# Mantle Dismantlers
# Scientific Visualization Project Report

Gavin Gray, Raphael Winkler, Rasmus Lüscher, Tal Rastopchin

June 12, 2023

## Introduction

Within the Earth's mantle, convection is driven by cooling forces from the planet's surface and heating forces from the core. The mantle circulation is of interest to researchers and can be visualized with large timescales that expose the geologic phenomena. The data set provided was collected by the Pysklywec Lab (Russell Pysklywec and Hosein Shahnas) at the University of Toronto [3, 4, 2] (and provided by the SciVis Contest 2021), it contains simulated data of the Earth's mantle over 500 million years. The data contains a velocity vector field and provides multiple scalar fields (e.g., temperature, thermal conductivity anomaly, thermal expansivity anomaly, temperature anomaly, and spin transition-induced density anomaly). Utilizing this information we implemented a tool based on VTK[1] (The Visualization Toolkit) library that uses several visualization techniques to dissect the data and highlight several geologic phenomena that occur within the Earth's mantle.

The "teaser video" used in our final presentation can be found at: `https://drive.google.com/file/d/1dOhv3K51hmPt6Hq-1Sd2QYLGKz2VjK5y/view` and the final code repository can be found on GitHub at: `https://github.com/gavinleroy/earth-mantle`.

## Methods Overview

We experimented with all of our visualization methods in ParaView in order to figure out which methods we wanted to use and how to use each method.

***Data Pipeline*** We wanted to visualize three kinds of structures: scalar and vector fields living on two-dimensional subsets of the input data; derived structures such as isosurfaces and streamlines; and scalar fields as direct volumes.

At the project onset, it was unclear how to plumb the data through the various pipelines to accommodate the separate visualizations. Most importantly, different VTK algorithms require different types of input data. Our team started with direct volume rendering. VTK volume rendering algorithms all require structured grid inputs, thus we resampled the input unstructured grid onto a lower resolution structured grid. This worked perfectly for visualizing the temperature anomaly field with direct volume rendering, however, we did not have much control in selecting the subset of the data we wanted to visualize. The best we could do was to add a slice plane to the mapper, where we could choose the plane origin and normal.

It turns out that other than adding a slice plane to the volume mapper, structured grids in VTK can only be sliced with continuous sub-ranges of indices; allowing one to select a rectangular-prism region of interest. One can use implicit functions to slice unstructured grids, giving much more fine-grained control over selecting regions of interest. However, we found that the overhead of resampling was expensive, and it was difficult to change the implicit functions at runtime. We settled on using the slice plane to select the region of interest for our direct volume rendering. Resampling the input data to a structured grid did allow us to use direct volume rendering. It also decreased the resolution of the data so that we could reasonably derive isosurfaces and streamlines.

We went through a similar process with visualizing the scalar fields, however, we were able to find a solution that gave us fine-grained control over the slicing to select two-dimensional subsets of the input data. At the start of the project, we used the slice plane on the poly data mapper in order to cut the spherical shell in half. This approach has the same lack of specificity as with the volume rendering. At the same time, we noticed that while the boundary of the spherical shell was recovered, there were often holes on the inside of the slice. We learned that this is due to VTK making assumptions about what two-dimensional subset of the input data to extract from the input data. Because VTK was making assumptions about what two-dimensional subset to use, we had too little control over the geometry and needed to come up with another way of slicing the data. We created a proxy geometry using the various VTK sources. This allowed us to create

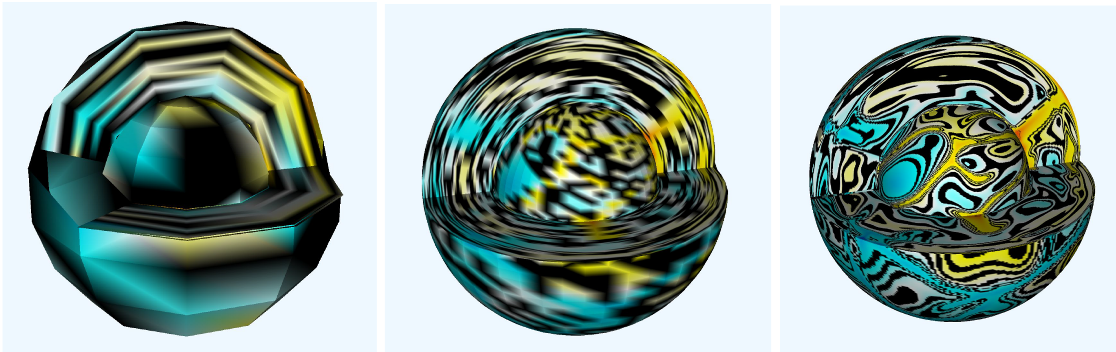

Figure 1: The provided data is interpolated from the unstructured grid input onto the proxy geometry. Shown are three different resolutions of sampling that affect the final rendered outcome, the precision of contour lines dramatize the resolution's effects.

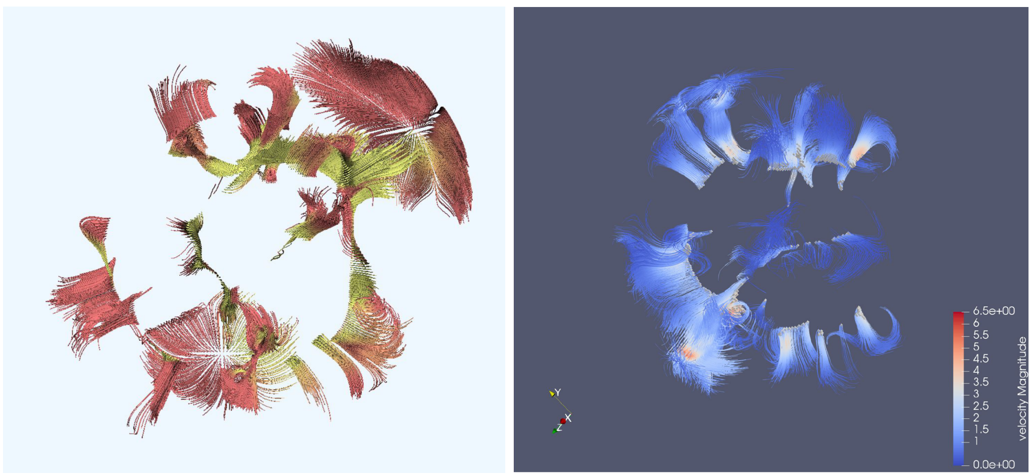


Figure 2: Hot plumes visualized as streamlines seeded from the high temperature anomaly regions of 500°K to 1100°K. **Left:** Streamlines visualized with our VTK implementation VTK **Right:** The same streamlines visualized in ParaView.

a geometric mesh representing the earth mantle with a wedge. We then used a point interpolator with a Gaussian kernel to interpolate the data from the original input unstructured grid onto the proxy geometry. This allowed us to decouple the two-dimensional subset of the input data from the unstructured grid input data, giving us finer grain control over the resolution of the displayed region of interest. Namely, higher resolutions for the proxy geometry lead to visually smoother and better looking scalar fields as shown in Figure 1. We ended up using the proxy geometry to visualize the temperature anomaly field and the vector field using line integral convolution.

***Direct Volume Rendering*** To visualize the temperature anomaly field using direct volume rendering we used the provided `vtkSmartVolumeMapper`, enabling an overview of the mantle's structure and spatial distribution of temperature anomalies. Like we mentioned in the Data Pipeline subsection, the most crucial part of getting the direct volume rendering working was understanding how input data needs to be formatted for the VTK volume rendering algorithms. We played around with a couple different data representations and learned that VTK's volume rendering algorithms all require structured grid inputs. We were able to convert the unstructured grid input into a structured grid using the `vtkResampleToImage` algorithm.

***Isosurface Rendering*** One main goal for the provided tasks involves visualizing the *cold slabs* and *hot plumes* of mantle material. We achieved this via isosurface rendering using `vtkMarchingCubes` and `vtkSmoothPolyDataFilter`. The extracted and filtered isosurfaces provide us with insights into the structure and distribution of these temperature anomalies within the mantle.

We initially tried to visualize the cold and warm slabs using direct volume rendering, but we found that the surface derived with the `vtkMarchingCubes` algorithm is visually clearer. Furthermore, using the `vtkSmoothPolyDataFilter` gave us smoother-looking results than the raw isosurfaces visualized with direct volume rendering.

***Tracing Streamlines*** Using `vtkStreamTracer` and `vtkTubeFilter`, we traced streamlines originating from regions with high temperature anomalies to visualize rising plumes in the mantle. This visualization allows us to study the flow patterns associated with convective processes in the mantle.

It took a bit of trial and error to both seed the streamlines and choose the appropriate algorithm parameters. We are happy with the end result, however, one issue that we ran into was that the length of the traced streamline is not a good termination criterion. Experimenting with different maximum streamline lengths, we found that the visualization becomes convoluted because streamlines get caught in convection cells and loop around multiple times. Our solution was to set the maximum length to be relatively short, which prevents the streamlines from looping in the convection cells, but also prematurely prevents streamlines from making it from the core all the way to the surface.

***Line Integral Convolution*** We implemented LIC using the `vtkSurfaceLICMapper` to visualize the velocity vector field with the overlayed temperature anomaly scalar field. We originally wanted to implement the LIC algorithm ourselves. But, we quickly realized that it would be hard to implement and integrate the algorithm into VTK because LIC requires computing the streamlines and performing convolutions on the surface, which would necessitate a parameterization of the proxy geometry. All of this is doable in theory, but as we were still learning how to use VTK, it was important to prioritize getting the LIC working before we experimented with implementing it from scratch.

Using the `vtkSurfaceLICMapper` was the most straightforward way for us to implement LIC, as all we had to do was create and pass to the algorithm a velocity array; VTK's implementation took care of the rest. Originally we were considering using `vtkLineIntegralConvolution2D`, but it has the same issue of us having to parameterize the proxy geometry, as it only works on 2D images (and it turns out it is used as a subroutine in `vtkSurfaceLICMapper`).

# Goals

At the onset of the course project, we wrote down several goals for what we had hoped to achieve with an interactive visualization tool for the Earth's mantle. The core of these goals revolved around the tasks provided by the SciVis contest which will be further discussed in Sec. *Explaining the Results.* Concretely, we wanted to implement direct volume rendering for the hot/cold mantel material, allow users to interact with the data with filters and brushes, rendering scalar fields directly on a geometry object, extracting topological data, and writing a custom shader for expansion tubes. The previous summary of implemented methods hinted at some of these goals which were not achieved (e.g., custom shaders) but the implementation was sufficient to show *what* we wanted to achieve.

# Explaining the Results

The original SciVis 2021 contest posed several questions to its contestants with the hopes of demonstrating several phenomena. The visualization should aid researchers in dissecting and observing interactions between the different scalar forces of the mantle.

***Tasks 1–2*** In the first task we look at slabs in the upper mantle, at around a depth of 660km. As seen in Figure 3 the cold slabs are seen at shallow depths in the earth's core. Combined with line integral convolution, we can see the convection cells that form around the falling slabs. The stagnated slabs hang around the surface of the earth and don't show any columns of cold material falling to the center. Unfortunately, one feature of the visualization that we failed to realize was depth markings on our earth geometry. Lacking this feature, it is hard to precisely point out the depth of 660km.

For the second task we were to show cold slabs at the mid-mantle depth, around 1600km. With our given images it is rather difficult to pinpoint slabs below the 1600km depth line, though they are observable in the contour plot. These slabs tend to interact differently in the mantle convection as they are diverted; again, the line integral convolution visualization can help to show the direction and resulting convection cells.

***Tasks 3–4*** The third task conversely aims to visualize the hot plumes within the mantle. We show again how the isosurface rendering of the temperature anomaly field can be used to highlight these plumes, as seen in Figure 4. Here, the plumes are rising to the surface from the lower and mid-mantle regions. For the plumes we additionally use stream tubes which show a fanning motion as some plumes are diverted. This sort of information cannot be provided by pure line integral convolution, and it would also serve our tool well to use it for the cold slabs. One interesting direction to explore would be the interaction the *spin anomaly* field has on the diverted plumes. It could be that this

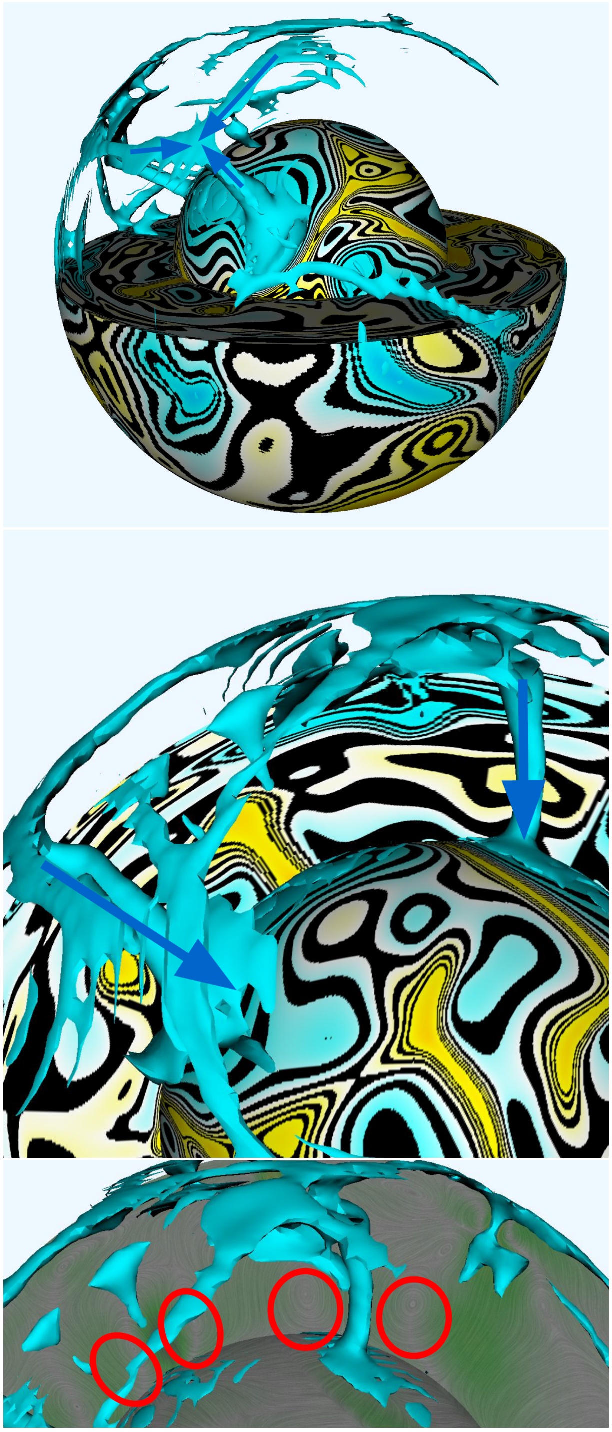

Figure 3: Cold slabs in the earth's mantle. **Above:** Cold slabs are rendered as isosurfaces of the temperature anomaly scalar field. A threshold filter is used to capture material $\leq -200$°K, a value chosen by inspecting different snapshots of the data. **Middle:** Columns of cold material spanning across the upper and mid-mantle regions. These columns demonstrate falling material that was not diverted. **Below:** correlation between the slabs and velocity demonstrate convection cells, demonstrated by the line integral convolution.

anomaly affects the fanning and direction of the diversion, it may also play a role of when plumes are diverted. This would have been an interesting interaction to explore had we more time for the project.

***Task 5*** In the final task contest participants were asked to describe the correlation between different variables and the flow patterns. As seen in Figure 3 and described in Sec. *Tasks 1—2*, we used the correlation between the cold material and the velocity to find convection cells produced by the falling material. Similar correlations are draw between the hot plumes and stream tubes to further show the direction and velocity of the diverted plumes.

## Team Contributions

***Gavin Gray*** Implemented the trace streamline visualization and the later version of isosurface rendering. He also created the first direct-volume rendering of the geometry that was superseded by a later implementation. Additionally, he worked on the overall code organization and dealing with VTK.

***Raphael Winkler*** Aided Gavin in earlier version of the direct-volume rendering, as well as debugging streamlines as well as the first few experiments of isosurface rendering. He later implemented the direct-volume rendering used in the current tool and experimented with using different data representations.

***Rasmus Lüscher*** Toggled visualizations in the scene using captured keystrokes and included the scalar bars showing the current color transfer function.

***Tal Rastopchin*** Implemented the initial visualization using line integral convolution, and mastered the final use of geometry and clipping planes in VTK. He additionally contributed to the configuration of various visualization parameters and isosurface rendering.

## Discussion

***Limitations and Future Work*** The tool developed is limited in several ways, many previously mentioned in the body text. They are repeated here for clarity. Interactivity of the tool suffers greatly, both in performance and in interactivity. Primarily, the performance observed is a result of the data representation used within VTK. Spherical coordinates do not lend themselves towards performant computation, we believe a better approach would be to use data held in a rectilinear grid using the $(r, \theta, \varphi)$ of spherical coordinates as the index into a rectilinear grid. Attempts

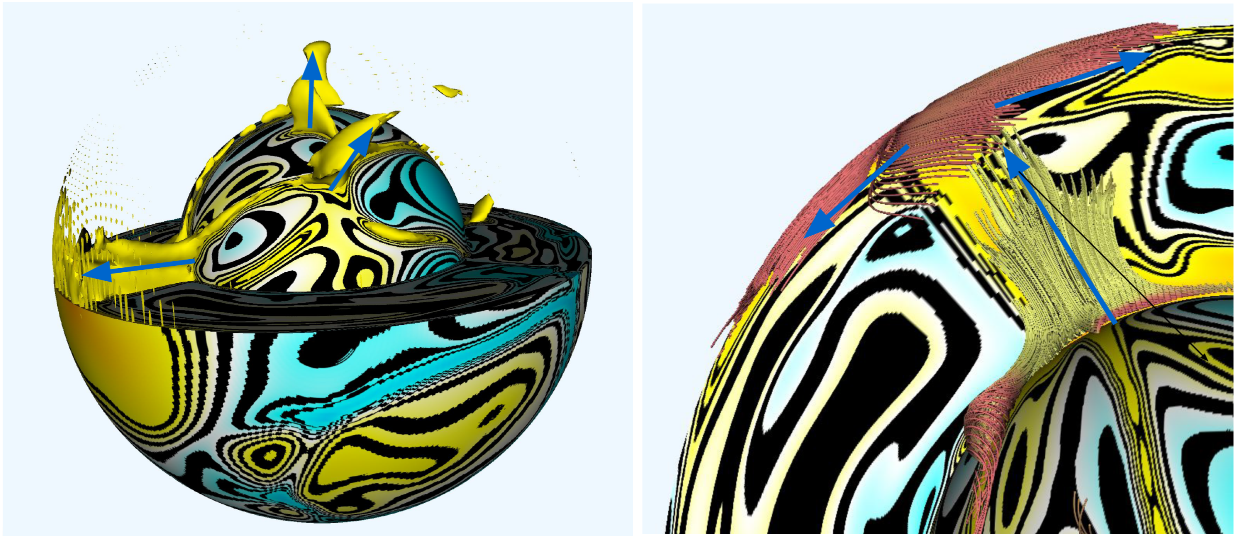

Figure 4: Hot plumes in the earths mantle. **Left:** Hot plumes rendered as an isosurface using a threshold of temperature anomaly $\geq 200$°K. **Right:** Streamlines with a tube filter are used to show the diversion of the plumes in the upper mantle; tubes enable visualizing diversion characteristics, such as fanning, that are not visible with line integral convolution.

were made to use this format, but ultimately some of the visualizations simply were not working properly and the lack of knowledge and support of VTK made us turn to the other data representations used. Furthermore, advanced rendering techniques using a fragment shader would be much faster—they would also provide better results for our contour lines visualization—but time was not left to attempt this.

While performance was a key factor in the interactivity of our tool, we also lacked to provide utilities that a user may find useful when exploring the data set. The major feature missing is interaction in the earth geometry itself. Ideally, the user would be able to adjust the clipping planes, as well as the radius of the inner sphere. The infrastructure we built does allow for dynamic modification of the clipping planes, however, updating the visualization was too slow and we decided to omit this interaction at the user-level. Other honorable mentions in the category of interactivity is color transfer function editing and data markings on the earth geometry such as latitude/longitude and depth lines.

Lastly, playing time steps of the data would be crucial for observing advanced interactions between the variables. Time steps would allow users to observe material diversions and growth of convection cells. While the infrastructure for animation was built, again the limiting factor here was performance and the feature was not exposed.

Overall, we feel that the semester project provided an interesting opportunity to work with real-world data. While not all original goals were met, the tool successfully provides a handful of visualizations that can be used in isolation or combined to view the interactions in the earths mantle convection. The given tasks from the SciVis 2021 contest could be addressed and visualized with relative ease. Though not always the most straightforward experience, using VTK gave us the opportunity to see how industrial-grade tools can be utilized, and even improved upon in the future. We would like to thank the course staff for their effort in the exercises and lectures and wish them all the best til we meet again in the exam.

# References

[1] Kitware. vtk. https://vtk.org/doc/nightly/html/. Accessed: September 27, 2023.

[2] M. Shahnas, D. Yuen, and R. Pysklywec. Mid-mantle heterogeneities and iron spin transition in the lower mantle: Implications for mid-mantle slab stagnation. *Earth and Planetary Science Letters*, 458:293–304, 2017.

[3] M. H. Shahnas, W. R. Peltier, Z. Wu, and R. Wentzcovitch. The high-pressure electronic spin transition in iron: Potential impacts upon mantle mixing. *Journal of Geophysical Research: Solid Earth*, 116(B8), 2011.

[4] M. H. Shahnas, R. N. Pysklywec, and D. A. Yuen. Spawning superplumes from the midmantle: The impact of spin transitions in the mantle. *Geochem-*

*istry, Geophysics, Geosystems*, 17(10):4051–4063, 2016.