

CMIS Hand-in 6: Finite Volume Method

Nikolai Plambeck Nielsen

lpk331@alumni.ku.dk

Niels Bohr Institute, University of Copenhagen

1 FINITE VOLUME METHOD

The finite volume method uses the approximation of piecewise integrals to solve a partial differential equation. One takes the governing equation and integrates it over some finite, fixed volume V . Now one can split the volume into several smaller volumes, called *control volumes*. One can then use numerical approximations to evaluate the integral and obtain a set of linear equations for each control volume. These equations can then be gathered into one matrix and solved for the unknown field.

In this case we are solving a magnetostatics problem. We have a domain S defined for $x \in [-3, 3]$ and $y \in [-1, 1]$, where we have a magnetisation on the unit disc defined as

$$\mathbf{M} = \begin{cases} (0, -1)^T, & x^2 + y^2 \leq 1 \\ (0, 0)^T, & \text{otherwise,} \end{cases} \quad (1)$$

with no charges or currents. In this case Maxwells equations read

$$\nabla \cdot \mathbf{B} = 0, \quad \nabla \times \mathbf{H} = 0 \quad (2)$$

The magnetic field \mathbf{B} is given by

$$\mathbf{B} = \mu_0(\mathbf{M} + \mathbf{H}) \quad (3)$$

and the auxiliary field \mathbf{H} can be written as the negative gradient of a scalar field, due to the condition of no rotation: $\mathbf{H} = -\nabla\phi$. Putting all this into the equation for the divergence of the magnetic field gives

$$\nabla \cdot \mathbf{B} = \mu_0(\nabla \cdot \mathbf{M} - \nabla^2\phi) = 0 \quad (4)$$

giving us the governing equation for this system:

$$\nabla^2\phi = \nabla \cdot \mathbf{M}. \quad (5)$$

Further, we impose the boundary conditions of

$$\phi(0, 1) = 0, \quad \nabla\phi \cdot \mathbf{n} = 0 \quad \forall \mathbf{x} \in \Gamma \quad (6)$$

where \mathbf{n} is the outward unit normal of the domain surface S , and $\Gamma = \partial S$ is the boundary of the domain.

The approach of the finite volume method is then to integrate the governing equation over the domain and split this integral into control volumes and manipulate the equation a bit:

$$\int_S \nabla \cdot (\nabla\phi - \mathbf{M}) \, dS = \sum_{v \in S} \int_{S_v} \nabla \cdot (\nabla\phi - \mathbf{M}) \, dS \quad (7)$$

$$= \sum_{v \in S} \int_{\Gamma_v} (\nabla\phi - \mathbf{M}) \cdot \mathbf{n} \, d\Gamma \quad (8)$$

$$= 0. \quad (9)$$

Where S_v denotes a control volume. In the second equality we make use of the Gauss divergence theorem to convert a surface integral into a line integral. Now, the governing equation must hold all over the domain, and as such, so must the integral equation over each

control volume. This means that we obtain an equation for each control volume:

$$\int_{\Gamma_v} (\nabla\phi - \mathbf{M}) \cdot \mathbf{n} \, d\Gamma = 0. \quad (10)$$

Next we further split the line integral over the control volumes boundary into smaller integrals - one for each edge on the boundary:

$$\int_{\Gamma_v} (\nabla\phi - \mathbf{M}) \cdot \mathbf{n} \, d\Gamma = \sum_{e \in \Gamma_v} \int_{\Gamma_e} (\nabla\phi - \mathbf{M}) \cdot \mathbf{n} \, d\Gamma = 0 \quad (11)$$

We now introduce the integral average of a quantity:

$$\overline{f(x)} = \frac{1}{b-a} \int_a^b f(x) \, dx \quad (12)$$

and rewrite the integrals as

$$\sum_{e \in \Gamma_v} \int_{\Gamma_e} (\nabla\phi - \mathbf{M}) \cdot \mathbf{n} \, d\Gamma = \sum_{e \in \Gamma_v} (\overline{\nabla\phi} - \overline{\mathbf{M}}) \ell_e = 0 \quad (13)$$

where ℓ_e is the length of the e 'th edge. Assuming each edge is short, we approximate the integral using the midpoint rule:

$$\int_a^b f(x) \, dx \approx (b-a) f\left(\frac{b+a}{2}\right) \quad (14)$$

With this approximation, the integral average becomes just the value of the integrand at the midpoint of the edge. This makes sense, since we are essentially saying that the integrand is constant over the integration range, and so the average is precisely this constant. All we need now, is a way to calculate the midpoint value, and then we are set to solve the problem!

1.1 Control volumes

Now, we have used the words control volume and made reference to its edges along the boundary. In general a control volume is any fixed, finite volume in the domain. We of course need the union of all control volumes to be exactly the whole domain, otherwise the solution obtained will not be a solution to the problem posed.

One could use a mesh of triangles, quadrilaterals, or any other shape, for that matter. The choice depends on where one stores the values of the unknowns as well as ease of computability. In our case we use a triangular mesh, where we store the values of ϕ on the vertices, and use a vertex centred dual control volume.

This means that for each vertex in our mesh, we compute the incenters of all triangles the vertex is a part of, and join these together. If the vertex is on the boundary of the domain, we join the incenter of the two boundary triangles to the nearest point on the boundary. See figure 1 for an example.

1.2 Approximations

This choice of control volumes gives us an easy way of calculating the directional derivative $\nabla\phi \cdot \mathbf{n}$, under the assumption that the

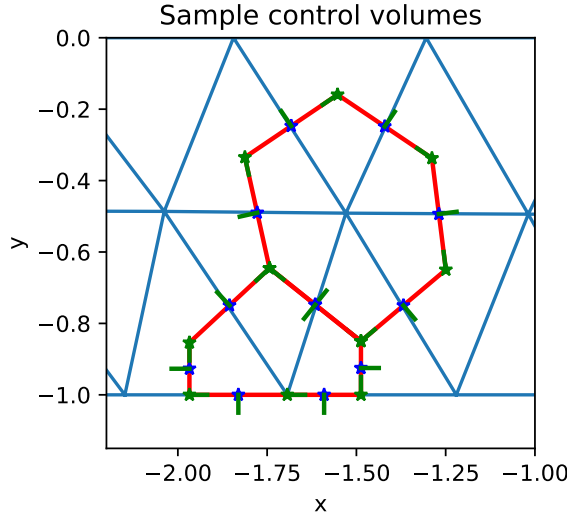


Figure 1: Sample vertex centred dual control volumes for the computational mesh. Triangle mesh is blue, control volumes are red. green points denote vertices of the control volumes. The centre of the upper control volume is an inner node of the triangle mesh, and the control volume is therefore a hexagon. The centre of the lower control volume is a boundary node of the triangle mesh, and therefore has boundary edges.

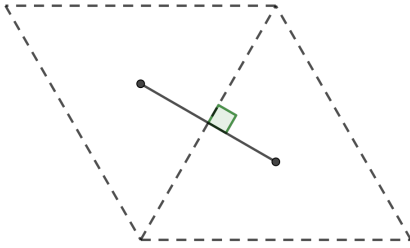


Figure 2: Incentres and surface normals for equilateral triangles. The line between incentres is perpendicular to the line joining the two vertices shared by the two triangles.

triangles are close to being equilateral. If the triangles are equilateral, then the line joining the incentres of two adjacent triangles will also be the perpendicular bisector between the two vertices shared by the triangles (see figure 2). This then means that the directional derivative from the i 'th vertex to the j 'th vertex can be approximated easily using a finite difference:

$$\nabla \phi \cdot \mathbf{n} \approx \frac{\phi_j - \phi_i}{|\mathbf{x}_i - \mathbf{x}_j|} \quad (15)$$

Of course as the triangles become “less” equilateral, this approximation will be more erroneous. This is evident in figure 1 where the outward surface normals between the two control volume do not

line up with the straight line segment between two control volume centres.

Further, when considering edges which end on the boundary of the domain we have no easy way of calculating the directional derivative at the midpoint of the edge. We therefore use the zeroth order approximation that the value at the midpoint is the same as that at the end point - precisely the same equation as above.

If the edge of the control volume is along the boundary of the domain, we need to apply the boundary condition $\nabla \phi \cdot \mathbf{n} = 0$. Putting the boundary condition into the integral gives us the solution: we just should not compute the integral for domain boundary. The integral on both sides are zero (the left hand side is zero since there is only a magnetisation on a single point in both the top and bottom domain boundary, which will not influence the integral).

Putting all these considerations together we get the equation for the i 'th control volume:

$$\sum_{e \in \Gamma_v} \frac{\phi_{j(e)} - \phi_i}{|\mathbf{x}_i - \mathbf{x}_{j(e)}|} \ell_e = \sum_{e \in \Gamma_v} \mathbf{M}_e \cdot \mathbf{n}_e. \quad (16)$$

where $j(e)$ is the neighbouring vertex corresponding to the edge segment e , \mathbf{M}_e is the value of \mathbf{M} at the midpoint of the edge and \mathbf{n}_e is the outward surface normal of the e 'th edge segment.

Now, \mathbf{M} has a discontinuity around the unit disc. If an edge intersects this (ie, one end is inside, whilst the other is outside), then using the midpoint value of \mathbf{M} no longer gives an accurate integral. Instead we split the integral into two:

$$\int_{\mathbf{x}_i}^{\mathbf{x}_j} \mathbf{M} \cdot \mathbf{n} \, d\Gamma = \int_{\mathbf{x}_i}^{\mathbf{p}} \mathbf{M} \cdot \mathbf{n} \, d\Gamma + \int_{\mathbf{p}}^{\mathbf{x}_j} \mathbf{M} \cdot \mathbf{n} \, d\Gamma \quad (17)$$

where one of these is 0. The intersection \mathbf{p} is calculated with

$$p_x = \frac{Dd_y \pm \text{sign}(d_y)d_x \sqrt{r^2 d_r^2 - D^2}}{d_r^2}, \quad p_y = \frac{-Dd_x \pm |d_y| \sqrt{r^2 d_r^2 - D^2}}{d_r^2} \quad (18)$$

with $d_y = y_i - y_j$, $d_x = x_i - x_j$, $d_r = \sqrt{d_x^2 + d_y^2} = \ell_e$, $D = x_i y_j - x_j y_i$, as per Wolfram Alpha. We only need the one intersection, defined by the equation

$$p_x = x_i + s(x_j - x_i) \quad s = \frac{p_x - x_i}{x_j - x_i} \quad (19)$$

if $0 \leq s \leq 1$, then we have the right intersection, as $x_i \leq p_x \leq x_j$, which implies the same for p_y .

The \mathbf{M} -integral for that side is then

$$\int_{\mathbf{p}}^{\mathbf{x}_j} \mathbf{M} \cdot \mathbf{n} \, d\Gamma = (\mathbf{M}_{\text{in}} \cdot \mathbf{n}) |\mathbf{p} - \mathbf{x}_j| \quad (20)$$

where we assume \mathbf{x}_j is inside the unit disc and \mathbf{x}_i is outside. The left hand (ϕ) integral remains the same, of course, as ϕ is assumed continuous.

Another way to handle this would be to alter the mesh, such that no control volume edges cross the unit disc. However, when using the vertex centered dual control volume, constructing the mesh such that this happens is much more of a challenge than simply putting in an if statement in the code and call it a day, even if the equations for the circle intersection are slightly cumbersome.

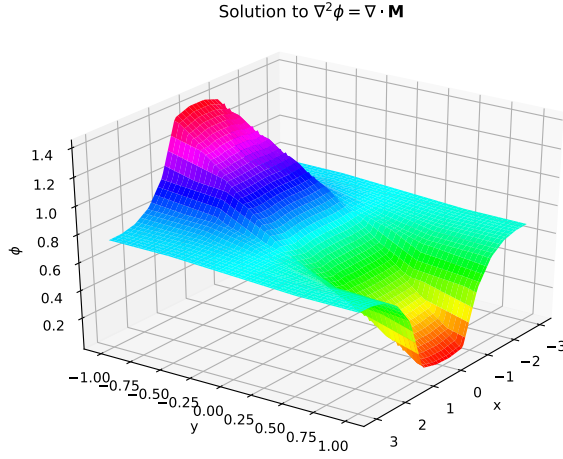


Figure 3: Solution to the magnetostatics problem with a unit disc of magnetised material. Indeed we see a scalar potential in the vicinity of the unit disc.

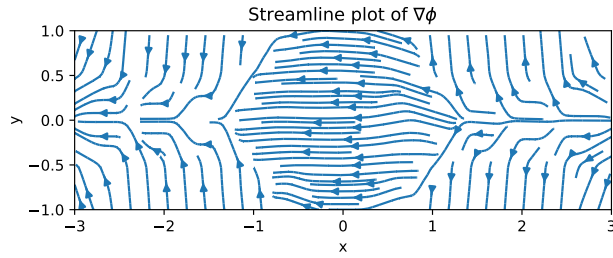


Figure 4: Streamline plot of the gradient of ϕ . Note the slightly hexagonal look of the unit disc.

1.3 Matrix assembly

With the theory in place, it is time to assemble the system of equations. We have N vertices in the system and therefore N control volumes, each with one linear equation relating the N 'th vertex to its neighbours. We therefore start with a $N \times N$ matrix, A , of zeros. In the i 'th row we add the value $\ell_e / |\mathbf{x}_i - \mathbf{x}_{j(e)}|$ in all the columns corresponding to neighbours, and minus the sum of all the others values in the i 'th diagonal element.

For the source term \mathbf{b} we start with a N -vector of zeros as above, and add the sum of \mathbf{M} -integrals above, taking care to split any integrals crossing the discontinuity of \mathbf{M} .

With the matrix and source term we can then easily compute the solution ϕ to the problem using a direct solver (assuming, of course, we have a non-singular matrix, which the boundary conditions should take care of).

This is all highly reminiscent of the matrix assembly for both FEM and FDM.

Residual on x-parity flip. Total: 8.390e-05

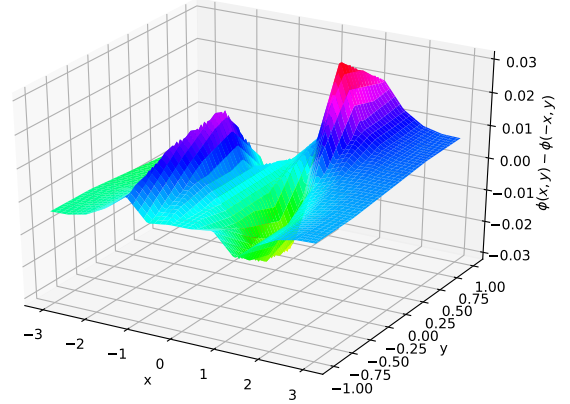


Figure 5: Residual upon a parity transformation in x .

2 EXPERIMENTS

First of all, let us plot the solution to the problem obtained using the method outlined above, on the sample mesh for this week. The result is seen in figures 3 and 4. As we see, we have a region around the unit disk with a non-zero scalar potential, while it is approximately zero “far” away from the disk. This precisely the expected behaviour from magnetostatics and electrostatics. Further more, on the streamplot of the gradient of ϕ (the electrical field) we see that the gradient is perpendicular to the magnetisation, as one usually expects in the case for induced fields. Overall, the solution makes physical sense.

A thing to note, which is especially visible in the streamline plot: the unit “disc”, which should be visible looks more like a unit “hexagon”, which is most likely an artefact of the somewhat low resolution mesh used. Upping the resolution should remedy this.

Since the governing equation has reflective or parity symmetry we should expect the solution to be the same for a flipped x -coordinate: $\phi(x, y) = \phi(-x, y)$. Similarly, if we flip the sign of \mathbf{M} and perform a parity transform in y , we should also expect the same solution (essentially charge conjugation / parity (CP) symmetry), so long as we also change the boundary condition of $\phi(0, 1) = 0$ to $\phi(0, -1) = 0$, as this fixes the offset between the scalar potentials (and as we know, we can always add and subtract constants to our potentials - it is always the gradient that matters!). To easily calculate the residual $\phi(x) - \phi(-x)$ we first interpolate ϕ to a grid, making sure that when we reflect x each point lands on top of its “parity partner”, then we subtract ϕ from ϕ , where we just reverse the direction of the columns (remember, that in Python, by default x is the second index). We then square each difference, sum up and divide by the total number of points to get the final residual. The result is seen in figure [ref](#). If we sum ϕ over the domain and divide by the number of points we get 0.7136, which means the residual is around 4 orders of magnitude smaller than the “integral”. There is also a small yet noticeable asymmetry in the x -direction of the residual. With a residual of this size we still conclude, however, that there is at least approximate parity symmetry in x for the solution.

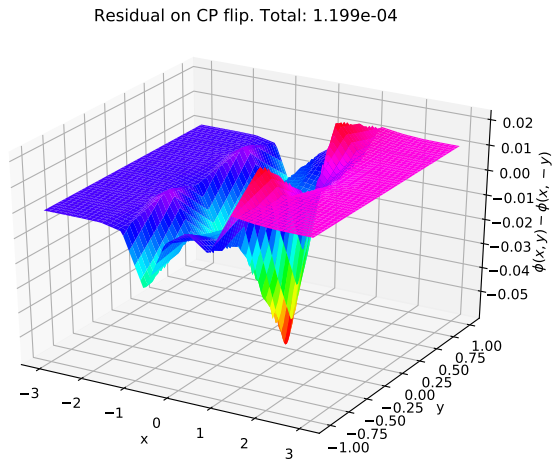


Figure 6: Residual upon a CP-like transformation. Note the distinct asymmetry in the residual.

For the y -residual we flip the sign on \mathbf{M} when creating the source term vector, and change the boundary condition as written above. Then we calculate the new solution and compare the first to the second, with the first argument reversed, corresponding to $\phi(x, y) - \phi(x, -y)$. The result is seen in figure [ref](#). Again, there is a small residual, though larger than the one for x -parity. It is still roughly 3 orders of magnitude smaller than the “integral”, but we also see a distinct asymmetry in the x -direction like for the x -parity, yet larger. The result is not quite as clear cut as for the x -parity, but it is still definitely in the “right” direction. Maybe a higher resolution grid would produce a better result.