Analyzing and Tracking Burning Structures in Lean Premixed Hydrogen Flames

P.-T. Bremer[1], G. Weber[2], V. Pascucci[3], M. Day[2], and J. Bell[2]

Abstract— This paper presents topology-based methods to robustly extract, analyze, and track features defined as subsets of isosurfaces. First, we demonstrate how features identified by thresholding isosurfaces can be defined in terms of the Morse complex. Second, we present a specialized hierarchy that encodes the feature segmentation independent of the threshold while still providing a flexible multi-resolution representation. Third, for a given parameter selection we create detailed tracking graphs representing the complete evolution of all features in a combustion simulation over several hundred time steps. Finally, we discuss a user interface that correlates the tracking information with interactive rendering of the segmented isosurfaces enabling an in-depth analysis of the temporal behavior.

We demonstrate our approach by analyzing three numerical simulations of lean hydrogen flames subject to different levels of turbulence. Due to their unstable nature, lean flames burn in cells separated by locally extinguished regions. The number, area, and evolution over time of these cells provide important insights into the impact of turbulence on the combustion process. Utilizing the hierarchy we can perform an extensive parameter study without re-processing the data for each set of parameters. The resulting statistics enable scientist to select appropriate parameters and provide insight into the sensitivity of the results wrt. to the choice of parameters. Our method allows for the first time to quantitatively correlate the turbulence of the burning process with the distribution of burning regions, properly segmented and selected. In particular, our analysis shows that counter-intuitively stronger turbulence leads to larger cell structures, which burn more intensely than expected. This behavior suggests that flames could be stabilized under much leaner conditions than previously anticipated.

1 Introduction

When analyzing and visualizing three-dimensional scalar fields isosurfaces play an important role as they are often easier to comprehend than volumetric representations. Furthermore, in many scientific applications isosurfaces have well established physical interpretations. In order to analyze multiple scalar fields concurrently it is, therefore, natural to project one field on the isosurface of second field analyzing the resulting two-dimensional function. For example, in combustion simulations flames are typically constructed as temperature isosurfaces, called isotherms. However, only areas of this "flame" where the fuel consumption rate is above a certain threshold are actually considered burning. This paper presents novel topological techniques that enable comprehensive statistical analysis of such thresholded features through a parameter independent hierarchy. Furthermore, the approach provides in-depth studies of the temporal evolution of features using both comprehensive tracking graphs as well as interactive visualization of time-correlated isosurfaces.

Understanding combustion processes over a broad range of operational regimes is of great interest for a variety of applications such as engine or power plant design. To this end, there has been considerable recent interest in the development of premixed burners capable of stably burning ultra-lean hydrogen-air fuel mixtures. Such burners could, for example, be used as one component of a clean-coal power plant utilizing hydrogen extracted from coal gasification. Lean premixed systems are subject to a variety of hydrodynamic and combustion instabilities that render practical flame stabilization, and traditional approaches to flame analysis, extremely difficult. The flames burn in a cellular mode that is highly nonuniform, time-dependent, and difficult to characterize [7].

The analysis presented here is performed using a representative set of numerically evolved flames. Each flame burns in a volume periodic in the x- and y-coordinate with pre-mixed fuel being injected through the plane z=0. The flame simulations were carried out with a low Mach number model that incorporates a detailed description of the combustion kinetics and differential species transport. For more details of the model and its implementation, see [8]; for example analysis of lean hydrogen flames in this regime, see [2]. Each solution analyzed here is represented as a sequence of three-dimensional snapshots, taken at uniform intervals in time, where each snapshot consists of cell-centered data on a uniform cartesian grid describing the temperature, chemical composition, and effective fuel consumption rate. The "flame" is represented as an isotherm extracted from the datasets using standard techniques. The local fuel consumption rate, interpolated to these isosurfaces, is used to divide the surface into burning "cells" separated by non-burning regions, as defined by a threshold in the consumption rate.

To gain new insights into the combustion process scientists are interested in two types of analysis: time-aggregate statistics and detailed analysis of cell evolution. Comprehensive statistics on the number and area of burning cells aggregated over time provide important information about quantitative and qualitative differences between flames of varying turbulence levels. A tracking graph representing the evolution of cells over time describes the local cell dynamics in detail and allows an in depth study of all temporal events such as cell births, deaths, merges, and splits. Furthermore, coupling the tracking graph to an interactive visualization of the segmented flame surfaces illustrates each event and is an important tool to verify and validate the parameter selection.

Given a fuel consumption threshold, we use topology based methods to segment the flames into burning cells from which we can extract the necessary information about their numbers, areas, and evolution over time. Furthermore, we present a hierarchical representation capable of storing all possible segmentations independent of a particular threshold. As a separate step in our analysis, this hierarchy can be quickly adapted to any specific threshold. This capability allows us to perform extensive parameter studies to determine and verify the choice of parameters as well as to study the sensitivity of the results to changes in these parameters. Once an appropriate choice of parameters has been determined we track the corresponding cells over time using the Reeb graph based tracking algorithm developed in [36]. The resulting graph not only encodes the evolution of cells over time but correlates each burning cell with a particular arc in the graph using a unique identifier. All flame surfaces can then be interactively rendered color-coded based on these identifiers which allows the user to easily study any graph event in detail.

[1] *Center of Applied Scientific Computing (CASC),*
Lawrence Livermore National Laboratory, ptbremer@acm.org
[2] *Lawrence Berkeley National Laboratory*
[3] *Scientific Computing and Imaging Institute,*
University of Utah

Manuscript received 31 March 2007; accepted 1 August 2007; posted online 27 October 2007.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

The methods presented here allow for the first time a quantitative analysis of the cellular burning structures and provide important scientific insights: Contrary to common intuition, higher turbulence levels, which generally lead to increased energy in the finer length scales of the approach flow to the flame, apparently lead to *larger* cellular burning structures. Moreover, these larger cells tend to burn more intensely than would be expected from simple theories of flame propagation. The combination of these two effects dramatically increases the global propagation speed of the turbulent flames over the steady flat flame case. Our contributions in detail are:

1. A topology-based segmentation of the burning cells of the flame front;
2. A novel topological hierarchy storing all possible segmentations for all possible fuel consumption thresholds. In a single pass we collect a parameter independent hierarchy for each time step. The hierarchies can be post-processed on any off-the-shelf PC to produce a wide range of statistical data;
3. An extensive parameter study analyzing the sensitivity of the results to changes in various parameters and providing methods for optimal parameter selection;
4. A quantitative comparison between simulations at different resolutions; and
5. An interactive environment to study the temporal evolution of burning cells using flame surfaces color-coded according to a global tracking graph.

1.1 Related Work

Topological techniques for scientific data analysis have become increasingly popular. Much of the recent work is based on Morse theory [24, 23] and in particular the Morse-Smale complex or substructures thereof. Related methods have been discussed as early as the 19th century [6, 22] and the Morse-Smale complex has been described using various names [25, 27]. In its modern form on which this paper is based the Morse-Smale complex was introduced for surfaces by Edelsbrunner et al. [13] and some extensions by Bremer et al. are discussed in [4]. Extensions to the three-dimensional case can be found in [12, 16, 17]. Notice, how this body of work allows to make major progress compared to early attempts using topology [31] since the simple qualitative feedback is expanded to a complete quantitative analysis.

The segmentation techniques presented here are similar to the one used by Laney et al. [21] to analyze the mixing layer in a Rayleigh-Taylor instability in that we also use stable manifolds to define features. However, unlike [21] we intersect the stable manifolds with level sets of varying iso-values to encode a one parameter family of threshold-based segmentations. Furthermore, we are not only interested in counting features but also in computing their size which makes the initial segmentation more challenging. Finally, as will be discussed in Section 3, our feature definition requires a topological hierarchy different from the one presented in [21]. In another application of Morse theory to data analysis, Gyulassy et al. [15] use the ascending one-manifolds of the three-dimensional complex to define and analyze core lines in atomistic simulations of porous media. In the context of analyzing protein structures surface features have also been analyzed using pockets [9, 37]. However, these structures are defined by the geometry of the manifold (the protein surface) rather than a function on the manifold and thus these techniques cannot be applied to analyze burning cells.

Defining and tracking features of interest has long been of interest to the visualization community. Samtaney et al. [30] apply methods related to object tracking in image processing to scientific data sets. Their method tracks connected, thresholded regions by correlating attributes such as centroid, volume, and moment between time steps. A first pass eliminates all objects that are close to identical in both time steps. Subsequent processing then checks whether objects bifurcate (split) or amalgamate (merge). A graph encoding the birth, death, merging, and splitting of features presents the results of this matching.

Another popular approach [32, 33, 21] is to use the volume overlap of features to identify correspondences. In this case the matching can be improved by incorporating motion prediction [29]. Finally, Ji et al. [19, 18] track the evolution of isosurfaces in a time-dependent volume by extracting the four dimensional space-time isosurface and defining features in successive time steps as linked if they belong to the same space-time surface.

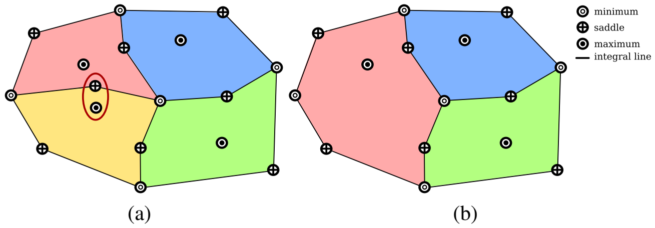
In another line of research Edelsbrunner et al. [11] describe the theory and algorithms to compute time-varying Reeb graphs by connecting them using Jacobi sets [10], i.e., paths traced by critical points. Szymczak [35] presents related techniques for contour trees. Sohn and Bajaj [34] use a hybrid approach also defining correspondences between contour trees but using volume matching similar to Silver and Wang [32, 33] rather than topological information as in [10] or [35]. Finally, to avoid displaying the tracking graphs directly Fujishiro et al. [14] use T-Maps as for visually exploring the structure of large time-dependent volumes.

2 THEORY

The algorithms for segmenting and tracking burning cells are based on Morse theory [24, 23]. Morse theory analyzes smooth functions on manifolds based on their gradient behavior. In this paper we use two related structures, the Morse complex [13] and the Reeb graph [28]. The Morse complex is a sub-structure of the Morse-Smale complex used in previous work [21, 16, 17].

2.1 Morse Complex

The Morse complex segments a smooth manifold $\mathbb{M}$ into regions of uniform gradient behavior. Let $f : \mathbb{M} \to \mathbb{R}$ be a smooth function with gradient ∇f. All points p on $\mathbb{M}$ are classified as either *critical* if $\nabla f(p) = 0$ or *regular* otherwise. If all critical points have pairwise distinct function value f is said to be *Morse*. Here we are only concerned with two-dimensional manifolds where a critical point can either be a minimum, a saddle, or a maximum. Furthermore, an *integral line* $L(t) : \mathbb{R} \to \mathbb{M}$ of f is defined as a line whose tangent is aligned with the gradient of f: $\delta L/\delta t = \nabla f(L(t))$. For $t \to \infty$ all integral lines converge towards a maximum where they are said to *end*. For each maximum p the union of all integral lines ending at p is called the *stable manifold* of p. Note that the stable manifolds defined in this manner are open disks that do not include their boundary. The collection of stable manifolds segments $\mathbb{M}$ forming a complex called the *Morse complex* of f. The *nodes* of this complex are the saddles and minima of f; its *arcs* the integral lines connecting saddles to minima; and its regions the stable manifolds of the maxima of f, see Fig. 1(a). For a more detailed description of Morse theory especially its application to piece-wise linear (pl-) functions we refer the reader to [13] and [4]. Note, that the Morse complex is a sub-structure of the Morse-Smale complex used in previous work [21, 16, 17]. In particular, the Morse complex does not contain information about the *unstable manifolds* of f (the stable manifolds of $-f$). However, the unstable manifolds are not of interest for our analysis and thus, we restrict ourself to the Morse complex which can be computed easier and more efficiently than the Morse-Smale complex. We use a subset of the algorithms discussed in [5] to compute the Morse complex.



(a) (b)

Fig. 1. (a) Example Morse complex with the stable manifolds highlighted; (b) Morse complex of (a) after the cancellation of the marked saddle maximum pair. The two critical point plus their incident integral lines are deleted and two stable manifolds merge.

In many applications it is useful to simplify an initial Morse complex. Simplification can remove noise as well as provide a way to analyze the complex at multiple scales. A Morse complex is simplified by *canceling* saddles with maxima of adjacent stable manifolds. Among the neighboring maxima one usually chooses the one with lowest function value and this is the strategy we employ here. Each cancellation removes one saddle along with its incident arcs as well as one maximum. As a result two regions merge simplifying the complex, see Fig. 1(b). Traditionally, *persistence*, i.e., the difference in function value between the two canceled critical points, has been used to rank cancellation. While persistence has been proven be an excellent choice for removing noise [21] this might not be the only purpose of simplification. In particular, using the function value of the saddle to rank cancellations thus removing the highest saddle first mimics a watershed like segmentation, see Section 3.1. In this case the "water level" is represented by the fuel consumption threshold and each island represents a burning cell. As the threshold decreases islands/cells merge which corresponds to cancellations in the hierarchy.

2.2 Reeb Graph

While the Morse complex is defined using the gradient of f, the *Reeb graph* stores information about the *level-sets* of f. The level-set of f at value s is defined as all points on $\mathbb{M}$ with function value s. The connected components of the level-sets are called *contours*. The Reeb graph of f is constructed by contracting the contours of f into points, see Fig. 2. The *nodes* of the Reeb graph are formed by the contours passing through critical points of f and its *arcs* by the remaining contours. Note, that contours change topology only at critical points. Therefore, an arc represents a family of contours that does not change topology.

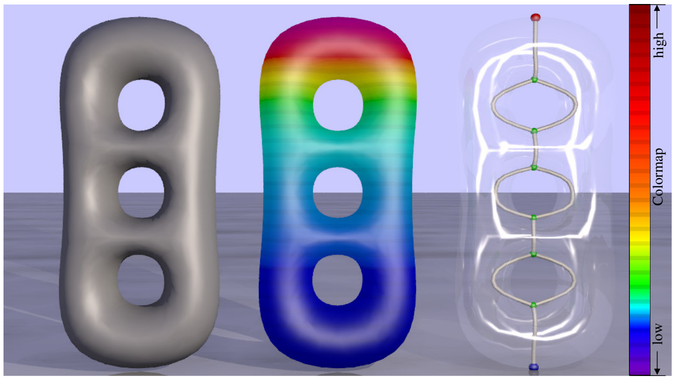


Fig. 2. Reeb graph of the height function on the triple torus. The three loops in the graph correspond to the three tunnels of the model.

Similar to the Morse complex the Reeb graph is simplified by canceling connected pairs of critical points. Following the algorithm in [1] one can either cancel an extremum-saddle or a saddle-saddle pair that forms a loop. The former case removes a leaf from the graph while the later simplifies a loop to a line. In this paper we use the Reeb graph for tracking cells over time (see Section 3.2), and leafs represent the birth or death of a cell. Therefore, we are mainly interested in simplifying loops. Using time as function to build the Reeb graph, small loops represent, for example, neighboring regions that split and merge in quick succession before finally separating entirely. These cases are due to instabilities in the parameter selection and simplifying the corresponding loops removes these artifacts from the Reeb graph. We use the algorithms presented in [26] to compute and simplify the graph.

3 Analysis

This section describes how to efficiently segment the flame surfaces into burning and non-burning regions and how to encode the results in a threshold independent hierarchy. Furthermore, the algorithm extends the hierarchy by computing the area of all burning cells for all possible thresholds. The resulting data structure can be quickly specialized to any given threshold and persistence parameter. Finally, the section briefly discusses the tracking algorithm and we refer the reader to [36] for a more detailed description.

3.1 Aggregate Analysis

First, we focus on the analysis performed on each time step independently and in particular on how this analysis can be accomplished simultaneously for all possible thresholds. For each time step we extract the flame front as an isosurface of a given temperature and interpolate the fuel consumption rate at each vertex of the (triangulated) surface. In this particular case the isosurface is a byproduct of the space-time isovolume we use to track burning cells and its construction will be described in more detail in Section 3.2. We then compute the Morse complex of the fuel consumption interpolated onto the isotherm, encode it hierarchically, and save it to disk. In a post-processing step the complex is adapted to a specific persistence and fuel consumption threshold from which one can extract all necessary information about number and areas of burning cells.

Given any single fuel consumption threshold, the number and areas of all burning cells could be computed by a flood-fill approach. Assuming a pre-determined threshold vertices are implicitly labeled as burning/non-burning and what remains to be done is to collect all connected components of burning regions, compute their area, and record their count. However, this strategy requires to re-process all time steps of all simulations for any change in threshold. This makes a thorough parameter study prohibitively expensive and likely flawed as only a small, finite number of thresholds could be tested. Instead, we use the segmentation given by the Morse complex to represent burning cells. Storing the areas of the stable manifolds along with the Morse complex allows one to extract information about any possible threshold by imposing an appropriate hierarchy on the Morse complex and aggregate the count and areas accordingly.

As discussed in Section 2, the Morse complex segments the surface into the stable manifolds of all local maxima. Since stable manifolds, by definition, are simply connected there can never be more burning cells than there are local maxima of fuel consumption (each cell has at least one maximum in its interior). As a first approximation one can view each maximum as representing a burning cell. One then uniformly subdivides the overall function range up to a sufficient resolution and computes the partial area of each stable manifold falling within each sub-range, see Fig. 4. Assuming the area varies linearly within each sub-range this information is sufficient to calculate the burning areas of all stable manifolds for any threshold by adding up all partial areas considered burning.

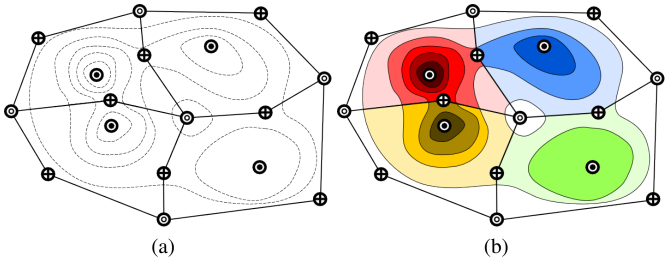


Fig. 4. (a) The Morse complex of Fig. 1(a) with several contours, drawn as dotted lines, indicating the subdivision of the stable manifolds into sub-regions based on function range; (b) The Morse complex of (a) with the partial areas color-coded wrt. their stable manifolds.

Nevertheless, treating all stable manifolds as single burning cells is clearly incorrect. First, if a maximum has a fuel consumption rate below the threshold its stable manifold does not define a cell since no part of the surface is actually considered burning. Second, whenever two neighboring stable manifolds are connected by burning vertices, the two manifolds are part of the same cell. For a given threshold the first case can be trivially handled. For the second case it is important to remember that, by definition, the vertex with the highest function value shared between two stable manifolds is always a saddle. Canceling all saddles with fuel consumption rate above the threshold therefore merges all stable manifolds that are part of the same cell, thus restoring the one-to-one correspondence between (burning) maxima and burning cells.

The advantage of counting burning cells in terms of (simplified)

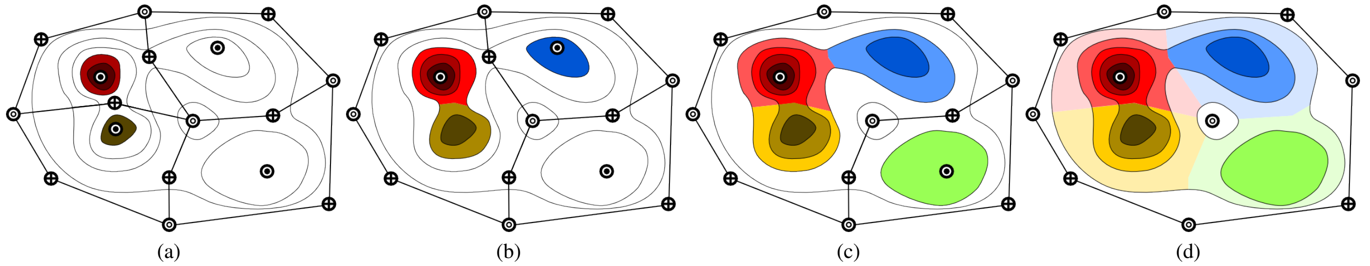


Fig. 3. Using a highest-saddle-first hierarchy to encode all possible cell segmentations: Consider the Morse complex of Fig. 1(a) together with the partial areas shown in Fig. 4(b). The sequence (a)-(d) shows how the burning cell segmentation can be constructed for all thresholds in a single top-to-bottom sweep. Colored portions of the complex are considered above the threshold, white regions below. (a) The segmentation at some initial threshold with two burning cells whose area is given by the sum of the pre-computed areas above the threshold; (b) Lowering the threshold it passes a maximum and a new cell appears (colored blue). Furthermore, the threshold has passed a saddle which is immediately canceled with its lower neighboring maxima (the yellow one). This causes the red and yellow cells to merge restoring the correct cell count for this threshold. The area of the merged cell is computed by simply summing the areas in the corresponding function ranges; (c) Further lowering the function value causes the green cell to appear and the blue one to merge with the red/yellow one. The combined area as before is given as the sum of the pre-computed areas; (d) Finally, lowering the threshold to the lowest contour shown merges the green cell with the rest.

stable manifolds is that it is easy to store the Morse complex in a hierarchical fashion based on whatever metric is appropriate. In this case the hierarchy is created by canceling saddles in order of decreasing fuel consumption. The resulting sequence of cancellations mimics a watershed segmentation with decreasing water level. Imagine a watershed segmentation as the fuel consumption threshold is swept top-to-bottom through its entire range. Each time the threshold passes a maximum, a new burning cell is born. Each time the threshold passes a saddle, two stable manifolds merge, see Fig. 3. The merging corresponds to the cancellation order of the hierarchy and the birth events are given by a sorted list of maxima. In practice, we store a list of maxima sorted by decreasing function value. Additionally, each maximum stores a *cancellation* value indicating at which threshold it will merge with a neighboring maximum and the corresponding neighbor index. As a result the maximum / burning cell count for each threshold can be determined by a single sweep through the list stopping at the given threshold. In this fashion all possible watersheds are stored in a single data structure. Note, that since the count can only change at discrete instances, namely at function values of maxima and saddles, the continuous function describing count vs. threshold can indeed be stored accurately. Finally, areas of the merged stable manifolds are collected by adding the appropriate pre-computed values at each cancellation.

In practice, we first perform a persistence-based simplification [13, 4] of the original Morse complex using a very small persistence threshold. While this simplification does not necessarily follow the watershed described above it has no effect on the results, see Section 4, but significantly reduces the number of maxima/stable manifolds present in the initial complex. For a detailed description of persistence-based simplification and its implementation for Morse complexes we refer the reader to [13, 4]. Since we pre-compute areas for all stable manifolds fewer maxima result in smaller file sizes necessary to store the hierarchy. Using a very conservative persistence of 0.001 and dividing the function range into 300 equally sized sub-ranges. File sizes range between 10-15MB for the coarse resolution and 20-30MB for the fine resolution per time step depending on the complexity of the isosurface.

3.2 Tracking

The algorithms discussed above support collection of cumulative statistical data for a complete time series, processing each time step separately. Another important aspect of combustion processes is the evolution of burning cells over time. In particular, scientists are interested in determining how and when burning cells are created/destroyed and merge/split.

To facilitate the tracking process, we first restrict our considerations to a particular isotherm, see Fig. 5(a), analogous to computing cumulative statistics via the Morse complex on an isotherm. Thresholding isotherm triangles by fuel consumption (Fig. 5(b)) yields a segmentation of the isotherm in burning regions. Our approach tracks these regions by considering their boundaries (Fig. 5(c)), tracing them over time (Fig. 5(d)) and determining when they split and merge. We can determine this split and merge events by computing the Reeb graph of the surface swept by the boundaries over time. (We note that the swept surface is actually embedded in 4D space using time as fourth dimension. Fig. 5(d) shows this swept surface projected in 3D space to illustrate the concept.)

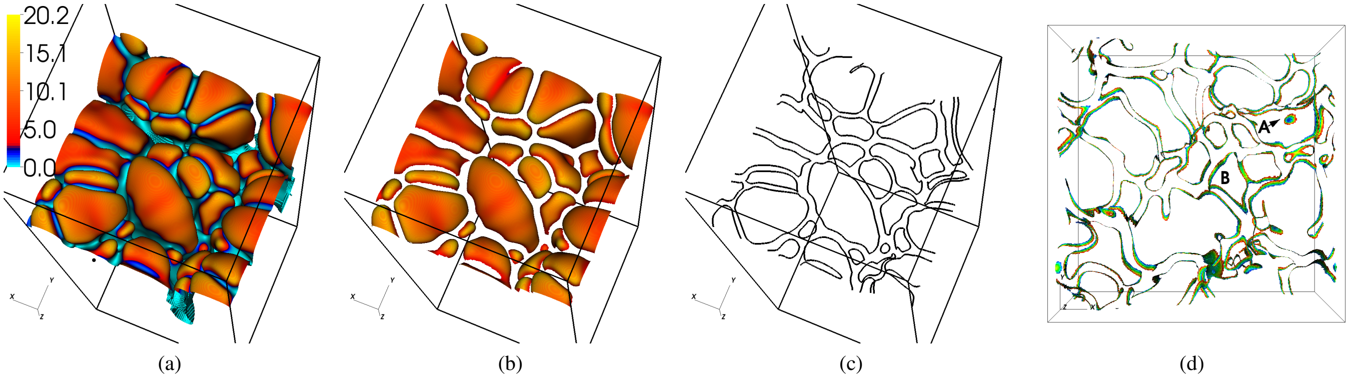
To obtain the Reeb graph of the swept surface, we first observe that burning region boundaries can be obtained by means of two subsequent contouring operations. First the isotherm is computed as isosurface of temperature and subsequently boundaries are obtained by contouring this isosurface by fuel consumption rate. This observation leads to the following construction scheme for the swept surface: (i) We concatenate all time steps to form a four dimensional rectilinear grid, where time is the fourth dimension and contour this grid using a 4D marching cubes algorithm [3]. In a second step we extract the boundaries of the burning regions in a second contouring step. The result is then used to calculate the Reeb graph, which is the desired tracking graph (after simplification).

Since we are only interested in creation, destruction, merging and splitting of burning regions, we do not need to extract exact boundaries of the burning region. Instead, it is sufficient to extract boundaries that preserve the connectivity of burning regions. While our pipeline performs all these steps on three dimensional data sets, we use the 2D case to illustrate the underlying concepts as shown in Fig. 6.

We extract the space-time surface swept by the boundaries of burning cells over time in a two-stage process. First, we connect all time steps of the simulations to a (virtual) 4D hyper-grid and extract a temperature isovolume using the algorithm presented in [3]. The isovolume encodes the evolution of the flame surfaces over time. Concurrently, we interpolate the fuel consumption scalar field at all vertices of the isovolume and label them as burning, non-burning based on a given threshold (we use 2.6 for all tracking related experiments). Finally, the space-time surface between the burning and non-burning regions of the isovolume is the surface swept by the boundaries of burning regions over time.

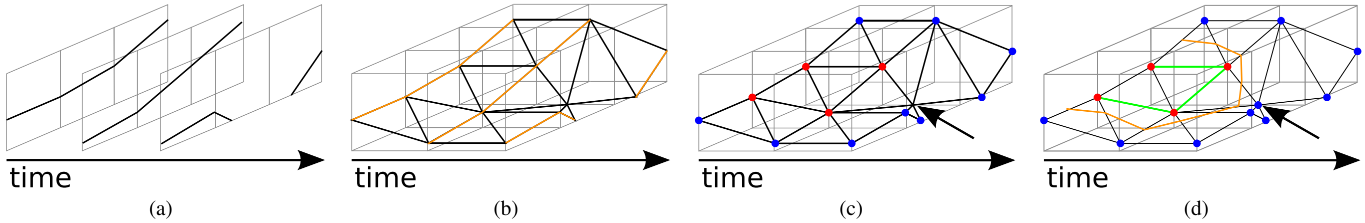
We optimize the basic algorithm described above in two important ways: The first optimization avoids extracting a true space-time isosurface between burning and non-burning regions. Later in the process cells will be tracked by computing the Reeb graph of the final space-time isosurface. The Reeb graph, however, only depends on the connectivity of the surface not its geometry. Therefore, instead of computing intersections along edges connecting burning to non-burning vertices we snap such vertices to the ones labeled burning. It can be shown that the resulting surface has the identical connectivity to the “true” space-time surface but can be computed by simple filter operation, see [36].

Second, we intersect the isovolume with planes of constant time at each of the original time steps of the simulation. Note, that for the original time steps this intersection reduces to another filtering operation and thus can be performed efficiently in a streaming fashion, see [36]. The resulting isosurface is then segmented according to Sec-



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

Fig. 5. Tracking burning regions. Like in the cumulative analysis, we restrict our considerations to a particular isotherm (a), on which fuel consumption is mapped as a scalar field. Thresholding triangles of the isotherm by fuel consumption allows identification of individual burning regions (b). We track these individual burning regions by considering their boundaries, which can be obtained in a second contouring operation (c). Traced over time, these boundaries sweep surfaces in 4D space (the figure shows their path in 3D space) (d). The Reeb graph of this time surface is the desired tracking graph.

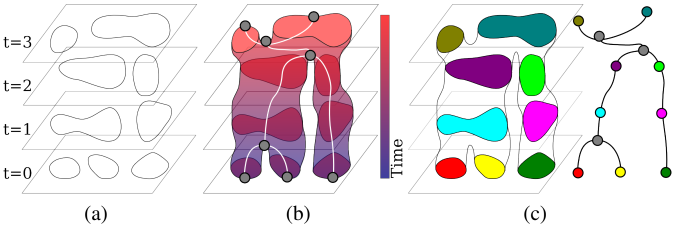


(a) (b) (c) (d)

Fig. 6. Tracing the evolution of 2D burning regions and extinction pockets by tracking their boundaries. (a) Input data comes as a set of discrete time slices. (b) Treating time as additional, third dimension we extract a space-time surface correlating isotherms from different time steps. The isotherms (contour lines in 2D) for all original time steps area extracted by selecting all lines that have only vertices in the time step of interest (orange lines). (c) Isotherm vertices at original time steps classified as either burning (red) or non-burning (blue) based on the local fuel consumption rate. Note that vertices between time steps (arrowed vertex) are not classified, yet. (d) Classify vertices between time steps (arrowed vertex) by thresholding and extract boundaries separating burning regions and extinction pockets (orange lines).

tion 3.1, the segmentation specialized for a persistence of 0.1 and a fuel consumption threshold of 2.6, and all burning vertices are labeled with their corresponding cell id. The specific parameter values are determined by extensive parameter studies described in Section 4.

Computing the Reeb graph of the time function on this space-time isosurfaces is the tracking graph of the cells, as shown in Fig. 7. Furthermore, all vertices at original time-steps are labeled by cell id and we augment the Reeb graph with this information if necessary by adding valence two nodes. This allows to correlate specific cells of the flame surface with points in the tracking graph. This correlation is crucial when analyzing the graph.



(a) (b) (c)

Fig. 7. Using the Reeb graph to compute a tracking graph. (a) Several time steps of a 2D data set with the boundaries of several cells indicated; (b) The space-time boundary created by the boundaries of the cells in (a) as they are interpolated over time. The surface is colored coded using time as function and its Reeb graph is shown in white and gray; (c) Each cell within a time step is assigned a unique identifier which we use to augment the Reeb graph. Note that the final tracking graph still contains nodes between time steps without identifier (shown in gray).

Once we have computed the Reeb graph we simplify all loops that span less than a full time step since they must represent artifacts of the construction (using linear interpolation between time steps no true feature should exist between time steps). Finally, we simplify all loops spanning exactly one time step to streamline the graph. This helps resolving segmentation instabilities caused by saddles close to the threshold value. As shown in Fig. 8 one sometimes finds “extended” split/merge events in which two regions merge and split several times before fully merging/splitting. We also remove components of the graph that have an overall life-span of less than two time steps. We use dot to layout the resulting graph [20].

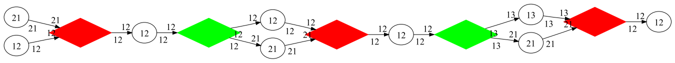
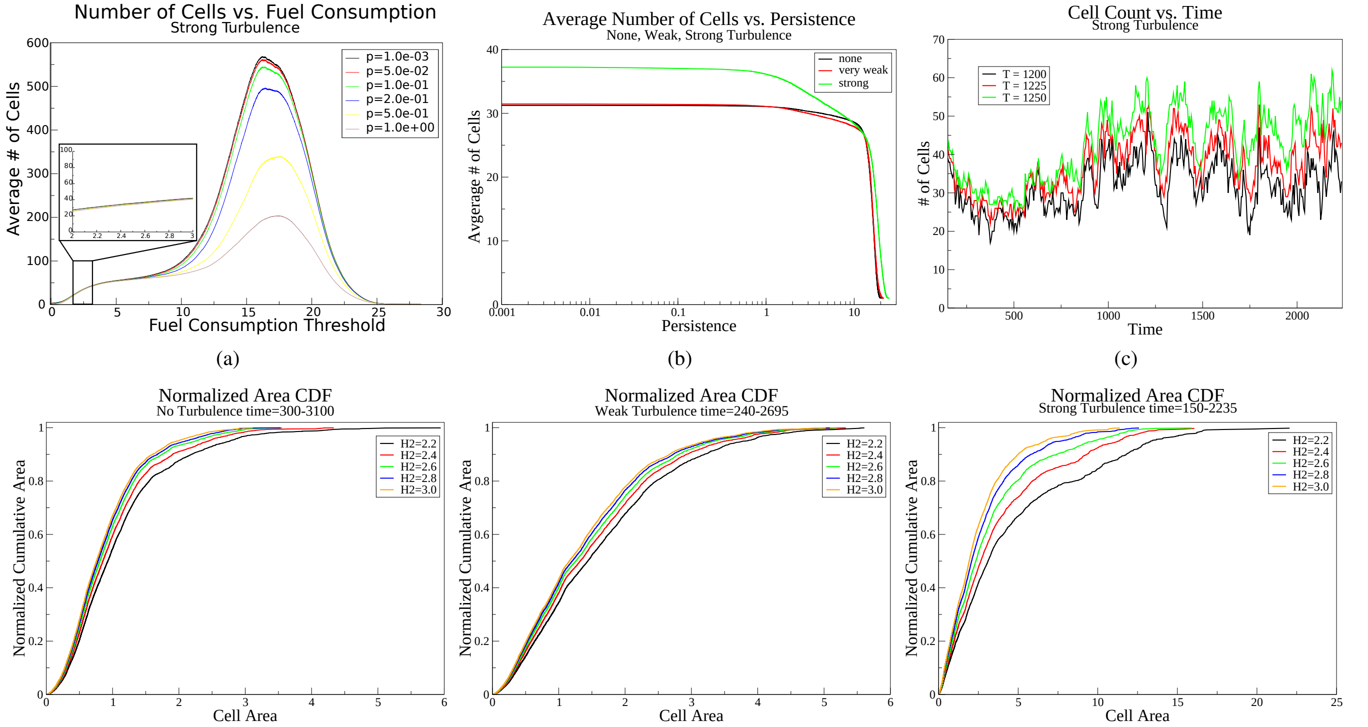


Fig. 8. An extended merge event: Two cells (initially labeled 12 and 21) merge and split twice before ultimately merging. These are inherent instabilities in segmenting by thresholding caused by saddles very close to the cut-off.

4 Parameter Studies

An often overlooked aspect of data analysis techniques is the ability to easily change selection parameters. In practice, application scientist are typically less interested in a single set of parameters but rather in understanding the phenomenon in question within a range of operational parameters. Furthermore, studying the sensitivity of the analysis with respect to the input parameters is an important piece of information in its own right. A key feature of the hierarchy constructed in Section 3.1 is the ability to quickly specialize it to any given set of parameters. To demonstrate advantages of this flexibility and the stability of the topology-based analysis in general this section presents a detailed parameter study for the three combustion simulations. It is important to note that each simulation was only processed once on a large 32 core server. All data presented in this paper was then extracted from the stored hierarchies in a post-processing step on a standard laptop.

The goal of the combustion research featured here is to analyze and compare three different numerical simulations of lean hydrogen flames interacting with varying levels of turbulence. The focus is on understanding how many independent burning cells exist at any one time, their areas, and evolution over time. Since this is the first time burning cells have been analyzed in this fashion it is important to understand the influence of various parameters on the analysis. Here, we present



(a) (b) (c)

(d) (e) (f)

Fig. 9. Parameter study: (a) Average number of cells per time steps depending on the fuel consumption threshold for various persistences; (b) Average number of cells depending on persistence; (c) Number of cells over time for three different temperature value in the strong turbulence case; (d)-(f) Cumulative density function of cell area weighted by overall burning area for various fuel consumption thresholds in the none, weak, and strong turbulence case respectively.

a sensitivity analysis with respect to persistence, temperature isovalue, and fuel consumption threshold as well as a comparison between the two levels of spatial resolutions. Unless specifically mentioned all graphs show the collective statistics of time steps 300, 240, and 150 onward for the none, weak, and strong turbulence case respectively using the coarse version of the data. Whenever, for brevity, only one of the three cases can be shown the strong turbulence case is used as it is the most volatile and expected to show the largest variance. Finally, the baseline values used for the final analysis are a temperature of 1225 Kelvin, a persistence of 0.1, and a fuel consumption threshold of 2.6.

Data. For each of the three simulations every fifth time step of a long time series was saved using a uniform grid of size 256x256x768 and a shorter temporal subset given at every single time step using a 512x512x1536 grid. For the turbulence free case the simulation ended at time $t = 3100$ resulting in 621 coarse time steps. In addition 102 fine resolution time steps have been saved covering the interval $[511, 612]$. For the weak turbulence case the simulation ended at time step 2695 providing 540 coarse time steps together with 82 fine resolution time step in the $[431, 512]$ interval. Finally, the strong turbulence case ended at time step 2235 producing 448 coarse files in addition to 91 fine resolution time steps covering time steps $[540, 630]$. Each time step consists of two floating point values: the temperature and the fuel consumption rate. Overall the input amounts to roughly 400GB of compressed floating point data.

Persistence. As discussed in Section 2 the persistence of the cancellation that merges two burning cells is the absolute distance in function value (here the fuel consumption) between the lower maximum and the shared saddle. Thus simplifying per persistence merges cells containing "shallow" maxima. In previous applications of topological analysis persistence has been shown to have a significant influence on the results [21, 15]. In this context, small persistences are used to remove noise while larger persistences to perform a multi-scale analysis. Nevertheless, in these cases persistence has been the only parameter acting on the various topological segmentations while in the work presented here the highest-saddle-first hierarchy is of primary interest. In particular, a high persistence threshold might skew the results as persistence based simplification would allow stable manifolds to be merged via saddles below the given fuel consumption threshold. In the worst case an absolute persistence of 1.0 at a fuel consumption threshold of 2.6 could allow a saddle with fuel consumption rate of 1.6 to be canceled. Effectively, this would count the offending saddle as burning even though this would be clearly undesirable. Therefore, the persistence should be restricted to a value significantly below 1.0.

The primary measure effected by varying the persistence is the number of cells. Fig. 9(a) shows the number of cells depending on the fuel consumption threshold for various persistences in the strong turbulence case. While the persistence does have a significant impact on the count for high fuel consumption thresholds its influence is insignificant in the vicinity of physically meaningful thresholds (around 2.6). Fig. 9(b) shows the average number of burning cells in each time step versus persistence for all three simulations. Clearly, the count is extremely stable up to persistences far beyond the physically meaningful range.

Temperature. The temperature value defining the flame front is the only parameter not easily adjusted in our frame work. For each temperature a separate set of surfaces must be extracted which given the large number of time steps takes several hours. However, the temperature value is usually hand-picked by the domain scientists using traditional visualization techniques making it less susceptible to miscalculation. Nevertheless, it is important to understand the stability of the results with respect to this choice. As a first test of stability we compare the cell count over time for three different choices of temperature namely 1200, 1225, and 1250 Kelvin. Fig. 9(c) shows the number of burning cells over time for the strong turbulence case. All three graphs show qualitatively the same behavior. Furthermore, the region count strictly increases with the temperature value. This suggests that the count is stable except that in some instances flame fronts at higher temperatures do not contain some of the "bridges" of high fuel consumption between cells. Therefore, cells that could merge for lower temperatures no longer have a connection thus increasing the count. Finally, Fig. 12(a) shows the area weighted normalized cumulative density function (NCDF) of the distribution of cell areas for the three temperature values. The normalized cumulative density function of cell areas for an area a is defined as the fraction of cells with size $\leq a$. For example, at 1200 Kelvin about 40% of the patches are larger than 4 a ratio that decreases to 30% for 1225 Kelvin, and to under 20% for 1250 Kelvin. Overall, the three graphs show an identical behavior with the only difference being that lower temperature values on average produce slightly larger cells. This is unsurprising as lower temperature surfaces have a larger overall surface area.

Fuel Consumption Threshold. The most important parameter in the analysis is the threshold of fuel consumption which defines what parts of the flame surface are considered burning. Fig. 9(d), (e), and (f) show the NCDF of cell areas for the none, weak, and strong turbulence case. As expected lower thresholds produce larger cells as more of the surface area is considered burning. Furthermore, the graphs for lower thresholds show long flat tails indicating the presence of few outliers with unusually large areas. While all graphs show the same trend they seem to become more stable for higher thresholds. Overall, this suggests that higher thresholds are likely to produce more stable results. However, based on the physical interpretation the fuel consumption threshold should be set rather lower around 2.0-2.6. Based on our analysis it was determined to choose the highest physically meaningful threshold of 2.6 as optimal value.

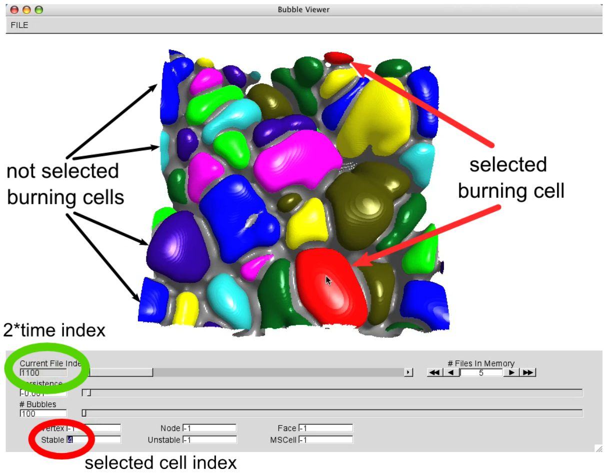
Spatial Resolution. As mentioned in Section 3 for each simulation a small temporal subset of about 100 time steps was computed with an additional level of subdivision and dumped every time step rather than every fifth. Once flattened into a regular grid these time steps have eight times the spatial resolution of the coarse data. To determine whether the higher resolution provides additional information we performed the full analysis and compared the results. Fig. 12(b) shows the NCDF of the high resolution data in solid and the NCDFs of the corresponding temporal sub-set of the coarse data as dotted lines. Considering, that the high resolution data is given at eight time the spatial and five time the temporal resolution the curves match remarkably well. The high similarity between the different resolutions allows us to use the coarse data with a high degree of confidence. As a result the full time sequences can be used providing a better statistical sampling. In fact, the statistics over the complete series look somewhat different from those of the temporal sub-set shown here. While the curves in Fig. 12(b) suggest a linear fit the final NCDFs show a clear logarithmic behavior, see Fig. 12(c).

5 Temporal Analysis

In addition to the statistical analysis we track all burning cells in the three cases using the coarse level time series. We started the tracking once the simulation are considered to be in a "steady state" which is time step 300 in the turbulence free, time step 240 in the weak, and 150, in the strong turbulence case. Since for this resolution only every fifth time step of the original simulation was stored we sub-sampled the coarse resolution in time creating one intermediate time step between each pair of original data files.

For each dataset we have created the full tracking graph and used it to render the accompanying animations. As discussed in Section 3.2 tracking graphs contain a node for every time step and every arc. However, to reduce visual clutter only nodes with a valence different from two are shown in the figures and movies. Since the complete graphs have an extreme aspect ratio and are very large they are provided as additional material rather than shown directly in the paper. To analyze the graphs we create multiple versions for each simulation using various levels of simplification and coloring. The simplification removes small scale (on the temporal scale) features primarily due to instabilities in the thresholding and makes the graph noticeably less cluttered. We also experimented with different colorings of the graphs to highlight various properties. Given the temporal interpretation of the graphs it is natural to color nodes at either "end" and observe the effects as the color propagates through the graph. However, the large number of splits make it impossible to create unique (and still recognizable different) colors for each branch. Furthermore, the existing merges often combine two arcs of different color leading to the removal of one color from the propagation. As a result there exist no single uniquely appropriate coloring algorithm and instead we focus on presenting different schemes to the user each with their advantages and disadvantages. In particular, we have applied four different strategies: a) In the first time step the four cells with largest fuel consumption are chosen and assigned a unique color. Each time a cell splits all children inherit the color and if two colored cells merge the color stemming from the arc with higher maximal fuel consumption passes on its color to the child; b) The same strategy as in a) is used but for the cells with the fourth to eighth largest fuel consumption; c) The eight cells with the highest fuel consumption in the first time step are assigned a color. Each time a cell splits only the child with the highest fuel consumption value among the children inherits the color. If two colored cells merge the cell with the previously higher fuel consumption passes on its color; and d) The same strategy as in c) is applied in reverse by coloring the eight cells with the largest fuel consumption in the last time step propagating the color backwards. The a),c), and d) coloring strategies are shown in the animations with colors matching the arc colors in the corresponding graphs.

Color strategies a) and b) provide a general notion of how cells in early time steps influence cells in later time steps. For example, in the turbulence free case all burning cells at the end of the simulations are descendent from the four cells with highest fuel consumption in the beginning. However, this is not true for the cell with fourth to eights highest fuel consumption. The differences between a), b) and c), d) indicate that a high maximal fuel consumption is not necessarily a good indicator for cell survival. While passing on a color to all children quickly colors most of the graph coloring only the child with highest fuel consumption has most colors disappearing.



Current File Inde 1100
Files In Memory 5
rsistence -0.001
Bubbles 100
Vertex -1 Node -1 Face -1
Stable 4 Unstable -1 MSCell -1

Fig. 10. Annotated screen-shot of the interactive user interface. Showing cell 4 of time step 550 highlighted in the turbulence free case, see Fig. 11.

Fig. 11 shows a small portion of the tracking graph for the turbulence free case using color scheme b) modified to only show colored branches. Round nodes indicate burning cells segmented from an actual flame surfaces using the Morse complex. The numbers inside the nodes and along the branches indicate the identifier assigned to this particular cell in the segmentation. This information allows us to quickly find cells of interest and also to color cells consistently in time. The diamond shapes represent topological events in between time steps and thus have no identifier attached. To provide better navigation we use the space inside the diamonds to indicate the simulation time. Red/green nodes and diamonds signal merge/split events and turquoise structures a birth or death event. Using the renderings above and below the graph one can follow the event chain.

Between time 550 and 552.5 the purple cell 4 merges with cell 18 via a small connection across the lower boundary (as mentioned above the simulation is periodic in x and y). Subsequently, what used to be

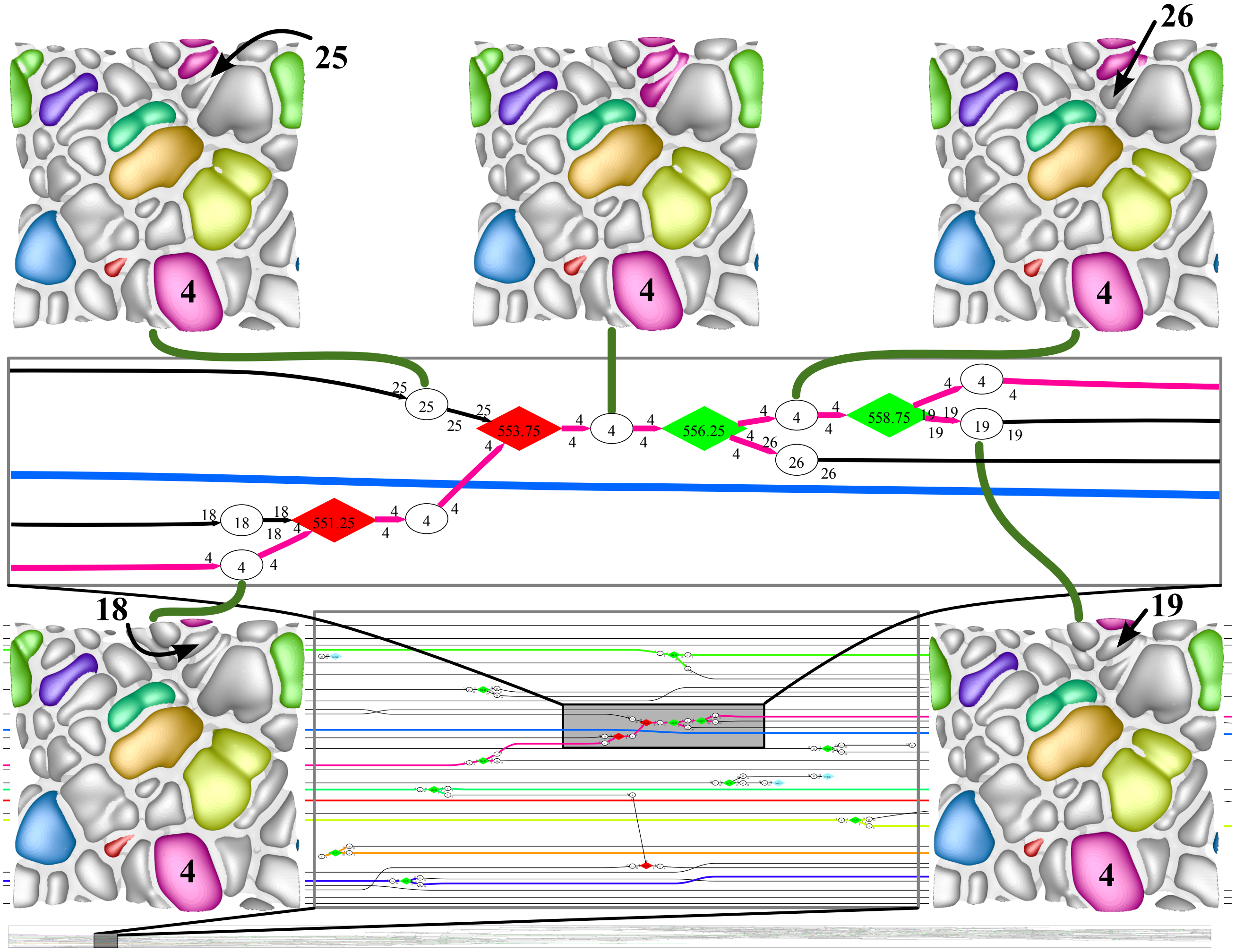


Fig. 11. Subsection of the tracking graph in the turbulence free case compressed to only show the colored arcs together with several corresponding segmentations. Round nodes correspond to cells explicitly segmented by the Morse complex, diamonds to topological events between time steps. Red signifies a merge, green a split, and turquoise a birth/death event.

cell 18 develops a connection with cell 25 resulting at time 555 in the three purple areas all being connected via small bridges into a single cell. At time 557.5 the cell formerly labeled 25 has split leaving a small portion attached to cell 4 and creating a new cell 26. Finally, at time 560 the remains of cells formerly labeled 18 and 25 have split off cell 4 forming a separate cell 19. A useful side-effect of the construction algorithm of [5] is that the labels within each time step correspond to the relative fuel consumption of the highest maximum within each cell. Therefore, the cell with overall highest fuel consumption is labeled 1 the next highest 2 etc.. As seen in Fig. 11 this provides consistency in the labeling across time steps even though the segmentations are performed independently. This consistency is not perfect nor is it strictly required since the graphs provide enough information to connect labels across time. However, it does make following the graph easier and more intuitive.

Besides the illustrations of the graphs and the corresponding animation we also developed a tool to interactively explore the time correlated isosurfaces. To this end we store the results of the segmentation (one 8-bit index per triangle) in a separate set of files. To explore a graph these files a read along side the original surfaces which allows us to color the surfaces by cell id. To support exploration of the entire time series we implemented a simple pre-fetching scheme that always keeps a user-defined number of surfaces and segmentations in memory. Fig. 10 shows a screen shot of the user interface that allows to interactively choose a time step as well as a cell index. At any time all burning cells are randomly colored using eight unique colors and a single selected cell/stable manifold is colored red. Together with a pdf viewer showing the corresponding graph the user can follow each event in the graph in detail. To further illustrate the user interface we have provided a live screen capture of the in-depth analysis of Fig. 11 using our tool. To facilitate understanding we added the relevant portions of the graph of Fig. 11 to the animation with the current time step highlighted in grey. As noted above, the inherent consistency in the labeling process makes following cell through time easy. In the majority of cases labels will persist over time thus relieving the user from continuously updating the highlighted cell index. Since the user interface is based entirely on a pre-computed segmentation it can be applied to a variety of surface based feature tracking applications such as, for example, the one presented by Laney et al. [21].

6 Results

Here, we present the results of the statistical analysis as well as the tracking. To illustrate the results we have provide a number of animations as well as pdf files of all graphs in the accompanying materials. Most of the computation was performed in parallel at the National Energy Research Scientific Computing Center at Berkeley using both an SGI Altix 350 server with 32 Intel Itanium-2 processors and a small portion of a 712-CPU Opteron cluster. Running on a single Itanium-2 processor the isovolume extraction between two time steps takes around eight minutes, extracting the isotherm for a single time step takes approximately 5 minutes, the segmentation of the resulting flame surface 20 seconds, and the extraction of the corresponding portion of the space-time boundary surface three and a half minutes. Merging all space-time partial surfaces into a single file takes around

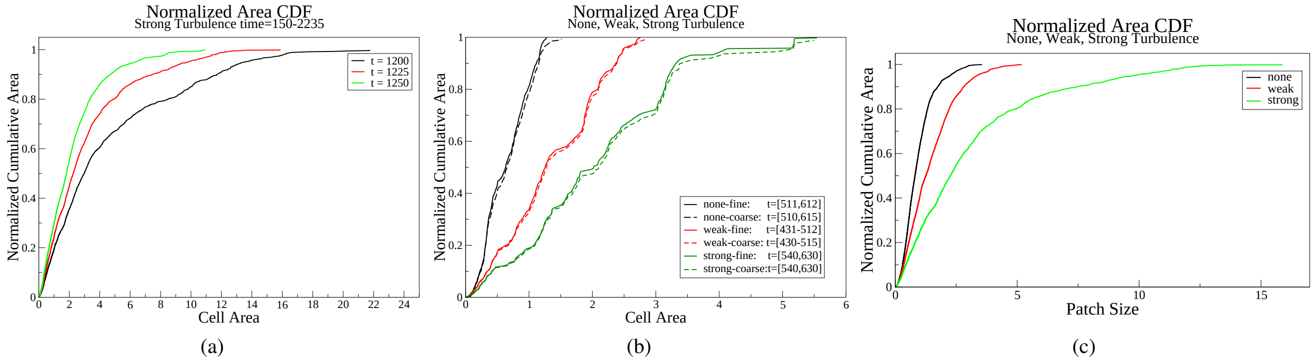


Fig. 12. Parameter study and results: (a) Cumulative density functions of the cell areas weighted by overall burning area for three temperature values in the strong turbulence case; (b) Comparison of the NCDF functions on fine and coarse resolution for all three cases. (c) NCDF function of the cell area distribution. As turbulence increases the distributions skews towards larger cells.

30 minutes. The final space-time surface, stored in obj format, occupies between 2GB in the none and weak turbulence case and 3GB in the strong turbulence case. The main bottleneck in isovolume and isotherm extraction as well as merging results of two time slices is file I/O, which accounts for approximately two thirds of the processing time. For the purpose of these measurements, we used the time steps that produced the largest output files. The Reeb graph computation was performed on a 2.33 GHz Intel Core Duo laptop with 2GB main memory in 6, 6.5, and 11 minutes for the none, weak, and strong case turbulence case respectively.

As discussed in Section 4 storing the segmentation as hierarchical Morse complex enabled us, for the first time, to perform extensive parameter studies. This provides the scientist with powerful tools to pick a final set of parameters and validate the results. Furthermore, our analysis demonstrates that, at least for the statistical data presented here, the coarser resolution of the simulations contains all necessary information. As a result the fine resolution computation might be skipped in future simulations significantly increasing the performance while decreasing the storage requirements.

The most significant scientific result is summarized in Fig. 12(c). It shows the normalized cumulative density function of the distribution of cell areas for the three simulations, and indicates a clear trend towards larger flame cells with increasing turbulence intensity. This trend is perhaps counterintuitive, since flows with a higher turbulence intensity contain more energetic fine-scale modes which may interact with the laminar flame. Apparently, however, these interactions do not tear apart the flame at smaller scales, but rather significantly modify the development of the thermodiffusive instabilities that lead to the cellular burning structures in the non-turbulent case.

As can be seen in the accompanying animations one reason for the increased cell size is that in the stronger turbulence, cells in the process of splitting remain attached longer. Fig. 13 shows a snapshot of each simulation in which a small number of single cells are highlighted. For the turbulence free case the cells remain somewhat convex, Fig. 13(a), while for the weak and especially the strong case more sprawling cells develop. The increasing "connectedness" of burning pockets creates overall larger cells which explains the curves in Fig. 12(c). Overall, our analysis suggests that taking turbulence into account might allow a stable combustion process using a much leaner hydrogen-air mixture than previously thought possible.

The tracking graphs presented here allow, for the first time, to correlate burning cells over time and to better understand their evolution. For example, based on simulations of flat laminar flames it has been expected that cells only split and do not merge. While the graphs indeed predominantly show a splitting behavior there are a significant number of merges present in all three simulations. As part of the future work it will be interesting to extend the statistical analysis to the graphs themselves determining whether there exist significant differences in the cell evolution.

While the tracking is generally very stable there do exist rare artifacts due to the low temporal resolution. In the unsimplified graphs one can sometimes find merge-split nodes which connect two regions in between time steps even though the Morse segmentation before and after shows two separate cells. These are generally artifacts of the low time resolution most likely coupled with tetrahedralization artifacts. The cells in question are usually spatially close and convoluted. In such cases the space-time isovolume can create spurious connections. A similar problem exists sometimes for dying cells which often consist of only a handful of vertices. Such cells tend to move fast while shrinking rapidly which results in a lost connection. In the graphs these dying cells appear as short lived components that are born and die sometimes with a single time step.

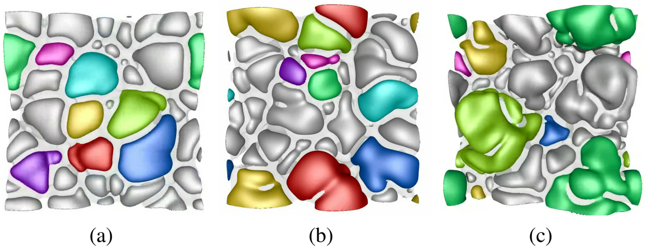


Fig. 13. Snapshots of early time steps in the turbulence free (a), weak (b), and strong turbulence (c) case. Selected burning cells are shown in color, non-selected cells in grey. The non-burning portions of the flame surfaces are drawn translucent. As the turbulence level increases cells become less convex and develop a higher genus.

7 Conclusions

We have presented a novel hierarchical segmentation strategy which allows us to analyze and track features embedded in time-dependent isosurfaces. We use these techniques to study burning cell structures in simulated lean premixed hydrogen flames. In a single pass over the data we collect a large amount of statistical information revealing new insights into the combustion process. In particular, the size and intensity of the burning cells are quantified and tracked over time to investigate the effect of turbulence on the propagation of these structures. Furthermore, we have presented new algorithms to track features over time by computing the Reeb graph of the boundary of their space-time volume and demonstrated an interactive tool to visualize the results. Overall, the procedures described in this paper provide the application scientists with fundamentally new analytic capabilities and thus significantly advance the state of the art.

In the future, we are aiming to extend our techniques to a fully three-dimensional segmentation to eliminate the need of choosing a temperature isovalue. Furthermore, we plan to statistically analyze the graphs and in particular collect time-dependent characterizations of individual cells. Finally, displaying the graphs is in itself an interesting research problem due to their large size and extreme aspect rations. We are currently exploring new graph drawing algorithms and in particular the possibility to provide linked views between the interactive user interface and a graph visualization.

ACKNOWLEDGEMENTS

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-JRNL-412904L. This work was supported by:

The Director, Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 through the Scientific Discovery through Advanced Computing (SciDAC) program's Visualization and Analytics Center for Enabling Technologies (VACET);

The SciDAC Program of the DOE Office of Mathematics, Information, and Computational Sciences under the U.S. Department of Energy under contract No. DE-AC02-05CH11231. Computational resources have been made available on the Fraklin machine at NERSC as part of an INCITE award and on the Columbia machine at NASA as part of an National Leadership Class System allocation; and

The National Science Foundation (NSF) through the Topology-based Methods for Analysis and Visualization of Noisy Data project.

This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

REFERENCES

[1] P. K. Agarwal, H. Edelsbrunner, J. Harer, and Y. Wang. Extreme elevation on a 2-manifold. *Discrete Comput. Geom.*, 36(4):553–572, 2006.

[2] J. B. Bell, R. K. Cheng, M. S. Day, and I. G. Shepherd. Numerical simulation of Lewis number effects on lean premixed turbulent flames. *Proc. Combust. Inst.*, 31:1309–1317, 2007.

[3] P. Bhaniramka, R. Wenger, and R. Crawfis. Isosurface construction in any dimension using convex hulls. *IEEE Transactions on Visualization and Computer Graphics*, 10(2):130–141, 2004.

[4] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 10(4):385–396, 2004.

[5] P.-T. Bremer and V. Pascucci. A practical approach to two-dimensional scalar topology. In *Topology-based Methods in Visualization (Proc. TopoInVis 2007)*, pages 151–171. Springer Verlag, 2007.

[6] A. Cayley. On contour and slope lines. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, XVIII:264–268, 1859.

[7] 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.

[8] M. S. Day and J. B. Bell. Numerical simulation of laminar reacting flows with complex chemistry. *Combust. Theory Modelling*, 4(4):535–556, 2000.

[9] H. Edelsbrunner, M. Facello, and J. Liang. On the definition and the construction of pockets in macromolecules. *Discrete Appl. Math.*, 88(1-3):83–102, 1998.

[10] H. Edelsbrunner and J. Harer. Jacobi sets of multiple Morse functions. In *Foundations of Computational Mathematics, Minneapolis 2002*, pages 37–57. Cambridge University Press, England, 2002.

[11] H. Edelsbrunner, J. Harer, A. Mascarenhas, and V. Pascucci. Time-varying Reeb graphs for continuous space-time data. In *20th Symposium on Computational Geometry*, pages 366–372. ACM Press, 2004.

[12] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-Smale complexes for piecewise linear 3-manifolds. In *Proc. 19th Sympos. Comput. Geom.*, pages 361–370, 2003.

[13] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. *Discrete Comput. Geom.*, 30:87–107, 2003.

[14] I. Fujishiro, R. Otsuka, S. Takahashi, and Y. Takeshima. T-map: A topological approach to visual exploration of time-varying volume data. In *Proc. of 6th International Symposium on High Performance Computing*, pages 176–190, 2008.

[15] A. Gyulassy, M. Duchaineau, V. Natarajan, V. Pascucci, E.Bringa, A. Higginbotham, and B. Hamann. Topologically clean distance fields. *IEEE Transactions on Computer Graphics and Visualization*, 13(6):1432–1439, 2007.

[16] A. Gyulassy, V. Natarajan, V. Pascucci, P.-T. Bremer, and B. Hamann. Topology-based simplification for feature extraction from 3D scalar fields. *IEEE Transactions on Computer Graphics and Visualization*, 12(4):474–484, 2006.

[17] A. Gyulassy, V. Natarajan, V. Pascucci, and B. Hamann. Efficient computation of Morse-Smale complexes for three-dimensional scalar functions. *IEEE Transactions on Computer Graphics and Visualization*, 13(6):1440–1447, 2007.

[18] G. Ji and H.-W. Shen. Efficient isosurface tracking using precomputed correspondence table. In *Proc. Symposium on Visualization (VisSym) '04*, pages 283–292. Eurographics Association, 2004.

[19] G. Ji, H.-W. Shen, and R. Wegner. Volume tracking using higher dimensional isocontouring. In *Proc. IEEE Visualization 2003*, pages 209–216. IEEE Computer Society, 2003.

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

[21] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization 2006)*, 12(5):1052–1060, 2006.

[22] J. C. Maxwell. On hills and dales. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, XL:421–427, 1870.

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

[24] M. Morse. Relations between the critical points of a real functions of n independent variables. *Transactions of the American Mathematical Society*, 27:345–396, July 1925.

[25] L. R. Nackman. Two-dimensional critical point configuration graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(4):442–450, July 1984.

[26] V. Pascucci, G. Scorzelli, P.-T. Bremer, and A. Mascarenhas. Robust on-line computation of reeb graphs: Simplicity and speed. *ACM Transactions on Graphics (Proc. SIGGRAPH 2007)*, pages 58.1–58.9, 2007.

[27] J. Pfaltz. Surface networks. *Geographical Analysis*, 8:77–93, 1976.

[28] G. Reeb. Sur les points singuliers d'une forme de pfaff completement intergrable ou d'une fonction numerique [on the singular points of a complete integral pfaff form or of a numerical function]. *Comptes Rendus Acad. Science Paris*, 222:847–849, 1946.

[29] F. Reinders, F. H. Post, and H. J. W. Spoelder. Visualization of time-dependent data with feature tracking and event detection. *The Visual Computer*, 17(1):55–71, 2001.

[30] R. Samtaney, D. Silver, N. Zabusky, and J. Cao. Visualizing features and tracking their evolution. *IEEE Computer*, 27(7):20–27, 1994.

[31] Y. Shinagawa, T. L. Kunii, H. Sato, and M. Ibusuki. Modeling contact of two complex objects: With an application to characterizing dental articulations. *Computers and Graphics*, 19(1):21–28, 1995.

[32] D. Silver and X. Wang. Tracking and visualizing turbulent 3d features. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):129–141, 1997.

[33] D. Silver and X. Wang. Tracking scalar features in unstructured datasets. In *Proc. IEEE Visualization '98*, pages 79–86. IEEE Computer Society Press, 1998.

[34] B.-S. Sohn and C. Bajaj. Time-varying contour topology. *IEEE Transactions on Visualization and Computer Graphics*, 12(1):14–25, 2006.

[35] A. Szymczak. Subdomain-aware contour trees and contour tree evolution in time-dependent scalar fields. In *Proc. Shape Modeling International (SMI) '05*, pages 136–144. IEEE Computer Society, 2005.

[36] G. Weber, P.-T. Bremer, J. Bell, M. Day, and V. Pascucci. Feature tracking using Reeb graphs. In *Topology-based Methods in Visualization (Proc. TopoInVis 2009)*. Springer Verlag, 2009. In review.

[37] X.Zhang and C. Bajaj. Extraction, visualization and quantification of protein pockets. In *Proc. of 6th Annual International Conference on Computational System Bioinformatics Conference (CSB2007)*, pages 275–286, 2007.