

NIST Special Publication 1017-3

**Smokeview (Version 6)
A Tool for Visualizing
Fire Dynamics Simulation Data
Volume III: Verification Guide**

Glenn P. Forney

<http://dx.doi.org/10.6028/NIST.SP.1017-3>

NIST Special Publication 1017-3

Smokeview (Version 6)
A Tool for Visualizing
Fire Dynamics Simulation Data
Volume III: Verification Guide

Glenn P. Forney
Fire Research Division
Engineering Laboratory

<http://dx.doi.org/10.6028/NIST.SP.1017-3>

August 2013
Smokeview Version 6
SVN Repository Revision : 16665



U.S. Department of Commerce
Penny Pritzker, Secretary

National Institute of Standards and Technology
Patrick D. Gallagher, Under Secretary of Commerce for Standards and Technology and Director

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

Preface

This guide is part of a three volume set of companion documents describing how to use Smokeview in Volume I, the Smokeview User's Guide [1], describing technical details of how the visualizations are performed in Volume II, the Smokeview Technical Reference Guide [2], and presents example cases verifying the various visualization capabilities of Smokeview in Volume III, the Smokeview Verification Guide [3]. Details on the use and technical background of the Fire Dynamics Simulator is contained in the FDS User's [4] and Technical reference guide [5] respectively. This guide is Volume III the Smokeview Verification guide.

Smokeview is a software tool designed to visualize numerical calculations generated by fire models such as the Fire Dynamics Simulator (FDS), a computational fluid dynamics (CFD) model of fire-driven fluid flow or the Consolidated Fire and Smokeview Transport (CFAST) model, a zone fire model. This Guide is Volume 3 of the Smokeview Reference Guides. This guide presents a series of images derived from FDS and Smokeview. The intent is to verify that the algorithms used by Smokeview for visualizing data are implemented correctly. These images are generated automatically through the use of scripts by first running FDS on a series of input cases and then running Smokeview, again using a set of scripts. The correctness of Smokeview may then be verified more easily as FDS and Smokeview are updated since the reference figures in this document may be generated simply and automatically.

Smokeview and associated documentation for Windows, Linux and Mac OS X may be downloaded from <http://fire.nist.gov/fds>.

About the Author

Glenn Forney is a computer scientist at the Engineering Laboratory of NIST. He received a bachelor of science degree in mathematics from Salisbury State College and a master of science and a doctorate in mathematics from Clemson University. He joined NIST in 1986 (then the National Bureau of Standards) and has since worked on developing tools that provide a better understanding of fire phenomena, most notably Smokeview, a software tool for visualizing Fire Dynamics Simulator data.

Disclaimer

The US Department of Commerce makes no warranty, expressed or implied, to users of Smokeview, and accepts no responsibility for its use. Users of Smokeview assume sole responsibility under Federal law for determining the appropriateness of its use in any particular application; for any conclusions drawn from the results of its use; and for any actions taken or not taken as a result of analysis performed using this tools.

Smokeview and the companion program FDS is intended for use only by those competent in the fields of fluid dynamics, thermodynamics, combustion, and heat transfer, and is intended only to supplement the informed judgment of the qualified user. These software packages may or may not have predictive capability when applied to a specific set of factual circumstances. Lack of accurate predictions could lead to erroneous conclusions with regard to fire safety. All results should be evaluated by an informed user.

Throughout this document, the mention of computer hardware or commercial software does not constitute endorsement by NIST, nor does it indicate that the products are necessarily those best suited for the intended purpose.

Contents

Preface	i
About the Author	iii
Disclaimer	v
1 Overview	1
2 Data File Tests	3
2.1 Surface contours (Boundary Files)	5
2.2 Iso-surfaces	7
2.2.1 Sensitivity Analysis	10
2.3 Particles	12
2.4 Slices	14
2.5 3D Smoke	22
2.6 Plot3D	23
3 Smoke Visualization Tests	29
3.1 Recording Smoke Levels	29
3.2 Verifying Smoke Levels	30
4 WUI Test Cases	35
4.1 FDS Cases	35
4.2 WFDS Cases	42
5 Other Tests	47
5.1 Obstacles	47
5.2 Devices	49
5.3 Vents	50
5.4 Conversion to Color	53
5.5 GPU Test	56
Bibliography	59
References	60
Appendices	60
A Input Files	61
A.1 colorbar	61

A.2	colorconv	61
A.3	fed_test	62
A.4	plume5c	62
A.5	sillytexture	64
A.6	smoke_sensor	65
A.7	smoke_test	65
A.8	smoke_test2	66
A.9	transparency	67
A.10	tree_test2	67
A.11	vcirctest	68
A.12	vcirctest2	69
B	Smokeview Scripts	71
B.1	colorbar	71
B.2	colorconv	71
B.3	fed_test	73
B.4	plume5c	73
B.5	sillytexture	84
B.6	smoke_sensor	84
B.7	smoke_test	84
B.8	smoke_test2	85
B.9	sprinkler_many	85
B.10	transparency	86
B.11	tree_test2	86
B.12	vcirctest	86
B.13	vcirctest2	87

List of Figures

2.1	The temperatures in the plume5c.fds case are initialized to 600 °C within the region outlined in blue.	4
2.2	Boundary file test of shaded and chopped contours	6
2.3	A test showing an isosurface enclosing a region initialized to 600 °C	8
2.4	A test showing isosurfaces colored orange and blue using the ISOCOLORS .ini keyword.	9
2.5	A portion of an iso-surface defined by $f(x,y,z) = T_m$ (for some arbitrary function f) crossing a line segment at the point (x_m, y, z)	10
2.6	Diagram setting up an inverse linear interpolation calculation	11
2.7	A test showing particles and particle streaks	13
2.8	A test showing shaded and chopped contours of node centered slice file data	15
2.9	A test showing shaded and chopped contours of cell centered slice file data	16
2.10	A test to demonstrate that cell centered slice file data is transferred from FDS to Smokeview properly.	17
2.11	A test showing vector slice file data	18
2.12	A test showing cell centered FED slice file data	19
2.13	A test showing arbitrarily oriented slice file planes generated from 3D slice files	20
2.14	A test showing slice contours generated by differencing two cases with Smokediff	21
2.15	A test showing 3D smoke drawn at 1 s, 10 s and 30 s	22
2.16	A test showing PLOT3D stepped and continuous temperature contours.	23
2.17	A test showing PLOT3D stepped contours at four positions along the y axis.	24
2.18	A test showing PLOT3D narrow band contours at four positions along the y axis.	25
2.19	A test showing PLOT3D vectors shaded by temperature at four positions along the y axis.	26
2.20	A test showing PLOT3D iso-surfaces for 50 °C, 230 °C, 410 °C and 590 °C.	27
3.1	A test verifying that the Smokeview smoke_sensor object is working properly	30
3.2	Shade of grey resolution test.	31
3.3	Side view of numerical smoke test compartment.	32
3.4	A test verifying the smoke opacity calculation	33
4.1	A test showing level set slices drawn along a sloped terrain	36
4.2	A test showing a fire line slice drawn along a sloped and textured terrain	37
4.3	A test showing a fire line slice drawn along a sloped and textured terrain at 1800 s simulated time.	38
4.4	A test showing level temperature slices drawn along a sloped terrain	39
4.5	A test showing wind vectors drawn using data obtained from a wind sensor.	41
4.6	A test showing trees drawn as Smokeview objects.	42
4.7	A test showing four states of smokeview tree objects.	42
4.8	A test showing trees drawn as particles.	43
4.9	A test showing trees drawn as isosurfaces.	44

4.10	A test showing particles drawn as canopy objects.	45
5.1	A test showing 3 drawing modes for representing blockages: solid, outline and hidden	48
5.2	A test showing that transparent objects are drawn properly	48
5.3	A test showing that devices are drawn properly	49
5.4	A test showing 3 modes for drawing vents: show all vents, hide open vents and hide all vents	50
5.5	A test showing circular vents drawn as specified in the input file and as implemented within FDS. The circular vents are applied to a blockage.	51
5.6	A test showing circular vents drawn as specified in the input file and as implemented within FDS. The circular vents are applied to the external boundary.	52
5.7	A test showing that data is converted to color properly	54
5.8	A test showing that data is converted to color properly	55
5.9	A test showing that the GPU is properly used to draw smoke	57

Chapter 1

Overview

Smokeyview is a scientific software tool designed to visualize numerical predictions generated by fire models such as the Fire Dynamics Simulator (FDS), a computational fluid dynamics (CFD) model of fire-driven fluid flow [5, 4] or the Consolidated Fire and Smoke Transport (CFAST) model, a zone model of compartment fire phenomena [6].

The feature set and user interface for Smokeyview is complex making it difficult to adequately test all of its features manually. A scripting capability has been added to solve this problem. Many of Smokeyview's features may now be run without user intervention through the use of scripts. A script is simply a text file containing one or more commands. Smokeyview may then read the script commands and perform the actions listed. Some of these actions are loading data files, setting view points, setting times and most importantly rendering images. By designing a set of scenarios and corresponding images that demonstrate Smokeyview's feature set, one may test Smokeyview running a series of scripts (one script for each test case) and examining the resulting images to ensure that the fire model generating the results and Smokeyview are working as expected.

This document then verifies that various Smokeyview features are working as intended by presenting a series of simulation results in the form of Smokeyview images. These images are generated using the various visualization features of Smokeyview such as tracer particles, 2D or 3D contours, or realistic smoke.

Verification in the context of Smokeyview is a process to check the correctness of the visualization. Verification does not imply that the underlying data is correct, only that the data is presented or visualized correctly. A separate document, the FDS Verification Guide [7], describes the verification process of FDS. One set of scripts is used to run FDS cases and a second set of scripts is used by Smokeyview to generate images. The verification process then becomes much easier to accomplish since the use of scripts (i.e., non-manual methods) guarantees that consistent figures (same view points, same time points, same data files loaded, etc.) are produced as new versions of this verification document are generated using updated versions of FDS and/or Smokeyview. Another way of looking at this verification process is to consider this document and the FDS Verification Guide [7] as being a *live* (not a static) document, easily updated as algorithms in FDS and/or Smokeyview are enhanced and improved.

Details on setting up and running FDS cases may be found in the FDS User's Guide [4]. Details on visualizing FDS simulated data using Smokeyview may be found in the Smokeyview User's Guide [1]. Details on some of the technical aspects used to implement algorithms in Smokeyview may be found in the Smokeyview Technical Guide [2].

The FDS version used to run the cases illustrated in this document is

Version: FDS 6, Release Candidate 4; MPI Disabled; OpenMP Disabled
SVN Revision Number: 16557
Compile Date: Mon, 12 Aug 2013

Consult FDS Users Guide Chapter, Running FDS, for further instructions.

Hit Enter to Escape...

The Smokeview version used to generate the verification figures in this document is

Smokeview 6.1.2 - Aug 13 2013

Version: 6.1.2
Smokeview (64 bit) Revision Number: 16567
Platform: WIN64 (Intel C/C++)
Build Date: Aug 13 2013

FDS generated data is presumed to be correct. FDS has its own set of verification cases to test the correctness of the data. The purpose of the cases used here is to confirm that data is drawn or visualized correctly. In particular, these cases confirm that correct files are loaded, data is scaled and drawn correctly, geometry is drawn correctly, etc. Three types of verification cases are presented. The first set are the most important. Those cases verify that data is drawn correctly. The second set of cases verify that various geometric elements are drawn correctly and the third set verifies that the various options and underlying features are implemented and perform properly.

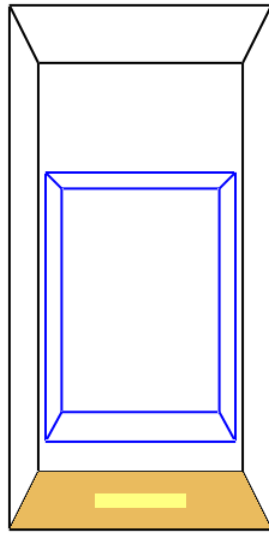
The FDS input files used for the verification cases are documented in [Appendix A](#). The Smokeview scripts used to generate the verification figures are documented in [Appendix B](#). Note that these input files and scripts are located in the FDS Subversion (SVN) repository. In fact, the entries in the appendices are included directly from the repository, and will therefore be up to date as this document is regenerated.

Chapter 2

Data File Tests

The tests in this chapter verify whether visualization types such as surface contours (boundary files), isosurfaces, particles, slice files, 3D smoke files and PLOT3D files are working as intended. These verification tests all use the same FDS input file, `plume5c.fds` (see Appendix A.4), to generate the simulation data. The case models a simple fire plume with two blockages. The upper blockage is initialized to 600 °C. The lower blockage is initialized to 20 °C. The gas phase is initialized to 600 °C within an interior region colored blue as illustrated in Fig. 2.1 and 20 °C everywhere else. This is done in order to verify that a known temperature is converted to the proper color (as shown in the color bar).

The verification figures are generated automatically using the Smokeview script file, `plume5c.ssf` (see Appendix B.4). The use of scripting allows the figures and hence this document to be updated easily as changes are made in FDS, Smokeview or the FDS input data files. This allows the verification process to be ongoing.



mesh: 1

Figure 2.1: The temperatures in the plume5c.fds case are initialized to 600 °C within the region outlined in blue.

2.1 Surface contours (Boundary Files)

Figure 2.2 presents images verifying the display of surface contours or boundary file data. A series of boundary file images are drawn at 2 s (approximately), 10 s and 30 s seconds. The temperature of the upper obstacle is initialized to 600 °C hence the red colors for the $t = 0$ s images. The first column of images colors data at cell nodes using temperatures averaged at surrounding cell centers. The second column of images colors data using data values at cell centers.

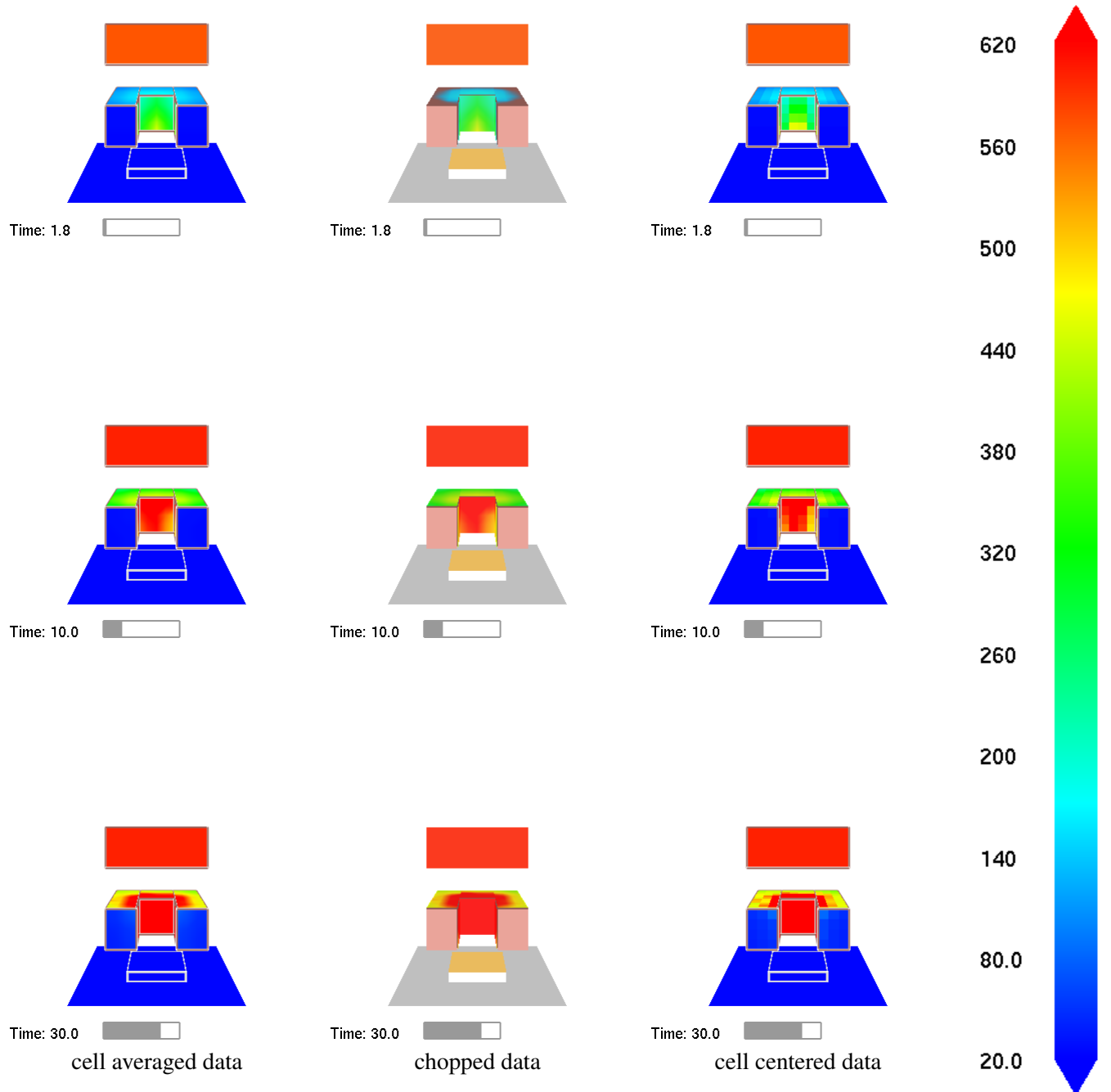


Figure 2.2: Boundary file test of shaded and chopped contours at 2 s, 10 s and 30 s. The upper obstacle is initialized to 600 °C and should be red for the $t \approx 2$ s images. For the chopped contours, the color near the chop boundary should match the color near 140 °C in the colorbar.

2.2 Iso-surfaces

An isosurface is a three-dimensional surface that defines constant values of a dependent variable. Figures 2.3 and 2.4 present images verifying the display of isosurfaces. A series of temperature isosurfaces are drawn at 0 s, 10 s and 30 s. A portion of the interior gas temperature is initialized to 600 °C, hence. This is the reason that the rectangular block appears in the first row of images. The first column in Fig. 2.3 presents the iso-surface using points. The second column presents the iso-surface using triangulated outlines. The third column presents the iso-surface using a solid surface. Figure 2.4 demonstrates color customization using the `ISOCOLORS .ini` keyword.

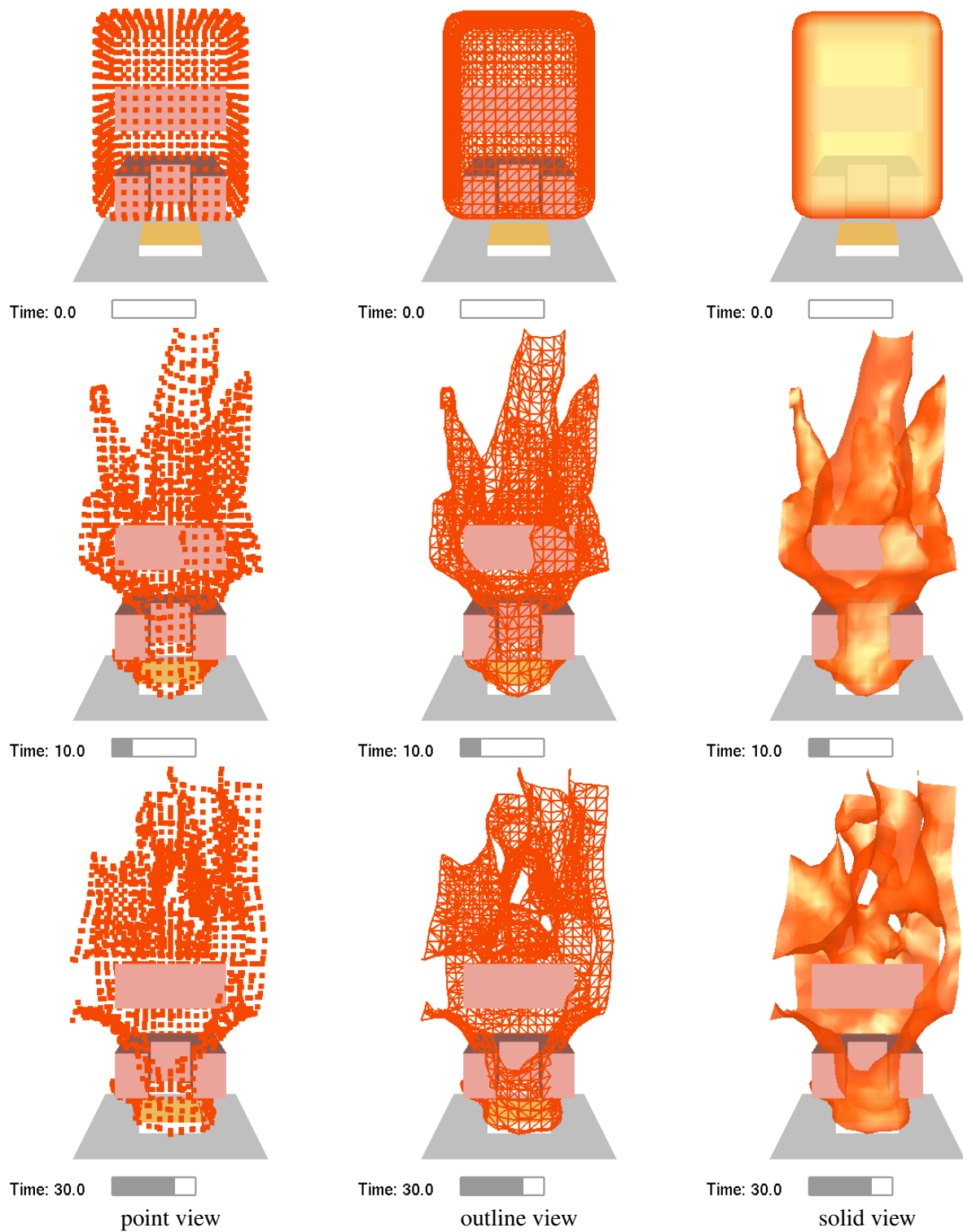


Figure 2.3: A test showing an isosurface enclosing a region initialized to 600 °C at 0 s, 10 s and 30 s. The isosurface should surround this region for the $t = 0$ s images.

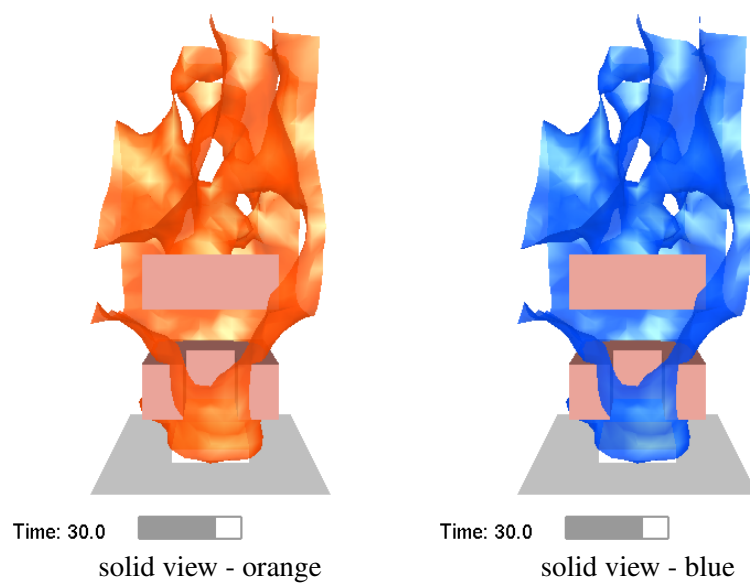


Figure 2.4: A test showing isosurfaces colored orange and blue using the ISOCOLORS .ini keyword at 30 s. The two isosurfaces are colored orange and blue using the ISOCOLORS .ini keyword.

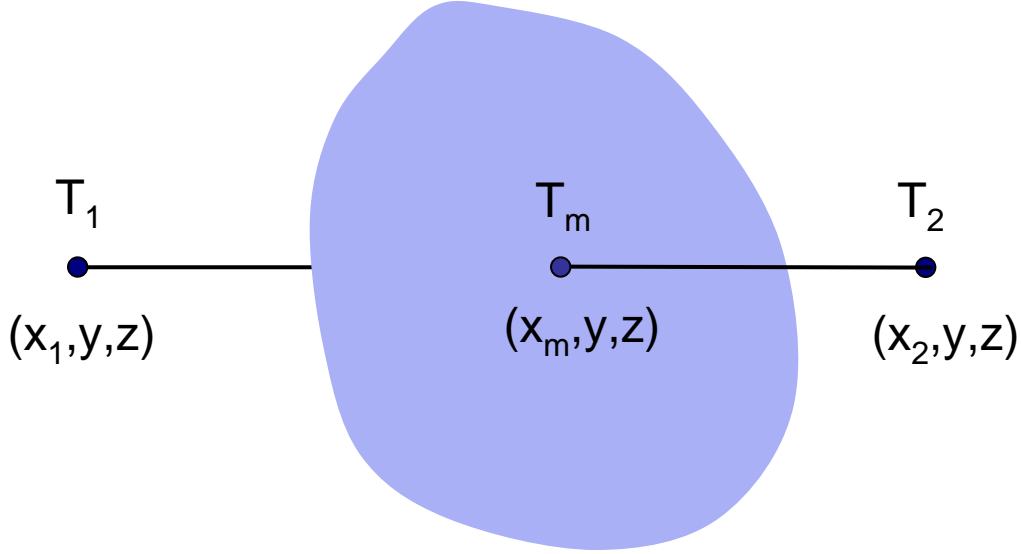


Figure 2.5: A portion of an iso-surface defined by $f(x, y, z) = T_m$ (for some arbitrary function f) crossing a line segment at the point (x_m, y, z) .

2.2.1 Sensitivity Analysis

Given that data used to generate an isosurface has uncertainty, an important question to consider is how sensitive is the isosurface location to uncertainty in the data used to define it? Figure 2.5 shows a portion of an iso-surface, $f(x, y, z) = T_m$, passing through a line segment at (x_m, y, z) . The line segment is defined by endpoints (x_1, y, z) and (x_2, y, z) . The key step in constructing an isosurface is solving an inverse interpolation problem. That is, determining the location, (x_m, y, z) , between two grid nodes, (x_1, y, z) and (x_2, y, z) , where interpolated data takes on a particular value (the isosurface level, T_m , being constructed).

Suppose, as illustrated in Fig. 2.6, that (x_1, y, z) , T_1 and (x_2, y, z) , T_2 represent two known data location, data value pairs and that T_m is also known satisfying $T_1 \leq T_m \leq T_2$. The inverse interpolation problem then is to find the location (x_m, y, z) that takes on the data value T_m . The location x_m is given by

$$x_m = (1 - \alpha)x_1 + \alpha x_2 \quad (2.1)$$

where

$$\alpha = \frac{T_m - T_1}{T_2 - T_1} \quad (2.2)$$

The sensitivity of x_m due to a change ΔT_1 in T_1 and to a change ΔT_2 in T_2 is given by

$$\Delta x_m = \frac{\partial x_m}{\partial T_1} \Delta T_1 + \frac{\partial x_m}{\partial T_2} \Delta T_2 \quad (2.3)$$

where

$$\frac{\partial x_m}{\partial T_1} = \frac{\partial x_m}{\partial \alpha} \frac{\partial \alpha}{\partial T_1} \quad (2.4)$$

$$\frac{\partial x_m}{\partial T_2} = \frac{\partial x_m}{\partial \alpha} \frac{\partial \alpha}{\partial T_2} \quad (2.5)$$

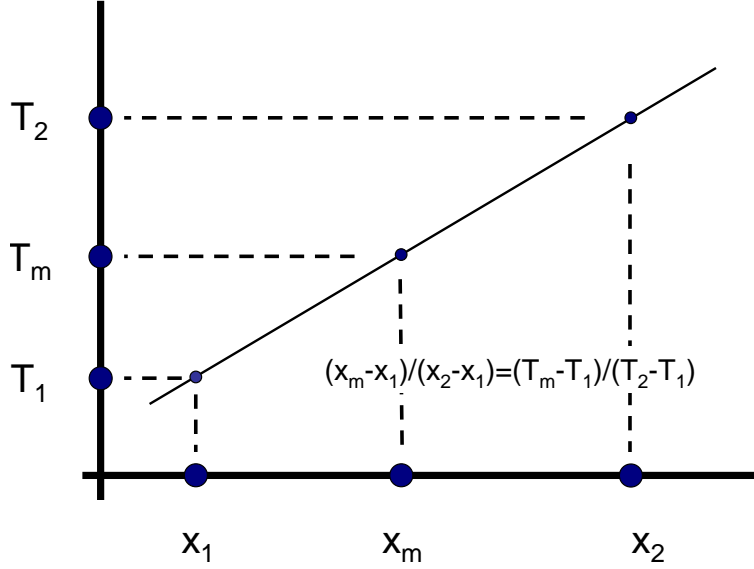


Figure 2.6: Diagram setting up an inverse linear interpolation calculation. The dependent data values T_1 and T_2 are known at locations x_1 and x_2 . The inverse interpolation problem is to find the location x_m that has value T_m . This is found noting that $(x_m - x_1)/(x_2 - x_1) = (T_m - T_1)/(T_2 - T_1)$.

and

$$\frac{\partial x_m}{\partial \alpha} = x_2 - x_1 \quad (2.6)$$

$$\frac{\partial \alpha}{\partial T_1} = \frac{T_m - T_2}{(T_2 - T_1)^2} \quad (2.7)$$

$$\frac{\partial \alpha}{\partial T_2} = -\frac{T_m - T_1}{(T_2 - T_1)^2} \quad (2.8)$$

Equation (2.3) may be re-written using terms in equations (2.4) through (2.8) to obtain

$$\frac{\Delta x_m}{x_2 - x_1} = \frac{T_m - T_2}{(T_2 - T_1)^2} \Delta T_1 - \frac{T_m - T_1}{(T_2 - T_1)^2} \Delta T_2 = -\left((1 - \alpha) \frac{\Delta T_1}{T_2 - T_1} + \alpha \frac{\Delta T_2}{T_2 - T_1} \right) \quad (2.9)$$

Equation (2.9) relates the relative error of x_m to the relative errors of T_1 and T_2 in terms of the interpolation parameter α . The error Δx_m may then be bounded to obtain

$$|\Delta x_m| \leq |x_2 - x_1| \frac{\max(|\Delta T_1|, |\Delta T_2|)}{|T_2 - T_1|} \quad (2.10)$$

Uncertainty in isosurface location is then proportional to the magnitude of data uncertainty, $\max(|\Delta T_1|, |\Delta T_2|)$, and inversely proportional to the data variation, $|T_2 - T_1|$.

2.3 Particles

Figure 2.7 presents images verifying the display of particles and streaks. Images are drawn at 1 s, 10 s and 30 s. The first column shows particles while the second and third columns shows streaks with duration 0.5 s and 1 s. Streaks are a good way of visualizing motion in a still image (i.e., on paper) since the streak shows a history of where the particle has been.

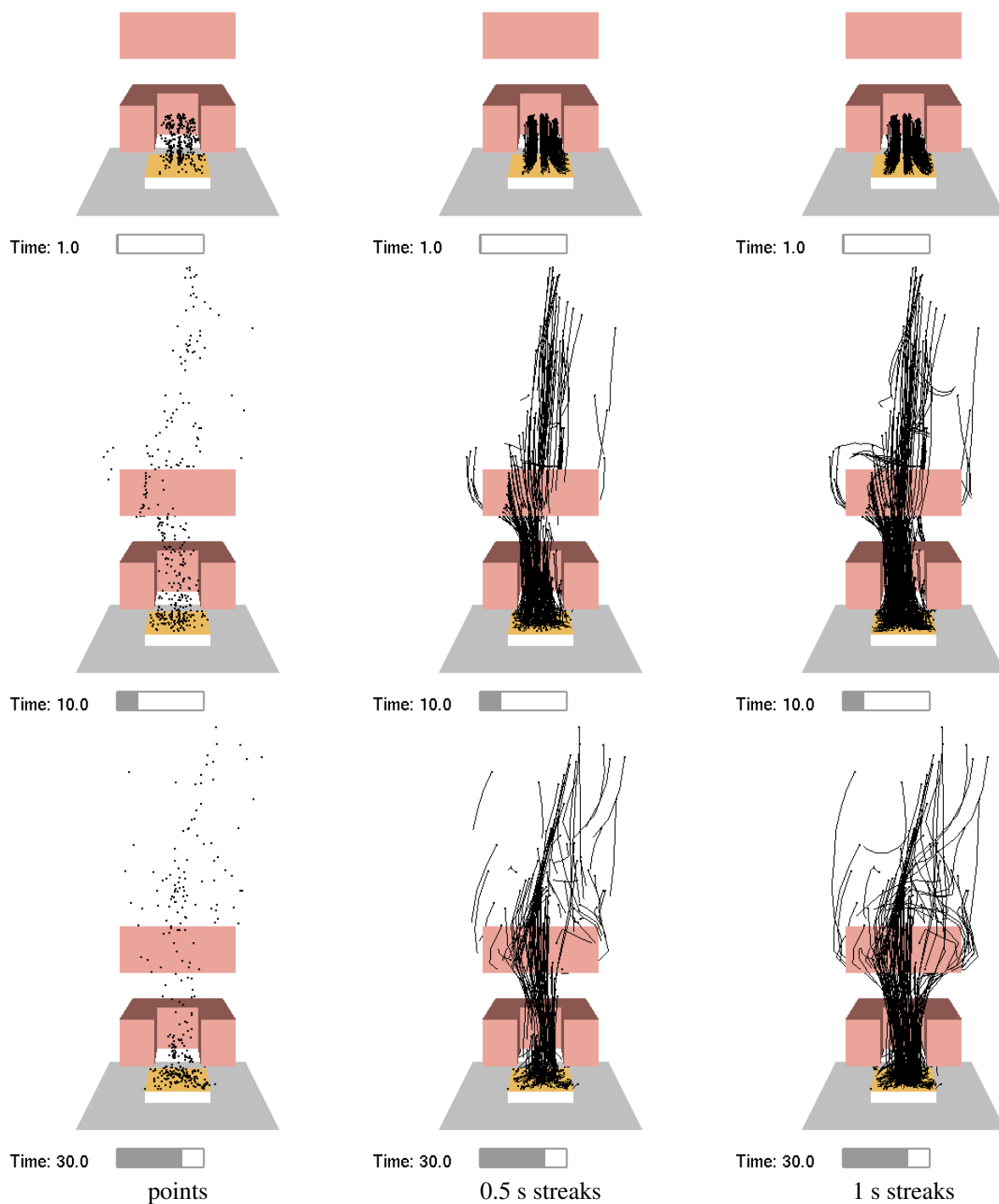


Figure 2.7: A test showing particles and particle streaks at 1 s, 10 s and 30 s

2.4 Slices

Figure 2.8 presents images verifying the display of node centered data slices. Images are drawn at 0 s, 10 s and 30 s. A portion of the interior gas temperature is initialized to 600 °C corresponding to the red rectangular block appearing in the first row. The first column visualizes all of the data in the slice. The second column discards or chops data below 140 °C. Note that the color near the chopped boundary should match the color in the color bar near 140 °C.

Figures 2.9 and 2.10 present images verifying the display of cell centered data. Images in Fig. 2.9 are drawn at 0 s, 10 s and 30 s. Images in Fig. 2.10 tests the transfer of data from FDS to Smokeview.

Figure 2.11 present images verifying the display of vector slices. Again, vector slice file images are drawn at 0 s, 10 s and 30 s. The first column draws all vectors while the second column discards or chops vectors below 140 °C.

Figure 2.12 presents slice file images verifying the computation of fractional effective dose (FED). FED is a measure of human incapacitation or hazard due to combustion gases[8]. Images are drawn at 30 s, 60 s and 120 s and 240 s. The FED is computed using three data slices with concentration of 1000 ppm for CO and volume fractions of 0.02 and 0.07 for CO₂ and O₂ respectively. These values result in FED values of 1.5 at 30 s, 3.0 at 60 s, 6.0 at 120 s and 12.0 at 240 s.

Figure 2.13 presents images of 3D slices oriented in two different ways. Images are drawn at 0 s, 10 s and 30 s.

Figure 2.14 presents images from data differenced by Smokediff. Smokediff computed the difference between two similar FDS cases and generated slice files. Difference contours from these slice files are displayed at 0 s, 10 s and 15 s.

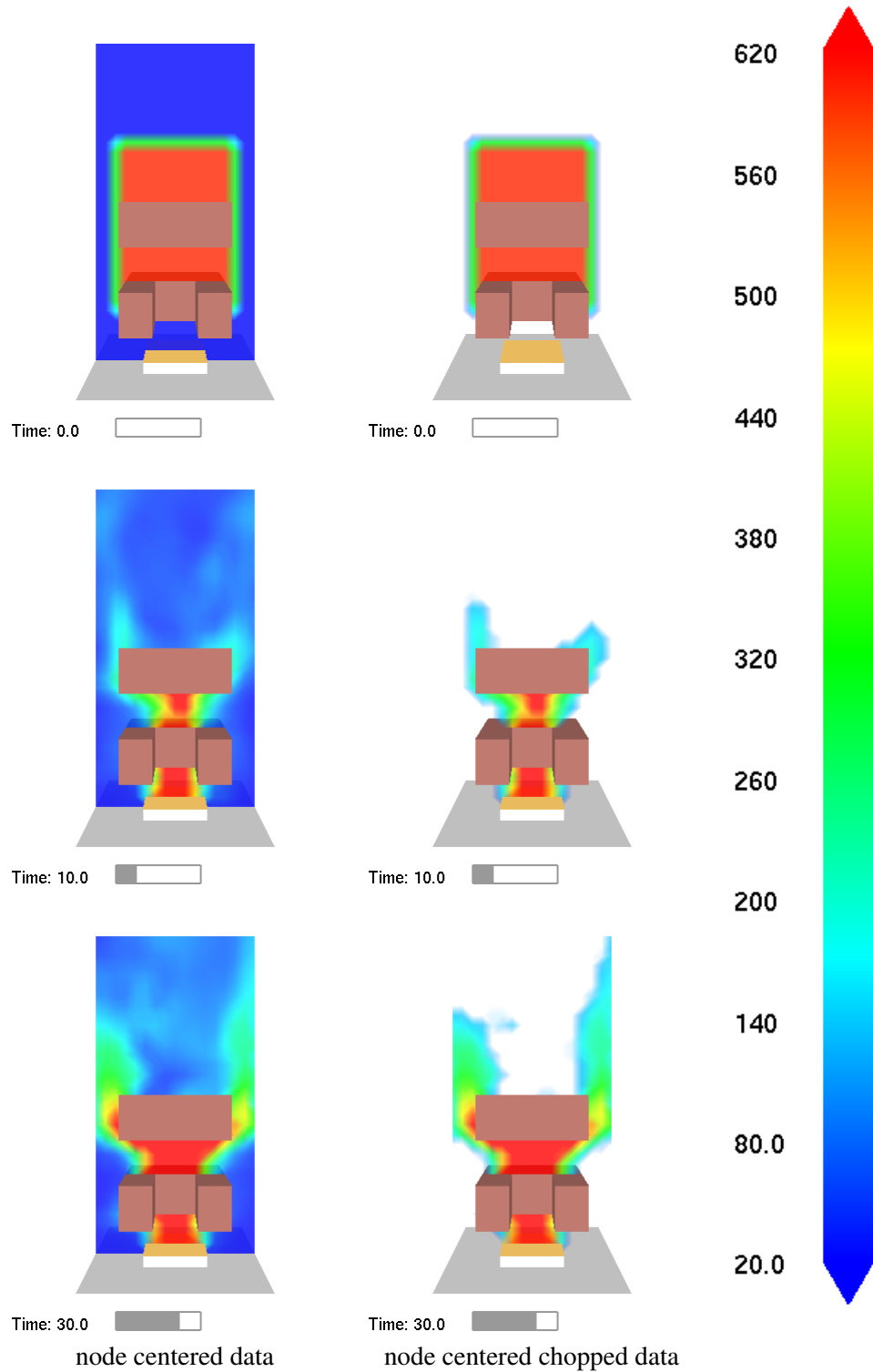


Figure 2.8: A test showing shaded and chopped contours of node centered slice file data at 0 s, 10 s and 30 s. A portion of the interior gas temperature is initialized to 600 °C. The slice file in this region should be red for the $t = 0$ s images. For the chopped contours, the color near the chop boundary should match the color near 140 °C in the colorbar.

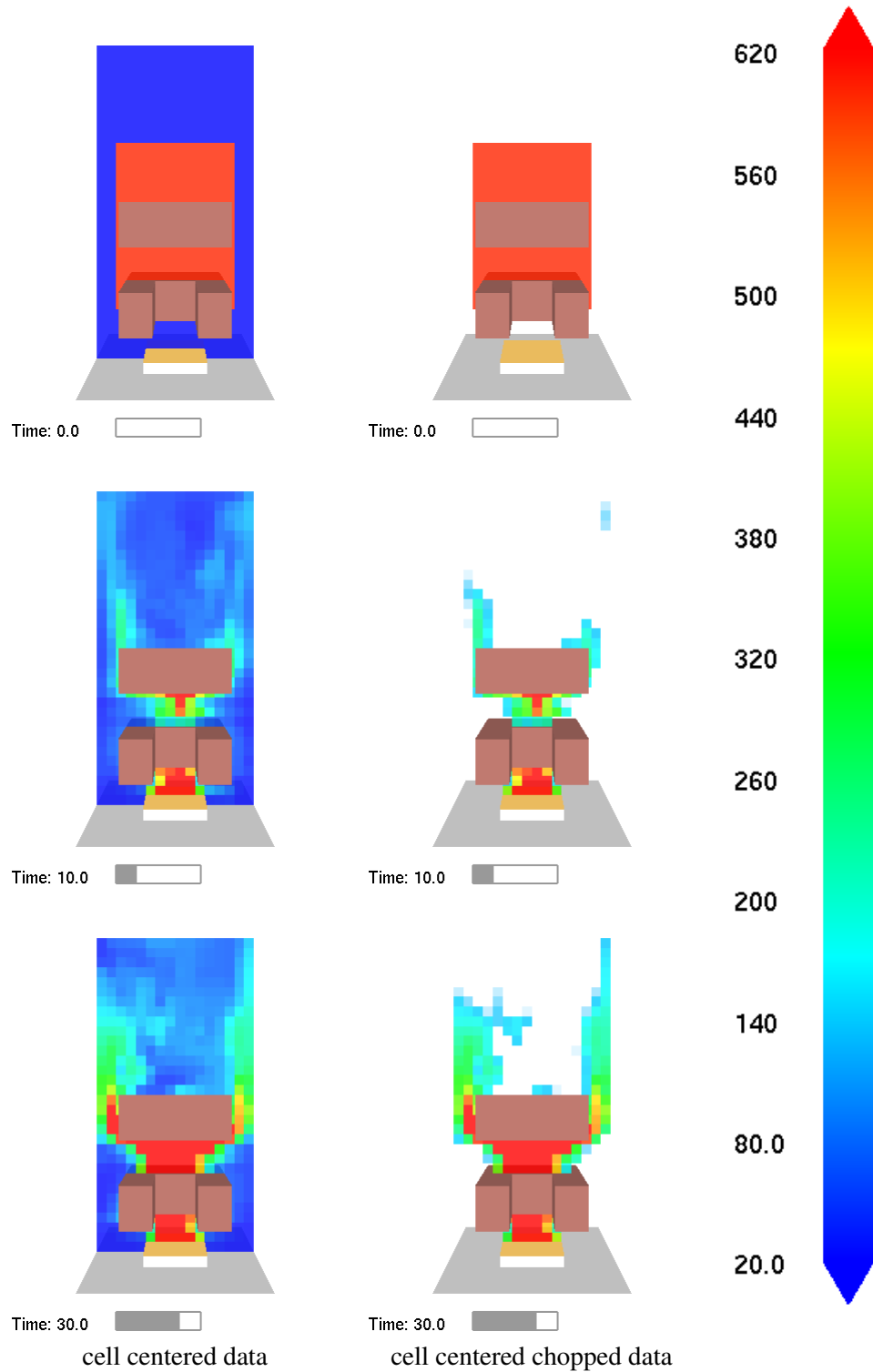
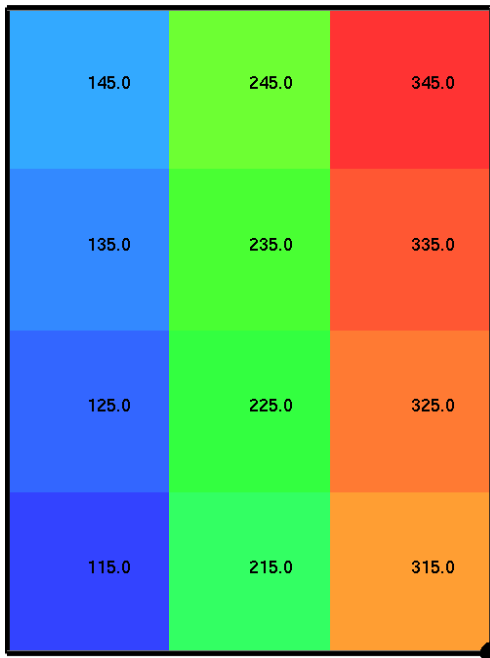
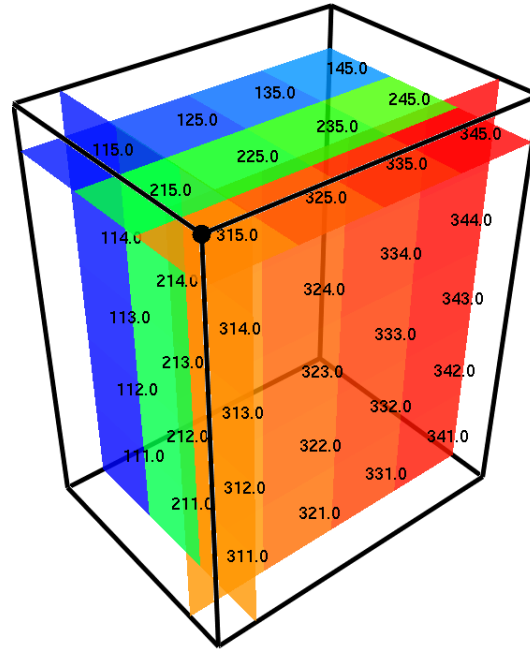


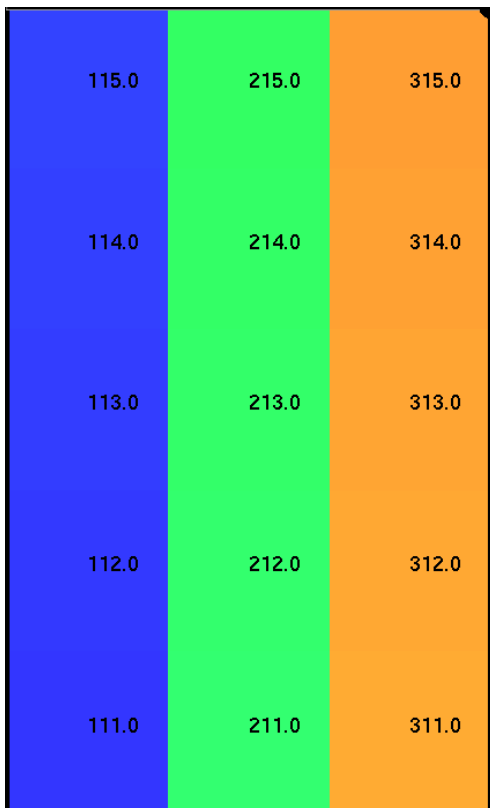
Figure 2.9: A test showing shaded and chopped contours of cell centered slice file data at 0 s, 10 s and 30 s. A portion of the interior gas temperature is initialized to 600 °C. The slice file in this region should be red for the $t = 0$ s images. For the chopped contours, the color near the chop boundary should match the color near 140 °C in the colorbar.



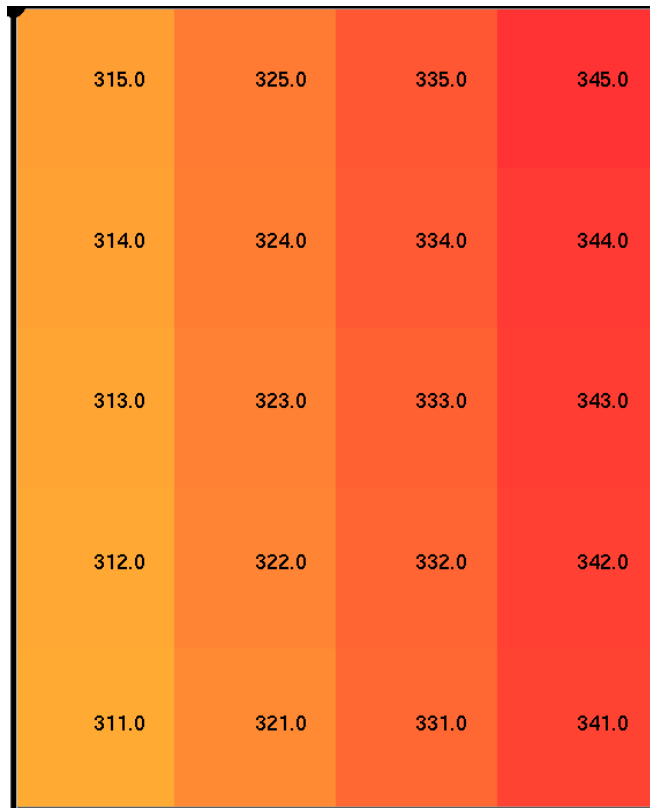
a) plan view (top slice)



b) 3D view



c) elevation view (front slice)



d) side view (right slice)

Figure 2.10: A test to demonstrate that cell centered slice file data is transferred to Smokeview properly. Cell values are initialized in FDS to $100 * i + 10 * j + k$ where $1 \leq i \leq 3$, $1 \leq j \leq 4$ and $1 \leq k \leq 5$. Cell values displayed should match the (i, j, k) grid cell indices. The three plots in the 3D view correspond to the plan, elevation and side view plots.

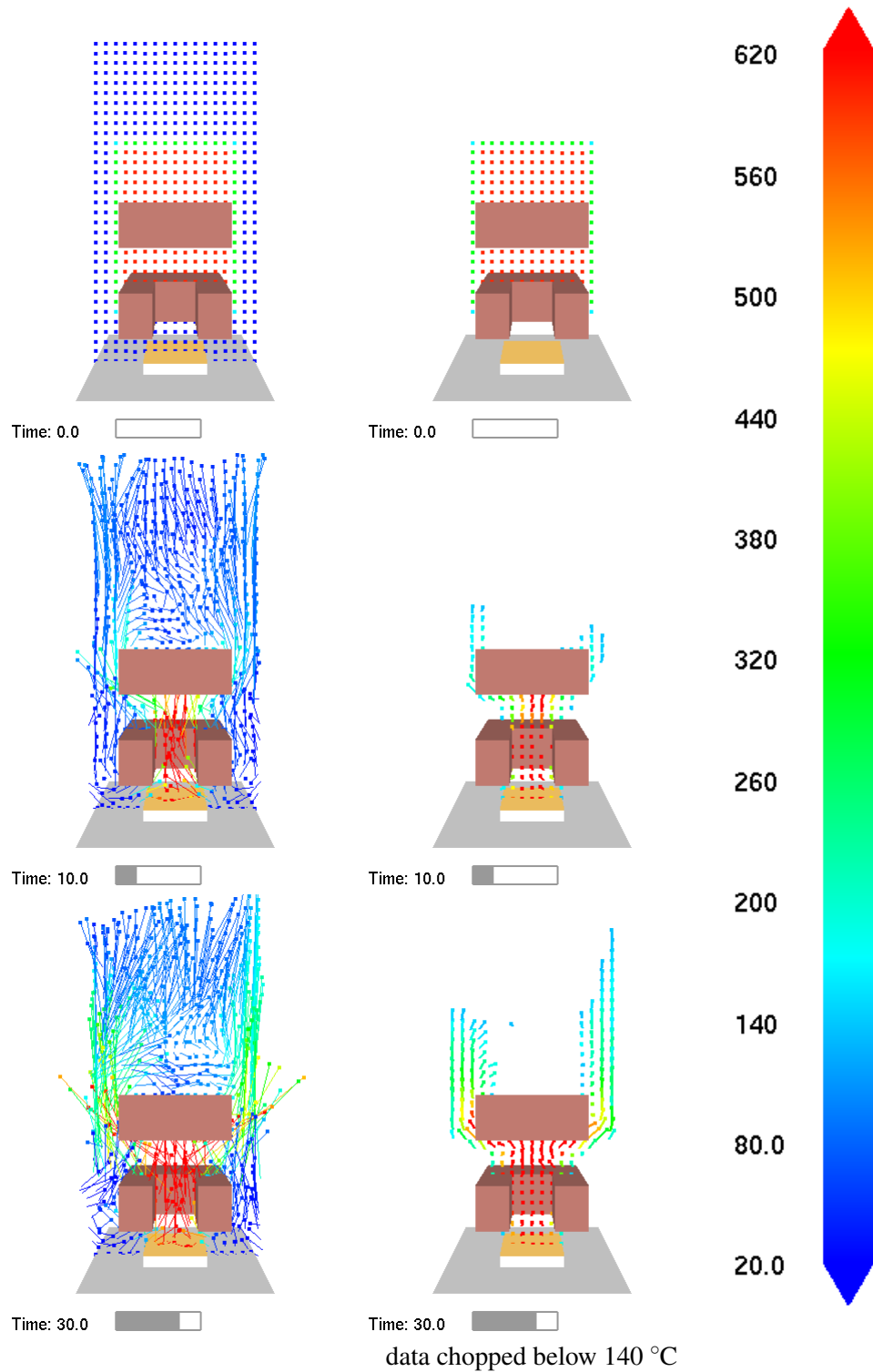


Figure 2.11: A test showing vector slice file data at 1 s, 10 s and 30 s. A portion of the interior gas temperature is initialized to 600 °C. The vectors in this region should be red for the $t = 0$ s images. For the chopped contours, the color near the chop boundary should match the color near 140 °C in the colorbar.

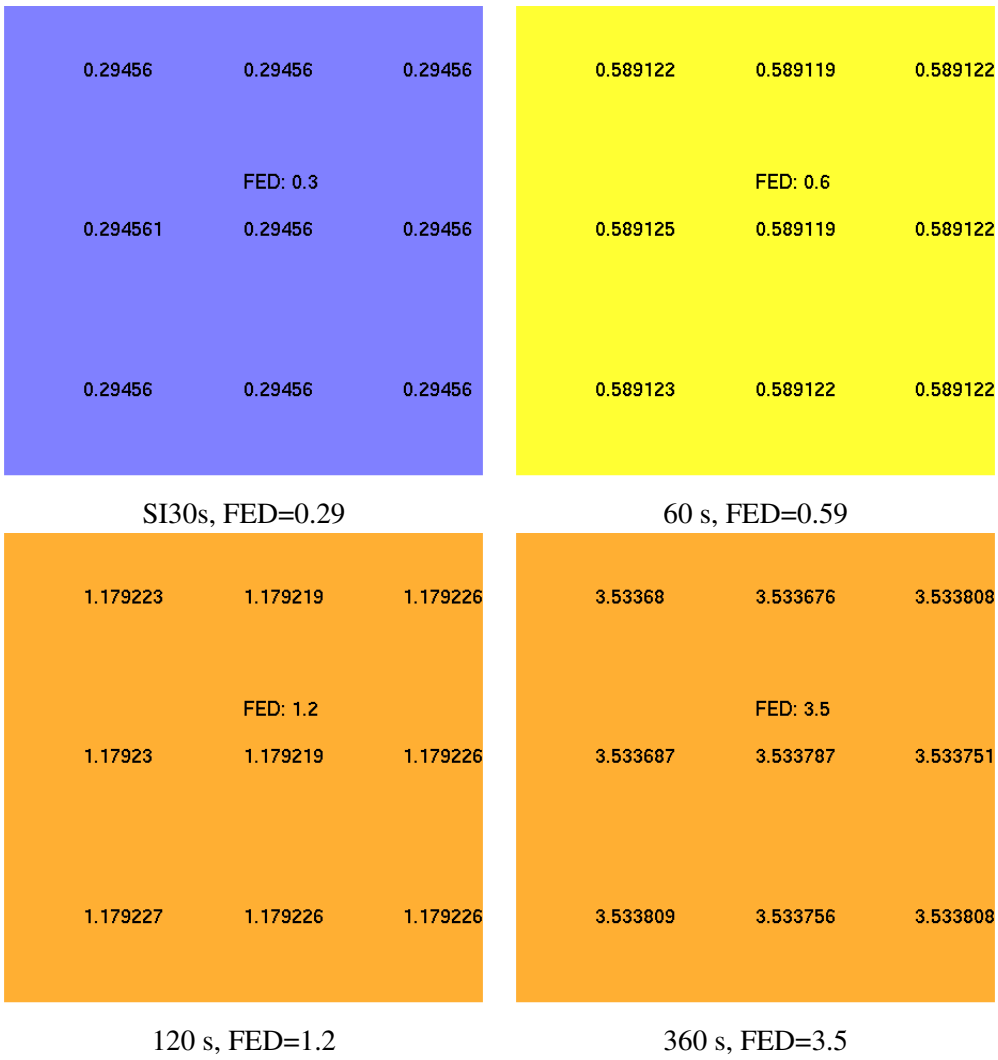


Figure 2.12: A test showing cell centered FED slice file data. These contours were generated using CO, CO₂ and O₂ data slices.

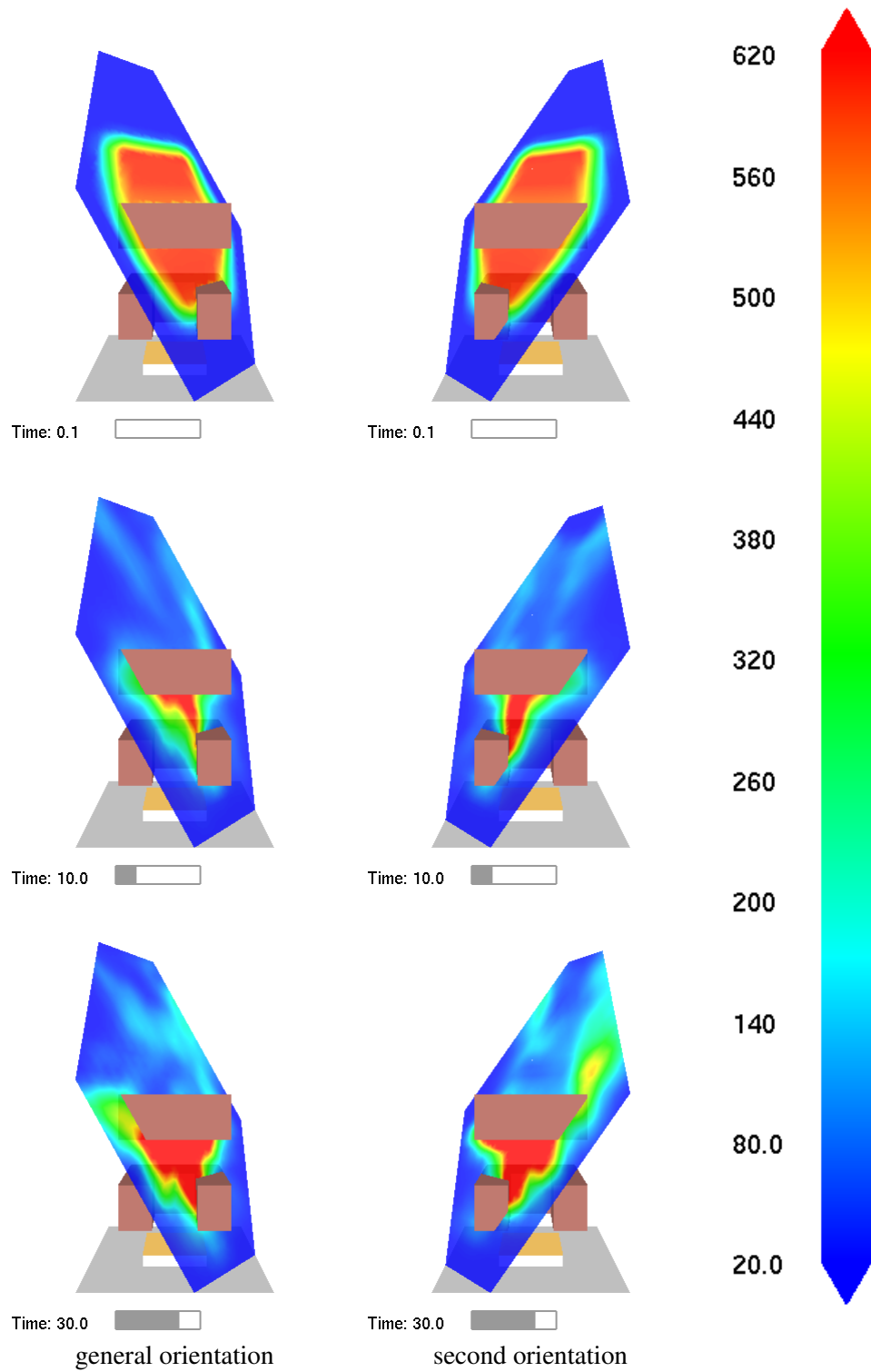


Figure 2.13: A test showing arbitrarily oriented slice file planes generated from 3D slice files at 0 s, 10 s and 30 s. Two orientations are shown at three time steps.

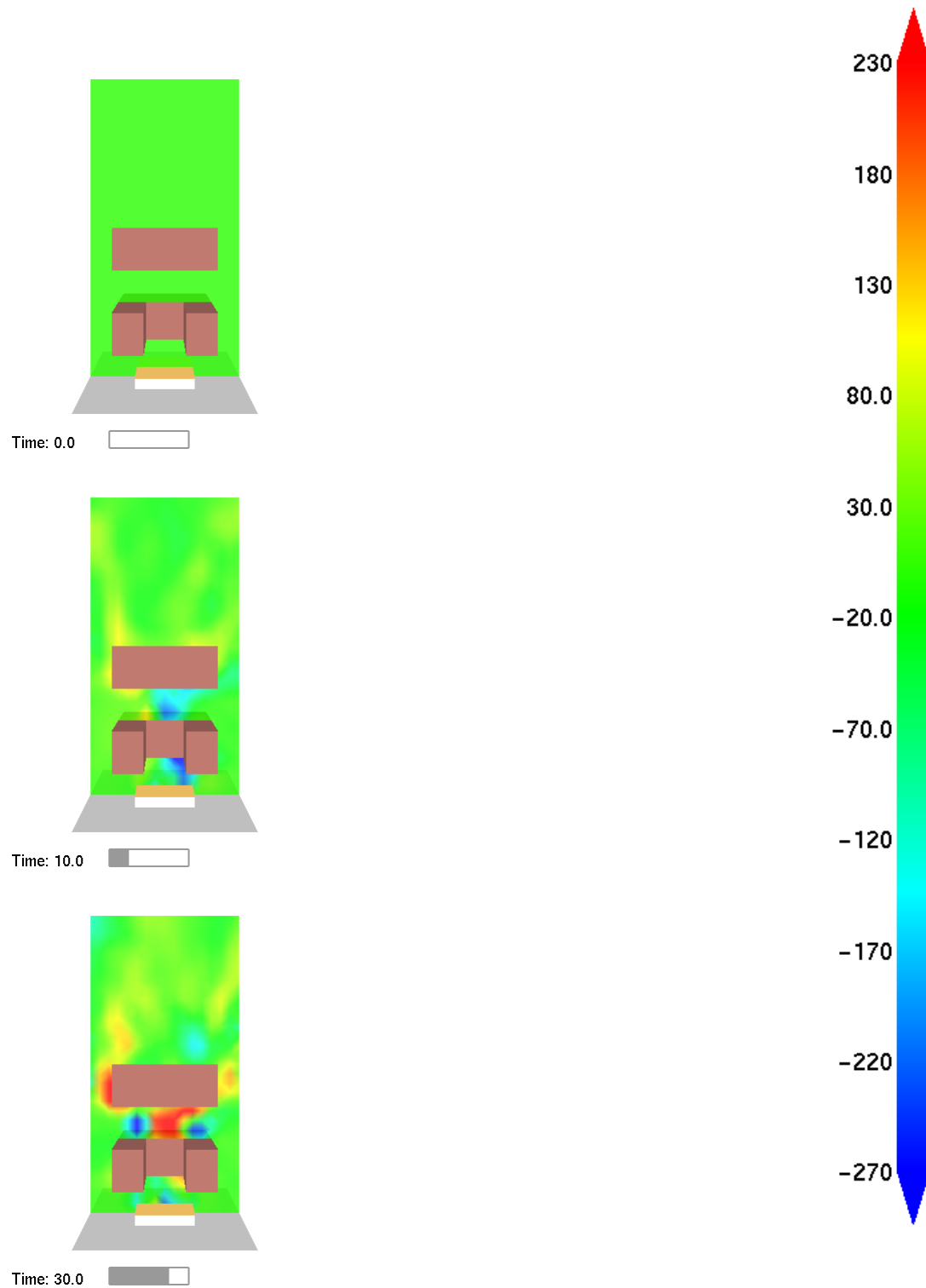


Figure 2.14: A test showing slice contours generated by differencing two cases with Smokediff at 0 s, 10 s and 30 s.

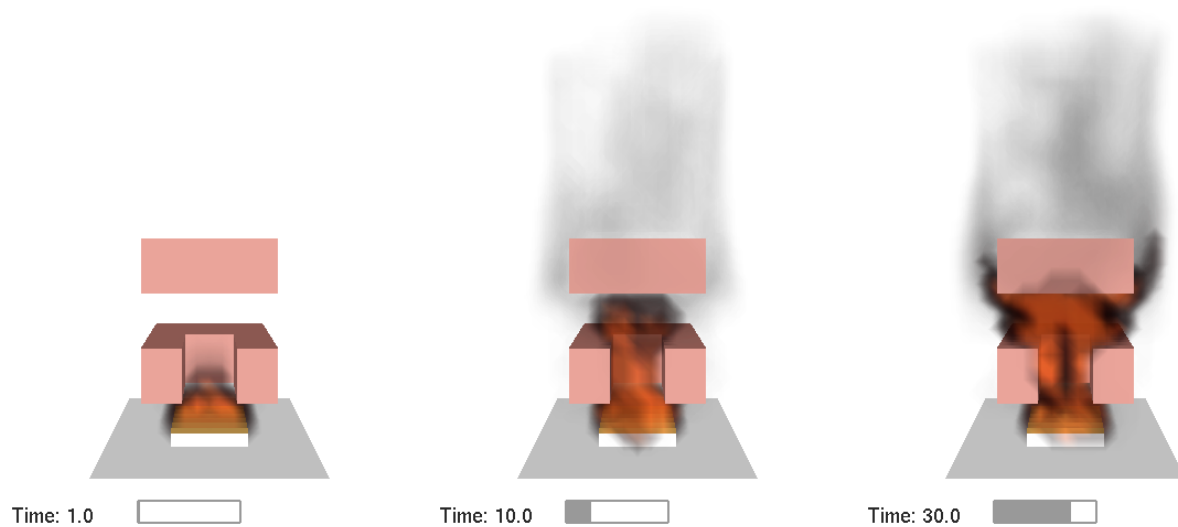


Figure 2.15: A test showing 3D smoke drawn at 1 s, 10 s and 30 s

2.5 3D Smoke

Figure 2.15 presents images verifying the display of 3D smoke and fire (heat release rate per unit volume or HRRPUV). A series of 3D smoke images are drawn at 1 s, 10 s and 30 s. The images contain semi-transparent slices derived from both soot density and HRRPUV data.

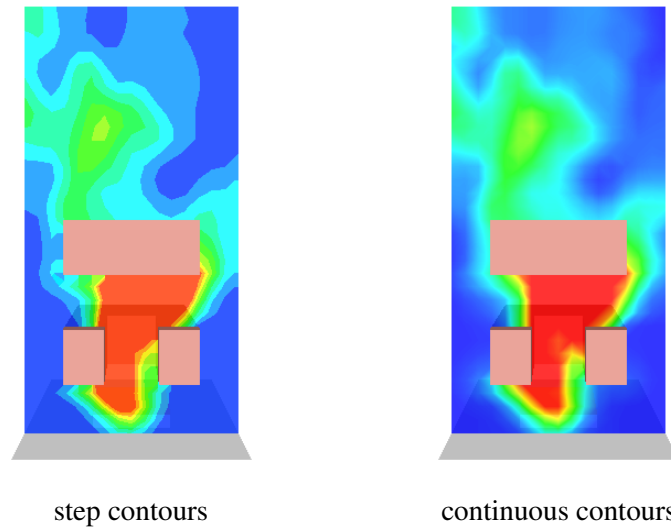


Figure 2.16: A test showing PLOT3D stepped and continuous temperature contours.

2.6 Plot3D

Figure 2.16 presents images verifying the display of PLOT3D contours. Three types of contours are available for visualizing PLOT3D data: stepped, continuous and narrow band. This figure shows stepped and continuous contours. Figure 2.17 shows stepped contours in at four positions along the y axis. Figure 2.18 shows narrow band contours at four positions front to back. Figure 2.19 shows velocity vectors at the same four positions as in Figure 2.17. Figure 2.20 shows PLOT3D iso-surfaces for 50 °C, 230 °C, 410 °C and 590 °C.

visualizing the motion of a fire across the simulation. The fire line slice is formed by setting the min and max temperature bounds to 20 °C and 200 °C respectively and chopping data below 150°C. The fire line slice file can then be made very small using Smokezip to compress it. Fire line images are drawn at 10 s, 20 s, 30 s and 40 s. The fire lines conform to the hill going through the middle of the scene. `fire_line.ssf`.

follows the terrain as

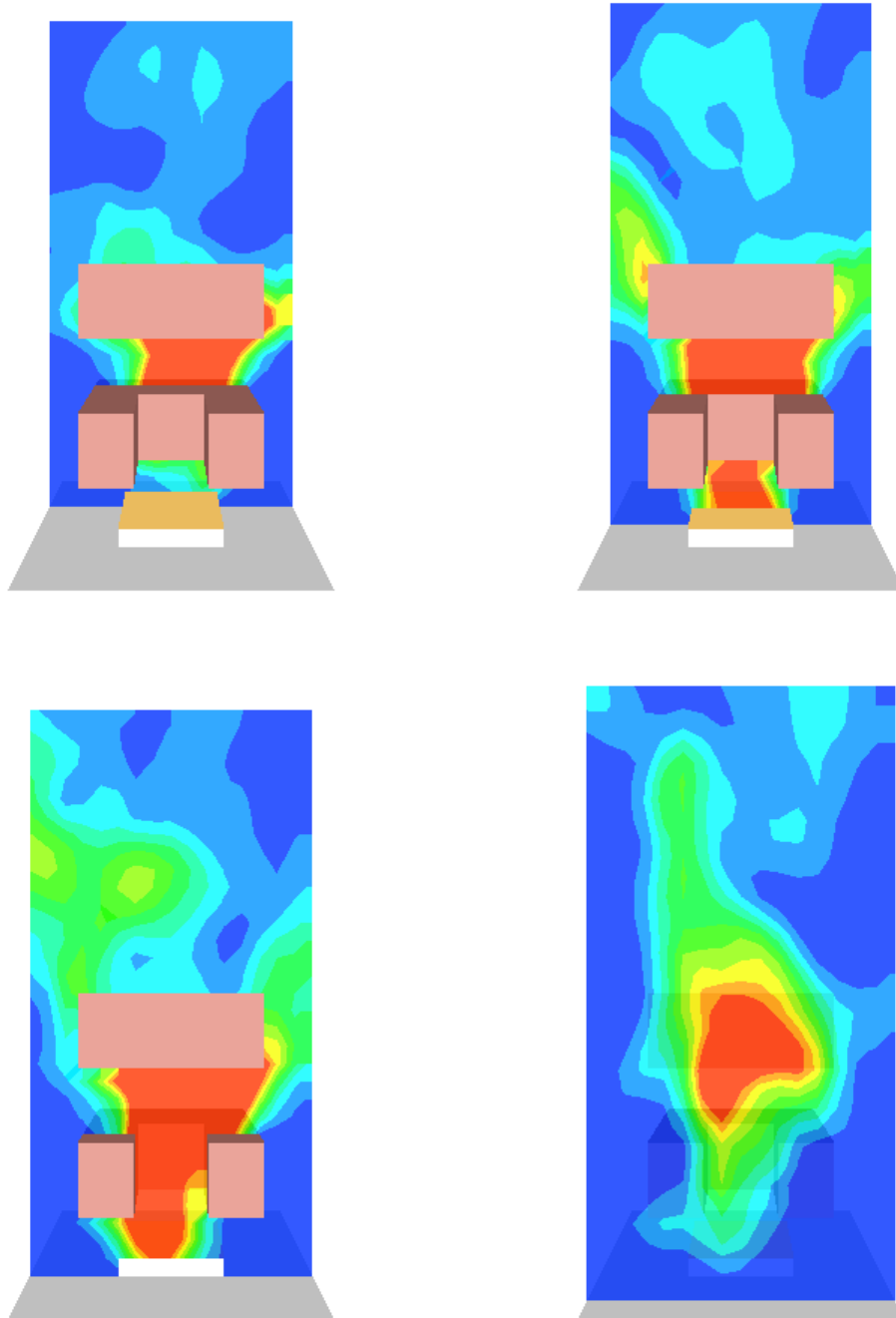


Figure 2.17: A test showing PLOT3D stepped contours at four positions along the y axis.

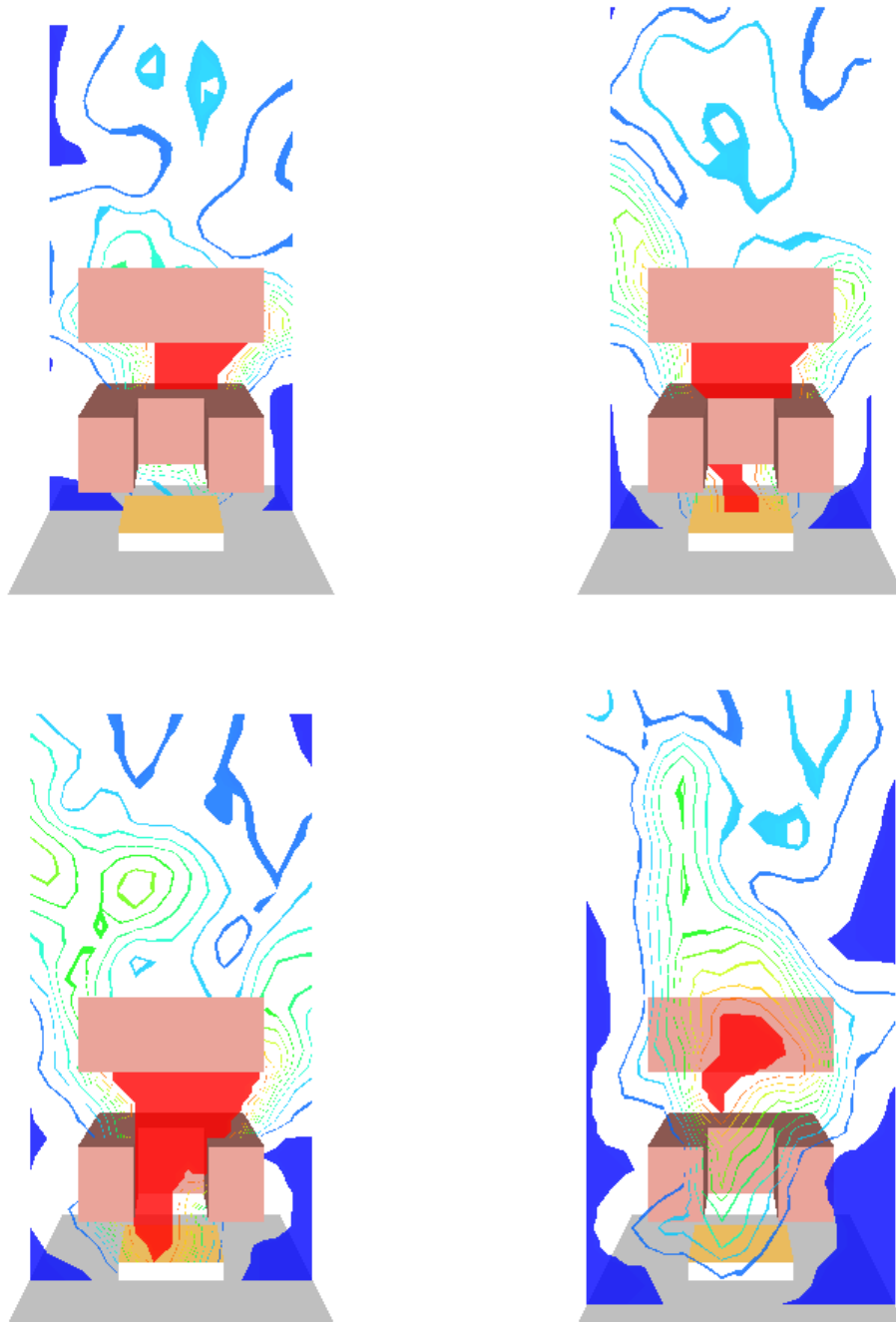


Figure 2.18: A test showing PLOT3D narrow band contours at four positions along the y axis.

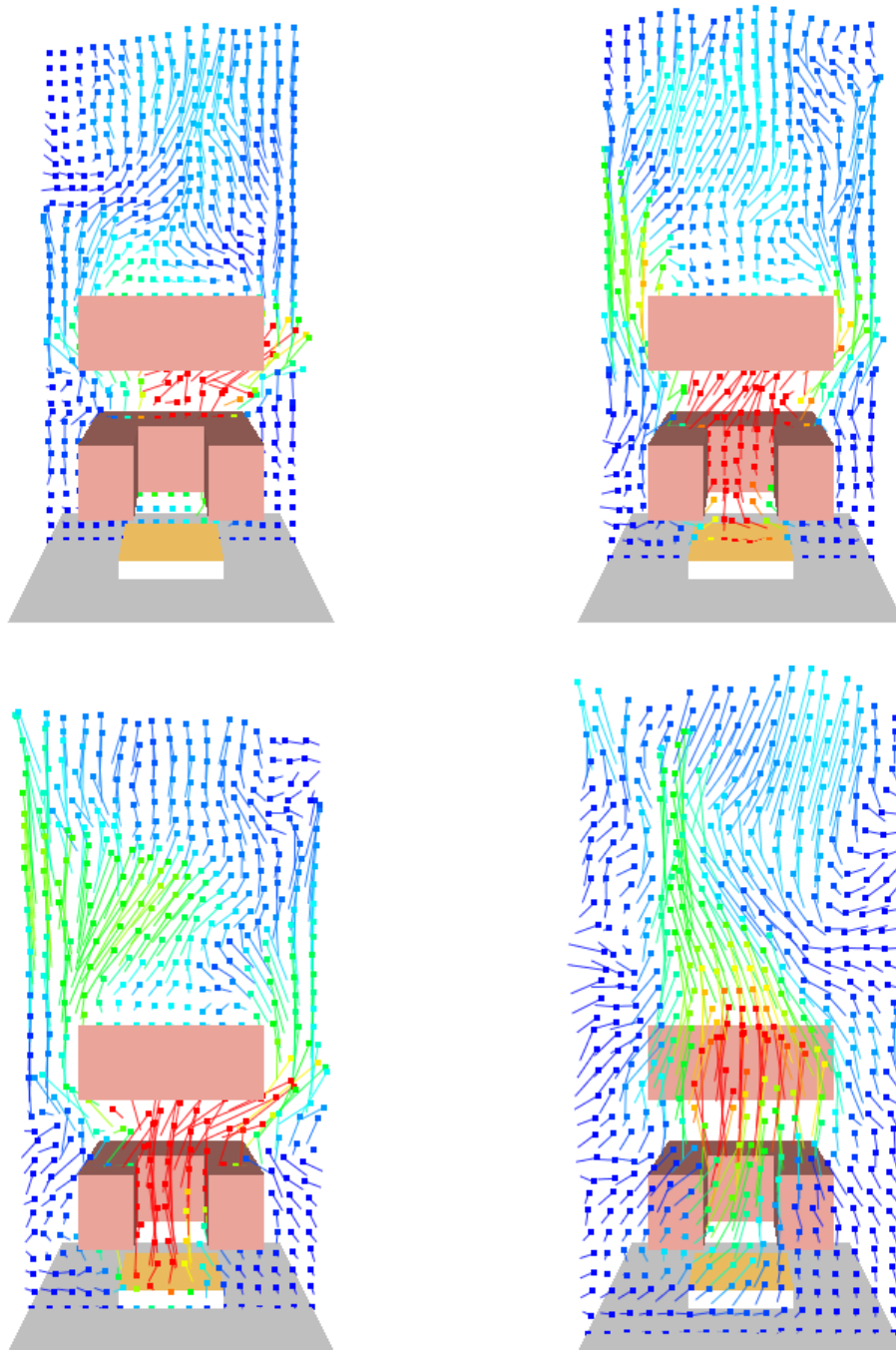


Figure 2.19: A test showing PLOT3D vectors shaded by temperature at four positions along the y axis.

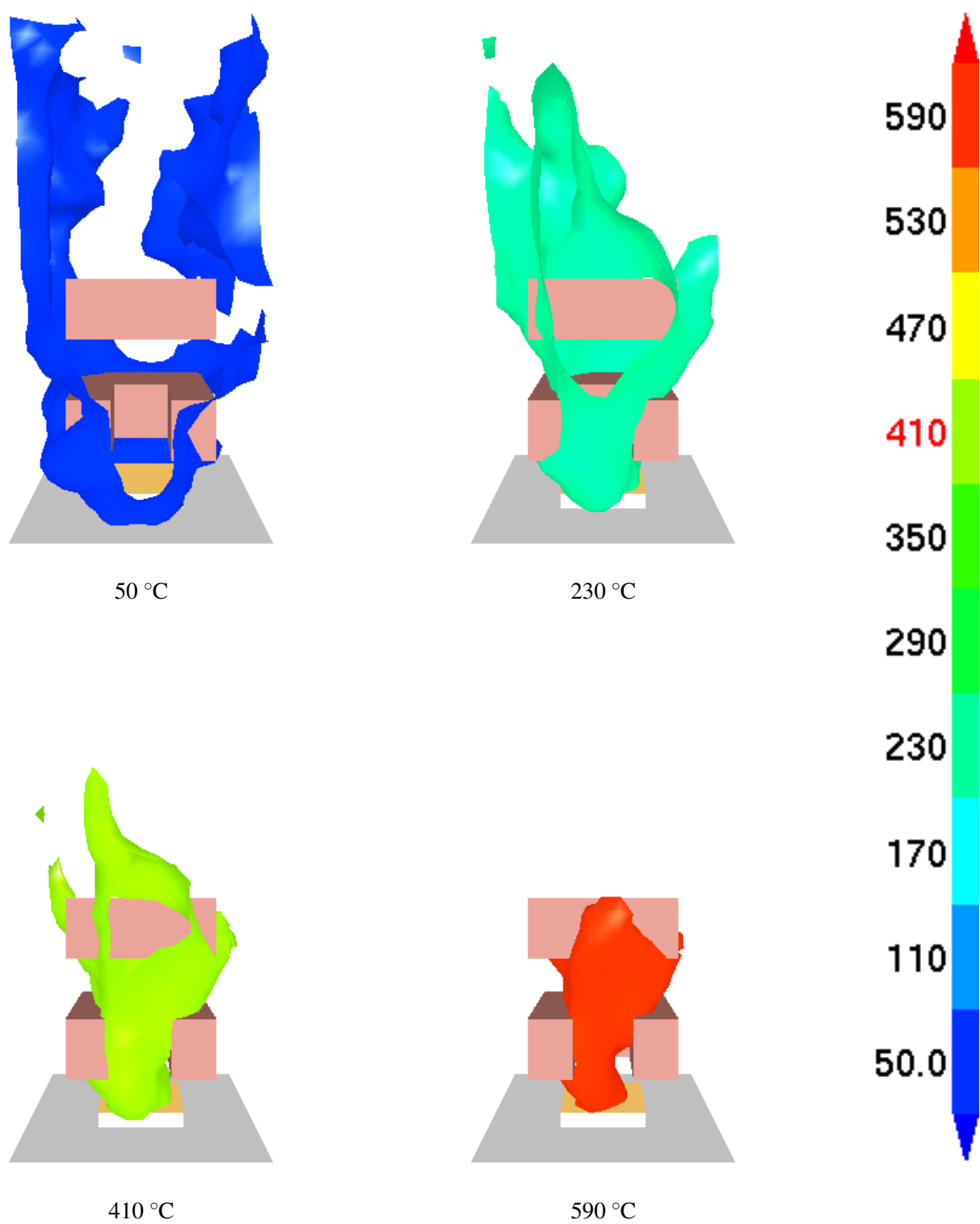


Figure 2.20: A test showing PLOT3D iso-surfaces for 50 °C, 230 °C, 410 °C and 590 °C.

Chapter 3

Smoke Visualization Tests

Correct smoke visualization requires that smoke flow be both computed and drawn correctly. The FDS Verification Guide [7] addresses computation in terms of soot/smoke production, transport, etc. A verification of how smoke is drawn is addressed in this chapter. More precisely, given a known density and distribution of soot, does the smoke drawn by Smokeview match how *theory* predicts it should appear. Presently, the main interest is in how smoke obscures objects in the background. Visualization effects due to light scattering are not modeled except to consider the smoke albedo when drawing its grey level.

The smoke drawing verification problem is partitioned into two steps. The first step, documented in Section 3.1, is to verify that Smokeview can record the correct grey level of smoke as it is drawn on the screen. This step is necessary because screen colors, in this case the smoke obscuration levels to be verified, are not directly accessible to Smokeview. They must be retrieved from the video card buffers. The second step, documented in Section 3.2, is to verify that Smokeview can draw the correct shade of grey given a known soot density level within the scene.

3.1 Recording Smoke Levels

Smokeview does not have direct access to colors drawn on the screen. Smokeview passes vertex colors to the video driver. The video driver in turn fills in interpolated colors within these vertices. To obtain these colors, Smokeview must ‘read’ them back in after they are drawn. A further complication is that smoke is drawn in Smokeview using a physical coordinate system. The smoke opacity levels are retrieved using the screen coordinate system, ie. at particular pixel locations. It is this transformation process, from physical to screen coordinates, that is verified in this step.

In the case of 3D smoke visualization, these colors represent smoke opacity levels. To obtain these levels and thereby verifying that they are correct, Smokeview makes use of a special sensor or device named *smoke_sensor* which Smokeview uses to obtain color values at the sensor location. This device behaves like other FDS devices but has the additional property that when used by Smokeview, it displays the grey level as viewed by the observer. This grey level is displayed as a number between 0 and 255.

The user places a device of type *smoke_sensor* at a particular (x, y, z) location. Smokeview displays the sensor as a white disk with color (255,255,255) always oriented towards the observer. When drawing smoke that resides between the sensor and the observer (your eye), the smoke sensor is partially obscured by the smoke. Smokeview then alters the smoke sensor color according to how much and how thick the intervening smoke is. It does this by blending each smoke plane one plane at a time using the color and opacity levels of that plane. The grey level is simply Smokeview’s computation of the integrated *smoke thickness* along a path between the sensor location and the eye. The colors displayed are the result computations performed by the video card using OpenGL, the graphics library used by Smokeview to visualize FDS



Figure 3.1: A test verifying that Smokeview can correctly transform the physical (x,y,z) location of the smoke_sensor object to its equivalent screen coordinates where the underlying smoke obscuration level is retrieved. A small white (255,255,255) smoke_sensor appears in front of a grey (128,128,128) obstacle. The value displayed should be 255, the value recorded by Smokeview. If the physical to screen transformation is performed incorrectly a value of 128 will be displayed (since it will pick up a color elsewhere).

scenarios.

Figure 3.1 illustrates an initial test of this process. It verifies that Smokeview can transform the sensor location from physical ‘FDS’ coordinates to screen coordinates (pixel locations) and further can correctly record the grey level at those screen coordinates even when surrounded by other objects with different colors. In this case, the smoke sensor is white and there is no intervening smoke. The background is a neutral grey with grey level of 128. The value displayed over the sensor then should always be 255 not 128 no matter how the scene is oriented. If Smokeview made an error determining the sensor screen coordinates the grey level displayed would be 128 (grey level of the adjacent grey background) not 255 (grey level of the sensor).

Figure 3.2 shows two colorbars, both containing shades of grey. One shows white and near-white shades, the other shows black and near black shades. This figure illustrates the difficulty one has in distinguishing nearly equal shades and by inference the difficulty in distinguishing two smoke scenes drawn using nearly the same *amounts* of smoke. When comparing a computed smoke shade with the actual (as in Figure 3.4) one must keep in mind the eye’s inability to distinguish nearly equal shades of grey.

3.2 Verifying Smoke Levels

Smoke levels are verified in Smokeview by setting up an FDS case with a constant smoke density throughout the domain. A smoke density $S = 79.67 \text{ mg/m}^3$ and mass extinction coefficient $K = 8700 \text{ m}^2$ are chosen in Eq. 3.1 to generate opacity levels 192, 128, 64, 32, 16 and 8 at path lengths $\Delta x = 0.4 \text{ m}, 1 \text{ m}, 2 \text{ m}, 3 \text{ m}, 4 \text{ m}, 5 \text{ m}$

$$\alpha = 255 \exp(-K S \Delta x) \quad (3.1)$$

where α is a scaled opacity (from 0 to 255 rather than the usual 0 to 1).

Figure 3.3 shows a side view of the numerical smoke box used to perform this test. In addition to the walls surrounding the box, this box consists of six parallel walls within the box spaced 1.0 m apart. The

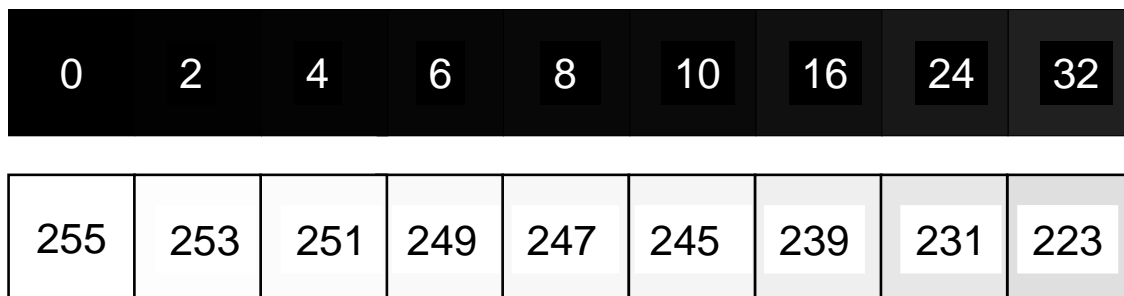


Figure 3.2: Shade of grey resolution test. The number within each square represents the shade of grey used to color that square, 0 for black and 255 for white. Adjacent squares are drawn with nearly equal shades testing the ability of sensors such as the eye, computer monitor or the printed page to distinguish them.

widths increase from one wall to the next (from front to back) so that when looking at the box from the front a different distance or path length occurs between the observer and the portion of the wall that is visible. Again, the spacing between the walls, the distance between the walls and the observer and the initial soot densities is chosen so that the computed smoke obscurations work out to *nice* values.

Figure 3.4 shows quantitative tests of the smoke opacity calculation performed in Smokeview for two different FDS grid resolutions and three different skip levels¹. This figure also gives the predicted shades of grey based upon the specified soot densities, mass extinction coefficient and path lengths. The smoke visualization algorithm is verified then if the shades of grey in the Smokeview visualizations match the corresponding predicted shades of grey. Each shaded rectangle is accompanied by a numerical value that can also be used to judge whether the visualization is verified.

The numbers displayed in this Fig. 3.4 represent the shade of the underlying rectangle. These numbers may be verified by using a program such as Adobe Photoshop to examine the pixel values of this rectangular region. The numbers are verified when they match the pixel values as reported by Photoshop or any other program that can report image values.

When soot densities are constant, smoke grey level or opacity, α may be computed by using Eq. (3.1). Solving Eq. (3.1) for Δx gives

$$\Delta x = -\frac{\ln(\alpha/255)}{KS} \quad (3.2)$$

Path lengths (smoke sensor locations) are chosen to obtain grey levels, α , of 192, 128, 64, 32, 16 and 8. These path lengths may be found by substituting these grey levels into Eq. (3.2) to obtain Δx values of 0.41 m, 1.0 m, 2.0 m, 3.0 m, 4.0 m and 5.0 m respectively. The walls in `smoketest2.fds` are placed at these distances from the front of the simulation domain. Comparing the Smokeview generated smoke levels with theoretical values one finds as expected that better results are achieved when using a more refined grid (0.1 m rather than 0.2 m) and using all planes (no skipping).

¹Smokeview allows one to skip grid planes when visualizing smoke. The smoke opacity is adjusted by Smokeview to account for the skips.

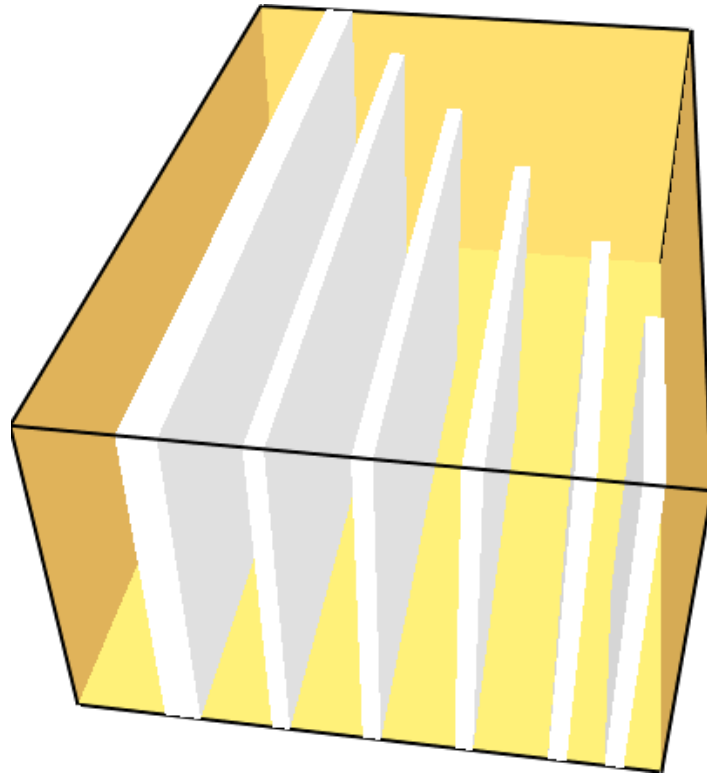


Figure 3.3: Side view of the numerical smoke test compartment. Walls are placed at 0.4 m, 1 m, 2 m, 3 m, 4 m, 5 m from the front to make the theoretical grey levels work out to 192, 128, 64, 32, 16 and 8.

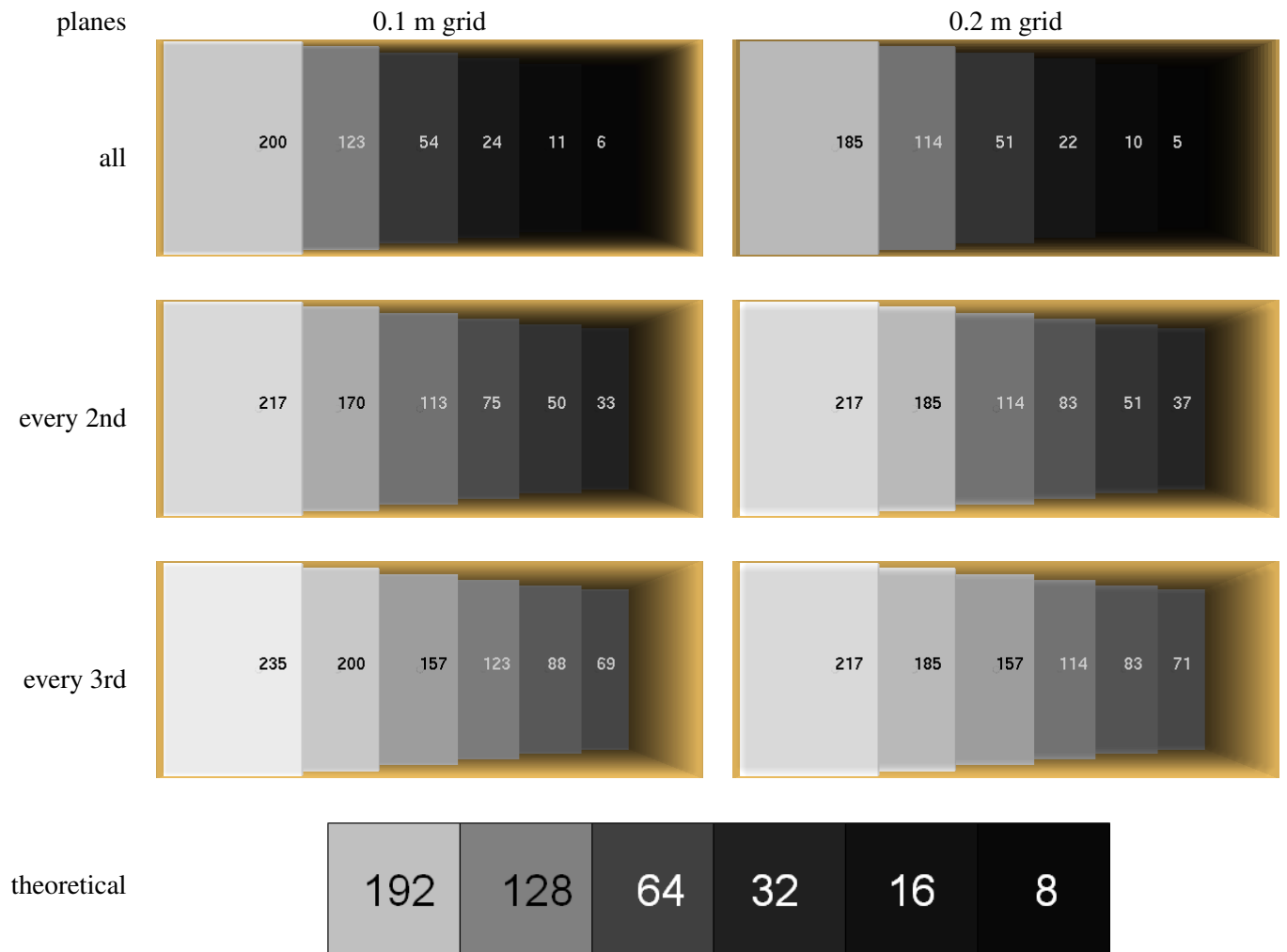


Figure 3.4: A test verifying the smoke opacity calculation. This test simplifies the general case by assuming a uniform distribution of smoke. The test is repeated for two grid resolutions and three grid spacings. The FDS input file is set up to result in theoretical grey levels of 8, 16, 32, 64, 128 and 192 for the different regions of the test.

Chapter 4

WUI Test Cases

This chapter presents techniques added to Smokeview [2] applicable for visualizing WUI (wildland urban interface) and wild fire scenarios. Two techniques involve 1) visualizing changing terrain using satellite images with modeling results overlayed and 2) making use of geometric objects for representing modeling elements such as trees or building structures. These two techniques use data generated by fire models such as FDS [9] or WFDS (FDS revision 9977)[10]. A third technique involves visualizing experimentally derived data rather than modeling data. Wind sensor data is visualized using flow vectors along with visual indicators of data uncertainty. Smokeview images are displayed in the following figures to illustrate these techniques.

4.1 FDS Cases

The figures in this section were generated with FDS revision 16557 . Figure 4.1 presents contours of a level set computation. The level set contours are drawn on a sloped terrain. These level set contours represent a fire line. A fire line is used with wildland fire simulations as an efficient method for visualizing the motion of a fire across the simulation. A fire line in the context of Smokeview is just a special case of a temperature slice contour. The thin red line represents the location where burning is currently taking place. The grey region represents where burning has occurred and the green region represents where burning has not occurred. Level sets are used to quickly (relative to a complete computational fluid dynamic calculation) model fire spread and fire line locations. In this particular case, images are drawn at 30 s, 60 s, 90 s and 120 s. The test verifies that the level set slices follow the sloped terrain. Figure 4.2 and 4.3 also present images of level set contours. These contour slices are drawn along with a satellite image of the region being modeled.

Figure 4.4 presents images of temperature contours drawn on a terrain geometry. The scenario consists of a small hill parallel to the vertical axis of the scene. Images are drawn at 15 s, 30 s, 45 s and 60 s. The test verifies that temperature contours follow the sloped terrain and that the red blockage (a non-terrain blockage representing a structure) is drawn correctly along with the terrain.

Smokeview typically visualizes data generated by software such as FDS [9], CFAST [11] or WFDS[10]. Figure 4.5 presents a visualization of data obtained from wind measuring sensors rather than simulation data. Smokeview can use the wind sensor data to visualize velocity profiles and variability and uncertainty present in the experimental data. The vertical array of green spheres represent the measurement locations. The distance from these spheres to the green or red spherical shells gives a relative measure of the wind speed. The shell diameter gives a measure of the wind direction measurement uncertainty. Likewise the

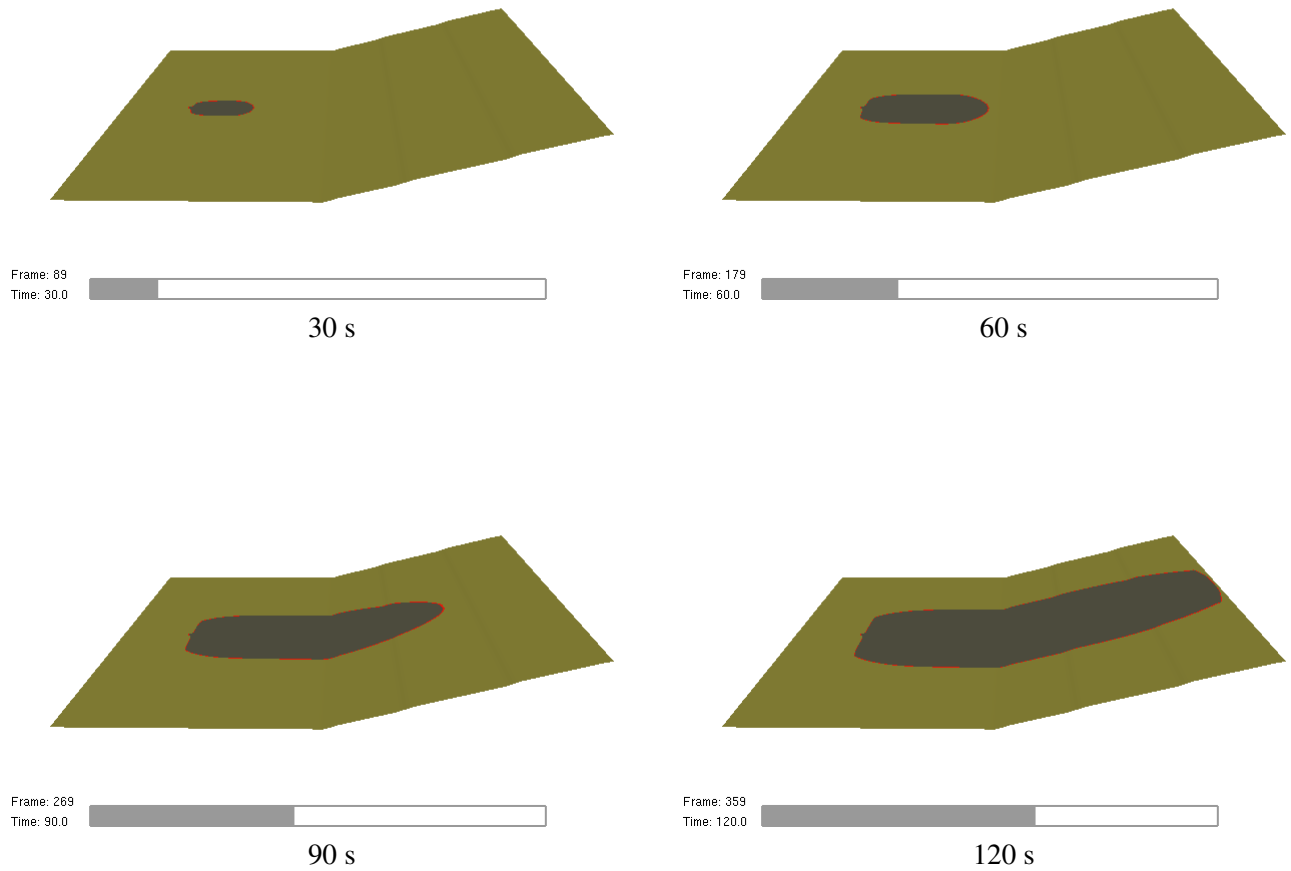


Figure 4.1: A test showing level set slices drawn along a sloped terrain at 30 s, 60 s, 90 s and 120 s. The test verifies that level set slices follow the sloped terrain. The thin red line represents where burning is currently taking place. The grey and green regions represent where burning has and has not occurred respectively.



0 s



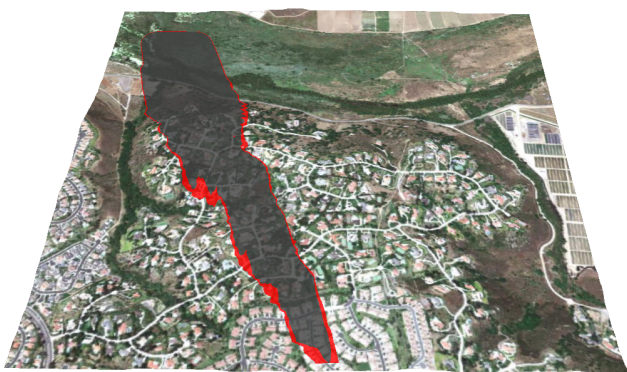
300 s



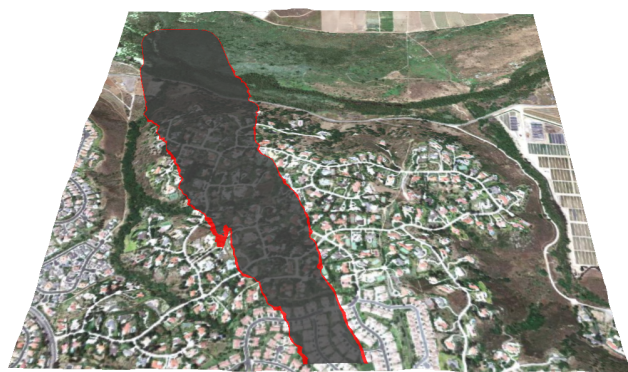
600 s



900 s



1200 s



1500 s

Figure 4.2: A test showing a fire line slice drawn along a sloped and textured terrain at 0 s, 300 s, 600 s, 900 s, 1200 s, and 1500 s of simulated time. The test verifies that the level set slices follow the sloped terrain.



Figure 4.3: A test showing a fire line slice drawn along a sloped and textured terrain at 1800 s simulated time.

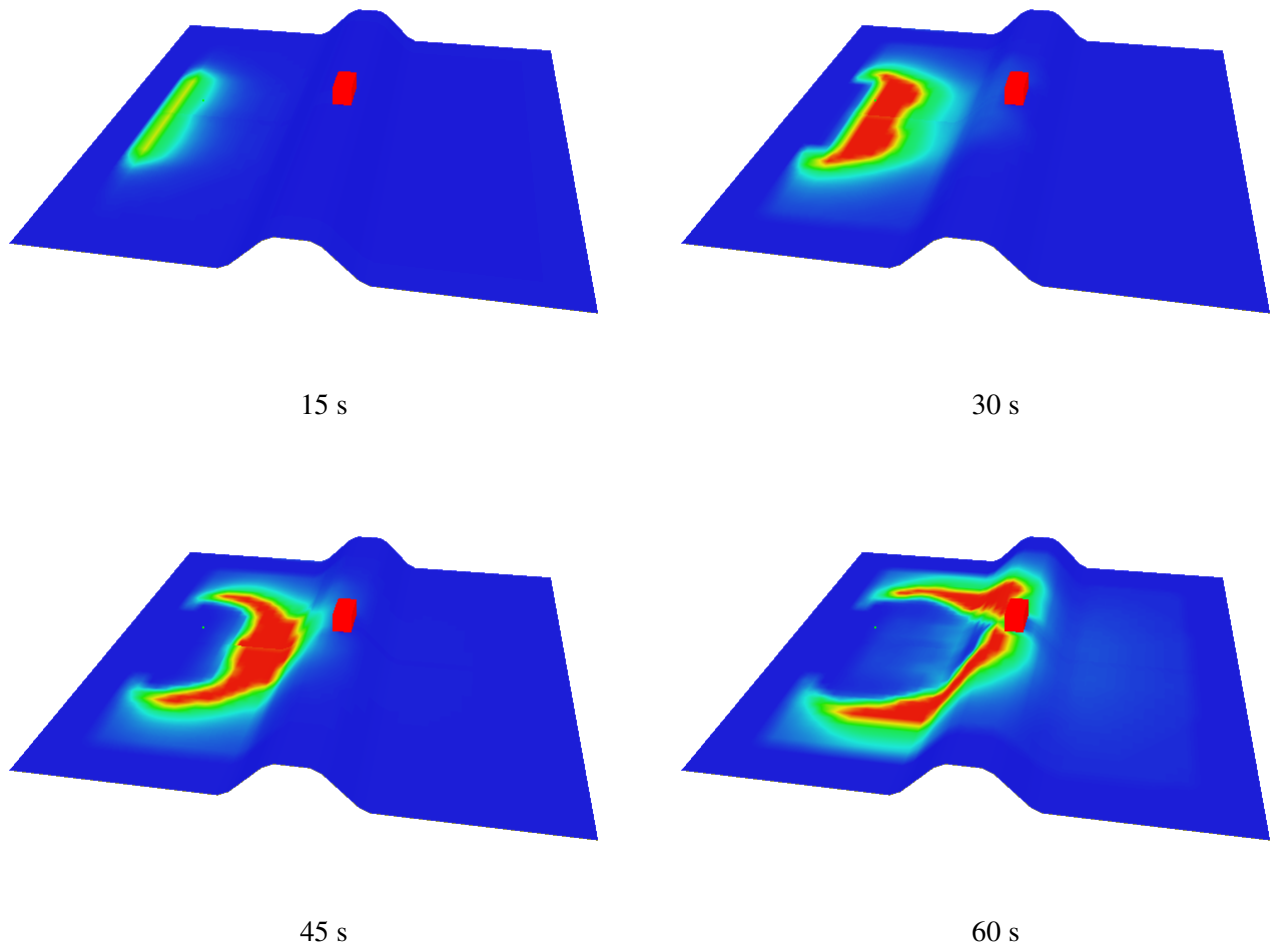


Figure 4.4: A test showing temperature slices drawn along a sloped terrain at 15 s, 30 s, 45 s and 60 s. The test verifies that the temperature slices follow the sloped terrain and that the non-terrain red blockage (representing a structure) is drawn with the terrain.

shell thickness give a measure of the wind velocity measurement uncertainty.

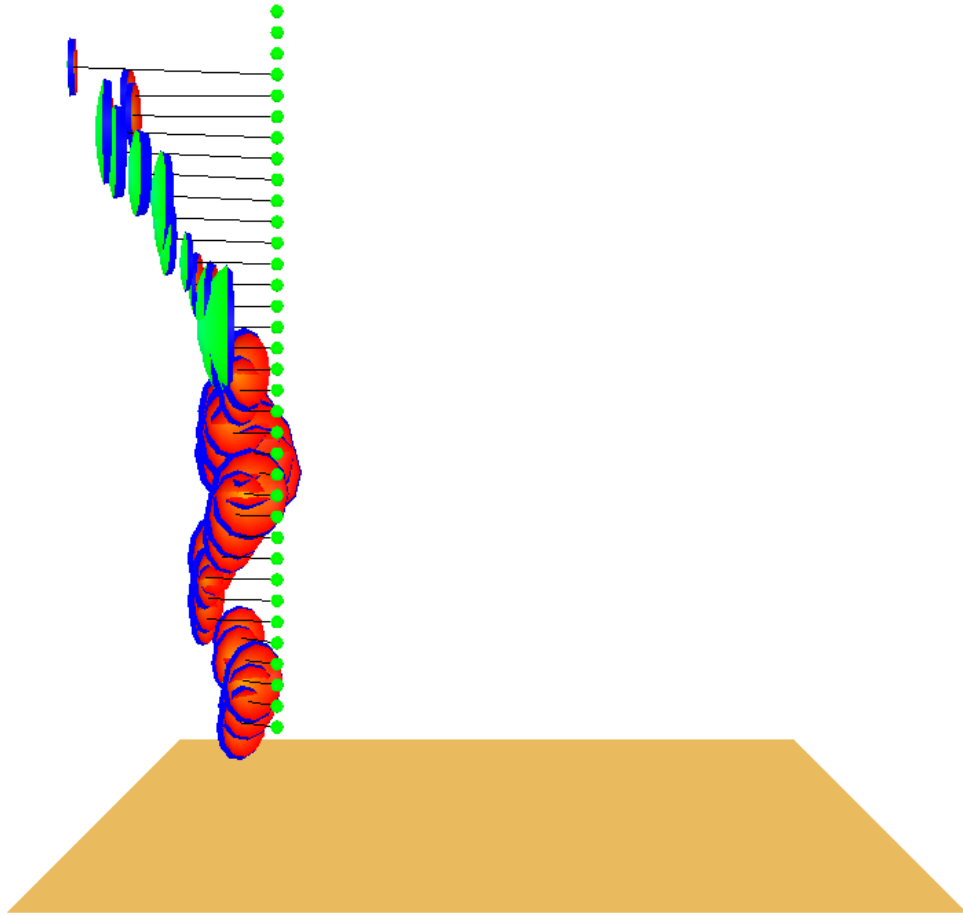


Figure 4.5: A test showing wind vectors drawn using data obtained from a wind sensor. The line segments represent wind speed and direction. The spherical shells represent uncertainty in wind direction (shell diameter) and wind speed (shell thickness).

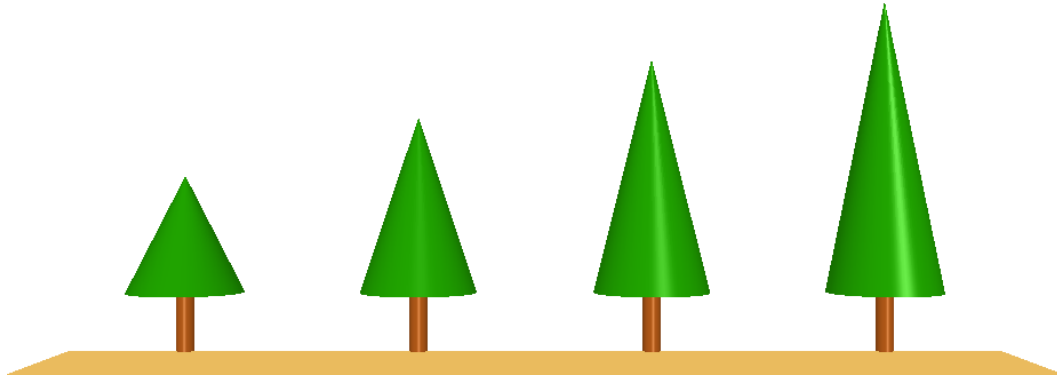


Figure 4.6: A test showing trees drawn as Smokeview objects. The tree canopy is colored green drawn as a cone, the tree trunk is colored brown drawn as a cylinder.

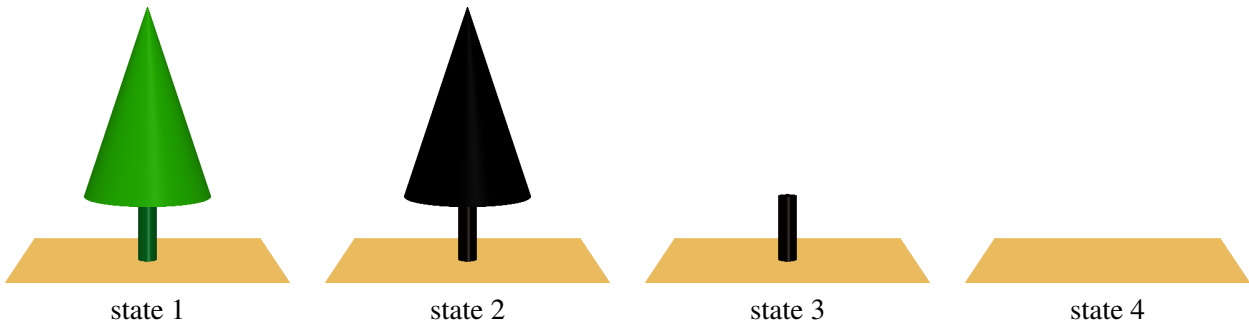


Figure 4.7: A test showing four states of the smokeview trunk and canopy objects. The states are unburned trunk and canopy, burned trunk and canopy, burned trunk (canopy burned away), both canopy and trunk burned away.

4.2 WFDS Cases

The figures in this section were created using data generated by WFDS (FDS svn revision 9977). Figure 4.6 shows an example of trees drawn as Smokeview objects. The tree is drawn as a cylinder colored brown and a cone colored green. Tree objects may change appearance as a simulation progresses as illustrated in Fig. 4.7. Tree states are a normal view, partially burned, partially burned (canopy burned away) and completely burned.

Figure 4.8 shows an example of trees drawn as particles. Particle represent small units of fuel. Particle color represents temperature. As particles use up fuel they disappear. When many trees are simulated, it is difficult to discern individual trees when drawn as particles. Fig. 4.9 shows an equivalent view using an isosurface. The isosurface location is determined from the boundary (between particle and non-particle regions in the simulation) of the particle cloud shown in Fig. 4.8. Figure 4.10 show a particle file drawn using canopy objects. The objects are colored according to the particle color as computed by FDS.

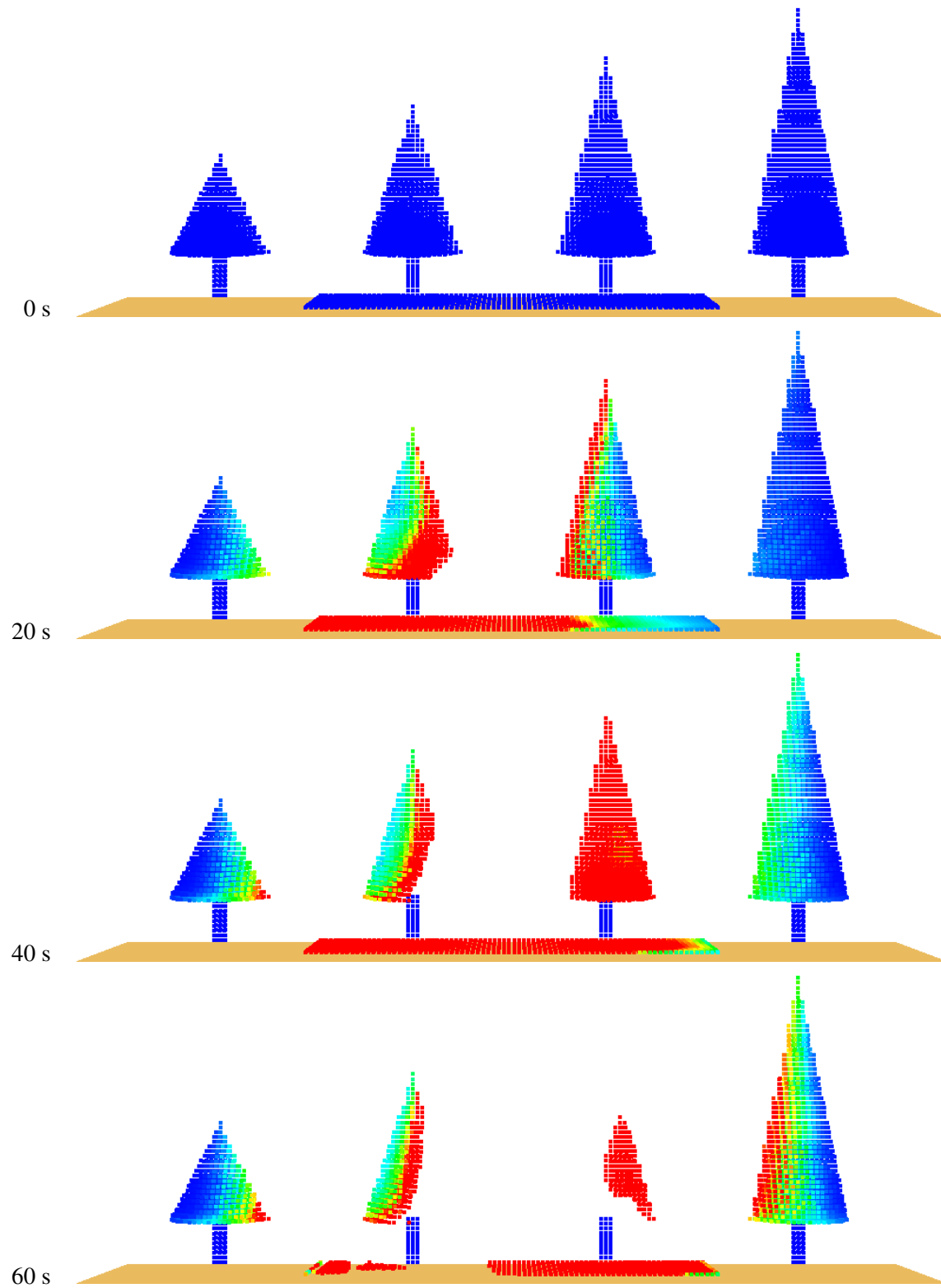


Figure 4.8: A test showing trees drawn as particles at 0 s, 20 s, 40 s and 60 s. Particle represent small units of fuel. Particle color represents temperature. As particles use up fuel they disappear.

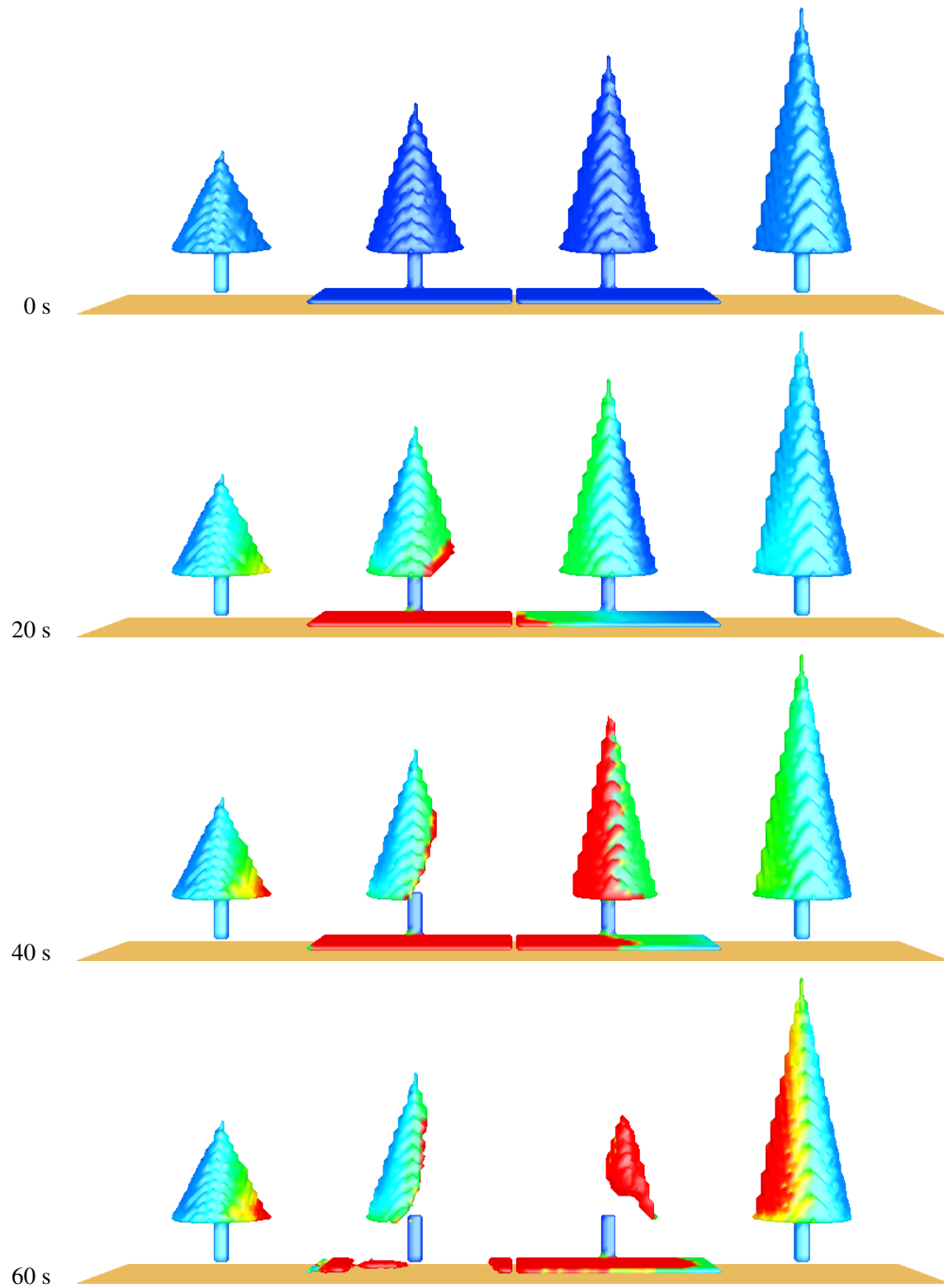


Figure 4.9: A test showing trees drawn as isosurface at 0 s, 20 s, 40 s and 60 s. The surface location is determined from the boundary of the particle cloud shown in Fig. 4.8.

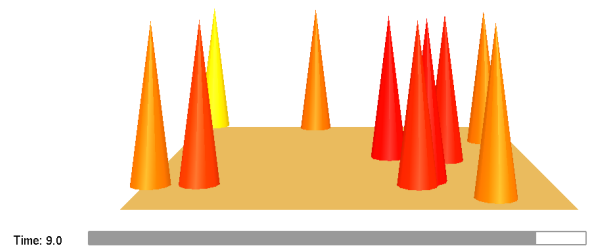
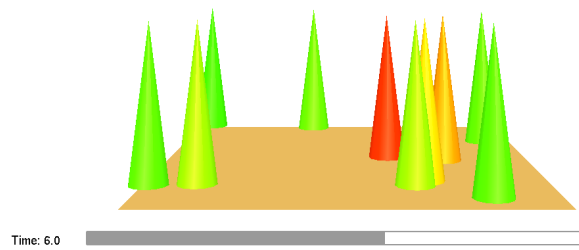
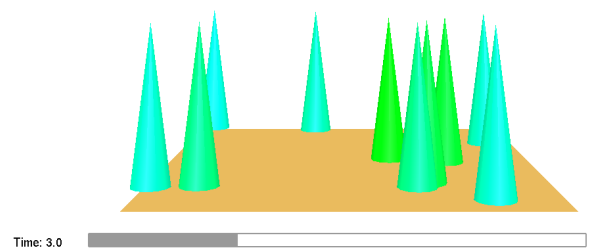
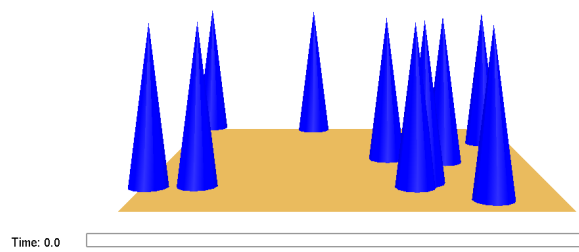


Figure 4.10: A test showing particles drawn as canopy objects at 0 s, 3 s, 6 s and 9 s. The canopy is colored according to temperature.

Chapter 5

Other Tests

5.1 Obstacles

Figures 5.1 and 5.2 test different methods for displaying obstacles. Figure 5.1 shows obstacles drawn as solids, as outlines or hidden. Figure 5.2 shows obstacles drawn transparently. The left view shows the opaque obstacle drawn in front (and blocking) a portion of the transparent obstacle behind. The center view shows both obstacles unobscured. The right view shows the transparent blockage drawn in front with the opaque blockage visible behind. The images for this test were created automatically by running the Smokeview script, `plume5c.ssf` (see Appendix B.4).

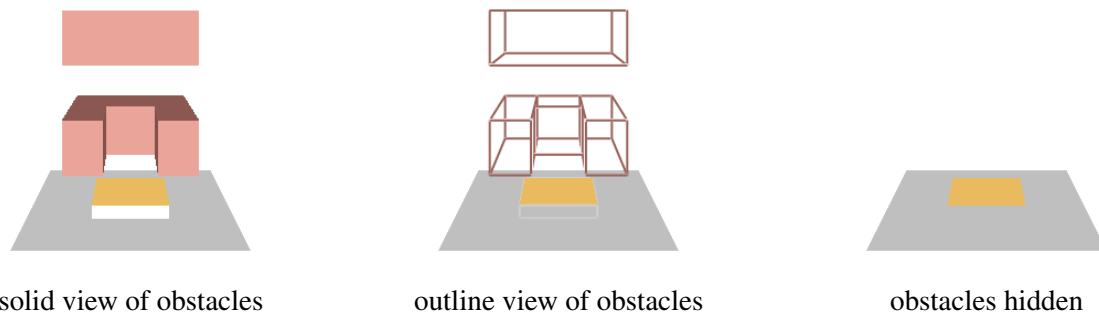


Figure 5.1: A test showing 3 drawing modes for representing blockages: solid, outline and hidden

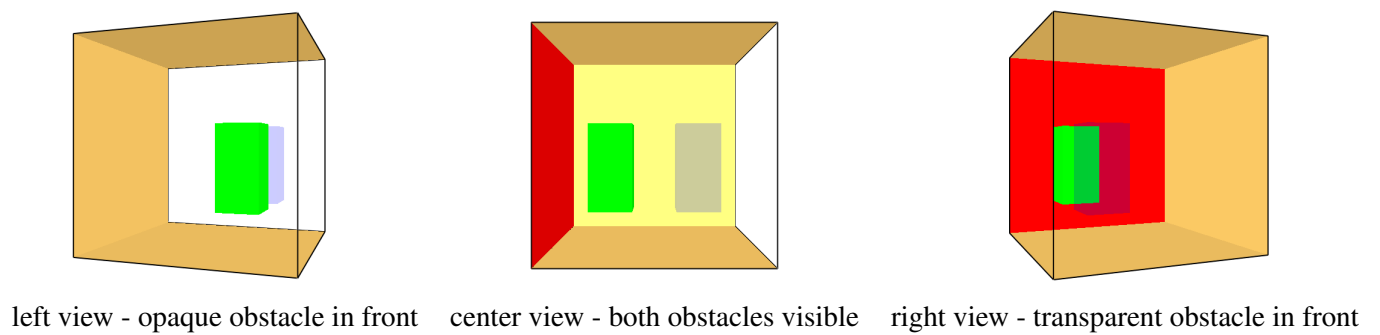


Figure 5.2: A test showing that transparent objects are drawn properly

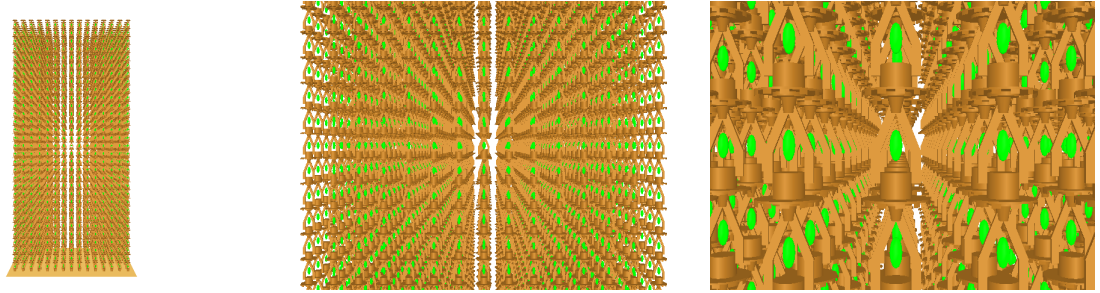


Figure 5.3: A test showing that devices are drawn properly

5.2 Devices

The case illustrated in Fig. 5.3 tests how well Smokeview can display many devices. This figure shows a series of sprinklers with three different levels of detail. Smokeview should be able to smoothly rotate this scene on computers with good video cards. The images for this test were created automatically by running the Smokeview script, `sprinkler_many.ssf` (see Appendix B.9).

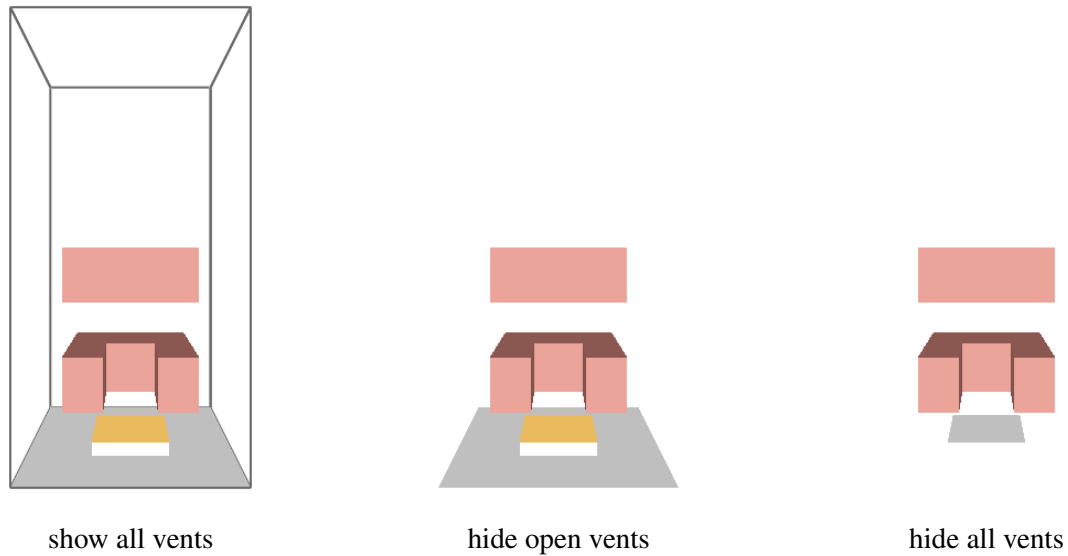


Figure 5.4: A test showing 3 modes for drawing vents: show all vents, hide open vents and hide all vents

5.3 Vents

Figure 5.4 tests vent display. This figure shows three different levels of vent visibility: all vents displayed, only non-open vents displayed and all vents hidden. The images for this test were created automatically by running the Smokeview script, `plume5c.ssf` (see Appendix B.4). Figure 5.5 shows circular vents drawn on the sides of a blockage. The circular vents are drawn as requested by the user (a circle) and as implemented within FDS. Figure 5.6 shows circular vents drawn on exterior boundary. The circular vents are drawn as requested by the user (a circle) and as implemented within FDS.

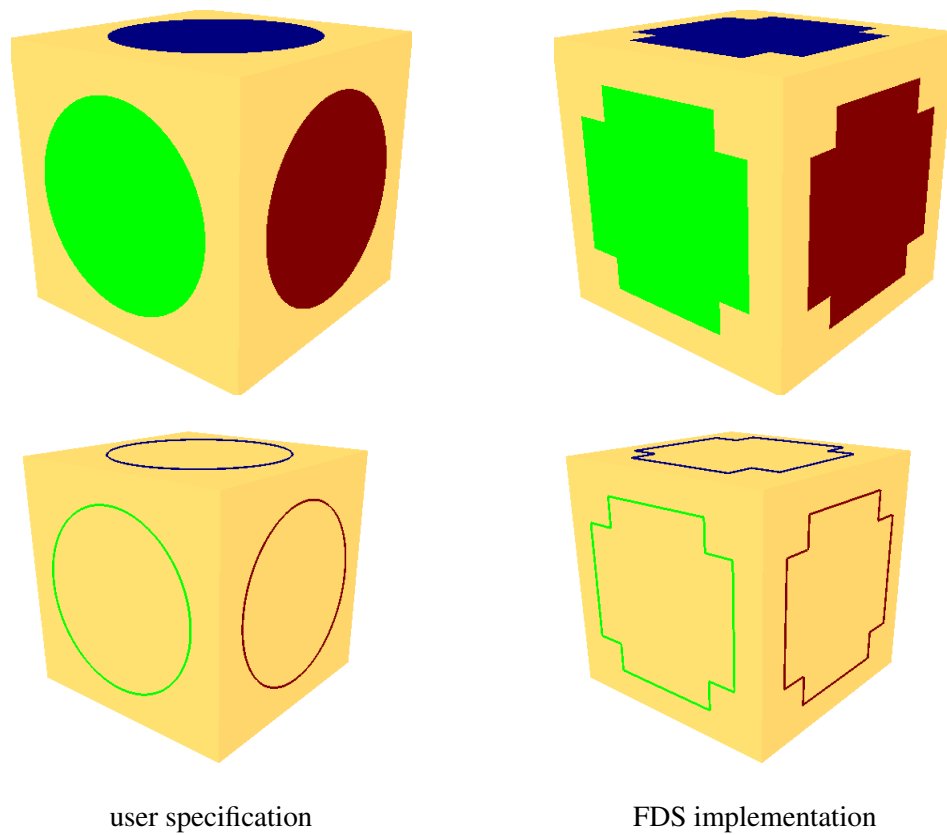


Figure 5.5: A test showing circular vents drawn as specified in the input file and as implemented within FDS. The circular vents are applied to a blockage.

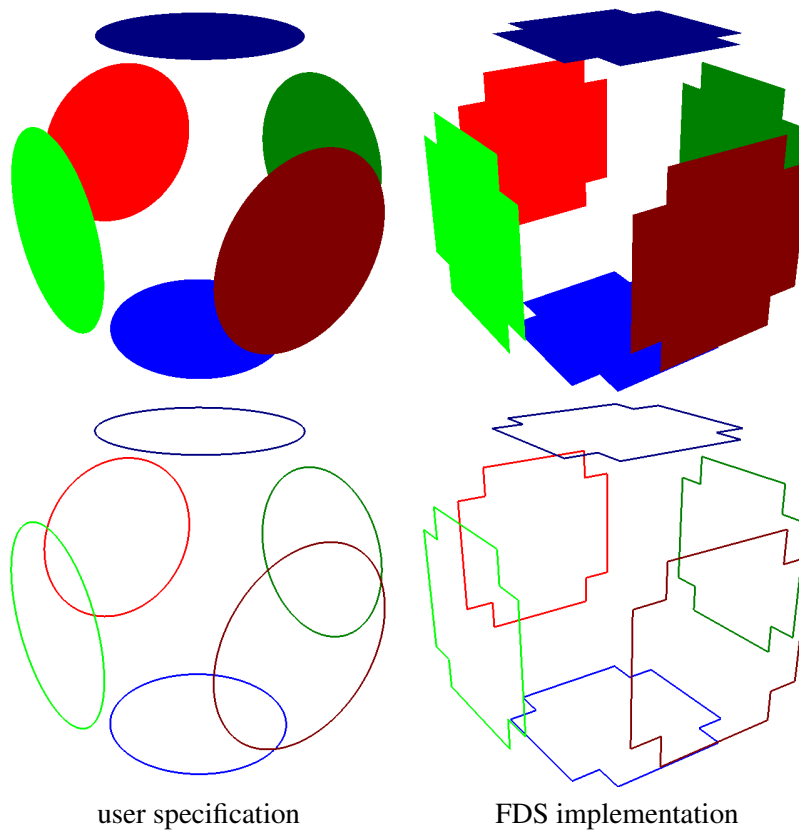


Figure 5.6: A test showing circular vents drawn as specified in the input file and as implemented within FDS. The circular vents are applied to the external boundary.

5.4 Conversion to Color

Smokeview converts data values obtained from an FDS calculation to color using a linear scaling of the form

$$C_i = 255 \frac{V_i - V_{\min}}{V_{\max} - V_{\min}} \quad (5.1)$$

where C_i is an index into a color table between 0 and 255, V_{\min} and V_{\max} are data bounds, and V_i is a data value to be converted. Figure 5.7 presents images verifying the conversion of data to colors. The input file, `colorconv.fds` (see Appendix A.2), was set up so that initially the left half of the domain (it is a 2D case) is 20 °C and the right half is 100 °C. The images for this test were created automatically by running the Smokeview script, `colorconv.ssf` (see Appendix B.2).

A second color conversion test involves the use of the colorbar to highlight portions of the data. The case was set up so that the vertical slice aligns with the colorbar. The gas temperatures were defined to increase from 20 °C to 100 °C from the floor to the ceiling. As a result, when one selects the colorbar, the selected region in the colorbar should match the selected region in the vertical slice. The images for this test were created automatically by running the Smokeview script, `colorbar.ssf` (see Appendix B.1).

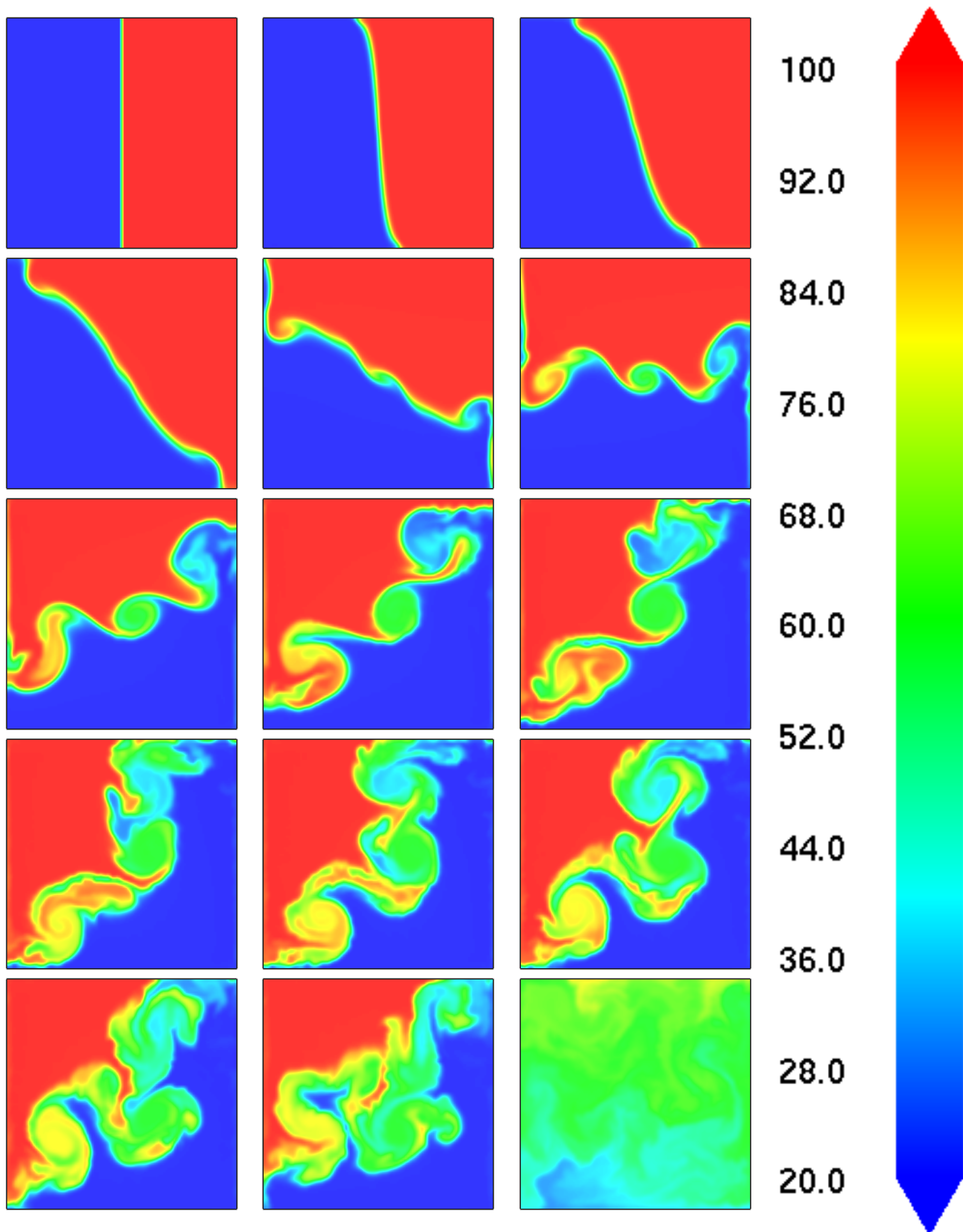


Figure 5.7: A test showing that data is converted to color properly. Temperature between 20 °C and 100 °C are converted to colors between blue and red.

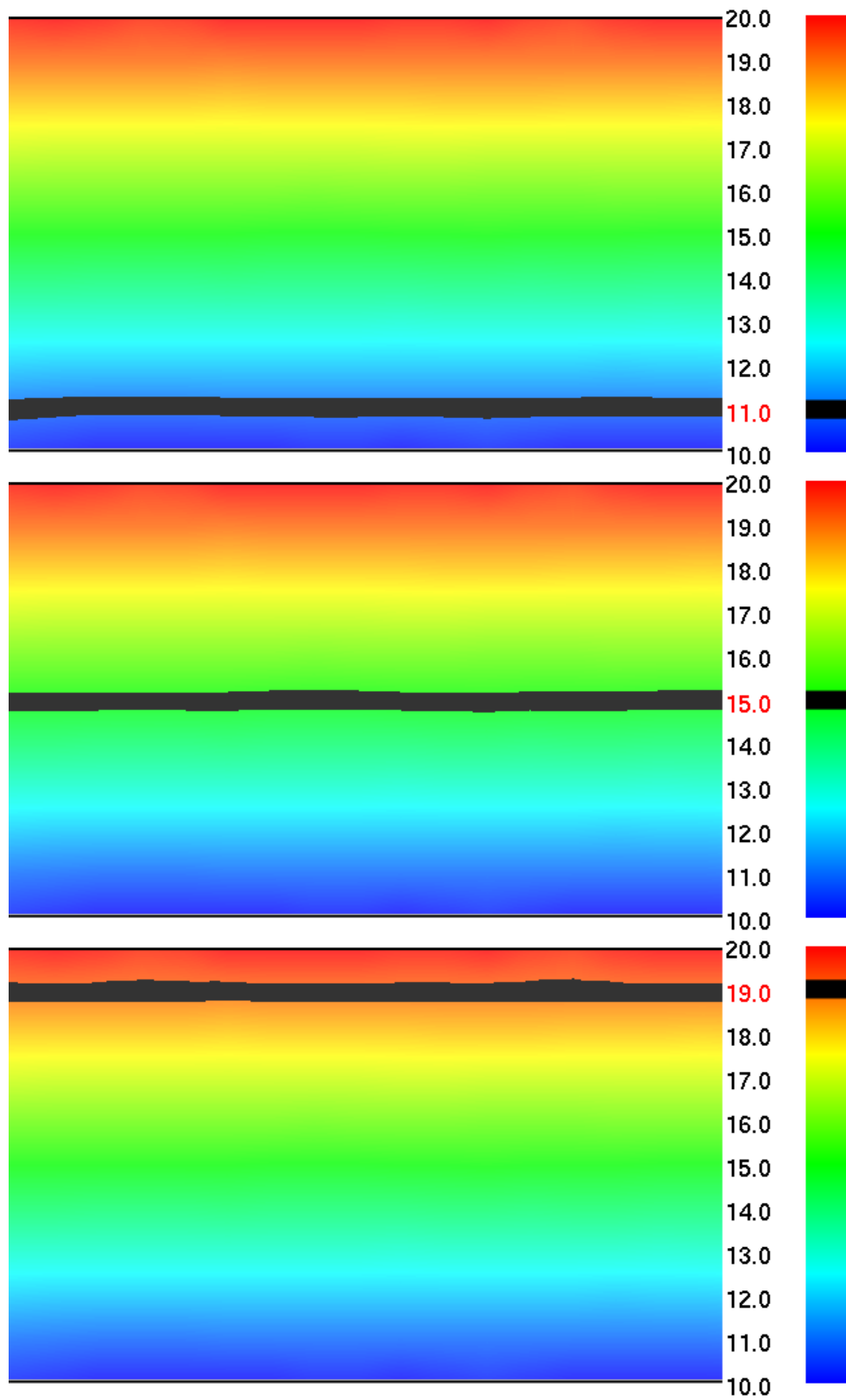


Figure 5.8: A test showing that data is converted to color properly. Temperature between 20 °C and 100 °C are converted to colors between blue and red. The highlighted region in the colorbar matches the highlighted region in the vertical slice.

5.5 GPU Test

Figure 5.9 tests whether Smokeview produces similar images when using the GPU and CPU for drawing 3D smoke. The first column shows CPU generated images at 5 s, 10 s and 30 s while the second column shows GPU generated images at the same times. The corresponding images in each column should be similar.

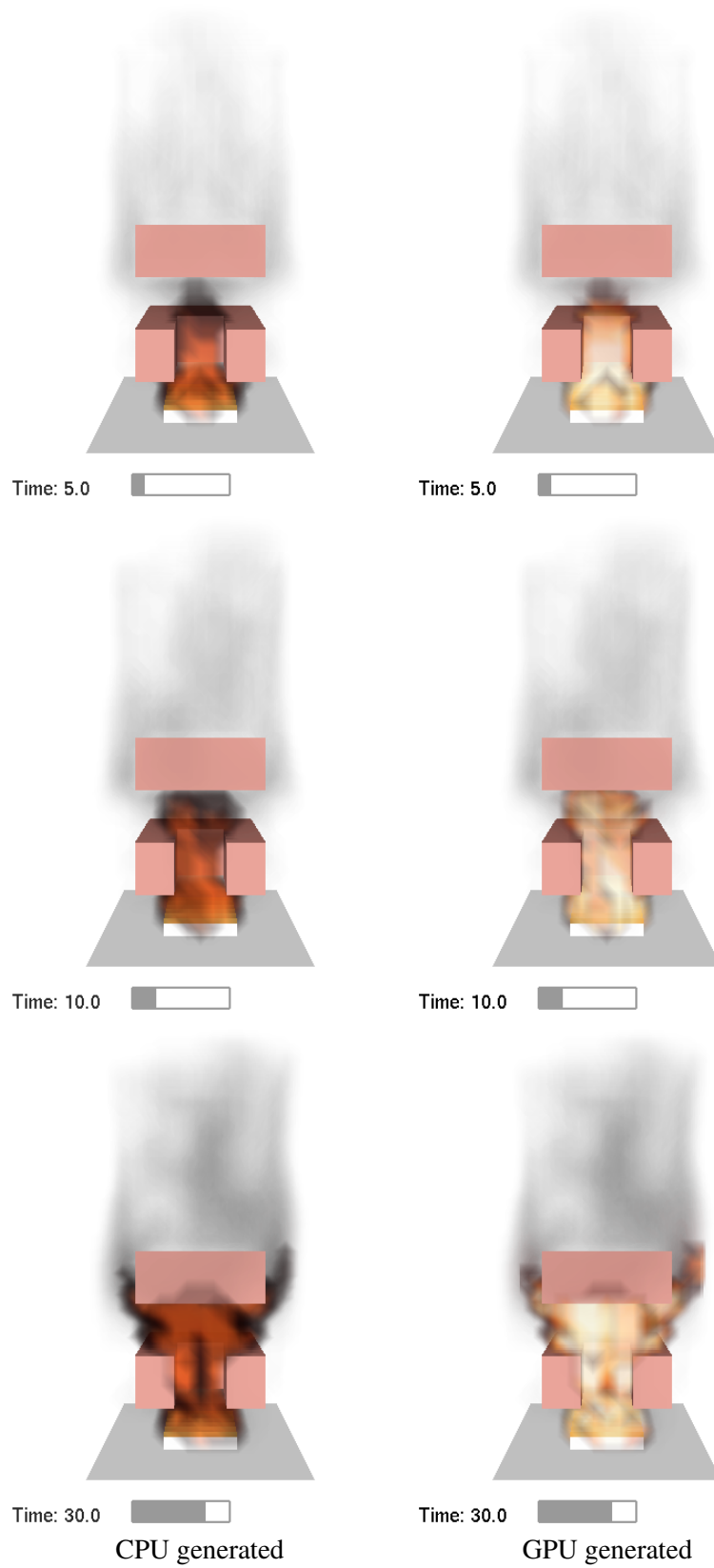


Figure 5.9: A test showing that the GPU is properly used to draw smoke

Bibliography

- [1] G.P. Forney. Smokeview (Version 6), A Tool for Visualizing Fire Dynamics Simulation Data, Volume I: User's Guide. NIST Special Publication 1017-1, National Institute of Standards and Technology, Gaithersburg, Maryland, May 2013. [i](#), [1](#)
- [2] G.P. Forney. Smokeview (Version 6), A Tool for Visualizing Fire Dynamics Simulation Data, Volume II: Technical Reference Guide. NIST Special Publication 1017-2, National Institute of Standards and Technology, Gaithersburg, Maryland, May 2013. [i](#), [1](#), [35](#)
- [3] G.P. Forney. Smokeview (Version 6), A Tool for Visualizing Fire Dynamics Simulation Data, Volume III: Verification Guide. NIST Special Publication 1017-3, National Institute of Standards and Technology, Gaithersburg, Maryland, May 2013. [i](#)
- [4] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, and K. Overholt. *Fire Dynamics Simulator; User's Guide*. National Institute of Standards and Technology, Gaithersburg, Maryland, USA, and VTT Technical Research Centre of Finland, Espoo, Finland, sixth edition, September 2013. [i](#), [1](#)
- [5] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, and K. Overholt. *Fire Dynamics Simulator, Technical Reference Guide, Volume 1: Mathematical Model*. National Institute of Standards and Technology, Gaithersburg, Maryland, USA, and VTT Technical Research Centre of Finland, Espoo, Finland, sixth edition, September 2013. [i](#), [1](#)
- [6] W. W. Jones, R. D. Peacock, G. P. Forney, and P. A. Reneke. CFAST, Consolidated Model of Fire Growth and Smoke Transport (Version 5. technical reference guide. NIST Special Publication 1030, National Institute of Standards and Technology, Gaithersburg, Maryland, October 2004. [1](#)
- [7] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, and K. Overholt. *Fire Dynamics Simulator; Technical Reference Guide, Volume 2: Verification*. National Institute of Standards and Technology, Gaithersburg, Maryland, USA, and VTT Technical Research Centre of Finland, Espoo, Finland, sixth edition, September 2013. [1](#), [29](#)
- [8] D.A. Purser. *SFPE Handbook of Fire Protection Engineering*, chapter Toxicity Assessment of Combustion Products. National Fire Protection Association, Quincy, Massachusetts, 3rd edition, 2002. [14](#)
- [9] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, and K. Overholt. *Fire Dynamics Simulator, Technical Reference Guide*. National Institute of Standards and Technology, Gaithersburg, Maryland, USA, and VTT Technical Research Centre of Finland, Espoo, Finland, sixth edition, September 2013. Vol. 1: Mathematical Model; Vol. 2: Verification Guide; Vol. 3: Validation Guide; Vol. 4: Configuration Management Plan. [35](#)

- [10] W. Mell, A. Maranghides, R. McDermott, and S. Manzello. Numerical simulation and experiments of burning Douglas fir trees. *Combustion and Flame*, 156:2023–2041, 2009. [35](#)
- [11] R. D. Peacock, G. P. Forney, and P. A. Reneke. CFAST – Consolidated Model of Fire Growth and Smoke Transport (Version 6): Technical Reference Guide. Special Publication 1026, National Institute of Standards and Technology, Gaithersburg, Maryland, July 2011. [35](#)

Appendix A

Input Files

A.1 colorbar

```
&HEAD CHID='colorbar', TITLE='Test of colorbar selection' /
REM This test case is used to test Smokeview's ability to
REM correctly select the colorbar and shade the appropriate
REM region black.

REM To do this,
REM 1) press "g" to turn on the grid.
REM 2) "Page Up" and put a horizontal grid slice "somewhere"
REM 3) select the colorbar, centering the resulting black band
REM    on the horizontal grid plane. Note the value.
REM 4) Use the File/BOunds dialog box to change the min and or max
REM    setting for the temperature slice file.
REM 5) repeaset setp 3. Value noted should remain the same.

&MESH IJK=20,10,10, XB=0.0,2.0,0.0,1.0,0.0,1.0 /

&TIME T_END=10.0 /

&MISC LAPSE_RATE=10.0, TMPA=10. /

&RADI RADIATION=.FALSE./

&SLCF PBY=0.6, QUANTITY='TEMPERATURE' /

&VENT MB='XMAX', SURF_ID='OPEN' /
&VENT MB='YMAX', SURF_ID='OPEN' /
&VENT MB='ZMAX', SURF_ID='OPEN' /
&VENT MB='XMIN', SURF_ID='OPEN' /
&VENT MB='YMIN', SURF_ID='OPEN' /
&VENT MB='ZMIN', SURF_ID='OPEN' /

&TAIL /
```

A.2 colorconv

A simple two-dimensional case testing data to color conversion

```
&HEAD CHID='colorconv', TITLE='Test data to color conversion, SVN $Revision: 12143 $' /
&DUMP NFRAMES=800 /

&MESH IJK=100,1,100, XB=0.0,100.0,-1.0,1.0,0.0,100.0 /
&SURF ID='COOLWALL' TMP_FRONT=20.0 /
```

```

&SURF ID='HOTWALL' TMP_FRONT=100.0 /
&SURF ID='INSWALL' ADIABATIC=.TRUE. /
&MISC FLUX_LIMITER=4, CFL_VELOCITY_NORM=1, CHECK_VN=.FALSE. /

&TIME T_END=200.0 /

&RADI RADIATION=.FALSE. /
&INIT XB= 0.0, 50.0,-2.0,2.0,0.0,100.0, TEMPERATURE=20. /
&INIT XB=50.0,100.0,-2.0,2.0,0.0,100.0, TEMPERATURE=100. /

&VENT XB= 0.0, 100.0, -1.0, 1.0, 0.0, 0.0, SURF_ID='COOLWALL' /
&VENT XB= 0.0, 100.0, -1.0, 1.0, 100.0, 100.0, SURF_ID='HOTWALL' /
&VENT XB= 0.0, 0.0, -1.0, 1.0, 0.0, 100.0 SURF_ID='INSWALL' /
&VENT XB= 100.0, 100.0, -1.0, 1.0, 0.0, 100.0 SURF_ID='INSWALL' /

&SLCF PBX=0.0, QUANTITY='TEMPERATURE' /

&TAIL /

```

A.3 fed_test

```

&HEAD CHID='fed_test', TITLE='Check FED computation in smokeview, SVN $Revision: 12143 $' /

&MESH XB=0.0,1.0,0.0,1.0,0.0,1.0, IJK=3,3,3 /

&TIME T_END=360.0, DT=0.1 /
&DUMP NFRAMES=2880 /

&SPEC ID='NITROGEN', BACKGROUND=.TRUE. /
&SPEC ID='OXYGEN' /
&SPEC ID='CARBON DIOXIDE' /
&SPEC ID='CARBON MONOXIDE' /

&INIT XB=0.0,1.0,0.0,1.0,0.0,1.0
      MASS_FRACTION(1)=0.078322, SPEC_ID(1)='OXYGEN',
      MASS_FRACTION(2)=0.030769, SPEC_ID(2)='CARBON DIOXIDE'
      MASS_FRACTION(3)=0.000979, SPEC_ID(3)='CARBON MONOXIDE' /

&DEVC ID='DEV1A', XYZ=0.5,0.5,0.6, QUANTITY='FED' /

&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON DIOXIDE' /
&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON MONOXIDE' /
&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='OXYGEN' /
&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON DIOXIDE', CELL_CENTERED=.TRUE. /
&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON MONOXIDE', CELL_CENTERED=.TRUE. /
&SLCF PBX=0.5,QUANTITY='VOLUME FRACTION' SPEC_ID='OXYGEN', CELL_CENTERED=.TRUE. /

&TAIL /

```

A.4 plume5c

```

&HEAD CHID='plume5c',TITLE='Plume whirl case SVN $Revision: 14561 $' /

      same as plume5a except there is a blockage in the middle of the scene to block the flow
      The purpose of this case is to demonstrate the curved flow (via streak lines) that results.

&MESH IJK=16,16,32, XB=0.0,1.6,0.0,1.6,0.0,3.2 /

&MISC HRRPUV_MAX_SMV=1300.0 /

&DUMP NFRAMES=400 DT_PL3D=8.0, DT_SL3D=0.1 /

&INIT XB=0.2,1.4,0.2,1.4,0.5,2.2 TEMPERATURE=600.0 /

```



```

&TIME T_END=40. / Total simulation time

&MATL ID = 'FABRIC'
FYI = 'Properties completely fabricated'
SPECIFIC_HEAT = 1.0
CONDUCTIVITY = 0.1
DENSITY = 100.0
N_REACTIONS = 1
NU_SPEC = 1.
SPEC_ID = 'PROPANE'
REFERENCE_TEMPERATURE = 350.
HEAT_OF_REACTION = 3000.
HEAT_OF_COMBUSTION = 15000. /

&MATL ID = 'FOAM'
FYI = 'Properties completely fabricated'
SPECIFIC_HEAT = 1.0
CONDUCTIVITY = 0.05
DENSITY = 40.0
N_REACTIONS = 1
NU_SPEC = 1.
SPEC_ID = 'PROPANE'
REFERENCE_TEMPERATURE = 350.
HEAT_OF_REACTION = 1500.
HEAT_OF_COMBUSTION = 30000. /

&SURF ID = 'UPHOLSTERY_LOWER'
FYI = 'Properties completely fabricated'
RGB = 151,96,88
BURN_AWAY = .FALSE.
MATL_ID(1:2,1) = 'FABRIC','FOAM'
THICKNESS(1:2) = 0.002,0.1
/

&SURF ID = 'UPHOLSTERY_UPPER'
FYI = 'Properties completely fabricated'
RGB = 151,96,88
BURN_AWAY = .FALSE.
TMP_FRONT = 600.0
/

&REAC C=3,H=8,SOOT_YIELD=0.01,FUEL='PROPANE'/
&SURF ID='BURNER',HRRPUA=600.0,PART_ID='tracers' / Ignition source

&VENT XB=0.5,1.1,0.5,1.1,0.1,0.1,SURF_ID='BURNER' / fire source on kitchen stove

&OBST XB=0.5,1.1,0.5,1.1,0.0,0.1 /
&OBST XB=0.3,1.3,0.3,1.3,0.4,0.8 SURF_ID='UPHOLSTERY_LOWER'/
&HOLE XB=0.6,1.0,0.2,0.8,0.3,0.9 /
&OBST XB=0.3,1.3,0.3,1.3,1.2,1.6 SURF_ID='UPHOLSTERY_UPPER' /

&VENT XB=0.0,1.6,0.0,0.0,0.0,3.2,SURF_ID='OPEN' /
&VENT XB=1.6,1.6,0.0,1.6,0.0,3.2,SURF_ID='OPEN' /
&VENT XB=0.0,1.6,1.6,1.6,0.0,3.2,SURF_ID='OPEN' /
&VENT XB=0.0,0.0,0.0,1.6,0.0,3.2,SURF_ID='OPEN' /
&VENT XB=0.0,1.6,0.0,1.6,3.2,3.2,SURF_ID='OPEN' /

&ISOFG QUANTITY='TEMPERATURE',VALUE(1)=100.0 / Show 3D contours of temperature at 100 C
&ISOFG QUANTITY='TEMPERATURE',VALUE(1)=200.0 / Show 3D contours of temperature at 200 C
&ISOFG QUANTITY='TEMPERATURE',VALUE(1)=620.0 / Show 3D contours of temperature at 620 C
&ISOFG QUANTITY='TEMPERATURE',VALUE(1:2)=200.0,400.0 / Show 3D contours of temperature at 200 C

&PART ID='tracers',MASSLESS=.TRUE.,
QUANTITIES(1:4)='U-VELOCITY','V-VELOCITY','W-VELOCITY'
SAMPLING_FACTOR=10 / Description of massless tracer particles. Apply at a
solid surface with the PART_ID='tracers'

X slices

```

```
&SLCF PBX=0.8,QUANTITY='TEMPERATURE',VECTOR=.TRUE. / Add vector slices colored by temperature
&SLCF PBX=0.8,QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. /
```

Y slices

```
&SLCF PBX=0.8,QUANTITY='TEMPERATURE',VECTOR=.TRUE. /
&SLCF PBX=0.8,QUANTITY='VOLUME FRACTION' SPEC_ID='PROPANE' /
&SLCF PBX=0.8,QUANTITY='VOLUME FRACTION' SPEC_ID='OXYGEN' /
&SLCF PBX=0.8,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON DIOXIDE' /
&SLCF PBX=0.8,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON MONOXIDE' /
&SLCF PBX=0.8,QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. /
```

Z slices

```
&SLCF PBZ=0.4,QUANTITY='TEMPERATURE',VECTOR=.TRUE. /
&SLCF PBZ=0.4,QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. /
&SLCF PBZ=1.6,QUANTITY='TEMPERATURE',VECTOR=.TRUE. /
&SLCF PBZ=1.6,QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. /
&SLCF PBZ=3.2,QUANTITY='TEMPERATURE',VECTOR=.TRUE. /
&SLCF PBZ=3.2,QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. /
```

3d slices

```
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2,QUANTITY='TEMPERATURE' ,VECTOR=.TRUE. /
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2, QUANTITY='TEMPERATURE',CELL_CENTERED=.TRUE. / 3D slice
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON DIOXIDE' /
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON MONOXIDE' /
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2,QUANTITY='VOLUME FRACTION' SPEC_ID='OXYGEN' /
&SLCF XB=0.0,1.6,0.0,1.6,0.0,3.2, QUANTITY='DENSITY',SPEC_ID='SOOT',CELL_CENTERED=.TRUE. / 3D slice
&SLCF PBX=1.6,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON DIOXIDE' /
&SLCF PBX=1.6,QUANTITY='VOLUME FRACTION' SPEC_ID='CARBON MONOXIDE' /
&SLCF PBX=1.6,QUANTITY='VOLUME FRACTION' SPEC_ID='OXYGEN' /
```

```
&BNDF QUANTITY='GAUGE HEAT FLUX' / Common surface quantities. Good for monitoring fire spread.
&BNDF QUANTITY='BURNING RATE' /
&BNDF QUANTITY='WALL TEMPERATURE' /
&BNDF QUANTITY='WALL TEMPERATURE' CELL_CENTERED=.TRUE. /
```

```
&DEVC XYZ=1.2,0.8,0.0 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,0.0 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,0.0 QUANTITY='W-VELOCITY' /
&DEVC XYZ=1.2,0.8,0.6 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,0.6 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,0.6 QUANTITY='W-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.2 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.2 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.2 QUANTITY='W-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.8 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.8 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,1.8 QUANTITY='W-VELOCITY' /
&DEVC XYZ=1.2,0.8,2.4 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,2.4 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,2.4 QUANTITY='W-VELOCITY' /
&DEVC XYZ=1.2,0.8,3.0 QUANTITY='U-VELOCITY' /
&DEVC XYZ=1.2,0.8,3.0 QUANTITY='V-VELOCITY' /
&DEVC XYZ=1.2,0.8,3.0 QUANTITY='W-VELOCITY' /
```

```
&TAIL /
```

A.5 sillytexture

```
&HEAD CHID='sillytexture', TITLE='Silly Test Case SVN $Revision: 12143 $' /
&MISC TEXTURE_ORIGIN=0.1,0.1,0.1 /

&MESH IJK=20,20,02, XB=0.0,1.0,0.0,1.0,0.0,1.0 /
```

```

&TIME T_END=0. /
&SURF ID      = 'TEXTURE 1'
      TEXTURE_MAP= 'nistleft.jpg'
      TEXTURE_WIDTH=0.6
      TEXTURE_HEIGHT=0.2 /

&SURF ID      = 'TEXTURE 2'
      TEXTURE_MAP= 'nistleft.jpg'
      TEXTURE_WIDTH=0.4
      TEXTURE_HEIGHT=0.2 /

&OBST XB=0.1,0.3,0.1,0.7,0.1,0.3, SURF_ID='TEXTURE 1' /
&OBST XB=0.5,0.9,0.3,0.7,0.1,0.5, SURF_ID='TEXTURE 2',
      TEXTURE_ORIGIN=0.5,0.3,0.1 /

&VENT XB=0.0,0.0,0.2,0.8,0.2,0.4, SURF_ID='TEXTURE 1',
      TEXTURE_ORIGIN=0.0,0.2,0.2 /
&VENT XB=0.3,0.9,0.0,0.0,0.3,0.5, SURF_ID='TEXTURE 1',
      TEXTURE_ORIGIN=0.9,0.0,0.3 /
&TAIL /

```

A.6 smoke_sensor

```

&HEAD CHID='smoke_sensor', TITLE='Test smokesensor device, SVN $Revision: 12143 $' /

A small white (255,255,255) smokesensor appears over top a grey (128,128,128) obstacle.
A red dot indicates where the smoke opacity is recorded and should appear in the middle of the
sensor. Another check is that the sensor should display 255 (the color fo the sensor) not
128 (the color of the background).

&MESH IJK=100,64,40, XB=0.0,10.0,0.0,6.4,0.0,4.0 /

&TIME T_END=1.0 /

&SPEC ID='SOOT',MASS_FRACTION_0=0.00007967/ DENSITY=1.0/

&OBST XB=0.0,10.0,6.3,6.4,0.0,4.0 RGB=128,128,128/
&PROP ID='smoketest' SMOKEVIEW_ID='smokesensor' /
&DEVC XYZ=8.5,6.0,0.50, ID='vis1' QUANTITY='TEMPERATURE' PROP_ID='smoketest' /

&TAIL /

```

A.7 smoke_test

```

&HEAD CHID='smoke_test', TITLE='Verify Smokeview Smoke3D Feature, SVN $Revision: 13127 $' /

A quantitative test of the smoke opacity calculation in Smokeview. This test simplifies
the general case by assuming a uniform distribution of smoke. Smoke grey levels are computed
using


$$\text{grey level (GL)} = 255 * \exp(-K * S * DX)$$


where K=8700 m2/kg is the mass extinction value, S=79.67 mg/m3 is the soot sensity
and DX is the path length through the smoke. This equation is inverted to obtain


$$DX = -\ln(GL/255) / (K * S)$$


and is used to place smoke sensors at particular distances so that the predicted
grey levels are 192, 128, 64, 32, 16 and 8 .

&MESH IJK=100,64,40, XB=0.0,10.0,0.0,6.4,0.0,4.0 /

&TIME T_END=1.0 /

```

```

&SPEC ID='SOOT',MASS_FRACTION_0=0.00007967 /

&OBST XB=0.0,2.6, 0.4,0.6,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,4.0, 1.0,1.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,5.6, 2.0,2.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,7.0, 3.0,3.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,8.6, 4.0,4.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,10.0,5.0,5.4,0.0,4.0, RGB=255,255,255 /

&PROP ID='smoketest' SMOKEVIEW_ID='smokesensor' /
&DEVC XYZ=1.8,0.4,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis1' PROP_ID='smoketest' /
&DEVC XYZ=3.2,1.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis2' PROP_ID='smoketest' /
&DEVC XYZ=4.8,2.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis3' PROP_ID='smoketest' /
&DEVC XYZ=6.2,3.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis4' PROP_ID='smoketest' /
&DEVC XYZ=7.8,4.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis5' PROP_ID='smoketest' /
&DEVC XYZ=9.2,5.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis6' PROP_ID='smoketest' /

&SLCF PBX=5.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF PBY=5.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF PBZ=2.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF XB=0.0,10.0,0.0,6.4,0.0,6.4,QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF XB=0.0,10.0,0.0,6.4,0.0,6.4,QUANTITY='TEMPERATURE' /

&TAIL /

```

A.8 smoke_test2

```

&HEAD CHID='smoke_test2', TITLE='Verify Smokeview Smoke3D Feature, SVN $Revision: 13136 $' /

```

A quantitative test of the smoke opacity calculation in Smokeview. This test simplifies the general case by assuming a uniform distribution of smoke. Smoke grey levels are computed using

$$\text{grey level (GL)} = 255 \cdot \exp(-K \cdot S \cdot DX)$$

where $K=8700 \text{ m}^2/\text{kg}$ is the mass extinction value, $S=79.67 \text{ mg}/\text{m}^3$ is the soot sensity and DX is the path length through the smoke. This equation is inverted to obtain

$$DX = -\ln(GL/255) / (K \cdot S)$$

and is used to place smoke sensors at particular distances so that the predicted grey levels are 192, 128, 64, 32, 16 and 8 .

```

&MESH IJK=50,32,20, XB=0.0,10.0,0.0,6.4,0.0,4.0 /

&TIME T_END=1.0 /

&SPEC ID='SOOT',MASS_FRACTION_0=0.00007967 /

&OBST XB=0.0,2.6, 0.4,0.6,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,4.0, 1.0,1.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,5.6, 2.0,2.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,7.0, 3.0,3.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,8.6, 4.0,4.2,0.0,4.0, RGB=255,255,255 /
&OBST XB=0.0,10.0,5.0,5.4,0.0,4.0, RGB=255,255,255 /

&PROP ID='smoketest' SMOKEVIEW_ID='smokesensor' /
&DEVC XYZ=1.8,0.4,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis1' PROP_ID='smoketest' /
&DEVC XYZ=3.2,1.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis2' PROP_ID='smoketest' /
&DEVC XYZ=4.8,2.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis3' PROP_ID='smoketest' /
&DEVC XYZ=6.2,3.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis4' PROP_ID='smoketest' /
&DEVC XYZ=7.8,4.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis5' PROP_ID='smoketest' /
&DEVC XYZ=9.2,5.0,2.00, QUANTITY='VISIBILITY', SPEC_ID='SOOT', ID='vis6' PROP_ID='smoketest' /

```

```

&SLCF PBX=5.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF PBY=5.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF PBZ=2.0, QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF XB=0.0,10.0,0.0,6.4,0.0,6.4,QUANTITY='DENSITY',SPEC_ID='SOOT' /
&SLCF XB=0.0,10.0,0.0,6.4,0.0,6.4,QUANTITY='TEMPERATURE' /

&TAIL /

```

A.9 transparency

```

&HEAD CHID='transparency', TITLE='Test of transparent vents SVN $Revision: 12143 $' /
&MESH IJK=10,10,10, XB=0.0,1.0,0.0,1.0,0.0,1.0 /

&TIME T_END=0.0 /

&SURF ID='BURNER',HRRPUA=600.0 /
&REAC C=3,H=8,SOOT_YIELD=0.01,FUEL='PROPANE' /
&MISC LAPSE_RATE=10.0, TMPA=10. /
&RADI RADIATION=.FALSE./
&SLCF PBY=0.6, QUANTITY='TEMPERATURE' /

&VENT MB='XMIN', SURF_ID='BURNER' RGB=255,0,0 TRANSPARENCY=0.6/
&VENT MB='XMAX', SURF_ID='OPEN' /

&OBST XB=0.2,0.4,0.2,0.4,0.2,0.6 RGB=0,255,0/
&OBST XB=0.6,0.8,0.2,0.4,0.2,0.6 RGB=0,0,255 TRANSPARENCY=0.2 /

&TAIL /

```

A.10 tree_test2

```

&HEAD CHID='tree_test2', TITLE='Test for showing tree objects' /

&MESH IJK=10,10,10, XB=0.0,1.0,0.0,1.0,0.0,1.0 /
&MISC HUMIDITY=0., Y_O2_INFTY = 0.01/
&TIME T_END=10., DT=0.01, WALL_INCREMENT=1 /

&VENT XB=0.0,0.0,0.0,1.0,0.0,1.0, SURF_ID='OPEN' /
&VENT XB=1.0,1.0,0.0,1.0,0.0,1.0, SURF_ID='OPEN' /
&VENT XB=0.0,1.0,0.0,0.0,0.0,1.0, SURF_ID='OPEN' /
&VENT XB=0.0,1.0,1.0,1.0,0.0,1.0, SURF_ID='OPEN' /
&VENT XB=0.0,1.0,0.0,1.0,1.0,1.0, SURF_ID='OPEN' /

&REAC FUEL = 'GLUCOSE'
      C = 6.
      H = 12.
      O = 6. /

&SPEC ID='WATER VAPOR' /

&SURF ID='HOT WALL', TMP_FRONT=1000., DEFAULT=.TRUE. /

&SURF ID
      MATL_ID(1,1:2) = 'wet vegetation'
      MATL_ID(1,1:2) = 'PINE','MOISTURE'
      MATL_MASS_FRACTION(1,1:2) = 0.8,0.2
      THICKNESS = 0.00025
      LENGTH = 0.1
      GEOMETRY = 'CYLINDRICAL' /

&MATL ID = 'PINE'

```

```

    DENSITY = 500.
    CONDUCTIVITY = 0.1
    SPECIFIC_HEAT= 1.0
    N_REACTIONS = 1
    REFERENCE_TEMPERATURE = 300.
    NU_MATL = 0.2
    NU_SPEC = 0.8
    SPEC_ID = 'GLUCOSE'
    HEAT_OF_REACTION= 1000
    MATL_ID = 'CHAR' /

&MATL ID = 'MOISTURE'
    DENSITY = 1000.
    CONDUCTIVITY = 0.1
    SPECIFIC_HEAT= 4.184
    N_REACTIONS = 1
    REFERENCE_TEMPERATURE = 100.
    NU_SPEC = 1.0
    SPEC_ID = 'WATER VAPOR'
    HEAT_OF_REACTION= 2500. /

&MATL ID = 'CHAR'
    DENSITY = 200.
    CONDUCTIVITY = 1.0
    SPECIFIC_HEAT = 1.6 /

&PART ID='pine needles', SAMPLING_FACTOR=1, SURF_ID='wet vegetation', PROP_ID='wood image'
    QUANTITIES='PARTICLE TEMPERATURE','PARTICLE MASS','PARTICLE DIAMETER', STATIC=.TRUE. /

&INIT PART_ID='pine needles', XB=0.,1.,0.,1.,0.,0.01, N_PARTICLES=10, MASS_PER_VOLUME=1. /

&PROP ID='wood image', SMOKEVIEW_ID='CANOPY', SMOKEVIEW_PARAMETERS='CANOPY_D=0.1','CANOPY_H=0.4' /

&DUMP MASS_FILE=.TRUE. /

&DEVC XB=0.,1.,0.,1.,0.,1., QUANTITY='DENSITY', ID='fuel gas mass', STATISTICS='VOLUME INTEGRAL', SPEC_ID='GLUCOSE'
&DEVC XB=0.,1.,0.,1.,0.,1., QUANTITY='DENSITY', ID='water vapor mass', STATISTICS='VOLUME INTEGRAL', SPEC_ID='WATER VAPOR'
&DEVC XB=0.,1.,0.,1.,0.,1., QUANTITY='MPUV', PART_ID='pine needles', ID='solid mass', STATISTICS='VOLUME INTEGRAL'

&SLCF PBY=0.5, QUANTITY='MASS FRACTION', SPEC_ID='WATER VAPOR' /
&SLCF PBY=0.5, QUANTITY='RELATIVE HUMIDITY' /

&TAIL /

```

A.11 vcirctest

```

&HEAD CHID='vcirctest', TITLE='Xin_velocity_specified' /

&MESH IJK=20,20,20, XB=0,2,0,2,0,2 / single coarse mesh for testing

&TIME T_END=10/

&MISC TMPA=15.
    TURBULENCE_MODEL = 'DEARDORFF'
    CHECK_VN=.TRUE. /

&REAC FUEL='METHANE'
    C=1.
    H=4.
    HEAT_OF_COMBUSTION=50350.
    SOOT_YIELD=0.01 /

&RADI RADIATIVE_FRACTION=0.1 /

&SURF ID='FLAME', COLOR='ORANGE' , VEL = -0.0314, MASS_FRACTION(1) = 1.0, SPEC_ID(1) = 'METHANE' /

```

```

&SURF ID='XMIN' RGB=255,0,0 /
&SURF ID='XMAX' RGB=128,0,0 /
&SURF ID='YMIN' RGB=0,255,0 /
&SURF ID='YMAX' RGB=0,128,0 /
&SURF ID='ZMIN' RGB=0,0,255/
&SURF ID='ZMAX' RGB=0,0,128 /

&VENT MB='XMIN', SURF_ID='OPEN' /
&VENT MB='XMAX', SURF_ID='OPEN' /
&VENT MB='YMIN', SURF_ID='OPEN' /
&VENT MB='YMAX', SURF_ID='OPEN' /
&VENT MB='ZMAX', SURF_ID='OPEN' /

&OBST XB=0.6,1.4,0.6,1.4,0.6,1.4 /
&VENT XB=0.6,0.6,0.6,1.4,0.6,1.4, XYZ=0.6,1.0,1.0,RADIUS=0.3, SURF_ID='XMIN' /
&VENT XB=1.4,1.4,0.6,1.4,0.6,1.4, XYZ=1.4,1.0,1.0,RADIUS=0.3, SURF_ID='XMAX' /
&VENT XB=0.6,1.4,0.6,0.6,0.6,1.4, XYZ=1.0,0.6,1.0,RADIUS=0.3, SURF_ID='YMIN' /
&VENT XB=0.6,1.4,1.4,1.4,0.6,1.4, XYZ=1.0,1.4,1.0,RADIUS=0.3, SURF_ID='YMAX' /
&VENT XB=0.6,1.4,0.6,1.4,0.6,0.6, XYZ=1.0,1.0,0.6,RADIUS=0.3, SURF_ID='ZMIN' /
&VENT XB=0.6,1.4,0.6,1.4,1.4,1.4, XYZ=1.0,1.0,1.4,RADIUS=0.3, SURF_ID='ZMAX' /

&VENT MB='ZMIN', SURF_ID='OPEN' /

&SLCF PBX=.05, QUANTITY='VELOCITY', VECTOR=.TRUE./
&SLCF PBX=.05, QUANTITY='MASS FRACTION', SPEC_ID='METHANE', CELL_CENTERED=.TRUE./

&BNDF QUANTITY='WALL TEMPERATURE' /
&BNDF QUANTITY='NET HEAT FLUX' /
&BNDF QUANTITY='RADIATIVE HEAT FLUX' /
&BNDF QUANTITY='WALL CELL COLOR',CELL_CENTERED=.TRUE./

&TAIL /

```

A.12 vcirctest2

```

&HEAD CHID='vcirctest2', TITLE='SMV-Test for RADIUS-Vent' /

&MESH IJK=10,10,10, XB= 0.0,1.0,0.0,1.0,0.0,1.0/

&TIME T_END=0.0/

&SURF ID='BURN' /

&VENT XB=0.0,0.0,0.0,1.0,0.0,1.0,RADIUS=0.3,XYZ=0.0,0.5,0.5,SURF_ID='BURN',RGB=255,0,0/
&VENT XB=1.0,1.0,0.0,1.0,0.0,1.0,RADIUS=0.3,XYZ=1.0,0.5,0.5,SURF_ID='BURN',RGB=128,0,0/
&VENT XB=0.0,1.0,0.0,0.0,0.0,1.0,RADIUS=0.3,XYZ=0.5,0.0,0.5,SURF_ID='BURN',RGB=0,255,0/
&VENT XB=0.0,1.0,1.0,1.0,0.0,1.0,RADIUS=0.3,XYZ=0.5,1.0,0.5,SURF_ID='BURN',RGB=0,128,0/
&VENT XB=0.0,1.0,0.0,1.0,0.0,0.0,RADIUS=0.3,XYZ=0.5,0.5,0.0,SURF_ID='BURN',RGB=0,0,255/
&VENT XB=0.0,1.0,0.0,1.0,1.0,1.0,RADIUS=0.3,XYZ=0.5,0.5,1.0,SURF_ID='BURN',RGB=0,0,128/

&VENT PBX=0.0, SURF_ID='OPEN' /
&VENT PBX=1.0, SURF_ID='OPEN' /
&VENT PBX=0.0, SURF_ID='OPEN' /
&VENT PBX=1.0, SURF_ID='OPEN' /
&VENT PBZ=0.0, SURF_ID='OPEN' /
&VENT PBZ=1.0, SURF_ID='OPEN' /

&TAIL /

```


Appendix B

Smokeview Scripts

B.1 colorbar

```
RENDERDIR
..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
LOADFILE
  colorbar_01.sf
SETTIMEVAL
  9.961175
LOADINIFILE
  colorbar_low.ini
SETVIEWPOINT
  view 1
SETVIEWPOINT
  view 1
RENDERONCE
  colorbar_low
RENDERONCE
  colorbar_low
LOADINIFILE
  colorbar_med.ini
SETVIEWPOINT
  view 1
SETVIEWPOINT
  view 1
RENDERONCE
  colorbar_med
RENDERONCE
  colorbar_med
LOADINIFILE
  colorbar_high.ini
SETVIEWPOINT
  view 1
SETVIEWPOINT
  view 1
RENDERONCE
  colorbar_high
RENDERONCE
  colorbar_high
```

B.2 colorconv

```
RENDERDIR
..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
LOADINIFILE
  colorconv_slice.ini
LOADFILE
```

```

    colorconv_01.sf
RENDERCLIP
  1 100 100 20 20
SETTIMEVAL
  0.000000
RENDERONCE
  colorconv_slice_00000
SETTIMEVAL
  2.557976
RENDERONCE
  colorconv_slice_00025
SETTIMEVAL
  5.039277
RENDERONCE
  colorconv_slice_00050
SETTIMEVAL
  7.506437
RENDERONCE
  colorconv_slice_00075
SETTIMEVAL
  10.023949
RENDERONCE
  colorconv_slice_00100
SETTIMEVAL
  12.538355
RENDERONCE
  colorconv_slice_00125
SETTIMEVAL
  15.063926
RENDERONCE
  colorconv_slice_00150
SETTIMEVAL
  17.526268
RENDERONCE
  colorconv_slice_00175
SETTIMEVAL
  20.040606
RENDERONCE
  colorconv_slice_00200
SETTIMEVAL
  22.521618
RENDERONCE
  colorconv_slice_00225
SETTIMEVAL
  25.042652
RENDERONCE
  colorconv_slice_00250
SETTIMEVAL
  27.545835
RENDERONCE
  colorconv_slice_00275
SETTIMEVAL
  30.060387
RENDERONCE
  colorconv_slice_00300
SETTIMEVAL
  32.531887
RENDERONCE
  colorconv_slice_00325
SETTIMEVAL
  200.0
RENDERONCE
  colorconv_slice_02000
UNLOADALL

```

B.3 fed_test

```
RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
XSCENECLIP
  0 -0.001000 0 1.001000
YSCENECLIP
  0 -0.001000 0 1.001000
ZSCENECLIP
  0 -0.001000 0 1.001000
SCENECLIP
  0
RENDERCLIP
  1 175 175 95 95
LOADFILE
  fed_test_05_fed.sf
SETTIMEVAL
  30.000000
RENDERONCE
  fed_test_030
SETTIMEVAL
  60.000000
RENDERONCE
  fed_test_060
SETTIMEVAL
  120.099998
RENDERONCE
  fed_test_120
SETTIMEVAL
  360.100006
RENDERONCE
  fed_test_360
```

B.4 plume5c

```
// put rendered files in specified directory

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES

UNLOADALL
LOADINIFILE
  plume5c_notexturebar.ini
LOADFILE
  plume5c_06.sf
SETTIMEVAL
  0.5
RENDERONCE
  plume5c_notexturebar
KEYBOARD
  T
RENDERONCE
  plume5c_texturebar
KEYBOARD
  (

// render slice files

UNLOADALL
LOADINIFILE
  plume5c_slice.ini
LOADFILE
  plume5c_06.sf
SETTIMEVAL
  0.0
```

```

RENDERONCE
  plume5c_slice_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_slice_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_slice_30

UNLOADALL
LOADINIFILE
  plume5c_gslice.ini
LOADFILE
  plume5c_30.sf
KEYBOARD
  *
KEYBOARD
  w
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_gslice_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_gslice_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_gslice_30

UNLOADALL
LOADINIFILE
  plume5c_gslice2.ini
LOADFILE
  plume5c_30.sf
KEYBOARD
  *
KEYBOARD
  w
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_gslice2_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_gslice2_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_gslice2_30

// render cell centered slice files

UNLOADALL
LOADINIFILE
  plume5c_slice_cell.ini
LOADFILE
  plume5c_14.sf
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_slice_cell_00
SETTIMEVAL
  10.05

```

```

RENDERONCE
  plume5c_slice_cell_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_slice_cell_30

// render cell centered slice files

UNLOADALL
LOADINIFILE
  plume5c_slice_cellchop.ini
LOADFILE
  plume5c_14.sf
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_slice_cellchop_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_slice_cellchop_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_slice_cellchop_30

// render slice files with data chopping

UNLOADALL
LOADINIFILE
  plume5c_slice_chop.ini
LOADFILE
  plume5c_06.sf
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_slice_chop_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_slice_chop_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_slice_chop_30

// render vector slice files

UNLOADALL
LOADINIFILE
  plume5c_vslice.ini
LOADVFILE
  plume5c_06.sf
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_vslice_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_vslice_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_vslice_30

// render vector slice files with chopping

```

```

UNLOADALL
LOADINIFILE
    plume5c_vslicechop.ini
LOADVFILE
    plume5c_06.sf
SETTIMEVAL
    0.0
RENDERONCE
    plume5c_vslicechop_00
SETTIMEVAL
    10.05
RENDERONCE
    plume5c_vslicechop_10
SETTIMEVAL
    30.05
RENDERONCE
    plume5c_vslicechop_30

// render iso files (solid)

UNLOADALL
UNLOADALL
LOADINIFILE
    plume5c_iso.ini
LOADFILE
    plume5c_01.iso
SETTIMEVAL
    0.0
RENDERONCE
    plume5c_iso_solid_00
SETTIMEVAL
    10.05
RENDERONCE
    plume5c_iso_solid_10
SETTIMEVAL
    30.05
RENDERONCE
    plume5c_iso_solid_30

LOADINIFILE
    plume5c_iso2.ini
LOADFILE
    plume5c_01.iso
SETTIMEVAL
    30.05
RENDERONCE
    plume5c_iso2_solid_30

// render iso files (outline)

LOADINIFILE
    plume5c_iso_outline.ini
UNLOADALL
LOADFILE
    plume5c_01.iso
SETTIMEVAL
    0.0
RENDERONCE
    plume5c_iso_outline_00
SETTIMEVAL
    10.05
RENDERONCE
    plume5c_iso_outline_10
SETTIMEVAL
    30.05
RENDERONCE
    plume5c_iso_outline_30

```

```

// render iso files (points)

LOADINIFILE
  plume5c_iso_points.ini
UNLOADALL
LOADFILE
  plume5c_01.iso
SETTIMEVAL
  0.0
RENDERONCE
  plume5c_iso_points_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_iso_points_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_iso_points_30

// render particle files using points

LOADINIFILE
  plume5c_part.ini
UNLOADALL
LOADFILE
  plume5c.prt5
PARTCLASSCOLOR
  Uniform color
SETTIMEVAL
  1.05
RENDERONCE
  plume5c_part_01
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_part_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_part_30
RENDERONCE
  plume5c_part

// render particle files using streaks

UNLOADALL
LOADINIFILE
  plume5c_part_streak.ini
LOADFILE
  plume5c.prt5
PARTCLASSCOLOR
  Uniform color
SETTIMEVAL
  1.05
RENDERONCE
  plume5c_part_streak_01
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_part_streak_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_part_streak_30
RENDERONCE
  plume5c_streak

```

```
// render particles using streaks (different length)
```

```
UNLOADALL
LOADINIFILE
  plume5c_part_streak2.ini
LOADFILE
  plume5c.prt5
PARTCLASSCOLOR
  Uniform color
SETTIMEVAL
  1.05
RENDERONCE
  plume5c_part_streak2_01
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_part_streak2_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_part_streak2_30
```

```
// render boundary files
```

```
UNLOADALL
LOADINIFILE
  plume5c_bound.ini
LOADFILE
  plume5c_03.bf
SETTIMEVAL
  2.0
RENDERONCE
  plume5c_bound_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_bound_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_bound_30
```

```
// render cell centered boundary files
```

```
UNLOADALL
LOADINIFILE
  plume5c_bound_cellcenter.ini
LOADFILE
  plume5c_04.bf
SETTIMEVAL
  2.0
RENDERONCE
  plume5c_bound_cell_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_bound_cell_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_bound_cell_30
```

```
// render chopped boundary files
```

```
UNLOADALL
LOADINIFILE
  plume5c_bound_chop.ini
```



```

LOADFILE
  plume5c_03.bf
SETTIMEVAL
  2.0
RENDERONCE
  plume5c_bound_chop_00
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_bound_chop_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_bound_chop_30

// render 3D smoke files

UNLOADALL
LOADFILE
  plume5c_01.s3d
LOADFILE
  plume5c_02.s3d
SETTIMEVAL
  1.05
RENDERONCE
  plume5c_smoke_01
SETTIMEVAL
  10.05
RENDERONCE
  plume5c_smoke_10
SETTIMEVAL
  30.05
RENDERONCE
  plume5c_smoke_30

// render PLOT3D stepped contours

UNLOADALL
LOADINIFILE
  plume5c_plot3d_step.ini
LOADPLOT3D
  1 40.0
RENDERONCE
  plume5c_plot3d_step

// render PLOT3D continuous contours

UNLOADALL
LOADINIFILE
  plume5c_plot3d_shaded.ini
LOADPLOT3D
  1 40.0
RENDERONCE
  plume5c_plot3d_shaded

// render OBSTs using solid view

UNLOADALL
LOADINIFILE
  plume5c_solid.ini
RENDERONCE
  plume5c_solid

// render OBSTs using outline view

LOADINIFILE
  plume5c_outline.ini

```

```

RENDERONCE
    plume5c_outline

// render geometry using hidden view (ie don't draw OBSTs)

LOADINIFILE
    plume5c_hidden.ini
RENDERONCE
    plume5c_hidden

// render vents - hide open vents

LOADINIFILE
    plume5c_noopenvents.ini
SETVIEWPOINT
    external
RENDERONCE
    plume5c_noopen

// render vents - don't show any vents

LOADINIFILE
    plume5c_novents.ini
SETVIEWPOINT
    external
RENDERONCE
    plume5c_novents

// render vents - show all vents

LOADINIFILE
    plume5c_allvents.ini
SETVIEWPOINT
    external
RENDERONCE
    plume5c_allvents

// render 3D smoke GPU off

UNLOADALL
LOADINIFILE
    plume5c_nongpu.ini
LOADFILE
    plume5c_01.s3d
LOADFILE
    plume5c_02.s3d
SETTIMEVAL
    5.05
RENDERONCE
    plume5c_smoke_nongpu_05
SETTIMEVAL
    10.05
RENDERONCE
    plume5c_smoke_nongpu_10
SETTIMEVAL
    30.05
RENDERONCE
    plume5c_smoke_nongpu_30

// render 3D smoke GPU on

UNLOADALL
LOADINIFILE
    plume5c_gpu.ini
LOADFILE
    plume5c_01.s3d
LOADFILE
    plume5c_02.s3d

```

```

SETTIMEVAL
5.05
RENDERONCE
  plume5c_smoke_gpu_05
SETTIMEVAL
10.05
RENDERONCE
  plume5c_smoke_gpu_10
SETTIMEVAL
30.05
RENDERONCE
  plume5c_smoke_gpu_30

// render PLOT3D contours

UNLOADALL
LOADINIFILE
  plume5c_plot3d.ini
LOADPLOT3D
  1 40.009998
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 1.100000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_y1
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.800000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_y2
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.500000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_y3
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.200000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_y4

```

```

// render PLOT3D vectors

LOADINIFILE
  plume5c_plot3dvector.ini
LOADPLOT3D
  1 40.009998
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 1.100000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 1 -1 1 20.0
RENDERONCE
plume5c_plot3d_v1
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.800000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 1 -1 1 20.0
RENDERONCE
plume5c_plot3d_v2
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.500000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 1 -1 1 20.0
RENDERONCE
plume5c_plot3d_v3
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.200000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 1 -1 1 20.0
RENDERONCE
plume5c_plot3d_v4

// render PLOT3D iso-surface

SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 0 0.400000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 1 0
PLOT3DPROPS
  1 0 6 1

```

```

RENDERONCE
plume5c_plot3d_i1
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 0 0.400000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 1 3
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_i2
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 0 0.400000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 1 6
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_i3
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 0 0.400000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 1 9
PLOT3DPROPS
  1 0 6 1
RENDERONCE
plume5c_plot3d_i4

// render PLOT3D line contours

LOADINIFILE
  plume5c_plot3dline.ini
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 1.100000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 2
RENDERONCE
plume5c_plot3d_l1
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.800000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 2
RENDERONCE
plume5c_plot3d_l2

```

```

SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.500000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 2
RENDERONCE
plume5c_plot3d_13
SHOWPLOT3DDATA
  1 1 1 0 0.800000
SHOWPLOT3DDATA
  1 2 1 1 0.200000
SHOWPLOT3DDATA
  1 3 1 0 1.600000
SHOWPLOT3DDATA
  1 4 1 0 6
PLOT3DPROPS
  1 0 6 2
RENDERONCE
plume5c_plot3d_14

```

B.5 sillytexture

```

RENDERDIR
  ..\..\Manuals\SMV_User_Guide\SCRIPT_FIGURES
SETVIEWPOINT
  external
RENDERONCE
  sillytexture

```

B.6 smoke_sensor

```

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
SETVIEWPOINT
  view 1
RENDERONCE
  smoke_sensor

```

B.7 smoke_test

```

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
UNLOADALL
LOADFILE
  smoke_test_01.s3d
SETTIMEVAL
  0.159000
LOADINIFILE
  smoke_test_all.ini
RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test_all
LOADINIFILE
  smoke_test_every2.ini

```

```

RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test_every2
LOADINIFILE
  smoke_test_every3.ini
RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test_every3
UNLOADALL
LOADINIFILE
  smoke_test_geom.ini
RENDERCLIP
  1 85 151 20 19
SETVIEWPOINT
  view 1
RENDERONCE
  smoke_test_side

```

B.8 smoke_test2

```

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
UNLOADALL
LOADFILE
  smoke_test2_01.s3d
SETTIMEVAL
  0.159000
LOADINIFILE
  smoke_test2_all.ini
RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test2_all
LOADINIFILE
  smoke_test2_every2.ini
RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test2_every2
LOADINIFILE
  smoke_test2_every3.ini
RENDERCLIP
  1 30 30 125 125
RENDERONCE
  smoke_test2_every3

```

B.9 sprinkler_many

```

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
SETVIEWPOINT
  view1
RENDERONCE
  sprink_many_view1
RENDERONCE
  sprink_many_view1
SETVIEWPOINT
  view2
RENDERONCE
  sprink_many_view2
SETVIEWPOINT

```

```
view3
RENDERONCE
  sprink_many_view3
```

B.10 transparency

```
RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
SETVIEWPOINT
  view_left
RENDERONCE
  transparency_left
RENDERONCE
  transparency_left
SETVIEWPOINT
  view_center
RENDERONCE
  transparency_center
SETVIEWPOINT
  view_right
RENDERONCE
  transparency_right
```

B.11 tree_test2

```
RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
XSCENECLIP
  0 -0.001000 0 1.001000
YSCENECLIP
  0 -0.001000 0 1.001000
ZSCENECLIP
  0 -0.001000 0 1.001000
SCENECLIP
  0
LOADFILE
  tree_test2.prt5
PARTCLASSCOLOR
  PARTICLE TEMPERATURE
SETTIMEVAL
  0.000000
RENDERONCE
  tree_test2_part_00
SETTIMEVAL
  3.000000
RENDERONCE
  tree_test2_part_03
SETTIMEVAL
  6.000000
RENDERONCE
  tree_test2_part_06
SETTIMEVAL
  9.000000
RENDERONCE
  tree_test2_part_09
```

B.12 vcirctest

```
RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
```



```

XSCENECLIP
  0 -0.002000 0 2.002000
YSCENECLIP
  0 -0.002000 0 2.002000
ZSCENECLIP
  0 -0.002000 0 2.002000
SCENECLIP
  0
RENDERCLIP
  1 87 126 36 34
SETVIEWPOINT
  view 1
RENDERONCE
  vcirctest_fds
KEYBOARD
  q
RENDERONCE
  vcirctest_circ
LOADINIFILE
  vcirctest_outline.ini
SETVIEWPOINT
  view 1
RENDERONCE
  vcirctest_fds_outline
KEYBOARD
  q
RENDERONCE
  vcirctest_circ_outline

```

B.13 vcirctest2

```

// put rendered files in specified directory

RENDERDIR
  ..\..\Manuals\SMV_Verification_Guide\SCRIPT_FIGURES
XSCENECLIP
  0 -0.001000 0 1.001000
YSCENECLIP
  0 -0.001000 0 1.001000
ZSCENECLIP
  0 -0.001000 0 1.001000
SCENECLIP
  0
RENDERCLIP
  1 217 187 81 71
RENDERONCE
  vcirctest2_circ
KEYBOARD
  q
RENDERONCE
  vcirctest2_fds
LOADINIFILE
  vcirctest2_outline.ini
RENDERCLIP
  1 217 187 81 71
RENDERONCE
  vcirctest2_circ_outline
KEYBOARD
  q
RENDERONCE
  vcirctest2_fds_outline

```