

## Background

- We have developed a **perpendicular Poisson solver** which can be used to simulate a thin shell ionosphere. Full model will include conductivity tensor based off of parallel, Pederson, and Hall conductivities<sup>2</sup>.
- Goal is to couple ionosphere model to multi-fluid solvers in Gkeyll to perform global magnetosphere simulations. Previous Gkeyll simulations have been on smaller bodies without ionospheres<sup>3</sup>.
- Generalized coordinate** approach allows for complex logical to physical mappings to avoid polar singularity. Finite volume method is used for its conservative characteristics.
- Global magnetospheric simulations are essential to understanding the mechanisms behind formation of geomagnetic storms following large CME's which are incident on the earth.

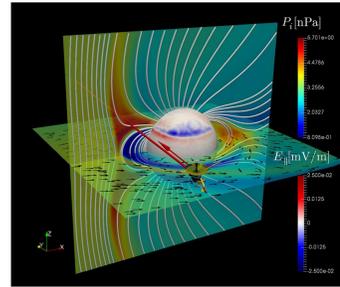


Fig. 1: Results from two fluid modelling of Ganymede's magnetosphere, performed in Gkeyll by Wang et al<sup>3</sup>.

## Numerical Methods

- Beginning with the standard Poisson's equation, we integrate over the area of a cell with potential  $\phi$ , cell normal  $\hat{n}$ , and source  $s$ :

$$\oint (\nabla\phi \cdot \hat{n}) dl = \int s dS$$

- We then assume the source is input as an average and estimate the average value of the gradients at the cell walls. Note the maps used here have quadrilateral cells.

$$\sum_{k=1}^4 (\nabla\phi \cdot \hat{n}) l_k = s \Delta x \Delta y$$

- Where  $l_k$  is the  $k^{\text{th}}$  cell wall. Correctly calculating the lengths involved in taking a derivative requires use of the metric tensor, the Voss-Weyl formula can be used to represent the gradient. The Einstein summation convention is used.

$$\nabla_i \phi^i = \frac{1}{\sqrt{g}} (\partial_i \sqrt{g} \phi^i)$$

- Here  $g$  represents the metric determinant. This is then placed within the sum form of the integrated Poisson equation. The gradient is covariant as with the cell wall normal, so the dot product requires raising the indices of the gradient with the inverse metric.

$$\sum_{k=1}^4 \sqrt{g_{(k)}} (g_{(k)}^{ij} \partial_i \phi_{(k)}^j n_{j(k)}) l_k = \sqrt{g} s \Delta x_c \Delta y_c$$

- $g^{ij}$  refers to the inverse metric, and  $\Delta x_c$  and  $\Delta y_c$  refer to the computational grid steps. The  $g_{(k)}$  are evaluated at the cell walls, however on the source side it is evaluated at the cell center. The finite differencing used to approximate the gradient at the cell edge is a **central difference nine point stencil**, which calculates the wall parallel derivative by averaging the four surrounding cells at each vertex of the wall. The term contributed by the right side wall ( $x_c = x_{c,\text{center}} + 0.5\Delta x_c$  and  $y_c = y_{c,\text{center}}$ ) is<sup>1</sup>:

$$\Delta y_c \sqrt{g_{i+1/2,j}} \left( g^{11} \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x_c} + g^{12} \frac{\phi_{i,j+1} + \phi_{i+1,j+1} - \phi_{i,j-1} - \phi_{i+1,j-1}}{4\Delta y_c} \right)$$

- Certain problems require special treatment: When all boundary conditions are periodic, each cell feeds into another, so the integral of the Laplacian of  $\phi$  over the whole domain must be zero. Therefore, when treating the source its integral over the domain is made to be zero.
- Secondly, when there are no Dirichlet conditions present, the solution is not unique due to an unknown additive constant. This is handled by setting the upper right corner of the computational grid to 0 for all such situations.
- The code is written in a combination of Lua, C, and C++, with the Eigen library being used to factorize the stiffness matrix/ solve for the potential. **Metric derivative terms are calculated using the SciLua automatic differentiation library.**

## Grid Generation

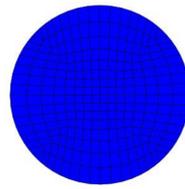


Fig. 2: Diagram of 2D circular mapping obtained from Calhoun et al.

- Circular and spherical mappings were obtained from Calhoun et al<sup>1</sup>.
- The **grids used are logically rectangular**, in the 2D case being square, and in the spherical case being a 2:1 rectangle.
- The grids shown here are 16 by 16 and 32 by 16. The change from disk to sphere simply involves shifting cells radially outward and adding a z coordinate.
- Corner cells on the spherical map become nearly but not totally triangular. However certain areas of the mapping exhibited singularities in the metric, mainly at the equator and exact poles.
- To solve this the number of cells is chosen carefully and the gradients are evaluated  $1e-5\Delta x_c$  or  $1e-5\Delta y_c$  within the cell walls.

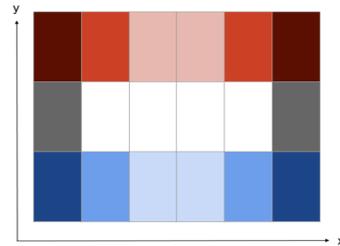


Fig. 5: Color coded illustration of periodic boundary condition communication on nonsingular sphere map.

- The exact spherical mapping method from Calhoun et al is not used in this solver. We found that elliptic solutions possess some error near the poles which lowered **convergence**. So, a modified redistribution of cells toward the edges of the disk is used prior to z coordinate assignment.
- The periodic boundary conditions require special treatment, where the **y bound edges communicate with themselves instead of opposite sides**<sup>1</sup>. Corners are handled without special treatment.

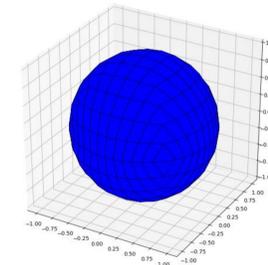


Fig. 3: 3D spherical mapping obtained from Calhoun et al.

## Numerical Testing Results

- Convergence tests were carried out on numerous skewed grids, shown here are results from testing on the disk and the sphere, **expected convergence for this method is 2<sup>nd</sup> order**.

- This test uses Dirichlet boundary conditions set to 0 at  $r = 1$ . The test source and solution are given below. The source is not averaged over the cell upon input, but rather evaluated at the cell centers for simplicity.

$$S(r, \theta) = (8r - 4.5) \cos(\theta)$$

$$V(r, \theta) = (r^3 - 1.5r^2 + 0.5r) \cos(\theta)$$

n	E	p
16	2.62e-3	
32	6.41e-4	2.03
64	1.40e-4	2.20
128	3.25e-5	2.10

Tbl. 1: Convergence study for circular disk test. E is the average absolute error and p is the convergence order.

Fig. 7: Area weighted absolute error. Most large error originates from the highly skewed grid points.

- The error displayed is weighted by the physical size of the cell, therefore it most nearly represents integrated error over the cell area.
- The special periodic boundary conditions were used for the first spherical test. Currently there is some sensitivity to strongly  $\phi$  dependent solutions, this is likely due to a preliminary implementation of the arctangent in our mapping to emulate "atan2" while it is being added to the autodifferentiation library.

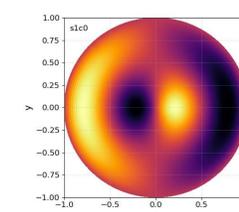
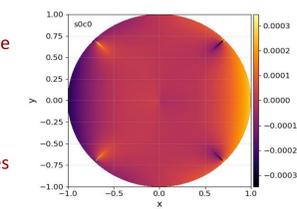


Fig. 6: Calculated solution from the given test source.



$$S(\theta, \phi) = (-11 \cos(\theta) \sin^2(\theta) - 64 \cos(\theta) \sin(\theta) + 9 \cos^3(\theta)) \sin(8\phi)$$

$$V(\theta, \phi) = \cos(\theta) \sin^3(\theta) \sin(8\phi)$$

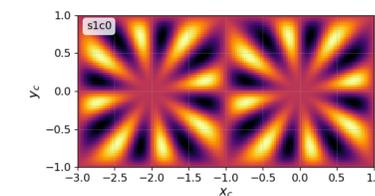


Fig. 9: Calculated solution from the given test source (computational space 128x64 grid).

$n_{xc}(=2n_{yc})$	E	p
16	0.1532	
32	0.0230	2.74
64	5.12e-3	2.17
128	2.28e-3	1.17

Tbl. 2: Convergence study for first spherical test.

- This test demonstrates the more ideal behavior expected when there is no  $\phi$  dependence. Two of these spherical test plots are displayed in computational space to illustrate the solution everywhere on the sphere.

$$S(\theta, \phi) = 4 \cos^2(\theta) - 2 \sin^2(\theta)$$

$$V(\theta, \phi) = \sin^2(\theta)$$

$n_{xc}(=2n_{yc})$	E	p
16	0.0144	
32	4.31e-3	1.74
64	1.11e-3	1.96
128	2.57e-4	2.11

Tbl. 3: Convergence study for second spherical test.

Fig. 10: Area weighted absolute error.

- Overall error is well distributed with **no large singularities** or particular problem areas which would cause significant trouble in electric field calculation.

## Summary and Future Work

- Expectations for the 2D and 3D solvers have been reached with 2<sup>nd</sup> order expected convergence being met on average.** The convergence issue present in the first spherical test case is expected to resolve with correction of the "atan2" substitution. **Further work to make this an ionospheric model requires the conductivity tensor.**
- The governing equation of the thin shell model assumes that current flows along field lines to the ionosphere where it spreads out across the surface, according to  $\nabla \cdot J_{\perp} = -\nabla \cdot J_{\parallel}$ . Using Ohm's law the equation dictating ionospheric potential is then:  $\nabla \cdot (\sigma \nabla \Phi) = -\nabla \cdot J_{\parallel}$  and similar conversions to earlier must be made to format the conductivity for representation in the computational space.
- Thin shell model uses 3 conductivities: the parallel, Pederson, and Hall conductivities. They describe the directions parallel to the magnetic field, perpendicular to the magnetic field and parallel to the electric field, and perpendicular to both fields<sup>2</sup>. Parallel conductivity is plasma conductivity, Pederson conductivity is based off mobility  $\mu$  across a magnetic field, and Hall conductivity is heavily dictated by  $E \times B$  drift. Combined with the geomagnetic dip angle they may represent a 2D conductivity tensor<sup>2</sup>.

## References and Acknowledgements

This work was supported by the DOE Summer Undergraduate Laboratory Internship.

- D. Calhoun, C. Helzel, R. LeVeque, Logically Rectangular Grids and Finite Volume Methods for PDEs in Circular and Spherical Domains, *SIAM Review* 50, 723-752, (2008).
- Maeda, K., Conductivity and drift in the ionosphere, *J. Atmos. Terr. Phys.*, Vol.39, 1041-1053, (1977).
- Wang, L., Germaschewski, K., Hakim, A., Dong, C., Raeder, J., Bhattacharjee, A. Electron physics in 3-D two-fluid 10-moment modeling of Ganymede's magnetosphere. *Journal of Geophysical Research: Space Physics*, 123, 2815-2830, (2018).