A Topological Framework for the Interactive Exploration of Large Scale Turbulent Combustion

Peer-Timo Bremer[1], Gunther H. Weber[2], Julien Tierny[3], Valerio Pascucci[3],
Marcus S. Day[2] and John B. Bell[2]
[1] Lawrence Livermore National Laboratory, [2] Lawrence Berkeley National Laboratory
[3] Scientific Computing and Imaging Institute, University of Utah

Abstract

The advent of highly accurate, large scale volumetric simulations has made data analysis and visualization techniques an integral part of the modern scientific process. To develop new insights from raw data, scientists need the ability to define features of interest in a flexible manner and to understand how changes in the feature definition impact the subsequent analysis of the data. Therefore, simply exploring the raw data is not sufficient.

This paper presents a new topological framework for the analysis of large scale, time-varying, turbulent combustion simulations. It allows the scientists to interactively explore the complete parameter space of fuel consumption thresholds for an entire time-dependent combustion simulation. By computing augmented merge trees *and their corresponding data segmentations, the system allows the user complete flexibility to segment, select, and track burning cells through time thanks to a linked view interface. We developed this technique in the context of low-swirl turbulent pre-mixed flame simulation analysis, where the topological abstractions enable an efficient tracking through time of the burning cells and provide new qualitative and quantitative insights into the dynamics of the combustion process.*

1 Introduction

High resolution numerical volumetric simulations have become an integral part of the scientific process. They allow scientists to observe a range of phenomena not easily captured by experiments and are an essential tool to develop and validate new scientific theories. However, as the spatial and temporal resolution of volumetric simulations increases so does the need for efficient methods to visualize and analyze the data. Traditionally, techniques such as iso-surface extraction [12] have been used to help scientists identify features of interest and their defining parameters. These fea-

tures are then extracted and analyzed using a secondary tool chain. However, after each change in parameters (typically in isovalues) the entire time-dependent simulation must be re-processed requiring significant time and effort.

This paper presents a new interactive framework for the analysis of large scale turbulent combustion simulations. Unlike previous scientific data analysis techniques [3, 17, 13], it allows the application scientists to interactively explore an entire one-parameter family of features thanks to a concise topological abstraction which is two orders of magnitude smaller than the input simulation data. Subsequently, the identification, tracking, and visualization of the burning cells (features of interest in combustion analysis) are made entirely interactive and flexible, enabling the scientists on-the-fly redefinitions of the burning cell detection criterion (fuel consumption burning threshold).

We have tested and validated our framework by processing more than 8 Terabytes of raw data. In collaboration with application scientists, we are actively using the system to form new hypotheses about the burning process. In particular, scientists have found that our technique can facilitate understanding the relationship between the size distribution of the burning regions, the threshold of fuel consumption, and the turbulence intensity.

Motivation: Understanding Turbulent Combustion. This research is motivated by the need for understanding the turbulence of low-swirl, pre-mixed hydrogen flames, used in the design of new low-emission burners. In such systems, flames burn in a dynamic cellular mode that is characterized by intensely burning cells separated by extinct regions. Burning cells are defined via a *burning threshold* on the fuel consumption rate and scientists are interested in the number, size, and temporal evolution of the cells. However, since no single "correct" burning threshold is known a priori, any analysis technique must allow for fast and easy changes of the threshold. Consequently, combustion scientists need new techniques for the interactive analysis of large scale turbulent combustion simulations, supporting interactive fea-
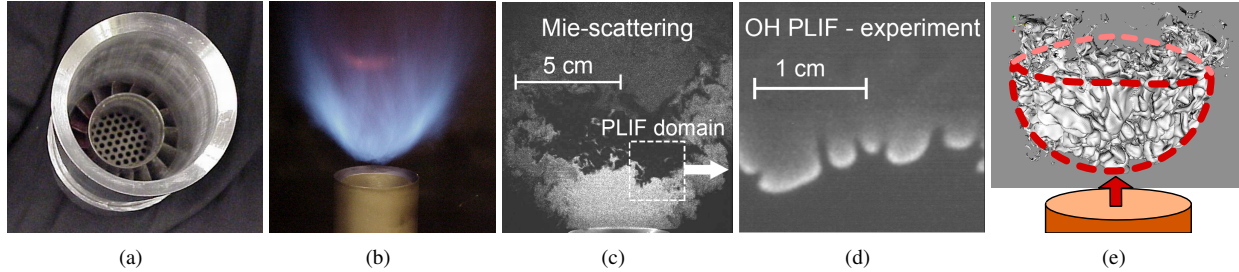
Figure 1. (a) Typical laboratory low-swirl nozzle. (b) Lean premixed CH₄ low-swirl flame. (c) Experimental Mie scattering image of a lean premixed H₂ flame. (d) PLIF data imaging the OH concentration in a lean premixed H₂ flame. (e) Rendering of the burning cells of the SwirlH2 simulation data. The cells form a bowl shaped structure with the arrow indicating the direction of the fuel stream. Highly wrinkled flame surfaces respond in a complex way to turbulent structures in the inlet flow-field.

ture detection, visualization, and tracking through time as well as interactive feature redefinition possibilities.

Related work. Isosurfaces [12], interval volumes [8] or thresholding combinations of various scalar quantities [15] have been used for the analysis and visualization of scientific data and as the building blocks of feature detection techniques. For example, in the case of combustion analysis, isosurfaces of temperature are often equated with flame surfaces. However, due to the massive size of the data and the absence of a fixed burning threshold, such techniques are costly to apply since any change in the threshold requires the re-processing of the entire time-dependent simulation.

Topological techniques address this problem by expressing a similar set of features using concepts of Morse theory [14]. Since Morse theory also provides a well developed notion of simplification, this makes the resulting feature analysis techniques hierarchical, allowing noise removal and multi-scale analysis. For example, the visualization community has used the Reeb graph [13], the contour tree [1, 5] or the Morse-Smale complex [9, 11] to define and extract features of interest. However, none of these techniques have focused specifically on the ability to handle interactive feature definition, detection and tracking for large-scale time-varying scientific data.

Contributions. In particular, this paper presents the following new contributions:

1. A fully interactive visualization system that allows scientists to easily explore, both in a qualitative and quantitative manner, large scale turbulent combustion simulations.

2. A new concise topology-based data representation that reduces the data size by more than two orders of magnitude (while maintaining all feature information).

3. A linked-view [16] data exploration interface, coupling 3D visualization with symbolic graph representa-

tations of the topology of burning cells through time, for real-time feature definition, detection and tracking.

Finally, we demonstrate the utility of our framework by describing a case study where our new framework enables experts to develop and validate new insights on the combustion process, especially regarding the relation between the size distribution of the burning regions and the turbulence of the flame.

2   Turbulent combustion simulations

Challenges. The considered combustion research targets the characterization of low-swirl, turbulent flames [2] burning a lean, premixed fuel through detailed simulations (Figure 1). The simulations are designed to augment and validate laser-based diagnostics, assess the underlying assumptions in their interpretation, and aid the development of models to characterize the salient behavior of these flame systems. Low-swirl injectors are emerging as an important new combustion technology with the potential for dramatically reducing pollutant emissions in turbines designed for stationary power generation. This design has the potential to stabilize lean hydrogen-air flames enabling the use of hydrogen for low emission power generation. However, low-swirl flames are inherently unstable, characterized by localized cells of intense burning that are separated by regions of complete flame extinction. Existing approaches to analyze the dynamics of such flames assume that the flame is a connected interface, which separates cold fuel from hot combustion products. In cellular hydrogen-air flames, however, many of the basic definitions break down. There is no connected interface between the fuel and products. In fact there is no concrete notion of a "progress variable" that can be used to normalize the progress of the combustion reactions through the flame. Development of models for cellular flames requires a new paradigm of flame analysis.

Simulation data. Results from the simulation are stored as a sequence of snapshots in time.Each snapshot incorpo-
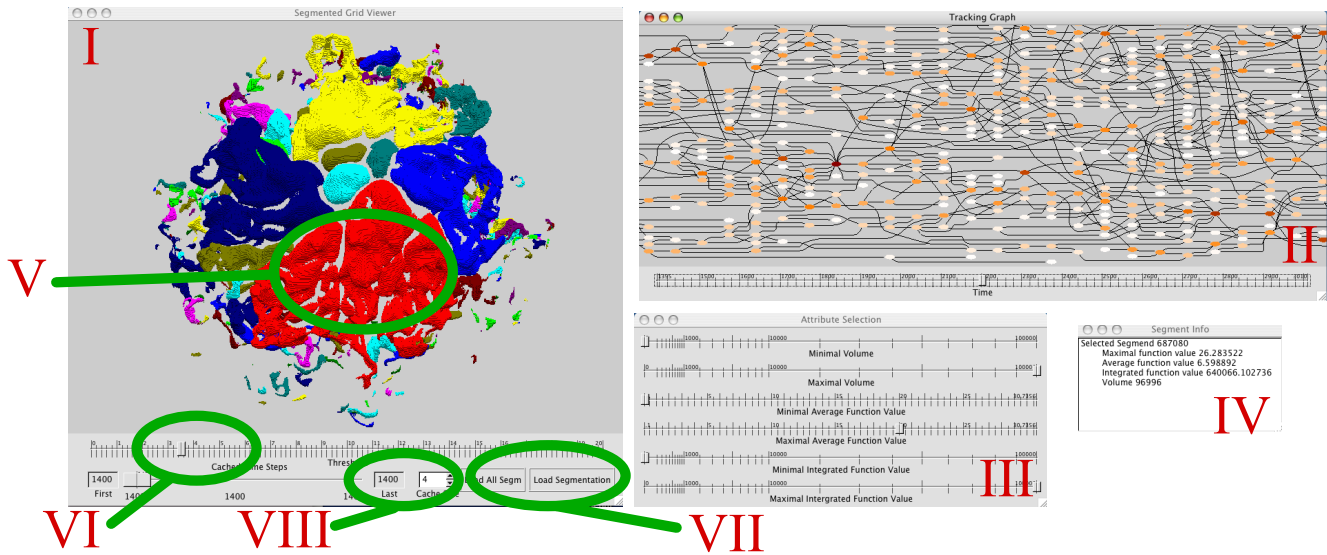
Figure 2. User interface of our framework: (I) 3D display of the segmentation including a slider to select the fuel consumption threshold (VI), the interface to determine the number of in-memory time steps (VIII), and the button to load the geometry (VII); (II) Interactive display of the tracking graph. Selecting node in either the 3D viewer or the graph display causes the corresponding cell to be highlighted (V) and its attribute information to be displayed in the info window (IV). The last window (III) provides the ability to sub-select segments based on attribute values.

rates a detailed description of the chemical kinetics (including the local fuel consumption rate). Also, each snapshot is arranged in a hierarchy of block-structured grid patches ordered by refinement level and tiling successively smaller regions of the domain with successively finer grid cells. Dynamic adaptivity focuses the highest resolution around the flame surface (or, region of high fuel consumption) and regions of high vorticity. As in the experiment, the simulation stabilizes a statistically stationary flame in a time-dependent turbulent flow field. We consider two simulations with different flow profiles called SwirlH2 and SwirlH2Fast where the "fast" version has an inlet flow profile 2.5 times faster and more turbulent than the "slow" one. The data sets consist of 332 and 284 snapshots for the slow and fast version, respectively, simulated with an effective resolution of 1024^3. The resulting snapshots are roughly 12 – 16 Gigabytes in size totaling a combined 8.4 Terabytes of raw data. The main features of interest are the intensely burning cells defined by a threshold on the local fuel consumption rate. All regions with fuel consumption above this threshold are considered burning, all others non-burning. However, no single "correct" threshold exists and to better understand the dynamics dependent of the threshold is of primary concern.

Framework overview Our framework encompasses two main components:

1. **Comprehensive topology analyzer:** The first component pre-processes the entire simulation and extracts a compact internal representation of the data. For each time-step, it computes the *augmented merge trees* (Section 3) of the fuel consumption rate and create the corresponding *meta-segmentations* of the volumetric data.

2. **Interactive exploration viewer:** The second component of the system consists of an interactive viewer for data exploration (Figure 2). Given a burning threshold and a time-step of the simulation, it displays in real time the 3D *burning cells* by exploiting the merge trees and the meta-segmentations. It also shows the user a time-tracking graph modeling the behavior (split or merge) of the burning cells through time. Finally, a linked view allows the user to interact with the burning cells either from the 3D visualization or from the time tracking graph.

3 Topology pre-processing

A key advantage of our system is the ability to present more than a single static segmentation. Instead, we allow the user to interactively browse the one-parameter family of all possible burning thresholds and their resulting segmentations into burning cells. Furthermore, we provide various conditional attributes along side the primary segmentation. This flexibility is based on computing augmented merge trees representing all possible segmentations. The resulting data structure forms a highly flexible *meta-segmentation* that we use for visualization as well as data analysis. This
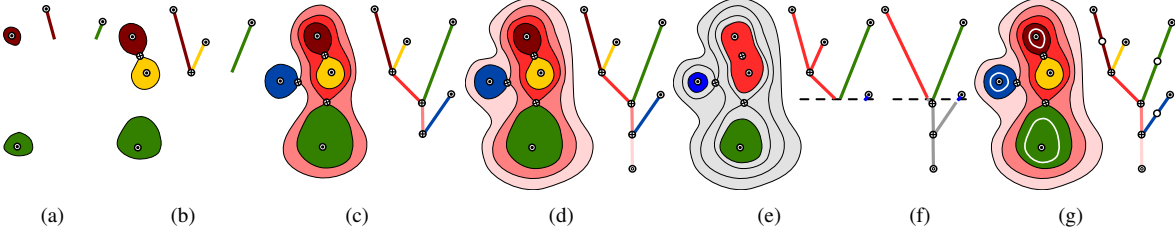
Figure 3. (a)-(d) Constructing a merge tree and corresponding segmentation by recording the merging of contours as the function value is swept top-to-bottom through the function range. (e) The segmentation for a particular threshold can be constructed by cutting the merge tree at the threshold, ignoring all pieces below the threshold and treating each remaining (sub-)tree as a cell. (f) The segmentation of (e) constructed by simplifying all saddles above the threshold. (g) The merge tree of (d) augmented by splitting all arcs spanning more than a given range.

section discusses the different algorithms and data structures used for processing individual time steps as well as constructing the global tracking graphs.

3.1 Augmented merge trees and burning cells

First, we define the augmented merge trees and show how they can store an abstract representation of the burning cell segmentation for any possible burning threshold.

Given a smooth simply connected manifold $\mathbb{M}$ and a function $f : \mathbb{M} \to \mathbb{R}$, the *level set* $L(s)$ of f at isovalue s is defined as the collection of all points on $\mathbb{M}$ with function value equal to s: $L(s) = \{p \in \mathbb{M} | f(p) = s\}$. A connected component of level set is called a *contour*.

The *merge tree* of f represents the merging of contours as the isovalue s is swept top-to-bottom through the range of f (Figure 3). Each time the isovalue passes a maximum a new contour is created and a new leaf appears in the tree. As the function value is lowered, contour merging is represented in the tree as the joining of two branches. Each branch in the merge tree corresponds to a neighboring set of contours and thus branches represent subsets of $\mathbb{M}$. Moreover, we augment the merge trees by with additional valence two nodes by splitting branches longer than some desired threshold interval (Figure 3(g)). We call the *meta-segmentation* the segmentation of $\mathbb{M}$ defined by the augmented merge tree of f.

Traditionally, merge trees are used as an intermediary data-structure for the computation of *contour trees* and can be computed with the algorithm presented by Carr et al. [4]. Here, we are interested in regions of high fuel consumption rate. In particular, the scientists define the *burning cells* as connected regions with a fuel consumption rate higher than a given *threshold*. Subsequently, we use the sub-trees of the merge tree above the given burning threshold to extract burning cells.

Given a threshold t on the fuel consumption rate f, we determine the corresponding burning cells by (conceptually) cutting the merge tree of f at t, creating a forest of sub-trees. Each individual sub-tree represents one connected

burning cell (Figure 3). Extracting a burning cell from the volumetric raw data amounts to collecting the segments of the *meta-segmentation* corresponding to the branches of the connected sub-tree (Figure 3(e)).

The augmented merge trees form the the fundamental data structure in our framework, storing in an abstract manner all possible burning cell segmentations, for any burning threshold. Even for the largest data sets, the resulting merge trees consists of only around 6Mb ASCII information per time step, compared to several Gigabytes of raw data, and are thus small enough to be loaded interactively from disk.

3.2 Burning cell tracking

Tracking graph computation. Given the augmented merge trees and corresponding segmentations for all time steps, we track all burning cells defined by a given static burning threshold.

To this end we load two consecutive time-steps and specialize the corresponding segmentations to the given threshold. Subsequently, we traverse the vertices of both segmentations in parallel determining the overlapping of burning cells through time. For each cell of the first time step that overlaps a cell of the second time step, a unique edge linking the two corresponding burning cells is added to the tracking graph. For each pair of time steps we dump the partial tracking graph to disk to be assembled at the end.

Tracking graph simplification. The tracking graphs can become highly complex and difficult to understand (Figure 2). Furthermore, they contain artifacts of the thresholding such as tiny cells existing for only one or very few time steps. Therefore, to reduce the graph complexity and eliminate some of the artifacts we simplify the tracking graphs by removing all valence zero cells as well as cells with a volume smaller than a given threshold (typically cells having less than 100 vertices). Removing small cells substantially unclutters the segmentation leaving only the larger cells of interest. Similarly, such simplification significantly streamlines the tracking graph by suppressing unnecessary details.

4 Interactive exploration

The primary focus of our work is to provide the application scientists with the ability to comfortably explore their data in a manner meaningful in their particular problem space. Visualization is an important tool to validate simulations as well as to deliver a high level understanding of the underlying phenomena. Furthermore, it is a powerful method to investigate and set up further in-depth data analysis. For example, allowing the user to easily explore conditional variables provides a simple way to understand whether various conditional statistics may provide new insights into the data.

To understand turbulent combustion, traditional visualization techniques are only of limited help. For example, while iso-surfaces at various thresholds would deliver geometry similar to our segmentations they contain none of the key information about whether two burning cells are connected, how many individual components exists, and/or their sizes, average values etc.. Clearly all this additional information could be computed but would require accessing the original data at 16 Giga bytes per time step which would make any interactivity infeasible. The augmented merge trees instead pre-process the data with regard to one of the most important aspects of the data (the burning cells) and store all additional information in accordance with this segmentation. Even for the more complicated SwirlH2Fast data set, the resulting information consists of a roughly 6Mb ASCII file describing the augmented merge trees including attributes and a 144Mb segmentation per time step. Using standard gzip compression, this reduce to roughly 70Mb, which corresponds to a data reduction of more than two orders of magnitude while still providing greater flexibility in the segmentation and selection than possible using standard techniques. In this section, we will describe the different aspects of our interface as well as the algorithms used to implement the various features. When discussing the interface we will use Roman numerals to refer to Figure 2 which illustrates the different components.

4.1 Graph display

One of our two main windows (II) is dedicated to the display of the tracking graph. We use *dot* [10] to layout the tracking graphs and display the resulting SVG file. To reduce the visual clutter, only the non-valence two nodes of the tracking graph are shown while sequences of valence two nodes are indicated by unbroken arcs. Furthermore, we can apply various color maps to the nodes of the graph presenting one additional attribute of the nodes. For exploration we typically use the cell volume (represented by the number of vertices within the cell) to highlight larger cells. To display the graph, we load its geometry into OpenGL, which allows us to draw even the largest graphs fully interactively.

The graph display not only provides a visualization of the graph but also allows the user to select nodes or arcs. When selecting an arc the system automatically selects the closest valence two node along this arc. The selection triggers two actions. First, the system loads the merge tree of the corresponding time step and, if desired, a number of neighboring time steps (see below). Since the merge trees are comparatively small, they are loaded interactively from disk without any caching or other acceleration mechanism. Second, the segment id and all its corresponding attribute information are extracted from the merge tree and displayed in the info window (IV). Finally, if the node the user has selected corresponds to any segment currently shown, this segment will be highlighted (V).

4.2 Segmentation display

The second main window (I) displays the segmentation and allows the user to vary the threshold (VI) and pick the number of in-memory time steps (VIII). Even though the segmentation is tiny when compared to the original data, parsing roughly 150Mb of binary data into the display data structures currently cannot be performed interactively. Thus we always load the merge tree information first providing the attribute and hierarchy information. If the user wants to explore a segmentation in more detail the necessary data is loaded via a button within about a second depending on the data size and available hardware. Note, that this performance could likely be improved significantly using compression to reduce I/O time or by preprocessing the segmentations further to become more closely aligned with the display data structures. The data is cell centered and piecewise constant. Therefore, we display each vertex of the segmentation as a box to preserve as much of the characteristics of the original data as possible. Individual cells are displayed using one of eleven colors at random preserving bright-red for highlighted cells. Looking at the segmentation it is important to remember that we use the full 26 neighborhood when computing the merge trees. Thus even cells touching only at their corners are considered connected. Similar to the graph display we allow the user to select individual segments displaying their information on the side (IV). Finally, we provide an additional window (III) which allows the user to sub-select segments based on the various attributes. Overall, the system allows the users explore the entire time series of a combustion simulation at arbitrary thresholds and using conditional selection criteria.

5 Results

This paper presents a new visualization and analysis framework to explore the dynamics of turbulent cellular flames. For the first time it is possible to study, in a qualitative and quantitative manner, the complete temporal evo-

	SwirlH2		SwirlH2Fast	
	Center	Complete	Center	Complete
Parallel File I/O	193sec	355sec	210sec	569sec
Serial Processing	56sec	1067sec	70sec	2684sec

Table 1. Run-times for the processing of a single representative time step for the SwirlH2 and SwirlH2Fast data set.

lution of burning cells under varying thresholds and using conditional subselection. This represents fundamentally new capabilities in studying turbulent flames and in collaboration with the scientists we are actively using the system to formulate new hypotheses on the combustion process.

Topology pre-processing All data processing was performed in parallel on an SGI Altix 350, with 32 Itanium-2, 1.4 GHz processors. The run times for a single representative time step for computing the merge trees and corresponding segmentation are given in Table 1. To better illustrate the cost split between file I/O and computation we report two numbers for each case: First, the time for extracting the raw data and dumping the resulting stream to a file. Second, the time to process this file.

Qualitative analysis. The first significant observation is that the flames in the low-swirl configuration seem to burn in two different modes, see Figure 1(e). Overall, the burning cells create a bowl shaped structure centered above the burner. To better highlight the center of the flame, the segmentations presented in the paper show the flame up-side-down looking towards the bottom of the bowl in the direction of the fuel stream. Around the middle of this bowl, cells appear to behave much like the idealized flames studied in [6]. On the outside, however, the flames burn more chaotically in smaller, irregularly shape regions. The behavior of these *fringe* cells is very unlike that of the idealized flames and it is not yet clear how to model them. Therefore, the scientists initially focused their analysis on the bowl center.

Based on observing the cell structures a radius cut-off is selected and we extract only the data on the interior of a cylinder with radius 2.5 cm centered on the burner, see Figure 4. We then compute the augmented merge trees and the corresponding segmentations necessary to explore the data in more detail. Studying the segmentations at different thresholds reveals significant differences to the idealized flames of [6, 3]. As can be seen in Figure 4, the previously used threshold of 2.6 no longer represents a viable choice. Rather than separating the volume into the cellular burning regions, it defines few very large cells inconsistent with the initial expectations. Instead, for the SwirlH2 data set the segmentations suggest a threshold of around 5, see Figure 5, and for the SwirlH2Fast an even higher threshold at around

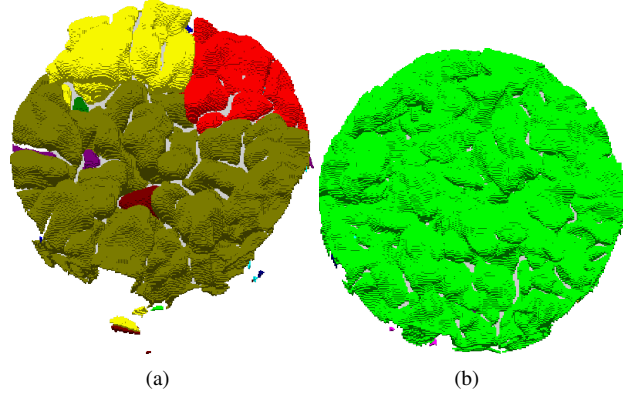(a)   (b)

Figure 4. Burning cells within a cylinder of radius 2.5 cm centered in the middle of the data (fuel consumption cut-off: 2.6). (a) SwirlH2 data (t=1500); (b) SwirlH2Fast (t=3000).
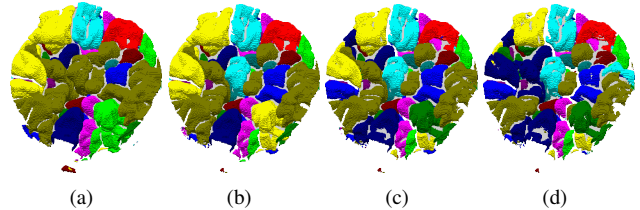

(a)   (b)   (c)   (d)

Figure 5. Burning cells in the SwirlH2 data (t=1500) using 4.0 (a), 5.0 (b), 6.0 (c), and 7.0 (d) as burning threshold.

8, see Figure 6.

Quantitative analysis. To validate these empirical observations we compute the weighted cumulative density functions (WCDFs) of the distribution of cell size and compared them to the idealized flames. We also repeated the surface based analysis introduced in [3] to arrive at two sets of distribution functions for each low-swirl experiment, see Figure 7. As suggested by the visualization, the distributions show a markedly different behavior for lower fuel consumption thresholds. For small thresholds, the distributions become exponential indicating a small number of larger cells rather than the logarithmic behavior seen in previous studies. However, as the threshold increases the distributions continuously change to a logarithmic shape. Furthermore, the area distributions of the 2D analysis switch character-
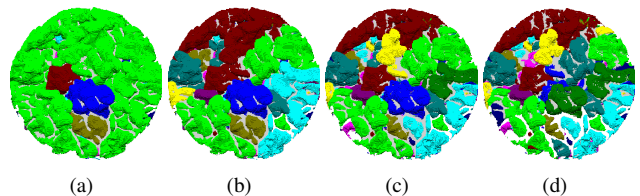

(a)   (b)   (c)   (d)

Figure 6. Burning cells in the center of the SwirlH2Fast data (t=3000) using 6.0 (a), 7.0 (b), 8.0 (c), and 9.0 (d) as burning threshold.
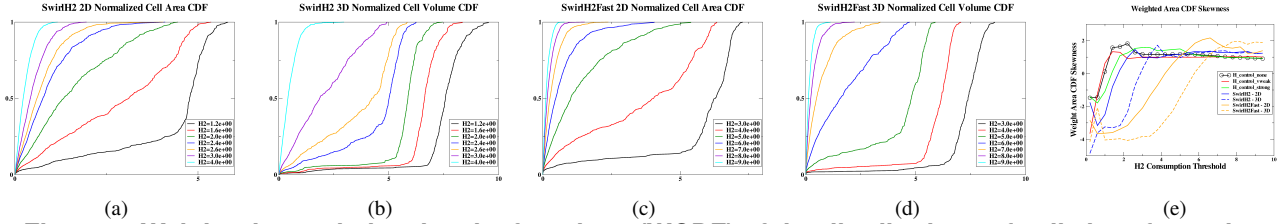
Figure 7. Weighted cumulative density functions (WCDF) of the distributions of cell sizes for various thresholds for the surface based analysis of [3] (a),(c) and the volumetric analysis presented here (b),(d) for the SwirlH2 and SwirlH2Fast data set respectively. Exponential distributions indicate few large cells. (e) Skewness plots of the WCDF of the idealized flames analyzed in [6] and of the SwirlH2 and SwirlH2Fast data.

istic at a significantly lower threshold. To further quantify this shift we compute the skewness [7] of the distributions of both the idealized flames at different turbulence levels as well as the low-swirl flames presented here. The skewness of an exponential type cumulative density function would be smaller than 0, that of a linear CDF 0, and that of a logarithmic CDF larger than 0. The resulting graphs show several interesting results, see Figure 7(e). Clearly the surface based analysis skews the distributions to become logarithmic for smaller thresholds. Furthermore, the graphs for the three-dimensional segmentations validate the visual impression of thresholds around 5.0 and 8.0 for the SwirlH2 and SwirlH2Fast case respectively. Finally, even the idealized flames show exponential WCDFs at very low thresholds something not seen in previous studies.

It is important to note that the entire statistical analysis presented here requires only the augmented merge trees and not the original data. This fact is crucial to allow such extensive studies since accessing the original data would be prohibitively expensive.

Using the skewness plots of Figure 7(e) as a guide, we choose 5 (SwirlH2) and 8 (SwirlH2Fast) as thresholds for the tracking and create the corresponding graphs. As discussed in Section 4 we can then use the tracking graphs along side the segmentations to explore an entire time sequence on commodity hardware.
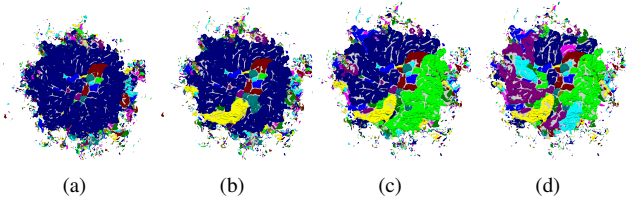

Figure 8. Burning cells in the SwirlH2 data (t=1500) using 4.0 (a), 5.0 (b), 6.0 (c), and 7.0 (d) as burning threshold.

Insights of the exploration. Combining the visual observations of Figure 5 and 6 with the statistical results shown in Figure 7 might suggest that the swirling flames behave similar to the idealized flames but at overall higher fuel con-

sumption rates. However, further exploration shows that the results of the spatially restricted data are misleading. Clearly some cells cross the cylindrical center region and thus get cut. One would therefore, expect some cells to appear slightly smaller, which could be taken into account during the analysis. However, a more significant problem is that cells can also become disconnected. In fact, segmentations of the entire data show that most cells that appear isolated in the center of the data are connected on the outside of the cylinder, see Figure 8 and 9. This creates small sets of large cells for even higher thresholds than the plots of Figure 7(e) suggest. Only for unreasonably high thresholds of 8.0 and 12.0 do the complete flames break apart into smaller components, see Figure 10. Overall, it appears that the low-swirling flames are in a substantially different regime than the idealized flames. Our interactive framework, coupled with the data analysis made possible by using augmented merge trees, has been instrumental in trying to better understand the underlying dynamics controlling the low-swirling flames and has open several new research directions.

5.1 Discussion and Future Work

While we can compute the tracking graphs for the full AMR based data including the cells on the fringes, the resulting graphs are difficult to handle. *Dot* currently does not scale gracefully to these large graphs and creating a layout can take hours or fail all together. Furthermore, assuming a layout is created, the resulting graphs are difficult to interpret even after heavy simplification. Moreover, currently the graphs, unlike the segmentations, are computed for a static threshold. The data structures contain enough information to efficiently create graphs for variable thresholds.
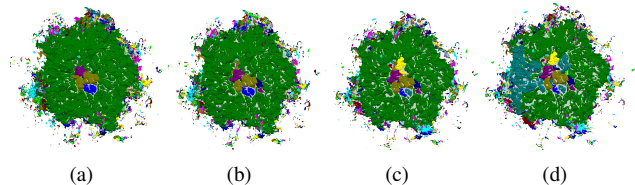

Figure 9. Burning cells in the SwirlH2Fast data (t=3000) using 6.0 (a), 7.0 (b), 8.0 (c), and 9.0 (d) as burning threshold.
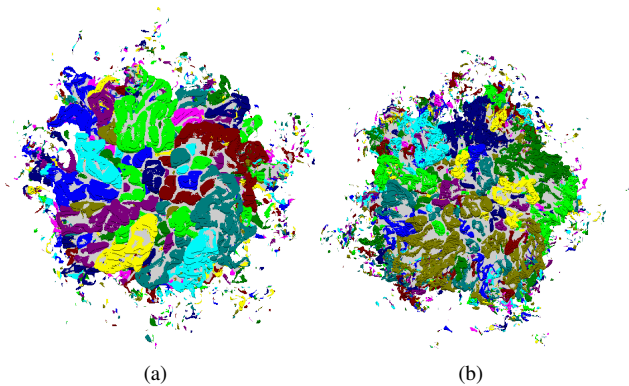
Figure 10. Burning cells in (a) the SwirlH2 data (t=1500, burning threshold: 8), (b) the Swirl-Fast data (t=1500, burning threshold: 12).

However, viewing these graphs requires an interactive layout, which is beyond the current state of the art. Thus new paradigms are needed to handle such graphs potentially involving more sophisticated simplification and hierarchical representations.

Finally, loading and drawing the geometry becomes noticeably slower for larger data and increasing the resolution by a factor of eight, as planned for the future will push the system beyond its current capabilities. However, the rendering code as well as the file I/O is currently entirely unoptimized and contains many opportunities for future improvements. One example would be a compressed file format to reduce file IO.

6 Conclusions

We have presented an interactive topological framework for exploring and analyzing large scale turbulent combustion. Using augmented merge trees and their corresponding segmentation, we allow to both visualize and post-process entire simulations using pre-processed data orders of magnitude smaller than the original data sets. Providing easy access to correct and comprehensive segmentations including derived attributes has proven to be a powerful tool to better understand turbulent combustion and to form new hypotheses on the underlying physical processes.

References

[1] C. Bajaj, V. Pascucci, and D. R. Schikore. The contour spectrum. In *IEEE Visualization*, pages 167–175, 1997.

[2] B. Bedat and R. K. Cheng. Experimental study of premixed flames in intense isotropic turbulence. *Combust. Flame*, 100:485–494, 2005.

[3] P.-T. Bremer, G. Weber, V. Pascucci, M. Day, and J. Bell. Analyzing and tracking burning structures in lean premixed hydrogen flames. *IEEE Transactions on Visualization and Computer Graphics*, 2009, to appear.

[4] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. In *Proc. of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 918–926, New York, NY, USA, Jan. 2000. ACM, ACM Press.

[5] H. Carr, J. Snoeyink, and M. van de Panne. Simplifying flexible isosurfaces using local geometric measures. In *Proc. IEEE Visualization 2004*, pages 497–504, 2004.

[6] M. Day, J. Bell, P.-T. Bremer, V. Pascucci, V. Beckner, and M. Lijewski. Turbulence effects on cellular burning structures in lean premixed hydrogen flames. *Combustion and Flame*, 156:1035–1045, 2009.

[7] J. L. Devore. *Probability and Statistics for Engineering and the Sciences*. Brooks/Cole - Thomson Learning, Belmont, CA, 2004.

[8] I. Fujishiro, Y. Maeda, and H. Sato. Interval volume: a solid fitting technique for volumetric data display and analysis. In *Proc. IEEE Visualization 1995*, pages 151–158, 1995.

[9] A. Gyulassy, V. Natarajan, V. Pascucci, P.-T. Bremer, and B. Hamann. Topology-based simplification for feature extraction from 3D scalar fields. In *Proc. IEEE Visualization 2005*, pages 535–542, 2005.

[10] E. Koutsofios and S. North. Drawing graphs with dot. Technical Report 910904-59113-08TM, AT&T Bell Laboratories, Murray Hill, NJ, 1991.

[11] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Trans. Vis. Comp. Graph.*, 12(5):1052–1060, 2006.

[12] W. Lorensen and H. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comp. Graph.*, 21(4):163–169.

[13] A. Mascarenhas, R. W. Grout, P.-T. Bremer, E. R. Hawkes, V. Pascucci, and J. Chen. *Topological feature extraction for comparison of terascale combustion simulation data*. Mathematics and Visualization. Springer, 2010. to appear.

[14] J. Milnor. *Morse Theory*. Princeton University Press, New Jersey, 1963.

[15] K. Stockinger, J. Shalf, K. Wu, and E. W. Bethel. Query-driven visualization of large data sets. In *Proc. IEEE Visualization 2005*, pages 167–174, 2005.

[16] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *AVI '00: Proc. of the working conference on Advanced visual interfaces*, pages 110–119, 2000.

[17] G. Weber, P.-T. Bremer, J. Bell, M. Day, and V. Pascucci. Feature tracking using Reeb graphs. In *Proceedings TopoInVis Workshop*, 2009. to appear.