Putting the Smoke into Smokeview

Recent Updates and Enhancements

Glenn P. Forney
Fire and Evacuation Modeling
Technical Conference
August 15, 2011

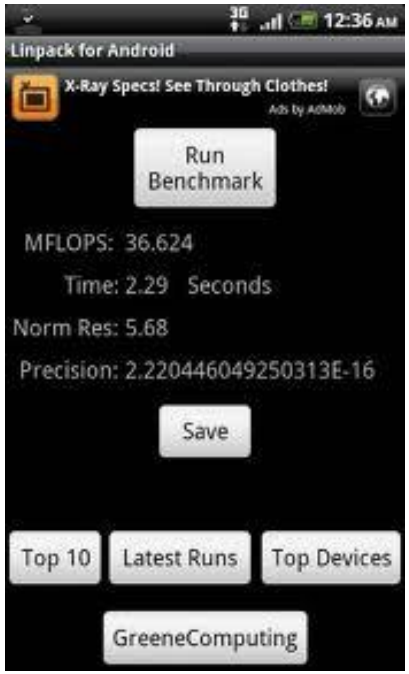
NIST
National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

Overview

- Improve algorithms for visualizing smoke
- Exploit the GPU (video card) to perform computations more efficiently
- Make better use of color for examining FDS results

Problem or Opportunity?

What do we do with all the numbers?
What can we do with all the numbers!



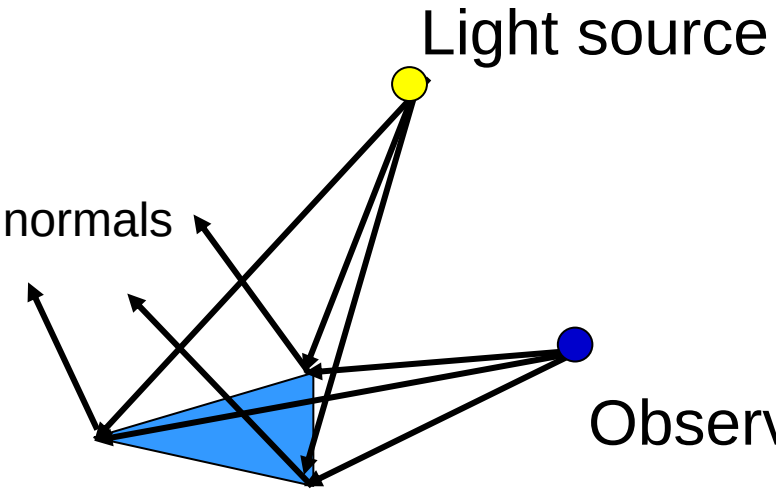
Cray 1
12 Mflops

Android smart phone
36 Mflops

NIST Fire Cluster
2240 Gflops

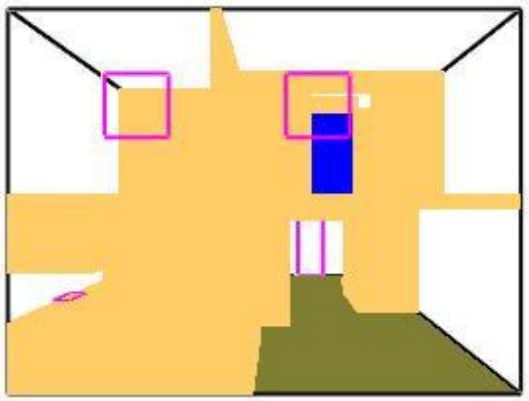
186,000 speedup!!

Lighting/Shading

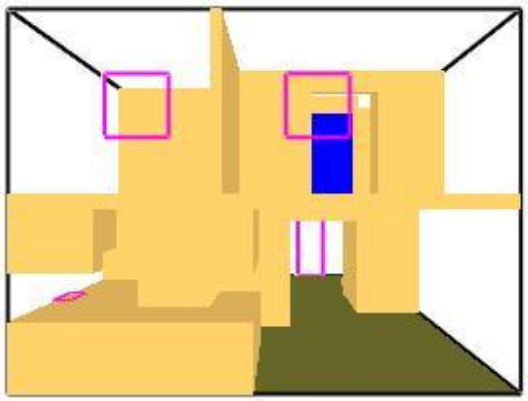


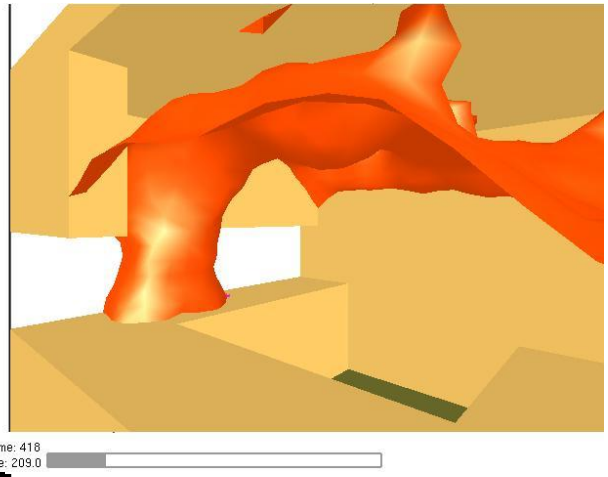
Lighting

- Adds more realism to 3D scenes
- Computed using normal vectors light source direction vectors



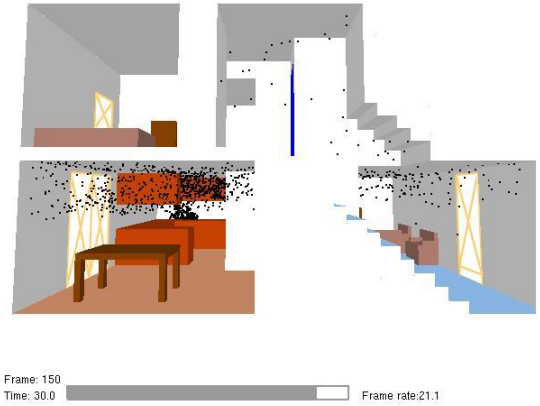
Unlit



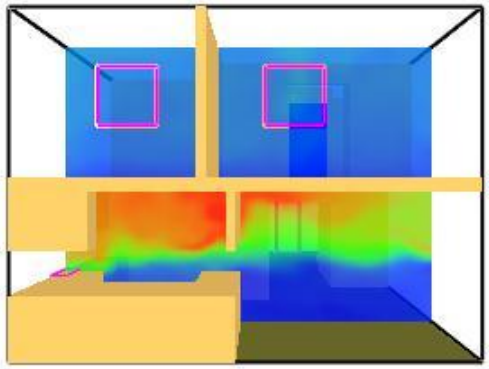


Lit

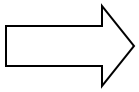
Evolution of Smoke Visualization Methods

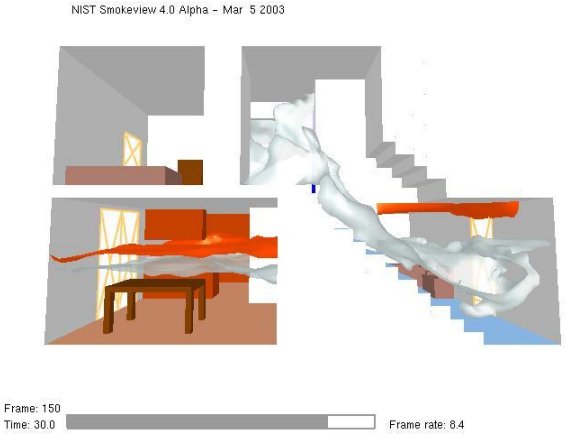


tracer particles

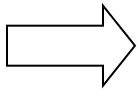


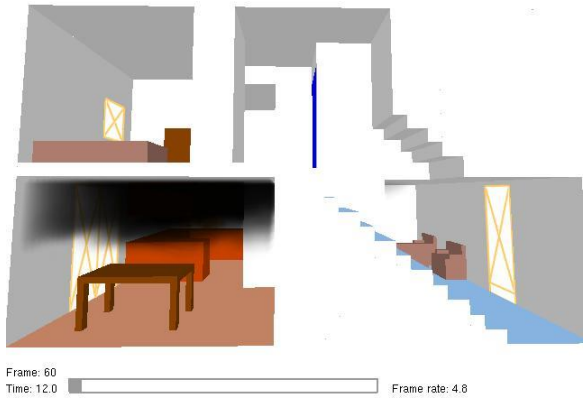
2d contours





3d contours





realistic/3D smoke

Light/Smoke interactions

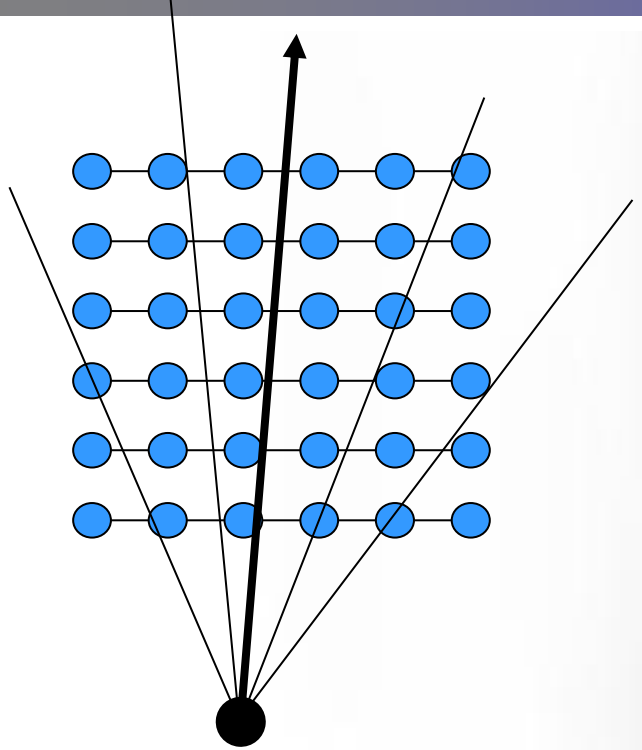
absorption out-scattering emission in-scattering

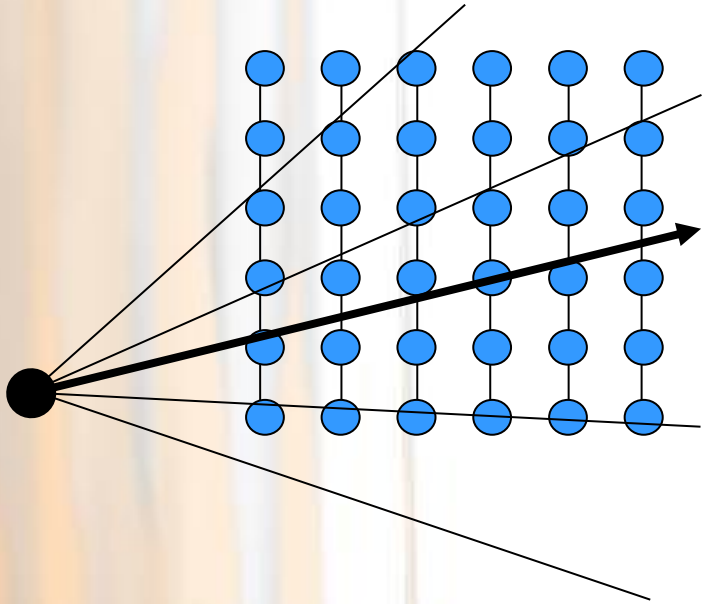
Volume Rendering Equation – Radiation Transport Equation

$$(\omega \cdot \nabla) L(x,\omega) = -\sigma_a(x) L(x,\omega) - \sigma_s(x) L(x,\omega) + \sigma_a(x) L_e(x,\omega) + \sigma_s(x) \int_{4\pi} p(x,\omega,\omega') L_i(x,\omega') d\omega'$$

$$\frac{dL(x)}{dx} = -\sigma_t(x) L(x) \quad \Longrightarrow \quad \frac{L(x)}{L_0} = e^{-\sigma_t x}$$

Beer's law

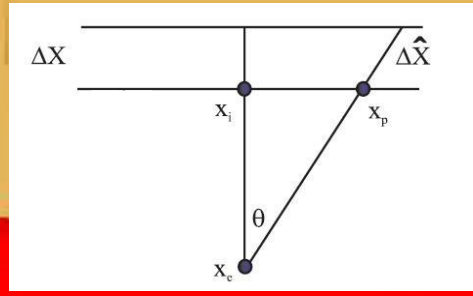




Beer's law
$I/I_0=\exp(-k_s\Delta x)$

Orient planes to be most perpendicular to line of sight

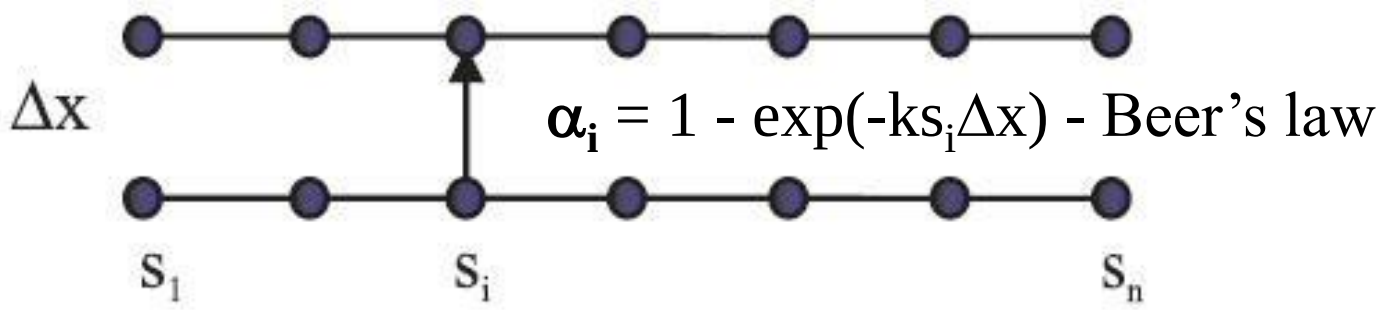
ΔX



3D Smoke

Δx	distance between adjacent grid planes
S_i	soot density
α_i	opacity

- FDS computes α for each grid node
- Smokeview combines α's using the video card



3D Smoke

Correcting α

FDS computed:

$$\alpha = 1 - \exp(-ks\Delta x)$$

Smokeview computed:

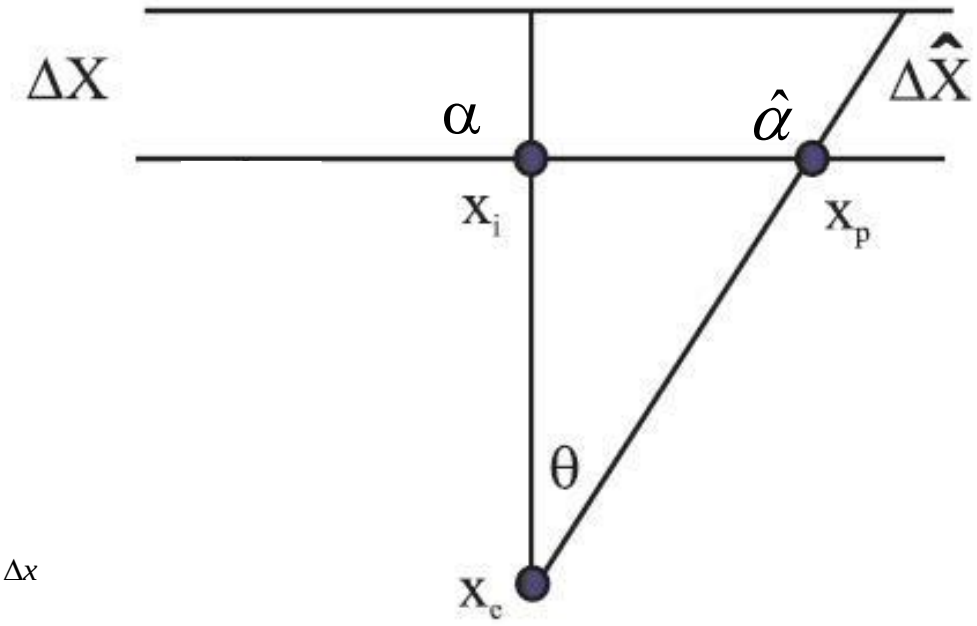
$$\hat{\alpha} = 1 - \exp(-ks\Delta \hat{x})$$

Solve for exp(-ks):

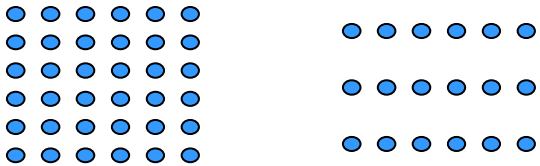
$$(1-\hat{\alpha})^{1/\Delta\hat{x}} = \exp(-ks) = (1-\alpha)^{1/\Delta x}$$

Solve for $\hat{\alpha}$

$$\boxed{\hat{\alpha} = 1 - (1-\alpha)^{\Delta\hat{x}/\Delta x}}$$



Skipping Frames

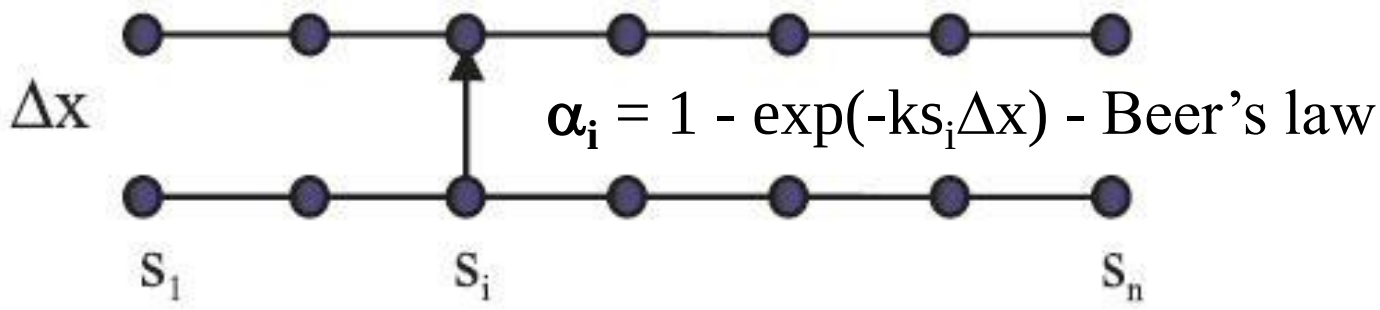


$$\Delta\hat{X}/\Delta X = 2 \qquad \hat{\alpha} = 2\alpha - \alpha^2$$

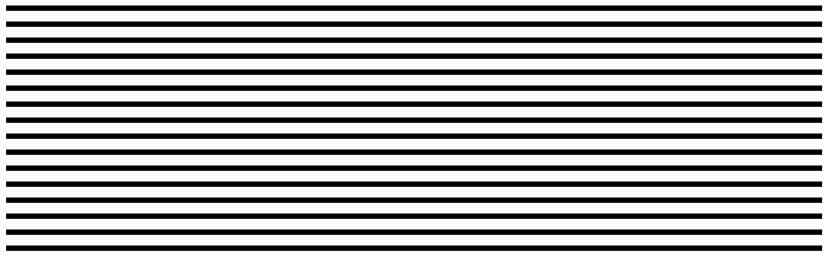
NIST
National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

3D Smoke

Problems can occur for large grids (ΔX small)



More refined grids ➔ smaller Δx ➔ smaller α ➔ increased error



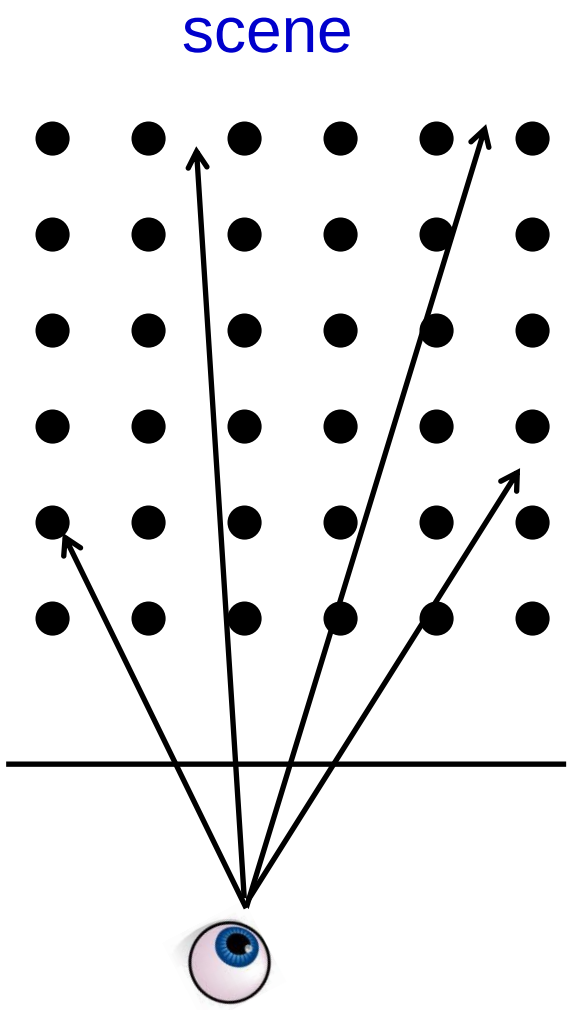
3D Smoke

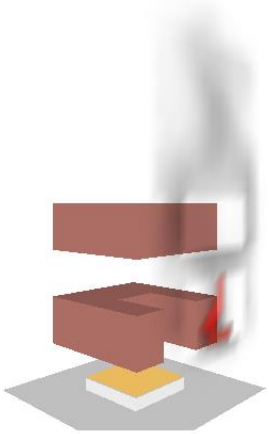
Solution: Integrate across entire mesh rather than one grid slice

$$\alpha = 1 - e^{-\int ksdx}$$

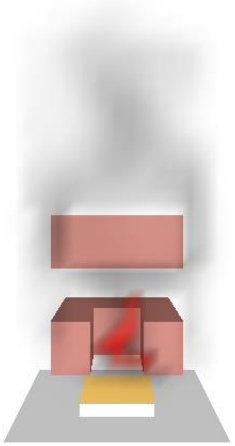
- Use 3D slice files (full precision)
- Compute a line integral for each pixel using the video card (GPU)

- Color – use a transfer function to map temperature to color
- Opacity - integrate soot densities

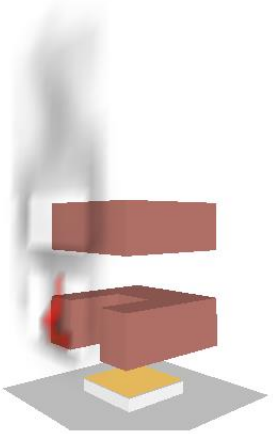




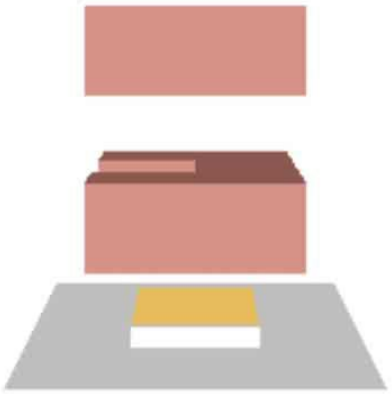
Time: 14.8



Time: 14.8

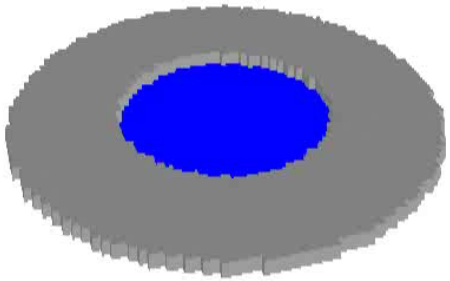


Time: 14.8



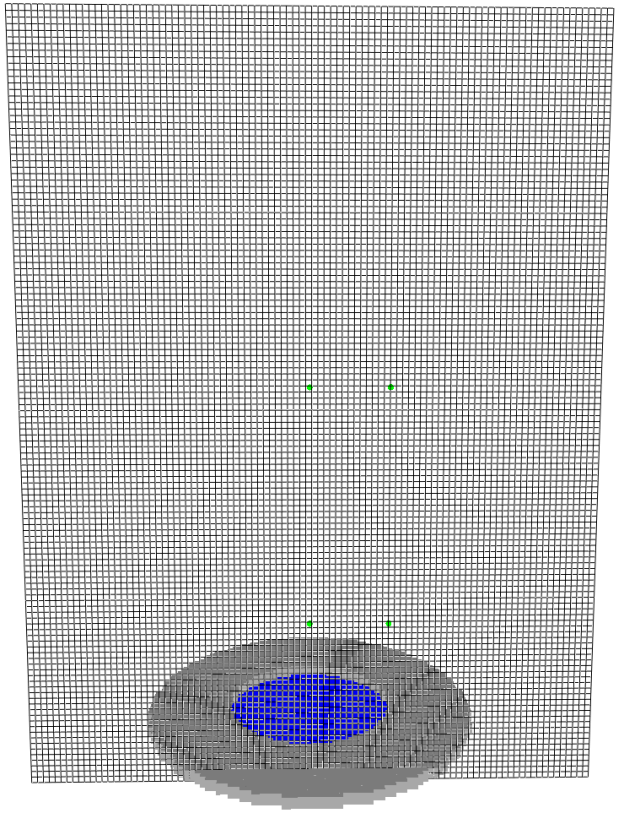
Time: 0.0

3D Smoke



Time: 0.05

96x96x64
16 meshes
300 frames
21 GB
1-3 minutes load time



Challenges

- Memory
- Computation
- Data load time

Solution Approaches

- compress data
- use the video card (GPU)
- load data in the background (while it is being displayed)
- load data as it is required
- Treat Smokeview more like FDS

Using the New Method

- Add following keyword to the &DUMP line

  DT_SL3D=xxx

  (where xxx is desired output time step)

- Add following &SLCF lines to case

  &SLCF XB=…, QUANTITY='TEMPERATURE‘, CELL_CENTERED=.TRUE. /

  &SLCF XB=…, QUANTITY='DENSITY',SPEC_ID='SOOT',CELL_CENTERED=.TRUE. /

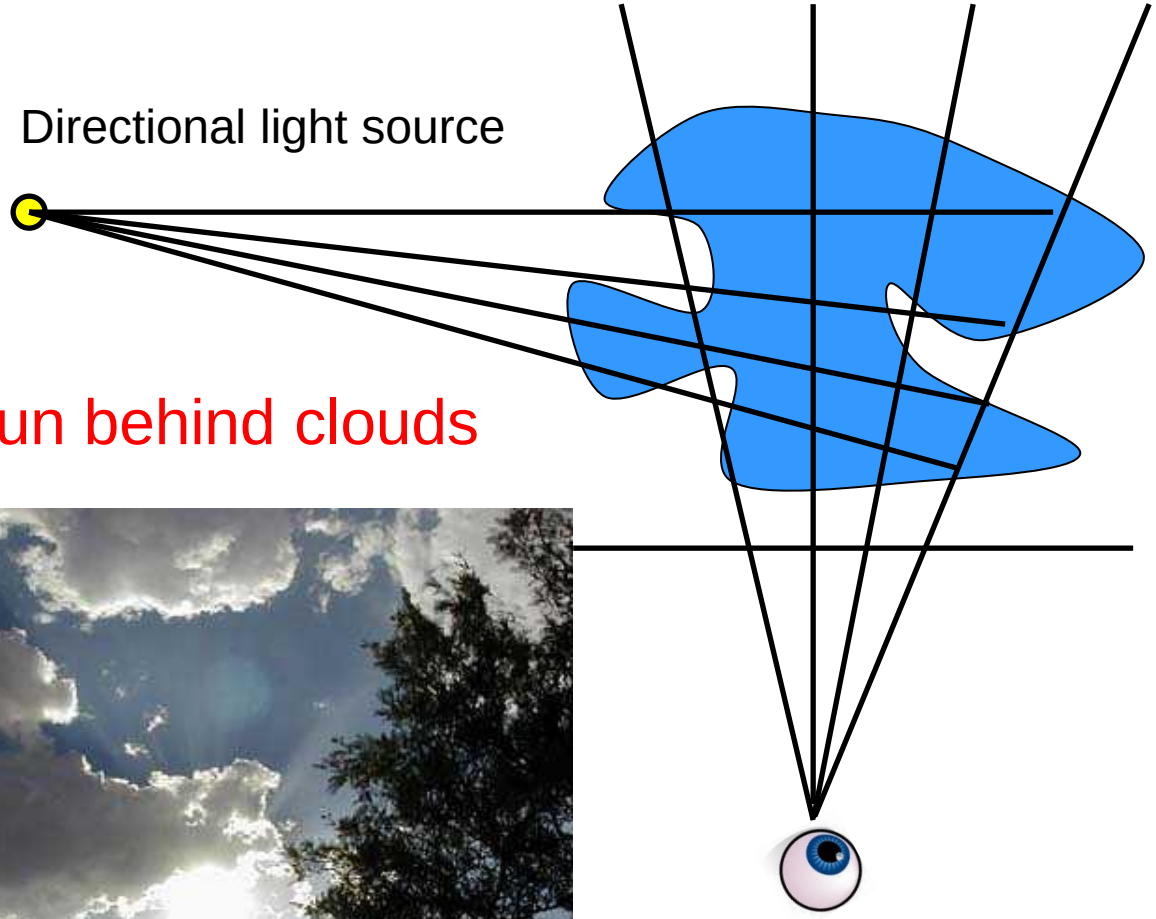
  where “…” are the bounds for your simulation domain

- In Smokeview load file type labeled “3D smoke (volume rendered)”
- In the repository look at cases in Verification/Visualization: plume5c, mplume8, vis_test1, vis_test2 or vis_test3

Future Possibilities

Include more terms in the Radiation Transport Equation

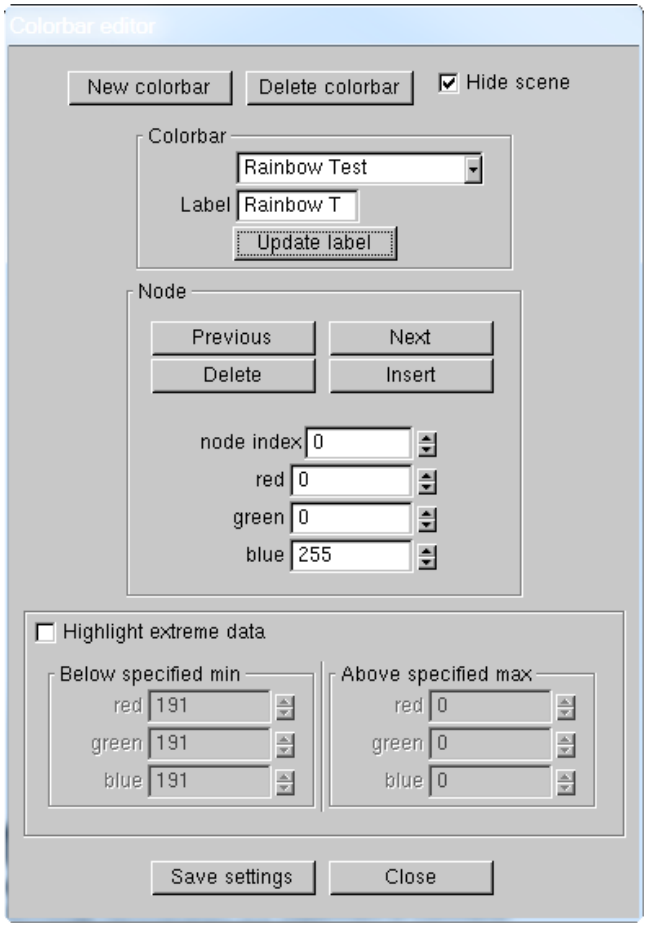
$$(\omega \cdot \nabla) L(x,\omega) = -\sigma_a(x) L(x,\omega) - \sigma_s(x) L(x,\omega) + \sigma_a(x) L_e(x,\omega) + \sigma_s(x) \int_{4\pi} p(x,\omega,\omega') L_i(x,\omega') d\omega'$$

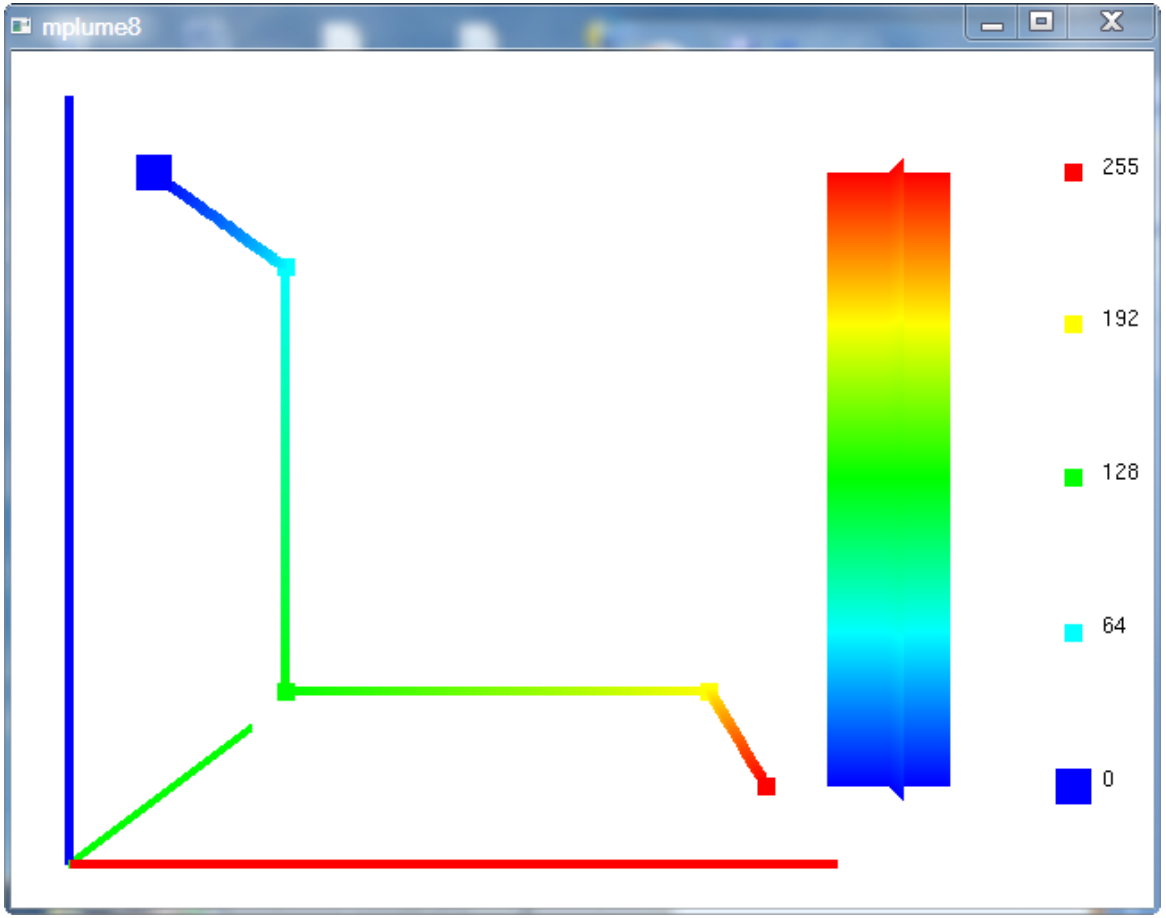


Sun behind clouds

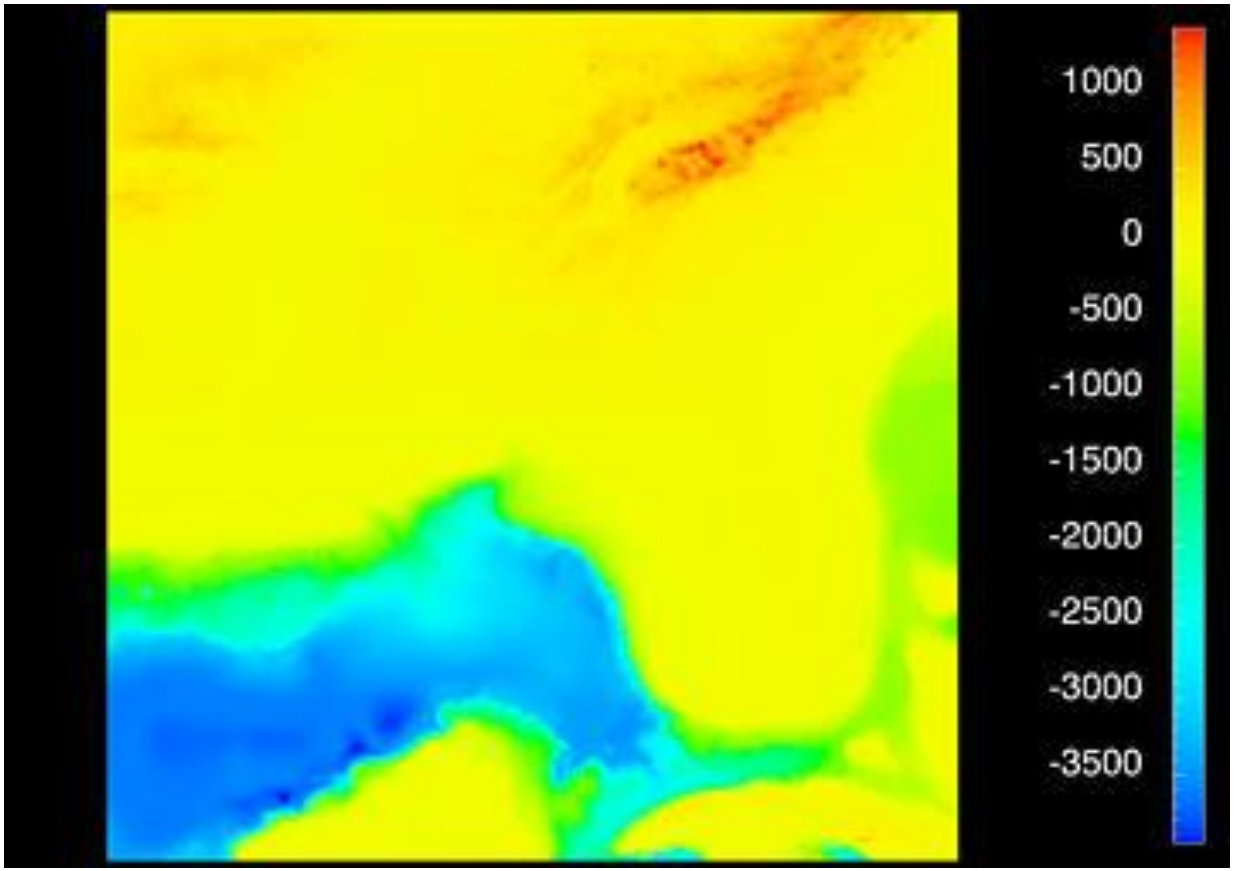
Diffuse/Ambient Light

Coloring Data

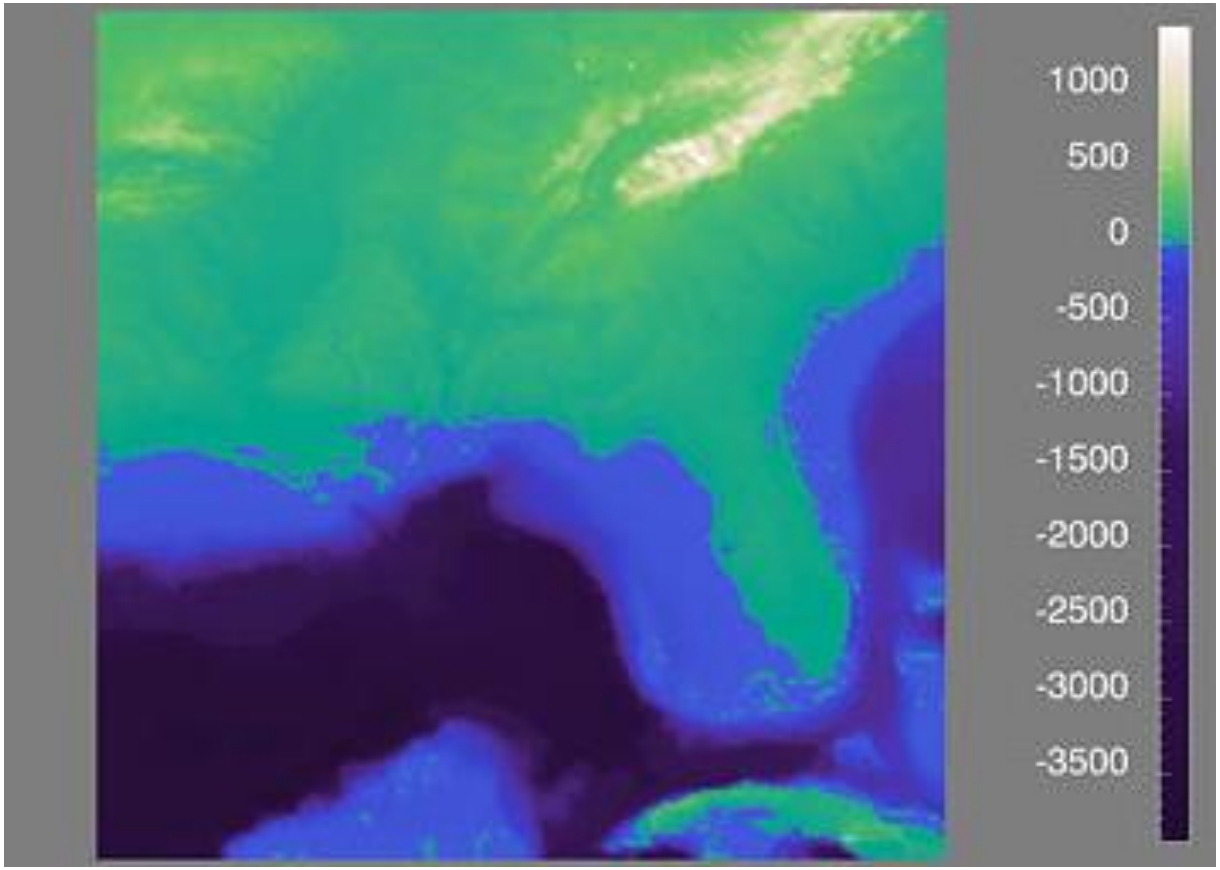




Rainbow Colorbar



Discontinous Colorbar



Summary

- Improve algorithms for visualizing smoke
- Exploit the GPU (video card) to perform computations more efficiently
- Make better use of color to examine FDS data

New Smokeview Capabilities

- New more flexible method for visualizing smoke
- Large cases (and small) may be visualized in “batch” mode one frame at a time
- Smokeview may be run remotely (using Linux command Xvfb)
- Smoke may be visualized immediately while the data is loading
- Smoke may be colored arbitrarily with user specified colorbars