

# Multi-Charts for Comparative 3D Ensemble Visualization

Ismail Demir, Christian Dick and Rüdiger Westermann

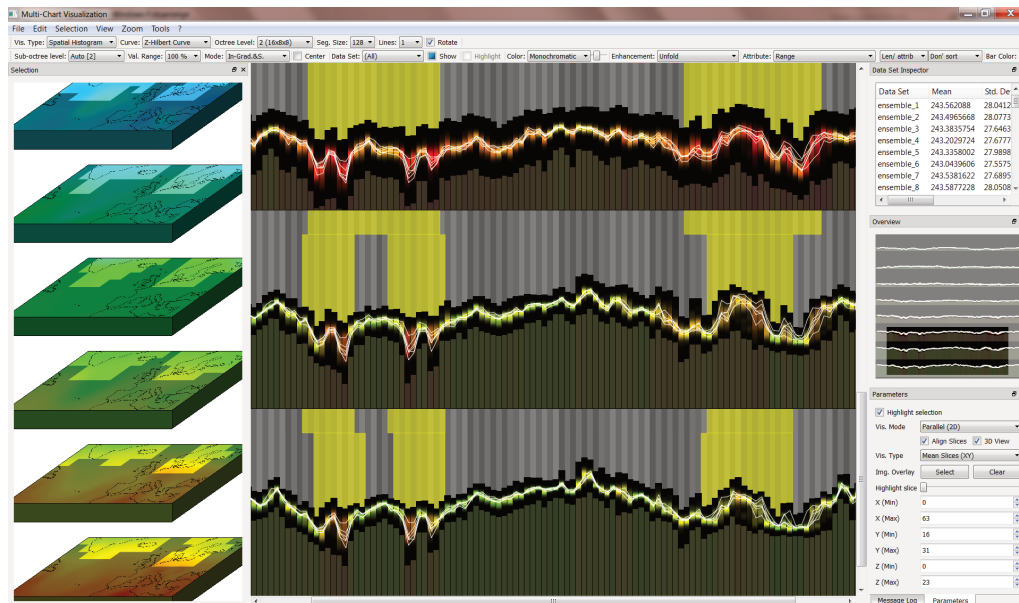


Fig. 1. Multi-chart visualization of a temperature ensemble forecast from the ECMWF Ensemble Prediction System (EPS), ECMWF's operational ensemble weather forecast system [30]. The ensemble consists of 51 members of resolution  $256 \times 128 \times 64$  each. Each bar in the multi-chart is associated with a distinct 3D subdomain, and encodes the distribution of the ensemble members in this subdomain by means of a histogram. In addition, a few user-selected ensemble members are depicted using polylines. By means of brushing in the multi-chart view (indicated by yellow background color), the user has selected regions where the range over the ensemble members and thus the uncertainty is high. The selected regions are instantly emphasized in the 3D view.

**Abstract**—A comparative visualization of multiple volume data sets is challenging due to the inherent occlusion effects, yet it is important to effectively reveal uncertainties, correlations and reliable trends in 3D ensemble fields. In this paper we present bidirectional linking of multi-charts and volume visualization as a means to analyze visually 3D scalar ensemble fields at the data level. Multi-charts are an extension of conventional bar and line charts: They linearize the 3D data points along a space-filling curve and draw them as multiple charts in the same plot area. The bar charts encode statistical information on ensemble members, such as histograms and probability densities, and line charts are overlaid to allow comparing members against the ensemble. Alternative linearizations based on histogram similarities or ensemble variation allow clustering of spatial locations depending on data distribution. Multi-charts organize the data at multiple scales to quickly provide overviews and enable users to select regions exhibiting interesting behavior interactively. They are further put into a spatial context by allowing the user to brush or query value intervals and specific distributions, and to simultaneously visualize the corresponding spatial points via volume rendering. By providing a picking mechanism in 3D and instantly highlighting the corresponding data points in the chart, the user can go back and forth between the abstract and the 3D view to focus the analysis.

**Index Terms**—Ensemble visualization, brushing and linking, statistical analysis

## 1 INTRODUCTION

Volume rendering is a standard method to visually convey the information content in 3D scalar fields. Especially when transfer functions in combination with interactive approaches are used, volume render-

ing allows effectively revealing the location and shape of relevant features, as well as their spatial relationships. Volume rendering, on the other hand, faces the problem of spatial occlusions and attenuation effects. This limits the number of features that can be visualized simultaneously, and it lowers the effectiveness of volume rendering without prior knowledge of features and their locations in space. In particular when volume rendering is used to explore multi-variate fields, requiring quantitative comparisons across multiple parameters, these limitations are further increased.

A solution to remedy this situation is to combine techniques from information visualization and volume rendering via brushing and linking [9, 6, 39, 56]. Such techniques first generate an overview of the data via attribute views, and brushing functionality in these views is then used to select data values meeting certain criteria. The attribute

- I. Demir, C. Dick and R. Westermann are with the Computer Graphics and Visualization Group, Technische Universität München, Informatik 15, Boltzmannstraße 3, 85748 Garching, Germany. E-Mail: ismail.demir@mytum.de, {dick,westermann}@tum.de

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014. Date of publication 11 Aug. 2014; date of current version 9 Nov. 2014.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

Digital Object Identifier 10.1109/TVCG.2014.2346448

views are linked to a 3D view generated via volume rendering, so that any selection in attribute space triggers an immediate visual feedback on the respective spatial locations. Popular attribute space representations are scatterplots, histograms, or parallel coordinates.

In this work we shed light on the use of brushing and linked volume rendering for visualizing 3D scalar ensemble fields  $f: \mathbb{R}^3 \times \{1, \dots, m\} \rightarrow \mathbb{R}$ . Ensembles are usually generated by  $m \in \mathbb{N}$  repeated simulations of one or multiple parameterized models using slightly modified parameter settings. Each simulation run, or ensemble member, captures the values of the same attribute at a set of locations in 3D space. In particular for large numbers of ensemble members, i.e., 40,000 in one of our applications, treating ensembles as multi-variate data with  $m$  parameters per spatial point and using visualization techniques for such data is unfeasible.

Our goal is to quickly guide the user towards locations where the data values and their distributions are similar or differ significantly over the ensemble members, and to further enable a quantitative comparison of a specific ensemble member to all others. In principle this task can be performed by computing confidence intervals at each point in space and visualizing these intervals via volume rendering. Yet due to the limitations of volume rendering it becomes difficult to visually resolve the relevant information in the generated views. Another well researched approach is to condense the displayed information by focusing on particular features in the data [35, 4], like isosurfaces, and show their possible spatial variations over the ensemble. Recently, the comparative visualization of the data variations at selected locations in space and time has shown a promising strategy for 2D ensembles [18, 19].

## 1.1 Contribution

In this work we present an alternative visualization technique for 3D ensemble fields when prior knowledge about relevant features and their spatial occurrences is not available. In this situation a global comparative visualization of the ensemble at the data level is required, and the use of a fully automated analysis technique is prohibited in general, due to the uncertainty about what we are looking for.

To address these aspects, we suggest a visual interface for analyzing statistical properties of 3D ensemble fields effectively and, thus, to enable the user to guide the analysis process towards relevant characteristics. Our approach builds upon a new volume representation, which provides an abstract view in which the spatial locations and the variations of the data values at these locations over the ensemble members can be distinguished simultaneously. The abstract view is generated via a linear layout of 3D space and by using multiple stacked and combined bar and line charts, so called *multi-charts*, to visualize the data. The bar charts encode statistical ensemble summaries in sub-areas of the 3D domain, and the line charts depict the variations of specific ensemble members. The user can interactively brush the multi-chart to select regions exhibiting certain statistical properties. Linked 3D views provide an immediate visual feedback to these operations. To further enable a visual analysis of the relationships between the data variations at different locations, we utilize similarity-based sorting and clustering techniques on these variations.

The specific features of our approach are:

- A 1D visual level-of-detail volume representation using a Hilbert curve linearization of space.
- The simultaneous use of bar charts to show statistical summaries and line charts to emphasize specific ensemble members.
- The combined use of automated statistical data mining techniques and visually guided user interaction to detect and analyze relevant data characteristics.

The remainder of this paper is organized as follows. Next, we review work that is related to ours. We then give an overview of our approach, including a description of the basic functionality it provides. Next, we introduce multi-charts, and we describe their principal layout as well as the kind of statistical information they encode. This

is followed by discussion of the similarity-based arrangement of the chart, and the bidirectional linking we propose to enable 3D spatial views of the selected data points. We then demonstrate applications of multi-charts in a number of real-world examples from meteorology, fluid simulation, and astrophysics. We conclude the paper with some remarks on future work in the field.

## 2 RELATED WORK

Our approach belongs to the category of summary-based ensemble visualization techniques, which compare sample properties at fixed locations and also investigate the relationships between the samples at different locations in the dataset. It uses descriptive statistics for data analysis, taking into account the frequency of occurrence of data values over multi-dimensional intervals, as well as their correlations. Besides ensemble visualization, our technique has overlap with techniques in the categories brushing and linking, and diagram techniques.

**Ensemble Visualization:** Ensemble visualization can be seen as a sub-area in the broader field of uncertainty visualization, where the data uncertainty is given by a set of possible occurrences of the data. The importance of uncertainty visualization has been acknowledged more than a decade ago [35], and since then a number of overviews and taxonomies of uncertainty visualization techniques have been published [22, 33, 50, 10, 42, 16]. The uncertainty information in ensemble fields has often been exposed by visualizing quantities such as mean and standard deviation via color maps, opacity, texture, animation, and glyphs [55, 4, 44, 32]. Feature-based approaches have been used to derive from 3D ensembles the effect of the uncertainty on the position and structure of particular features in the data, such as iso-surfaces, and to visualize these effects via confidence envelopes [35, 57, 40, 38], geometric displacements [11], or surface animation [3].

Potter et al. [43] have demonstrated the combination of linked statistical visualization techniques and user interaction for 2D weather ensembles. Thompson et al. [49] proposed a volume rendering technique showing statistical summaries at different resolution levels in 3D space. In meteorology and geoscience, so called spaghetti plots are widely used to show simultaneously all iso-contours in the members of a scalar ensemble. Sanyal et al. [46] enhanced spaghetti plots by glyphs and confidence ribbons to highlight the Euclidean spread of contour ensembles. Recently, Höllt and co-workers [18, 19] have demonstrated the use of time-series glyphs for the comparative visualization of 2D ensemble values at two different positions over time.

**Brushing and Linking:** Brushing and linking is based on the concept of using two or multiple different views on the same data points to emphasize different aspects in the data [1, 48, 53, 26]. Furthermore, these views can be linked to each other, so that selections in one view—performed via any kind of brushing functionality—are directly applied in the other views. Variations and extension of this concept have been proposed with respect to alternative attribute views in which brushing is performed as well as improved selection strategies. Popular attribute space representations are multi-dimensional scatterplots [39], (time-dependent) histograms [29], or parallel coordinates [20, 15]. The combination of brushing and linking with focus+context techniques has been proposed in [14, 5]. For analyzing (time-varying) multi-variate 3D fields, brushing and linked 3D views has been proposed [8, 9, 6, 24, 56, 23, 25]. In volume rendering applications, a number of multiple view approaches have been used for advanced structure exploration and navigation, for instance, in [51, 52, 12], to name just a few.

**Diagram Techniques:** Diagram based techniques like bar, box or line charts, like the graphical primitives we use in this work, are frequently employed in information visualization to analyze large sets of categorical and numerical data. Bar charts are widely used for summarizing categorical data, and they are typically applied to show aggregated values for the categories they present. To also allow bar charts to present detailed information of the categories, pixels within the bars have been used to further encode this information [27, 28]. Box plots [7], on the other hand, use bars as graphical elements, but they are augmented by additional visual cues to display the data values

as well as important characteristics of the data, such as spread or quartile. Potter et al. [41] have proposed an extension of box plots to also show statistical data summaries. They have also made use of the concept of violin plots [17] to reveal data distributions via the shape of the plotted primitives. Recently, Whitaker et al. [54] introduces contour boxplots for improved visualization of uncertainty in spaghetti plots of isocontours or level sets, taking into account the ordering of multivariate data. A two-sided violin glyph was introduced by Höllt et al. [19] to show the data distributions at two different locations in space. Another diagram based technique are voxel bars [34], an extension of pixel bar charts, where groups of 3D voxels are packed into graphical primitives in a 2D coordinate system, and structures and relationships are emphasized via specific bar colorings.

### 3 OVERVIEW

Our method starts with an ensemble of scalar data sets given on a fixed 3D grid structure. We restrict to 3D Cartesian grids, even though arbitrary grids can be handled in much the same way as described. The data is first structured in an hierarchical manner, computing at each node statistical information of the data values this node represents (see Section 5). The visualization is then performed according to the following concepts and principles:

**Multi-Charts:** First an abstract volume representation is constructed. It is based on the idea to view a 3D ensemble data set by linearizing the data locations and visualizing the data variations over the ensemble members via bar and line charts, so called multi-charts. The data values are drawn as a dense sequence of primitives over a 1D domain, and multiple such sequences are shown at once depending on the size of the plot area. Figure 1 shows the basic layout of a multi-chart in the plot area.

**Multiresolution Ensemble Summaries:** Every bar encodes aggregate (statistical) information about the data variations across multiple ensemble members in a certain spatial area, and every line speaks for a particular member and allows comparing this member against all others (see Figure 1). Since the viewport into which the chart is drawn has a fixed resolution, only a limited number of locations can be visualized at once. Therefore, overviews taken from a multi-resolution representation of the 3D ensemble are shown first. In line with the information seeking mantra “Overview first, zoom and filter, then details-on-demand” proposed by Shneiderman [47], the user can then zoom into the data to guide the analysis towards those regions showing interesting behaviors.

**Brushing and Querying:** To enable a data-driven analysis, the user can select regions in which the data values meet certain criteria interactively, and these regions are instantly highlighted in the 3D view. Selecting is performed by brushing or querying in the chart domain, as outlined in Figure 1 by the yellow shaded region. To ease the selection process, regions, i.e., their corresponding bars, can be sorted automatically according to increasing/decreasing values of derived statistical measures. The selection can also be manipulated in the 3D view, with the multi-chart view being synchronized accordingly.

**Spatial Clustering and Correlations:** When analyzing ensembles, one further important operation is clustering of regions which show similar value distributions over the ensemble members. Such operations can help domain experts obtain insight into the sensitivity of the simulation processes to the kind of parametrization, and they are especially important to detect short- and long-range correlations in the data. To enable this, we provide automatic cluster mechanisms based on the similarity of value distributions (or local histograms), and we arrange bars according to the data similarity in the corresponding regions automatically. Spatial clustering is demonstrated in Figure 8.

We now describe the different components of the ensemble visualization technique and their realization. Throughout the following discussion the main focus is laid on an interactive visual exploration of scalar 3D ensembles, to locate regions in which the data distributions exhibit certain properties. Since locating and picking regions via 3D volume rendering is difficult, we ease the navigation in 3D space via a slice-based visualization of the 3D domain, i.e., the volume is decomposed into a set of slices having a width of one voxel *block* (see

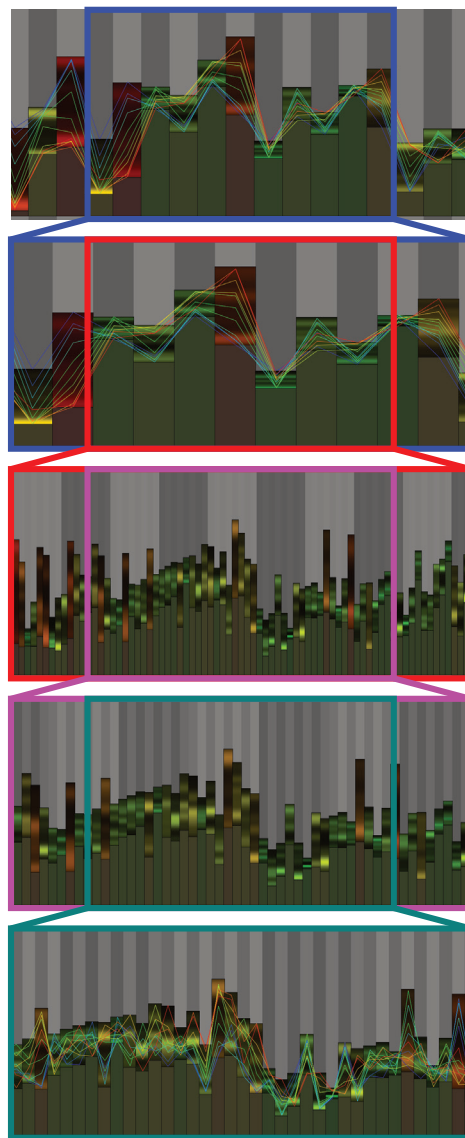


Fig. 2. From top to bottom: Zooming into the multi-chart. The bars are automatically refined when their width exceeds a certain threshold. To improve visual coherence, a vertically striped background pattern is employed to illustrate set of refined bars.

Figure 1). In this visualization, spatial locations and relationships can be perceived far more effectively, and interesting regions can be picked in an easy way.

### 4 MULTI-CHARTS

We assume that the ensemble data set is given on a 3D voxel grid. To provide multi-charts at multiple levels of detail, we partition the voxel grid into approximately equally-sized voxel blocks consisting of roughly  $2^\ell \times 2^\ell \times 2^\ell$  voxels, where the level  $\ell$  is selected by the user.  $\ell = 0$  denotes the finest level, where each voxel block consists of a single voxel. The multi-chart is then constructed at the granularity of voxel blocks, i.e., each voxel block is represented by a single bar, or by a single point which is connected to its neighboring points via line segments. Bar and line charts can be displayed together.

Our multi-chart visualization method is based on linearizing the 3D grid of voxel blocks. We have identified two particular design goals for this linearization: First, the linearization should preserve spatial relationships as good as possible, in particular, linearization-adjacent voxel blocks should also be adjacent in 3D space. Second, the lin-

earization should be consistent over multiple levels of detail, i.e., the eight voxel blocks that constitute a single voxel block at the next coarser level should be adjacent within the linearization. This ensures visual coherence when the user switches between two different levels of detail, in that a bar is decomposed into eight adjacent bars when switching to the next finer level, or eight adjacent bars are merged into a single bar when switching to the next coarser level.

To construct a linearization with these properties, we employ a space filling curve. In particular, the Hilbert curve is known to preserve spatial relationships well, and due to its octree-based hierarchical construction principle, leads to the desired consistency over multiple levels of detail. Since the standard Hilbert curve is defined on grids with the same power-of-two extent in each dimension, we use a generalized Hilbert curve [13] that supports unequal power-of-two extents in each dimension. To allow for the application of our method to a voxel grid of arbitrary size  $n_1 \times n_2 \times n_3$ , we proceed as follows: Let  $k_i := \lfloor \log_2(n_i) \rfloor$ , such that  $2^{k_i}$  is the largest power-of-two less or equal than  $n_i$  ( $i = 1, 2, 3$ ). Then, at level  $\ell \in \{0, 1, \dots, \min\{k_1, k_2, k_3\}\}$ , the voxel grid is decomposed into  $2^{k_1-\ell} \times 2^{k_2-\ell} \times 2^{k_3-\ell}$  voxel blocks. The voxel block grid thus has power-of-two extents, allowing for the construction of a generalized Hilbert curve according to [13]. Let blocks and voxels be identified by 3D integer indices, counting from zero in each dimension. Voxel block  $(\bar{x}_1, \bar{x}_2, \bar{x}_3)$  then consists of all voxels  $(x_1, x_2, x_3)$  satisfying

$$\left\lfloor \bar{x}_i \cdot 2^\ell \cdot \frac{n_i}{2^{k_i}} \right\rfloor \leq x_i < \left\lfloor (\bar{x}_i + 1) \cdot 2^\ell \cdot \frac{n_i}{2^{k_i}} \right\rfloor \quad (i = 1, 2, 3).$$

The resulting voxel blocks are almost of equal size (along each direction, their size varies by at most one voxel), and exhibit compact shape (the ratio of the largest to the smallest edge is at most two). Per construction, subsets of  $2 \times 2 \times 2$  voxel blocks constitute a single block at the next coarser level. At level  $\ell = 0$ , the blocks consist of between one and eight voxels. To allow the user to explore the data set also at voxel granularity, an additional level  $\ell = -1$  is added, where voxel blocks consist of individual voxels. The linearization at this level is derived from the linearization at level  $\ell = 0$ , with each voxel block being further linearized in lexicographical order.

The multi-chart then is generated by displaying bars and line segments corresponding to voxel blocks in the order imposed by the described linearization. To fully utilize the size of the screen, the chart is displayed in a set of multiple rows. At the beginning of the visual exploration process, the number of rows is set to eight, and we automatically select a level of detail such that the number of bars per row is at most 256.

Considering the navigation within the chart, we provide zooming and panning by operating the mouse. Zooming can be performed independently for each axis, or simultaneously for both axes. To simplify navigation in the chart, the current position and extent of the viewport are displayed in an overview window (see Figure 1). The user can prescribe a specific level of detail, or use automatic level of detail selection coupled to horizontal zooming. The latter is implemented by automatically switching to the next finer (coarser) level of detail, whenever the bar pixel width exceeds (falls below) a certain threshold (we use a threshold of 64 and 8 pixels, respectively). To achieve visual coherence, the width of the bars is decreased (increased) by a factor of eight, when switching to the next finer (coarser) level, such that a bar and its associated set of refined bars occupy the same position. The bars are displayed using a uniform width, except for level  $\ell = -1$ , where we have to use a variable width in order to compensate that the bars at level  $\ell = 0$  are decomposed into a variable number of bars.

To further improve the visual coherence when switching between levels of detail during zooming, we employ a background pattern based on vertical stripes that are aligned with the bars (see Figure 2). The stripes are colored alternately using gray at two different intensities, and are used to indicate the sets of refined bars resulting from the decomposition of the bars at the previous coarser level. Consider the case of zooming into the chart (zooming out is similar). When switching to the next finer level, the stripes initially remain unmodified, such that the sets of refined bars become visible. Only when further zoom-

ing into the chart, the stripes are also refined using a gradual transition. This transition is completed before switching to the next finer lever again, such that the process restarts from the beginning.

While in our current implementation we only consider data sets on 3D voxels grids, in principle our method can also be applied to unstructured grids. This can be achieved by using a hierarchy of rectangular grids to partition the 3D domain at multiple levels of details, such that vertices of the unstructured grid lying in the same grid cell are assigned to the same subset. Our method then is applied to these subsets, i.e., the subsets replace the voxel blocks in the previous description. This construction principle also works when the unstructured grid is time-dependent, since the rectangular grid remains fixed.

## 5 MULTIREOLUTION ENSEMBLE SUMMARIES

To guide the visual exploration of the ensemble data set, we provide two different types of multi-charts.

Our first type of multi-chart (see Figure 3, top) is devoted to analyzing how uncertainty is quantitatively distributed over the 3D domain. For each voxel of the data set, we compute the range (i.e., the difference between the maximum and the minimum value) over the set of ensemble members, and use this value to quantify the level of uncertainty at that voxel. In order to reduce the influence of outliers, the quantile range between the  $p$  and  $(1-p)$  quantiles can be used alternatively ( $p \in [0, 0.5]$ ). Then, we depict the distribution of the range values within each voxel block by means of a histogram encoded into a bar as follows: The histogram is determined by means of kernel density estimation [45, 36], using 128 bins and Gaussian kernels. The vertical extent of the bar reaches from zero to the maximum of the range values in the considered block. The histogram is visualized by using the histogram density values (scaled by a user-specified factor) as the color intensity along the vertical range of the bar. To indicate an overflow of the color intensity, a special color (blue) is used. Between zero and the minimum range value, a constant color intensity of 0.25 is used. In this way, the minimum and maximum range values become clearly visible.

It is worth noting that by showing the distribution of range values (rather than for example only depicting their average or maximum), each voxel is accurately represented at all levels of detail, and the method is robust in the context of outliers. To support the user in rapidly identifying regions with low and high uncertainty, we additionally color-code the range value by means of a linear green-red color scale, where green and red denote low and high uncertainty, respectively. Note that this color-coding is redundant, in that the range value is already determined by the distance from the zero line.

In addition to the range value, it is also possible to visualize per-voxel-block histograms of other statistical quantities, such as the mean or the standard deviation when analyzing ensemble data sets that are known to follow a Gaussian distribution.

While the first type of multi-chart can effectively reveal the distribution of uncertainty over the 3D domain, the second type (see Figure 3, bottom) is designed for providing a detailed insight into the particular distribution of the ensemble members, as well as studying the relationships among a subset of the members, or between a member and the entire ensemble. For each ensemble member, we first compute the average of the values as well as the difference (here referred to as delta value) between the maximum and the minimum value within each voxel block. The delta value describes the spatial variance of each ensemble member in the considered block. Our visualization approach then is based on representing each ensemble member by its average value within each voxel block. Clearly, using average values can lead to misinterpretations if the spatial variations are too high. In this case, it is necessary to perform the analysis at a spatially higher resolution.

We then visualize the distribution of the ensemble members by plotting the distribution of the corresponding average values within each voxel block by means of a histogram encoded into a bar. We proceed in the same way as for the first type of multi-chart, i.e., we again construct the histogram via kernel density estimation, which is then depicted by means of adapting the color intensity along the vertical

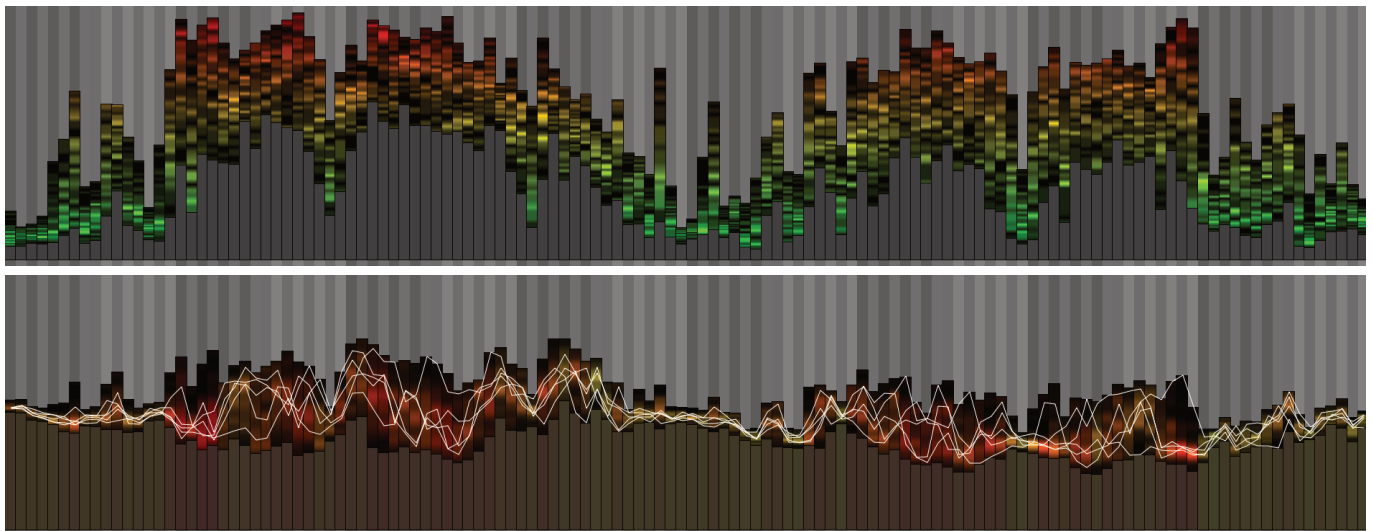


Fig. 3. Two different types of multi-charts are provided. The bars are encoding histograms, either showing per-voxel statistical quantities (here: the range over all ensemble members) within the subregion associated with each bar (top), or the distribution of the ensemble members, based on the average of each ensemble member within each subregion (bottom). In the latter type of multi-chart, a user-selected subset of the ensemble members can be additionally displayed via horizontal line segments or polylines. In the top and bottom image, corresponding bars show the same subregion.

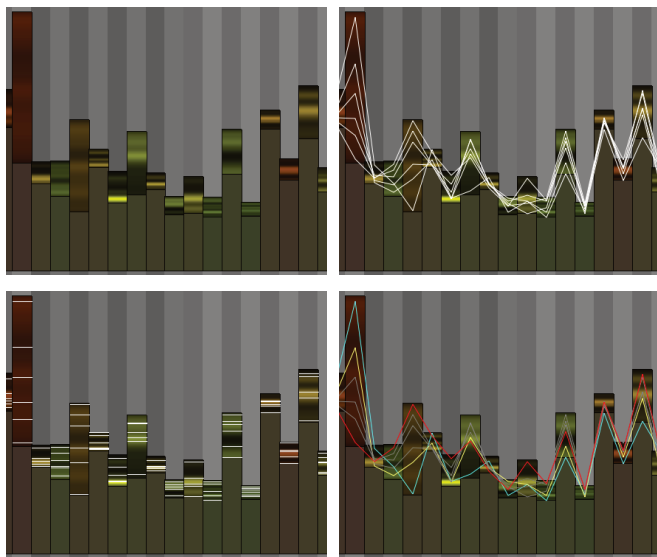


Fig. 4. Visualization of the distribution of the ensemble members. Top left: Histogram only. Top right: Histogram and polylines. Bottom left: Histogram and horizontal line segments. Bottom right: Polylines with color-coding of individual members.

range of the bar. To incorporate the delta values, for each block, we compute the maximum delta value over the set of ensemble members, and depict this maximum by means of color-coding of the entire bar using a linear green-red color scale. Using green and red for low and high delta values, respectively, the red color indicates that it is necessary to switch to a finer level of detail, in order to perform an accurate data analysis in the respective subregion associated with the bar.

In addition to showing the distribution of the ensemble members via bars, we provide the possibility to visualize the ensemble members—or a user-specified subset—via lines, which are laid over the bars. We have implemented two variants (see Figure 4): Depicting ensemble member using horizontal line segments, or using polylines. Optionally, color-coding of the lines can be used to support tracking of indi-

vidual ensemble members or groups of ensemble members throughout the diagram. Colors can be assigned automatically by member id using a discrete color table, or manually by mouse operation. We further provide the option to encode the delta value using a linear blue-red color scale, where blue and red denote low and high delta values. It can be seen that the polyline-based approach facilitates tracking of individual ensemble members throughout the multi-chart. In contrast, it is almost impossible to track individual members in the visualization based on horizontal line segments. However, the latter approach can be employed as an alternative to histograms for visualizing the distribution of the ensemble members, when the ensemble consists of only a few members. A drawback of the polyline-based approach is that line segments can have a very high slope, when the bar width is small, leading to visual cluttering. We address this issue by automatically hiding the polylines when the bar width is less than 16 pixels.

To facilitate the interactive exploration of ensemble data sets with a large number of ensemble members, the ensemble summaries are pre-computed and stored on disk. In particular, for each voxel block (on all levels), we precompute the histogram of the per-voxel ranges and the histogram of the per-member averages, as well as statistical quantities such as the minimum/maximum/mean of the per-voxel ranges, the minimum/maximum/mean/standard deviation of the per-member averages, and the maximum of the delta values. In total, the ensemble summaries require about 1 kB per voxel block (using 128 bins for the histograms).

Depicting individual ensemble members requires the per-member average values within each voxel block. These values are stored in a separate data file, consecutively for each voxel block. Note that on the finest level, where each voxel block consists of a single voxel, the initial ensemble data set is stored.

The ensemble summaries are completely loaded into main memory at the beginning of the interactive visualization session. This is required to enable the sorting of voxels blocks according to specific criteria, as described in the following section. It is worth noting that the memory size of the ensemble summaries is independent from the number of ensemble members, but depends on the spatial resolution of the data set. For very high-resolution data sets, a viable option to reduce memory requirements is to skip pre-loading of histograms for the very finest levels (which however disables sorting of voxel blocks according to histograms on these levels). In contrast, the per-member average values are fetched from disk on demand, i.e., only data values

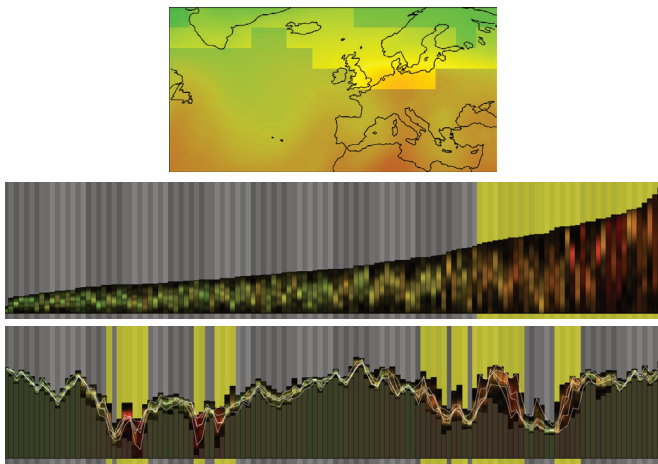


Fig. 5. Sorting of the bars to support selection of regions with similar characteristics: After sorting of the bars, the user brushes into the multi-chart (middle). The selection remains active after the bars have been rearranged into their original order (bottom). The respective selection is also shown in the 3D view (top). Here and in the following figures, only a single slice of the 3D volume is shown for illustration purposes.

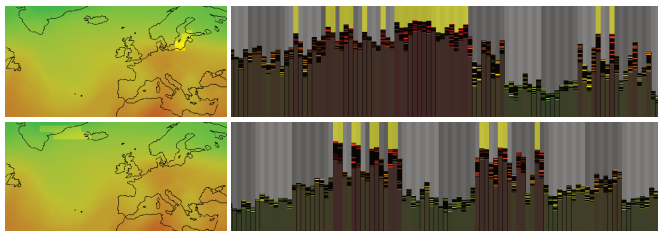


Fig. 6. Top and bottom row: After zooming into two of the regions determined in Figure 5, visualizing histograms of per-voxel ranges reveals areas of particular high uncertainty.

that are currently visible in the viewport are loaded into memory. To hide disk access latencies, a prefetching scheme based on rectangular prefetching regions around the viewport (one per level of detail) has been integrated.

### 6 BRUSHING AND QUERYING

The user can brush arbitrary bars interactively in the multi-chart view, and the spatial regions corresponding to the selected bars are instantly highlighted in the 3D view. Brushed bars are highlighted by a yellow background as shown in Figure 5 for the ECMWF temperature ensemble forecast, introduced in Figure 1.

In addition, we provide the option to sort the bars with respect to different statistical measures, such as mean value and standard deviation of the average values per ensemble member in each region, or the maximum/minimum/mean of the per-voxel statistics in each region. Sorting with respect to histogram similarity will be described in the next section. By brushing in the sorted representation, the user can select all relevant spatial regions in an easy way. This is shown in Figure 5 (middle), where the spatial regions exhibiting highest difference between any two ensemble members are selected. Note that in this chart the bar height encodes only the range (difference between maximum and minimum), rather than the maximum over all ensemble members. In the 3D view the selected regions are highlighted instantly.

When switching back to the order given by the Hilbert curve, as shown in Figure 5 (bottom) (now the bar height encodes the maximum), the brushed selection remains active (see yellow background), and, in the current example, the selected regions distribute on roughly four clusters in the abstract multi-chart view. Due to the good preser-

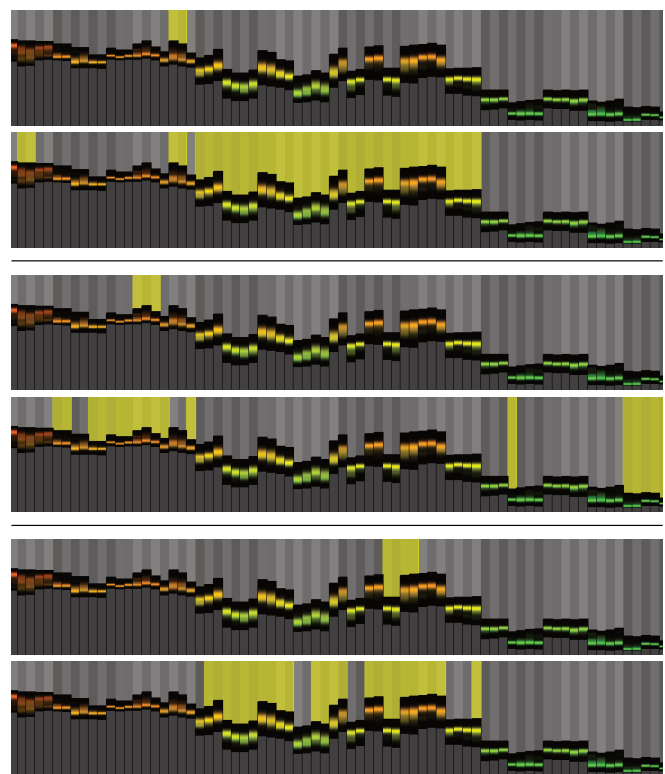


Fig. 7. Selection of bars by querying operations. Each pair of images, from top to bottom: The user selects some bars (top), and asks for all other bars having a range that is higher than the minimum / lower than the maximum / within the interval given by the minimum and maximum of the ranges of the selected bars (bottom).

vation of locality of the Hilbert curve order, these cluster correspond to spatial clusters, as shown in Figure 5 (top). One can clearly see that a higher degree of uncertainty occurs in regions exhibiting lower mean values. Interestingly, although not shown here, this property holds true for the whole domain.

When relating each cluster in the multi-chart view with its corresponding geographical area on the map, one finds that the third cluster from the left roughly embodies the region containing the North Sea and Baltic Sea. To further analyze the selected regions we visualize the per-voxel range and zoom into the third (Figure 6, top row) as well as first and second (Figure 6, bottom row) clusters. Since the delta values are relatively high for these regions, as indicated by the bars' red color, we switch to a chart depicting histograms of per-voxel ranges. We then select regions where the uncertainty is particular high. This guides us to a region containing two larger islands in the Baltic Sea, and two regions in the coastal area between Greenland and Iceland.

To select regions based on specific data values or statistical measures, the user can also use select queries. This is realized by using expressions, such as *Min* and *Max* on all the different statistical measures. *Min* and *Max* queries give all bars and corresponding spatial regions where the measures are respectively below and above a specified value. Range queries allow selecting regions exhibiting measures in specific intervals. Furthermore, the user can select particular members and let them highlight as a line chart in the multi-chart view. Some typical query operations are shown in Figure 7.

### 7 SPATIAL CLUSTERING AND CORRELATIONS

Up to now we have presented visual analysis techniques for ensembles which are based on the analysis of statistical measures of the data in selected spatial regions. Another important requirement is to analyze the (relative) behavior of the data values among selected regions of interest or different ensemble members. The rationale behind this

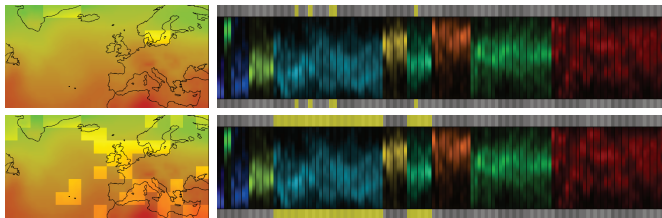


Fig. 8. Selection based on clustering of bars according to histogram similarity. After brushing into the 3D view (top left), the selected regions fall into two clusters (top right). By selecting the remaining bars from the two clusters that contain initially selected bars (bottom right), regions with similar distributions are identified (bottom left).

is to identify regions showing similar data distributions and variations over the ensemble members, and to reveal dependencies between the data values in different regions. To achieve this, we have integrated techniques for grouping of data histograms, and we consider the correlations between the data values in different spatial regions to analyze their behavior relative to each other.

### 7.1 Histogram Clustering

Per-region histograms are computed as described in Section 5, and clustered based on their similarity using the  $k$ -means algorithm [31]. The similarity of two histograms  $h_1, h_2 : \{1, \dots, n\} \rightarrow \mathbb{R}_0^+$  with  $n \in \mathbb{N}$  bins is computed by their mean squared distance as

$$\frac{1}{n} \sum_{i=1}^n (h_1(i) - h_2(i))^2.$$

The user can either select the number of clusters to be generated, or she can select a similarity threshold so that only histograms having higher similarity than indicated by this threshold are grouped.

To compute an initial set of  $k$  means to start with by the  $k$ -means algorithm, we proceed as follows: We begin with an empty set of clusters and a fixed similarity threshold, and we then iterate over all histograms. For every histogram, the cluster with the least difference between the considered histogram and the respectively first histogram assigned to that cluster is identified. If the difference is smaller than the threshold, the histogram is assigned to that cluster. Otherwise, or if no cluster exists at all, a new cluster is generated and the histogram is assigned to it. This step is repeated until all histograms have been assigned. If the number of generated clusters is greater (less) than the requested number of clusters, we divide (multiply) the threshold by 2 and start clustering all over again. This is repeated until the number of clusters has crossed the desired number. The respectively first histograms that were assigned to each cluster are used as initial cluster centroids in the upcoming  $k$ -means algorithm.

The benefits of our clustering approach are demonstrated in Figure 8. In our example, 10 clusters have been determined. To make the visualization easier to understand, a unique color is assigned to each cluster, and all bars are scaled to a uniform height. The user has selected some regions within the 3D view (Figure 8, top left). It can be observed that the bars fall into two different clusters (Figure 8, top right). If one picks all bars in these clusters (Figure 8, bottom right), one obtains a remarkable result which is depicted in Figure 8 (bottom left). It can be clearly seen that mostly coastal areas have been selected, meaning that these regions feature similar distributions of the ensemble members.

### 7.2 Correlations

Besides a statistical analysis focusing purely on the data variations in certain regions, an important task is to find regions in which the data values show certain correlations to each other. While in principle local correlations can be visualized via glyph-based approaches, e.g., as proposed in [37], visualizing long-range interdependencies in 3D is challenging. It is, on the other hand, especially the existence of such

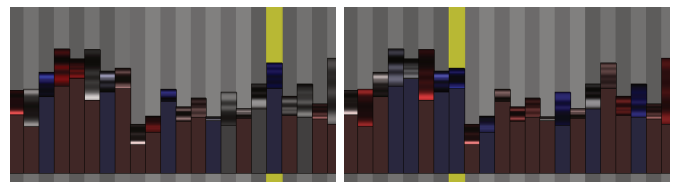


Fig. 9. Visualization of the correlations between the subregions associated with the bars based on per-member average values. Left and right image, respectively: After picking a single bar (highlighted in yellow), the correlation between this subregion and each other subregion in the viewport is computed and visualized by color-coding of the bars.

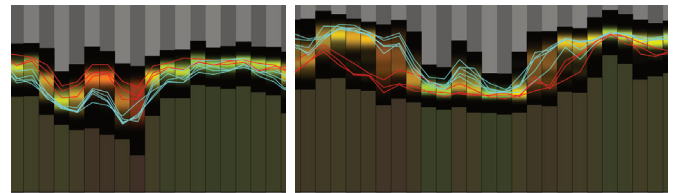


Fig. 10. Two areas (left and right image, respectively) that were identified as being inversely correlated. To demonstrate the inverse correlation, several ensemble members are depicted by means of colored polylines. Whereas red members have higher values than blue members in the left area, the situation is vice versa in the right area.

long-range interdependencies which are important in a number of applications. For instance, in meteorology local features are often dependent on distant phenomena. To analyze such dependencies, we have built into our tool means to compute a) the correlations between the average member values in the region represented by one selected bar and all other regions represented by bars in the current view, and b) the correlation between the variation of the data values in two user-selected sub-regions over all ensembles. In the former case, the correlations are instantly visualized by mapping the correlation values to the colors of the involved bars, as depicted in Figure 9, using a red-white-blue color scale, where red/white/blue denote negative/no/positive correlation, respectively. Furthermore, we use the correlation as a distance metric for computing clusters of regions in which the data values are strongly correlated using the  $k$ -means algorithm. Here we exploit the fact that for correlations sufficiently close to 1, the correlation property is transitive, thus making the metric an equivalence relation. Regions not strongly correlated will fall apart into single-element clusters.

Correlation computation always yields a numerical value in the interval  $[-1, 1]$ , serving as a measure of the strength of the relationship between the values in pairs of regions. In particular we make use of the Pearson's correlation coefficient to measure the strength of a linear association. Thus, negative and positive values indicate an inverse and positive correlation. To generate robust results even under weak assumptions with regard to the distribution, we utilize a variant of the quadrant correlation coefficient as proposed by Blomqvist [2]. We have modified this algorithm slightly, since it relies on the signum function which turned out to produce inconclusive results in many of our cases. Therefore we make use of a weakened signum function as described below.

Let  $I = \{1, \dots, m\}$  denote an index set of  $m \in \mathbb{N}$  ensemble members and let  $A, B$  denote two distinct spatial areas represented by bars in our visualization. Also let us denote the value distribution of the ensemble with respect to areas  $A$  and  $B$  by  $h_A : I \rightarrow \mathbb{R}^{|A|}, h_B : I \rightarrow \mathbb{R}^{|B|}$ . Then we obtain the modified quadrant correlation coefficient as

$$\frac{1}{m} \sum_{i=1}^m \text{wsgn}(h_A(i) - \tilde{h}_A) \cdot \text{wsgn}(h_B(i) - \tilde{h}_B).$$

Here  $\tilde{h}_A, \tilde{h}_B$  describe the componentwise median of  $h_A, h_B$ , and we

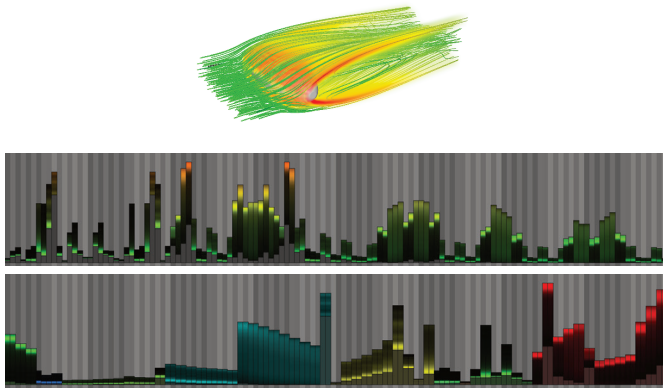


Fig. 11. Ensemble data set for an incompressible 3D Navier-Stokes fluid simulation. Top: A single member is visualized by streamlines and volume rendering. Middle: Multi-chart, showing the distribution of the ensemble members via histograms. Bottom: Bars are clustered according to histogram similarity.

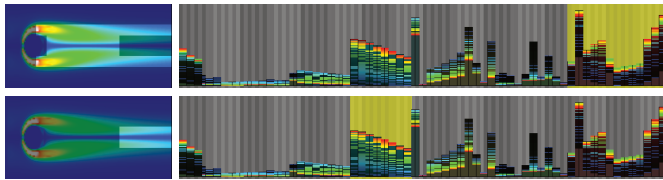


Fig. 12. Top row: Selection of the red cluster from Figure 11 via brushing. The selected region is highlighted in the 3D view. Bottom row: Selecting the emphasized region in the 3D view yields a part of the cyan cluster. In both rows, the individual ensemble members are additionally depicted by horizontal line segments, which are color-coded according to ensemble member id.

define  $wsgn : \mathbb{R}^k \rightarrow [-1, 1]$ ,  $k \in \mathbb{N}$  as the weakened signum function

$$wsgn(x) = \begin{cases} 1 - neg(x)/pos(x) & \text{if } pos(x) > neg(x) \\ -1 + pos(x)/neg(x) & \text{if } neg(x) > pos(x), \\ 0 & \text{else} \end{cases}$$

where,  $pos(x)$  and  $neg(x)$  denote the number of positive and negative entries of the vector  $x$ , respectively.

To demonstrate the kind of information that can be revealed by the proposed correlation visualization, let us take another look at the temperature forecast ensemble introduced in Figure 1. Here we focus on the region containing Greenland (first block of selected bars from the left in Figure 5, bottom) and the region containing the North Sea and Baltic Sea (part of the third block of selected bars). To study the interdependencies between the ensemble members in these regions, the user selects these regions and asks for the correlation coefficient. In the current example, a correlation coefficient of  $-0.45$  is computed, meaning that the distributions in both regions are inversely correlated. To illustrate this result, in Figure 10 a few representative ensemble members are shown.

## 8 FURTHER RESULTS

In addition to the ECMWF temperature ensemble we used throughout the paper to demonstrate the effects of our ensemble visualization technique, we now present further results on the basis of additional scalar ensembles. We begin with an ensemble featuring an incompressible fluid flow evolving around an ellipsoid obstacle, which was numerically simulated on a  $145 \times 49 \times 49$  Cartesian grid using the Navier-Stokes equations. 56 simulation runs were performed using slightly different viscosities. The vorticity magnitude produced by each simulation run after the same simulation time was written out as the scalar

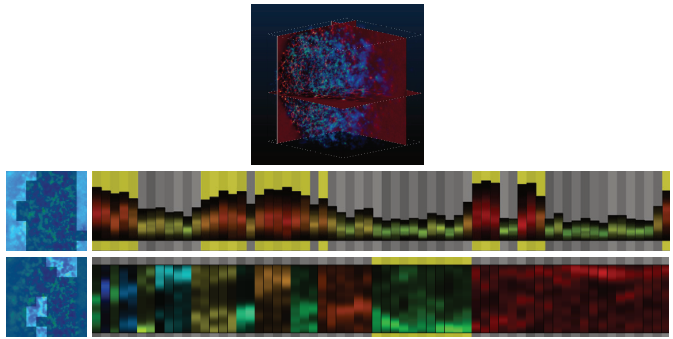


Fig. 13. Application of our multi-chart visualization method to study a large-scale data set, consisting of 40,000 possible maps of the universe. Top: Volume rendering of a single ensemble member. Image courtesy of Jasche et al. [21]. Middle row: Multi-chart showing the distribution of the ensemble members. By selecting bars corresponding to areas with high uncertainty, it can be observed that these areas are primarily located in outer regions of the domain. Bottom row: After clustering the bars according to histogram similarity, a cluster is selected where the ensemble members have low values (here: density), yielding two distinct regions.

ensemble field. A visualization of a single time step is shown in Figure 11 (top).

When visualizing the ensemble, we start with an overview first to isolate regions where the data values show an outlier behavior compared to other regions. The overview is shown in Figure 11 (middle), where the distribution of the ensemble members is shown in a multi-chart. It can be clearly seen that there are regions where the vorticity varies significantly over the ensemble members, whereas in other regions it changes only to a very small extent.

Next we study the distribution of vorticity among regions corresponding to the bars and try to discover striking spatial relations between these distributions. For this we order the bars in such a way that bars with similar histograms are clustered. To make it easier for the user to visually recognize this order a unique color is assigned to the bars of each cluster. The result is shown in Figure 11 (bottom). We can now analyze common features of specific clusters. For instance, the cluster colored red features a peak at higher values, meaning that vorticity in this region is high throughout most of the simulations. By looking at individual members, color-coded from blue to red, we find that a low vorticity occurs in this region only for a few members. After selecting this cluster by brushing, we see the associated regions in spatial context. As depicted in Figure 12 (top row), one can observe that these regions are located around the obstacle and in an outer region behind the obstacle. Note however, that a certain region behind the obstacle is not covered by that cluster. When we select this region, we discover that it belongs to a cluster which has no clear peak. Instead, the ensemble members are distributed evenly with a steadily increasing vorticity, as can be seen in Figure 12 (bottom row) from the color-coded line chart.

The third scalar ensemble field we analyze is a cosmic map comprising 40,000 different, yet possible versions of the observable universe. Each version constitutes a scientifically plausible cosmic density field on a  $256^3$  voxel grid according to the available probabilistic model. A volume rendering of one of these cosmic maps is shown in Figure 13 (top). For further information let us refer to [21]. For this data set, the ensemble summaries occupy about 16 GB, the per-member average values (which also include the initial ensemble data set) about 2.8 TB.

In the analysis we first start by visualizing the uncertainty at a certain level of detail (see Figure 13, middle row). Note that in this figure, the bar height encodes the range over the ensemble members (difference between maximum and minimum) rather than the maximum. By selecting bars indicating high uncertainty and linking to the 3D view, it can be seen immediately that the uncertainty almost only occurs in



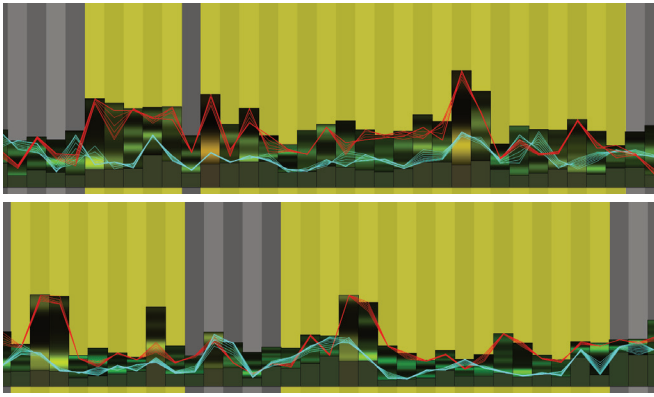


Fig. 14. Two areas (top and bottom image, respectively) in the cosmological ensemble data set that were identified to be positively correlated. Several ensemble members are exemplarily depicted. In both areas, red members exhibit higher values than blue members.

the outer regions of the domain. This conforms with the way the data set was generated. Because far less observational data was used for the galaxy reconstruction in the outer regions than in the inner ones, much higher uncertainties are introduced in the outer regions.

We further cluster bars by histogram similarity and then pick a cluster where the highest density is largely on the bottom of each bar (see Figure 13, bottom row). In other words, most values in that region lie at the lower bound with a small ratio of outliers, which accounts for the low intensity on the top of each bar. Since these bars belong to two distant regions in space we are now interested in finding out whether there exists a certain correlation between these two regions. Therefore we zoom into the region of selected bars until the bars decompose into smaller bars at the next finer level. After picking both regions and asking for the correlation between them, one finally discovers that the ensemble members at these two regions are positively correlated. This is also illustrated by a few chosen members as demonstrated in Figure 14. Note that it will be a very daunting task to find such relations by conventional methods like volume rendering. This is because of the occlusion effects inherent to such methods and, in particular, because displaying even a small part of the current ensemble becomes virtually infeasible for the user to recognize.

## 9 CONCLUSION AND EVALUATION

In this work we have presented a novel technique for the interactive visual exploration of large 3D scalar ensembles. When using this technique, users spend most of the exploration time in an abstract view comprising bar and line charts to convey region-specific statistical data measures. Since the abstract view is not obscured, salient characteristics of the data distributions as well as interdependencies between the data values in different regions can be communicated very effectively. Linked to a view in which the spatial locations are shown in which specific characteristics occur, the technique enables an effective exploration of the whole ensemble space. A particular feature of the proposed technique is, that it does not limit the number of ensemble members to be analyzed. In our opinion this can significantly change the way scientists analyze 3D ensembles in the future.

We have developed the technique in close collaboration with few domain experts from meteorology and astrophysics, yet we did not perform a formal user study to assess the technique's effectiveness. We intend to do this in the near future, including experienced and non-expert users, and domain-specific questionnaires. However, in the experiments we have performed so far, and in which the domain experts took part actively, we have made the following interesting observations: Firstly, the linear arrangement of spatial locations and corresponding sample sets didn't introduce any mental hurdle in the analysis process. We were initially not sure that the abstraction from 3D physical space would be accepted, yet it turned out that all users were

immediately willing to accept this abstraction—and the possible loss of spatial coherence in this abstraction—and perform the analysis in the multi-chart view. Even more surprisingly, all users spent most of the analysis time in the abstract view, linking to the 3D view only very rarely to obtain an overview of the spatial locations of the regions actually being analyzed. As reported by the domain experts, the reason for this behavior was that during multi-chart-based analysis the user more or less entirely focusses on the analysis of the variability of the ensemble members regardless the spatial locations at which the variations occur. Only at the very end of the analysis, when interesting relationships between the members were detected in the abstract view, a further look at the spatial view was taken to locate the corresponding regions in space.

Lastly, all users confirmed the effectiveness of the multi-chart view because they could not imagine any alternative technique allowing for an in-depth sample analysis as proposed. Some of the domain experts had used 3D fields—or cross-sections through these fields—containing mean values and standard deviations to analyze the ensembles. Yet there was strong agreement that this kind of analysis prohibits obtaining any meaningful insights into the sample distributions across the ensemble members. The experts found it especially appealing that sub-domains exhibiting strong variations could already be observed in the overview, and that by zooming into these sub-domains even the distribution characteristics of the data samples across the members could be analyzed. All users confirmed that the richness of specific statistical features that could be retrieved seemed a very unique feature to them, especially due to the fact that all selections could be steered interactively. Concerning the spatial view it was conjectured that a simultaneous visualization of correlations in the abstract and the spatial view would increase the value of the proposed analysis technique significantly.

We further aim at developing improved approaches for handling large and high-resolution ensembles, including compression and parallelization strategies. In particular, we aim to bring our technique into the cloud to exploit remote computing capacities for statistical analysis. Since our technique condenses the data effectively and requires only limited information for graphical display, it seems very well suited for remote cloud environments. Furthermore, the extension to ensembles which vary over time will be challenging, as well as the integration of abstract parameter views to support a visual exploration of multi-parameter data.

## ACKNOWLEDGMENTS

Access to ECMWF prediction data has been kindly provided in the context of the ECMWF special project “Support Tool for HALO Missions”. We are grateful to the special project members Marc Rautenhaus and Andreas Drnbrack for providing the ECMWF ENS dataset of 17 October 2012. We would like to thank Jens Jasche and Torsten Enßlin from the Max-Planck-Institute for Astrophysics for providing the galaxy ensemble. This work was supported by the European Union under the ERC Advanced Grant 291372 SaferVis - Uncertainty Visualization for Reliable Data Discovery.

## REFERENCES

- [1] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.
- [2] N. Blomqvist. On a measure of dependence between two random variables. *The Annals of Mathematical Statistics*, 21(4):593–600, 1950.
- [3] R. Brown. Animated visual vibrations as an uncertainty visualisation technique. In *Proc. GRAPHITE*, pages 84–89, 2004.
- [4] S. Djurcilov, K. Kim, P. Lermusiaux, and A. Pang. Visualizing scalar volumetric data with uncertainty. *Computers & Graphics*, 26(2):239–248, 2002.
- [5] H. Doleisch. SimVis: Interactive visual analysis of large and time-dependent 3D simulation data. In *Proc. Winter Simulation Conference*, pages 712–720, 2007.
- [6] H. Doleisch and H. Hauser. Smooth brushing for focus+context visualization of simulation data in 3D. *Journal of WSCG*, 10(1–3):147–154, 2002.

- [7] W. W. Esty and J. D. Banfield. The box-percentile plot. *Journal of Statistical Software*, 8(17):1–14, 2003.
- [8] A. Goel, C. Baker, C. A. Shaffer, B. Grossman, R. T. Haftka, W. H. Mason, and L. T. Watson. VizCraft: A multidimensional visualization tool for aircraft configuration design. In *Proc. IEEE Visualization*, pages 425–428, 1999.
- [9] D. L. Gresh, B. E. Rogowitz, R. L. Winslow, D. F. Scollan, and C. K. Yung. WEAVE: A system for visually linking 3-D and statistical visualizations, applied to cardiac simulation and measurement data. In *Proc. IEEE Visualization*, pages 489–492, 2000.
- [10] H. Griethe and H. Schumann. The visualization of uncertain data: Methods and problems. In *Proc. SimVis*, pages 143–156, 2006.
- [11] G. Grigoryan and P. Rheingans. Point-based probabilistic surfaces to show surface uncertainty. *IEEE TVCG*, 10(5):564–573, 2004.
- [12] H. Guo, H. Xiao, and X. Yuan. Scalable multivariate volume visualization and analysis based on dimension projection and parallel coordinates. *IEEE TVCG*, 18(9):1397–1410, 2012.
- [13] C. H. Hamilton and A. Rau-Chaplin. Compact Hilbert indices: Space-filling curves for domains with unequal side lengths. *Information Processing Letters*, 105(5):155–163, 2008.
- [14] H. Hauser. Generalizing focus+context visualization. In *Scientific Visualization: The Visual Extraction of Knowledge from Data*, pages 305–327. Springer, 2005.
- [15] H. Hauser, F. Ledermann, and H. Doleisch. Angular brushing of extended parallel coordinates. In *Proc. IEEE Symposium on Information Visualization*, pages 127–137, 2002.
- [16] C. Heinzl, S. Bruckner, M. E. Gröller, A. Pang, H.-C. Hege, K. Potter, R. Westermann, T. Pfaffelmoser, and T. Möller. Uncertainty and parameter space analysis in visualization. IEEE VisWeek Tutorial, 2012.
- [17] J. L. Hintze and R. D. Nelson. Violin plots: A box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.
- [18] T. Höllt, A. Magdy, G. Chen, G. Gopalakrishnan, I. Hoteit, C. Hansen, and M. Hadwiger. Visual analysis of uncertainties in ocean forecasts for planning and operation of off-shore structures. In *Proc. IEEE Pacific Visualization Symposium*, pages 185–192, 2013.
- [19] T. Höllt, A. Magdy, P. Zhan, G. Chen, G. Gopalakrishnan, I. Hoteit, C. D. Hansen, and M. Hadwiger. Ovis: A framework for visual analysis of ocean forecast ensembles. *IEEE TVCG*, PP(99), 2014.
- [20] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, 1985.
- [21] J. Jasche, F. S. Kitaura, C. Li, and T. A. Enflin. Mapping of the universe beyond the known. [http://www.mpa-garching.mpg.de/mpa/research/current\\_research/hl2009-12/hl2009-12-en.html](http://www.mpa-garching.mpg.de/mpa/research/current_research/hl2009-12/hl2009-12-en.html).
- [22] C. R. Johnson and A. R. Sanderson. A next step: Visualizing errors and uncertainty. *IEEE Computer Graphics and Applications*, 23(5):6–10, 2003.
- [23] J. Kehrer, P. Filzmoser, and H. Hauser. Brushing moments in interactive visual analysis. *Computer Graphics Forum*, 29(3):813–822, 2010.
- [24] J. Kehrer, F. Ladstädter, P. Muigg, H. Doleisch, A. Steiner, and H. Hauser. Hypothesis generation in climate research with interactive visual data exploration. *IEEE TVCG*, 14(6):1579–1586, 2008.
- [25] J. Kehrer, P. Muigg, H. Doleisch, and H. Hauser. Interactive visual analysis of heterogeneous scientific data across an interface. *IEEE TVCG*, 17(7):934–946, 2011.
- [26] D. A. Keim. Information visualization and visual data mining. *IEEE TVCG*, 8(1):1–8, 2002.
- [27] D. A. Keim, M. C. Hao, U. Dayal, and M. Hsu. Pixel bar charts: a visualization technique for very large multi-attribute data sets. *Information Visualization*, 1(1):20–34, 2002.
- [28] D. A. Keim, M. C. Hao, U. Dayal, and M. Lyons. Value-cell bar charts for visualizing large transaction data sets. *IEEE TVCG*, 13(4):822–833, 2007.
- [29] R. Kosara, F. Bendix, and H. Hauser. TimeHistograms for large, time-dependent data. In *Proc. Joint Eurographics - IEEE TCVG Symposium on Visualization*, pages 45–54, 2004.
- [30] M. Leutbecher and T. Palmer. Ensemble forecasting. *Journal of Computational Physics*, 227(7):3515–3539, 2008.
- [31] S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
- [32] C. Lundstrom, P. Ljung, A. Persson, and A. Ynnerman. Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE TVCG*, 13(6):1648–1655, 2007.
- [33] A. MacEachren, A. Robinson, S. Hopper, S. Gardner, R. Murray, M. Gahegan, and E. Hetzler. Visualizing geospatial information uncertainty: What we know and what we need to know. *Cartography and Geographic Information Science*, 32(3):139–161, 2005.
- [34] W.-H. Mak, M.-Y. Chan, Y. Wu, K.-K. Chung, and H. Qu. VoxelBars: An informative interface for volume visualization. In *Advances in Visual Computing*, volume 5358 of *Lecture Notes in Computer Science*, pages 161–170. Springer, 2008.
- [35] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997.
- [36] E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.
- [37] T. Pfaffelmoser, M. Reitingner, and R. Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. *Computer Graphics Forum*, 30(3):951–960, 2011.
- [38] T. Pfaffelmoser and R. Westermann. Visualization of global correlation structures in uncertain 2D scalar fields. *Computer Graphics Forum*, 31(3):1025–1034, 2012.
- [39] H. Piringer, R. Kosara, and H. Hauser. Interactive focus+context visualization with linked 2D/3D scatterplots. In *Proc. International Conference on Coordinated & Multiple Views in Exploratory Visualization*, pages 49–60, 2004.
- [40] K. Pothkow and H.-C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE TVCG*, 17(10):1393–1406, 2011.
- [41] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson. Visualizing summary statistics and uncertainty. *Computer Graphics Forum*, 29(3):823–831, 2010.
- [42] K. Potter, P. Rosen, and C. R. Johnson. From quantification to visualization: A taxonomy of uncertainty visualization approaches. In *Uncertainty Quantification in Scientific Computing*, pages 226–249. Springer, 2012.
- [43] 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 International Conference on Data Mining Workshops*, pages 233–240, 2009.
- [44] P. J. Rhodes, R. S. Laramee, R. D. Bergeron, and T. M. Sparr. Uncertainty visualization methods in isosurface rendering. In *Proc. Eurographics*, pages 83–88, 2003.
- [45] M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956.
- [46] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. J. Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE TVCG*, 16(6):1421–1430, 2010.
- [47] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. IEEE Symposium on Visual Languages*, pages 336–348, 1996.
- [48] B. Shneiderman. *Designing the User Interface. Strategies for Effective Human-Computer Interaction*. Addison-Wesley, 3rd edition, 1998.
- [49] D. C. Thompson, J. A. Levine, J. Bennett, P.-T. Bremer, A. Gyulassy, V. Pascucci, and P. P. Pébay. Analysis of large-scale scalar data using hixels. In *Proc. IEEE Symposium on Large Data Analysis and Visualization*, pages 23–30, 2011.
- [50] J. Thomson, E. Hetzler, A. MacEachren, M. Gahegan, and M. Pavel. A typology for visualizing uncertainty. In *Proc. SPIE 5669, Visualization and Data Analysis*, pages 146–157, 2005.
- [51] M. Tory, S. Potts, and T. Möller. A parallel coordinates style interface for exploratory volume visualization. *IEEE TVCG*, 11(1):71–80, 2005.
- [52] C. Wang and H.-W. Shen. LOD Map - a visual interface for navigating multiresolution volume visualization. *IEEE TVCG*, 12(5):1029–1036, 2006.
- [53] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *Proc. Working Conference on Advanced Visual Interfaces*, pages 110–119, 2000.
- [54] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE TVCG*, 19(12):2713–2722, 2013.
- [55] C. M. Wittenbrink, A. T. Pang, and S. K. Lodha. Glyphs for visualizing uncertainty in vector fields. *IEEE TVCG*, 2(3):266–279, 2002.
- [56] S. Zachow, P. Muigg, T. Hildebrandt, H. Doleisch, and H.-C. Hege. Visual exploration of nasal airflow. *IEEE TVCG*, 16(6):1407–1414, 2009.
- [57] B. Zehner, N. Watanabe, and O. Kolditz. Visualization of gridded scalar data with uncertainty in geosciences. *Computers & Geosciences*, 36(10):1268–1275, 2010.