CIS 410/510: Isosurfacing
Announcements

• Will survey some advanced particle advection techniques later in term
• We need to switch to isosurfacing so we can keep on schedule with homeworks
• Upcoming:
  – Quiz Friday Feb 2\textsuperscript{nd}
  – Hank gone week of Feb 5.
    • Monday Feb 5: Brent lecture
    • Weds Feb 7: Sudhanshu lecture
    • Fri Feb 9: no class (possible YouTube lecture)
Quiz 1

• Will derive directly from Project 4
  – Euler advection steps
• Will use final 30 minutes of class
• Closed book. No notes, calculators, internet, etc.
Height field over a terrain
Height field over a terrain
Transparent grey plane at fixed elevation (height=20)
Rendering just the intersection of the plane and the height field
Projecting the “height = 20” lines back to 2D space

These lines are called isolines.

Isolines represent a region where the field is constant.

The isovalue for this plot is 20.
Plotting both isolines and height

Have you ever seen a plot like this before? Where?
Elevation Map of a Shield Volcano
Neat Project for Understanding Isolines
Neat Project for Understanding Isolines
Isolines vs Isosurfaces

• Isolines:
  – Input: scalar field over 2D space
  – Output: lines

• Isosurfaces:
  – Input: scalar field over 3D space
  – Output: surface

• Commonalities:
  – Reduce topological dimension by 1
  – Produce output where scalar field is constant
Isosurface of temperature from star explosion simulation

(ignore blue haze)
Iterating Over Cells

• For isosurface/isoline calculation, we can iterate over the cells.
• At the end, we take the results from each cell and put them into a single scene.
First isoline calculation

Goal: calculate isoline for field=0.5.

Quiz: do you have all the information you need?

Quiz: draw this graph and sketch the isoline.
10x10 sampling of the field

Colors:
>0.5: red
<0.5: blue

What observations can we make from this sampling?
100x100 sampling of the field

Colors:
- >0.5: red
- <0.5: blue
- 0.5: white

What observations can we make from this sampling?
Per pixel sampling of the field

Colors:
- >0.5: red
- <0.5: blue
- 0.5: white

What observations can we make from this sampling?
Isolines

- Correct isoline appears to be close to a quarter circle around (0,0) of radius 0.5.
- Quiz: How would we represent this quarter circle?

Colors:
- >0.5: red
- <0.5: blue
- 0.5: white
Reality Check: context for isolines

• We have millions of cells
• If one cell can produce tens or hundreds of lines, then the isolines could take much more memory than the input data set
Quiz

• You want to understand which will take more memory: the isolines or the input data set.

• What facts do you need to know?
Quiz answer

• You want to understand which will take more memory: the isolines or the input data set.
• What facts do you need to know?
• Need to know:
  – How much data per cell
  – How much data per isoline in a cell
  – How many cells contain isolines
How many cells contain isolines?

This cell contains isolines if the isovalue is between 0 and 1. Otherwise, it does not.

This question is data dependent & then depends on the isovalue.
How much data per isoline in a cell?

Straight lines are easy to represent. The memory for the correct answer is variable.
Big picture: what do we want from our isosurface?

• Tractable computation
  – Can’t create 100s or 1000s of line segments per cell for super-accurate representation

• Continuous surface
  – Triangles (or lines for isolines) need to connect up … no gaps.
Big idea #1: approximate the isolines / isosurface

- Isolines: represent them with a minimal # of segments
- Isosurface: represent them with a minimal # of triangles
Quiz: how to approximate our “quarter circle”?
Big picture: what do we want from our isosurface?

• Tractable computation
  – Can’t create 100s or 1000s of line segments per cell for super-accurate representation

• Quiz: did we accomplish this?
  – Yes: very few per cell

• Continuous surface
  – Triangles (or lines for isolines) need to connect up ... no gaps.

• Quiz: did we accomplish this?
  
  Answer: we got the answer exactly right at the edge of the cell ... hence no gaps. 😊
Effect of different isovalues

What are the similarities between these pictures?

Assume this cell layout

\[ X: 0 \rightarrow 1, \ Y: 0 \rightarrow 1, \ F(0,0) = 0, \ F(1,0) = F(1,1) = F(0,1) = 1 \]

Quiz: write pseudocode to calculate the isoline for any \( V, 0 < V < 1 \)
Assume this cell layout
\[ X:0 \rightarrow 1, \ Y:0 \rightarrow 1, \ F(0,0) = 0, \ F(1,0) = F(1,1) = F(0,1) = 1 \]

Quiz: write pseudocode to calculate the isoline for any \( V, \ 0 < V < 1 \)

Answer: \{ return ((V, 0), (0, V)); \}
Consider arbitrary layout

\[ F(0,1) = C \quad F(1,1) = D \]

\[ A < V \quad V < B \quad V < C \quad V < D \]

(V == isovalue)

Note that the mesh coordinates are pretty simple … you will need to take real coordinates into account.
Consider arbitrary layout

F(0,1) = C  F(1,1) = D

A < V
V < B
V < C
V < D
(V == isovalue)

Where is isoline?

P1 = (x,0)
P2 = (0,y)

Quiz:
What are x and y?

t = (V-A)/(B-A)
x = 0+t*(1-0)
t = (V-A)/(C-A)
y = 0+t*(1-0)
Claim: we understand one case

• Case “1”:
  – \( F(P0) < V \)
  – \( V < F(P1) \)
  – \( V < F(P2) \)
  – \( V < F(P3) \)

• Quiz: how many cases are there?
• Answer:
  – 2 possible states for each point \( P \): \( F(P) < V \) or \( V < F(P) \)
    • (note we ignore \( F(P) == V \) for today)
  – 4 vertices, so ... \( 2^4 = 16 \) cases
    • Some cases are similar to each other
The 16 cases
We explored case 14
Quiz: write down cases #s that are similar to case 14
Quiz: how many different groupings are there?
Quiz answer:
There are 4 groupings
Problem case: ambiguity!!

Solution: just pick one and go with it.
Physical interpretation of ambiguity

One way connects them up, the other separates them. What’s right?
Big idea #2: pre-computation of all cases

If you knew which case you had, then you would know how to proceed

- Pre-compute correct answers for all 16 cases and store them in a lookup table
- For each cell, identify which case it is in
- Then use corresponding lookup table to generate isosurface
Big idea #2: pre-computation of all cases

- Pre-compute correct answers for all 16 cases and store them in a lookup table
- For each cell, identify which case it is in
- Then use corresponding lookup table to generate isosurface

If you knew which case you had, then you would know how to proceed
Pre-compute correct answers for all 16 cases and store them in a lookup table

- Observations about correct answers for a case:
  - It contains one or two line segments
  - The ends of the line segments are always along edges.
Pre-compute correct answers for all 16 cases and store them in a lookup table.

- The ends of the line segments are always along edges.
  - We will need to number the edges.
Pre-compute correct answers for all 16 cases and store them in a lookup table

- Correct answer for this case:
  - There is one line segment
  - That line segment has end points on edge 0 and edge 3
Big idea #2: pre-computation of all cases

If you knew which case you had, then you would know how to proceed

- Pre-compute correct answers for all 16 cases and store them in a lookup table
- For each cell, identify which case it is in
- Then use corresponding lookup table to generate isosurface
For each cell, identify which case it is in:

- 4 vertices
- Each has one of 2 possible classification values
  - Lower than isovalue
    - Call this “0”
  - Higher than isovalue
    - Call this “1”
  - (ignore equality case)

Quiz: write down classification value for each vertex
For each cell, identify which case it is in

- **Goal:** turn classification values into a number
  - Number should be between 0 and 15
- **Idea:** use binary numbers
  - \( V3V2V1V0 \rightarrow 1\ 1\ 1\ 0 \rightarrow 14 \)

This is case 14
The 16 cases

Case 0

Case 1-5

Case 6-10

Case 11-15
Big idea #2: pre-computation of all cases

If you knew which case you had, then you would know how to proceed

- Pre-compute correct answers for all 16 cases and store them in a lookup table
- For each cell, identify which case it is in
- Then use corresponding lookup table to generate isosurface
Then use corresponding lookup table to generate isosurface

```c
int numSegments[16];
numSegments[0] = 0;
...
numSegments[6] = 2;
...
numSegments[14] = 1;
numSegments[15] = 0;
```
Then use corresponding lookup table to generate isosurface

```c
int lup[16][4]; // lup == lookup
lup[0][0] = lup[0][1] = lup[0][2] = lup[0][3] = -1;
...
lup[6][0] = 0; lup[6][1] = 1; lup[6][2]=2; lup[6][3] = 3;
...
lup[14][0] = 0; lup[14][1] = 3; lup[14][2] = lup[14][3] = -1;
```
Then use corresponding lookup table to generate isosurface

```c++
int icase = IdentifyCase(cell); // case is a reserved word in C++
int nsegments = numSegments[icase];
for (int i = 0; i < nsegments; i++)
{
    int edge1 = lup[icase][2*i];
    float pt1[2] = // Interpolate position along edge1
    int edge2 = lup[icase][2*i+1];
    float pt2[2] = // Interpolate position along edge2
    AddLineSegmentToOutput(pt1, pt2);
}
```
Upcoming

• Review 2D
• Extensions for 3D
• More discussion of isosurface calculation