Eur. Phys. J. Special Topics **227**, 1741–1755 (2019)

https://doi.org/10.1140/epjst/e2019-800158-6

THE EUROPEAN PHYSICAL JOURNAL SPECIAL TOPICS

Review

Multivariate visualization of particle data

Liang Zhou[a] and Daniel Weiskopf[b]

Visualization Research Center (VISUS), University of Stuttgart, Stuttgart, Germany

Received 1 October 2018 / Revised in final form 2 December 2018
Published online 8 March 2019

Abstract. In this review paper, we review methods for interactive particle rendering techniques, multi-view particle visualization systems, multivariate visualization techniques, and methods for correlation visualizations. Visualization is vital for gaining insight into particle data. Multivariate particle data are generated to understand different aspects of the underlying physics. The visualization of multivariate particle data is typically performed in multiple linked view systems (multi-view systems) that render particles of interest that are selected by the user interactively with brushing-and-linking. To this end, the non-spatial aspects of particles are explored with multivariate visualization methods, e.g., scatter plots, scatter plot matrix, parallel coordinates, dimensional reduction and radial plots.

1 Introduction

Particle simulation is widely used in physical sciences, for example, in solid mechanics, fluid dynamics, and astrophysics. Usage of particle simulation is found in molecular dynamics that studies evolution of a discrete system. Systems described by partial differential equations can also be studied using particle simulation: for example, smoothed particle hydrodynamics (SPH) is a popular simulation approach in computational fluid dynamics.

We abstract and formulate the visualization problem as follows. Each particle has spatial and temporal information as well as physical attribute(s). We denote a general particle as

$$P_i : (x, y, z, t, S_1, S_2, \ldots, S_m)\,, \tag{1}$$

where x, y, z are the spatial coordinates, t is time, and $S_1, S_2, \ldots, S_m$ are physical attributes of this particle, for example, pressure, temperature, and speed, and m denotes the number of attributes. This abstraction allows us to cover a wide range of applications in particle-oriented simulations. With the advances in computational and storage capacities of computers, the number of particles as well as their attributes keep increasing, in order to provide more accurate descriptions of the physics.

Therefore, visualization has become increasingly important in the analysis of particle simulations – both for gaining insight into the simulation data to understand the underlying physics and debugging the simulation algorithms and implementations.

[a] e-mail: Liang.Zhou@visus.uni-stuttgart.de
[b] e-mail: Daniel.Weiskopf@visus.uni-stuttgart.de

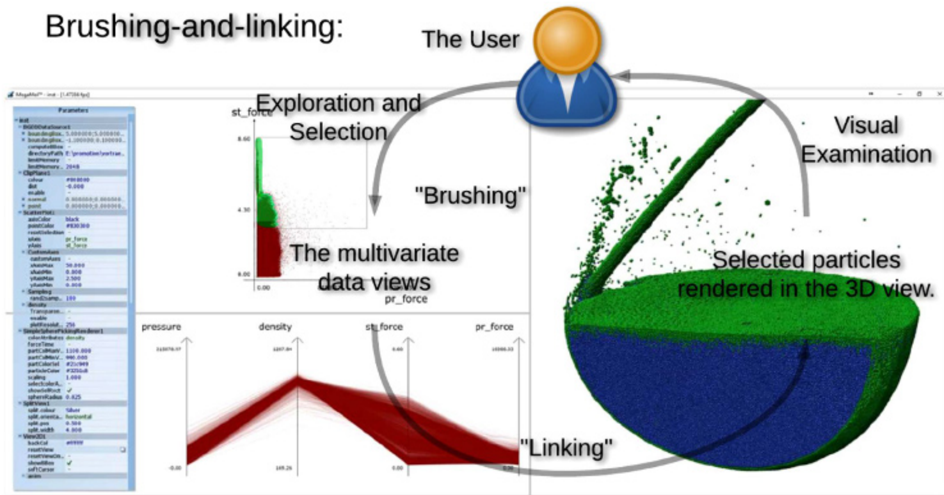


Fig. 1. Visual debugging system of SPH simulations [2]. The MegaMol-based system uses multiple linked views with brushing-and-linking.

Visualizations of particle data are used in two major ways: visualizing the spatial-temporal information of particles, i.e., x, y, z, t – the rendering of actual particles in space and time, and showing the non-spatial multivariate attribute domain of $S_1, S_2, \ldots, S_m$. In practice, these two parts, namely, a particle renderer and multivariate visualizations, are linked and integrated into multiple linked view systems to allow for effective and flexible analysis of particle data. The idea behind such systems is to perform closed-loop visual data analytics using brushing-and-linking [1] as illustrated in Figure 1: the user explores and selects data points of interest with brushing (selecting) in the multivariate attribute domain, and the particles of attribute values within the selection are rendered in the 3D view through linking; then, the user visually inspects the rendering to make refinement on the queries. Through this tight linking, we can build an overall mental model and understanding of the data. The loop ends whenever the user is satisfied with the refined results.

In the remainder of this review paper, we first briefly report on popular particle rendering techniques and multi-view visualizations for particle simulations (Sect. 2). We then summarize representative multivariate visualizations in Section 3: scatter plot matrix (SPLOM) – a direct extension of scatter plots, parallel coordinates, radial plots, and dimension reduction approaches. Next, we review relevant multivariate correlation visualizations (Sect. 4), followed by a discussion of aggregation methods and frequency plots (Sect. 5).

2 Visualizing particle simulations

It is important to have efficient 3D rendering techniques that handle large-scale particle data and generate easy-to-perceive visualizations. Particle rendering techniques concern the (x, y, z, t)-tuple of a particle as in equation (1). Furthermore, effective and flexible methods that allow for interactive brushing-and-linking to explore and visualize the particle data are fundamental for the understanding of particle simulations, i.e., the full description $(x, y, z, t, S_1, S_2, \ldots, S_m)$ of particles. This section reviews relevant works in these two areas.

2.1 Interactive particle rendering techniques

Particle data visualization is utilized in many physical sciences applications [3]. For general-purpose particle rendering, GPU-based ray casting is a popular and very fast method for directly rendering large amount of spheres [4] as well as complex glyphs [5]. It is shown that interactive rendering can be achieved for multiple millions of particles on a single GPU with ray casting. Based on the GPU-based rendering technique, an open source particle rendering framework MegaMol [6] is implemented. CPU-based ray casting can also achieve interactivity [7], and is advantageous for larger datasets due to the larger memory a CPU can access than a GPU. For large datasets, acceleration structures, for example, octree [8] and KD-tree [7] have been exploited. For the topic of SPH simulations, a survey of high quality visualizations can be found elsewhere [9].

Particle rendering techniques are not our focus in this review paper. For a more comprehensive view of particle rendering techniques, we refer the reader to the article of project D.3. Overall, interactive particle rendering techniques are an important building block of full-fledged multivariate particle visualization methods using interactive brushing-and-linking. Such methods closely link a particle rendering view with one or several complementary multivariate non-spatial visualization views to analyze particle simulations.

2.2 Multi-view particle visualization

Brushing-and-linking [1,10] is an effective way to explore and visualize multivariate datasets. It is common practice to use brushing-and-linking with multiple linked views to visualize and analyze multivariate data having spatial information [11,12]. Multiple linked scatter plots in 2D and 3D are used to analyze computational fluid simulations that are rendered with particles [13]. A multi-view application using various types of plots, e.g., scatter plots and histograms, that are linked to a particle renderer, has been used to analyze combustion simulations [14]. Parallel coordinates plots are linked with a renderer that supports both particle and volume rendering to facilitate visual analysis of large multi-dimensional simulations [15].

Figure 1 shows a visual debugging system for multivariate SPH simulations that combines scatter plots, parallel coordinates, and a particle renderer [2]. The system is based on the MegaMol framework [6], and is designed to aid the debugging of fluid simulations for digital entertainment purposes, e.g., animations and movies.

A star coordinates plot (explained in Sect. 3.3) can be used for interactive selection of particle clusters for surface reconstruction of SPH simulations [16]. Here, particles are clustered automatically using their multidimensional attributes, and the results are projected into the 3D star coordinates where the user has the flexibility to control the projection to better examine and select clusters. Another method allows the user to explore the multivariate SPH data with continuous star coordinates [17]. The data are explored through interactive manipulation of the star coordinates and selected particles are rendered in a spatial view. Details of continuous representations can be found in Section 5.

In the next section, we review popular multivariate visualization techniques that are commonly used as methods for visualizing the multivariate attribute domain in multi-view particle visualizations.

3 Multivariate visualization techniques

Visualizing the non-spatial attribute domain, i.e., the $(S_1, S_2, \ldots, S_m)$-tuple in equation (1), of particle simulations require appropriate multivariate visualization techniques. Direct visualization is feasible and effective for datasets with up to two attributes by using 2D scatter plots. In a 2D scatter plot, a visual element (usually a dot) is drawn for a data entry in the 2D image space, and its location is determined by taking a first attribute and a second attribute as the horizontal and vertical coordinates respectively. Afterward, the accumulated dots are usually mapped to colors using color maps.

For three attributes, it is still possible by direct visualization with 3D scatter plots, but the inherent occlusion problem in 3D makes it less effective. It is more difficult if not impossible with scatter plots alone to visualize datasets with more than three attributes. Therefore, variants and alternatives are needed. In the following, we summarize representative and popular multivariate visualization techniques: SPLOM, parallel coordinates, dimensionality reduction methods, and radial plots. A broader introduction and taxonomy of multivariate visualization techniques can be found elsewhere [18,19].

3.1 Scatter plot matrices

A SPLOM is a matrix consisting of individual 2D scatter plots as its elements. For a number of m attributes: $S_1, S_2, \ldots, S_m$ of particles, a SPLOM is a $m \times m$ matrix, where each of its non-diagonal element is a scatter plot as shown in Figure 2. For each particle, it is mapped to a dot in each scatter plot in the SPLOM.

SPLOM inherits advantages of 2D scatter plots of direct visualization of a pair of attributes, for example, SPLOM provides good support for cluster detection in an attribute pair. However, SPLOM increases its number of cells quadratically for increasing dimensionality, which results in two major issues.

First, a SPLOM has to be shown in a limited display area, and the dots could become too small to be seen with increasing dimensionality of the SPLOM as they become subpixel-sized. For example, the outlier in the zoomed-in inset in Figure 2 cannot be seen in the full-dimensional version. However, with excessive zooming-and-panning, it breaks the mental picture of the visualization during analysis.

Second, it is difficult to trace data points of interest across attributes in a SPLOM. Tracing requires a good amount of visual navigation and is further compounded by the first issue if the dimensionality is high. To avoid breaking the flow of analysis, Elmqvist et al. [20] propose an interactive exploration solution called ScatterDice. There, the user performs structured navigation in the multidimensional space and refines queries from different viewports.

3.2 Parallel coordinates

Parallel coordinates [21,22] are a popular visual mapping method for multivariate data. Parallel coordinates show all attributes of multivariate data on parallel axes where a data point is mapped to a polygonal line (polyline) across all axes. For particle data with attributes $S_1, S_2, \ldots, S_m$, each particle is mapped to a polyline across all m axes. A survey of parallel coordinates can be found elsewhere [23].

Figure 3 illustrates the basic concept of parallel coordinates: a 2D point in a scatter plot is mapped to a line (in practice, we often cull parts that are outside of the attribute pair) in parallel coordinates by connecting locations on vertical axes of attribute values of that data point, and a line in a scatter plot is mapped to a point in parallel coordinates. This relationship is referred to as point-line duality [24].

Parallel coordinates can be easily extended to higher dimensions by adding more axes and connecting the polylines. In this way, we can see patterns of multivariate

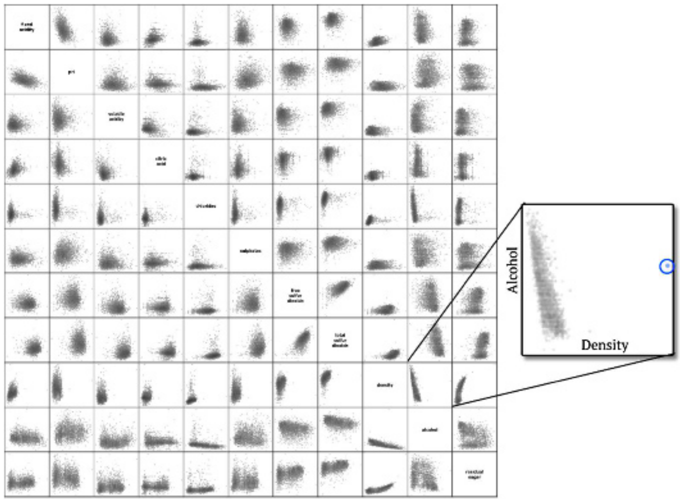
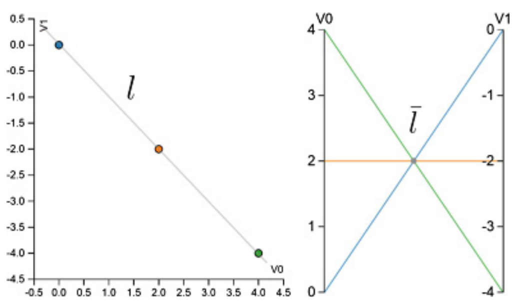


Fig. 2. A SPLOM of a multivariate dataset of 11 attributes. An outlier in the "density attribute" is highlighted in red in the zoom-in inset.



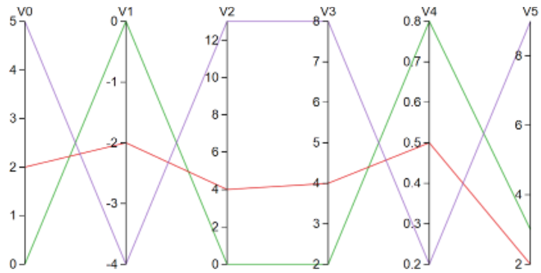
(a) (b)

Fig. 3. The mapping of data points from a scatter plot (a) to parallel coordinates (b). For each data point in scatter plot, a line is created in parallel coordinates by connecting locations of corresponding attribute values on vertical axes; conversely, a line l in scatter plot is mapped to a point $\bar{l}$ in parallel coordinates.

relationships. For example, Figure 4 shows a simple example of a 6D dataset of three entries. In fact, the scalability of dimensions is a great advantage of parallel coordinates: with increasing number of data dimensions, we just need to add further axes (which works fine up to limitations imposed by the display space).

Another advantage of parallel coordinates is the traceability of data across dimensions. There is evidence that better performance is achieved with parallel coordinates than with SPLOM for value-retrieval tasks [25], and a controlled eye-tracking study [26] shows that parallel coordinates are better than SPLOM for value-estimation tasks when the number of dimensions becomes higher (8 dimensions in that specific case). For example, one can easily follow the outlier colored in blue in Figure 5 across all dimensions, whereas it is very difficult if not impossible with the SPLOM as in Figure 2.

A major issue of parallel coordinates is occlusion – polylines start to occlude each other when data size increases – making it difficult for visualizing local features and detecting clusters in pairs of attributes. In fact, scatter plots and SPLOMs



V0	V1	V2	V3	V4	V5
0	0	0	2	0.8	3
2	-2	4	4	0.5	2
5	-4	13	8	0.2	9

Fig. 4. A parallel coordinates plot (left) of a simple 6D dataset (right).

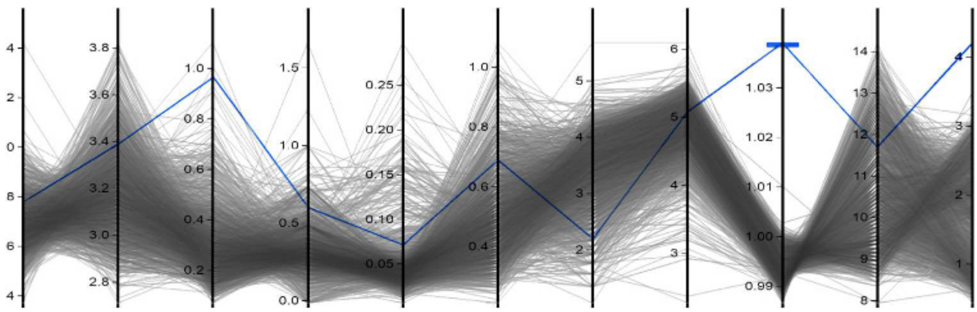


Fig. 5. Parallel coordinates of the multivariate dataset used in Figure 2 with the same outlier highlighted in blue.

complement parallel coordinates. Therefore, in practice, parallel coordinates are usually used in conjunction with scatter plots or SPLOMs with brushing-and-linking.

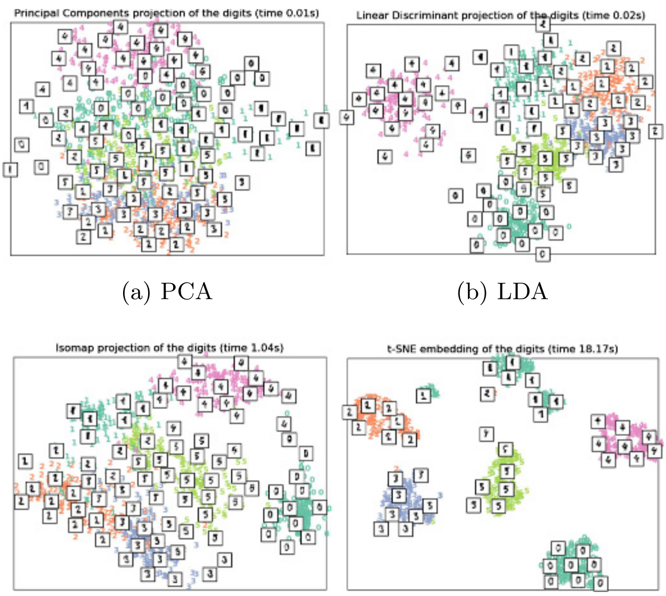
3.3 Radial plots

Multidimensional data $S_1, S_2, \ldots, S_m$ of particles can be visualized in plots with radial layouts as well. The radar or spider chart arranges vertical axes in a parallel coordinates plot in a circular layout with shared origin at the center with the polylines drawn as polygons. Conversions between a parallel coordinates plot and a radar chart allows for flexible linked-axes visualizations [27].

The star coordinates plot [28] uses circularly arranged attribute axes that share the same origin at the center to represent basis vectors of an affine projection. The projection matrix is changed via user interactions with the axes: the user can flexibly modify the orientation and length of axes. However, this affine projection could lead to strong distortions, and it could be laborious to find a good projection interactively.

Radviz [29] is another popular multidimensional circular plot. Unlike star coordinate plot, Radviz does not define a transformation matrix but uses a spring tension minimization algorithm to place data points. The attributes are defined as points that are arranged equally over a unit circle. Each data point then connects to each of the attribute points with a virtual spring whose stiffness is proportional to its value. The final position of a data point, where the summed forces is equal to zero, is found through minimization.

Aforementioned visualization techniques work well for multidimensional data, however, for high-dimensional data, e.g., with hundreds or thousands of dimensions, dimensional reduction and projection methods are necessary. The radial plots use projections to 2D for visualizations.



(a) PCA (b) LDA

(c) Isomap (d) tSNE

Fig. 6. Dimensional reduction methods on a high-dimensional dataset. The results are color-coded by cluster ID, and the computational time of each method is labeled above the figure. The scikit-learn package (http://scikit-learn.org/stable/) is used to generate the results.

3.4 Dimension reduction methods

Dimensional reduction and projection are a popular method for multivariate data visualization. These techniques result in 2D visualizations based on similarity of samples in the multidimensional space. In general, these techniques can be classified into two categories: linear and nonlinear methods. Linear methods use linear transformations to project data from an m-dimensional space to a low-dimensional space, e.g., 2D in most cases. Typical linear methods include Principal Component Analysis (PCA), Multidimensional Scaling (MDS), and Linear Discriminant Analysis (LDA). Linear methods usually have low computational and storage complexity, and are easier to interpret; however, such methods fail to preserve complex multidimensional structures of the data.

In contrast, nonlinear methods could better preserve complex structures in the multidimensional space. Representative nonlinear methods include: Isomap [30], local linear embedding (LLE) [31], Laplacian Eigenmap (LE) [32], and t-Distributed Stochastic Neighbor Embedding (tSNE) [33]. Notably, tSNE is particularly popular nowadays as it is advantageous in preserving multidimensional clusters that have complex structures. Figure 6 shows typical dimensional reduction methods on a 64-dimensional dataset: Linear methods, PCA and LDA, are shown on the top row; while nonlinear methods, Isomap and tSNE, are shown on the second row. With cluster-based color coding, it can be seen that tSNE provides clearer cluster separation than other methods, while the computational times of linear methods are much shorter.

Our review paper does not focus on dimensional reduction methods, which are an important topic in statistics and data sciences. We refer the reader to [34] for details of dimensional reduction techniques.

Aforementioned techniques visualize multivariate data as independent data items; however, these techniques do not concern relationships between data items. Understanding such relationships, i.e., correlations, is an important aspect of multivariate data visualization and analysis. In the next section, we review visualization methods for correlations.

4 Visualization of correlation information

Correlation information is important for multivariate data analysis – in the case of particle data, we are interested in the relationships between $S_1, S_2, \ldots, S_m$ of particles. The global correlation of attributes provides overall similarity information between these attributes. In contrast, local fitting, i.e., correlation and regression, can be used to approximate complicated correlational relationships between data attributes. Furthermore, global linear relationships can be approximated by local linear relationships [35].

Dedicated correlation coordinate plots are proposed for correlation analysis of multidimensional data [36]. The correlation coordinate plot combines scatter plots and star plots to show the strength as well as the shape of global correlation of transformed data points. This technique focuses on (many) pairwise correlations of two attributes but also has some support to combine several pairs of attributes.

Alternatively, correlation information can be visualized directly in multivariate visualization techniques that are covered in Section 3 for a more comprehensive, and coherent understanding of the dataset without context switching. In the remainder of this section, we focus on techniques that superimpose local fitting visualizations with multivariate visualizations.

4.1 Visualizing correlations with scatter plots and SPLOMs

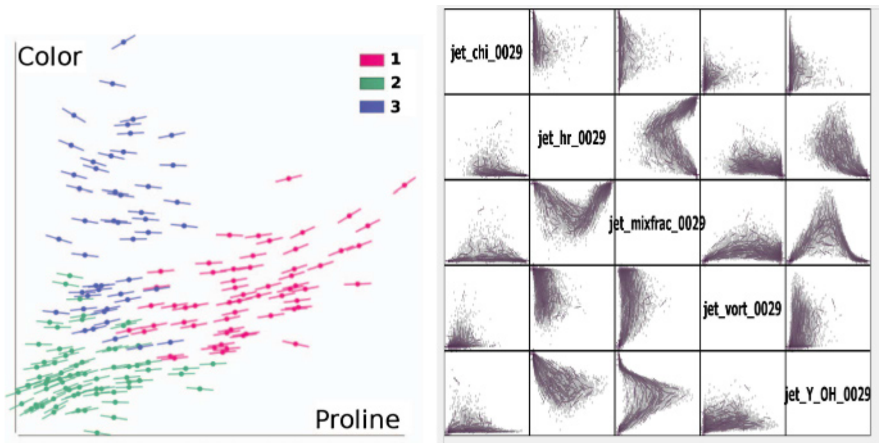
A dot-line representation with streamlines is used to visualize locally computed trend lines in 2D scatter plots [37,38] for sensitivity analysis.

An example of a dot-line-based SPLOM visualization can be seen in Figure 7. In Figure 7a, the flow-based scatter plot [37] draws trend lines estimated from the local neighborhood of data points in the 2D space. Alternatively, a dot-line-based SPLOM (Fig. 7b) can be used for correlation visualization; here, the trend lines are orthogonally projected from local estimation calculated in the multidimensional space into corresponding 2D subspaces.

Another technique uses illuminated 3D scatter plots to visualize local fitting information [39]. The neighborhood of each data point is classified into linear, planar, or volumetric structures by an eigen-analysis of the covariance matrix. Given the classification result, different illumination models are applied to improve shape perception. Nevertheless, these methods are limited to show correlation information of 2 and 3 attributes, and do not scale to higher dimensions.

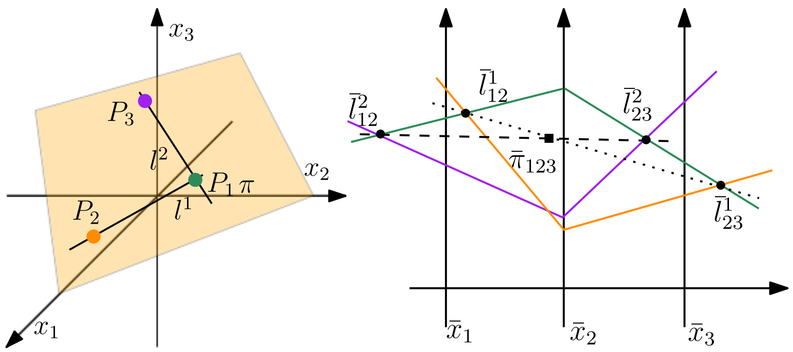
4.2 Correlation visualizations in parallel coordinates

Direct visualization of local fitting information is also made possible in parallel coordinates [40]. Thanks to the asymmetric nature of positive and negative correlations in parallel coordinates, positively correlated data points are transformed to fit into the same display area of negatively correlated data. However, their method can visualize local linear relationships limited to 2D, and the positive mapping violates the point-line duality.



(a) (b)

Fig. 7. Correlation visualization with scatter plots: (a) a flow-based 2D scatter plot for sensitivity analysis [38] (image courtesy of IEEE) and a dot-line SPLOM visualization of local fittings of a combustion simulation data (b).



(a) (b)

Fig. 8. Recursive construction of an indexed point in parallel coordinates (b) of a 2-flat (plane) π in Cartesian coordinates (a). The first indexed point $\bar{\pi}_{123}$of 2-flats can be constructed from 1-flat indexed points as shown in (b).

Instead, we propose a method for visualizing local multivariate correlations in parallel coordinates using indexed points of p-flats [41], which are generalized flat surface of dimension p in high-dimensional space. An indexed point of a p-flat is the point representation of the p-flat in the same 2D domain of parallel coordinates.

An example of the indexed point of 1-flat is shown in Figure 3 denoted as $\bar{l}$. Indexed points of higher order p-flats can be derived recursively from indexed points of lower order p-flats. Figure 8 shows the recursive construction of an indexed point $\bar{\pi}_{123}$ of 2-flat π (plane), from 1-flat indexed points.

In our method, the indexed points are used to represent linear fittings of the local neighborhood of each data point directly on parallel coordinates. Specifically, local lines and planes are estimated for 2D and 3D subspaces in the local neighborhood of a point via principal component analysis. Then, we plot indexed points of 1- and 2-flats for local lines and planes respectively in parallel coordinates.

Our technique supports visualization of local multivariate correlations (of three attributes or more) in parallel coordinates, which was previously impossible. In

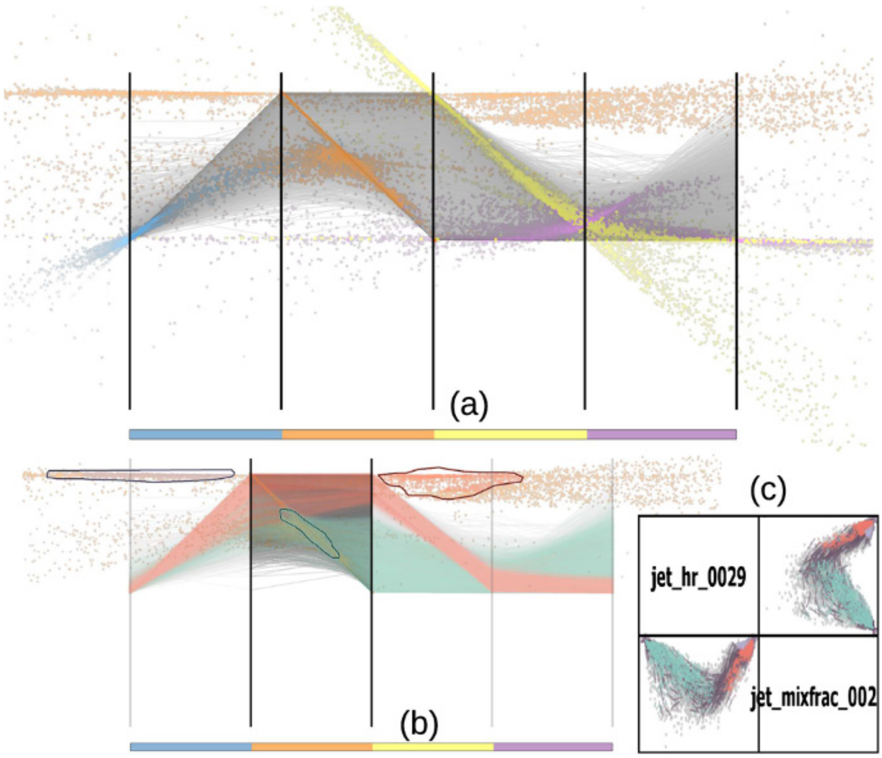


Fig. 9. Indexed point parallel coordinates of 1-flats of a combustion particle simulation. The 1-flat indexed points of all dimensions are shown in (a), while (b) shows the indexed point selection within the second parallel pair with a lasso. Highlighted samples are shown in scatter plots in (c).

theory, an arbitrarily high order correlations can be supported in our technique, which was also not possible in existing correlation visualization techniques as described in Section 4. Another important benefit of our method is that it enables the visualization of positive and negative correlations with clear visual patterns.

With the support of brushing-and-linking interactions, our method allows the user to select regions of interest in data that was impossible with existing techniques. We demonstrate the usefulness of our method with a particle-based combustion simulation data. Visualizations of 1-flat indexed points are shown in Figure 9.

Figure 10 shows 2-flat indexed points indicating planes in the dataset. Two major parts are selected by the user within the subspace of last three attributes (b), showing two surfaces joining together in the data domain as seen in the SPLOM (c).

Overall, our method gives a new prospective in multivariate data visualization and analysis. It is a promising new technique that can be used as an add-on for traditional parallel coordinates in multi-view particle visualization systems.

5 Aggregation methods for multivariate visualizations

In a typical simulation, a large amount of particles are used and above-mentioned techniques are faced with scalability issues due to over-plotting. In fact, a rather small amount of particles, e.g., several dozens of thousands, could cause severe over-plotting

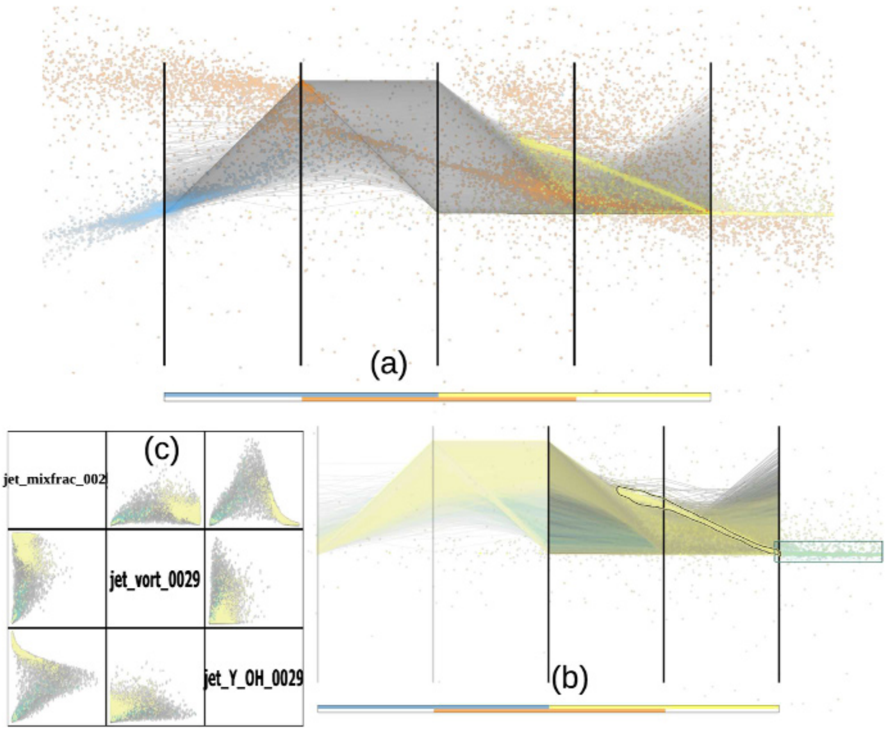


Fig. 10. Visualization of 2-flat indexed points of the combustion data: (a) shows indexed points from all dimensions, (b) shows user selected samples of the subspace of last three attributes, and (c) shows the SPLOM with selected samples which form two surfaces.

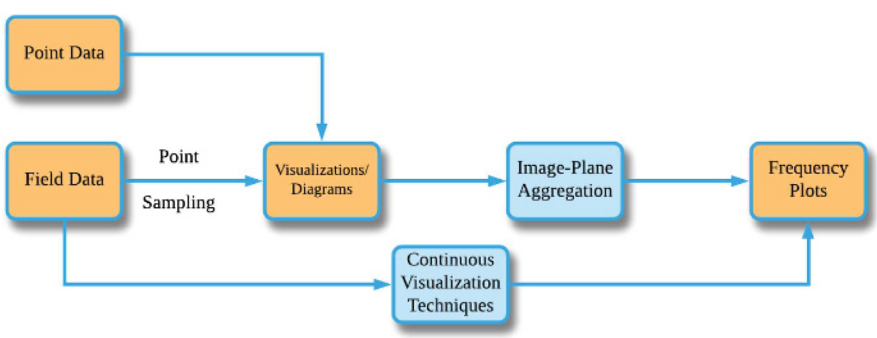


Fig. 11. Strategies for visualizing point-based and field-based, e.g., SPH, particle simulations.

that makes those visualizations useless. Therefore, aggregation-based visualization methods and frequency plots are proposed to address the scalability issue.

One solution is to aggregate data on the image plane (the upper path of Fig. 11). Binning methods divide the multivariate data domain into intervals, i.e., bins, and aggregate each data point into the bin it belongs to [42]. It is common practice to use binning in scatter plots and SPLOMs, and the idea is extended for parallel coordinates visualization [43] by applying the point-line duality (Sect. 3.2). Gaussian filtering can be further applied to the binned data to create a density field. It is critical to choose a suitable bin number for good visualzation that balances details of features and

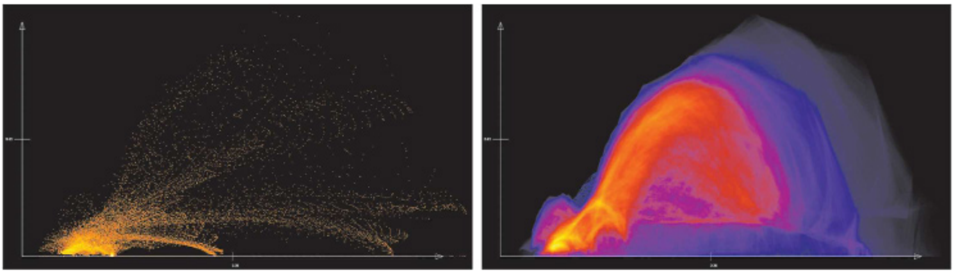


Fig. 12. Scatter plots of the "blunt-fin" simulation dataset. The traditional scatter plot is shown to the left, whereas the continuous scatter plot [12] is shown to the right. Image courtesy of IEEE.

performance. Alternatively, kernel density estimation, a nonparametric method, is used to generate multivariate density field [42,44] on the image plane. Similar to the problem of choosing an optimal number of bins, kernel density estimation is sensitive to the choice of bandwidth.

Image-plane-based techniques work well for point-based particle simulations, however, artifacts are introduced when applying such techniques to field-based and multi-scale simulations due to point sampling (the middle path in Fig. 11). Rigorous and accurate methods that aggregate in the data domain are proposed for field data (the bottom path in Fig. 11).

The continuous scatter plot [12] uses a generic and rigorous mathematical model to map an arbitrary density defined on an n-dimensional input domain to an m-dimensional scatter plot. An example of a 2D continuous scatter plot (Fig. 12(right)) of a "blunt-fin" fluid simulation reveals the multiple arc structures in the data and the smooth density function in the scatter plot domain. In comparison, a traditional scatter plot (Fig. 12(left)) does not show the arches nor the overall density distribution of the data well.

Likewise, a continuous density model can be applied to parallel coordinates [45]. Here, the continuous version of parallel coordinates is generated by transforming density from the continuous scatter plot to the parallel coordinates using the point-line duality. Figure 13 shows a comparison between traditional parallel coordinates (left) and continuous parallel coordinates (right) generated from the same input data. The continuous parallel coordinates provide an estimation of the density function of the whole data domain, whereas the traditional parallel coordinates are only able to show discrete data items in the input data.

The continuous representation is also proposed for star coordinates plots [17]. The method supports any sample arrangement using an isotropic density function without the need of a grid or a mesh as in [12]. Notably, a multi-view system built around an interactive continuous star coordinates widget is proposed for analyzing multivariate SPH data, where the continuous star coordinates is consistent with the kernel function of the simulation.

In-depth analysis of features in continuous visualizations is conducted [46,47]. Discontinuities in continuous scatter plots are studied and the continuous scatter plot is extended by a discontinuity-based visualization approach [46]. Based on point-line duality, features in continuous parallel coordinates [47].

The continuous representations can be combined with correlation information visualization. With the p-flat indexed points [41], density fields can be estimated in a similar fashion with either the structured [12] or the unstructured method [17]. This way, a more comprehensive picture of the correlations can be drawn.

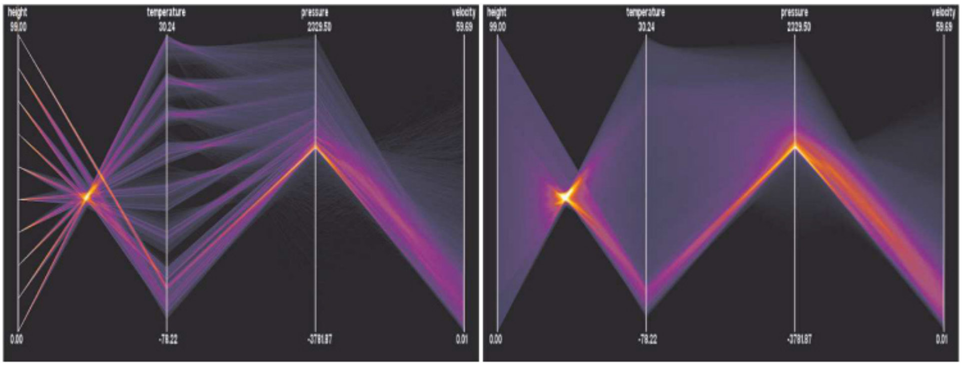


Fig. 13. Parallel coordinates of the hurricane Isabel simulation. To the left, shows the traditional parallel coordinates, while the continuous version is shown to the right. Image courtesy of IEEE.

6 Conclusion

In conclusion, we have reviewed relevant techniques for multivariate particle visualizations. Representative interactive particle rendering techniques and multi-view visual analytics systems were summarized. Then, we reviewed the basics of commonly used multivariate visualizations: SPLOM, parallel coordinates, radial plots, and dimensional reduction methods. Next, we discussed correlation visualization for multivariate data including our method of indexed point parallel coordinates for multivariate correlation visualizations. Finally, we covered relevant aggregation techniques for point- and field-based particle simulations.

The spatial configuration of particle simulations can be effectively visualized with interactive rendering techniques. The non-spatial, multivariate aspect of particle simulations are of equal importance as they could provide more insight into the simulations. Multi-view visualizations that closely link the spatial rendering and non-spatial multivariate techniques are fundamental for the understanding of the increasingly more complicated particle simulations.

This work was supported by SFB 716 D.5. The following project-related research works were published during the third funding period (2015–2018): [2,41,48,49].

References

1. A.R. Martin, M.O. Ward, High dimensional brushing for interactive exploration of multivariate data, in *Proceedings of the 6th Conference on Visualization '95, VIS '95* (IEEE Computer Society, Washington, DC, USA 1995), p. 271
2. S. Reinhardt, M. Huber, O. Dumitrescu, M. Krone, B. Eberhardt, D. Weiskopf, Visual debugging of SPH simulations, in *2017 21st International Conference Information Visualisation (IV), 11–14 July 2017* (IEEE, London, 2017), pp. 117–126
3. D.R. Lipşa, R.S. Laramee, S.J. Cox, J.C. Roberts, R. Walker, M.A. Borkin, H. Pfister, Comput. Graphics Forum **33**, 2317 (2012)
4. S. Gumhold, Splatting illuminated ellipsoids with depth correction, in *Proceedings of the Vision, Modeling, and Visualization Conference 2003 (VMV 2003), München, Germany, 19–21 November 2003* (2003), pp. 245–252
5. G. Reina, T. Ertl, Hardware-accelerated glyphs for mono- and dipoles in molecular dynamics visualization, in *EUROVIS 2005: Eurographics/IEEE VGTC Symposium on*

Visualization, edited by K. Brodlie, D. Duke, K. Joy (The Eurographics Association, 2005)
6. S. Grottel, M. Krone, C. Müller, G. Reina, T. Ertl, IEEE Trans. Visual. Comput. Graphics **21**, 201 (2015)
7. I. Wald, A. Knoll, G.P. Johnson, W. Usher, V. Pascucci, M.E. Papka, CPU ray tracing large particle data with balanced P-k-d trees, in *2015 IEEE Scientific Visualization Conference (SciVis), 25–30 October 2015* (IEEE, Chicago, IL, USA, 2015), pp. 57–64
8. R. Fraedrich, S. Auer, R. Westermann, IEEE Trans. Visual. Comput. Graphics **16**, 1533 (2010)
9. M. Ihmsen, J. Orthmann, B. Solenthaler, A. Kolb, M. Teschner, SPH fluids in computer graphics, in *Eurographics 2014 – State of the Art Reports*, edited by S. Lefebvre, M. Spagnuolo (The Eurographics Association, 2014)
10. M.O. Ward, XmdvTool: integrating multiple methods for visualizing multivariate data, in *Proceedings of the IEEE Visualization Conference* (Washington, DC, 1994), pp. 326–333
11. L. Zhou, C. Hansen, Transfer function design based on user selected samples for intuitive multivariate volume exploration, in *2013 IEEE Pacific Visualization Symposium (PacificVis), 27 February–1 March 2013* (IEEE, Sydney, NSW, Australia 2013), pp. 73–80
12. S. Bachthaler, D. Weiskopf, IEEE Trans. Visual. Comput. Graphics **14**, 1428 (2008)
13. H. Piringer, R. Kosara, H. Hauser, Interactive focus+context visualization with linked 2D/3D scatterplots, in *Proceedings. Second International Conference on Coordinated and Multiple Views in Exploratory Visualization, 2004, 13 July 2004* (IEEE, London, 2004), pp. 49–60
14. H. Doleisch, M. Mayer, M. Gasser, R. Wanker, H. Hauser, Case study: visual analysis of complex, time-dependent simulation results of a diesel exhaust system, in *Eurographics/EEE VGTC Symposium on Visualization*, edited by O. Deussen, C. Hansen, D. Keim, D. Saupe (The Eurographics Association, 2004)
15. O. Rubel, Prabhat, K. Wu, H. Childs, J. Meredith, C.G.R. Geddes, E. Cormier-Michel, S. Ahern, G.H. Weber, P. Messmer, H. Hagen, B. Hamann, E.W. Bethel, High performance multivariate visual data exploration for extremely large data, in *SC '08: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing, 15–21 November 2008* (IEEE, Austin, TX, USA, 2008), pp. 1–12
16. L. Linsen, T.V. Long, P. Rosenthal, S. Rosswog, IEEE Trans. Visual. Comput. Graphics **14**, 1483 (2008)
17. V. Molchanov, A. Fofonov, L. Linsen, Comput. Graphics Forum **32**, 301 (2013)
18. M. Ward, G. Grinstein, D. Keim, *Multivariate density estimation: theory, practice, and visualization*, 2nd edn. (A. K. Peters, Ltd., Natick, MA, USA, 2015)
19. D.A. Keim, IEEE Trans. Visual. Comput. Graphics **8**, 1 (2002)
20. N. Elmqvist, P. Dragicevic, J. Fekete, IEEE Trans. Visual. Comput. Graphics **14**, 1539 (2008)
21. A. Inselberg, Visual Comput. **1**, 69 (1985)
22. E.J. Wegman, J. Am. Stat. Assoc. **85**, 664 (1990)
23. J. Heinrich, D. Weiskopf, State of the art of parallel coordinates, in *Eurographics 2013 – State of the Art Reports*, edited by M. Sbert, L. Szirmay-Kalos (The Eurographics Association, 2013)
24. A. Inselberg, *Parallel coordinates: visual multidimensional geometry and its applications* (Springer, Berlin, 2009)
25. X. Kuang, H. Zhang, S. Zhao, M. McGuffin, Comput. Graphics Forum **31**, 1365 (2012)
26. R. Netzel, J. Vuong, U. Engelke, S. O'Donoghue, D. Weiskopf, J. Heinrich, Visual Inf. **1**, 118 (2017)
27. J.H.T. Claessen, J.J. van Wijk, IEEE Trans. Visual. Comput. Graphics **17**, 2310 (2011)
28. E. Kandogan, Star coordinates: a multi-dimensional visualization technique with uniform treatment of dimensions, in *Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics* (Citeseer, 2000), pp. 9–12

29. P. Hoffman, G. Grinstein, K. Marx, I. Grosse, E. Stanley, DNA visual and analytic data mining, in *Proceedings. Visualization '97 (Cat. No. 97CB36155), 24 October 1997* (IEEE, Phoenix, AZ, USA, 1997), pp. 437–441
30. J.B. Tenenbaum, V.D. Silva, J.C. Langford, Science **290**, 2319 (2000)
31. S.T. Roweis, L.K. Saul, Science **290**, 2323 (2000)
32. M. Belkin, P. Niyogi, Neural Comput. **15**, 1373 (2003)
33. L. van der Maaten, G.E. Hinton, J. Mach. Learn. Res. **9** 2579 (2008)
34. J.A. Lee, M. Verleysen, *Nonlinear dimensionality reduction* (Springer-Verlag New York, New York, NY, USA, 2007)
35. W.S. Cleveland, S.J. Devlin, E. Grosse, J. Econ. **37**, 87 (1988)
36. H. Nguyen, P. Rosen, Improved identification of data correlations through correlation coordinate plots, in *Proceedings of the International Conference on Information Visualization Theory and Applications (IVAPP)* (SciTePress, 2016), Vol. 2, pp. 60–71
37. Y.H. Chan, C.D. Correa, K.L. Ma, Flow-based scatterplots for sensitivity analysis, in *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST), 25–26 October 2010* (IEEE, Salt Lake City, UT, USA, 2010), pp. 43–50
38. Y.H. Chan, C.D. Correa, K.L. Ma, IEEE Trans. Visual. Comput. Graphics **19**, 1768 (2013)
39. H. Sanftmann, D. Weiskopf, Comput. Graphics Forum **28**, 751 (2009)
40. H. Nguyen, P. Rosen, IEEE Trans. Visual. Comput. Graphics **24**, 1301 (2018)
41. L. Zhou, D. Weiskopf, IEEE Trans. Visual. Comput. Graphics **24**, 1997 (2018)
42. B.W. Silverman, *Density estimation for statistics and data analysis* (Chapman & Hall, London, 1986)
43. M. Novotny, H. Hauser, IEEE Trans. Visual. Comput. Graphics **12**, 893 (2006)
44. D.W. Scott, in *Multivariate density estimation and visualization* (Springer Berlin, Heidelberg, 2012), pp. 549–569
45. J. Heinrich, D. Weiskopf, IEEE Trans. Visual. Comput. Graphics **15**, 1531 (2009)
46. D.J. Lehmann, H. Theisel, IEEE Trans. Visual. Comput. Graphics **16**, 1291 (2010)
47. D.J. Lehmann, H. Theisel, IEEE Trans. Visual. Comput. Graphics **17**, 1912 (2011)
48. L. Zhou, C.D. Hansen, IEEE Trans. Visual. Comput. Graphics **22**, 2051 (2016)
49. L. Zhou, D. Weiskopf, Contrast enhancement based on viewing distance, in *Proceedings of the 11th International Symposium on Visual Information Communication and Interaction, VINCI '18, Växjö, Sweden, 13–15 August 2018* (ACM, New York, NY, USA, 2018), pp. 25–32