Feature Mining Paradigms for Scientific Data[*]

Ming Jiang[†] Tat-Sang Choy[‡] Sameep Mehta[†] Matt Coatney[†] Steve Barr[‡]
Kaden Hazzard[‡] David Richie[§] Srinivasan Parthasarathy[†] Raghu Machiraju[†]
David Thompson[¶] John Wilkins[‡] Boyd Gatlin[¶]

Abstract

Numerical simulation is replacing experimentation as a means to gain insight into complex physical phenomena. Analyzing the data produced by such simulations is extremely challenging, given the enormous sizes of the datasets involved. In order to make efficient progress, analyzing such data must advance from current techniques that only visualize static images of the data, to novel techniques that can mine, track, and visualize the important features in the data. In this paper, we present our research on a unified framework that addresses this critical challenge in two science domains: computational fluid dynamics and molecular dynamics. We offer a systematic approach to detect the significant features in both domains, characterize and track them, and formulate hypotheses with regard to their complex evolution. Our framework includes two paradigms for feature mining, and the choice of one over the other, for a given application, can be determined based on local or global influence of relevant features in the data.

Keywords: *feature mining, computational fluid dynamics, molecular dynamics, shock detection, vortex verification, defect evolution*

1 Introduction

The physical and engineering sciences are increasingly concerned with the study of complex, large-scale evolutionary phenomena. Such studies are often based on analyzing data generated from either traditional experiments or numerical simulations. Given recent concurrent advances in computer hardware and numerical methods, it is now possible to simulate physical phenomena at very fine spatial and temporal resolutions. As a result, the amount of data generated is overwhelming and unprecedented. Examples of such large-scale simulations can be found in numerous science domains, including computational fluid dynamics and molecular dynamics. Computational fluid dynamics (CFD) seeks to understand flow patterns to enhance, for instance, drug delivery schemes for pulmonary treatments of asthma. Molecular dynamics (MD), on the other hand, seeks to understand the evolution of defect structures that can affect the properties or performance of industrial materials.

The size of many simulation datasets significantly challenges our ability to explore and comprehend the generated data. Currently, a well-trained individual may need several days, or even weeks, to analyze the data generated by an MD simulation and create a list of viable defect structures. Similarly, in the extremely large datasets generated by simulations of complex fluid flows, locating and tracking relevant features are daunting tasks, given their large number and vast multitude of interactions.

Therefore, we believe it is crucial that some degree of automation be incorporated into the data exploration process for large-scale datasets. One such successful approach is described in [24] and is based on a representational scheme that facilitates ranked access to macroscopic features in the dataset. However, other than identifying, denoising, and ranking the features, no attempt is made to extract useful information about the features, such as geometrical and dynamical attributes, that can be used for tracking features.

A seemingly obvious approach would be to apply traditional data mining techniques to these scientific data. However, it is our contention that existing data mining techniques, applied in isolation, are simply too general to be of any use for our applications. One way to remedy this problem is to embed domain expertise in the data mining process. While some data mining techniques do allow domain expertise to be incorporated in specific ways, in general, they are not flexible enough

[*] Supported by the National Science Foundation (ACI-9982344, ACS-0085969, EEC-0121807, 764AT-51028A), the Department of Energy (DE-FG02-99ER45795), and the U.S. Army Research Office (DAA-D19-00-1-0155).

[†] Dept. of Computer and Information Science, The Ohio State University, {jiang,srini,raghu}@cis.ohio-state.edu

[‡] Dept. of Physics, The Ohio State University

[§] High Performance Technologies, Inc.

[¶] Dept. of Aerospace Engineering, Mississippi State University

to meet all the subtle requirements of domain experts. Furthermore, the application of traditional data mining techniques may not be the most efficient of solutions, particularly for analyzing time-varying simulation data, which can easily surpass the terabyte range. What is needed, in this case, are linear- or quasilinear-time algorithms, as opposed to high-order polynomial-time algorithms.

To address these perceived shortcomings, we are developing a feature-centric unified framework for mining scientific data that we term *generalized feature mining* and are applying it to CFD and MD simulation data. Our intent is to exploit the underlying physics of the problems in order to develop highly discriminating, application-dependent detection, characterization, and tracking algorithms for features of interest. Then using available data mining algorithms where appropriate, to classify, cluster, and categorize the identified features. We further claim that the large-data exploration methodology we describe is sufficiently general so that other application domains can be incorporated into our approach. This claim is demonstrated by the application of our framework to two disparate application domains.

It should be noted that Yip and Zhao [43] proposed a similar, albeit more general, framework. They relied on spatial aggregation to cluster both physical and abstract entities and constructed imagistic solvers to gain insights into physical phenomena. The main outcome is a spatial aggregation language (SAL) which is offered as the basis for further data mining or exploration. Their claim, as is ours, is that feature mining, including aggregation and tracking, is the first step towards a comprehensive analysis of scientific data. Our work differs from theirs in that we focus more on three-dimensional computational datasets, demonstrate our framework on time-varying datasets, and exploit more of the underlying physics.

A version of our physics-based feature mining approach appeared recently in [40]. However, we did not describe it as part of our unified framework for mining scientific data; moreover, we only demonstrated its effectiveness in the CFD application domain. In this paper, we demonstrate the generality of our feature mining paradigms by including results from both CFD and MD simulations. The remainder of this paper is organized as follows. In Section 2, we describe related research conducted elsewhere. Section 3 presents the two feature mining paradigms in detail. Section 4 shows preliminary results we have obtained so far for the two science domains. Finally, we summarize our results and provide a road map for future work in Section 5.

2 Related Work

Knowledge discovery and data mining (KDD) refers to the overall process of discovering new patterns or building models from a given dataset. Fundamental KDD research in the last decade has primarily focused on: i) new techniques to preprocess, mine, and evaluate the data, ii) efficient algorithms that implement these techniques, and iii) the applications of above techniques in business and marketing domains.

More recently, researchers have started tackling the problem of mining scientific data. In particular approaches for mining astronomical [5, 16], biological [8, 21, 41], chemical [7], and fluid dynamical [11] datasets have been recently proposed by various researchers. Few of the above methods actually account for the kinematically and dynamically driven characteristics of the data. Abstract representations in the form of graphs are often extracted and variants of the traditional frequent itemset discovery technique are employed to glean insights into the physical phenomena.

Frequent substructure discovery is a pertinent example of traditional data mining techniques for scientific data, with applications ranging from fluid dynamics to chemical compounds. By modeling the features in a dataset with graphs, the problem of finding frequent patterns in the dataset becomes that of discovering subgraphs that occur frequently throughout the entire set of graphs. Kuramochi and Karypis [17, 18] developed an efficient algorithm for discovering frequent subgraphs and applied it to chemical compound datasets. A similar technique can also be applied to two-dimensional turbulent flow fields. Graphs can be constructed with the nodes representing the spatial location of coherent structures (e.g., vortices) within the turbulent flow field and the edges connecting nodes based on spatial proximity. Subgraphs thus discovered can suggest various forms of spatial interactions among the coherent structures across time. However, such an abstract approach cannot exploit many of the inherent physical and dynamical properties of flow fields.

For MD simulation data, we initially experimented with variants of the molecular substructure discovery algorithms [7, 8, 41]. We adapted the technique by Parthasarathy and Coatney [26], for mining biological macromolecules in protein data, in order to locate defects in silicon bulk lattices. The technique relies on range pruning and candidate pruning for reducing the search space of possible frequent substructures. Since a large part of the lattice that we are considering is bulk, atoms associated with any instantiation of a highly frequent substructure can be safely pruned, and the remaining atoms would constitute the defect [30]. The algorithm uses fuzzy recursive hashing for rapid matching of structures to determine frequency of occurrence. The hashing technique is used in lieu of the geometric hashing method [19, 42], given its lower costs and flexibility. The fuzziness is introduced to handle noise effects in the data [35], especially for lattices at finite temperatures. The approach in general worked reasonably well but required a lot of manual tuning of various parameters [6]. The disadvantage of this approach is its computational complexity and hence the need to search for more efficient alternatives.

Visualization literature is also replete with works on feature extraction and data analysis. Numerous schemes have been devised to automatically locate shocks and other flow field discontinuities. These schemes are typically defined in terms of the numerical gradients of solution variables [22, 23, 25, 38] and are generally effective for locating features that can be adequately described by magnitude changes in scalar fields. Critical points define the structural topology of vector fields and provide a convenient method of analysis [10, 20, 36]. However, fluid dynamists are typically interested in macroscopic features, such as vortices, rather than the location of discrete critical points. Significant progress has been made in the area of identifying vortices, or regions of swirling flow [2, 4, 27, 33, 39].

The consideration of time-varying data introduces an additional complexity through the need for tracking features across multiple time steps of the dataset [29, 37]. According to [34], five distinct evolutionary events can occur to features in scientific simulations: amalgamation, bifurcation, continuation, creation, and dissipation. Each of these events must be accounted for in designing a comprehensive feature tracking algorithm.

3 Feature Mining

Essentially, there is a need to detect features and derive their structures and characteristics from large-scale datasets to better understand the evolutionary phenomena. In addition to feature detection algorithms, aggregation, characterization, and spatio-temporal tracking algorithms must be utilized, in conjunction with traditional data mining techniques, to gain scientific insights. We propose a general framework utilizing the following techniques to address the problems associated with analyzing large-scale datasets generated from CFD and MD simulations:

1. *Event Detection*:
   A "trigger-based" multiscale approach to temporal event detection. Physically derived attributes, or triggers, are monitored for temporal events at multiple time scales. For instance, this attribute can be *swirl* in a CFD simulation or *energy* in an MD simulation.

2. *Feature Mining*:
   A systematic approach to locating, characterizing, and tracking features in unsteady phenomena. The most salient aspects of feature mining include: feature detection and aggregation, shape and attribute characterization, and spatio-temporal tracking.

3. *Interaction Discovery*:
   Mining for important kinematical and dynamical interactions among the features of interest. Our intention is to apply traditional data mining techniques, such as frequent substructure discovery, on the processed features rather than the unprocessed raw data.

We term this integrated approach *generalized feature mining*. In the following subsections, we describe in detail the two feature mining paradigms that comprise the second step of the proposed framework.

3.1 What is a Feature?

Perhaps the most appropriate response to this question is, "It depends on what you're looking for." In general, a feature is a pattern occurring in a dataset that is the manifestation of correlations among various components of the data. For instance, a shock in a supersonic fluid flow would be considered a significant feature. When such a shock occurs, the pressure increases abruptly in the direction of the flow, and the fluid velocity decreases in a prescribed manner. A significant feature also has spatial and temporal scale coherence. In most cases, an adequately resolved feature spans several discrete spatial or temporal increments. One can find similar examples in the MD domain.

For many applications, generic data mining techniques such as clustering, association, and sequencing can reveal statistical correlations among various components of the data. Returning to the shock example, we could use statistical mining to ferret out associations, but it might be difficult to attach precise spatial associations for the rules discovered. A fluid dynamicist, however, would like to locate features with a rather high degree of certainty. Such qualitative assertions alone will not suffice.

This is where our approach to feature mining comes in: we take advantage of the fact that, for simulations of physical phenomena, the field variables satisfy certain physical laws. We can exploit these kinematic and dynamic considerations to locate regions of interest (ROIs). The resulting feature detection algorithms, by their very nature, are highly application-specific. However, the fidelity improvements garnered by tailoring

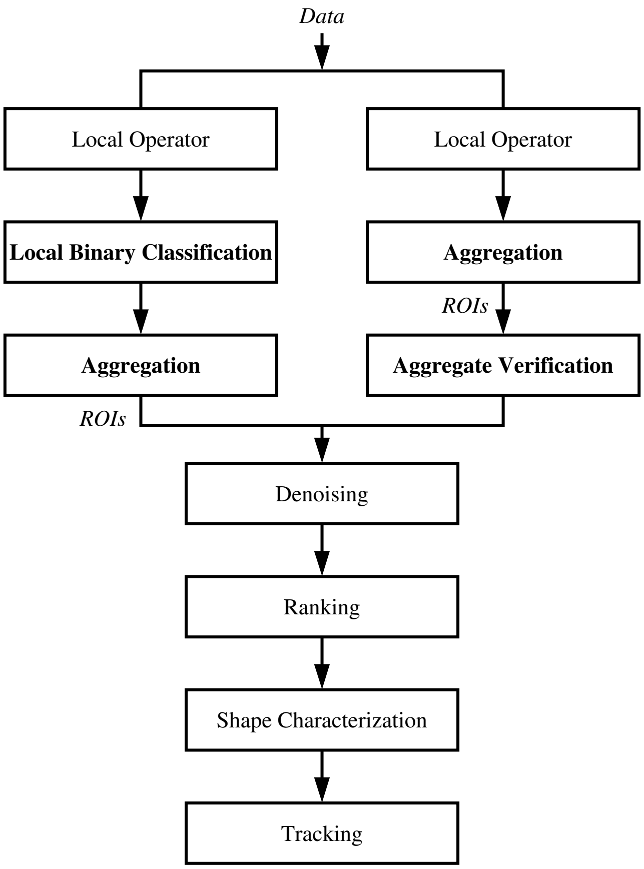


Figure 1: The figure shows the two feature mining paradigms being discussed here. The point-classification paradigm is depicted on the left branch, and the aggregate classification paradigm is depicted on the right branch.

these highly discriminating feature detection algorithms to the particular application far outweigh any loss of generality.

Since physically-based feature detection algorithms tend to be application specific, we defer discussion of these techniques until a later section. We now describe two distinct feature mining paradigms. The common thread is that both are bottom-up feature constructions with underlying physics-based criteria. These two paradigms are used depending on the local and global nature of the features. As will become more evident from the examples, it is unlikely that non-physics-based techniques can provide the fidelity needed to locate complex flow field structures. Below we provide general methodologies that can locate features across a vareity of domains. Figure 1 shows the various stages of the two paradigms.

3.2 Point Classification Paradigm

Many features can be identified using only the local characteristics of the data. These types of features lend themselves to the feature mining paradigm which we term point classification. More specifically, for the point classification paradigm to be appropriate for a given feature, that feature must be amenable to a detection operator and a classification criterion that are both based only on local characteristics of the data. A shock in fluid mechanics simulation is an example of a feature of this type. Similarly, a quenched defect which has attained local equilibria will also qualify for mining with the point classification paradigm. The point classification paradigm requires several operations in the following sequence:

- Local feature detection at every point
- Point-based binary classification (verification)
- Aggregation of similarly classified points
- Denoising to eliminate *weak* features
- Ranking based on saliency of derived attributes
- Shape and dynamical attributes characterization
- Spatial and temporal feature tracking

This approach first identifies individual points as belonging to a feature and then aggregates them to form contiguous regions. Although the points are the entities that are classified, it is the aggregated point sets that are identified as features. The points are obtained from a tour of the discrete domain and may be the vertices of the mesh or sites of atoms used for the simulation. The ROIs can be represented as a list of volumetric elements (for CFD) or a point cloud (for MD).

3.3 Aggregate Classification Paradigm

Identification of other types of features may require information that is less localized than the neighborhood of a point. Further, we may need to classify a set of points based on their collective behavior. We term this second feature mining paradigm aggregate classification. A vortex is an example of a feature in fluid dynamics for which this approach is appropriate. Point classification techniques can also be used to locate vortices; however, without the verification step, false positives can pervade the analysis [40] and invariably cause erroneous conclusions to be drawn or meaningless models to be built.

Aggregate classification follows a somewhat different sequence of operations than point classification:

- Local feature detection at every point
- Aggregation of contiguous candidate points
- Binary classification (verification) of aggregates
- Denoising to eliminate *weak* features
- Ranking based on saliency of derived attributes
- Shape and dynamical attributes characterization
- Spatial and temporal feature tracking

This approach first identifies individual points as being candidate points in a feature using a local operator. Then these points are aggregated to identify regions that potentially contain a feature. The classification algorithm, a physics-based regional criteria, is applied to the aggregate to determine whether the candidate region constitutes a feature. More detail is provided in the next section. Note that the local operator applied here can be very efficient and simultaneously liberal with its criteria, since aggregate verification will eliminate the false positives.

3.4 Denoising, Ranking, and Tracking

The set of detected features, or ROIs, may contain anomalous entries. These entries may have inadequate spatial or temporal resolution. Denoising is often conducted to remove these features that may not be significant. This operation can be as simple as using a threshold for sizes of ROIs or can use a scale-space denoising technique [3, 40]. The ROIs can then be ranked based on their size and an appropriate measure of feature strength (e.g., the density change for a shock or the total energy for a defect). By according ranks, the most significant features can be accessed first. Geometrical shape attributes are then extracted from the features. Note that we can also extend the concept of shape attributes to include other dynamical attributes of the science domain. Tracking ROIs can be conducted on the extracted attributes. This approach provides an efficient solution to the problem of correspondence between features at different time steps. In this paper, we do not discuss denoising, ranking, and tracking in detail.

4 CFD and MD Applications

Feature mining enables us to gain insights about data in disparate application domains in an abstract way. In the subsections that follow, we demonstrate the application of feature mining to CFD and MD simulation data. We also describe our efforts to characterize features, such as vortices and defects, with a sequence of elliptical frusta and tagged point clouds, respectively.

4.1 Paradigm 1: Point Classification

We describe below examples of features which can be mined using the point classification paradigm. The first example is mining shock waves in a CFD dataset. We then describe how a defect can be mined in a silicon bulk lattice, which is quenched and at equilibrium.

4.1.1 CFD Example: Shocks

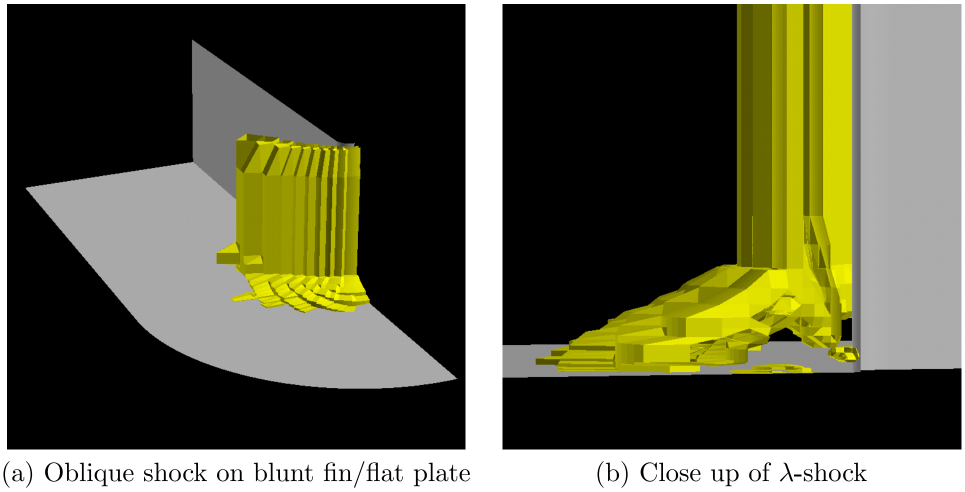
As the first example of the point classification paradigm, we consider shock wave detection in flow fields. A shock is a compression wave that may occur in fluid flows when the velocity of the fluid exceeds the local speed of sound. A shock is characterized by abrupt changes in flow quantities such as pressure, velocity, and density. A physical shock is nearly a singularity–the changes occur over a distance equal to a few mean free paths of the fluid molecules. In the case of numerical data, however, the discontinuity is typically smeared over several cells due to the errors inherent in the discrete approximation. The properties of shocks are explained in more detail in [1]. These properties have to be exploited to develop highly discriminating shock detection algorithms [15, 22, 25].

The key quantity in a shock detection algorithm is the Mach number (ratio of the velocity magnitude to the local sound speed) normal to the shock wave. For a stationary shock, the normal Mach number changes from greater than unity to less than unity in the direction of the flow. Further, near a shock, the pressure gradient is aligned along the local shock normal. Therefore, the scalar product of the normalized pressure gradient $\nabla P/|\nabla P|$ and velocity vector $\vec{V}$ can be used to identify compression as well as the normal Mach number M_n (after division by the local acoustics speed a) as given by

$$M_n = \frac{\vec{V}}{a} \cdot \frac{\nabla P}{|\nabla P|}. \tag{4.1}$$

In regions where M_n changes from greater than unity to less than unity in the direction of the flow, a shock exists. Notice that our operator, the change of M_n in the direction of the flow, is a *local* definition therefore making this approach amenable to the point-classification paradigm. The detected point may then be aggregated to identify the region containing the shock. Of course, care must be taken not to divide by zero in regions where the pressure gradient is zero. Further, a correction term must be computed to account for the temporal variation of the flow field [22].

As an example of our shock detection technique, we show two images for the standard blunt fin/flat plate flow field solution in Figure 2. Figure 2(a) shows a detached oblique shock that wraps around the blunt fin.


(a) Oblique shock on blunt fin/flat plate

(b) Close up of λ-shock

Figure 2: Point classification paradigm shock waves in flow fields.

Figure 2(b) shows the symmetry plane which intersects a λ-shock. It should be noted that the grid used here is relatively coarse which contributes to the rapid dissipation of the shock.

4.1.2 MD Example: Defect Evolution in Silicon

We address the challenge of mining structural defects during an ongoing MD simulation with the application of real-time multiresolution analysis (RTMRA) techniques and the point classification paradigm. Wavelet analysis is exploited in the time domain to analyze the dynamics. For each atom, its sequence of positions are projected on a wavelet basis, with the expansion coefficients generated incrementally, in real-time, using components supplied by the StormRT Scientific Wavelet Package [31]. These components treat streaming data more efficiently than more conventional "fast" wavelet transform (fwt) techniques.

In any persistent structure, "defect" atoms must be distinguished from "bulk" atoms. While this task is more challenging at finite temperature due to the thermal noise, a *single* local operator works perfectly for all quenched (clean) structures and surprisingly well for thermal (noisy) structures. A bulk silicon atom has precisely four neighbors within the distance of 2.6 Å and the angles between any two bonds lie within 90°-130°. Any other atom is a defect. Similar definitions can be formulated for other systems. In a solid, the periodic boundary condition has to be treated with care to obtain the correct bond lengths and distances near the boundary.

Here, we illustrate how the point classification paradigm can be employed toward the defects in the quenched and finite temperatures structures. Each atom site is visited and the atom is tested for membership in a defect ensemble. The classification criterion for this application is as follows. We define two conditions C_1 (number of neighbors as above) and C_2 (bond angle as above) to accurately classify bulk atoms. The *conjunction* of the above two conditions as well as the *disjunction* are evaluated for all atom sites. The atom sites which satisfy the conjunction are the ones which definitely belong to the bulk. Those that satisfy the disjunction would, with some likelihood, belong to the bulk. The remaining atom sites are definitely part of the defect. Such atoms are referred to as *defect atoms.* The defect atoms are then spatially clustered, or aggregated, into possible defect structures. One can generalize this method to include other local conditions, as may be the case for alloy lattices.

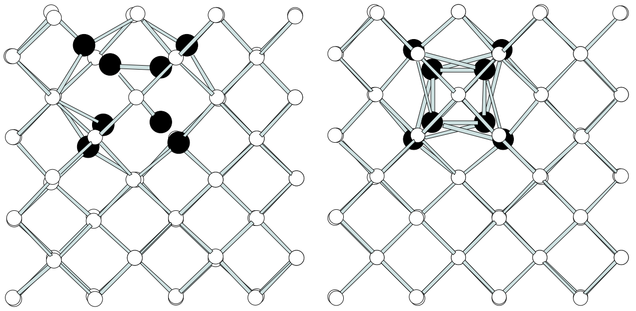


Figure 3: Defect atoms are marked black. Structure (left) is identified at 1000K, which is quenched, using first principles quantum machines, into structure (right). Even though the atoms in the top structure are displaced due to thermal noise, the same atoms are marked as defect in both structures.

Aggregation is crucial to large-scale simulation due to the presence of multiple, spatially disconnected, defect clusters. A line is drawn connecting all defect atoms that lie within 4 Å of each other. Each cluster is then a connected graph, which is computationally inexpensive to obtain given the relatively small number of atoms in a defect. Figure 4 shows two defects embedded in a 512 atom lattice. The different shades represent distinct and seperated spatial clusters.

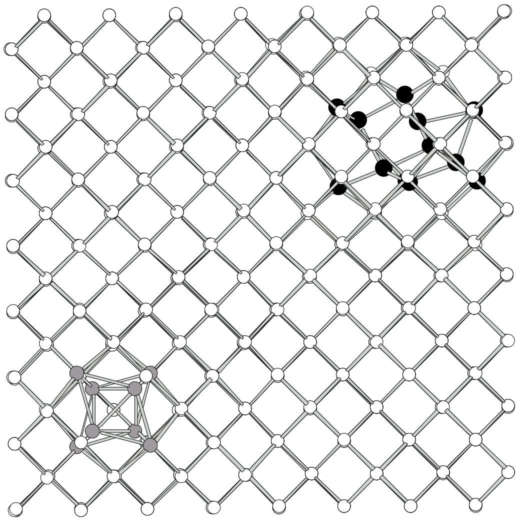


Figure 4: Two spatially separated defect clusters (black and grey) marked by local binary classification, followed by aggregation code.

We have empirically validated that this method works well even on noisy data. Figure 3(left) shows a persistent structure at 1000 K. The atoms marked black are those identified as defect atoms. Figure 3(right) shows the same structure quenched with a first-principles approach. Quenching removes thermal noise at a heavy computational cost. The same atoms are marked in both structures which demonstrates that this method works on noisy structures.

4.2 Paradigm 2: Aggregate Classification

We describe below examples of features which can be mined using the aggregate classification paradigm. The first example is from CFD. We examine how vortices can be automatically detected and verfied by resorting to the underlying physics of the flow field. Similarly, we allude to how the aggregate classification paradigm can be used to mine defects in a silicon bulk lattice at finite temperature.

4.2.1 CFD Example: Vortices

We now consider the application of the aggregate classification paradigm to mining vortices in flow fields. By most accounts [27, 32], a vortex is characterized by the swirling motion of fluid around a central region. This characterization stems from our visual perception of the swirling phenomena that are pervasive throughout the natural world. However, translating that perceptual understanding of a vortex into a formal definition has been quite a challenge. Robinson [32] proposed the following definition for the presence of a vortex:

> *A vortex exists when instantaneous streamlines mapped onto a plane normal to the vortex core exhibit a roughly circular or spiral pattern, when viewed from a reference frame moving with the center of the vortex core.*

We recently developed vortex detection and verification algorithms based on the aggregate classification paradigm [13, 14]. For the local detection operator, a combinatorial labeling scheme is employed to identify all the grid points that belong to vortex cores. What makes this approach so effective at detecting vortex core regions is its close resemblence to Sperner's lemma from Fixed Point Theory in combinatorial topology [12].

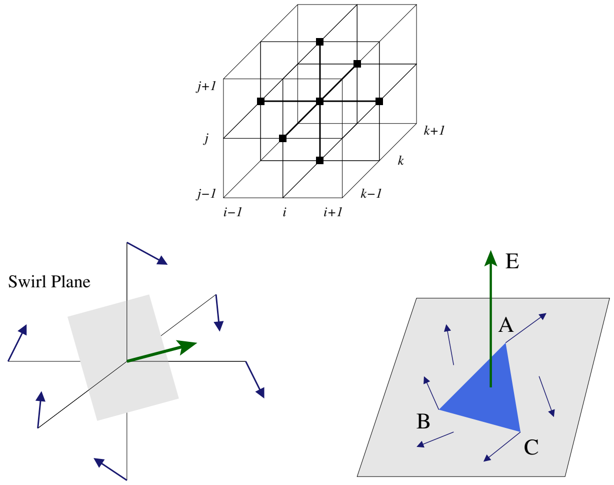
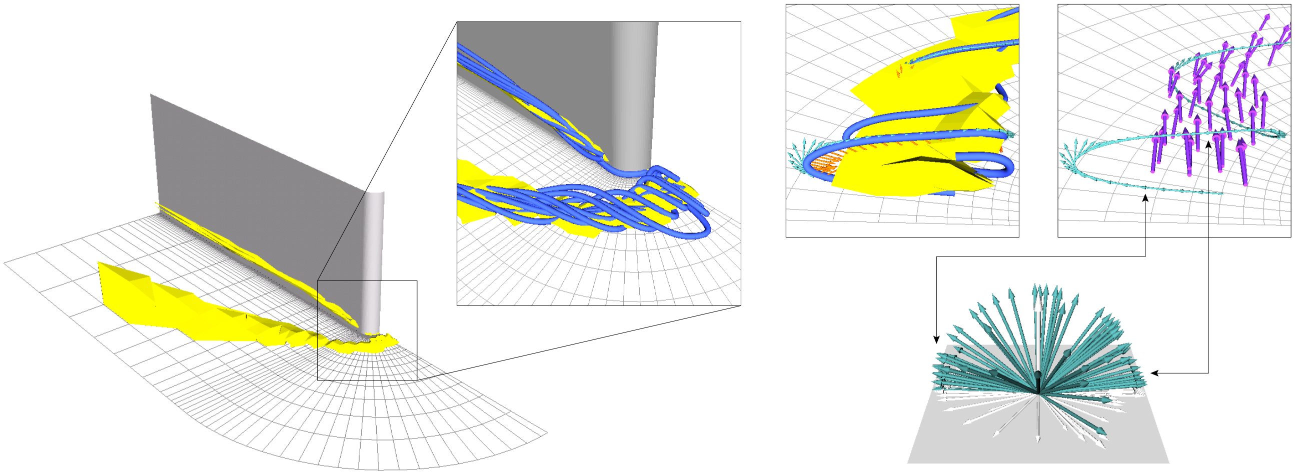


Figure 5: 3D vortex core region detection algorithm.

The connection between vortices and fixed points (i.e., critical points) are well known [39]. Whereas Sperner's lemma labels the vertices of a simplex and identifies the fixed points of a labeled subdivision [12], our detection algorithm labels the velocity vectors of the grid points and identifies grid cells that are most likely to contain critical points. Each velocity vector is labeled according to the direction in which it points. Since velocity vectors around core regions exhibit certain flow patterns that are unique to vortices, it is sufficient to



(a) Detected core regions for blunt fin/flat plate

(b) Illustration of verification algorithm

Figure 6: Aggregate classification paradigm for mining vortices in flow fields. Note the highly elliptic cross section of the vortex core region.

examine the immediate neighborhood of a grid point for the existence of those flow patterns. Figure 5 illustrates the detection algorithm in three dimensions, where it is necessary to approximate the vortex core tangent vector first, and then project the neighboring velocity vectors onto the swirl plane normal to it, before applying the above procedure to the projected velocity vectors. Techniques for approximating the vortex core tangent vector are based on local velocity gradient tensors, such as the vorticity vector or the real eigenvector.

Our approach segments candidate vortex core regions by aggregating points identified from the detection step. We then classify (or verify) these candidate core regions based on the existence of swirling streamlines surrounding them. For features that lack a formal definition, such as the vortex, we must choose the verification criterion so that it concurs with the intuitive understanding of the feature. In this case, verifying whether a candidate core region is an actual vortex core requires checking for swirling streamlines surrounding it. Checking for swirling flow in three dimensions is a non-trivial problem since vortices can bend and twist in various ways. Our verification algorithm measures three-dimensional swirling in terms of the differential geometry properties of the streamline. The technique we developed essentially checks to see if the local tangent to the streamline, when projected onto the swirl plane normal to the candidate core tangent vector, spans a measure of 2π. The aggregate nature of this classification step is apparent. Checking for swirling streamlines is a global (or aggregate) approach to feature classification (or verification) because swirling is measured with respect to the entire core region, not just individual points within the core region.

As a demonstration of our algorithm, we applied the aggregate classification paradigm to the blunt fin/flat plate flow field as shown in Figure 6. In Figure 6(a) we show the detected vortex core regions and the the verifying streamlines near the fin/plate intersection. In Figure 6(b) (top left) we show a close-up of the primary vortex located adjacent to the flat plate. Notice that even though the vortex cross section is highly elliptical, the tangent vectors projected into the local swirling plane, gray vectors in Figure 6(b) (bottom), still satisfy the 2π swirling criterion.

4.2.2 MD Example: Defect Evolution

The large number of identified structures must be sorted into a smaller set of distinct types. Quenching solves this problem but is computationally expensive. In addition, some structures are stable only at high temperatures. By quenching, these structures are lost. Identifying time averaged structures is a great challenge. Occasionally with noisy data, too many or too few atoms are marked. Figure 7 illustrates this situation. These two structures have different numbers of defect atoms marked, yet when quenched are the same. The application of the aggregate classification paradigm for mining MD data is still under development.

4.3 Shape Characterization

Having extracted features either as vortices (CFD) or defects (MD), we describe the next step: characterizing them using shapes attributes. Our objective is to

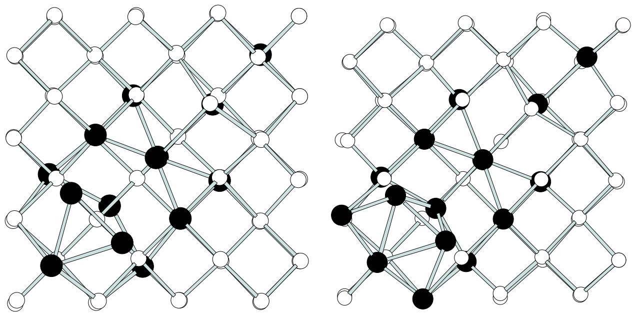


Figure 7: These two structures have a different number of defect atoms marked. When quenched however they are the same structure.

provide a characterization mechanism to facilitate the correspondence of features at different time steps to improve tracking efficiency. Therefore, it is critical that the geometrical and dynamical attributes selected represent salient characteristics of the feature. Although not discussed here, we anticipate that geometrical shape attributes will be an invaluable tool when we endeavor to categorize evolutionary phenomena.

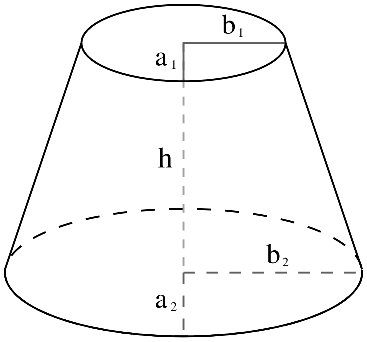


Figure 8: Elliptical frustum

4.3.1 Vortices as Shapes

The swirling region of vortices can be characterized using a sequence of elliptical frusta. An elliptical frustum resembles a conical frustum, except that its two ends are ellipses rather than circles. This is illustrated in Figure 8, where h is the height of the frustum and a_i and b_i are the length of the minor and major axes of the two ellipses. The restriction here is that the axes must be aligned and their length must be proportional, (i.e. $\frac{a_1}{a_2} = \frac{b_1}{b_2}$). The volume can be computed by the following analytical formula:

$$V = \frac{1}{3}h(A_1 + A_2 + \sqrt{A_1 A_2}) \quad (4.2)$$

where $A_i = \pi a_i b_i$ are the areas of the two ellipses.

The shape of an ellipse is usually expressed by its eccentricity, conventionally denoted by e, which is related to a_i and b_i by the formula: $b_i^2 = a_i^2(1 - e_i^2)$. Eccentricity is a positive number less than 1, or 0 in the case of a circle. A higher value corresponds to a more elongated ellipse. This is an important geometrical aspect of swirling regions that can be captured by using elliptical frusta. In Reinders et al. [28], they used conical frusta to "flesh out" the skeleton graph representation of vortices. For modeling the shape of vortical regions, this approach is not ideal.

The shape characterization process starts at the finest level of approximation to the swirling region of the vortex. Starting from the upstream extent of the vortex core, the modeling process seeds a set of swirling streamlines surrounding the core region. The advantage of this paradigm is that the swirling streamlines are already known from the previous verification step. Each elliptical frustum is oriented along the longest segment of the vortex core that does not curve by more than a user-specified amount ϵ^C. The ellipses at each end of the frustum is fitted to the set of streamlines intersecting the swirling plane at each end of the frustum. This adaptive approach to shape modeling requires the minimal number of elliptical frusta to characterize the swirling region.

Once the modeling process is finished at the finest level of approximation, the next level of elliptical frusta are produced by merging every contiguous pair of frusta at the current level. By merging existing elliptical frusta instead of remodeling at the next level, the expensive cost of ellipse fitting the swirling streamlines is amortized [9]. Given two contiguous elliptical frusta at the same level, the merging process preserves their volumes in the new frustum, while averaging the rest of their shape attributes. By averaging the shape attributes of the merged elliptical frusta, the merging process also serves the purpose of a low-pass filter, smoothing the combined shape attributes at higher levels of approximation. In this fashion, a complete and concise hierarchical shape characterization of the swirling region is produced.

We illustrate the adaptive and hierarchical shape characterization process using the blunt fin/flat plate flow field. Figure 9 shows the sequence of elliptical frusta obtained at the finest level in (a) and after one and three levels of merging in (b) and (c), respectively.

4.3.2 Defects as Shapes

We next illustrate how to represent defect structures and distinguish among them. Molecules and their substructures are often described as three-dimensional co-

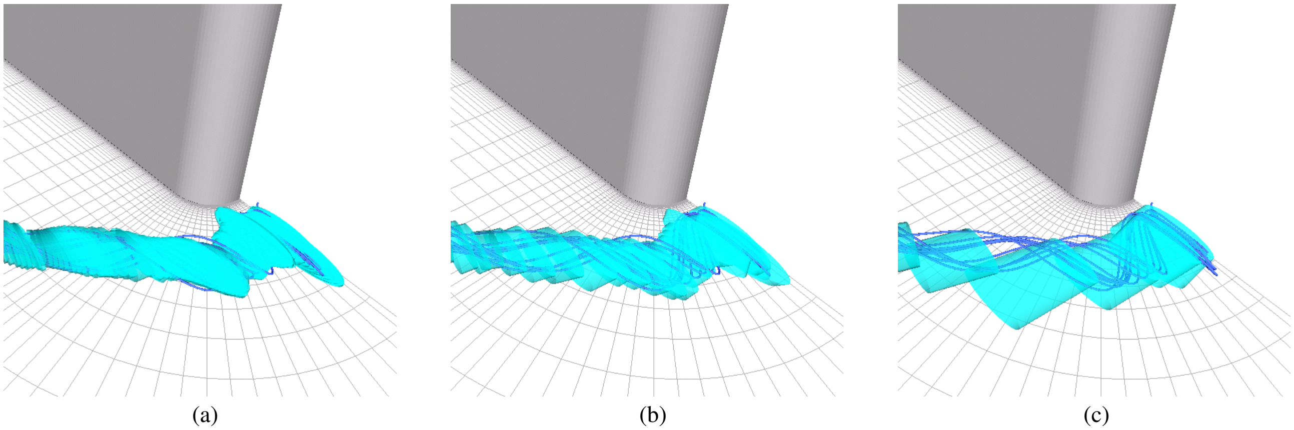

(a)
(b)
(c)

Figure 9: Adaptive and hierarchical shape characterization of the vortices in the blunt fin data set: (a) level 1, (b) level 2, and (c) level 4.

ordinate graphs, where atoms are nodes and chemical bonds are edges. Two substructures are considered equal if, after an arbitrary number of spatial translations (and/or rotations) on one substructure, both substructures are described by the same graph.

Defects are represented simply by atom locations, (i.e., their (x, y, z) coordinates). In order to match defect structures, one approach is to store complete three-dimensional information between all pairs of atoms (mining bonds) belonging to a defect. The mining bond $M(A_iA_j)$ is a 3-tuple of the form

$$M(A_iA_j) = \{A_i type, A_j type, distance(A_iA_j)\}$$

A k-atoms et X, which is a substructure containing k connected atoms, is then defined as a tuple of the form

$$X = \{\mathbf{S_X}, A_1, A_2, \ldots, A_k\},$$

where A_i is the i^{th} atom and $\mathbf{S_X}$ is the set of mining bonds describing the atomset. By defining atom pair combinations with mining bonds, the 3D graph is completely represented in a form, such that two atom sets X and Y are considered to be the same chemical substructure if $\mathbf{S_X} = \mathbf{S_Y}$. To compare two defects we simply need to compare the corresponding X tuples of the corresponding atomsets represented by the two defects. The problem with this naive approach is that it is not very robust to noise (missing and/or perturbed defect atoms) and the memory requirements are $O(k^2)$ where k represents the number of atoms in the defect structure.

As an alternative to the above approach, we decompose each defect structure into k spherical regions, each one centered around every atom belonging to the defect. The substructures formed by all those atoms within the ROI are represented as using mining bonds. The key twist is that substructures of interest are matched using recursive fuzzy hashing, a technique that allows us to handle noise-effects in the data [26]. The memory utilization is $O(km)$ where $m < k$.

This set of “interesting” substructures is used to generate a substructural fingerprint for a given defect. A substructure fingerprint for a particular defect is a vector representation of the set of “interesting” substructures; elements representing substructures contained in that defect are marked, either with a 1 (present) or a 0 (absent) [26]. Defect disambiguation can then be conducted by comparing the corresponding fingerprint vectors. It turns out that this simple approach is robust to noise as well as efficient to compute. Moreover, as we can see from Table 10, the disambiguation is near perfect (non-diagonal elements are 0, diagonal elements are 1) for a set of 14 defect types across multiple simulation runs (0 indicating perfect dissimilarity and 1 representing high similarity).

This approach clearly lends itself to identifying discriminating motifs, (i.e., substructures that can disambiguate between different defects). Moreover these motifs are likely to be much smaller than the defect structures enabling one to develop more efficient defect classifiers, based on the presence or absence of these discriminatory motifs *d-motifs*. Essentially, instead of generating all the motifs for a given defect and then constructing a fingerprint, we would only need to check for the presence or absence of these d-motifs. The robustness of this strategy is currently being evaluated.

5 Summary

The steady increase in computing power perennially challenges our ability to learn new science from the massive amount of data that are being generated. Our science applications, computational fluid dynamics and

Defect	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	1.0	0.08	0.04	0.04	0.04	0.08	0.0	0.0	0.0	0.0	0.0	0.07	0.09
3	0.0	0.08	1.0	0.04	0.04	0.04	0.04	.0.0	0.0	0.0	0.08	0.04	0.0	0.09
4	0.0	0.04	0.04	1.0	0.1	0.11	0.0	0.0	0.0	0.04	0.0	0.0	0.04	0.05
5	0.0	0.04	0.04	0.1	1.0	0.15	0.04	0.05	0.04	0.03	0.04	0.04	0.07	0.04
6	0.0	0.04	0.04	0.11	0.15	1.0	0.0	0.05	0.05	0.14	0.09	0.0	0.04	0.05
7	0.0	0.08	0.04	0.0	0.04	0.0	1.0	0.0	0.0	0.0	0.0	0.04	0.03	0.04
8	0.0	0.0	0.0	0.0	0.05	0.05	0.0	1.0	0.13	0.0	0.05	0.0	0.04	0.0
9	0.0	0.0	0.0	0.0	0.04	0.05	0.0	0.13	1.0	0.0	0.04	0.0	0.04	0.0
10	0.0	0.0	0.0	0.04	0.03	0.14	0.0	0.0	0.0	1.0	0.12	0.0	0.0	0.0
11	0.0	0.0	0.08	0.0	0.04	0.09	0.0	0.05	0.04	0.12	1.0	0.0	0.03	0.0
12	0.0	0.0	0.04	0.0	0.04	0.0	0.04	0.0	0.0	0.0	0.0	1.0	0.03	0.0
13	0.0	0.07	0.0	0.04	0.07	0.04	0.03	0.04	0.04	0.0	0.03	0.03	1.0	0.0
14	0.0	0.09	0.09	0.05	0.04	0.05	0.04	0.0	0.0	0.0	0.0	0.0	0.0	1.0

Figure 10: Familiarity based on Sub structural Fingerprinting

molecular dynamics, raise central problems that we plan to address using a common framework which we have christened *generalized feature mining*. Components of this approach include temporal event detection, local and global feature mining paradigms, and kinematical and dynamical interactions discovery.

Our science applications are very distinct and attempt to characterize very diverse phenomena. However, they both have commonalities that are exploited by the above set of techniques. We reported some ongoing research in designing feature mining paradigms for large-scale simulation datasets. The results obtained have been very encouraging. A systematic approach to feature mining was conceived to locate both local and global features. However, much more remains to be done to realize the complete unified framework mentioned above. Our next main task will be tracking the characterized features by exploiting the similarities in the extracted geometrical and dynamical attributes to address the correspondence problem between features at different time steps. Finally, we plan to find associations between features and events to gain further scientific insights.

References

[1] J. D. Anderson. *Fundamentals of Aerodynamics.* McGraw-Hill Book Company, 1984.

[2] D. C. Banks and B. A. Singer. A Predictor-Corrector Technique for Visualizing Unsteady Flow. *IEEE Trans. on Visualization and Computer Graphics*, 1(2):151–163, June 1995.

[3] D. Bauer and R. Peikert. Vortex Tracking in Scale-Space. In *Joint Eurographics–IEEE TCVG Symposium on Visualization*, pages 140–147, May 2002.

[4] C. H. Berdahl and D. S. Thompson. Eduction of Swirling Structure using the Velocity Gradient Tensor. *AIAA J.*, 31(1):97–103, January 1993.

[5] M. C. Burl, L. Asker, P. Smyth, U. M. Fayyad, P. Perona, L. Crumpler, and J. Aubele. Learning to Recognize Volcanoes on Venus. *Machine Learning*, 30(2-3):165–194, 1998.

[6] M. Coatney, S. Mehta, T.-S. Choy, S. Barr, S. Parthasarathy, R. Machiraju, and J. W. Wilkins. Defect Detection in Silicon and Alloys. In *IEEE Workshop on Visualization in Bioinformatics and Cheminformatics*, October 2002.

[7] L. Dehaspe, H. Toivonen, and R. D. King. Finding Frequent Substructures in Chemical Compounds. In *4th ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining*, pages 30–36, August 1998.

[8] S. Djoko, D. Cook, and L. Holder. Analyzing the Benefits of Domain Knowledge in Substructure Discovery. In *1st ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining*, pages 75–80, August 1995.

[9] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct Least Square Fitting of Ellipses. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(5):476–480, May 1999.

[10] A. Globus, C. Levit, and T. Lansinski. A Tool for Visualizing the Topology of Three-Dimensional Vector Fields. In *IEEE Visualization '91*, pages 33–39, October 1991.

[11] E.-H. Han, G. Karypis, and V. Kumar. Data Mining for Turbulent Flows. In R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, and R. R. Namburu, editors, *Data Mining for Scientific and Engineering Applications*, pages 239–256, 2001.

[12] M. Henle. *A Combinatorial Introduction to Topology.* Dover, 1979.

[13] M. Jiang, R. Machiraju, and D. Thompson. A Novel Approach to Vortex Core Region Detection. In *Joint Eurographics–IEEE TCVG Symposium on Visualization*, pages 217–225, May 2002.

[14] M. Jiang, R. Machiraju, and D. Thompson. Geometric Verification of Swirling Features in Flow Fields. In *IEEE Visualization '02*, pages 307–314, October 2002.

[15] V. Juvvigunta and R. Machiraju. Shock Detection and Analysis Using Wavelet Based Techniques. In *7th Intl. Conference on Numerical Grid Generation in Computational Field Simulations*, pages 559–568, September 2000.

[16] C. Kamath, E. Cantu-Paz, I. K. Fodor, and N. A. Tang. Classifying Bent-Double Galaxies. *IEEE Computing in Science & Engineering*, 4(3):52–60, 2002.

[17] M. Kuramochi and G. Karypis. Frequent Subgraph Discovery. In *IEEE Intl. Conference on Data Mining '01*, pages 313–320, December 2001.

[18] M. Kuramochi and G. Karypis. Discovering Frequent Geometric Subgraphs. In *IEEE Intl. Conference on Data Mining '02*, December 2002.

[19] Y. Lamdan and H. Wolfson. Geometric Hashing: A General and Efficient Model Based Recognition Scheme. In *Intl. Conference on Computer Vision*, pages 238–249, July 1988.

[20] Y. Lavin, R. Batra, and L. Hesselink. Feature Comparisons of Vector Fields using Earth Movers' Distance. In *IEEE Visualization '98*, pages 103–109, October 1998.

[21] H. Li and S. Parthasarathy. Automatically Deriving Multi-Level Protein Structures Through Data Mining. In *HiPC Workshop on Bioinformatics and Computational Biology*, December 2001.

[22] D. Lovely and R. Haimes. Shock Detection from Computational Fluid Dynamics Results. In *AIAA 14th Computational Fluid Dynamics Conference, Paper 99-3285*, June 1999.

[23] K.-L. Ma, J. van Rosendale, and W. Vermeer. 3D Shock Wave Visualization on Unstructured Grids. In *1996 Volume Visualization Symposium*, pages 87–94, October 1996.

[24] R. Machiraju, J. E. Fowler, D. S. Thompson, B. K. Soni, and W. Schroeder. EVITA–Efficient Visualization and Interrogation of Tera-Scale Data. In R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, and R. R. Namburu, editors, *Data Mining for Scientific and Engineering Applications*, pages 257–279, 2001.

[25] D. L. Marcum and K. P. Gaither. Solution Adaptive Unstructured Grid Generation Using Pseudo-Pattern Recognition Techniques. In *AIAA 13th Computational Fluid Dynamics Conference, Paper 97-1860*, June 1997.

[26] S. Parthasarathy and M.Coatney. Efficient Discovery of Common Substructures in Macromolecules. In *IEEE Intl. Conference on Data Mining '02*, December 2002.

[27] L. M. Portela. *Identification and Characterization of Vortices in the Turbulent Boundary Layer*. PhD thesis, Stanford University, 1997.

[28] F. Reinders, M. E. D. Jacobson, and F. H. Post. Skeleton Graph Generation for Feature Shape Description. In *Joint Eurographics–IEEE TCVG Symposium on Visualization*, pages 73–82, May 2000.

[29] F. Reinders, F.H. Post, and H.J.W. Spoelder. Attribute-Based Feature Tracking. In *Joint Eurographics–IEEE TCVG Symposium on Visualization*, pages 63–72, May 1999.

[30] D. A. Richie, J. Kim, R. Hennig, K. Hazzard, S. Barr, and J. W. Wilkins. Large-Scale Molecular Dynamics Simulations of Interstitial Defect Diffusion in Silcon. In *Material Research Society Symposium*, volume 731, pages W9.10.1–5, 2002.

[31] D. A. Richie, J. Kim, and J. W. Wilkins. Applications of Real-Time Multiresolution Analysis for Molecular Dynamics Simulations of Infrequent Events. In *Material Research Society Symposium*, volume 677, pages AA5.1.1–7, 2001.

[32] S. K. Robinson. Coherent Motions in the Turbulent Boundary Layer. *Ann. Rev. Fluid Mechanics*, 23:601–639, 1991.

[33] M. Roth. *Automatic Extraction of Vortex Core Lines and Other Line-Type Features for Scientific Visualization*. PhD thesis, Swiss Federal Institute of Technology, 2000.

[34] R. Samtaney, D. Silver, N. Zabusky, and J. Cao. Visualizing Features and Tracking Their Evolution. *IEEE Computer*, 27(7):20–27, July 1994.

[35] K. B. Sarachik. Limitations of Geometric Hashing in the Presence of Gaussian Noise. Technical Report AIM-1395, MIT, 1992.

[36] G. Scheuermann. *Topological Vector Field Visualization with Clifford Algebra*. PhD thesis, University of Kaiserslautern, 2000.

[37] D. Silver and X. Wang. Tracking and Visualizing Turbulent 3D Features. *IEEE Trans. on Visualization and Computer Graphics*, 3(2):129–141, June 1997.

[38] B. K. Soni, R. Koomullil, D. S. Thompson, and H. Thornburg. Solution Adaptive Grid Strategies Based on Point Redistribution. *Computer Methods in Applied Mechanics and Engineering*, 189:1183–1204, 2000.

[39] D. Sujudi and R. Haimes. Identification of Swirling Flow in 3D Vector Fields. In *AIAA 12th Computational Fluid Dynamics Conference, Paper 95-1715*, June 1995.

[40] D. S. Thompson, R. Machiraju, M. Jiang, J. Nair, G. Craciun, and S. Venkata. Physics-Based Feature Mining for Large Data Exploration. *IEEE Computing in Science & Engineering*, 4(4):22–30, July 2002.

[41] X. Wang, J. T.-L. Wang, D. Shasha, B. A. Shapiro, S. Dikshitulu, I. Rigoutsos, and K. Zhang. Automated Discovery of Active Motifs in Three Dimensional Molecules. In *3rd ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining*, pages 89–95, August 1997.

[42] H. Wolfson and I. Rigoutsos. Geometric Hashing: An Overview. *IEEE Computational Science and Engineering*, 4(4):10–21, 1997.

[43] K. Yip and F. Zhao. Spatial Aggregation: Theory and Applications. *J. of Artificial Intelligence Research*, 5:1–26, August 1996.