Visualizing Smoke and Fire

Goal: Improve quality and efficiency of methods used to visualize smoke and fire
Overview

- Smoke/Fire Visualization Examples
- Brief overview of new visualization algorithms
- Exploit the GPU (video card) to perform computations more efficiently
- Making movies using ffmpeg
Challenges

• Memory
• Computation
• Data load time

Time sinks

• FDS – multiplications
• Smokeview – drawing triangles

Solution Approaches

• Compress data
• Use the video card (GPU)
• Load data in the background (while it is being displayed)
• Display only data that is visible

100x100x60
400 meshes
800+ time frames
240 million grid cells
192 GB data
Smoke Visualization Methods

- particles
- 3D contours
- realistic/3D smoke
- 2D contours
Light/Smoke interactions

Volume Rendering Equation – Radiation Transport Equation

\[
(\omega \cdot \nabla) L(x, \omega) = -\sigma_a(x)L(x, \omega) - \sigma_s(x)L(x, \omega) + \sigma_a(x)L_e(x, \omega) + \sigma_s(x) \int \frac{p(x, \omega, \omega')L_t(x, \omega')d\omega'}{4\pi}
\]

\[
\frac{dL(x)}{dx} = -\sigma_t(x)L(x)
\]

\[
\frac{L(x)}{L_0} = e^{-\sigma_t x}
\]

Beer’s law
Orient planes to be **perpendicular** to line of sight

Beer’s law

$$\frac{l}{l_0} = \exp(-ks\Delta x)$$
3D Slices
(like 3d smoke/fire uses 3d interpolation)

- FDS input file
  
  &DUMP NFRAMES=100, DT_SL3D=0.1 /
  &SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2, QUANTITY='TEMPERATURE' /

keyboard shortcut: w
3D Slices
Overview of Smoke/Fire Visualization Method

- Intersect a series of equally spaced planes with each mesh
- Generate triangles in each plane
- Obtain smoke and fire data at each triangle vertex
- Draw each triangle using smoke and fire data to generate opacity and color

Assign color and opacity to each vertex
Overview of Smoke/Fire Visualization Method
Smoke/Fire Visualization Using ‘New’ Triangulation Method
Compress Data – Run Length Encoding

• Replace four byte soot density with one byte opacity
  Compress using “run length encoding”
• Replace repeated runs with a count and a data value

0000222223333344  →  #40#52#5344

• This step is performed automatically by FDS when outputting 3D smoke files
Fire Visualization Using Slice files – max blending method

- Replace color only if it is ‘greater’ than current color in screen buffer

FDS input file
&DUMP NFRAMES=1000, DT_SL3D=0.1 /
&SLCF XB=…. QUANTITY='TEMPERATURE' /

Smokeview
- Select ‘slice fire’ options
- Select ‘fire 3’ color bar
Max Blending Method - Examples

Color drawn
- 192,192,192
- 64,64,64

Current screen buffer
- 128,128,128
- 128,128,128

Updated screen buffer
- 192,192,192
- 128,128,128
Compress Data – Smokezip

- Use Smokezip for 3D slice files (max blending example)
  Smokezip uses the ZLIB library for compression [https://zlib.net](https://zlib.net)

- Open case in smokeview and define min and max slice temperature
- Save a .ini file
- Run smokezip

  smokezip –t n casename

Set n simultaneous processes you want to run
Max Blending Example

Time: 0.02
Making Movies

- Download ffmpeg and ffmpegplay from: https://www.ffmpeg.org/download.html
- Smokeview adds a movie dialog box if it finds ffmpeg in your path
Normal view – one screen

- Objects between the eye and the screen are projected onto the screen
Making Movies

360 rendering – use 8 of 26 views

- Compute azimuth and elevation of each pixel in rectangular view
- Find pixel with same azimuth and elevation (or close) in one of the 26 views

rectangular view
Making Movies

360 rendering – use 26 views to ‘flatten’ the sphere
Making Movies

360 rendering – use 26 views to ‘flatten’ the sphere
Making Movies
Future Possibilities

- Include more terms from the RTE
- Use color based on flame temperature
- Improve integration of the RTE
- Make better use of the GPU
Thank You and Questions