Available online at www.sciencedirect.com

ScienceDirect

Procedia IUTAM 18 (2015) 56 – 65

Procedia IUTAM

www.elsevier.com/locate/procedia

IUTAM Symposium on Particle Methods in Fluid Mechanics

High order Poisson solver for unbounded flows

Mads Mølholm Hejlesen[a], Johannes Tophøj Rasmussen[a], Philippe Chatelain[b], Jens Honoré Walther[a,c,*]

[a] *Department of Mechanical Engineering, Technical University of Denmark, Building 403, DK-2800 Kgs. Lyngby, Denmark*
[b] *Institute of Mechanics, Materials and Civil Engineering, Université catholique de Louvain, B-1348, Belgium*
[c] *Computational Science and Engineering Laboratory, ETH Zürich, Universitätsstrasse 6, CH-8092 Zürich, Switzerland*

Abstract

This paper presents a high order method for solving the unbounded Poisson equation on a regular mesh using a Green's function solution. The high order convergence was achieved by formulating mollified integration kernels, that were derived from a filter regularisation of the solution field. The method was implemented on a rectangular domain using fast Fourier transforms (FFT) to increase computational efficiency. The Poisson solver was extended to directly solve the derivatives of the solution. This is achieved either by including the differential operator in the integration kernel or by performing the differentiation as a multiplication of the Fourier coefficients. In this way, differential operators such as the divergence or curl of the solution field could be solved to the same high order convergence without additional computational effort. The method was applied and validated using the equations of fluid mechanics as an example, but can be used in many physical problems to solve the Poisson equation on a rectangular unbounded domain. For the two-dimensional case we propose an infinitely smooth test function which allows for arbitrary high order convergence. Using Gaussian smoothing as regularisation we document an increased convergence rate up to tenth order. The method however, can easily be extended well beyond the tenth order. To show the full extend of the method we present the special case of a spectrally ideal regularisation of the velocity formulated integration kernel, which achieves an optimal rate of convergence.

Selection and/or peer-review under responsibility of the Technical University of Denmark, Department of Mechanical Engineering.

Keywords: Poisson solver; Elliptic solver; unbounded domain; infinite domain; isolated system; Green's function solution; Numerical integration; Vortex methods; Particle-mesh methods

1. Introduction

In particle based vortex methods the Poisson equation arises when calculating the stream function or velocity field from the vorticity. This is usually solved by applying a Green's function solution in form of the Biot-Savart equation [1]. Here each particle induces a velocity onto every other particle in the system and thus poses an N_p-body problem. The direct evaluation of the mutual particle interaction amounts to an inefficient scaling of $\mathcal{O}(N_p^2)$ and alternative methods have been developed in order to obtain a more efficient scaling.

* Corresponding author. Tel.: + 45 4525 4327; fax: + 45 4588 4325.
E-mail address: jhw@mek.dtu.dk

Selection and/or peer-review under responsibility of the Technical University of Denmark, Department of Mechanical Engineering.
doi:10.1016/j.piutam.2015.11.006

One solution to the scaling problem is found in tree algorithms such as the Barnes-Hut [2] or the fast multi-pole method (FMM) [3]. Here the particles are grouped according to particle density and distance to the evaluated particle and only the collective contribution from the group is considered. The FMM does not rely on any mesh and is capable of achieving an optimal scaling of $\mathcal{O}(N_p)$, however, it is subject to a large pre-factor when constructing and traversing the tree data structure.

Another inconvenience of mesh-free vortex methods is the lack of a provable convergence theory. This has motivated development of particle re-meshing [4, 5] and hybrid particle-mesh methods such as the vortex-in-cell (VIC) method by Christiansen [6]. The VIC method solves the Poisson equation by interpolating the particle vorticity to a regular mesh where the velocity is calculated by a mesh based solver. This ensures the numerical convergence of the method and makes possible efficient mesh based Poisson solvers and finite difference stencils for the evaluation of derivatives.

The Poisson solver of Hockney and Eastwood [7] utilizes the regular mesh of the VIC method by using highly efficient fast Fourier transforms (FFT). The method is based on calculating a Green's function solution through the convolution of the vorticity field with the Green's function. This is effectively performed by Fourier transforming the vorticity field and integration kernel after which the convolution is then performed by a simple multiplication of the discrete Fourier coefficients. In a periodic domain, spectral differentiation is obtained in Fourier space simply by multiplying the coefficients by their respective wave number. The periodic boundary condition is naturally treated by the periodicity of the Fourier series and a spectral convergence is therefore easily obtained as shown by Rasmussen [8]. For the unbounded free-space problem, the integration kernel cannot be represented exactly in Fourier space due to the inherent periodicity of the Fourier transform. On the same grounds the domain must be extended to twice the size, to avoid the periodicity of the discrete convolution.

Hockney and Eastwood [7] showed that it is possible to impose free-space boundary conditions by zero-padding the vorticity field and convolving this with a free-space integration kernel of equal size. At first sight, extending the domain size to $2N$ in each unbounded direction increases the computational scaling of the d-dimensional solver to $\mathcal{O}(2^d dN^d \log(2N))$ and requires an allocated memory of $(2N)^d$ for a fully unbounded domain. However, an FFT based convolution algorithm, as proposed by [7], significantly reduces the computational scaling and the memory needed. This is achieved by exploiting the fact that a multidimensional Fourier transform consists of series of one-dimensional transforms carried out in each direction, sequentially. The method of Hockney and Eastwood can be extended to mixed periodic-free boundary conditions, as was shown by Chatelain and Koumoutsakos [9].

In the present work the Green's function based Poisson solver was extended to achieve a high order of convergence of the convolution integral for a continuous field by formulating regularized integration kernels. These were derived based on the work of [10, 11, 12, 13, 14], who used smoothing functions to achieve high order regularized integration kernels for mesh-free vortex methods. However, unlike mesh-free methods, the mesh-based Poisson solver requires a non-singular, continuous integration kernel to ensure an accurate numerical solution. Hence, the centre value of the regularised integration kernels must be carefully defined as this is a crucial factor for achieving high order convergence [8].

In this paper the derivation and validation of a high order Poisson solver for the two-dimensional case ($d = 2$) is described. Mollified integration kernels as used in [10, 11, 12, 13, 14], were derived by regularisation of the vorticity field. Gaussian smoothing was used for regularisation and an increased order was achieved by de-filtering the regularisation function to conserve higher order moments of the vorticity field. A special case is presented where a velocity kernel was derived using a spectrally ideal regularisation by which an optimal convergence rate was achieved. The solver was extended to directly calculate the velocity field to the same high order as the Poisson solver without any additional numerical effort. For the three-dimensional case, which is similar in both derivation and results, the reader is referred to [15].

2. Method

2.1. The Poisson equation in fluid mechanics and its Green's function solution

The vorticity ω is an intrinsic flow component which arises when performing a functional orthogonal decomposition of the velocity field v:

$$\nabla^2 v = \nabla(\nabla \cdot v) - \nabla \times (\nabla \times v) = \nabla \vartheta - \nabla \times \omega. \tag{1}$$

Here the divergence and curl of the velocity field are defined as the dilation $\vartheta \equiv \nabla \cdot v$, and the vorticity $\omega \equiv \nabla \times v$, respectively. For incompressible flow, the velocity field is solenoidal leaving $\vartheta = \nabla \cdot v = 0$, which yields

$$\nabla^2 v = -\nabla \times \omega. \tag{2}$$

This equation can be inverted to find the velocity field from a known vorticity field by convolving the equation with a Green's function:

$$G * \nabla^2 v = -G * \nabla \times \omega. \tag{3}$$

As derivatives are commutative with respect to convolution we may an expression for the velocity field may be obtained by assuming that the Laplacian of the Green's function is equal to the Dirac delta function: $\nabla^2 G(x) = \delta(x)$. This leaves:

$$v = -\nabla \times (G * \omega). \tag{4}$$

From the definition of the stream-function $v \equiv \nabla \times \psi$ it may be seen that this is easily obtained from

$$\psi = -G * \omega, \tag{5}$$

which corresponds to solving the Poisson equation $\nabla^2 \psi = -\omega$. The well-known Green's function G for the two-dimensional free-space Poisson equation is then used to solve for the stream function. Alternatively, the velocity field is calculated directly by constructing integration kernels for the velocity field by

$$v = K * \omega \tag{6}$$

where $K(x) = \nabla G(x) \times$. Hence the two integration kernels are

$$G(x) = -\frac{1}{2\pi} \log |x| \qquad \text{and} \qquad K(x) = -\frac{1}{2\pi |x|^2} \begin{pmatrix} y \\ -x \end{pmatrix}. \tag{7}$$

The integration kernels are both singular at $|x| = 0$ and results in an $\mathcal{O}(h^2)$ convergence of the Poisson solver where h is the discretisation length.

2.2. Unbounded Poisson solver based on fast Fourier transforms

The convolutions of Eqs. (5) and (6) can be solved for free-space boundary conditions by performing the discrete convolution in a linear sense. This is achieved by zero-padding the vorticity field to twice the domain size and convolving the now zero-padded vorticity field and the fully defined integration kernel of equal size. The convolution can be performed efficiently in Fourier space where the linearity of the problem also makes differentiation and thereby curl of Eq. (4) capable of being carried out in Fourier space, thus reducing the number of integration kernels needed. The convolution equations in Fourier space are written as

$$\widehat{\psi} = \widehat{G}\,\widehat{\omega} \qquad \text{and} \qquad \widehat{u} = \widehat{K}\,\widehat{\omega} \qquad \text{or} \qquad \widehat{u} = \imath k \times \widehat{\psi} \tag{8}$$

where $\widehat{\cdot}$ denotes the Fourier coefficients obtained by discrete Fourier transforms, $\imath$ is the imaginary unit and $k = \{k_x, k_y(, k_z)\}$ is the wave number corresponding to the respective Fourier coefficient. In 2D only the out-of-plane component exists as $w = \{0, 0, \omega_3\}$ and $\psi = \{0, 0, \psi_3\}$ which explains the rank of K in Eq. (7).

Evidently, the domain doubling technique has a severe impact on the convolution footprint, however it can be reduced from $(2N)^d$ to $2N^d + (2N)^{d-1}$. Hockney and Eastwood [7] propose to carry out a transform in one direction and then proceed row by row for $d = 2$ or plane by plane for $d = 3$. Here the remaining dimension(s) is zero-padded, Fourier-transformed, multiplied by $\widehat{G}$ and inverse transformed whereas the zero-padding is truncated before moving on to next row or plane. This method, however, still requires the integration kernels G or K to be calculated initially and stored in their full size of $(2N)^d$.

2.3. Derivation of regularized integration kernels

In the derivation of high order integration kernels for a given scalar or vector field defined on a mesh, is achieved by using the regularized particle representation [10, 11, 12, 13, 14]. Effectively, a regularization of the flow field is considered

$$v_\zeta = \zeta * v \qquad \text{and} \qquad \psi_\zeta = \zeta * \psi \tag{9}$$

where ζ is a conserving regularization function. Re-writing Eq. (3) where the derivatives have been commuted and using a regularized Greens function G_ζ yields:

$$\nabla^2 G_\zeta * v = -\nabla \times (G_\zeta * \omega). \tag{10}$$

Here it is seen that the flow regularization can be achieved by fulfilling:

$$\nabla^2 G_\zeta(x) = -\zeta(x). \tag{11}$$

Hence for a given regularization function ζ, the corresponding integration kernel G_ζ can be derived. For simplicity this was performed in polar (r,θ) coordinates for $\mathbb{R}^2$. The regularized Green's equation (Eq. (11)) is given in polar coordinates by

$$-\zeta(r) = \frac{1}{r}\frac{d}{dr}\left(r\frac{dG_\zeta}{dr}\right) \tag{12}$$

The weight function w is defined from the regularization function as

$$w(r) = \int_0^r \zeta(t)\,\mathrm{d}t = 2\pi\int_0^r t\zeta(t)\,\mathrm{d}t. \tag{13}$$

Combining Eqs. (12) and (13) renders an expression for the regularized integration kernel

$$G_\zeta(r) = -\frac{1}{2\pi}\int \frac{w(r)}{r}\,\mathrm{d}r \tag{14}$$

2.4. High order smoothing functions and corresponding integration kernels

From Eq. (11) it may be seen that for the derivation of higher order integration kernels the Green's equation is replaced by a regularized version and thus ζ represents some approximation to the delta function. Its conservation properties can be determined by its frequency response as given by the Fourier transform. For this the radial Fourier transforms are used, where $k = 1/r$ is the radial wave number

$$\widehat{\zeta}(k) = (2\pi)^{\frac{d}{2}}\int_0^\infty \zeta(r)\frac{J_{\frac{d}{2}-1}(kr)}{(kr)^{\frac{d}{2}-1}}r^{d-1}\,\mathrm{d}r \qquad \text{and} \qquad \zeta(r) = \frac{1}{(2\pi)^{\frac{d}{2}}}\int_0^\infty \widehat{\zeta}(k)\frac{J_{\frac{d}{2}-1}(kr)}{(kr)^{\frac{d}{2}-1}}k^{d-1}\,\mathrm{d}k \tag{15}$$

Here J_ν is the Bessel function of order ν.

The regularization function may be described from a number of different functions, which as a minimum must have conservative properties i.e. $\int_{\mathbb{R}^d}\zeta(x)\,dx = 1$ or equally $\widehat{\zeta}(0) = 1$. In this work the derivation and convergence of a class of high order Gauss functions is described and a special case of ideal regularization based on Bessel functions is demonstrated.

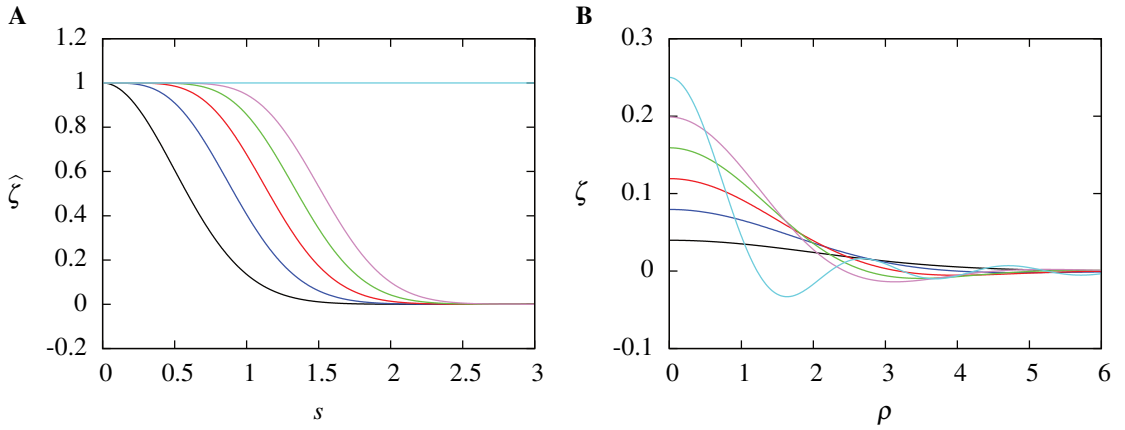

Fig. 1. The frequency response of the regularization function (A) and the corresponding regularization functions in real space (B). The Gauss functions ζ_g with $\varepsilon = 2h$ for $m = 2$: (——); $m = 4$: (——); $m = 6$: (——); $m = 8$: (——); $m = 10$: (——), and the Bessel function regularization ζ_J (——).

2.5. Regularization by Gaussian smoothing

One type of regularization is obtained by Gaussian smoothing of the vorticity field. For this, the regularization is given by the Gauss function ζ_g which, along with its frequency response $\widehat{\zeta}_g$, is given by

$$\zeta_g(r) = \frac{1}{(2\pi\varepsilon^2)^{d/2}} \exp\left(-\frac{\rho^2}{2}\right) \qquad \text{and} \qquad \widehat{\zeta}_g(k) = \exp\left(-\frac{s^2}{2}\right). \tag{16}$$

Here $\rho = r/\varepsilon$ and $s = k\varepsilon$ are the radial coordinate and wave number normalized with ε, which in this case represents a smoothing radius. The frequency response has a bandwidth of $\sqrt{2}$ and gives a second order smoothing. A higher order smoothing is obtained by increasing the bandwidth of the frequency response, thus increasing the conservation properties of the regularization function, as is apparent in Fig. 1.

A high order Gaussian regularization ζ_m of order m can be obtained by applying a deconvolution

$$\widehat{\zeta}_m = D_m \widehat{\zeta}_g. \tag{17}$$

Here D_m is an approximation of the inverse regularization response. To find D_m, the Taylor expansion of the inverse Gauss function was used given by:

$$\left(\exp\left(-\frac{s^2}{2}\right)\right)^{-1} = 1 + \frac{1}{2}s^2 + \frac{1}{8}s^4 + \frac{1}{48}s^6 + \ldots = \sum_{n=0}^{\infty} \frac{(s^2/2)^n}{n!}. \tag{18}$$

An m-th order Taylor expansion is then obtained by the truncated series

$$D_m = \sum_{n=0}^{m/2-1} \frac{(s^2/2)^n}{n!}. \tag{19}$$

Thus the frequency response of an m-th order Gaussian function is given by

$$\widehat{\zeta}_m(k) = \sum_{n=0}^{m/2-1} \frac{(s^2/2)^n}{n!} \exp\left(-\frac{s^2}{2}\right). \tag{20}$$

Using the inverse Fourier transform of Eq. (15) the smoothing functions are obtained. The 2D smoothing functions for $m = 4, 6, 8, 10$ are given by

$$\zeta_4(r) = \left(2 - \frac{1}{2}\frac{r^2}{\varepsilon^2}\right)\frac{\exp\left(-\frac{r^2}{2\varepsilon^2}\right)}{2\pi\varepsilon^2} \qquad \zeta_8(r) = \left(4 - 3\frac{r^2}{\varepsilon^2} + \frac{1}{2}\frac{r^4}{\varepsilon^4} - \frac{1}{48}\frac{r^6}{\varepsilon^6}\right)\frac{\exp\left(-\frac{r^2}{2\varepsilon^2}\right)}{2\pi\varepsilon^2}$$
$$\zeta_6(r) = \left(3 - \frac{3}{2}\frac{r^2}{\varepsilon^2} + \frac{1}{8}\frac{r^4}{\varepsilon^4}\right)\frac{\exp\left(-\frac{r^2}{2\varepsilon^2}\right)}{2\pi\varepsilon^2} \quad \zeta_{10}(r) = \left(5 - 5\frac{r^2}{\varepsilon^2} + \frac{5}{4}\frac{r^4}{\varepsilon^4} - \frac{5}{48}\frac{r^6}{\varepsilon^6} + \frac{1}{384}\frac{r^8}{\varepsilon^8}\right)\frac{\exp\left(-\frac{r^2}{2\varepsilon^2}\right)}{2\pi\varepsilon^2}. \quad (21)$$

The smoothing functions and their frequency response are shown in Fig. 1 for $\varepsilon = 2h$. It may be seen as the order of the regularization is increased, the bandwidth of the frequency response increases. This manifests in real-space as oscillations of the regularization function and consequentially, the conservation of higher order moments which is given by

$$0^\beta = 2\pi\int_0^\infty \rho^\beta \zeta_m(\rho)\rho \, d\rho \quad (22)$$

where $\beta = \{0, 2, ..., m-2\}$.

Once defined, the smoothing functions can be used to derive the integration kernel by Eqs. (13) and (14). The regularized integration kernels from m-th order Gaussian smoothing can be written on the general form

$$G_m(x) = -\frac{1}{2\pi}\left(\log(|x|) - R_m\left(\frac{|x|}{\varepsilon}\right)\exp\left(\frac{-|x|^2}{2\varepsilon^2}\right) + \frac{1}{2}\mathrm{E}_1\left(\frac{|x|^2}{2\varepsilon^2}\right)\right) \quad (23)$$

where R_m is an even polynomial which for $m = 4, 6, 8, 10$ yields:

$$R_4(\rho) = \frac{1}{2} \qquad R_8(\rho) = \frac{11}{12} - \frac{7}{24}\rho^2 + \frac{1}{48}\rho^4$$
$$R_6(\rho) = \frac{3}{4} - \frac{1}{8}\rho^2 \qquad R_{10}(\rho) = \frac{25}{24} - \frac{23}{48}\rho^2 + \frac{13}{192}\rho^4 - \frac{1}{384}\rho^6. \quad (24)$$

In order to define $G_m(0)$ the exponential integral function $\mathrm{E}_1(z)$ can be expanded to an infinite series [16]:

$$\mathrm{E}_1(z) = -\gamma - \log(z) - \sum_{n=1}^{\infty}\frac{(-1)^n z^n}{n\,n!}. \quad (25)$$

The centre value $G_m(0)$ is then given by

$$G_m(0) = \lim_{x\to 0}(G_m) = \frac{1}{2\pi}\left(\frac{\gamma}{2} - \log\left(\sqrt{2}\varepsilon\right) + R_m(0)\right) \quad (26)$$

where $\gamma = 0.5772156649$ is Euler's constant. The corresponding velocity kernels, obtained by $K_m = \nabla G_m \times$, is in the general form given by:

$$K_m(x) = -\frac{1}{2\pi|x|^2}\begin{pmatrix} y \\ -x \end{pmatrix}\left(1 - Q_m\left(\frac{|x|}{\varepsilon}\right)\exp\left(\frac{-|x|^2}{2\varepsilon^2}\right)\right) \quad (27)$$

where Q_m is an even polynomial which for $m = 2, 4, 6, 8$ yields:

$$Q_4(\rho) = 1 - \frac{1}{2}\rho^2 \qquad Q_8(\rho) = 1 - \frac{3}{2}\rho^2 + \frac{3}{8}\rho^4 - \frac{1}{48}\rho^6$$
$$Q_6(\rho) = 1 - \rho^2 + \frac{1}{8}\rho^4 \qquad Q_{10}(\rho) = 1 - 2\rho^2 + \frac{3}{4}\rho^4 - \frac{1}{12}\rho^6 + \frac{1}{384}\rho^8. \quad (28)$$

The centre value of the velocity kernel is defined as $K_m(0) = 0$.

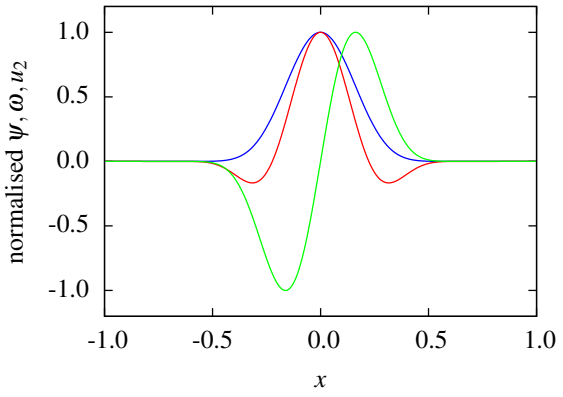

Fig. 2. The normalized test function and the corresponding solution for $c = 20$ and $R = 1$ at $(x, y) = (\cdot, 0)$. (——): Vorticity $\omega/\max(\omega)$; (——): Stream function $\psi/\max(\psi)$; (——): Velocity $v/\max(u)$.

2.6. Ideal regularization by Bessel functions

As we wish to conserve the moments of the non-regularized field, an ideal regularization is obtained when $\widehat{\zeta}(s) = 1$ i.e. when every Fourier coefficient is conserved. For the 2D case it is seen from the radial Fourier transform (Eq. (15)) that an ideal regularization can be obtained by the Bessel function

$$\zeta_J(r) = \frac{J_1(\rho)}{2\pi r} \tag{29}$$

after which the frequency response is obtained as:

$$\widehat{\zeta}_J(k) = 2\pi \int_0^\infty \zeta_J(r) J_0(kr) r \,\mathrm{d}r = \int_0^\infty J_1(r/\varepsilon) J_0(kr) \,\mathrm{d}r = 1. \tag{30}$$

Here the regularization parameter ε determines the frequency of the Bessel function oscillations and not a smoothing radius as with the Gaussian smoothing. The oscillation of Bessel functions J_ν converges to a length of 2π. This means that the regularization parameter is restricted by $\varepsilon \geq h/\pi$ and the optimal rate of convergence is found when $\varepsilon = h/\pi$.

From Eqs. (13) and (14) we obtain the corresponding integration kernel

$$G_J(r) = -\frac{1}{2\pi} \int \frac{1 - J_0(\rho)}{r} \, d\rho \tag{31}$$

which by radial differentiation and Cartesian curl yields the velocity kernel:

$$K_J(x) = -\frac{1 - J_0(\rho)}{2\pi |x|^2} \begin{pmatrix} y \\ -x \end{pmatrix} \tag{32}$$

3. Results

3.1. Validation: compact vortex blob

To investigate the convergence of the free-space Poisson solver, the test function i.e. the vorticity distribution, must be compact within the computational domain. This can be achieved in the two-dimensional case by using a vortex blob to form a vorticity patch. In this study, the use of a bump function distribution with an infinite number of continuous derivatives was used to avoid limitations of the convergence rate. The bump function is defined as:

$$f(z) = \begin{cases} \exp\left(-\dfrac{c}{1 - z^2}\right) & \text{for } |z| < 1 \\ 0 & \text{for } |z| \geq 1 \end{cases} \tag{33}$$

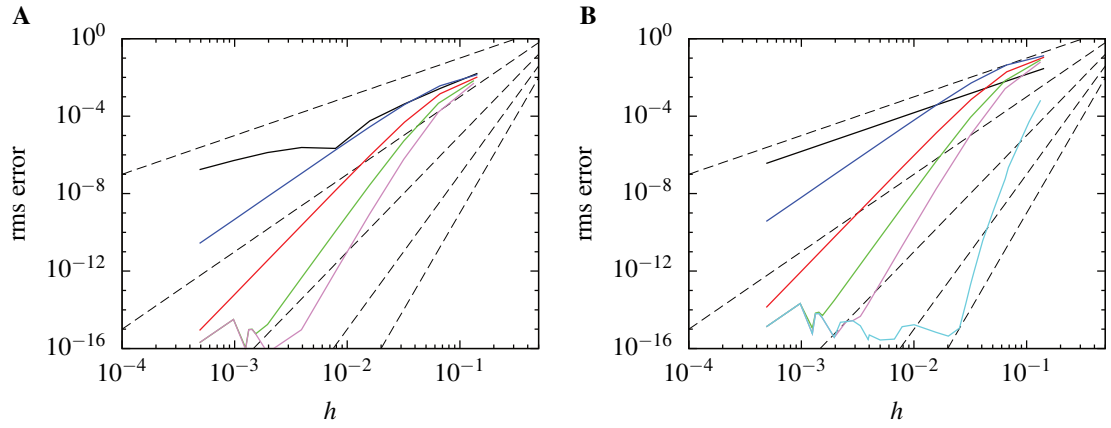

Fig. 3. The rms error of the resulting stream function (A) and velocity field (B) using Gaussian regularized integration kernels G_m and K_m with $\varepsilon = 2h$ and the ideal velocity kernel K_J. The calculated error for the velocity in (B) is virtually identical when calculated by the velocity kernels K_m as when using the integration kernel G_m with spectral differentiating in Eq. (8). (– – – –): from top $h^2, h^4, h^6, h^8, h^{10}$; (———): non-regularized kernels G and K; (———): $m = 4$; (———): $m = 6$; (———): $m = 8$; (———): $m = 10$; (———): K_J.

where c is an arbitrary positive constant set to 20 in this study. Using centred polar (r, θ) coordinates, the stream function is defined as:

$$\psi = f\left(\frac{r}{R}\right) \tag{34}$$

where R is the radius of the vorticity patch. The initial vorticity field and the resulting velocity field were found analytically, and is shown in Fig. 2.

The root-mean-squared (rms) error of the test case is shown in Fig. 3 for the regularized integration kernels G_m and K_m with $m = 4, 6, 8$ and K_J. As may be seen, the convergence rate of the Gaussian smoothed integration kernels corresponds to the respective design of the smoothing functions and the ideal velocity kernel K_J exhibits a spectral convergence. The same convergence rate was observed with the maximum error (not shown). The solution of the velocity field for the Gaussian based kernels were found to be practically identical, when using the velocity kernels K_m and when using spectral differentiation in Eq. (8). As spectral differentiation uses a single integration kernel, it is considered the favourable choice in order to reduce the memory that must be allocated for the integration kernels.

3.2. Investigations of the Gaussian smoothing radius

Introducing a regularization of the field causes a smoothing error which is governed by the order of the regularization function [17, 18, 19, 20] and is is bounded by

$$\|\omega_\zeta - \omega\| < C_1\,\varepsilon^m\,\|\omega\|, \qquad \|\psi_\zeta - \psi\| < C_2\,\varepsilon^m\,\|\omega\| \qquad \text{and} \qquad \|u_\zeta - u\| < C_3\,\varepsilon^m\,\|\omega\| \tag{35}$$

where m is the order of the regularization function.

Note that the discretisation of the convolution above leads to a second error contribution. It is due to the underlying quadrature of the integral at the grid points and leads to a discretisation error of

$$\|\psi_\zeta - \psi_\zeta^h\| < C_4\,\varepsilon\left(\frac{h}{\varepsilon}\right)^l \|\omega\| \qquad \text{and} \qquad \|u_\zeta - u_\zeta^h\| < C_5\,\varepsilon^2\left(\frac{h}{\varepsilon}\right)^l \|\omega\|\,. \tag{36}$$

where l depends on the smoothness of the smoothing function ζ. The discretisation errors (Eqs. (36)) converges only when $\varepsilon > h$. This means that the particle kernel must overlap other particle locations in order to ensure the convergence of the method, which is a well-known fact in particle methods [17, 18, 19, 20, 21].

In the previous section, the validation relied on simply keeping $\varepsilon = \alpha h$ where $\alpha = 2$. A more rigorous approach, which is consistent with the error behaviour of the convolution, was suggested by Beale and Majda [20, 19, 12] in

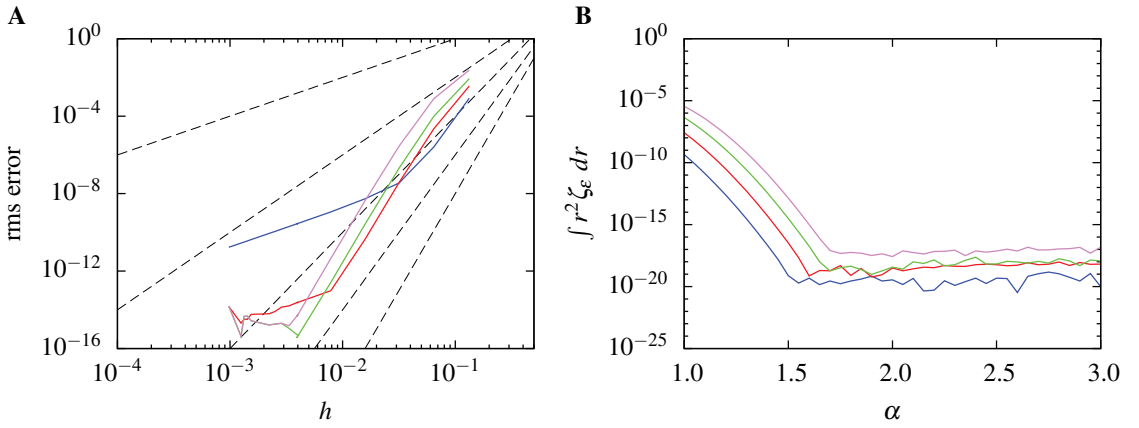

Fig. 4. (A) The rms error of the G_m integration kernel with $m = 10$ for $\varepsilon = \alpha h$ using different values of α. (– – – –): from top h^2, h^4, h^6, h^8, h^{10}; (——): $\alpha = 1.0$; (——): $\alpha = 1.25$; (——): $\alpha = 1.5$; (——): $\alpha = 1.75$. (B) The calculated discrete 2nd order moments of the Gaussian smoothing function ζ_m using different order kernels. (——): $m = 4$. (——): $m = 8$; (——): $m = 12$; (——): $m = 16$.

their convergence proof. They choose the smoothing radius as $\varepsilon = h^q$ with $0 < q < 1$. The corresponding error then scales as $\mathcal{O}(h^{qm})$, which can be made close to $\mathcal{O}(h^m)$ for q close to 1.

However, choosing the smoothing radius ε is not trivial as a too conservative choice of α might lead to a breakdown of the convergence rate, as seen in Fig. 4A, for the two-dimensional G_m kernel where $m = 10$. The same behaviour is also found using $\varepsilon = h^q$ when q is close to 1 (not shown). It may be seen that keeping ε proportional to h will produce a $\mathcal{O}(h^{10})$ error, but only up to a certain point: eventually, the discretisation error overwhelms the smoothing error, which causes the convergence rate to break down. This crossover point can be offset by increasing the overlap α, albeit at the cost of a higher smoothing error, as is apparent in Fig. 4A.

In Fig. 4B it may be seen to what extent the second discrete moment is conserved when using different values of α. The Gaussian smoothing functions of $m = 4, 8, 12, 16$ were considered. It is clear that in order for the discrete moments to be satisfied to the accuracy of the quadrature i.e. at the kinks of the curves, the smoothing radius must be greater than the grid size by a margin or overlap α which is related to the order of the smoothing function m. A higher order kernel will require a larger overlap to achieve the same moment error as a lower order one: as expected, higher order polynomials (Eq. (21)) require more points to be captured accurately.

4. Conclusion

A convolution integral method was presented for calculating the solution to the Poisson equation for a continuous field in an unbounded domain to an arbitrarily high order of convergence. The regularization method applied in mesh-free vortex methods to produce vortex blobs was combined with the FFT-based Poisson solver of particle-mesh methods to achieve a high rate of convergence without additional computational cost.

The method was shown to be able to calculate the derivative of the solution explicitly, either by direct spectral differentiation or by analytically differentiating the integration kernel. This enabled the evaluation of the curl to be performed directly by the solver and eliminated the need for an additional numerical scheme while maintaining the specified order of convergence. The two methods of calculating the derivative were found to give practically identical results and convergence behaviour. As spectral differentiation requires only a single integration kernel, the allocated memory can be reduced by choosing this method.

5. Acknowledgements

The research was supported by the Danish Research Council of Independent Research (Grant. No. 274-08-0258). We would also like to acknowledge the helpful discussions with Thomas Beale, Grégoire S. Winckelmans, Diego Rossinelli, Wim van Rees and Petros Koumoutsakos.

References

[1] Leonard A. Vortex Methods for Flow Simulation. J Comput Phys. 1980;37:289–335.

[2] Barnes J, Hut P. A hierarchical $O(N \log N)$ force-calculation algorithm. Nature. 1986;324(4):446–449.

[3] Carrier J, Greengard L, Rokhlin V. A fast adaptive multipole algorithm for particle simulations. SIAM J Sci Stat Comput. 1988;9(4):669–686.

[4] Koumoutsakos P, Leonard A. High-resolution simulation of the flow around an impulsively started cylinder using vortex methods. J Fluid Mech. 1995;296:1–38.

[5] Koumoutsakos P. Multiscale Flow Simulations Using Particles. Annu Rev Fluid Mech. 2005;37:457–487.

[6] Christiansen JP. Numerical Simulation of Hydrodynamics by the Method of Point Vortices. J Comput Phys. 1973;13:363–379.

[7] Hockney RW, Eastwood JW. Computer Simulation Using Particles. 2nd ed. Institute of Physics Publishing, Bristol, PA, USA; 1988.

[8] Rasmussen JT. Particle Methods in Bluff Body Aerodynamics [Ph.D. thesis]. Technical University of Denmark; 2011.

[9] Chatelain P, Koumoutsakos P. A Fourier-based elliptic solver for vortical flows with periodic and unbounded directions. J Comput Phys. 2010;229:2425–2431.

[10] Leonard A. Vortex Methods for Flow Simulation. J Comput Phys. 1980;37:289–335.

[11] Perlman M. On the accuracy of vortex methods. J Comput Phys. 1985;59:200–223.

[12] Beale JT, Majda A. High order accurate vortex methods with explicit velocity kernels. J Comput Phys. 1985;58:188–208.

[13] Hald OH. Convergence of vortex methods for Euler's equations, III. SIAM J Numer Anal. 1987;24(3):538–582.

[14] Winckelmans GS, Leonard A. Contribution to Vortex Particle Methods for the Computation of Three-Dimensional Incompressible Unsteady Flows. J Comput Phys. 1993;109:247–273.

[15] Hejlesen MM, Rasmussen JT, Chatelain P, Walther JH. A high order solver for the unbounded Poisson equation. J Comput Phys. 2013;252:458–467.

[16] Abramowitz M, Stegun IA. Handbook of Mathematical Functions With Formulas, Graphs and Mathematical Tables. National Bureau of Standards. Applied Mathematics Series. 55; 1972.

[17] Raviart PA. Particle approximation of first order systems. J Comput Math. 1986;4(1):50–61.

[18] Hald OH. Convergence of vortex methods for Euler's equations. II. SIAM J Numer Anal. 1979;16:726–755. 5.

[19] Beale JT, Majda A. Vortex methods. I: Convergence in Three Dimensions. Math Comput. 1982;39(159):1–27.

[20] Beale JT, Majda A. Vortex Methods. II: Higher Order Accuracy in Two and Three Dimensions. Math Comput. 1982;39(159):29–52.

[21] Cottet GH, Koumoutsakos P. Vortex Methods – Theory and Practice. New York: Cambridge University Press; 2000.