Visual Trends Analysis in Time-Varying Ensembles

Harald Obermaier, *Member, IEEE*, Kevin Bensema, *Member, IEEE*, and Kenneth I. Joy, *Member, IEEE*

Abstract—Visualization and analysis techniques play a key role in the discovery of relevant features in ensemble data. Trends, in the form of persisting commonalities or differences in time-varying ensemble datasets, constitute one of the most expressive feature types in ensemble analysis. We develop a flow-graph representation as the core of a system designed for the visual analysis of trends in time-varying ensembles. In our interactive analysis framework, this graph is linked to a representation of ensemble parameter-space and the ensemble itself. This facilitates a detailed examination of trends and their correlations to properties of input-space. We demonstrate the utility of the proposed trends analysis framework in several benchmark data sets, highlighting its capability to support goal-driven design of time-varying simulations.

Index Terms—Ensemble visualization, trend visualization, data analysis

1 Introduction

Ensemble data is being created in an increasing number of fields in science and engineering. The main reason for this trend is the growing complexity of simulation and modeling processes in applications such as weather forecasting or tool design. In such complex scenarios the "correct" choice of parameters for a given process is unknown to designers and scientists. Fortunately, advancements in parallel computing have opened novel ways to solving this problem. Instead of performing parameter-optimization by iterating through a set of prototype solutions in sequence, several candidate solutions with a range of different parameter settings can now be computed in parallel. The outcomes of these simulation runs, the *ensemble*, are subsequently compared and analyzed to enable a goal-driven parameter-optimization. The goal of ensemble visualization is to provide support for these data analysis and parameter-optimization tasks. Herein, two of the primary challenges of ensemble visualization are the robust identification of trends or outliers and the integration of parameter-space exploration techniques [30] (Section 2). Without appropriate solutions for these challenges, analysis and design of simulations becomes a form of black-box optimization. We propose to solve these problems through the combined application of multiple techniques from scientific and information visualization in the form of an ensemble analysis system. Specifically, we develop an alluvial flow graph for the analysis of time-varying trends (Section 3) behavior. This view of the data is extended by an interactive parallel-coordinates (PC) graph for parameter-space visualization and a rendering of the ensemble itself. The combination of these visualization components creates a system (Section 4) capable of providing in-depth support for trends analysis in time-varying ensembles. In summary, the work presented in this paper makes the following contributions:

- A system for visual trends analysis in time-varying ensembles.
- Visualization of statistical ensemble consistency and variability.
- Full integration of parameter-space into time-varying ensemble analysis.

We use the proposed system to analyze several application data sets (Section 5) and demonstrate its usefulness for visual trends analysis and goal-driven system design.

- *The authors are with the Department of Computer Science, University of California, Davis, One Shields Avenue, Davis, CA 95616. E-mail: {hobermaier, kbensema, kijoy}@ucdavis.edu.*

Manuscript received 26 Sept. 2014; revised 19 Nov. 2015; accepted 1 Dec. 2015. Date of publication 10 Dec. 2015; date of current version 7 Sept. 2016. Recommended for acceptance by S. Miksch.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2015.2507592

2 Related Work

The techniques proposed in this paper are related to a number of fields. In the following we summarize relevant related work from these four areas.

2.1 Uncertainty Visualization

The statistical quality of ensemble data makes work in this area closely related to uncertainty visualization. Data with numerical errors or statistical and stochastic descriptions of fields may be interpreted as data with uncertainties [9], [11], [19], [33] and visualized accordingly, for example through the use of pseudo-coloring, or glyphs and geometry generation. Frequently, uncertain data arises in applications and scientific simulations that produce output in the form of scalar, vector, and tensor-fields.

In 2D vector fields vector glyphs [51] may be adapted to represent uncertain information, such as variability in direction and magnitude. In a related approach, Hlawatsch et al. [15] focus on the comparison of flow behaviors in time-varying vector fields, employing radar-style flow glyphs, which can be seen as a single-run spatial trends

analysis. For dense representations, texture-based visualization techniques of uncertainties have proven to be especially useful [4]. In combination with overlays, these techniques can be employed to compare flow and scalar fields [45].

More indirectly, it is possible to use divergence-based measures to represent flow variability in uncertain vector fields. In this context, Schneider et al. [40] propose the use of PCA as a measure similar to the Finite Time Lyapunov Exponent (FTLE) [13] to compute variance based flow divergence. Other measures of spatial distribution may be used to examine uncertainty inherent in local features, such as critical points, or cores of swirling motion, Petz et al. [34]. Our work employs some of these techniques, such as pseudo-coloring and estimations of vector field variability to convey summary statistics of an ensemble.

2.2 Ensemble Visualization

Uncertain data visualization is related to ensemble visualization in the sense that ensemble data is often aggregated into distributions, which allows for the direct application of uncertainty visualization techniques. Such a direct aggregation, however, is often not suitable for ensemble analysis, since it neglects the benefit of being able to identify individual ensemble members. Being able to identify contributions of individual runs while at the same time providing summary statistics is one of the key capabilities of ensemble visualization. Sanyal et al. [39] approach the visualization of ensembles of numerical weather simulations by extracting sets of isocontour lines (so-called spaghetti plots) and designing glyphs that illustrate local variances. In three dimensions, slicing can help create an impression of differences in isosurfaces, as demonstrated by Alabi et al. [1]. Recently, Whitaker et al. [48] have devised methods that allow for the visualization of contours in the form of contour box-plots, encoding statistical and uncertainty quantities into spaghetti-plot type visualizations.

Means and variances of scalar quantities in climate simulations have been employed by Potter et al. [36], [37] and Wilson and Potter [50]. Notions of variances can also be employed to detect agreement and disagreement in ensembles for arbitrary flow simulations [16]. Trajectories may alternatively be grouped directly, as presented by Guo et al. [12]. Similarly, vector fields may also be compared based on their topological skeletons [31], [32] or trajectories [46]. In simulation ensembles, where clear trends in qualitative outcome are detectable, an ensemble clustering step can be beneficial before performing statistical data averaging, as demonstrated by Smith et al. [42].

Ensemble visualizations can be made especially expressive if (some) ground truth data is available. This allows for an estimation of predictive uncertainty of the ensemble and can aid in identifying outliers in the set of ensemble members [10]. Gosink et al. also propose a visualization of parameter sensitivity in a static setting, which is a key component of efficient parameter space analysis. The overwhelming majority of these ensemble visualization techniques relies on comparative visual analysis of simulation outcomes [35]. However, from an analysis point-of-view it is also of immense importance to relating these outcomes to parameters of individual runs. Only then one is able to perform targeted simulation analysis and parameter optimization. We bridge this gap and unify methods from ensemble visualization with techniques from uncertainty visualization and computational steering.

2.3 Computational Steering

In computational steering, a user is allowed and encouraged to alter parameters of a numerical simulation to steer it towards a desired result. Often, computational steering involves the use of multiple (concurrent) simulation runs, presenting the user with a set of viable alternative options. An ensemble of simulation results generated by a thorough sampling of the parameter space provides a post-hoc analogue to this kind of computational steering.

Visualization and computational steering have been closely interconnected since the 1990s. Mulder et al. [27] give an overview of computation steering environments while Johnson et al. [18] propose to couple interactive visualization and simulation closely in order to step away from scenarios where visualization has a pure post-processing role. Computational steering has especially made an impact in tool design where the contribution of visualization techniques to the steering process becomes evident immediately [7], [26].

In this work, an analogue to the steering process may be performed in output space and as such represents an implicit exploration of parameter space. As demonstrated by Waser et al. [47] alternative scenarios, as given by samples in parameter-space, may also be presented explicitly. We exploit computational steering concepts in the context of ensemble analysis by creating close links between parameter- and output-space. This enhances the ensemble analysis task with capabilities for output-guided parameter optimization, as proposed by Coffey et al. [7], by presenting quantitative information from parameter-space.

2.4 Parameter-Space Exploration

Parameter-space exploration is a ubiquitous task in numerical computing. Especially design and optimization processesrequire the careful selection of (initial) parameters. Such optimal parameters are often the result of a long and tedious exploration phase. As an example, one can use parameter-exploration to optimize parameters in light bulb design [44], or to improve and tune an electronic unit injector [24]. Visualization is related to parameter-space exploration in two ways. First, it may be used to expedite and enhance the exploration task [20], and second, it may be the target of the parameter exploration process. Since complex visualization algorithms are often sensitive to parameter-choice, i.e., changes in parameters may result in strongly dissimilar visual output, researchers have explored how parameter-space analysis may be used to improve the result of visualization techniques. Ma [23] and Jankun-Kelly and Ma [17], for example, use parameter space exploration to find optimal visualization results and create new visualizations by combining desirable properties of known outcomes. Sedlmair et al. [41] present a conceptual framework for parameter space exploration and evaluate some previous works in light of that framework.

The connection to ensemble analysis becomes clear, as soon as parameter presets are used during the exploration process. Wijk and Overveld [49] employ such presets to

find preferred parameters or create desired variations. Parameter-space exploration may also be driven from the other side of an ensemble – by inspecting or modifying outcomes of a simulation. Bruckner and Möller [5] use visualization and exploration techniques to design visual effects. Such gallery-based parameter exploration tends to neglect the importance of input-output relationships, a key element in efficient parameter-space exploration. Other work shifts the focus towards explicit parameter-space exploration. Berger and Piringer [2] search for pareto-optimal solutions in parameter-space. Pretorius et al. [38] use parameter exploration for image analysis and hint at the use of machine learning processes as advanced input-output correlation techniques. Obermaier et al. [29] visualize the parameter space of a clustering algorithm, allowing for guided parameter selection in a simple 2D space. Bergner et al. [3] create a meaningful input-output relation by clustering parameter-space based on outcomes.

We provide a framework for the correlation of ensemble input- and output space. This work contributes to the effectiveness of parameter-space exploration, by coupling interactive parameter selection to feedback in the form of trends visualization in ensemble output space. Through the use of Neural Networks (NN) we achieve a basic correlation between trends in output and clusters in input space. It is notable that machine learning concepts are increasingly used in the context of parameter optimization and simulation analysis [6], [21], [25] including applications in visualization [22], such as volume rendering [8].

3 Trends in Time-Varying Ensembles

An *ensemble* E is a collection of $k \in \mathbb{N}$ functions $f_i : \mathbb{R}^n \rightarrow R^m$. Discrete representations of these ensemble members, f_i, are obtained as the outcomes of measurements or simulation runs. In the context of this paper, all f_i share a common domain and represent time-varying 2D or 3D data, i.e., $\mathbb{R}^n = \mathbb{R}^{\{2,3\}} \times \mathbb{R}$. Furthermore, we assume that each f_i is uniquely associated with a set of (simulation) parameters $P_i = \{p_i^j\} \in P$ with P being the shared *ensemble parameter space*.

3.1 Definition of Trends

A feature definition $\mathcal{F}$, that is applicable to individual ensemble members f_i, results in a set F_t of concrete realizations $F_{i,t}$ for each time step if applied to all members of an ensemble at a location x. Some examples of feature realizations would be the location of a particle at time t seeded at x at t_0 and advected through a flow field or a computed scalar value at x such as temperature or pressure. We restrict feature definitions to p-dimensional points. Clustering these points results in a partition $\mathcal{P}_t = \{C_1 \dots C_w\}$ on the ensemble members at time t and location x. Notationally, outliers are considered single-element clusters in the partition. We define a trend $T = (C_1 \in \mathcal{P}_1, C_2 \in P_2 \dots C_n \in \mathcal{P}_n)$ at a location x to be a sequence of clusters, one from each partition in the time interval under consideration, where temporally adjacent clusters in the sequence share at least one ensemble member. Trends are coherent or meaningful if the partition clusters in the sequence contain a large enough core of the same ensemble members throughout a long enough portion of the ensemble's temporal extent to be useful to the end user of the analysis.

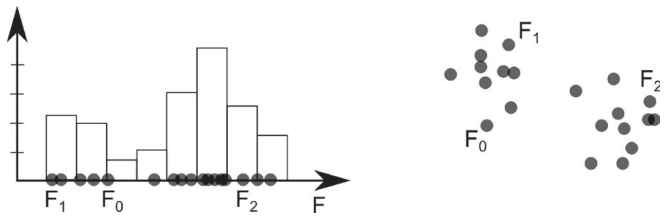

Fig. 1. Examples of F in 1D and 2D. Individual realizations F_i are points in p-dimensional space.

3.2 Extraction of Clusters

The restriction of feature definition realizations to p-dimensional points allows the extraction of clusters in F_t by applying a standard multi-dimensional clustering algorithm to the p-dimensional point-set. Examples of clusters are illustrated in Fig. 1. We extract clusters by performing a nearest-neighbor-based density estimate on this point-set. Our approach follows the minimum-tree based implementation of runt pruning as proposed by Stuetzle [43], and we chose this particular algorithm because it does not require a priori the number of clusters and because of its ease of implementation.

First, the complete graph K on F is computed. Each edge $e = (F_i, F_j)$ is weighted according to the Euclidean distance between F_i and F_j, i.e., $w(e) = \|F_i - F_j\|$. A minimum spanning tree T is then computed on this graph. For each edge $e_i = (F_i, F_j) \in T$, we then compute the *runt score* for e_i as follows: Let $\mathbb{F}_{e_i}$ be the forest generated by removing e_i and any edge e_j such that $w(e_j) > w(e_i)$. Let T_i and T_j be trees in $\mathbb{F}$ rooted at F_i and F_j. We then calculated the number of nodes in T_i and T_j, and choose the lesser value r_i to be the runt score of the edge e_i.

Once the runt score is recorded for each edge, we choose the runt threshold r. Similar to the heuristic given by Stuetzle [43] we choose this r to be the largest runt score that follows a discontinuity in runt score distributions.

Given the runt threshold, we can compute the edge break weight $b = \min_{e_i \in T, r_i < r} w(e_i)$. We remove all edges from T that have an edge weight exceeding the break weight to form the cluster forest F_c. The clusters of F consist of the connected components of F_c.

3.3 Properties of Trends and Clusters

The identification of trends in the form of common or special behaviors present in simulation output is a key analysis technique in a variety of domains. In ensemble analysis, however, the significance and meaning of extracted trends is influenced by a number of characteristic properties.

- *Parameter-space correspondence:* The clusters of feature descriptors at a given timestep and location a clustering of ensembles in E. Due to the assumption of a shared parameter-space amongst members of an ensemble, this translates to a clustering in ensemble parameter-space P. If the induced clusters in P are well separable, the obtained clustering in F corresponds to a clustering in P. A low correspondence between clusters in F and separability in P can

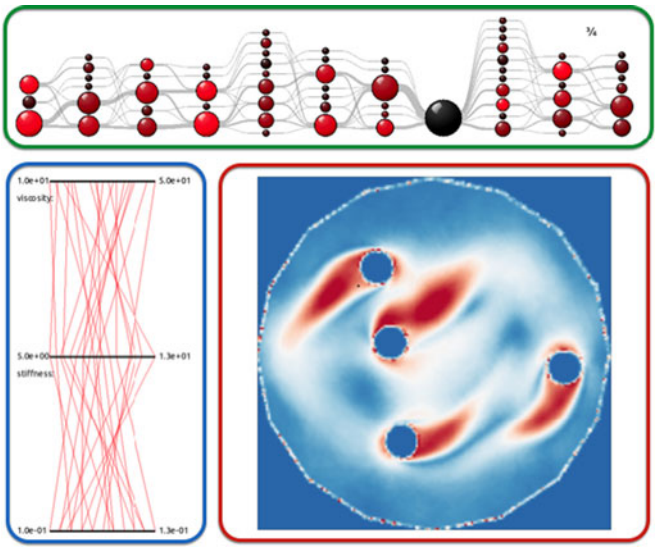

Fig. 2. Overview of the trends analysis framework. The three main components serve distinct purposes and are linked semantically and through user interaction. The rendering of the ensemble introduces the main color map used throughout this paper - a mapping, in this case of temperature, ranging from blue (min) over white to red (max). Colors of trend graph nodes follow a brightness-based color-map with colors ranging from black (min) to bright red (max).

indicate events of interest, including numerical artifacts, physical phenomena, or random chance (Section 4.3.4).

- *Consistency:* In time-varying ensembles, groups of ensemble members may form clusters that largely persist over multiple time-steps. A trend T is *perfectly consistent* over an interval of time I, if its clusters consist of the same ensemble members for all $t_j \in I$, i.e., $C_T(t_i) = C_T(t_j)$ for all $t_i, t_j \in I$. There are multiple degrees of non-consistency over time, all of which facilitate important insights into the behavior of an ensemble.

We believe these two properties to play a central role in the efficient analysis of time-varying ensembles. To support visual analysis of trends we therefore develop an integrated visualization system that enables in-depth study of trend consistency and parameter correlation.

4 Visual Trends Analysis

Our system builds upon the given definitions and enables the investigation of trends and their relationship to parameter-space. The system consists of several linked components. In the following we detail the three main data views of our trends analysis framework and present interaction and linking capabilities.

4.1 System Overview

Our system (shown in Fig. 2) consists of three main components, that are closely linked and give complementary views of 2D and 3D time-varying ensemble data:

1) *A trend graph:* An interactive graph representing time-varying trends behavior. This component facilitates trend consistency analysis and provides a view at the clustering behavior of features over time. In Fig. 2, the trend graph is in the green box at the top of the figure
2) *A visualization of parameter-space:* An interactive rendering of parameter-space that adapts to trends/clusters present in the data. This view provides means for parameter-space correspondence analysis, parameter optimization, and interactive ensemble filtering. The parameter-space view may be found on the lower left side of Fig. 2, outlined in blue.
3) *An ensemble view:* A summary view of the ensemble data, providing a rendering of the domain along with summary statistics and data classification. The large window outlined in red in the lower right of Fig. 2 is the ensemble view.

The concept and implementation of these components is described in detail in the following sections.

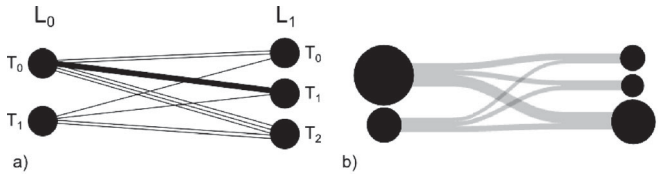

Fig. 3. Visualizations of a trend graph. Two time-steps are shown. The first level (set of realizations) has two nodes (clusters), the second level has three. A total of 10 edges (realizations/simulation runs) are present in the graph (ensemble). (a) A rendering of a trend graph with two levels. The feature realization F_i (of run f_i) marked in bold moves from trend T_0 in level L_0 to trend T_1 in level L_1. Note that relative ordering of trends within a level has no meaning and is determined through layout rules. (b) Edge bundling and weighing creates a more comprehensive alluvial diagram.

4.2 The Trend Graph

We represent the partitions of feature realizations $P(t_j) = \{C_k(t_j)\}$ in consecutive time-steps t_j in the form of an alluvial diagram. Alluvial diagrams are a specialization of classic flow graphs and are suitable technique for the representation of changes in classifications or groupings of information. We refer to our implementation as a *trend graph*.

4.2.1 Trend Graph Specification

A *trend graph* G contains at least one level L. Levels in the graph are strictly ordered. Each level has at least one node. Nodes belonging to different but adjacent levels may be connected through edges E. The total number of edges between each consecutive pair of levels is constant with each interior node having exactly one outgoing edge per incoming edge.

This graph definition allows the visualization of trend behavior over time. Specifically, trends information is incorporated as follows: A level L_j in G correspond to the set of realizations $F(t_j)$ and its nodes represent clusters C_k of the partition on $F(t_j)$. An edge between nodes in level L_j and L_{j+1} signifies that a specific feature realization F_i (run f_i) is contained in the respective clusters of $F(t_j)$ and $F(t_{j+1})$. Paths through the graph are trends. Fig. 3 presents a graph corresponding to the given specification, along with its representation as an alluvial diagram with edge bundling.

4.2.2 Graph Visualization

The alluvial trend graph's capability to represent trend consistency and change in a concise fashion makes it a core

component of our visualization system. Through the application of appropriate graph visualization techniques, we are able to convey complex trend behaviors over time. Instead of creating a direct rendering of the trend graph, as defined in Section 4.2 (Fig. 3a), we implement the following visual enhancements to arrive at an expressive 2D rendering of a trend graph:

- *Nodes:* Nodes are rendered as spheres. This shape makes it clearly distinguishable from edges in the graph. Node areas are scaled according to the size of the cluster it represents. Furthermore, nodes are colored according to various scalar quantities, as defined throughout this paper.
- *Edges:* Edges are bundled into ribbons with widths reflecting the total number of realizations contained in the edge. For better visibility of total incoming/outgoing realizations and branching information, edges are rendered as transparent sigmoid curves. Parts of edge bundles are highlighted in a distinct color, if specific subsets of ensemble runs are to be emphasized in the graph, e.g., to reflect user selection.

These modifications result in a graph rendering as seen in Fig. 3b. Note that the given example graph only contains two levels and a maximum of three trends per level. With possible ensemble sizes beyond a few simulation runs and a handful of time-steps, graph sizes grow significantly. While there are no theoretical limits to graph sizes, the screen space available for graph drawing is typically limited, raising questions about scalability. The dimensions of a trend graph in x and y-direction are determined directly by the number of time-steps of the ensemble, and the maximal number of trends, respectively. The latter directly reflects behavioral complexity in the ensemble and is influenced by characteristics of the trends extraction/clustering process. The former leaves room for scalability improvement. If a visualization of the trend graph is to be shown in its entirety (as opposed to a sub-graph representing an interval in time), different resolutions of *summary graphs* may be employed. We create a summary graph at level n by keeping every 2^nth level, while removing intermediate ones. Edges between these levels are merged appropriately. Complexity of the merge is encoded and visualized as scalar-value per node. An example is given in Fig. 4.

4.2.3 Interpretation

Given two partitions on the feature descriptors $\mathcal{P}(t_j)$ and $\mathcal{P}(t_{j+1})$ at contiguous time steps, we say an edge e connecting a cluster $C_k(t_j) \in \mathcal{P}(t_j)$ at time j to a cluster $C_p(t_{j+1})$ at time $j+1$ is a *pure transfer* if the features associated with the edge completely make up both the source and destination clusters. That is, all the runs/feature realizations from the source trend form the destination trend, and there are no feature realizations in the destination trend originating in a trend at time j that is not the source of e.

4.2.4 Linked Displays and Interaction

A trend graph is a visualization of a time-varying set of trends that are present in an ensemble. Thus, the trend graph is created from a set of time-varying features F along with a time-interval $[t_s, t_e]$. A new trend graph is created

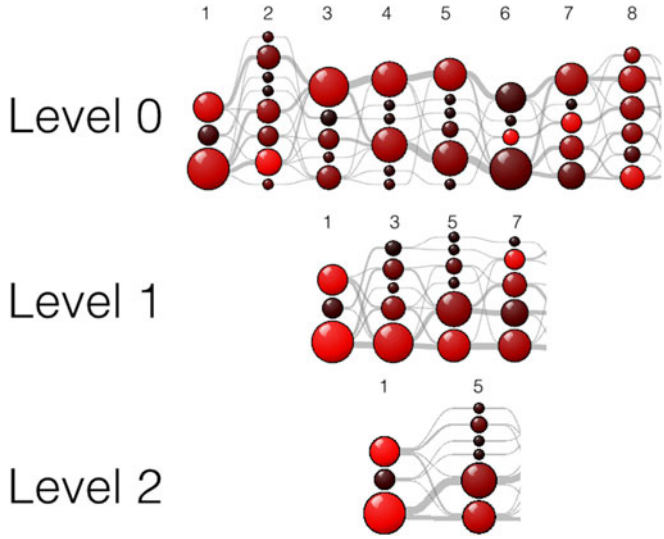

Fig. 4. Trend graphs may be displayed in multiple levels to summarize trend evolution over time by subsampling the time steps in the ensemble data.

whenever this active set of trends changes. Consequently, the regeneration of the trend graph is commonly caused by user interaction in one of the other views of the ensemble data. The trend graph itself allows for one main type of interaction: The selection of clusters. Selection of a cluster (node in the graph) $T_k(t_j) = \{F_{i1}(t_j), F_{i2}(t_j), \ldots\}$ prompts visual feedback in the trend graph. All edges in the graph that contain any $F_i \in C_k(t_j)$ are marked, thus showing the trends that cluster is a part of and allowing the visual tracing of ensemble runs over time, enabling analysis of respective trend memberships. Furthermore, all trend nodes are colored according to their overlap with $C_k(t_j)$, described by a scalar-measure of trend consistency in the range $[0, 1]$:

$$s(T; T_k(t_j)) = \frac{|T \setminus T_k(t_j)|}{|T|}. \quad (1)$$

This node coloring combined with edge highlighting supports direct analysis of trend consistency and evolution over time. An example of this visualization is given in Fig. 5. Furthermore, the selection of individual trends in the graph view prompts updates in the remaining views of the trends analysis system.

4.3 Integrated Parameter-Space

A tight integration of parameter-space into the ensemble analysis framework is crucial [30] and serves to answer several important questions: *Are trends in simulation behavior influenced by parameter choice? How can parameter-choice affect the behavior of the simulation?* For these reasons, we integrate a view of parameter-space into the trends analysis framework.

4.3.1 Parameter-Space Properties

Ensemble parameter-space P is a multi-dimensional space with dimensions corresponding to a wide range of simulation parameters such as temperature, resolution, fluid viscosity, or pressure. Due to the incomparability of individual parameters, P is not a metric space. Despite the absence of a metric, it can generally be assumed that a total order exists for all values along the same axis (e.g., pressure values). For

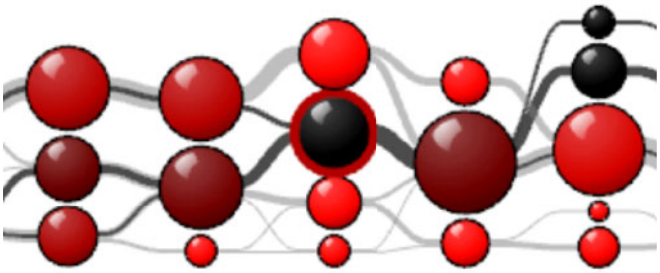

Fig. 5. Partial trend graph with one selected cluster (red outline). Edges in the graph are highlighted to facilitate the tracing of sets of feature realizations over time. This allows for the in-depth analysis of cluster behaviors in the ensemble. Additionally, nodes are shaded according to *consistency*—Eq. (1).

the purposes of this paper, we only deal with real-valued scalar parameters, which always have a total order.

4.3.2 Parameter-Space Visualization

For this reason, we favor a visualization approach based on parallel coordinates over classic 2D or 3D projective approaches, where relative scaling of axes would play a major role. In our parallel-coordinates graph, each dimension in P is represented as a parallel coordinate axis. An individual simulation runs f_i is consequently represented as a poly-line that intersects coordinate axes at its parameter-values $p_i^j \in P$. We enhance this classic parallel-coordinates technique with clustering capabilities.

Given a partition of F_t, which in turn implies a classification of $f_i \in E$ for the given time, we highlight individual clusters of ensemble members by enforcing selective bundling in our parallel-coordinates view. This form of clustering has significant advantages over approaches that solely rely on edge-coloring, and more clearly resolves cluttering and occlusion issues (see work by Heinrich et al. [14] for an evaluation of edge bundling in PC). Fig. 6 presents the resulting visualization for a 4D parameter-space. We create these edge bundles by replacing poly-lines with piecewise cubic Hermite splines. For reasons of notational simplicity we refer to the direction perpendicular to parameter axes as x and to the direction along parameter axes as y. We replace pairs of control points p_i^j and p_i^{j+1} of a poly-line with triples p_i^j , $c_i^{j+1/2}$, p_i^{j+1} and tangent vectors according to the following steps. Hereby, j is the index of the left coordinate axis and $j+1$ the index of the right coordinate axis. The index $j+1/2$ refers to the implicit axis half way in-between axis j and $j+1$.

1) For every cluster C_k, we compute its average location along y as $o^k = \frac{1}{2|T|}\sum_{F_i \in T_k}(p_i^j + p_i^{j+1})$.
2) Following the relative order determined by o^k, trend centers are placed at equidistant positions c^k along axis $j+1/2$.
3) For every line p_i^j, p_i^{j+1} that is part of trend T_k with center c^k, we create one additional control point as $c_i^{j+1/2} = (p_i^j + p_i^{j+1}) \cdot 0.1 + c^k \cdot 0.8$
4) We compute tangent vectors at p_i^j , $c_i^{j+1/2}$ and p_i^{j+1} as $(1,0)$, $p_{j+1} - p_j$, and $(1,0)$.
5) We create two piecewise cubic Hermite splines from these new control points and tangent vectors.

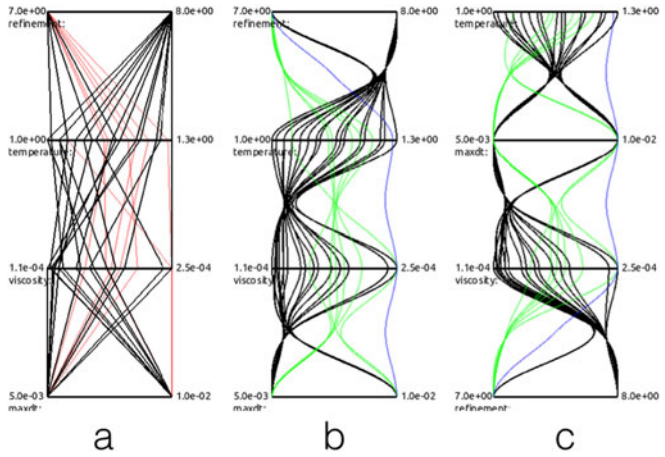

Fig. 6. a) Parallel coordinates visualization of a four-dimensional parameter-space, showing runs from a selected cluster in black. b) The same space with edge bundling by cluster. c) Axes re-ordered to highlight correlation.

Additionally, we reorder axes by pairing axes with largest absolute correlation to highlight correlation between pairs of coordinate axes.

4.3.3 Linked Views and Interaction

The basic input to the parameter-space view consists of parameter settings for all ensemble members ($P = \{P_i\}$). Additionally, a partition of the runs into feature definition clusters ($\mathcal{P}_t = \{C_k\}$) at some time t may be given from other components of the trends analysis system, such as the trend graph. There are two ways for direct interaction with the rendering of parameter-space. First, the user may select individual runs. These selected runs are highlighted both in the trends graph and are used for runs-based visualization in the ensemble view. Second, the user may deactivate ensemble members. This effectively takes runs out of the ensemble and removes them from consideration in any of the components of the trends analysis system. In the same manner, users can reactivate runs that have previously been excluded from the ensemble.

4.3.4 Separability of Trends

There are several insights to be gained from parameter-space enhanced ensemble analysis. One key insight related to ensemble parameter-space is the concept of separability. Trend separability is a measure of how well trends in the ensemble correspond to distinct clusters in parameter-space. Linear separability of a pair of two trends, for example, means that a hyperplane exists in the multi-dimensional parameter-space that separates members of trend 1 from members of trend 2. Being able to measure separability of trends in parameter-space facilitates in-depth analysis of trends: If trends are consistent in the ensemble but not well-separable in parameter-space, this implies that the shown behaviors may be due to complex physical interactions or due to numerical artifacts. Trends that are consistent and clearly separable in parameter-space, however, are prime candidates for computational steering and simulation design. The parameter-space visualization presented in this section allows for an easy identification of behaviors/trends that are linearly separable along one dimension (e.g., *"All simulation runs with a fluid temperature*

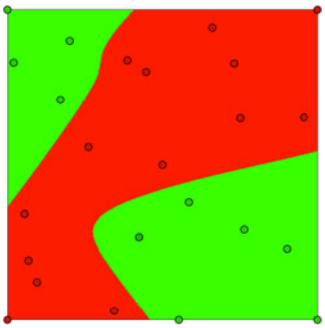

Fig. 7. An adaptive neural network is used to estimate a 2D decision boundary for two classes. Boundary complexity can be readily estimated through the analysis of NN connectivity.

above 60F belong to trend 1*, all other simulation runs show a behavior corresponding to trend* 2″). With an appropriate ordering of axes, linear separation of trends along two axes may also be identified. In general, however, the complexity of separability of trends in parameter-space is hard to estimate in a visual framework. For these reasons, we employ automatic approximations of separability.

In the context of machine learning processes, data classifications are performed by computing a decision boundary, i.e., a (geometric) boundary that separates distinct classes. The complexity of this decision boundary is therefore a direct indicator of the degree of separability. Since our system uses arbitrarily high-dimensional parameter-spaces without intrinsic metrics and may contain arbitrarily complex cluster boundaries, we employ Neural Networks to approximate given clusters and estimate the corresponding decision boundary. Specifically, we use the C++ FANN-library [28] for NN training and evaluation. An adaptive cascade training algorithm ensures that trained NNs approximate the clustered point-set appropriately. The dynamic nature of the created NNs allows us to estimate decision-boundary complexity directly by examining the number of nodes and connections present in the final NN. To avoid random bias, we train several instances of NNs with varying initializations and use average network complexity as the final estimate for decision boundary complexity and separability. Fig. 7 gives an example for 2D decision boundary complexity estimation with NNs. The separability of a given trend T_k is thus given as the complexity of its boundary (its NN) with $T \setminus T_k$.

4.4 The Ensemble View

The two views presented so far do not provide a view of the spatial data given by the ensemble. In order to provide a summary view of the ensemble data, we include what we term an *ensemble view*.

4.4.1 Visualization Technique

Since all members of the ensemble share a common domain, we are able to apply standard single-field visualization techniques after compositing or aggregating the ensemble data into a single field. Given a time-step of interest, we employ such standard scalar- and vector-field visualization techniques to present a summary view of the ensemble to a user. Scalar values given per cell of the data set, such as numerical averages, or scalar-valued trends descriptors allow the application of volume rendering techniques in 2D and 3D. In order to enable direct interaction with 3D data, we employ slicing. Additionally, the feature set F used for trends analysis is shown upon selection of a location in the ensemble view, as well. This includes rendering of sets of pathlines, curves, or distributions. Scalar distributions are represented by a histogram, and pathlines and curves are just directly rendered on the domain.

4.4.2 Scalar Quantities

We provide a range of useful scalar measures for data summarization, which are used during rendering in the ensemble view.

- *Means:* Shown scalar values are averages over all runs, e.g., average temperature, pressure, velocity magnitude.
- *Disagreement*: The maximum number of trends per time-step over an interval in time. A low disagreement score means that realizations of the investigated feature behave similarly within the ensemble.
- *Similarity:* For a given set of ensemble runs, f_i, as given by selection in the trend graph or PC, scalar values at every voxel of the data set indicate the length of the maximal time interval for which all f_i belong to the same trend.
- *Trend Complexity:* We define two metrics to help classify the trend graph: branchiness and summarization. We say the *branchiness* of a trend graph is the total number of edges which are not pure transfers. *Summarization* is how well the trends summarize or reduce the ensemble members. Precisely, summarization $S = \mathsf{NumMembers}/\mathsf{NumTrends}$. For each trend graph in the domain, we compute the branchiness and summarization metrics, and formulate a four-way classification on that metric space. Average branchiness and average summarization is chosen as the splitting point, with trend graphs classified as having high branchiness and high summarization (colored in blue), low branchiness and high summarization (colored in green), high branchiness and low summarization (colored in yellow), and low branchiness and low summarization (colored in pink).

4.4.3 Linking and Interaction

Besides visualizing scalar quantities, as determined through user-interaction in the system, the ensemble view also provides a visualization of the feature set F. If a certain trend is selected in the analysis framework, the ensemble view highlights this particular subset of F for the currently active time step. Direct interaction with the ensemble view consists of feature selection. More specifically, we provide functionality to select locations in the data set through standard mouse interaction. This selection initializes the extraction of features F, such as time-varying flow particle positions (path-line extraction), or simple sets of time-varying scalar values. In 3D, location selection is performed on manually placed slicing planes. This new feature set is displayed either as a histogram of scalar values or is rendered directly on the domain as pathlines. Additionally, once a domain

location is selected, a trend graph is generated and displayed for that point, starting at the currently selected timestep. Selecting a cluster at a particular time in the trend graph will bundle the edges in the parameter space view based on the partition of the ensemble predictions computed for that time step. For scalar ensemble datasets, selecting a cluster at a time step changes the display histogram to that timestep and highlights in red the portion of that histogram belonging to the selection cluster. In flow datasets, the displayed pathlines remain rooted at the timestep selected when the user chose a location, but the particle positions at the time selected on the trend graph are rendered as circles. Particle positions belonging to the selected cluster are drawn in red, and those not belonging to the selected cluster are drawn in black. The user may also use mouse interaction to select a single edge from the parameter space view, representing one ensemble constituent. Doing so will superimpose a thin black line on the edges of the trend flow graph to show the 'path' of the ensemble constituent through time and it's trend affinity.

5 Results and Discussion

5.1 Data Sets

We present analysis results obtained by applying the proposed techniques to three different ensembles. All three ensembles are results of simulating flow phenomena and contain fields that encode vector and scalar quantities.

5.1.1 Heated Cylinder: 2D

This ensemble contains a set of 25 2D fields, simulating turbulent flow around a heated cylinder. Simulation were driven by a standard finite-element simulation code and represent 121 time-steps (12 seconds in real time) in a 129 by 385 or 257 by 769 uniform grid. Velocity, temperature, and vorticity data are available for analysis.

This ensemble studies the effects of parameters on a simulation of convective heat transfer (Boussinesq approximation). More specifically, the characteristics and control of the vortex-shedding process are investigated. Such vortices may be used to control vibrations and drag in a variety of application scenarios.

Ensemble members are generated through the variation of four parameters. Two parameters pertain to physical quantities – viscosity of the fluid and temperature of the heated cylinder, whereas two are simulation settings – length of the simulation time-step, and resolution of the grid.

5.1.2 Stirring Tank: 2D

The data examined in this application is a representation of a 2D stirring tank. The ensemble contains 21 members with a total range of 75 time-steps (7.5 seconds in real time). The simulation was performed by a custom mesh-free simulation code (SPH) and resampled to a uniform grid for persistent storage. Spatial resolution of the data set is 168 by 168 cells. Only velocity data is available for analysis.

The design of stirring devices is heavily coupled to the optimization of mixing behavior. One factor that strongly influences the performance of stirring devices is the presence of "dead" pockets, i.e., regions in the stirring device that remain homogeneous and do not mix well with the rest of the mixing matter. This ensemble provides data to investigate the quality of a stirring device for a range of physical properties.

Assuming a fixed stirring device architecture, the present ensemble varies viscosity and stiffness of the mixing matter to explore mixing conditions for various materials. Additionally, we vary the spatial scale of the simulation to establish scale invariant conclusions. Thus, parameter-space of this ensemble is three dimensional.

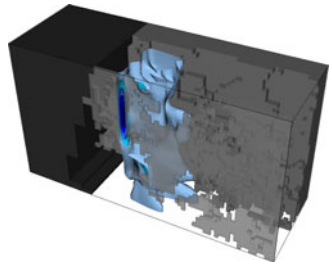

Fig. 8. The Rifle Site dataset simulates the groundwater flow of a tracer chemical through a medium of varying physical properties. Here, the plume of bromine is depicted in blue contours at an early timestep.

5.1.3 Groundwater Transport: 3D

This data set is an ensemble of 20 simulations modeling the propagation of a bromine tracer chemical through groundwater in Rifle, Colorado [52]. Each realization makes a prediction of bromine concentration every 60 hours on a $56x40x43$ cell simulation grid, given a distribution of soil types at each cell. We use average values for three material properties over the mesh for each ensemble constituent as our parameter space values. Fig. 8 shows the bromine plume at an early timestep in blue, with some of the more impermeable regions rendered in gray.

5.2 Ensemble Analysis

We start by examining the velocity output of the 2D heat convection example. We evaluate the formation and consistency of trends through selection in the ensemble view. For the selected position, a set of pathlines is created – one per ensemble member. Positions on these pathlines serve as feature instances F_i during trend extraction. Fig. 9 shows the resulting trend graph and corresponding parameter-space. The selection of trends in the trend graph highlights modes and features corresponding to the selected trend, and causes axis reordering and clustering in the parameter-space view. The proposed visualization technique makes it easy to identify outliers together with their parameters. Interestingly, it becomes evident that, for this selected position, the resolution of the underlying simulation (refinement level) has a major impact on the formation of trends. This is an important insight as it indicates that internal simulation parameters may locally influence the validity of research conclusions – a notion that highlights the importance of collaboration between computational scientists and domain experts.

This result prompts us to further investigate the similarity of ensemble runs with low and high refinement. Using interaction in the parameter-space view, we group ensemble members based on refinement. The resulting similarity

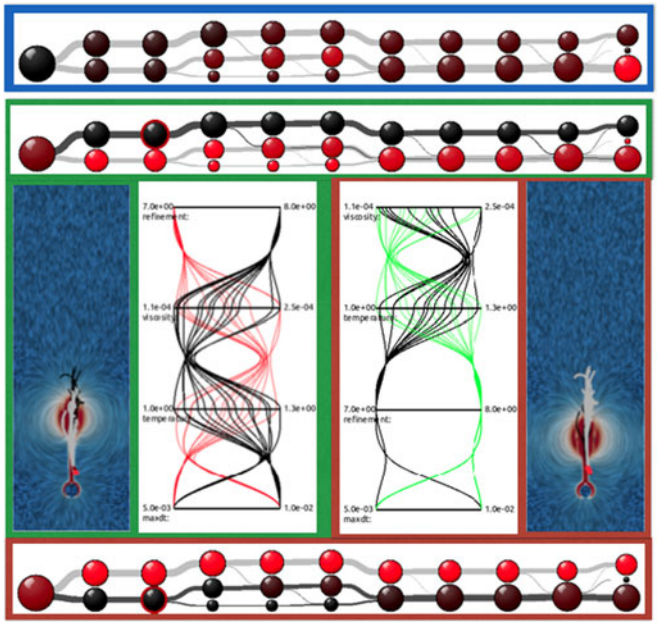

Fig. 9. Views from analyzing flow transport in the 2D heat convection ensemble. The feature realizations are pathline trajectories seeded at the location in question. Red dots are the locations of the seeds for the selected cluster. Gray lines are the entire trajectories of ensemble constituents in the currently selected cluster, and black lines are the trajectories corresponding to non-selected ensemble constituents. The initial flow graph, boxed in blue at the top, shows trend-clusters colored by their seperability in parameter-space, with darker shades corresponding to lower parameter space seperation complexity (and thus higher seperability) and brighter reds indicating higher complexity seperations. We note that the early split in the trend graph shows high seperability, and that trends are mostly consistent over time. Ensemble, trend flow graph, and parameter-space views for each of the trends at the third time step are depicted with red and green backgrounds. We note very strong, even perfect, seperability on the refinement axis.

score is shown in Fig. 10. We conclude that while simulation resolution has an impact on concrete transport behaviors present in this 2D heat convection, the two observed resolution trends only emerge from locations close to the heating cylinder and on lateral regions of the domain. That we are examining a Lagrangian property of the ensemble means that trends forming in these locations govern ensemble behavior in other regions of the ensemble, as the fluid is transported upwards.

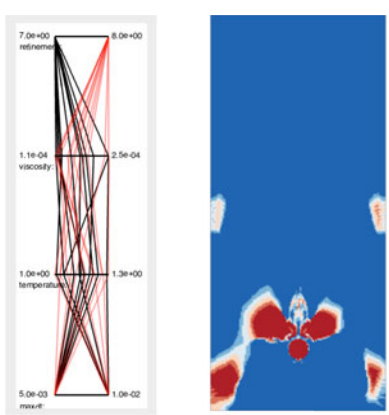

Fig. 10. We examine parameter-space separability for transport in the 2D heat convection data set. This scalar field shows the similarity score for ensemble member groups defined by “refinement”.

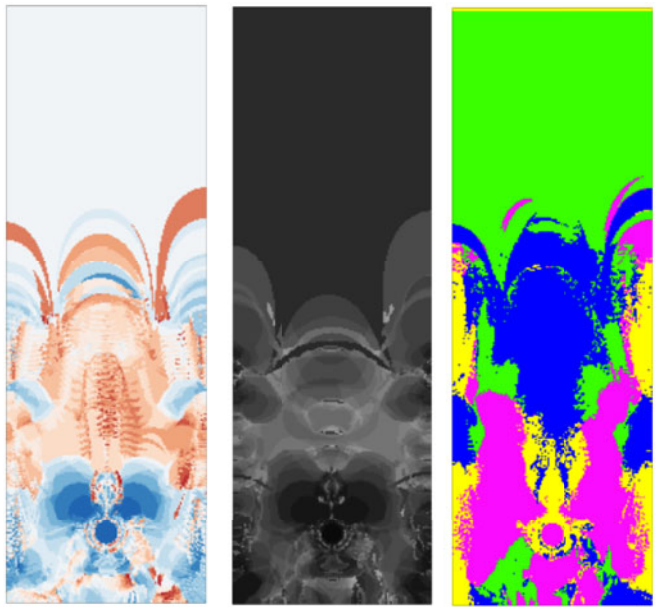

Fig. 11. Different enesemble visualization metrics for transport in the 2D heat convection ensemble. On the left, disagreement score is displayed with a blue-white-red color map, showing the maximum number of clusters the ensemble splits into for a given point. In the center, with a gray-scale colormap, is the summarization score. On the right is a four-way classification of trend complexity. Locations which have flow graphs with high branchiness and high summarization are rendered in blue. Low branchiness and high summarization maps to green. High branchiness and low summarization are denoted by yellow locations, and low branchiness and summarization locations are shown in pink.

Fig. 11 uses additional scalar quantities for ensemble visualization to provide further insights into spatially varying characteristics of ensemble trends. Notice how trend complexity is high along turbulent regions of the ensemble, as expected.

We now examine non-Lagrangian quantities in the 2D heat ensemble, specifically the behavior of heat at spatially fixed locations in the ensemble. While the analysis of Lagrangian properties revealed the formation of consistent trends, the non-Lagrangian investigation of locations in an ensemble that models transport produces significantly less consistent trends.

We examine heat distributions in the data set. In Fig. 12, we see four modes present in the ensemble. Since the investigated feature instances are scalar (temperature), the set of feature realizations is displayed as a histogram, highlighting the currently selected trend in black. For the given selection, we can observe four modes with two outliers. Note how the relevancy of the computed trends decreased with distance from the selection location, as indicated by blurred modes.

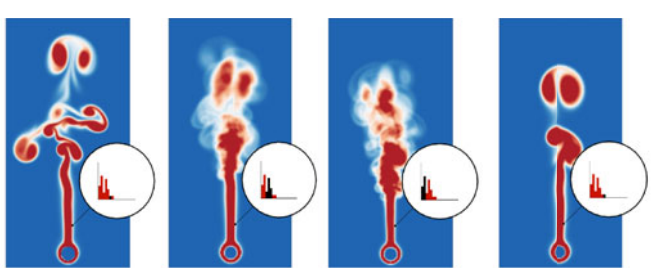

Fig. 12. Analyzing temperature distributions in the 2D heat convection ensemble.

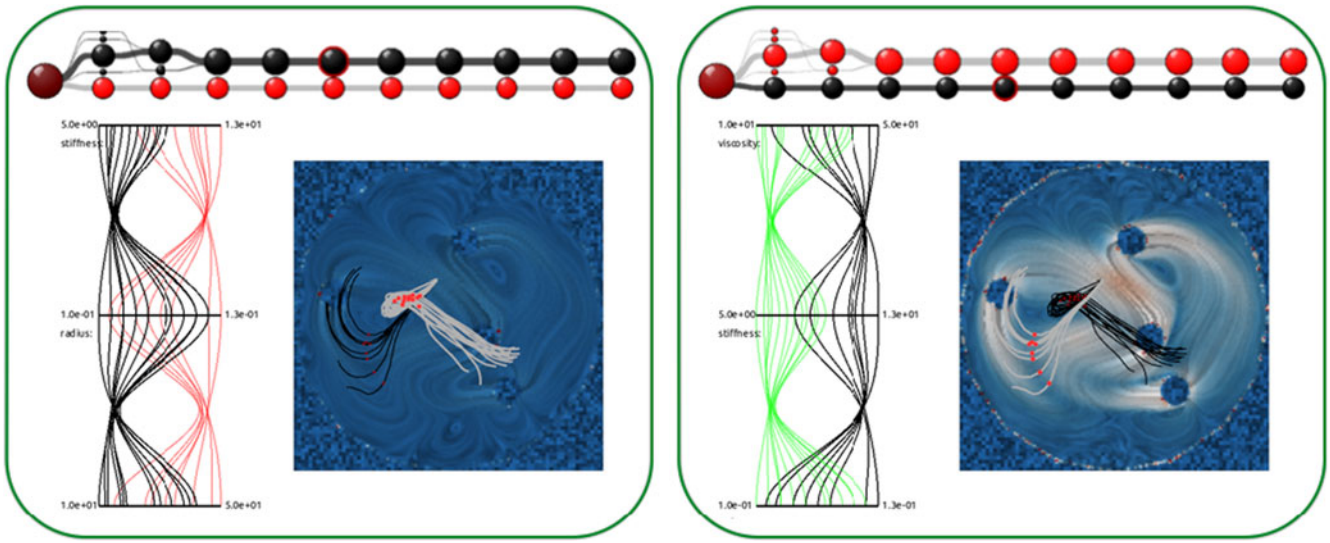

Fig. 13. Analyzing transport trends in the 2D stirring ensemble. Notice how clustering behavior of the path-lines over time is well-reflected by splitting in the trend graph. The parameter-space visualization allows for identification of parameter settings of runs with outlier behaviors.

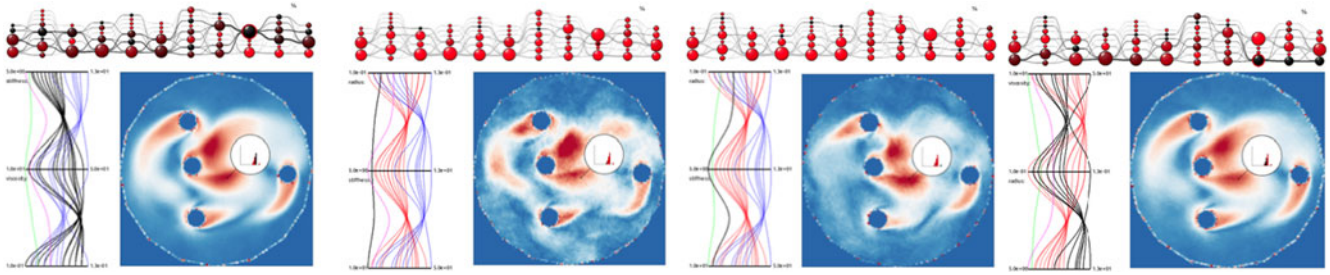

Fig. 14. Analyzing modes in speed for the 2D stirring ensemble. Note how outliers appear to be well-separated in output and remain consistently different from the rest of the ensemble. However, there is no clear visual separability in parameter-space apart from a general tendency for extrema in parameter-space to cluster together.

In a similar fashion we analyze properties of the stirring ensemble. The analysis of transport in Fig. 13 reveals stiffness as a governing factor in the formation of trends. Here, outlying behavior seems to correlate to extremal values in stiffness.

A different impression is gained by investigating whether distinct trends in velocity magnitude are detectable in the ensemble. Fig. 14 gives an example of velocity magnitude clustering in the stirring ensemble. Besides the detection of an outlier in the ensemble, no consistent trends are evident. The presence of a consistent outlier is an interesting observation. This outlying behavior may indicate a boundary case or numerical artifact in the simulation code and prompts us to initiate review of the particular simulation code.

The last aplication studied in this paper, the Rifle site groundwater predictions, focuses on the analysis of bromine transport behaviors through rock/soil media of varying physical properties. In Fig. 15 we select a location on a 2D slice plane placed through the injection well. We note some bimodal behavior at our first location that is not well-reflected by visually apparent seperation in parameter-space. We note that the trend complexity in the temporal vicinity of this bimodality indicates that this two-fold behavior pattern is likely transitory. Clicking on preceding and proceeding nodes confirm this hypothesis. However, a different location, shown in Fig. 16 shows an outlier which is indeed obviously apparent in the parameter-space. This outlier also can be seen to persist in the flow graph without splitting or merging for several time-steps, potentially indicating a phenomenon of some importance.

6 Conclusions and Future Work

We have presented a system for the analysis of time-varying trends in ensembles. The proposed visualization and interac-

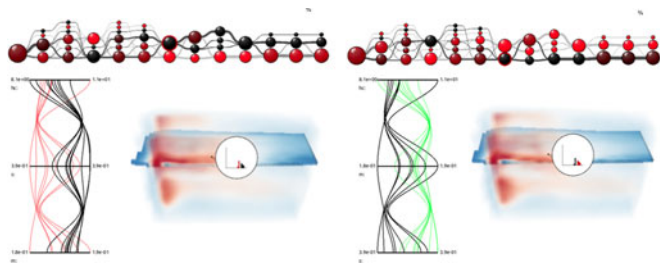

Fig. 15. Analyzing the bromine plume trends in the 3D Rifle site data. Selecting the two-cluster level near the middle of the trend graph, we note that this corresponds to a cleanly bimodal (see histograms) set of predictions. While some weak bimodal grouping is evident in the parameter-space, this may be evidence of some other influencing factor not captured by global parameter space values for each ensemble constituent, or that the bimodal behavior is short-lived. A glance at the trend complexity at preceding and proceeding timesteps indicate that the latter is indeed the case.

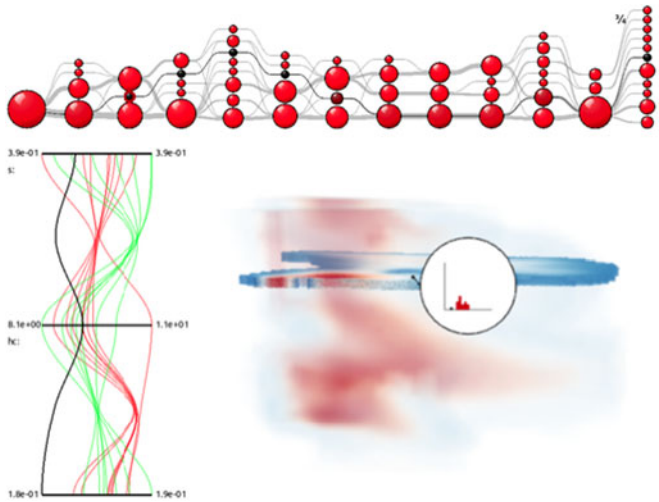

Fig. 16. Analyzing the bromine plume trends in the 3D Rifle site data. We have selected a node whose location on the histogram and node size show it to be a low-bromine concentration outlier. Here, the parameter-space immediately shows us that the outlier prediction corresponds to an extreme parameter choice.

tion mechanisms have proven to provide suitable solutions for the analysis of parameter-dependence in complex data ensembles. We observe that while the presented graphs and scalar metrics allow for the detection and analysis of clustering behaviors within ensembles, the expressiveness of obtained results is dependent on the task and data availability. The proposed solution requires the knowledge of used parameter-value data for every run in the ensemble, an assumption that is often hard to fulfill in practice, especially in a purely post-processing based approach. We plan to investigate the impact and feasibility of incorporating estimated (or abstract) parameter-values into the analysis process.

Acknowledgments

This work was supported in part by the National Science Foundation (IIS 0916289 and IIS 1018097), and the Office of ASCR, Office of Science, through the Department of Energy SciDAC contract DE-FC02-06ER25780, and contract DE-FC02-12ER26072 (Scalable Data Management, Analysis, and Visualization, SDAV). The authors thank Pacific Northwest National Lab for the Rifle site data and the members of the Institute for Data Analysis and Visualization (IDAV) at UC Davis for fruitful discussions.

References

[1] O. S. Alabi, X. Wu, J. M. Harter, M. Phadke, L. Pinto, H. Petersen, S. Bass, M. Keifer, S. Zhong, C. Healey, and R. M. Taylor II, "Comparative visualization of ensembles using ensemble surface slicing," *Proc. SPIE Vis. Data Anal.*, vol. 8294, pp. 82940U-1–82940U–12, 2012.

[2] W. Berger and H. Piringer, "Interactive visual analysis of multiobjective optimizations," in *Proc. IEEE Symp. Vis. Analytics Sci. Technol.*, 2010, pp. 215–216.

[3] S. Bergner, M. Sedlmair, T. Möller, S. N. Abdolyousefi, and A. Saad, "Paraglide: Interactive parameter space partitioning for computer simulations," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 9, pp. 1499–1512, Sep. 2013.

[4] R. P. Botchen, D. Weiskopf, and T. Ertl, "Interactive visualisation of uncertainty in flow fields using Texture-based techniques," in *Proc. 12th Int. Symp. Flow Vis.*, 2006.

[5] S. Bruckner and T. Möller, "Result-driven exploration of simulation parameter spaces for visual effects design," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1467–1475, Oct. 2010.

[6] D. Butnaru, B. Peherstorfer, H. Bungartz, and D. Pfluger, "Fast insight into High-dimensional parametrized simulation data," in *Proc. 11th Int. Conf. Mach. Learn. Appl.*, 2012, vol. 2, pp. 265–270.

[7] D. Coffey, C.-L. Lin, A. G. Erdman, and D. F. Keefe, "Design by dragging: An interface for creative forward and inverse design with simulation ensembles," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2783–2791, Dec. 2013.

[8] R. Fuchs, J. Waser, and M. Gröller, "Visual human+machine learning," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 6, pp. 1327–1334, Nov./Dec. 2009.

[9] N. Gershon, "Visualization of an imperfect world," *IEEE Comput. Graph. Appl.*, vol. 18, no. 4, pp. 43–45, Jul./Aug. 1998.

[10] L. Gosink, K. Bensema, T. Pulsipher, H. Obermaier, M. Henry, H. Childs, and K. I. Joy, "Characterizing and visualizing predictive uncertainty in numerical ensembles through Bayesian model averaging," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2703–2712, Dec. 2013.

[11] H. Griethe and H. Schumann, "The visualization of uncertain data: Methods and problems," in *Proc. SimVis*, 2006, pp. 143–156.

[12] H. Guo, X. Yuan, J. Huang, and X. Zhu. "Coupled ensemble flow line advection and analysis," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2733–2742, Dec. 2013.

[13] G. Haller, "Distinguished material surfaces and coherent structures in Three-dimensional fluid flows," *Physica D: Nonlinear Phenomena*, vol. 149, no. 4, pp. 248–277, 2001.

[14] J. Heinrich, Y. Luo, A. E. Kirkpatrick, and D. Weiskopf, "Evaluation of a bundling technique for parallel coordinates," in *Proc. Int. Conf. Inf. Vis. Theory Appl.*, 2012, pp. 594–602.

[15] M. Hlawatsch, P. Leube, W. Nowak, and D. Weiskopf, "Flow radar Glyphs – static visualization of unsteady flow with uncertainty," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1949–1958, Dec. 2011.

[16] M. Hummel, H. Obermaier, C. Garth, and K. I. Joy, "Comparative visual analysis of Lagrangian transport in CFD ensembles," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2743–2752, Dec. 2013.

[17] T. J. Jankun-Kelly and K.-L. Ma, "A spreadsheet interface for visualization exploration," in *Proc. Vis.*, 2000, pp. 69–76.

[18] C. Johnson, S. Parker, C. Hansen, G. Kindlmann, and Y. Livnat, "Interactive simulation and visualization," *Computer*, vol. 32, no. 12, pp. 59–65, Dec. 1999.

[19] C. Johnson and A. Sanderson, "A next step: Visualizing errors and uncertainty," *IEEE Comput. Graph. Appl.*, vol. 23, no. 5, pp. 6–10, Sept./Oct. 2003.

[20] D. A. Keim, "Information visualization and visual data mining," *IEEE Trans. Vis. Comput. Graph.*, vol. 8, no. 1, pp. 1–8, Jan.–Mar. 2002.

[21] J. P. C. Kleijnen. *Design and Analysis of Simulation Experiments*, volume 111 of *International Series in Operations Research and Management Science*. New York, NY, USA: Springer, 2008.

[22] G. Lebanon and F. Sha. (2010). Machine learning for information visualization. *Proc. IEEE Conf. Vis. - Tutorial* [Onlie]. Available: http://www.cc.gatech.edu/~lebanon/pub/tutorialPart1.pdf

[23] K.-L. Ma, "Image graphs—a novel approach to visual data exploration," in *Proc. Conf. Vis.*, 1999, pp. 81–88.

[24] K. Matkovic, D. Gracanin, M. Jelovic, A. Ammer, A. Lez, and H. Hauser, "Interactive visual analysis of multiple simulation runs using the simulation model view: Understanding and tuning of an electronic unit injector," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1449–1457, Nov./Dec. 2010.

[25] K. Matkovic, D. Gracanin, M. Jelovic, and Y. Cao, "Adaptive interactive multi-resolution computational steering for complex engineering systems," in *Proc. Int. Eurovis. Workshop Vis. Analytics*, 2011, pp. 45–48.

[26] K. Matkovic, D. Gracanin, M. Jelovic, and H. Hauser, "Interactive visual steering - rapid visual prototyping of a common rail injection system," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1699–1706, Nov./Dec. 2008.

[27] J. D. Mulder, J. J. van Wijk, and R. van Liere, "A survey of computational steering environments," *Future Gener. Comput. Syst.*, vol. 15, no. 1, pp. 119–129, Feb. 1999.

[28] S. Nissen. (2014). Fast artificial neural network library (FANN) [Online]. Available: http://leenissen.dk/fann/wp/

[29] H. Obermaier, M. I. Billen, K. I. Joy, H. Hagen, and M. Hering-Bertram, "Visualization and multivariate clustering of scattered moment tensors," *Inf. Vis.*, vol. 11, no. 1, pp. 43–59, 2012.
[30] H. Obermaier and K. I. Joy, "Future challenges for ensemble visualization," *IEEE Comput. Graph. Appl.*, vol. 34, no. 3, pp. 8–11, May/Jun. 2014.
[31] M. Otto, T. Germer, H.-C. Hege, and H. Theisel, "Uncertain 2d vector field topology," *Comput. Graph. Forum*, vol. 29, no. 2, pp. 347–356, 2010.
[32] M. Otto, T. Germer, and H. Theisel, "Uncertain topology of 3d vector fields," in *Proc. IEEE Pacific Vis. Symp.*, 2011, pp. 67–74.
[33] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha, "Approaches to uncertainty visualization," *Vis. Comput.*, vol. 13, pp. 370–390, 1997.
[34] C. Petz, K. Poethkow, and H.-C. Hege, "Probabilistic local features in uncertain vector fields with spatial correlation," *Comput. Graph. Forum*, vol. 31, no. 3pt2, pp. 1045–1054, 2012.
[35] H. Piringer, S. Pajer, W. Berger, and H. Teichmann, "Comparative visual analysis of 2d function ensembles," *Comput. Graph. Forum*, vol. 31, pp. 1195–1204, 2012.
[36] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson, "Ensemble-vis: A framework for the statistical visualization of ensemble data," in *Proc. IEEE Workshop Knowl. Discovery Climate Data: Prediction, Extremes*, 2009, pp. 233–240.
[37] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson, "Visualization of uncertainty and ensemble data: Exploration of climate modeling and weather forecast data with integrated ViSUS-CDAT systems," *J. Phys.: Conf. Series*, vol. 180, no. 012089 (published online), p. 012089, 2009.
[38] A. Pretorius, M.-A. Bray, A. Carpenter, and R. Ruddle, "Visualization of parameter space for image analysis," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2402–2411, Dec. 2011.
[39] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead, "Noodles: A tool for visualization of numerical weather model ensemble uncertainty," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1421–1430, Nov./Dec. 2010.
[40] D. Schneider, J. Fuhrmann, W. Reich, and G. Scheuermann, "A variance based FTLE-like method for unsteady uncertain vector fields," in *Topological Methods in Data Analysis and Visualization II*, Mathematics and Visualization, R. Peikert, H. Hauser, H. Carr, and R. Fuchs, Eds. Berlin, Germany: Springer, 2012, pp. 255–268.
[41] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller, "Visual parameter space analysis: A conceptual framework," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2161–2170, Dec. 2014.
[42] K. M. Smith, D. C. Banks, N. Druckman, K. Beason, and M. Y. Hussaini, "Clustered ensemble averaging: A technique for visualizing qualitative features of stochastic simulations," *J. Comput. Theoretical Nanosci.*, vol. 3, pp. 752–760, 2006.
[43] W. Stuetzle, "Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample," *J. Classification*, vol. 20, no. 1, pp. 025–047, 2003.
[44] L. Tweedie, B. Spence, H. Dawkes, and H. Su, "The influence explorer (Video) – a tool for design," in *Proc. Conf. Companion Human Factors Comput. Syst.*, 1996, pp. 390–391.
[45] T. Urness, V. Interrante, E. Longmire, I. Marusic, S. O'Neill, and T. Jones, "Strategies for the visualization of multiple 2d vector fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 4, pp. 74–82, Jul.-Aug. 2006.
[46] V. Verma and A. Pang, "Comparative flow visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 10, no. 6, pp. 609–624, Nov./Dec. 2004.
[47] J. Waser, R. Fuchs, H. Ribicic, B. Schindler, G. Blöschl, and E. Gröller, "World lines," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1458–1467, Nov./Dec. 2010.
[48] R. Whitaker, M. Mirzargar, and R. Kirby, "Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2713–2722, Dec. 2013.
[49] J. J. Wijk and C. Overveld, "Preset based interaction with high dimensional parameter spaces," in *Data Visualization, vol. 713 of The Springer International Series in Engineering and Computer Science*, F. Post, G. M. Nielson, and G.-P. Bonneau, Eds. New York, NY, USA: Springer, 2003, pp. 391–406.
[50] A. T. Wilson and K. C. Potter, "Toward visual analysis of ensemble data sets," in *Proc. Workshop Ultrascale Vis.*, 2009, pp. 48–53.
[51] C. M. Wittenbrink, A. T. Pang, and S. K. Lodha, "Glyphs for visualizing uncertainty in vector fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 2, no. 3, pp. 266–279, Sep. 1996.
[52] S. B. Yabusaki, Y. Fang, P. E. Long, C. T. Resch, A. D. Peacock, J. Komlos, P. R. Jaffe, S. J. Morrison, R. D. Dayvault, D. C. White et al., "Uranium removal from groundwater via in situ biostimulation: Field-scale modeling of transport and biological processes," *J. Contaminant Hydrol.*, vol. 93, no. 1, pp. 216–235, 2007.

Harald Obermaier received the master's and PhD (Dr rer nat) degrees in computer science from the University of Kaiserslautern, Germany, in 2008 and 2011, respectively. He is currently a postdoctoral researcher at the Institute for Data Analysis and Visualization, University of California Davis. His research interests lie in visualization, data analysis, and computational science with applications in areas such as continuum mechanics and earth and environmental sciences. He is a member of the IEEE Computer Society.

Kevin Bensema received the BS degree in computer science from Grove City College. He is currently a graduate student at the University of California at Davis. His research interests lie in visualization and analysis of scientific data ensembles.

Kenneth I. Joy received the BA and MA degrees in mathematics from UCLA in 1968 and 1972, respectively, and the PhD degree from the University of Colorado, Boulder, in 1976. He is a professor with the Department of Computer Science and the director in the Institute for Data Analysis and Visualization at the University of California, Davis (UC Davis). He came to UC Davis in 1980 in the Department of Mathematics and was a founding member of the Computer Science Department in 1983. He is a faculty computer scientist at Lawrence Berkeley National Laboratory and is a participating guest researcher at Lawrence Livermore National Laboratory. His areas of interest lie in the fields of visualization, geometric modeling, and computer graphics, where he leads research efforts in multiresolution representations of large-scale data sets, visualization of multidimensional data, applications and visualization algorithms to imaging problems, and simplification of data sets resulting from terascale simulations. He is a member of the Association for Computing Machinery (ACM) and the IEEE.