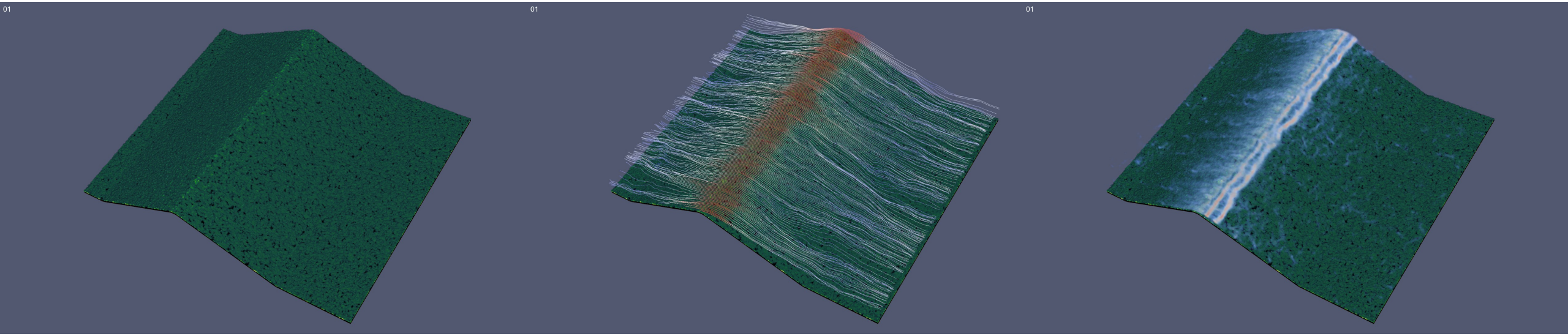
Wild Fire (B)
Final Presentation

Stuart Heeb
Johannes Kurz
Yitong Xia
Yingyan Xu

Teaser



Fire

Streamline

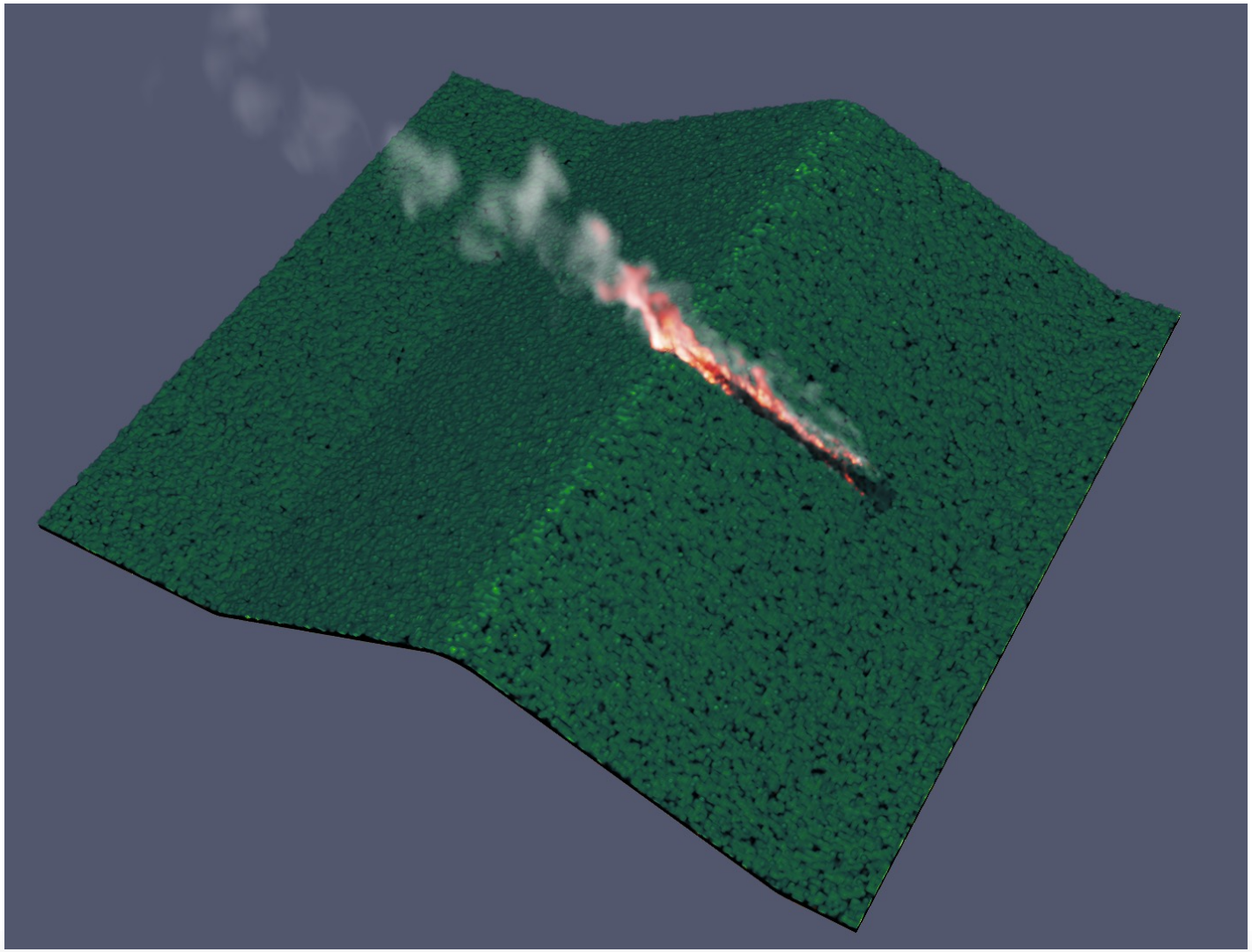
Vorticity

Project Overview

- Recap on what we want to achieve?
  - Visualize temperatures under burning process.
  - Visualize bulk density (vegetation density) under burning process.
  - Visualize wind velocity, vorticity and divergence.
  - Analyzing fire spreading behavior.

- Feature List
  - [5 pts] **Direct volume rendering**: fire, water vapor, grass.
  - [5 pts] **Isosurface**: fire-air boundary.
  - [5 pts] **Compute vorticity and divergence**.
  - [5 pts] **Animations**: on complete SciVis22 Wildfire datasets.
  - [5 pts] **Trace integral lines**: streamlines of wind velocity.
  - [Bonus] **Line Integral Convolution**: for VLS analysis.
  - [5 pts] **Transfer function editing** in GUI & in codes.
  - [Bonus] **Optimized GUI** with fluent interaction.

Direct Volume Rendering



Color Map Editor

Search ... (use Esc to clear text)

Array Name: theta

Rescale On Visibility Change

Mapping Data

Select a color map from default presets

Data:

Enable Freehand Drawing Of Opacity Transfer Function

Use Log Scale When Mapping Data To Colors

Enable Opacity Mapping For Surfaces

Use Log Scale When Mapping Data To Opacity

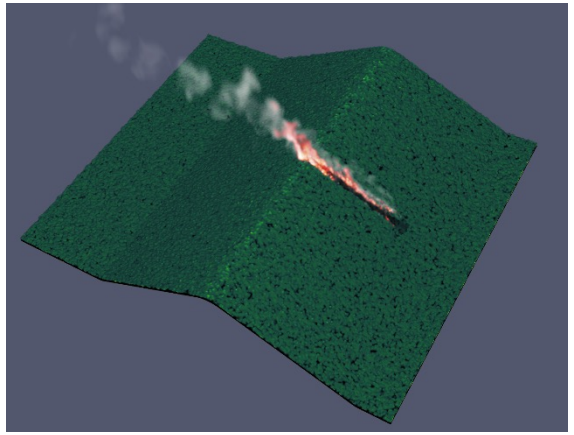
Color transfer function values

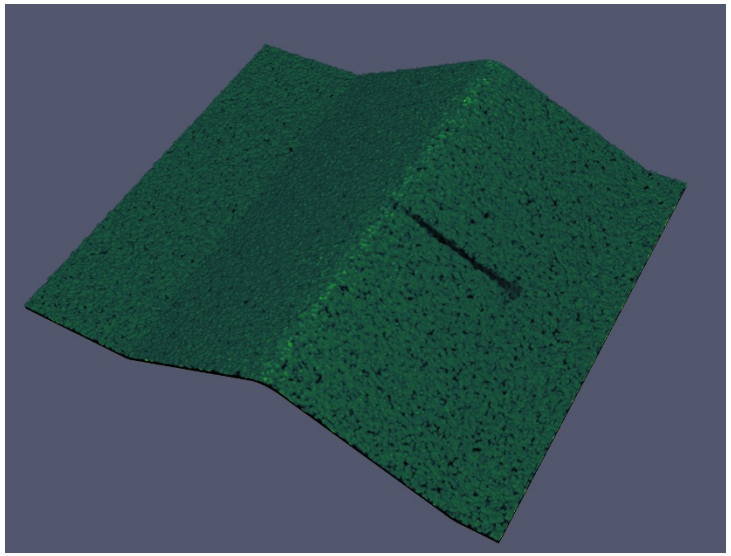
	Value	R	G	B
1	300.753	0	0	0
2	423.157	0.901961	0	0
3	545.56	0.901961	0.901961	0
4	606.762	1	1	1

Opacity transfer function values

	Value	Opacity
1	300.753	0
2	322.354	0.00297619
3	392.556	0.83631
4	606.762	1

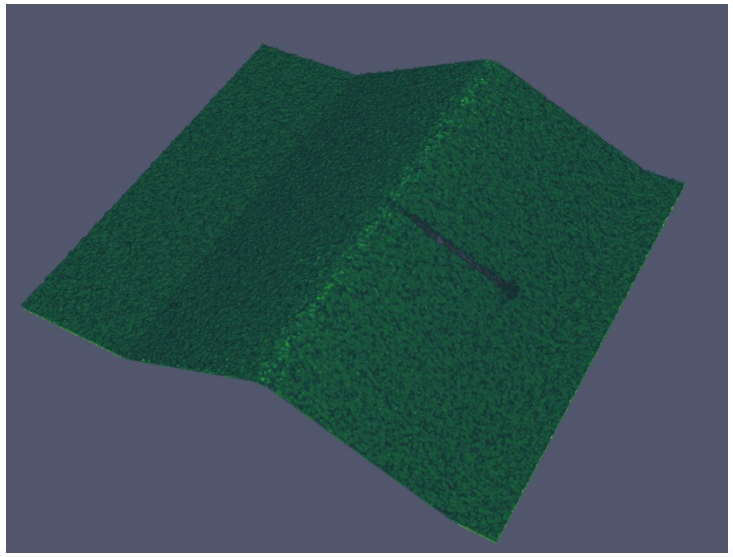
Direct Volume Rendering





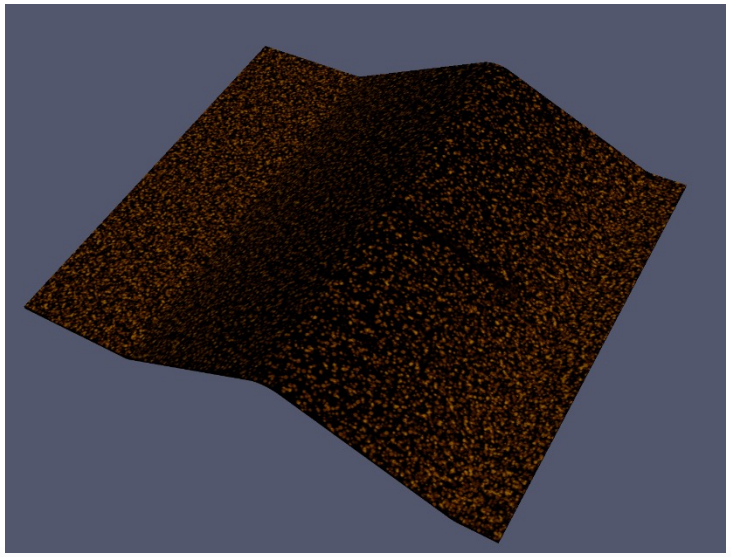
Grass & Soil

=



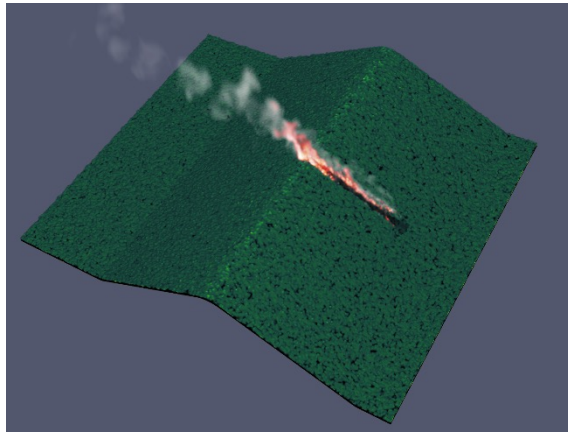
Grass

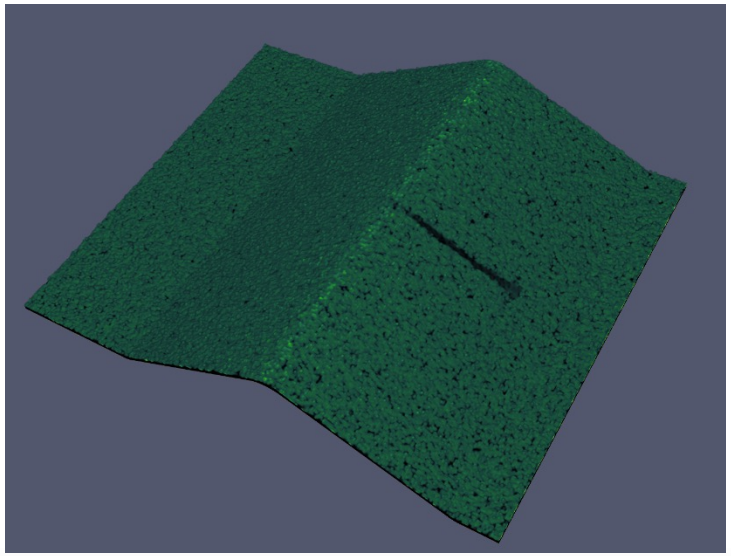
+



Soil

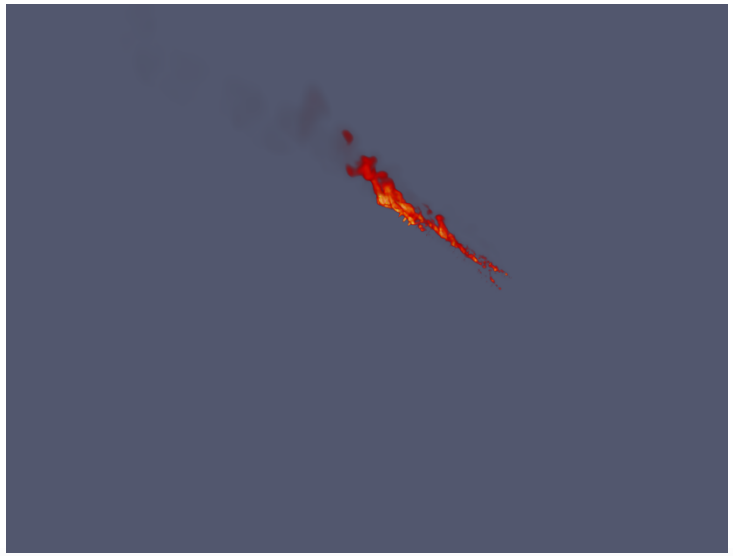
Direct Volume Rendering





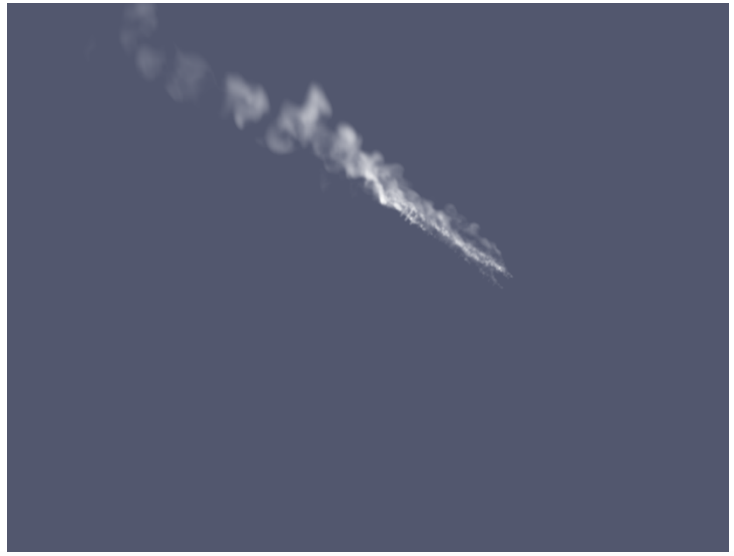
Grass & Soil

+



Fire

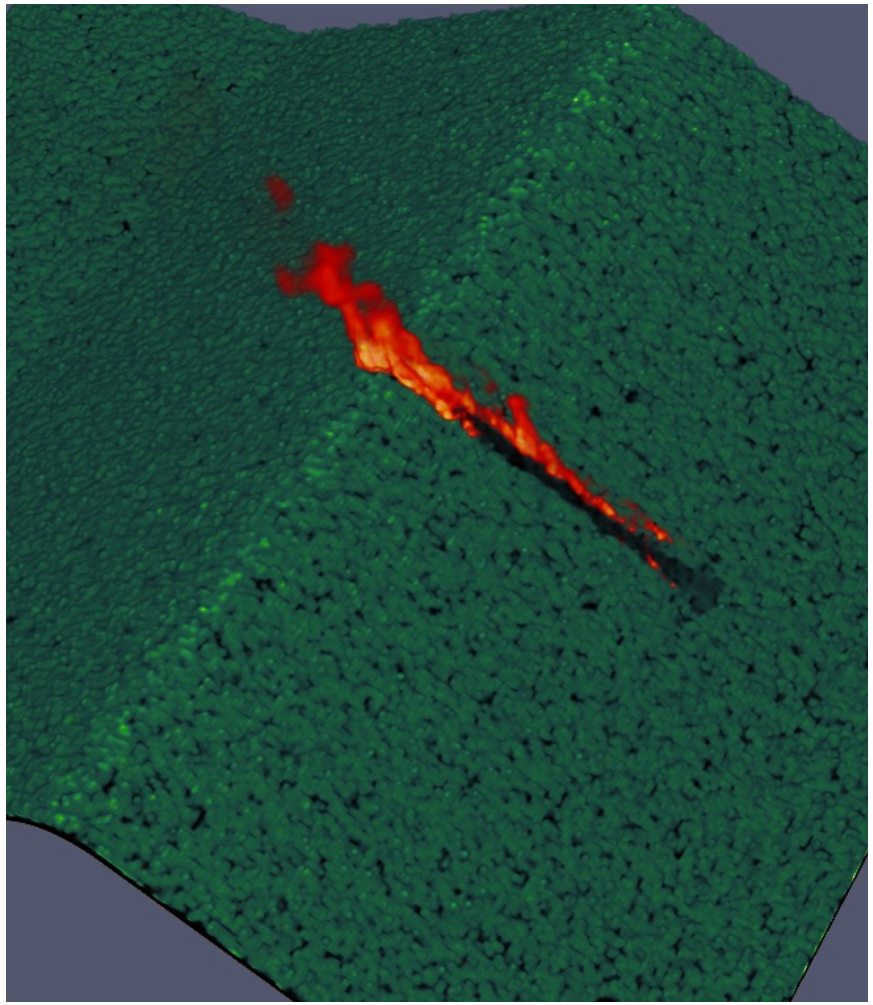
+



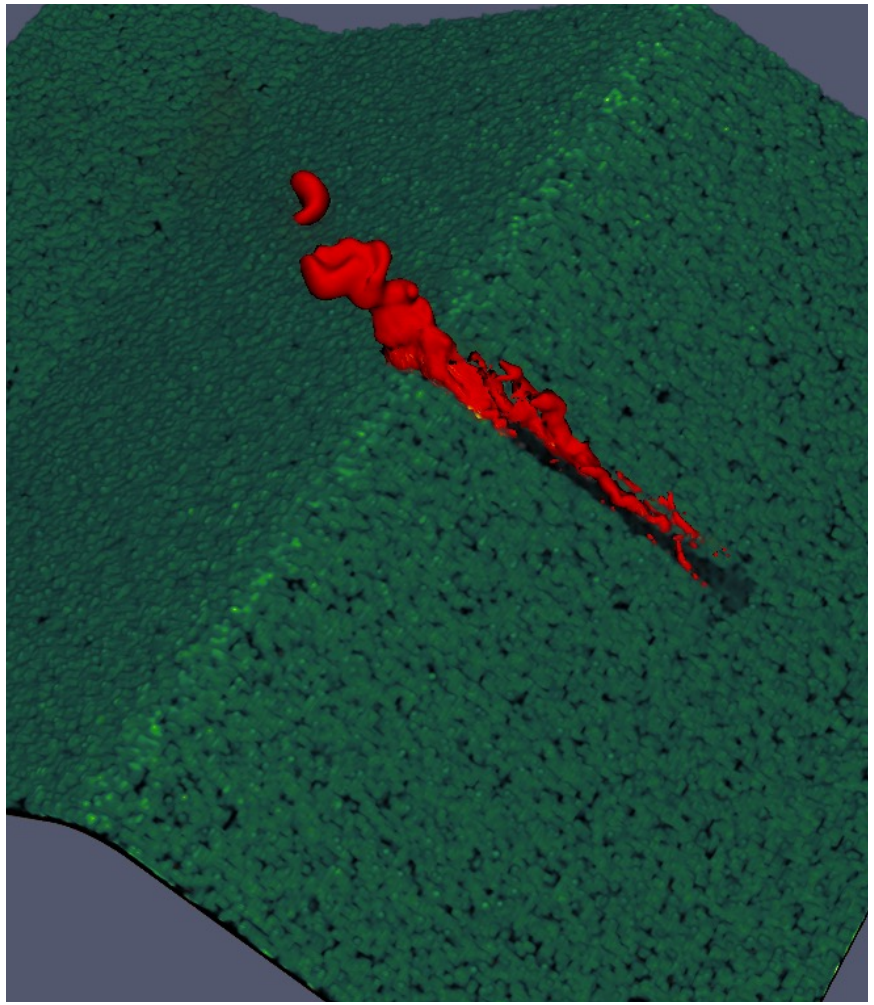
Vapor

Isosurface

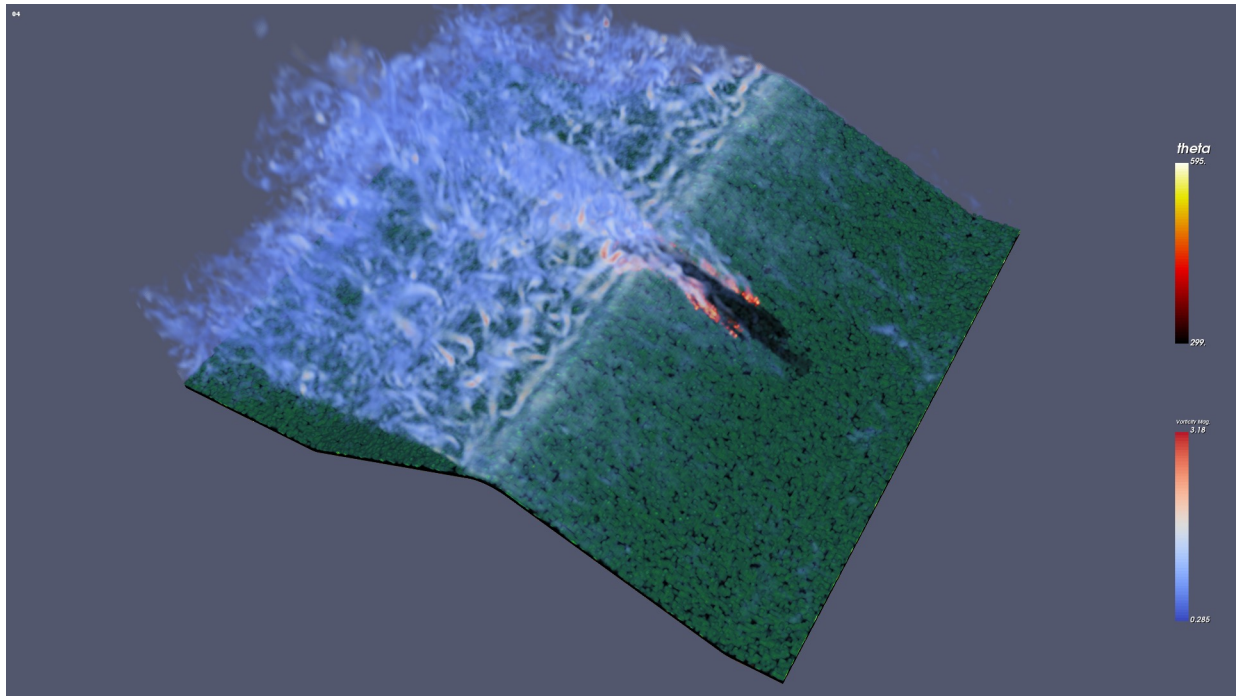
Volume



Isosurface

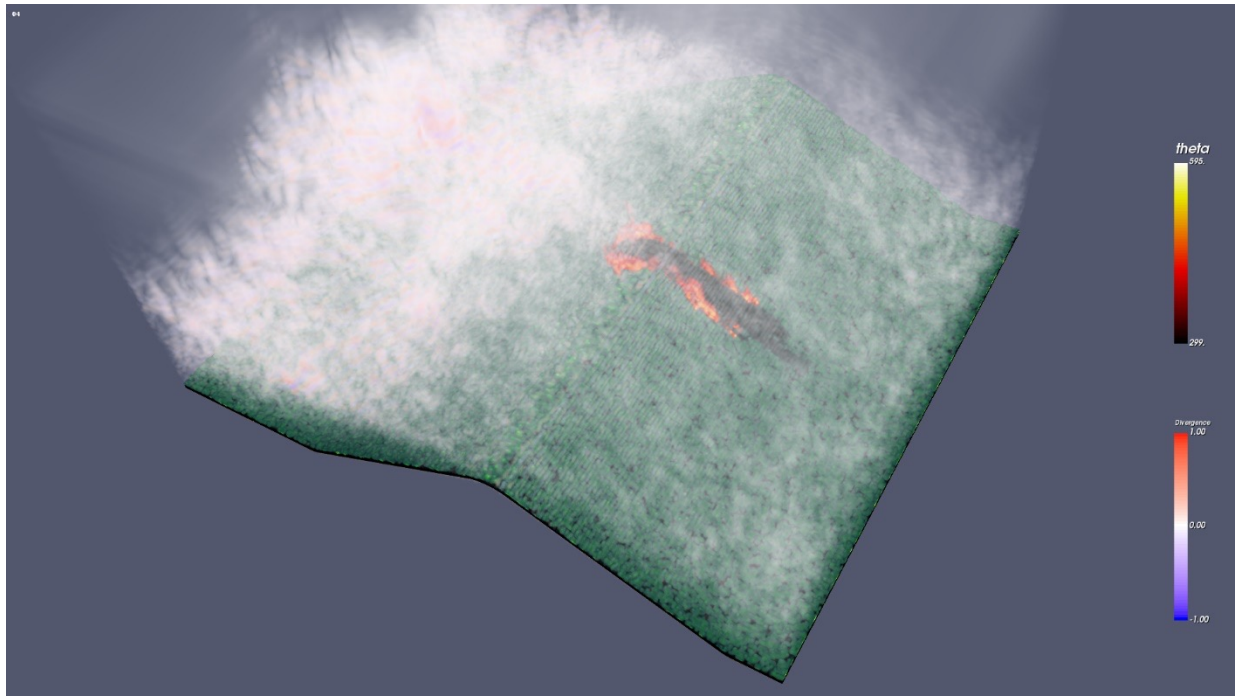


Vorticity



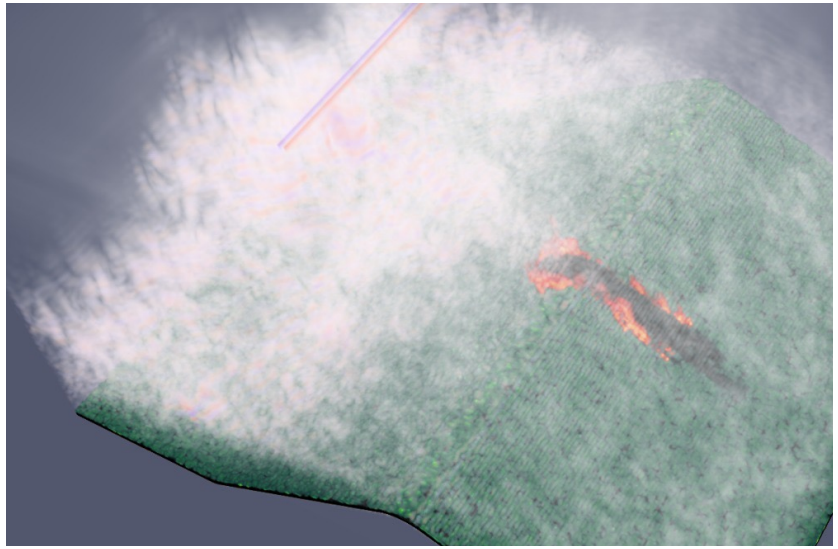
- Cell Derivatives with VectorModeToComputeVorticity()
- Convert to PointData
- Calculate Magnitude of Vorticity

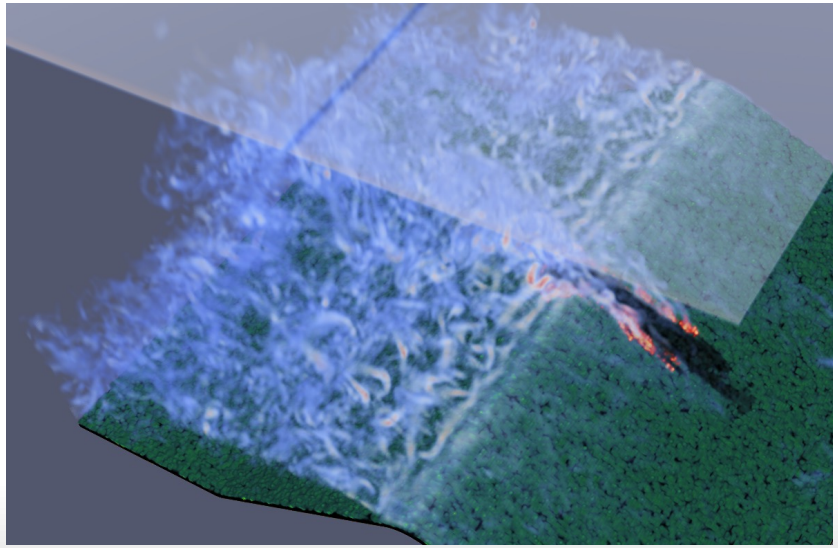
Divergence



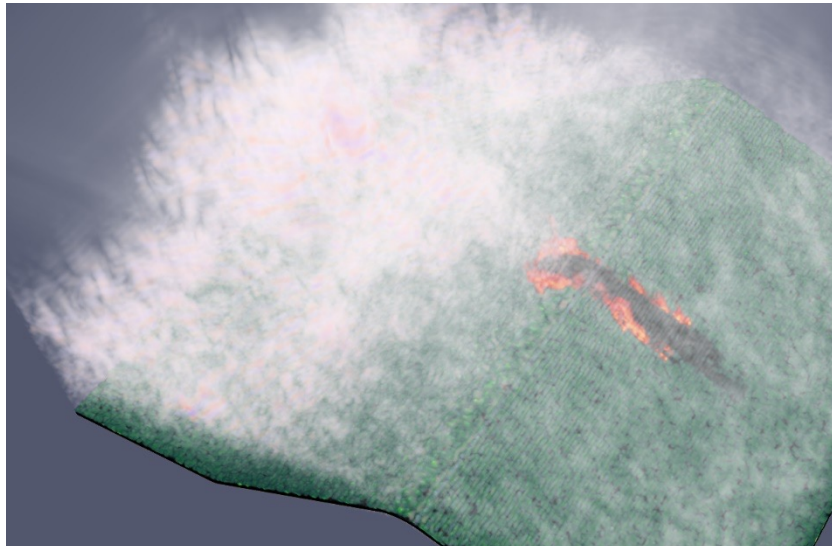
- Cell Derivatives with VectorModeToComputeGradient()
- Convert to PointData
- Calculate Divergence

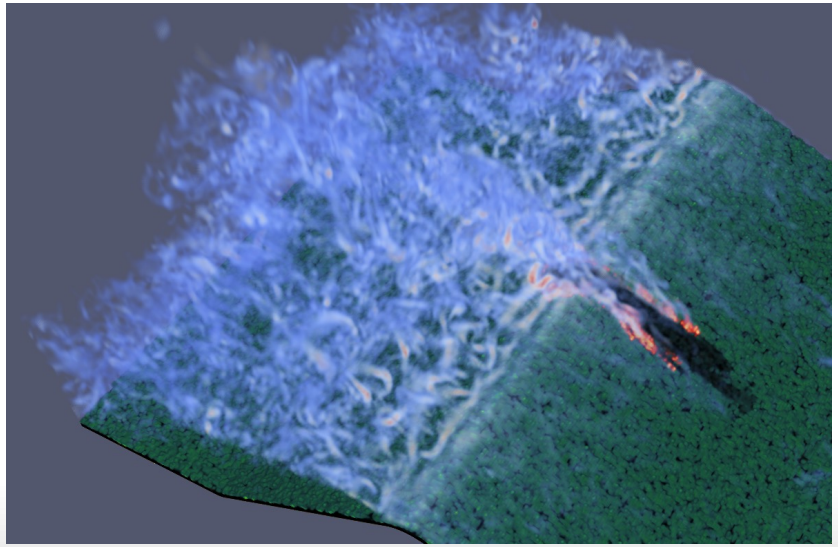
Removing Unexpected Artifacts





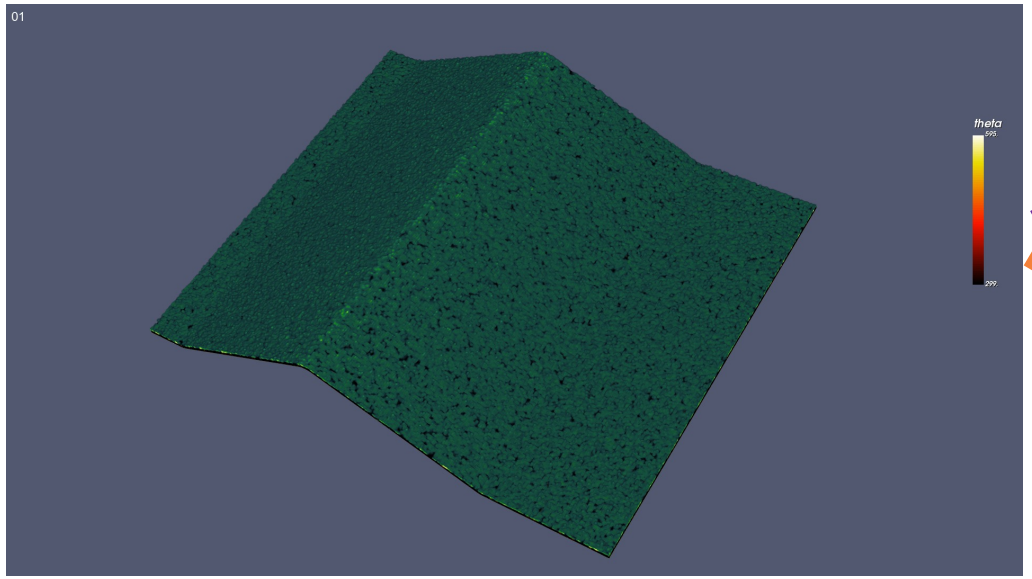
Masking

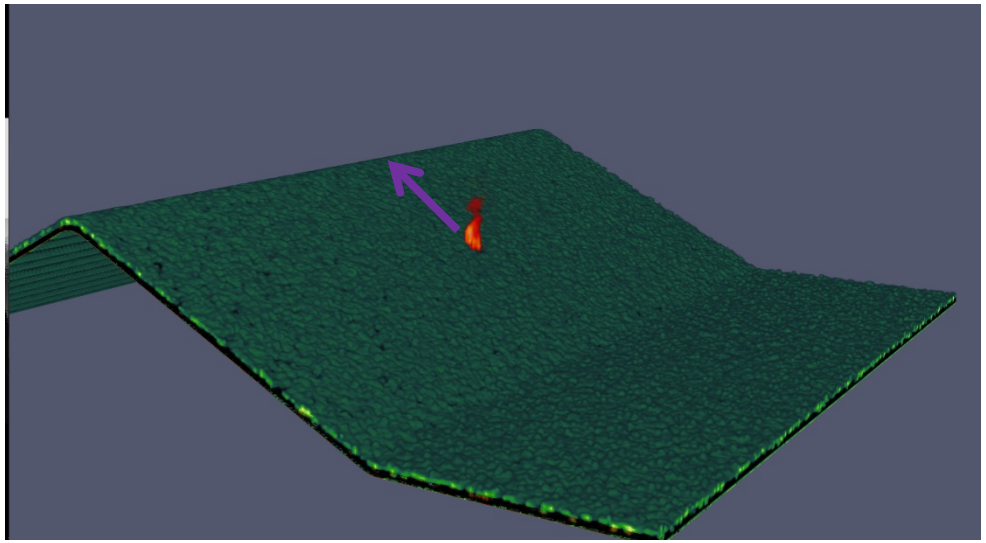


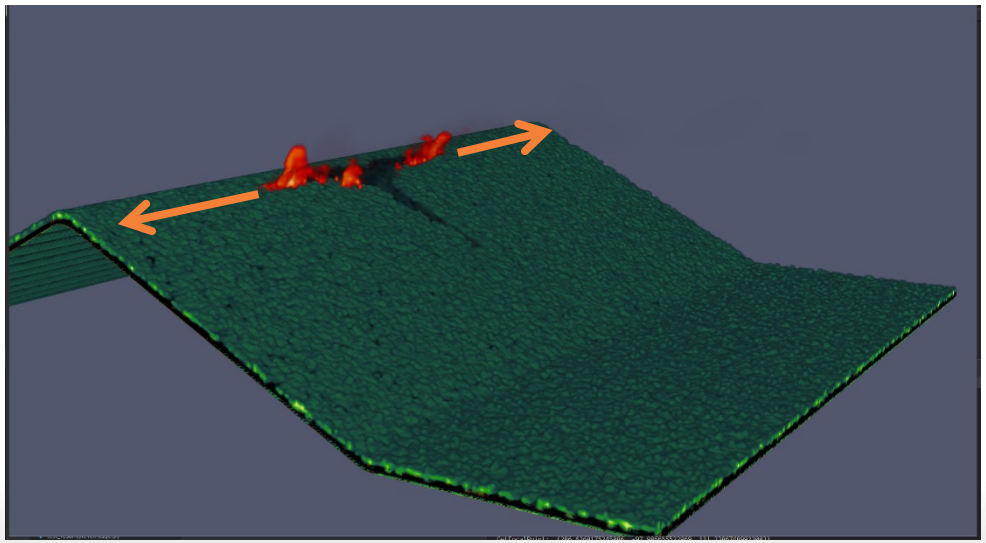


Analysis on Fire Spreading

- The spreading can be divided into 2 stages：
  - Stage 1: Going uphill.
  - Stage 2: Lateral spreading.







[1] Gould, Jim S., and Andrew L. Sullivan. "Two methods for calculating wildland fire rate of forward spread." *International journal of wildland fire* 29.3 (2020): 272-281.

Analysis on Fire Spreading

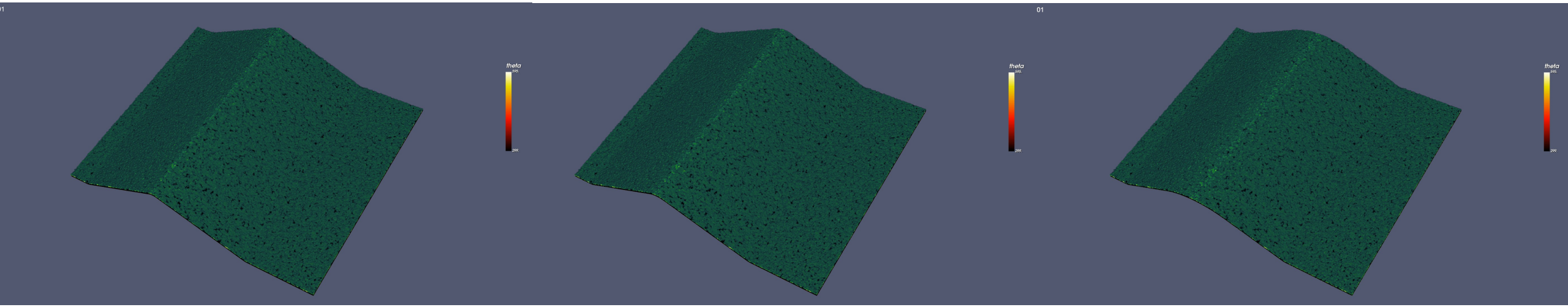
- The spreading can be divided into 2 stages：
  - Stage 1: Going uphill.
  - Stage 2: Lateral spreading.
- Spreading speed measure: **Cumulative Spread Rate (CSR)** [1] : **CSR** = slope distance / duration.
- *"**back**" = fire spreads against wind; "**head**" = fire spreads along wind.*
- *Smaller dataset number corresponds to smaller curvature radius, i.e. mountain is steeper.*

Dataset Type	Stage 1 Duration ($\times 10$s)	Stage 1 CSR ($\times 10^{-1}$m/s)	Stage 2 Duration ($\times 10$s)	Stage 2 CSR ($\times 10^{-1}$m/s)
back40	25	**~6.0**	40	**~17.8**
back80	31	~5.1	38	~10.2
back320	42	~3.1	28	~1.3
head40	15	**~19.5**	54	**~6.8**
head80	15	~18.7	70	~6.0
head320	14	~18.8	56	~1.9

Results

- **Finding 1**: Pointy peak leads to faster uphill & lateral spreading at both ignition positions.

Dataset Type	Stage 1 Duration (×10s)	Stage 1 CSR ($\times 10^{-1}$m/s)	Stage 2 Duration (×10s)	Stage 2 CSR ($\times 10^{-1}$m/s)
back40	25	**~6.0**	40	**~17.8**
back80	31	~5.1	38	~10.2
back320	42	~3.1	28	~1.3
head40	15	**~19.5**	54	**~6.8**
head80	15	~18.7	70	~6.0
head320	14	~18.8	56	~1.9

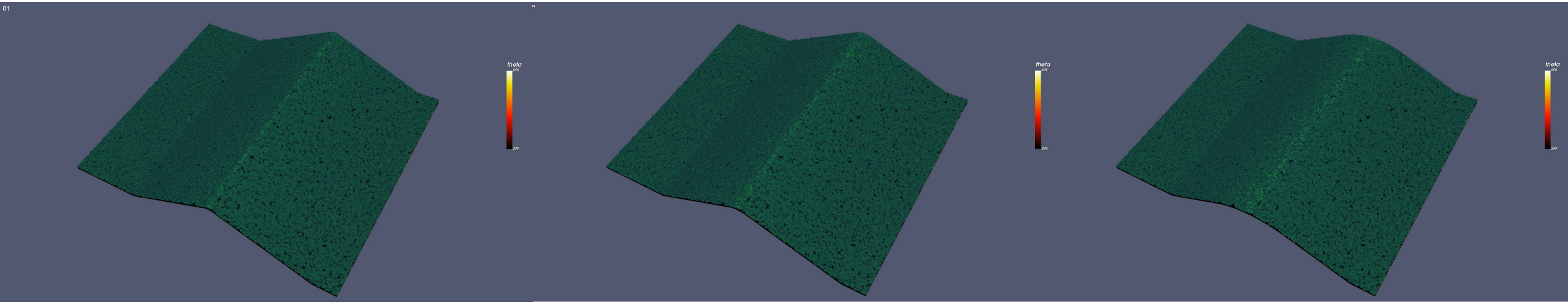


backcurve40 backcurve80 backcurve320

Results

- **Finding 1**: Pointy peak leads to faster uphill & lateral spreading at both ignition positions.

Dataset Type	Stage 1 Duration (×10s)	Stage 1 CSR ($\times 10^{-1}$m/s)	Stage 2 Duration (×10s)	Stage 2 CSR ($\times 10^{-1}$m/s)
back40	25	**~6.0**	40	**~17.8**
back80	31	~5.1	38	~10.2
back320	42	~3.1	28	~1.3
head40	15	**~19.5**	54	**~6.8**
head80	15	~18.7	70	~6.0
head320	14	~18.8	56	~1.9



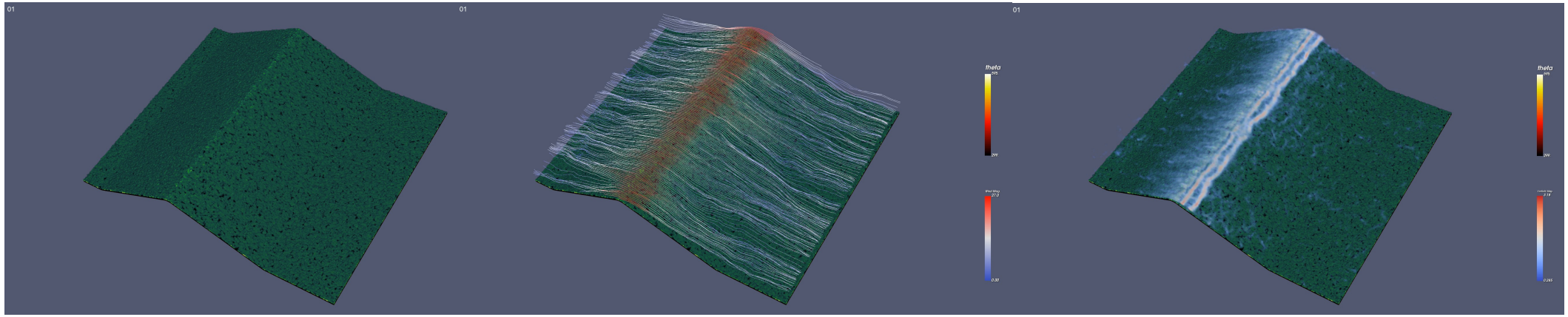
headcurve40 headcurve80 headcurve320

Results

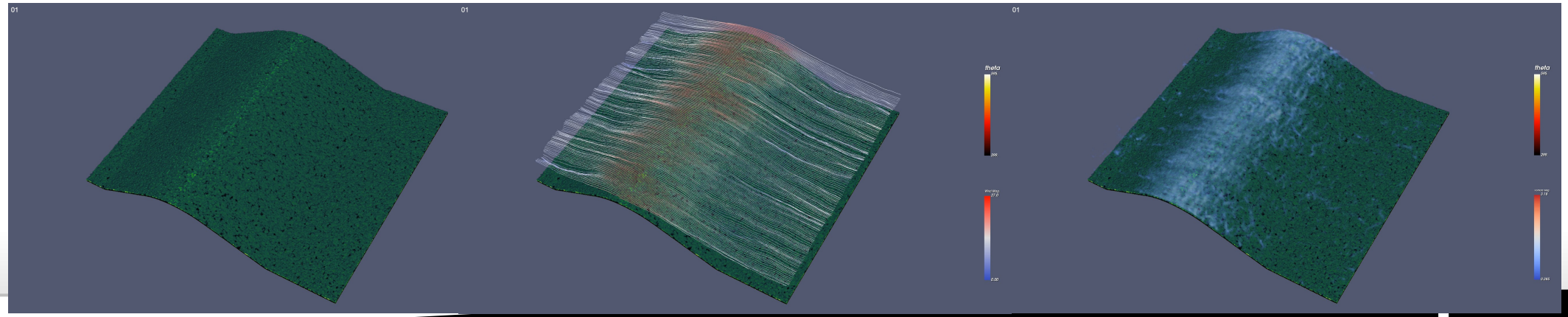
- **Finding 2**: The more pointy the mountain peak is, the faster lateral spreading it causes. More obvious on “backcurves”.

Dataset Type	Stage 1 Duration (×10s)	Stage 1 CSR ($\times 10^{-1}$m/s)	Stage 2 Duration (×10s)	Stage 2 CSR ($\times 10^{-1}$m/s)
back40	25	**~6.0**	40	**~17.8**
back80	31	~5.1	38	~10.2
back320	42	~3.1	28	~1.3
head40	15	**~19.5**	54	**~6.8**
head80	15	~18.7	70	~6.0
head320	14	~18.8	56	~1.9

backcurve40



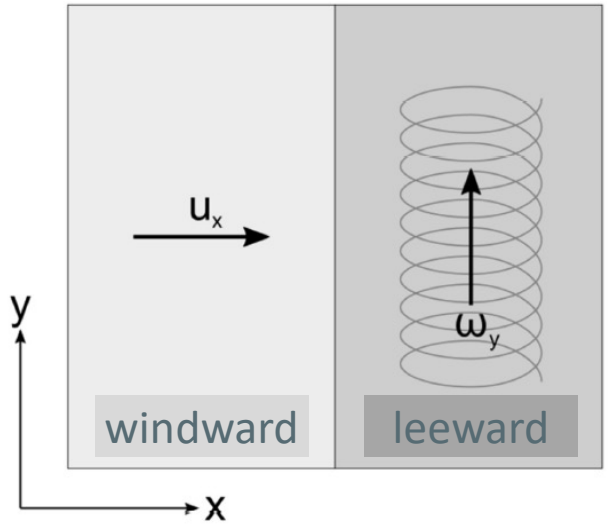
backcurve320

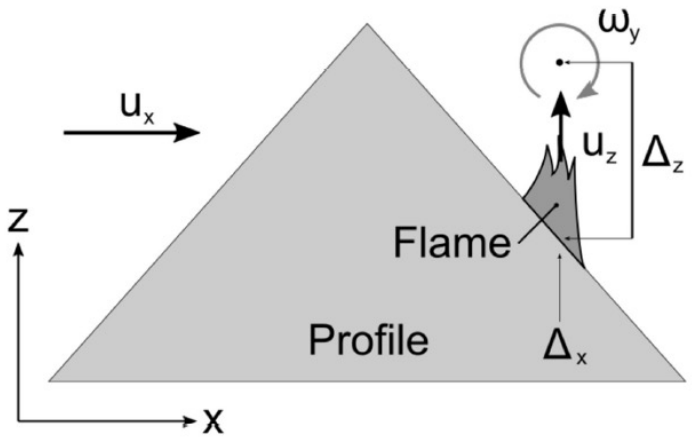


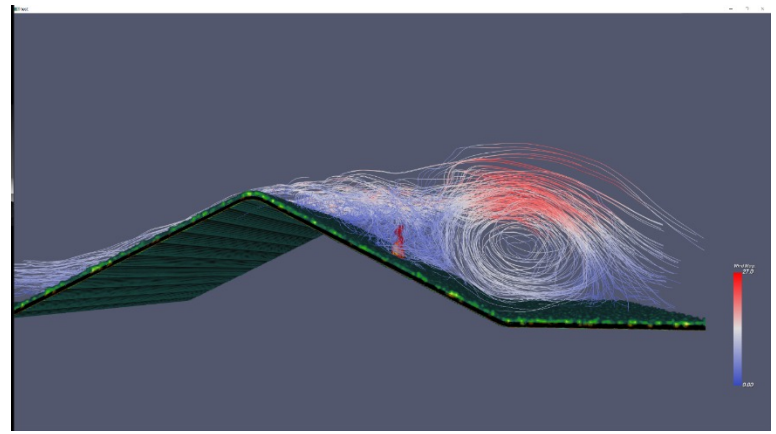
Results

Vorticity-drive Lateral Spread[1]

[1] Sharples, Jason J., and James E. Hilton. "Modeling vorticity-driven wildfire behavior using near-field techniques." Frontiers in Mechanical Engineering 5 (2020): 69.
[2] Sharples, J. J., et al. "Pyrogenic vorticity from windward and lee slope fires." Int. Congr. Model. Simul., Gold Coast, Aust. 29 Nov.–4 Dec (2015): 291-97.



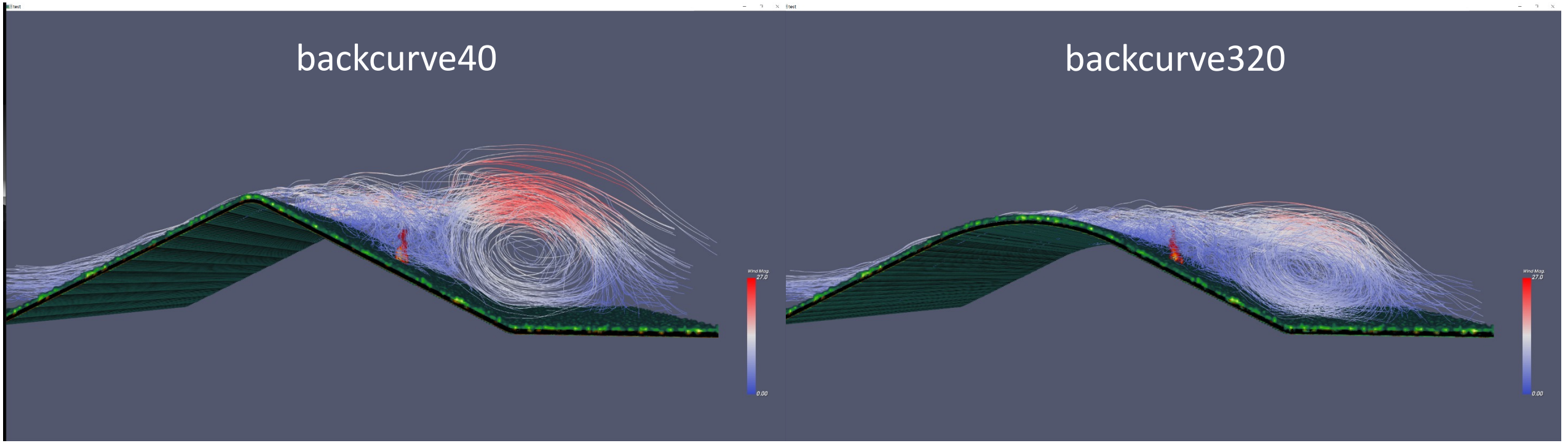




- Steep plane is more likely to cause separation of the air.
- “Separation of the flow creates horizontal vorticity over the leeward slope.” [1]
- …, this vertical vorticity promotes lateral fire propagation across the slope i.” [2]

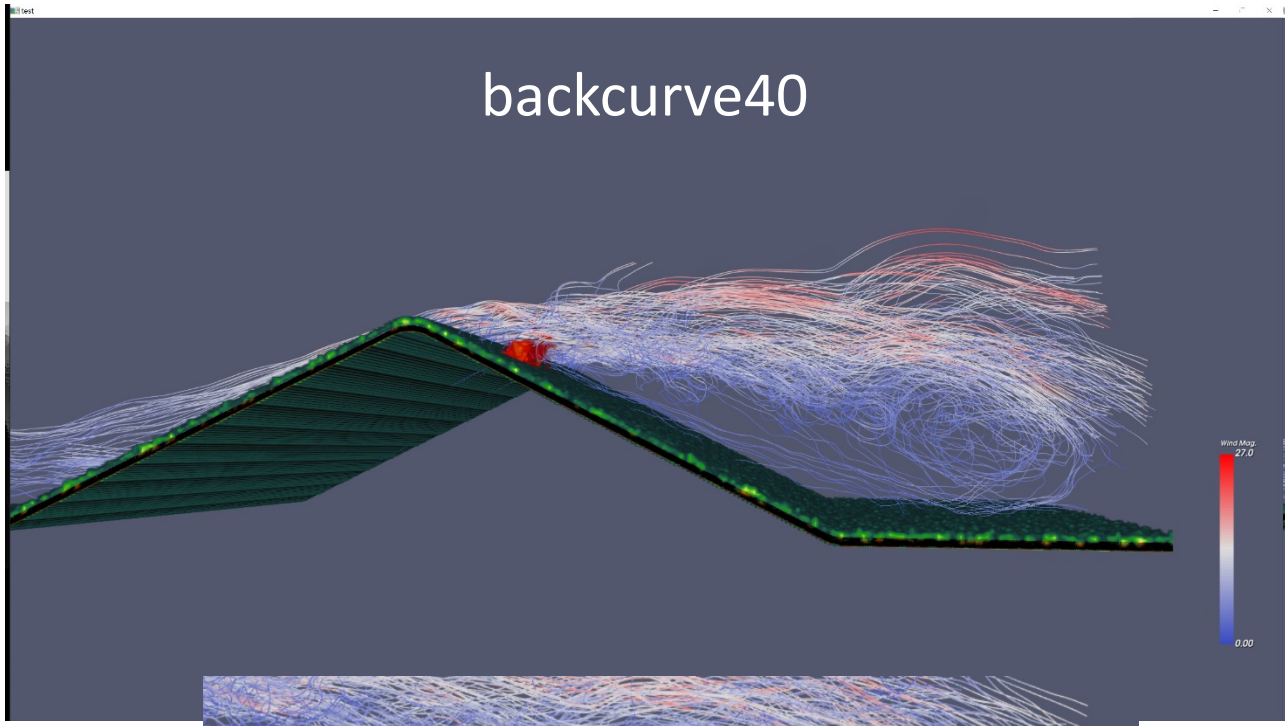
Impacts of terrain on spreading

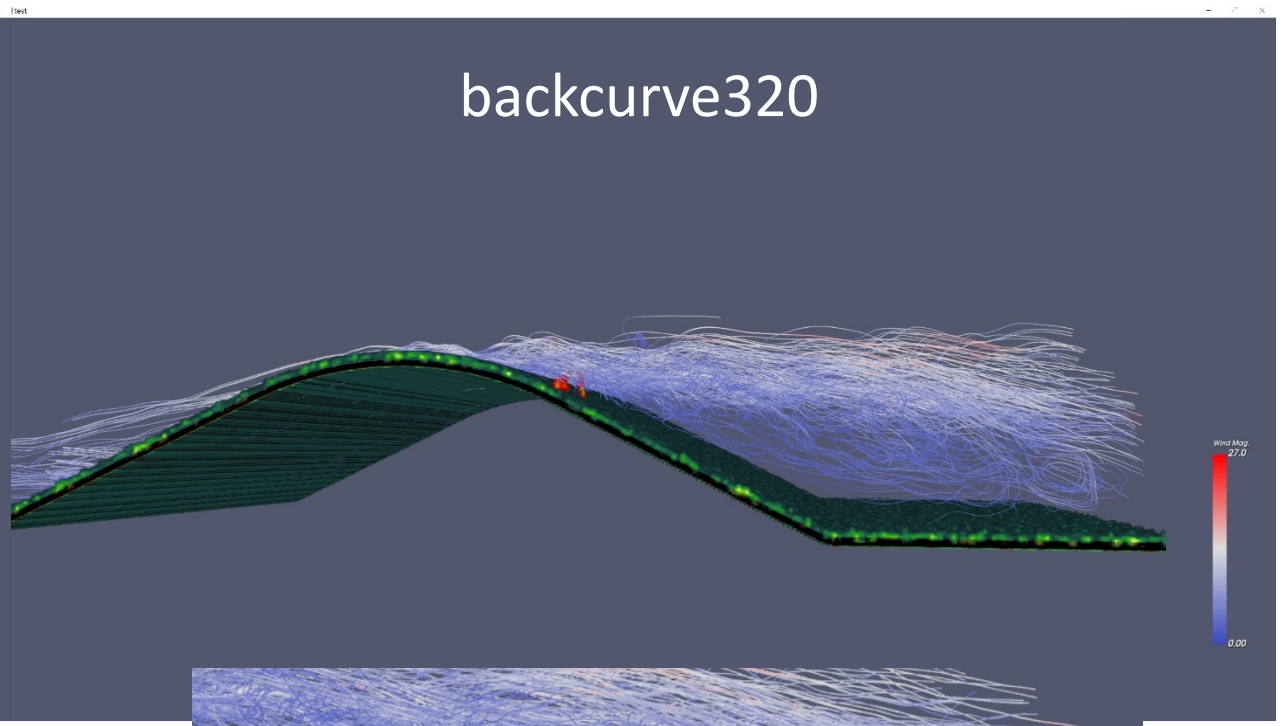
- Side view of streamlines at frame #8/70. *Seeding curves are placed at different heights.*

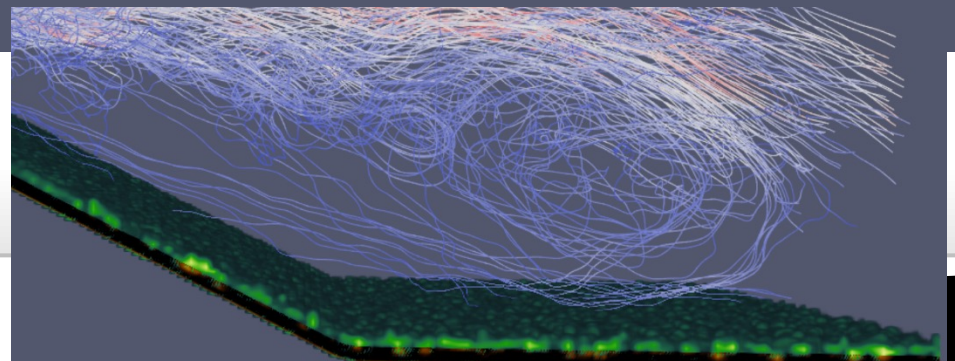


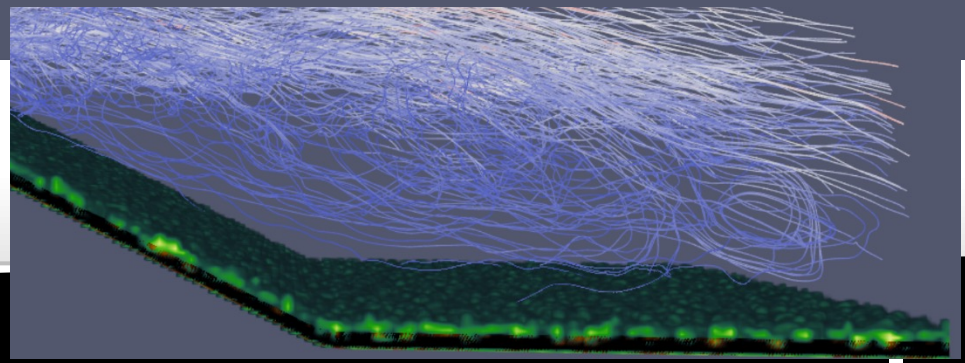
Impacts of terrain on spreading

- Side view of streamlines at frame #40/70. *Seeding curves are placed at different heights.*



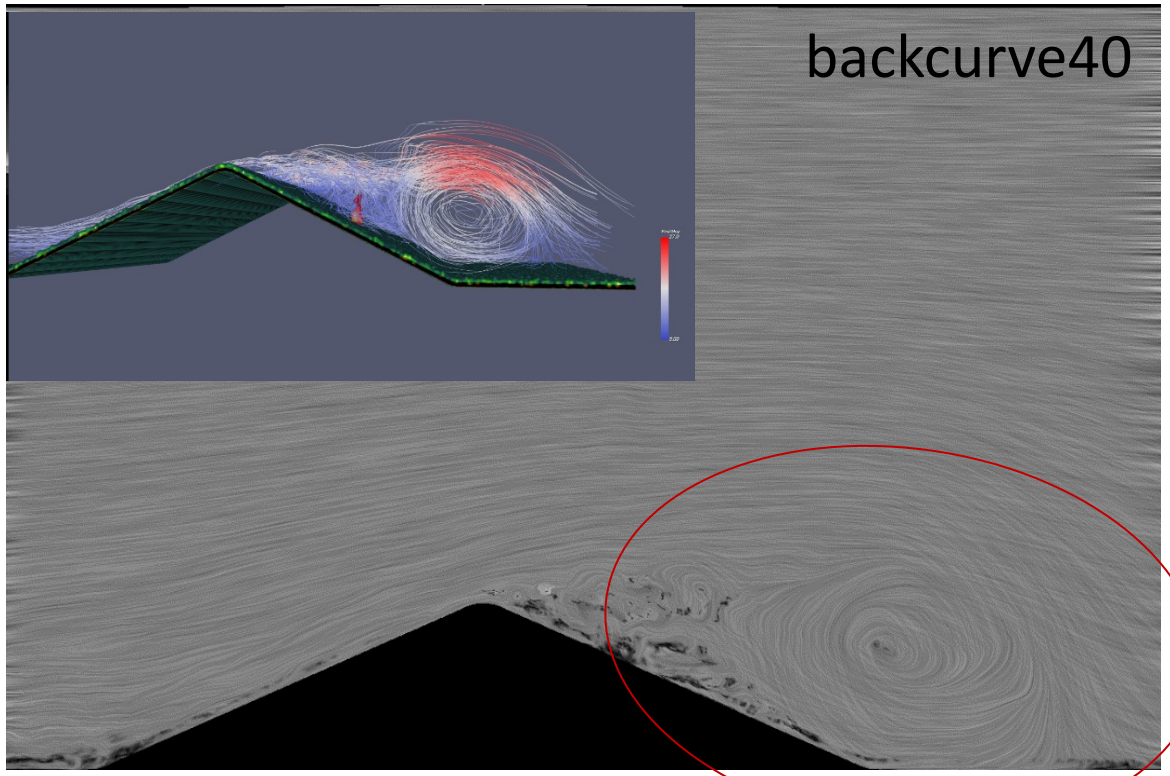


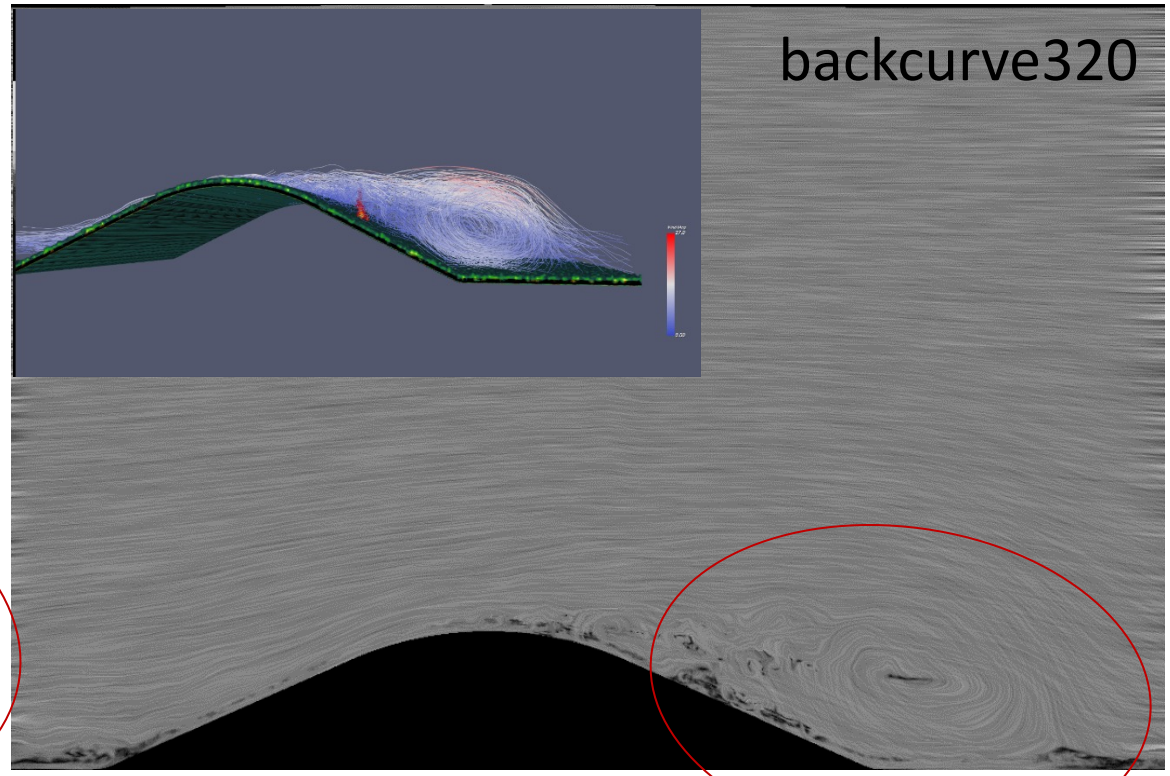




Impacts of terrain on spreading

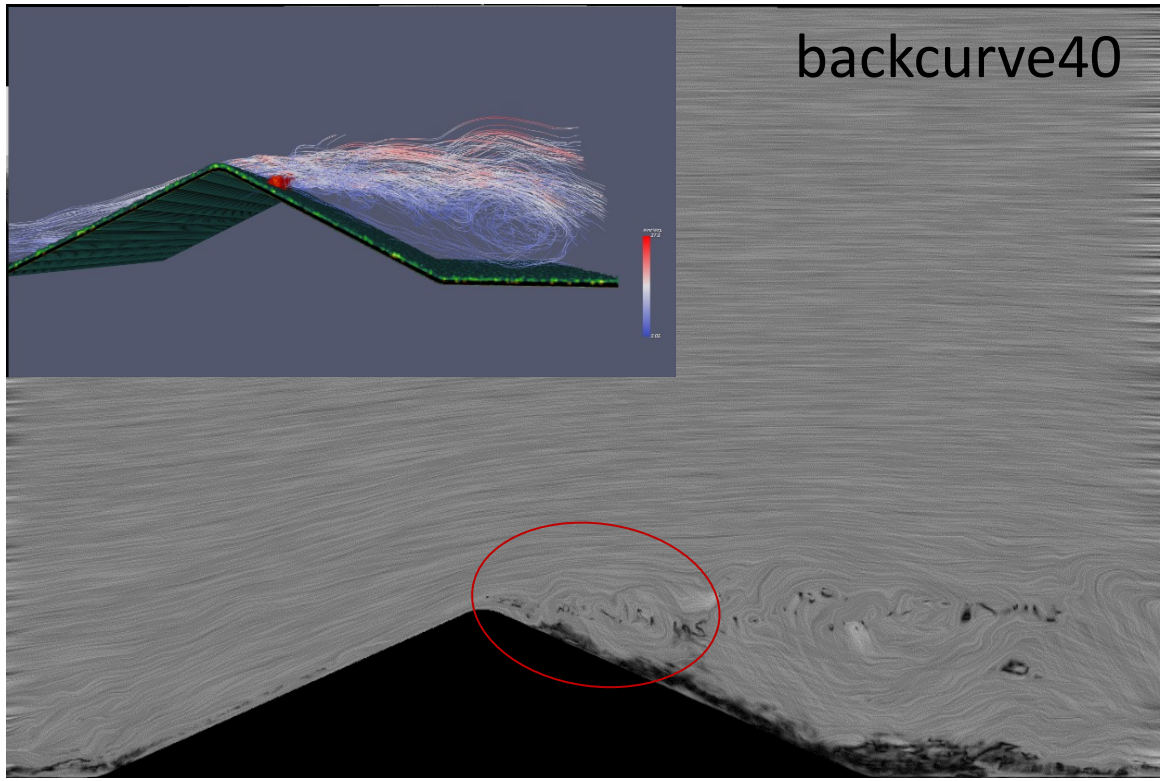
- Line Integral Convolution on wind velocity slice of frame #8/70.
- *Side view of streamlines doesn't show vortex as clear as LIC in some cases.*

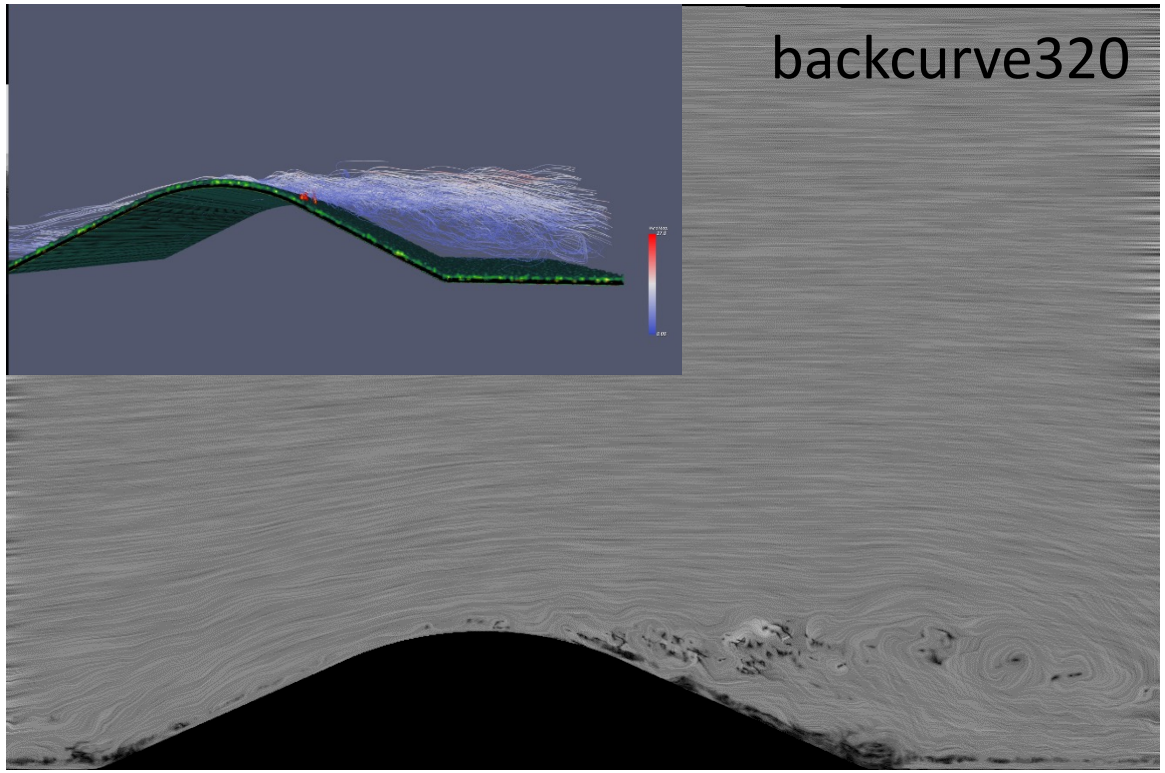




Impacts of terrain on spreading

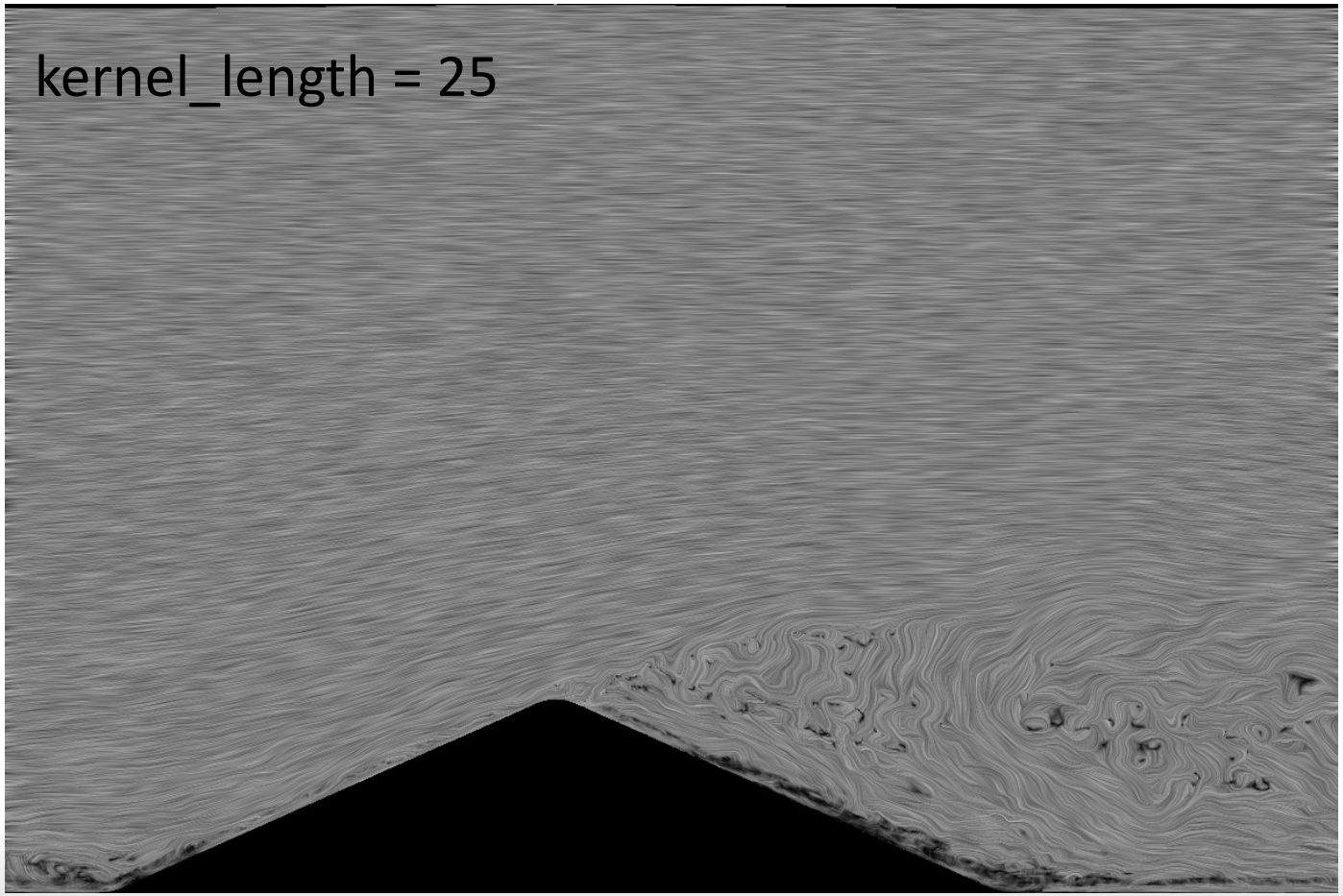
- Line Integral Convolution on wind velocity slice of frame #40/70.
- *Side view of streamlines doesn't show vortex as clear as LIC in some cases.*





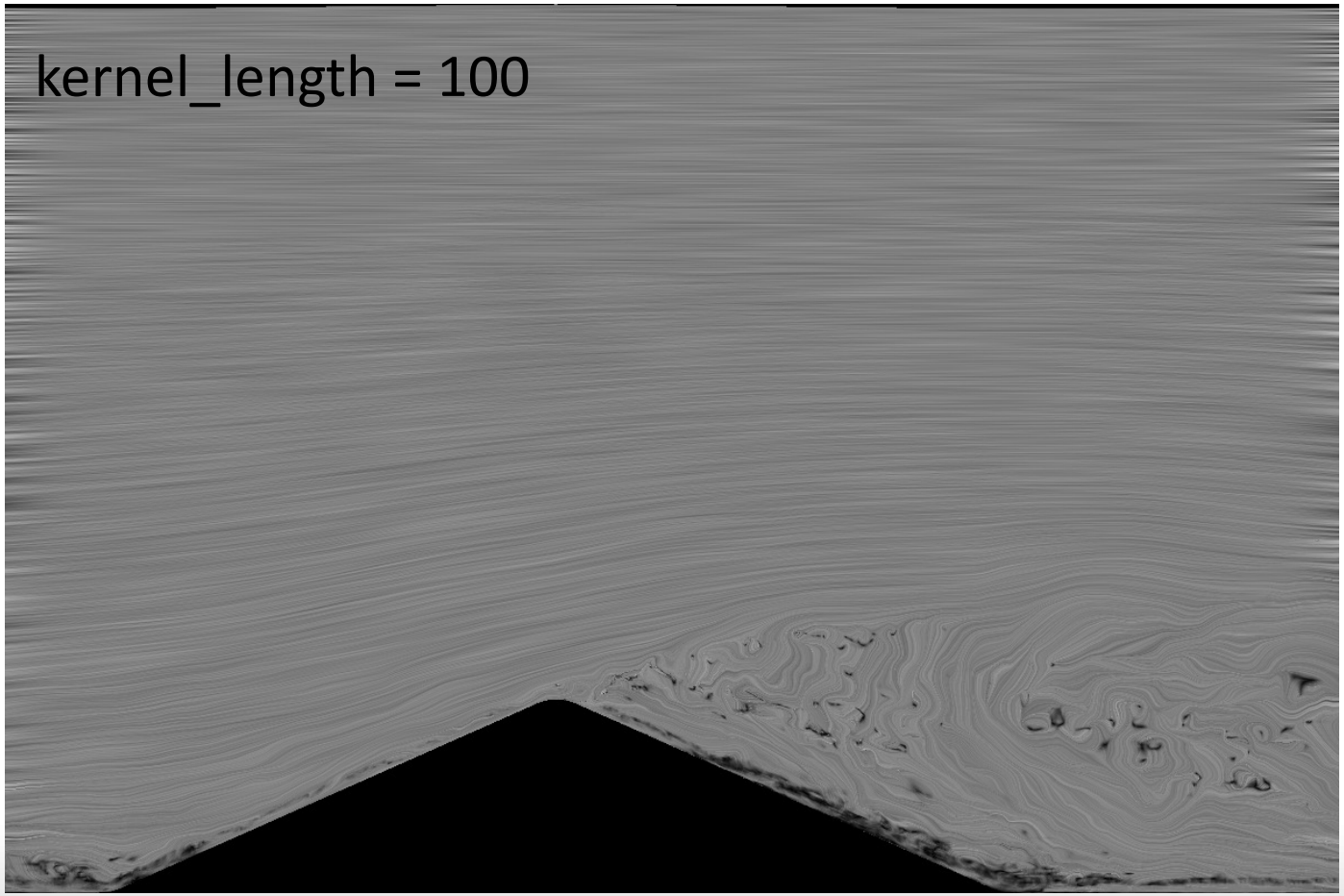
Impacts of terrain on spreading

- Line Integral Convolution with different kernel length.

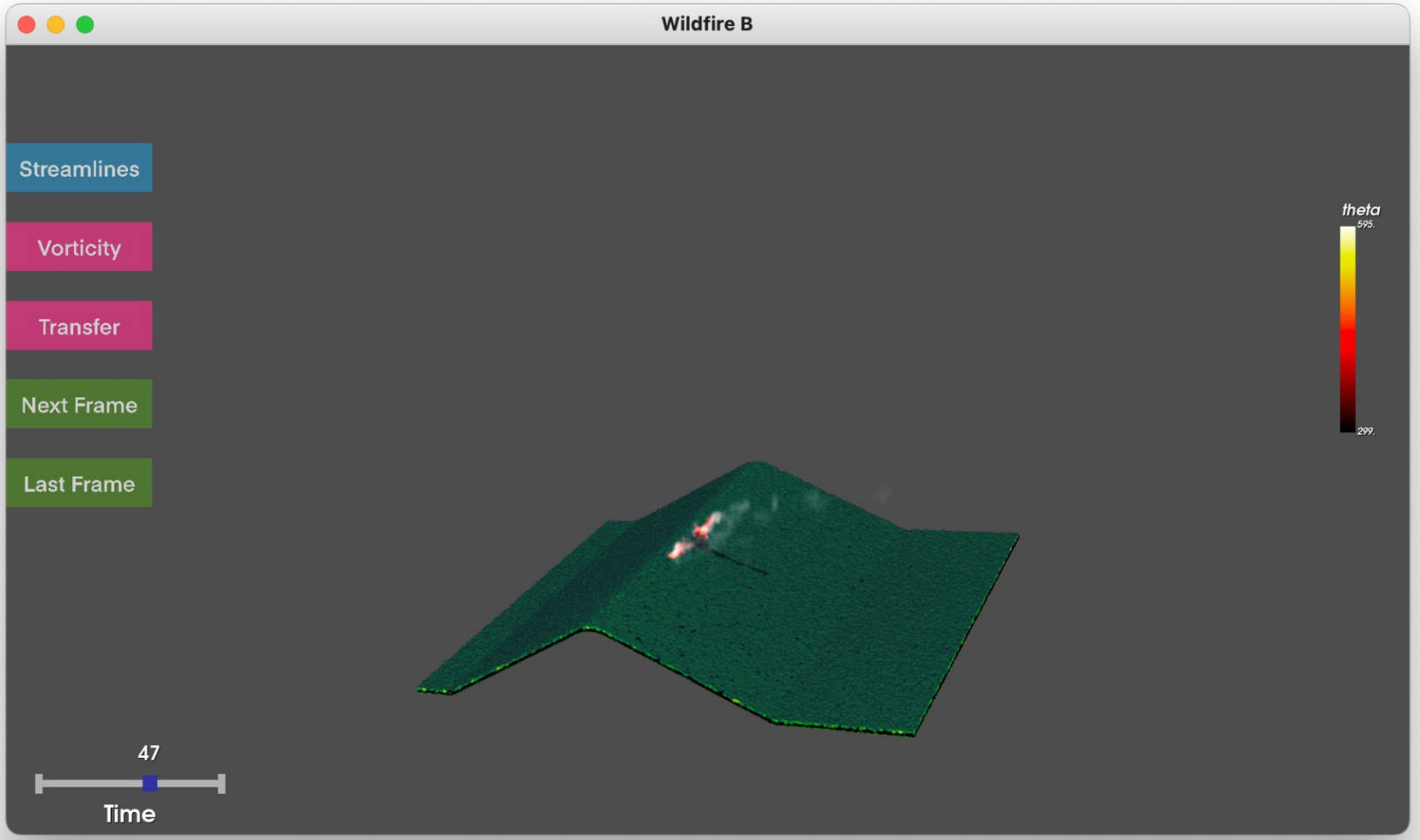


Impacts of terrain on spreading

- Line Integral Convolution with different kernel length.



GUI


Wildfire B
Streamlines
Vorticity
Transfer
Next Frame
Last Frame
47
Time
theta
595.
299.

Interactive GUI

- GUI was not very interactive–loading times too long
- Each frame takes ~18s to render

```
datasetReader = vtk.vtkXMLStructuredGridReader()
datasetReader.SetFileName('mountain_backcurve80/output.1000.vts')
datasetReader.Update()
dataset = datasetReader.GetOutput()
```

Too slow

GUI: Preprocessing

- 1st approach: Preprocessing to .vti, ~5s per frame
  - Still quite slow
- 2nd: approach: Preprocessing to .npz using numpy, **<1s per frame**
  - Significant speed up for interactivity!

GUI: Preprocessing to .npz using numpy

- Save contents as .npz file

```
preprocessed['grass'] = grassVtkArray
preprocessed['theta'] = thetaVtkArray
…
preprocessed['windVelocity'] = windVelocityVtkArray
preprocessed['vorticity'] = vortVtkArray

np.savez('preprocessed/output.1000.npz', **preprocessed)
```

GUI: Preprocessing to .npz using numpy

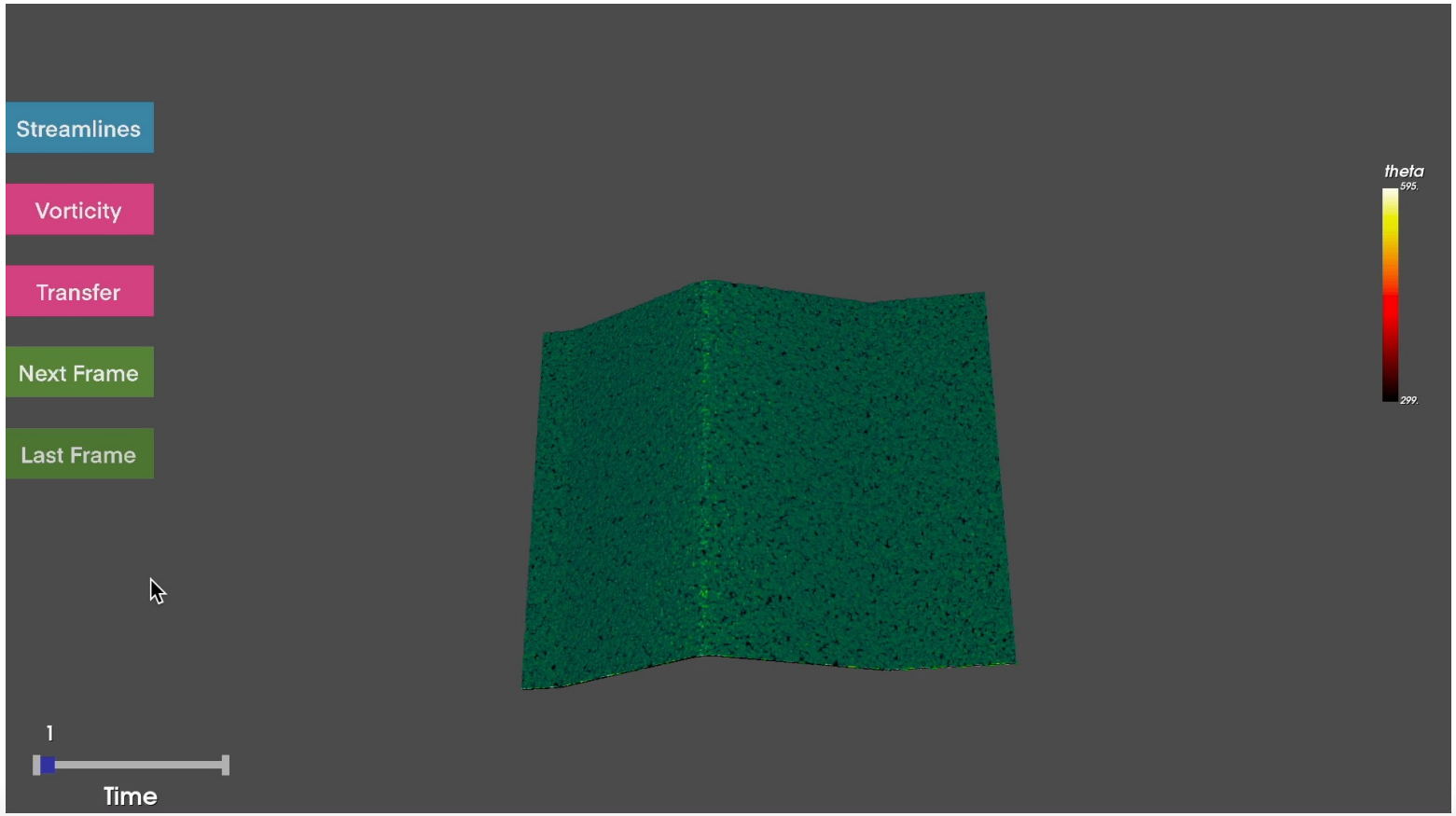
- Read contents from .npz file

```
dataset = np.load('preprocessed/output.1000.npz')
```

- Only for certain parts of the data
  - Still need a single .vts load on startup
- **Now takes <1s per frame**

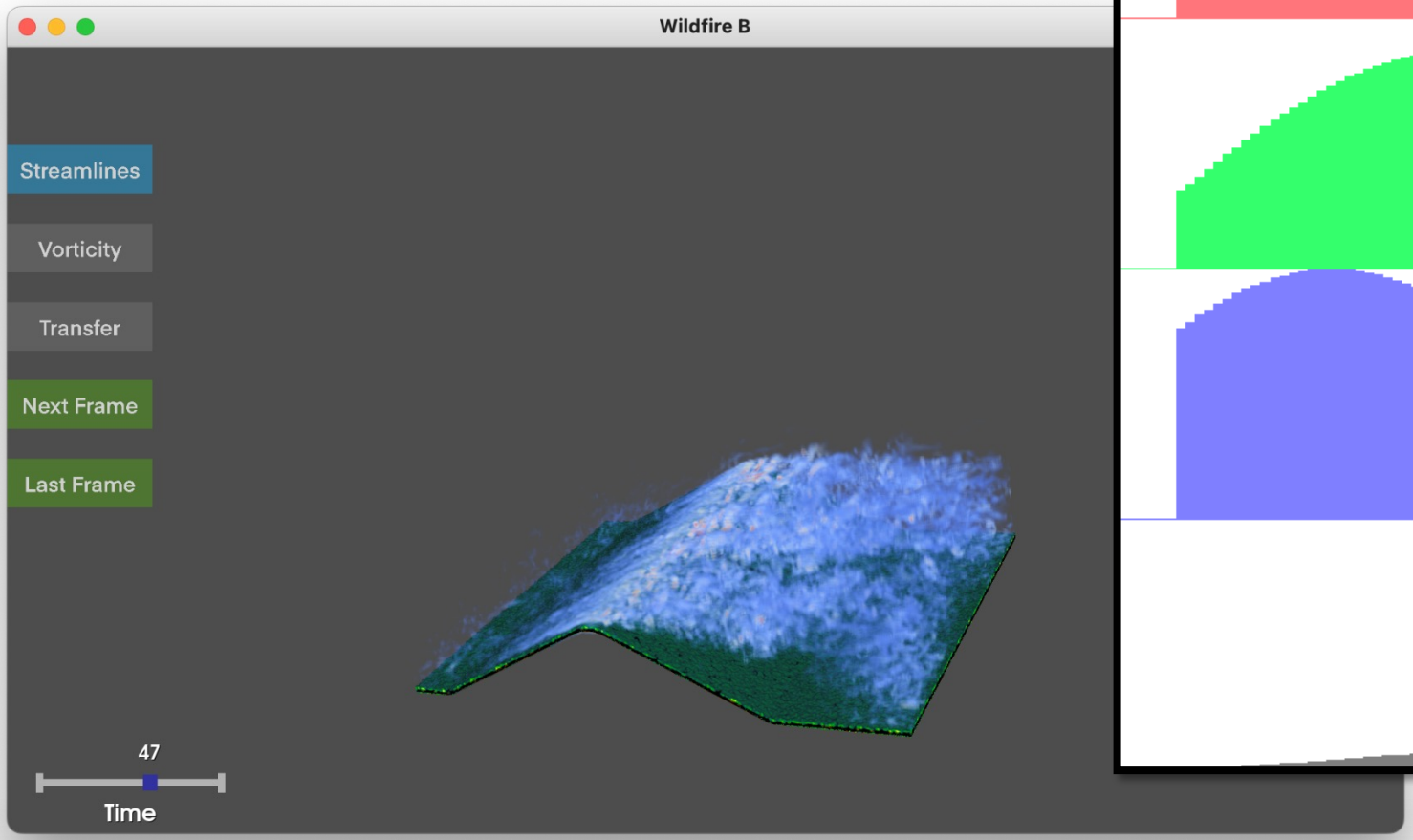
Interactive GUI

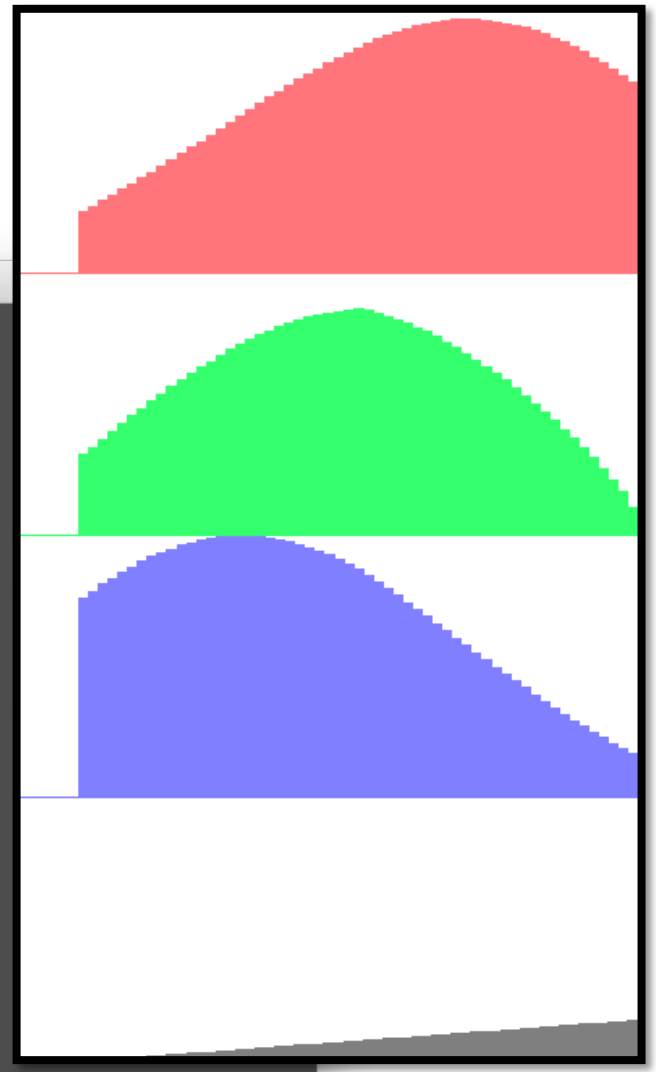
- With this change, we can interactively use the GUI



GUI: Transfer Function Editing

- Allows to edit color map (RGB and opacity)





GUI: Transfer Function Editing

- Load custom colormap as default
- Need to „unify“ domains

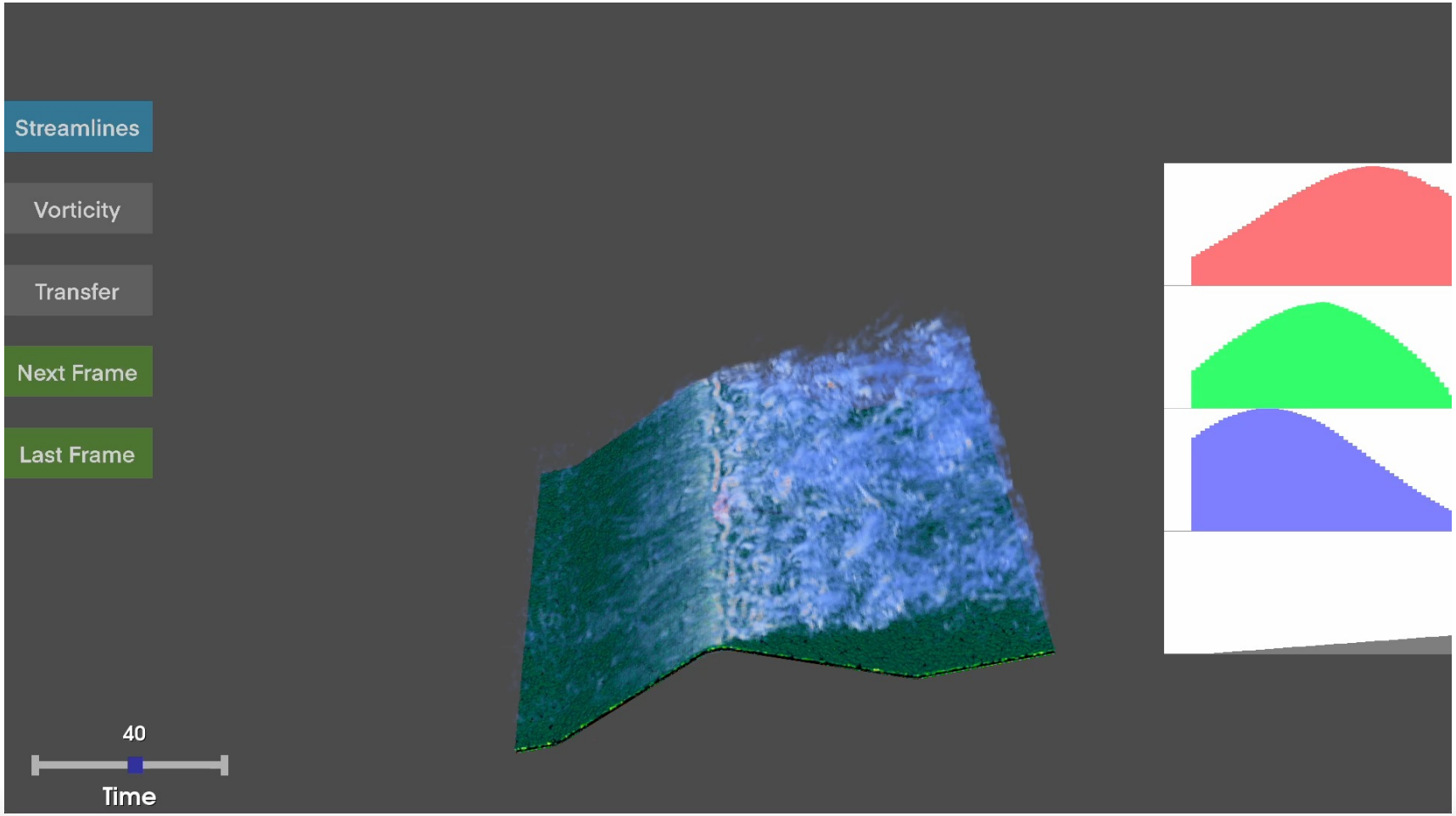
```
self.minValue = min(color_min, op_min)
self.maxValue = max(color_max, op_max)

for i in range(self.numPoints):
        t = i / (self.numPoints - 1)
        x = self.minValue + t * (self.maxValue - self.minValue)

        rgb = colormap.GetColor(x) // Zero when out of bounds
        op = opacitymap.GetValue(x) // Zero when out of bounds

        // Set data and transfer functions
```

GUI: Transfer Function Editing

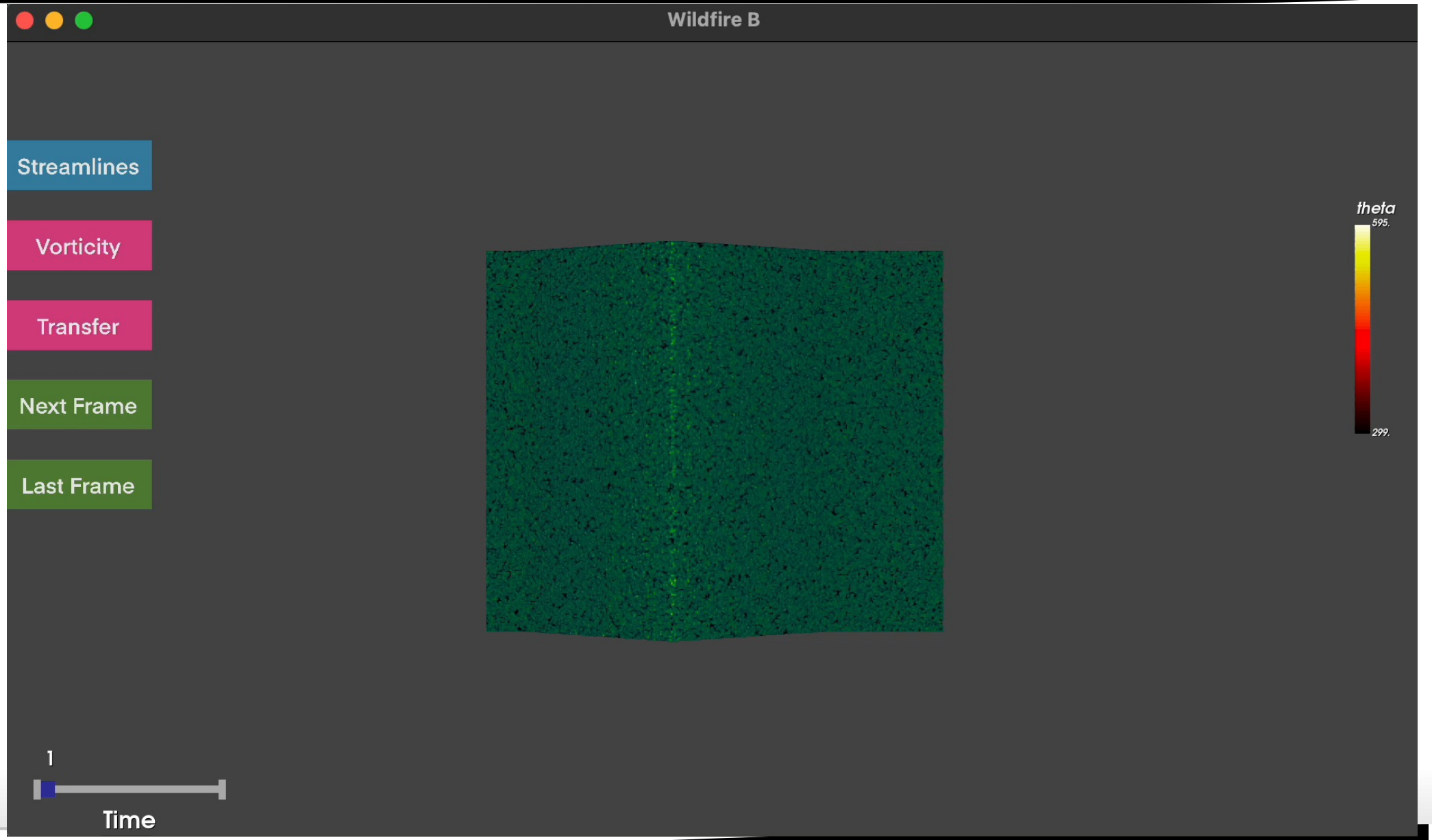


Conclusion

- Is there anything that is not yet up to the level you want it to be?
  - Hope LIC to achieve asymptotic convolution that reflects the direction of flow.
  - Hope to make use of rest variables (oxygen, heat transfers) for preciser visualization.

- What would you add if you would have more time?
  - Integrate LIC into GUI.
  - Rendering grass as an isosurface directly.

Demo


Wildfire B
Streamlines
Vorticity
Transfer
Next Frame
Last Frame
theta
595.
299.
1
Time