^{lb}_{k,n}(s)$ depending on the sign of $x - y$.

We will discuss through numerical examples the choice of contour parameters that reinforce the guidelines to choose a near-optimal contour in Section 3. Then, we demonstrate the convergence properties of the low-rank approximation (Eq. (17)) numerically and compare the results with theoretical estimates (Section 5). We also compare the results with that of [16]. In the subsequent section, we study the behavior of the full multilevel scheme. Finally, we demonstrate that the multilevel algorithm scales like $\mathcal{O}(N \log N)$ when the number of levels is chosen appropriately.

6.1. *Contour parameters*

For each path, P_k is chosen according to Eq. (36) (see also Algorithm 1). To show how $P = \sum_k P_k$ depends on the choice of contour parameters a and b in (25), consider $x > y$ (the right contour of Fig. 4) and fix a box size D.

For each of the three kernels introduced above and for $D \in \{0.025, 0.25, 2.5, 25\}$, the typical behavior of P as a function of a and b is shown in Fig. 7 and 8. Generally, the rank is larger when a or b is close to its lower bound. This represents the case when the ellipse encroaches closely on the interval $[D, 4D]$ containing $x - y$. P then quickly decreases to its minimum as a and b get larger.

Similar results were obtained in every case with the exception of the Helmholtz kernel with $D \in \{2.5, 25\}$ for which an example is shown in Fig. 8. There, we notice an increase of the rank as b gets larger. This is related to the growth of the kernel (exponential) along the positive imaginary axis and results in a linear growth of the rank with the height (b).

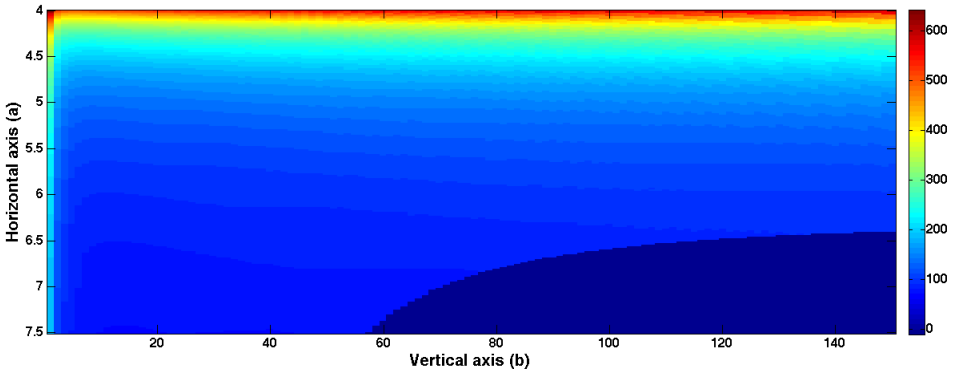

Fig. 7. Typical behavior of P as a function of the contour Γ. This result was obtained using the multiquadric kernel with box size $D = 2.5$, $\varepsilon = 10^{-6}$. The dark blue in the bottom right corner corresponds to inadmissible parameters. This corresponds to elliptic contours that contain a pole or branch of the kernel.

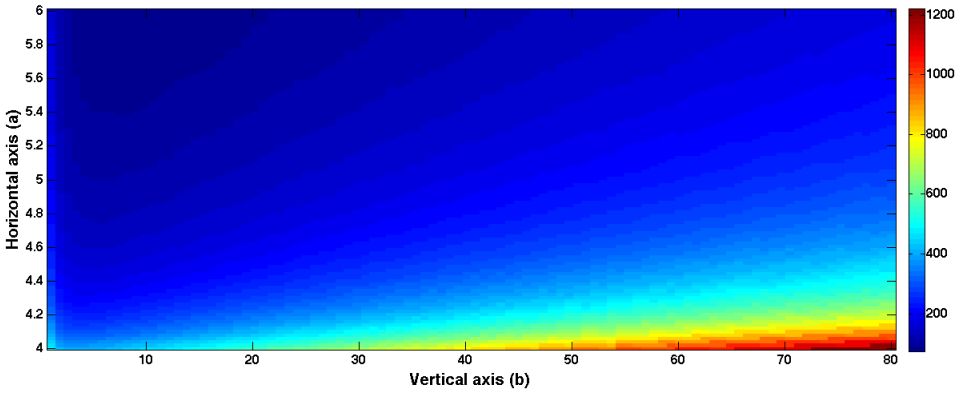

Fig. 8. Typical behavior of P as a function of the contour Γ. This result was obtained using the Helmholtz kernel with box size $D = 2.5$, $\varepsilon = 10^{-6}$.

6.2. *Low-Rank Approximation*

We study numerically the accuracy of the low-rank approximation obtained earlier [Eq. (17)] as a function of the number of quadrature points. Symmetries of the kernel are used to reduce the number of points whenever possible. For example, the multiquadric kernel is conjugate symmetric about the real axis. That is, the terms pertaining to Γ_2 in Fig. 1 are equal to the conjugate of those along Γ_4. This is used to reduce the number of terms in the sum.

In Fig. 9, 10 and 11, the L^∞ error is plotted as a function of the effective number of quadrature points. That is, we fix the box size D, consider all possible combinations of well-separated points x and y belonging to boxes of this size and display the largest absolute error between the approximation and the exact value for various ranks.

In these figures, the error exhibits exponential decay as a function of the rank for every kernel. In addition, for a fixed accuracy, the rank associated with nonoscillatory kernels is more or less independent of the box size. The choice of contour Γ only weakly affects the convergence as long as Γ surrounds the interval of interest and stays away from the poles as described previously.

For the oscillatory kernel, the rank, for small D, is a function primarily of the target error, ε. However for large D, the rank grows approximately linearly with D, at fixed accuracy. This increase is expected. It can be traced back to the fact that for large D, we may have several periods for e^{iz} inside one box of size D. In the complex plane, e^{iz} grows exponentially fast with $\mathrm{Im}(z)$ for $\mathrm{Im}(z) < 0$ and as a result the contour cannot move "away" from the real axis. The error analysis that we derived previously shows that in this case the number of quadrature points must increase approximately linearly with D.

The method of Messner et al. [36] can be used in this case to derive a fast method in which the rank is independent of D. Messner et al. [36] takes out the oscillatory part by using a pre- and post-multiplication step. The resulting kernel is then smooth and does not grow exponentially like e^{iz} in the complex plane. As a result we would recover a convergence similar to the smooth kernels. We will not demonstrate this point numerically in this paper however.

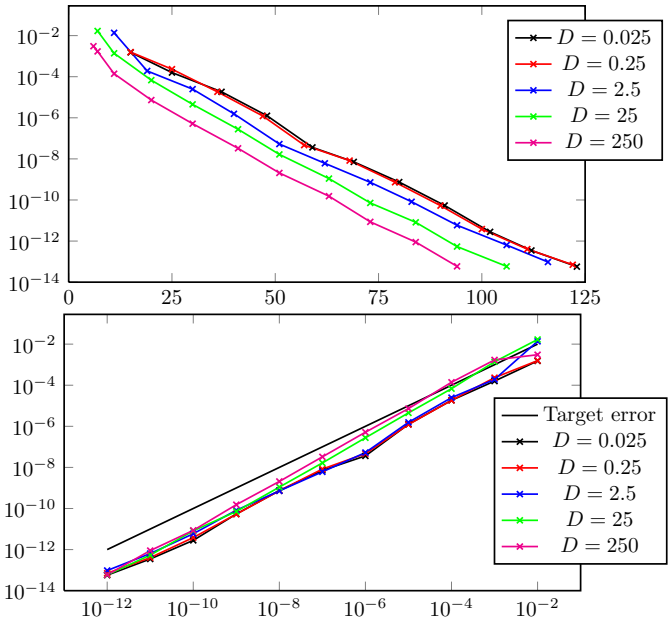

Fig. 9. Inverse Multiquadric $1/\sqrt{1+x^2}$; top: L^∞ error vs rank; bottom: L^∞ error vs target error (Eq. (18)).

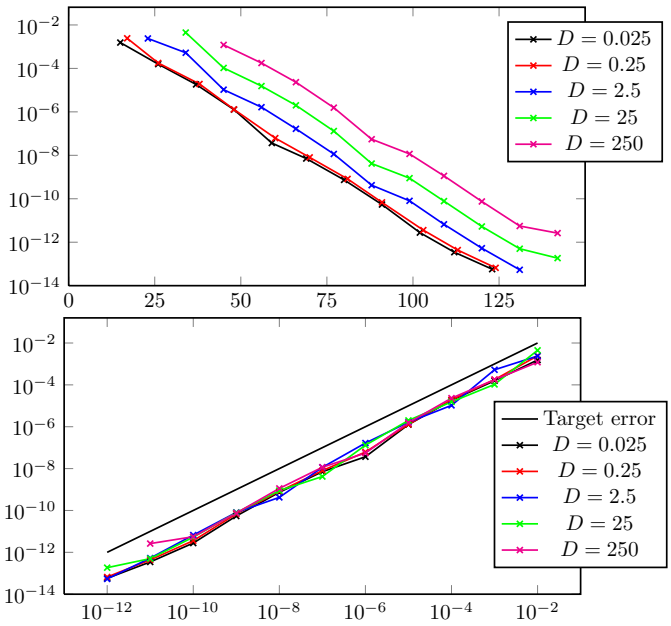

Fig. 10. Multiquadric $\sqrt{1+x^2}$; top: L^∞ error vs rank; bottom: L^∞ error vs target error (Eq. (18)).

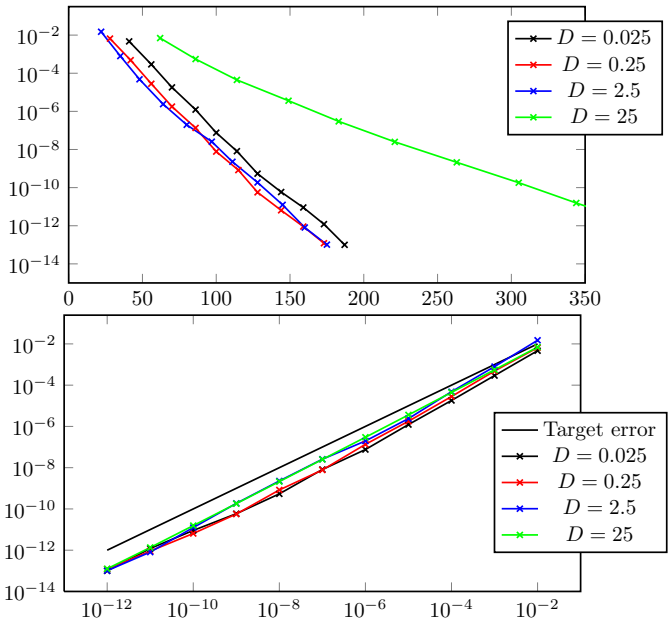

Fig. 11. Helmholtz $e^{\imath z}/z$; top: L^∞ error vs rank; bottom: L^∞ error vs target error (Eq. (18)).

The error estimates provided by Eq. (18) are reasonably sharp. For all cases, the error estimate was within a factor of 100 from the actual error.

For the purpose of comparison, it is also useful to look at Fig. 12, 13 and 14 that provide results concerning the normalized L^∞ error (absolute L^∞ error divided by the maximum of the kernel in $[D, 4D]$) using the scheme presented in Fong and Darve [16] together with ours. The maximum of the kernel is attained at $x - y = D$ for the case of decaying kernels (inverse multi quadric, Helmholtz) and $x - y = 4D$ for growing kernels (multiquadric).

The Fong and Darve scheme is also applicable to general non-oscillatory kernels. The main difference is that it does not possess diagonal translation and transfer operators. For example, in the Cauchy-based FMM (CFMM, this paper) the cost of M2L is $\mathcal{O}(P)$. In contrast, the Chebyshev interpolation based FMM (ChebFMM) [16] (accelerated using a singular value decomposition – SVD) has cost $\mathcal{O}(n^2)$ where n the is rank of the approximation. This is because the M2L operator is dense and therefore a matrix vector product of size $n \times n$ is required. This difference in scaling is the reason why CFMM is faster in some cases. ChebFMM uses an optimal SVD and therefore n is in general smaller than P. However the quadratic scaling makes the method less competitive at large n.

The error estimate in [16] is the following: given $K(z)$ analytic in the ellipse E_r of the complex plane given by the locus of points $\frac{1}{2}(re^{\imath t} + r^{-1}e^{-\imath t})$, $t \in [0, 2\pi]$, for some $r > 1$, and a constant M such that $|K(z)| \leq M$, $\forall\, z \in E_r$, the Chebyshev polynomial approximation $p_n(x)$, $x \in [-1, 1]$, of rank n is such that,

$$|K(z) - p_n(x)| \leq \frac{(r + r^{-1})M}{(r^{n+1} + r^{-(n+1)})(r + r^{-1} - 2)} \leq \mathcal{G}_r e^{-n \ln r} \max_{z \in E_r} |K(z)| \tag{37}$$

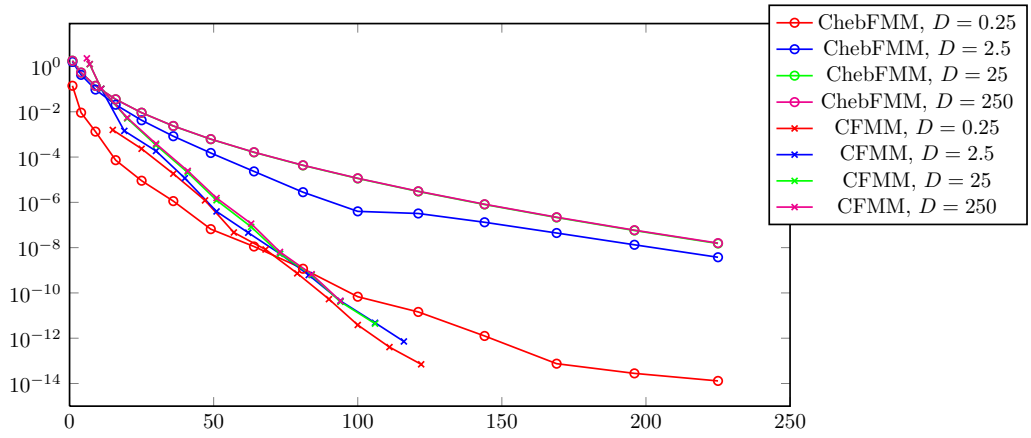

Fig. 12. Inverse Multiquadric $1/\sqrt{1+x^2}$; normalized L^∞ error vs computational cost. This cost is estimated as n^2 for the Chebyshev interpolation FMM [16] (ChebFMM) and P for the Cauchy-based FMM (CFMM). The curves $D = 25$ and $D = 250$ for ChebFMM overlap.

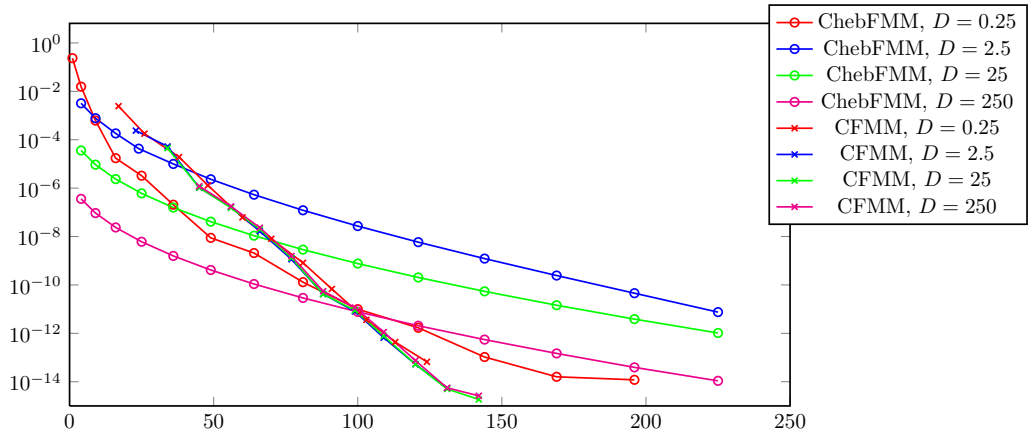

Fig. 13. Multiquadric $\sqrt{1+x^2}$; normalized L^∞ error vs computational cost.

where $\mathcal{G}_r > 0$ is a function of r. We assume that r is bounded away from 1, that is $r > 1+\varepsilon$. Comparing with Eq. (21) we notice that both methods have the same scaling. This means that as $\varepsilon \to 0$:

$$P(\varepsilon) \in \Theta(n(\varepsilon))$$

This explains why we expect our scheme to have a lower computational cost; the cost of applying a dense operator in the case of Fong and Darve is $\mathcal{O}(n^2)$ whereas in our case it is $\mathcal{O}(P)$ thanks to the fact that our operators are diagonal. The effect is even more severe at higher dimension where the cost of a dense operator is now $\mathcal{O}(n^{2d})$ whereas it is $\mathcal{O}(P^d)$ for a diagonal operator, d being the dimension.

Fig. 12 and 13 reveal some inefficiencies in our implementation of the Cauchy formula. In Fig. 12, CFMM is faster than ChebFMM for $D \geq 2.5$. However, for $D = 0.25$ and cost below ≈ 70, we find that CFMM is slower. In this case the segment containing the

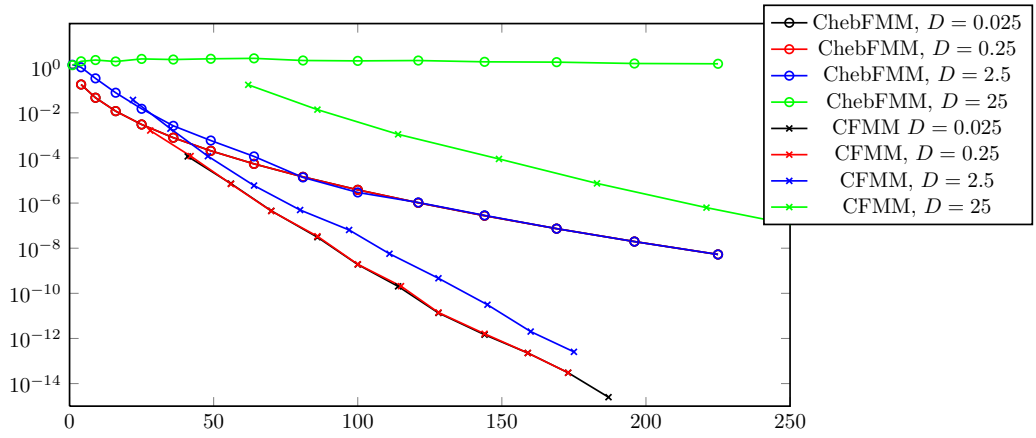

Fig. 14. Helmholtz $e^{\imath x}/x$; normalized L^∞ error vs computational cost. The cases $D = 0.025$ and $D = 0.25$ for ChebFMM overlap.

points $x - y$ is small and close to the origin. The optimal contour should contain the origin (pass to the left of the axis $x = 0$). See Fig. 15. As discussed in Section 3 this is an unusual application of the FMM since the condition of well-separateness is not required. This is because we are considering $1/\sqrt{1+x^2}$ with $x \ll 1$. We did not consider the case of contours that could cross the axis $x = 0$ in our implementation and are therefore using a sub-optimal contour in that case. This is why we observe an inefficiency in CFMM for that case. We observe that ChebFMM has improved performance for $D = 0.25$ compared to the other values of D. This can be interpreted in a similar fashion to the argument above, by observing that in (37) we can choose a relatively large r.

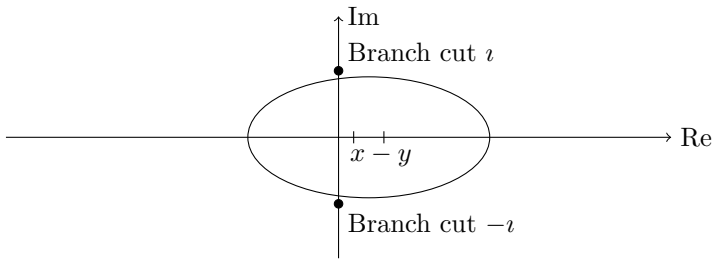

Fig. 15. Optimal contour in the case $D \ll 1$.

In Fig. 13, ChebFMM outperforms CFMM significantly. We kept this result although it is somewhat spurious and results from the fact that we limited our analysis with ChebFMM to 1D. In that case $\sqrt{1+x^2}$ is very close to a linear function for $x \gg 1$ and therefore is very-well approximated by Chebyshev polynomials. More precisely we have the following Taylor expansion in $1/x$ for large $x > 0$:

$$\sqrt{1+x^2} = x\,(1 + \frac{1}{2x^2} - \frac{1}{8x^4} + \cdots) = x + \frac{1}{2x} - \frac{1}{8x^3} + \ldots$$

The function $\sqrt{1+x^2}$ grows like x as x becomes large and therefore assumes large values. In (37), we can equivalently replace $\max_{z\in E_r} |K(z)|$ by $\max_{z\in E_r} |K(z) - P_n^0(z)|$ for any

polynomial $P_n^0(z)$ of degree at most n. This is because Chebyshev interpolation is exact for polynomials. When D is large, the term x becomes quite large while x^{-n}, $n > 0$, remains small. For this reason, the Chebyshev interpolation method is unusually accurate. This explains the reduction in error by about 10^4 between Fig. 13 and 12. This behavior would not be observed in 2D or 3D.

Note that in general the accuracy of the Chebyshev method cannot be improved, at least not without changing significantly the method, whereas the Cauchy algorithm can be improved by optimizing the choice of contour in the complex plane and the subdivision of the ellipse into four arcs $\Gamma_1, \ldots, \Gamma_4$. We did not pursue this further in this paper but many improvements are possible.

6.3. *Multi-Level Algorithm*

We present results regarding the multi-level version of the scheme. The error considered is the relative L^2-error given by

$$\sqrt{\frac{\sum_i (\phi(x_i) - \tilde{\phi}(x_i))^2}{\sum_i \phi^2(x_i)}} \tag{38}$$

where $\phi(x_i) = \sum_{j \neq i} K(x_i, x_j)$ is the exact value of the product at position x_i and $\tilde{\phi}(x_i)$ is the approximation obtained through our scheme. The error is plotted against the largest rank used in the tree (constant for nonoscillatory kernels, largest at the top of the tree for oscillatory kernels). For every simulations, $2^{16} = 65536$ points uniformly distributed in an interval of length $\mathcal{L} \in \{0.1, 1, 10, 100\}$ and 8 levels were used. The results are shown in Fig. 16, 17 and 18. Again, we notice exponential decay of the error as a function of the rank.

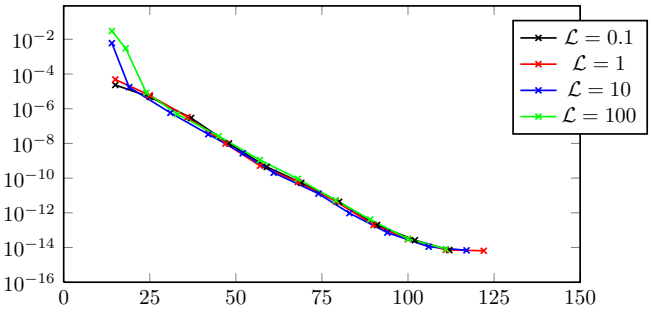

Fig. 16. L^2 relative error vs rank of approximation for Inverse Multiquadric ($\frac{1}{\sqrt{1+x^2}}$) with 65536 points uniformly distributed in the interval $[0, \mathcal{L}]$.

6.4. *Timing*

We benchmarked the algorithmic complexity of the algorithm and showed that it agrees with our $\mathcal{O}(N \log N)$ estimate. The methodology used was as follows. Particles were randomly generated in the interval $[0, 100]$ using a uniform random distribution. The number of particles was varied from $N = 2^8$ to $N = 2^{22}$. For each such distribution,

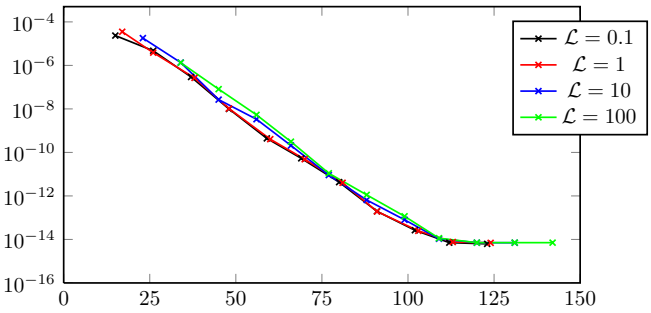

Fig. 17. L^2 relative error vs rank of approximation for Multiquadric ($\sqrt{1+x^2}$) with 65536 points uniformly distributed in the interval $[0, \mathcal{L}]$.

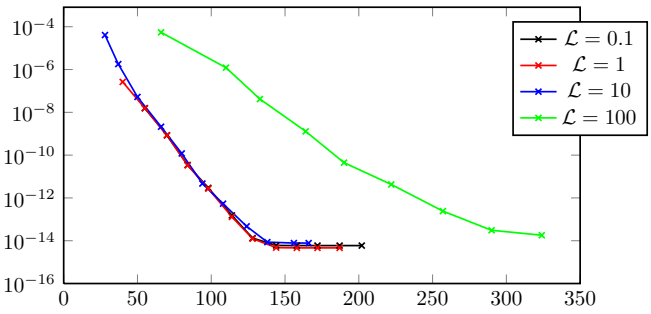

Fig. 18. L^2 relative error vs rank of approximation for Helmholtz ($\frac{e^{ix}}{x}$) with 65536 points uniformly distributed in the interval $[0, \mathcal{L}]$.

a tree was built and the number of levels that minimizes the running time was chosen. The target accuracy in each case was chosen to be $\varepsilon = 10^{-8}$. We found that the number of levels in the tree varied from 2 for $N = 2^8$ to 15 for $N = 2^{22}$.

Fig. 19, 20 and 21 show the time taken by the CFMM, and the dense (direct) matrix-vector product. The black line is the function $C\,N \log N$, shown for comparison. We see that $N \log N$ is a good model for the running timing of our scheme.

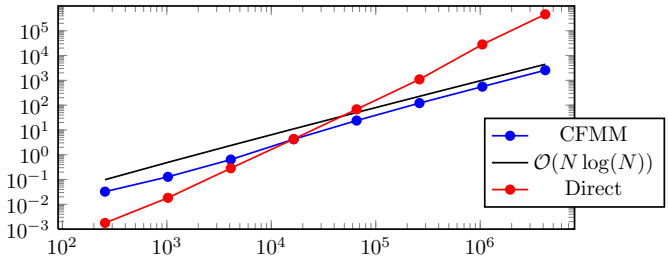

Fig. 19. Running time vs number of points (N) under a uniform distribution for the multiquadric $\sqrt{1+x^2}$

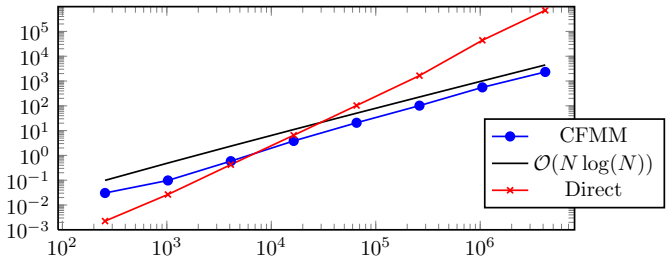

Fig. 20. Running time vs number of points (N) under a uniform distribution for the inverse multiquadric $1/\sqrt{1+x^2}$

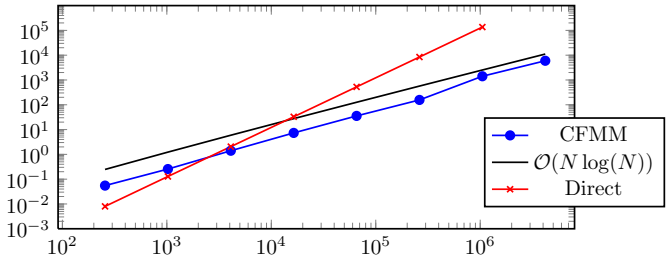

Fig. 21. Running time vs number of points (N) under a uniform distribution for Helmholtz $\frac{e^{\imath x}}{x}$

7. Conclusion

In conclusion, we have established a new general $\mathcal{O}(N \log N)$ fast multipole method based on the Cauchy integral formula and the Laplace transform. We have demonstrated the validity of the concept numerically in 1-D and obtained analytical error estimates. The computational cost can be reduced to $\mathcal{O}(N)$ by coupling this new approach with the Chebyshev interpolation technique of Fong et al. [16].

The method has multiple advantages including the fact that it works independently of the nature of the kernel as long as it is analytic in the region of interest, it has a *diagonal* multipole-to-local (M2L) transfer operator which implies a reduced complexity, and it exhibits exponential decay of the error as a function of the rank of the approximation.

Future work involves the study of various techniques that could lead to approximations with lower rank (at a given accuracy). In particular, our approximation involves sums of exponentials and in this sense shares characteristics with the work of Beylkin and Monzon [22,23]. The method in [22,23] has been in principle developed for 1D applications. Our approach can be extended to multiple dimensions and lends itself to a simple error analysis. Our approximation is however suboptimal since we consider exponentials $e^{\mu x}$ with μ either on the real axis or on the imaginary axis. The work of Beylkin and Monzon [22,23] could potentially be extended to produce optimal exponential functions with μ lying in the complex plane. This could lead to a more efficient formulation. We also mention that we have not explored all possible optimizations of the method, in particular in the choice of contours for z in the complex plane.

8. Acknowledgements

The authors would like to thank the Army High Performance Computing Research Center (AHPCRC) at Stanford and the Army Research Laboratory (contract no. W911NF-07-2-0027) for providing financial support to the authors, P-D. Létourneau and E. Darve.

Connection between optimal exponentials and the Cauchy FMM expansion

We would like to discuss extensions of the Cauchy FMM showing connections with the optimal exponentials of Beylkin et al. [22,23]. The problem we are concerned with is approximating a function $f(x)$ using exponentials. Consider the Cauchy formula:

$$f(x) = \frac{1}{2\pi i} \int_\Gamma \frac{f(z)}{z-x}\, dx$$

Prior to using the Laplace transform we need to rotate $z - x$ in the complex plane such that the real part is positive:

$$\frac{1}{z-x} = \frac{e^{-i\theta}}{e^{-i\theta}(z-x)} = e^{-i\theta} \int_0^\infty e^{-se^{-i\theta}(z-x)} ds$$

Since the contour Γ must go around the point x, avoiding the poles of f, we need to decompose Γ into separate segments and for each segment choose an appropriate θ.

The number of segments will in general vary from 1 to 4, depending on the growth of $f(z)$. In general, the upper bound can in fact be 3 (a "triangle"), although for simplicity we decided to split Γ into 4 straight segments in this work.

The integral over s can be deformed in the complex plane, making s a complex number. The form of the expansion then looks like ($se^{-i\theta} \to s$):

$$f(x) = \frac{1}{2\pi i} \sum_j \int_{\Lambda_j} \Big[\int_{\Gamma_j} f(z) e^{-sz} dz \Big] e^{sx} ds = \frac{1}{2\pi i} \sum_j \int_{\Lambda_j} F_j(s)\, e^{sx} ds$$

where Λ_j is a curve in the complex plane that starts from 0, and with asymptote the line $te^{-i\theta_j}$, $t \to \infty$, with the condition

$$\text{real}(e^{-i\theta_j}(z-x)) \geq \varepsilon > 0, \qquad \forall z \in \Gamma_j$$

As pointed out previously, we recognize a general form of the Fourier and inverse Fourier transforms; the Fourier transform is obtained by basically transforming Γ_j such that only two lines parallel to the real axis remain. After discretization of the integrals along s we obtain the desired representation in complex exponentials:

$$f(x) = \sum_j \rho_j\, e^{\tau_j x}$$

We note that with the choice of a general set of curves Λ_j in the complex planes the values of τ_j can in some sense be any point in the complex plane. In this work, we focused on the special choice $\theta_j = j\pi/2$, and with s a real positive number. The problem then becomes one of choosing optimal curves Λ_j.

This new form of the approximation resembles the general form obtained in [22,23]. In [22], several figures show results where in fact the points τ_j lie roughly along curves

(e.g., Fig. 2, 3, 5, 7). Take Fig. 2 for example. On the left panel are plotted $e^{\tau_j/b}$ with $b = 100\pi$.

The two methods are different and their results cannot be exactly equated. Nonetheless there are important connections.

References

[1] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, J. Comp. Phys. 135 (1987) 280–292.

[2] T. Hrycak, V. Rokhlin, An improved fast multipole method algorithm for potential fields, SIAM J. Sci. Comp. 19 (1998) 1804–1826.

[3] N. Yarvin, V. Rokhlin, An improved fast multipole method algorithm for potential fields on the line, SIAM J. Num. Anal. 36 (1999) 629–666.

[4] C. Cecka, E. Darve, Fourier Based Fast Multipole Method for the Helmholtz Equation, ArXiv e-prints, Mathematics — Numerical Analysis (math.NA), 65T40, 0911.4114.

[5] B. Engquist, L. Ying, Fast directional algorithm for the Helmholtz kernel, SIAM J. Sci. Comp. 29 (2007) 1710.

[6] E. Darve, The fast multipole method: a numerical implementation, J. Comp. Phys. 160 (1990) 195–240.

[7] V. Rokhlin, Rapid solution of integral equations of scattering theory in two dimensions, J. Comp. Phys. 86 (1990) 414–439.

[8] V. Rokhlin, Diagonal forms of translation operators for the Helmholtz equation in three dimensions, Tech. rep., Yale University (1992).

[9] V. Rokhlin, Diagonal forms of translation operators for the Helmholtz equation in three dimensions, Appl. Comput. Harmon. Anal. 1 (1993) 82–93.

[10] Y. Fu, G. Rodin, Fast solution for three-dimensional Stokesian many-particles problems, Comm. Numer. Meth. Eng. 16 (2000) 145–149.

[11] A. Dutt, M. Gu, V. Rokhlin, Fast algorithm for polynomial interpolation, integration and differentiation, SIAM J. Numer. Anal. 33 (5) (1996) 1689–1711.

[12] Z. Gimbutas, V. Rokhlin, A generalized fast multipole method for nonoscillatory kernels, SIAM J. Sci. Comp. 24 (2002) 796–817.

[13] L. Ying, G. Biros, D. Zorin, A kernel-independent adaptive fast multipole algorithm in two and three dimensions, J. Comp. Phys. 196 (2004) 591–626.

[14] L. Ying, A kernel-independent fast multpole algorithm for radial basis functions, J. Comp. Phys. 213 (2006) 451–457.

[15] P. Martinsson, V. Rokhlin, An accelerated kernel-independent fast multipole method in one dimension, SIAM J. Sci. Comp. 29 (3) (2007) 1160–1178.

[16] W. Fong, E. Darve, The black-box fast multipole method, J. Comp. Phys. 228 (23) (2009) 8712–8725.

[17] B. Zhang, J. Huang, N. Pitsianis, X. Sun, A Fourier-series-based kernel-independent fast multipole method, J. Comp. Phys. 230 (2011) 689–1711.

[18] A. De Boer, M. Van der Schoot, H. Bijl, Mesh deformation based on radial basis function interpolation, Comp. and Struct. 85 (11-14) (2007) 784–795.

[19] J. Carr, R. Beatson, J. Cherrie, T. Mitchell, W. Fright, B. McCallum, T. Evans, Reconstruction and representation of 3D objects with radial basis functions, in: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, ACM, 2001, p. 76.

[20] M. Messner, M. Shanz, E. Darve, Fast directional multilevel summation for oscillatory kernels based on Chebyshev interpolation, J. Comp. Phys. 231 (2012) 1175–1196.

[21] L. Greengard, V. Rokhlin, A new version of the fast multipole method for the Laplace equation in three dimensions, Acta numerica 6 (2008) 229–269.

[22] G. Beylkin, L. Monzón, On approximation of functions by exponential sums, J. Appl. Comp. Harm. Anal. 19 (2005) 17–48.

[23] G. Beylkin, L. Monzón, Approximation of functions by exponential sums revisited, J. Appl.Comp. Harm. Anal. 28 (2010) 131–149.

[24] F. Andersson, M. Carlsson, M. de Hoop, Nonlinear approximation of functions in two dimensions by sums of exponential functions, App. Comp. Harm. Anal. 29 (2010) 156–181.
[25] F. Andersson, M. Carlsson, M. de Hoop, Nonlinear approximation of functions in two dimensions by sums of wave packets, App. Comp. Harm. Anal. 29 (2010) 198–213.
[26] P. Létourneau, C. Cecka, E. Darve, Generalized fast multipole method, in: World Congress on Computational Mathematics Proceedings, 2010.
[27] C. Cecka, P. Létourneau, E. Darve, Fast multipole method using the Cauchy integral formula, in: Banff International Research Proceedings, 2010.
[28] A. Dutt, V. Rokhlin, A Fourier-series-based kernel-independent fast multipole method, SIAM J. Sci. Comput. (1993) 1368–1393.
[29] G. Beylkin, On the fast Fourier transform of functions with singularities, Appl. Comput. Harmon. Anal. (1995) 363–381.
[30] A. Appel, An efficient program for many-body simulation, SIAM J. Sci. Stat. Comp. 6 (1) (1985) 85–103.
[31] J. Barnes, P. Hut, A hierarchical O(N-Log-N) force-calculation algorithm, Nature 324 (6096) (1986) 446–449.
[32] M. Abramowitz, I. Stegun, Handbook of Mathematical Functions, tenth Edition, National Bureau of Standards, 1972.
[33] L. Greengard, J. Huang, V. Rokhlin, S. Wandzura, Accelerating fast multipole methods for the Helmholtz equation at low frequencies, IEEE Comput. Sci. Eng. 5 (1998) 32–38.
[34] H. Cheng, W. Y. Crutchfield, Z. Gimbutas, L. F. Greengard, J. F. Ethridge, J. Huang, V. Rokhlin, N. Yarvin, J. Zhao, A wideband fast multipole method for the Helmholtz equation in three dimensions, J. Comp. Phys. 216 (2006) 300–325.
[35] G. Beylkin, G. Kurcz, L. Monzón, Fast convolution with the free space Helmholtz Green's function, J. Comp. Phys. 228 (2009) 2770–2791.
[36] M. Messner, M. Schanz, E. Darve, Fast directional multilevel summation for oscillatory kernels based on Chebyshev interpolation, J. Comp. Phys. 231 (2011) 1175–1196.