Edge Bundling for Parallel Coordinates

Seminar: Visualisierung Multidimensionaler Daten

Denise J. Rappold

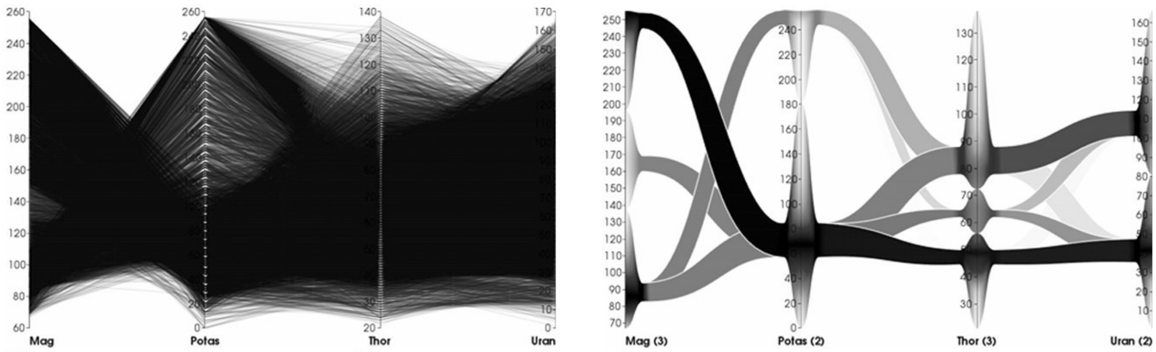

Figure 1. Comparison of a standard Parallel Coordinates Plot (left) and an edge-bundled version (right) introduced by Palmas et al. [12], illustrating how edge bundling can reduce visual clutter.

Abstract—Parallel Coordinate Plots (PCPs) are a widely used visualization tool for data analysis, providing an overview of complex, multidimensional datasets and revealing relationships across multiple dimensions. When used with large datasets, however, PCPs often suffer from excessive overplotting of lines, making data exploration challenging. To address this issue, various clutter reduction techniques have been developed, including data reduction techniques, visual encoding optimizations, spatial adjustments, and interaction-based techniques. Among visual encoding techniques, edge bundling has gained significant attention for its ability to effectively reduce visual clutter, improving data analysis tasks such as cluster identification.

Nevertheless, edge bundling comes with certain limitations. By reducing visual clutter through the bundling of edges, it often increases overdraw, potentially obscuring individual data points and thus, affecting information preservation. Additionally, the computational complexity of these methods raises concerns regarding real-time interaction, particularly for large datasets.

To identify areas for improvement, this work provides an overview of the existing edge bundling techniques. It introduces, compares, and evaluates the most commonly used edge bundling approaches within the context of PCPs, focusing on the most relevant methods. In general, edge bundling bundles similar curves together. While many approaches transform original polygonal lines (polylines) into curves, they differ in their similarity measures, which determine how observation points are selected for bundling. To provide clarity, this work presents a taxonomy distinguishing between the *data-centric*, and *emerging* edge bundling techniques.

By offering a comprehensive overview of edge bundling for PCPs, this work examines its effectiveness, limitations, and potential future directions within the field of multidimensional data visualization.

1 Introduction

Parallel Coordinate Plots (PCPs) are a visualization technique commonly used for multidimensional data, particularly in the exploratory data analysis, which involves investigating data sets to uncover patterns, relationships, or anomalies. First introduced by Inselberg [8] in 1985, PCPs allow us to display N-dimensional data in two-dimensional space by arranging parallel axes, each representing a dimension. Individual data points are depicted as polygonal lines (polylines) that intersect each axes at their corresponding values, enabling analysts to visually explore relationships across multiple dimensions simultaneously and gain an intuitive overview.

However, in times of Big data, increasing dataset size and complexity present significant challenges, such as visual clutter and overplotting, which obscure meaningful patterns and significantly hinder effective analysis. These issues make pattern recognition and visual exploration nearly impossible.

To address these issues, researchers have explored multiple clutter reduction techniques, including edge bundling. Originally developed for graph and network visualization, edge bundling groups multiple polylines into cohesive bundles, serving two main purposes: reducing visual clutter and revealing underlying cluster structures. In the context of PCPs, the primary motivation for applying edge bundling is clutter reduction, as it simplifies dense plots and improves readability. Additionally, cluster visualization is another key benefit [5], as bundling highlights shared trajectories and relationships between similar data points.

By leveraging the Gestalt principles [9] of similarity, proximity and continuity, edge bundling enhances the perceptual cohesion of the paths, making structures easier for the human eye to follow. As illustrated in Figure 1, edge bundling transforms an overplotted PCP into a more structured, abstract representation, emphasizing dominant trends while reducing visual clutter.

While edge bundling is an effective method for simplifying dense visualizations, its application requires careful consideration. The bundling of edges can obscure fine-grained trends and patterns, potentially leading to information loss. Therefore, maintaining a delicate balance between clutter reduction and preserving critical details is essential to uncover meaningful insights.

This paper introduces and evaluate various edge bundling approaches documented in the literature, ranging from data-centric, cost-based, and angular-based techniques. By analyzing these methods, this work aims to highlight their strengths, limitations and suitability for PCPs. A particular focus is placed on how different edge bundling strategies exploit Gestalt principles to enhance interpretability while balancing

Institute for Visualization and Interactive Systems, University of Stuttgart, Winter Term 2024/25.

clarity and information preservation.

2 RELATED WORK

The literature on edge bundling for parallel coordinates remains sparse, which makes finding a well-organized taxonomy of existing edge bundling techniques rather difficult. Zhou et al. [15] propose a taxonomy that categorizes edge bundling techniques into cost-based, geometry-based and image-based approaches. While image-based methods and cost-based approaches, particularly energy minimization techniques have been applied to PCPs, geometry-based methods are more prevalent in the field of graph visualization. This work introduces an extended version of the technique-based taxonomy originally proposed by Zhou et al. [15].

2.1 Contributions

Despite the increasing adoption of edge bundling in PCPs, a comprehensive taxonomy of existing approaches is still lacking. This work seeks to bridge the gap by:

- Providing a structured overview of edge bundling techniques applied to PCPs, categorizing them based on their core principles
- Assessing their effectiveness in reducing clutter while preserving meaningful patterns and drawing connections to the Gestalt principles.
- Identifying gaps and further directions in edge bundling for PCPs, outlining potential improvements and research opportunities

2.2 Structure of the paper

This work is structured as follows. Section 2 provides background information, including an overview of PCPs, edge bundling, and the Gestalt laws of perception relevant to PCP visualization. Section 3 reviews existing edge bundling techniques, classifying them based on their approach to selecting similar edges for bundling. Section 4 analyzes these methods, discussing their advantages and limitations. Section 5 further discusses general limitations that come with the application of edge bundling for parallel coordinates, outlining potential future directions. Finally, Section 6 concludes the paper.

3 THEORETICAL BACKGROUND

In order to better understand edge bundling in the context of parallel coordinate plots, this section provides an overview of its origin and fundamental concepts. Starting with the basics of pattern recognition in traditional PCPs and the initial development of edge bundling in graph and network visualization. Next, Gestalt principles such as similarity, proximity and continuity are introduced. They play a crucial role in the perceptual effectiveness of bundling techniques. Finally, the importance of geometric curve models is discussed, as they form a foundation for many edge bundling approaches by determining how polylines are transformed into smooth curves.

3.1 Pattern Recognition in PCPs

In classical parallel coordinates adjacent variables can be analyzed by observing the pattern of line segment intersections [6]. A perfect positive correlation between the adjacent dimensions is found, when the lines are parallel or fan-shaped and no intersections occur. Intersections in the center of the plot indicate a strong negative correlation, while intersecting lines in a smeared-out accumulation region indicate a less strong negative correlation. When the intersections seem random, no correlation is implied.

3.2 Origin in Graph Visualization

Originally introduced by D. Holten in "Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data" [7] in 2006, edge bundling was developed to reduce visual clutter in graphs. The original approach leverages the inherent tree structure of hierarchical data to group adjacency edges into bundles, similar to how electrical wires are bundled together in structured cabling to avoid tangling. By bending adjacency edges along the hierarchical paths that connect nodes, edge bundling visually aggregates these edges into coherent bundles. In order to achieve smooth bending, each adjacency edge is modeled as a piecewise cubic B-spline curve, resulting in visually appealing and clear structures [7]. Figure 2 provides a visual example of this concept.

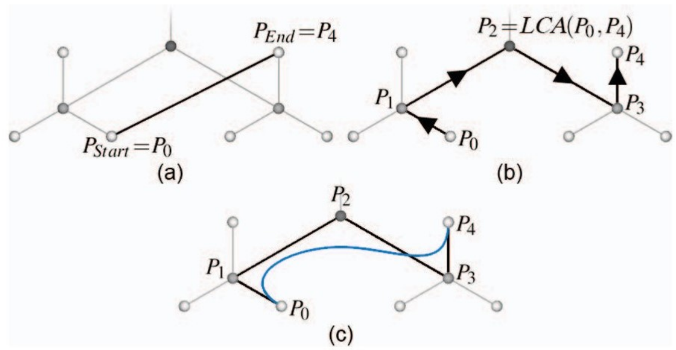

Figure 2. Bundling of adjacency edges: converting (a) a straight line from P_0 to P_4 into (c) a spline curve using (b) control points P_i along the hierarchy. Figure reprinted from [7]

While Holten et al. [7] use piecewise cubic B-splines as their curve model due to its ability to locally control coherent bundles in a computational effective way, other geometric curve models, such as Bézier curves also have been explored to transform polylines into curves for edge bundling [12, 4].

Early edge bundling techniques were initially designed for graphs with arbitrary node positions. However, many methods focused on solely deforming edges in order to group similar ones into bundles, leaving starting and end nodes at their original positions. This concept naturally extends to parallel coordinate plots, which suffer from similar challenges of edge clutter and overplotting as traditional graphs.

In the context of PCPs, where polylines represent multidimensional data, edge bundling techniques have been adapted to reduce visual clutter and highlight underlying clusters. The transformation of polylines into curves is essential in this adaptation, as it enables a clearer, more organized bundled visualization. This transformation not only improves readability but also guides the viewer's perspective, making relationships across dimensions more apparent.

3.3 Use of Gestalt Laws

Such visual transformations of polylines to curves, align closely with the Gestalt laws of visual perception, which describe how humans naturally perceive and organize visual elements based on principles such as similarity, proximity, and continuity. Edge bundling is found to be leveraging these Gestalt principles to establish clarity and readability in graph visualizations [9].

The law of similarity suggests that the human brain tends to group visual similar elements together. In the context of PCPs, this is often implemented through color coding edges assigned to the same cluster, making cluster mapping more intuitive. Depending on the edge bundling method, visual cues like color or shape can be used to group edges, taking advantage of the principle of similarity [9].

While similarity is commonly applied to original polyline-based PCPs, edge bundling further frees up the visual color channel by utilizing another Gestalt law: proximity [12].

The principle of proximity states that elements closer together are often instinctively perceived as part of the same group. Edge bundling exploits this by bringing related edges physically closer, thereby making it easier to identify clusters and relationships between them [9]. This is particularly evident in the approach by Heinrich et al. [4],

which will be looked at in Section 3.
According, to the principle of continuity, smooth curves guide the viewer's eye more naturally than polylined edges with sharp turns [9]. This principle has already been applied to PCPs to address the cross-over problem, where multiple lines intersect at a common point on an axis. By replacing the traditional zigzagging polylined paths with smooth cubic curves, the approach introduced by Graham and Kennedy [3] helps users follow individual lines more easily, utilizing the principle of continuity to improve the clarity of the visualization. As shown in Figure 3, this approach enhances readability [3]. Edge bundling takes advantage of this principle by creating smooth curves, using curve models like B-Splines or Bézier curves to bundle around cluster centroids as seen in the methods proposed by Zhou et al. [16] and Palmas et al. [12], more in Section 3.

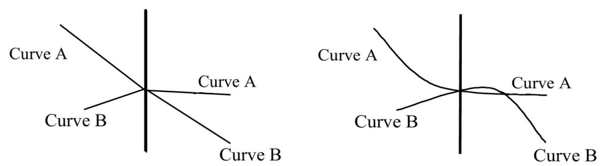

Figure 3. The cross-over problem can easily be mitigated by the Gestalt law of continuity. Figure reprinted with adjustments from [3].

3.4 Importance of Curve Models

A common aspect among many edge bundling approaches applied to PCPs is the transformation of the polylines into curves. In this context, we can view the classical polyline on PCPs as a special case of continuous but non-smooth curves [5]. This transformation is significant, as curves are often easier for the human eye to follow, as demonstrated in the cross-over problem [3]. Furthermore, when attempting to bundle edges closer together, following the principle of proximity, straight lines inevitably need to bend. This bundling process is when geometric curve models become essential, providing a structured way to transform polylines into curves [2].
While smooth and continuous curves significantly enhance the readability of PCPs, not all edge bundling approaches rely on such smooth curves. Some methods such as the one introduced by McDonnell et al. [11] use B-splines, which are piecewise-defined, thus not continuous. Although B-splines offer flexibility and efficiency, they do not achieve the smoothness as seen in e.g. Bézier curves, which can impact the overall clarity of the visualization. This will further be discussed in Section 3 and 4.

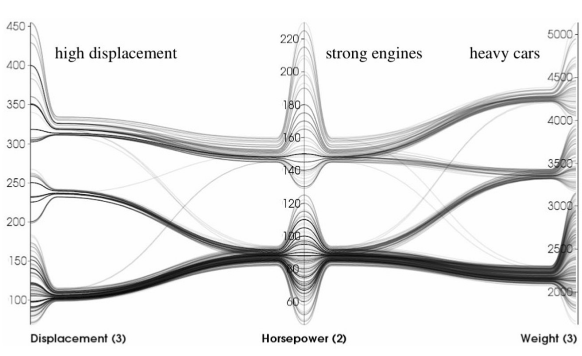

Figure 4. By turning each polyline into a smooth cubic Bézier curve the visual clustering approach can turn a PCP into a structured, implicit visualization with less visual clutter. Figure reprinted from [12]

4 Methods

Most edge bundling methods focus on the process rather than defining a bundle explicitly. Following the idea introduced by Lhuillier et al. [10], edge bundling refers to the grouping of similar edges into compact shapes. However, we extend this with the notion presented by Heinrich et al. that bundles can either be "visualized implicitly, as set of curves, or explicitly, using polygons", [5]. It is also important to note that the type of similarity measures used to determine *similar* edges, differ between various approaches.

4.1 Edge Bundling Categories

An extended taxonomy of edge bundling techniques applied to parallel coordinates is presented, building on prior work to offer a more coherent categorization. I adopt a technique-based taxonomy [10] inspired by Zhou et al. [15], which classifies edge bundling methods into cost-based, geometry-based, and image-based categories. I further build on the idea presented by Heinrich et al. [5] that a bundle represents all data points "belonging to a cluster defined a priori or emerging from the bundling algorithm", [5]. Accordingly, I divide methods into two broader categories: *data-centric* edge bundling and *emerging* edge bundling methods.

Data-centric edge bundling involves grouping data based and on underlying attributes before rendering the plot [12]. Techniques of this category cluster data a priori [4, 11] using data clustering algorithms, which includes the image-based approach as defined in the taxonomy proposed by Zhou et al.[11]. Bundles are created by bending the edges towards the previously assigned cluster centroid, involving curve transformation.

Emerging edge bundling methods, which define bundles through the bundling algorithm itself, are further divided into cost-based bundling and angular-based bundling. Adopted from the work of Zhou et al. [15], the cost-based category can further be divided into ink minimization and energy minimization, bundling edges based on geometric relationships between the lines themselves. And lastly, the angular-based approach [13] bundles edges based on their angular distribution information.

4.2 Data-Centric Edge Bundling

First, *data-centric* approaches will be examined. While all of these methods cluster data a priori, they differ in the clustering algorithms they use, the curve models they employ, and their edge bundling processes.

4.2.1 Multidimensional clustering

Data-centric edge bundling involves preprocessing the data to assign cluster memberships before rendering.
Heinrich et al. [4] assume multidimensional clustering and represent polylines as piecewise cubic Bézier curves Figure 5. This is achieved through the insertion of a virtual bundling axis in the middle of two value axes, splitting each original polyline into two curve segments. While the starting and end point P_i and P_{i+1} keep their position, the smoothness scale α determines the curvature by controlling the placement of secondary control axes which set the position of two other control points per curve segment. The bundling strength parameter β then adjusts the intersection point Q_i verticaly on the bundling axis, shifting Q_i towards the corresponding cluster centroid C_i. The updated position of Q'_i is computed as follows:

$$Q'_i = (1-\beta)Q_i + \beta C_i$$

Hereby, the parameter β is interactively adjustable within the range $0 \leq \beta \leq 1$. When $\alpha = \beta = 0$, the result is a standard polyline, while increasing β moves Q'_i progressively closer to C_i, strengthening the bundling effect.
At $\beta = 1$, strict edge bundling occurs, with all data points of the same cluster collapsing into a single intersection at the cluster centroid:

$$Q'_i = C_i$$

This approach relies on precomputed cluster membership as a similarity measure for edge bundling, ensuring that visually grouped edges represent inherent data structures.

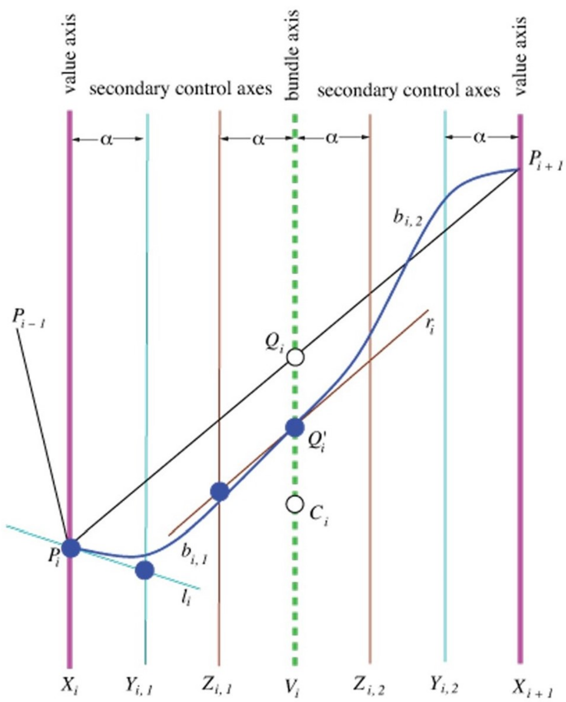

Figure 5. By inserting a bundling axis V_i between two adjacent data axes X_i and X_{i+1} a polyline can be turned into a cubic Bźier curve pieces using the blue control points to adjust its shape with the smoothness scale α. The position of Q_i' is adjusted based on the bundling strength β, pulling it closer to the cluster centroid C_i with increasing value. Figure reprinted from [4]

4.2.2 Image-based

Among various artistic rendering techniques introduced as *Illustrative Parallel Coordinates* (IPC), the image-based edge bundling approach is introduced by McDonnel et al. [11]. Since each data point is assigned to a cluster beforehand, which determines the bundling process, this approach is also categorized as *data-centric*. The method builds upon hierarchical edge bundling, as introduced by Halten et al. [7]. It transforms polylines into end-point interpolating B-splines curves, with a tension parameter β controlling the spline curvature. Higher values of β result in straighter lines. The control points are positioned based on the original polylines and their respective cluster membership, ensuring smooth transitions within the area between the data axes. This results in an implicit bundling representation Figure 6. To further enhance the visualization, McDonnel et al. [11] extend their approach by introducing semi-transparent colored polygons that explicitly represent clusters. Between adjacent axes, quadrilaterals are formed by connecting sample points from the upper and lower boundaries, generating an explicit visualization of the bundles (Figure 7.
As is typical for *data-centric* approaches, the a priori defined cluster assignment serves as similarity measure in this approach.

4.2.3 Density-based clustering

In Palmas et al.'s approach [12], edge bundling utilizes one dimensional clustering, more precisely the *Gaussian kernel density estimation* (KDE) to group the data into clusters. The KDE function is given by:

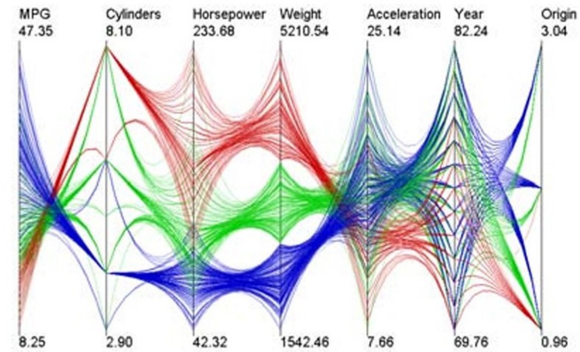

Figure 6. Implicit visualization of bundled polycurves. Same colored curves are assigned to the same cluster following the Gestalt law of similarity. Figure reprinted from [11]

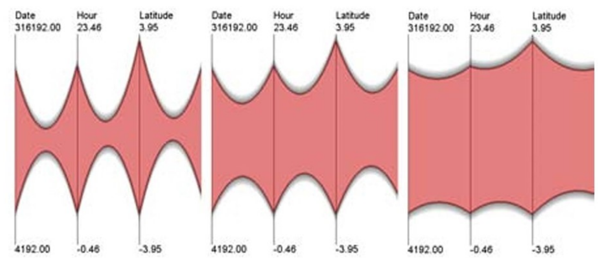

Figure 7. Explicit spline-based cluster rendering. Spline tension β values 0.15, 0.50, 0.85. Figure reprinted from [11]

$$f(x) = \frac{1}{n\sigma\sqrt{2\pi}} \sum_{i=1}^{n} e^{-\frac{(x_i - x)^2}{2\sigma^2}}$$

This density-based method defines clusters as 3-tupels (x^-, x^0, x^+), where x^0 represents the maximum of the kernel density function, and x^- and x^+ correspond to the minima. Maxima of the density function indicate the densest region, which intuitively correspond to the cluster centroid, while the minima represent the sparsest regions, describing the cluster boundaries. This method allows for interactive adjustments in the number of clusters by manipulating the bandwidth parameter σ. A smaller σ produces narrower kernels, increasing the number of extrema, and consequently, the number of clusters. Figure 8 shows a simple example of six observation points.

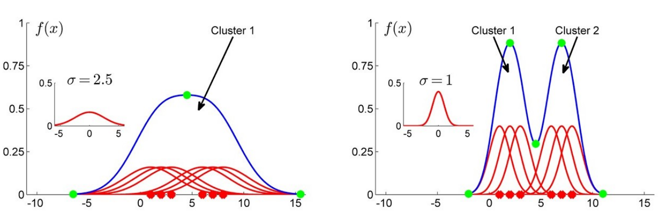

Figure 8. A kernel density estimation function for six example data points. A larger value for σ (right) results in more clusters. Figure reprinted from [12]

Once the 3-tupel representation for each cluster is computed, it can be used to define the area on the data axis that each cluster spans. For the rendering process Palmas et al. [12] pursue two different techniques: one that creates implicit bundles by rendering each observation point as an individual curve, and another that replaces individual curves with polygonal strips for a more explicit bundling.
To render the data, two virtual bundling axes are inserted between

adjacent data axes, dividing each polyline into three segments. The 3-tupel representation is mapped onto the data axes, and on the virtual bundling axis. For the mapping onto the bundling axes, the values are adjusted with a scaling factor s, narrowing down the interval of the cluster boundaries and thus, creating the bundling effect. This adjustment is given by: $(x^0 + s(x^- - x^0), x^0, x^0 + s(x^+ - x^0))$.
By including two additional control points per segments G^1-continuous curves are created. The result is an implicit bundle for each cluster, reducing visual clutter. However, the rendering performance depends on the number of lines, which makes the approach rather inefficient for large datasets. To address this, Palmas et al. [12] introduce an optimized explicit version, where the clusters are presented as polygonal strips rather than individual curves. In this explicit approach, a Bézier curve is drawn between the cluster centroid of adjacent bundling axes, and offset curves are drawn perpendicular to the Bézier curve. The width of each bundle, along with additional greyscale shading, now indicates the cluster size. This method, shown in Figure **??**, reduces rendering time, since the performance now depends on the number of polygonal strips.

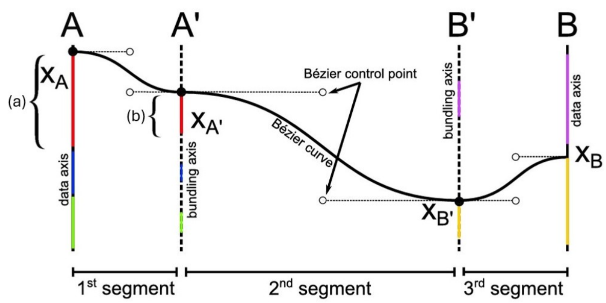

Figure 9. Each polyline is transformed into a Bézier curve consisting of three segments. The 3-tupel (x^-, x^0, x^+) representation on the data axis (a) is directly derived from the kernel density function. The interval (e.g red) for the corresponding cluster along the bundling axis (b) must be modified with a scaling factor s, causing the desired bundling effect. Figure reprinted from [12]

In this case the similarity measure used to determine the data points of each bundle is based on proximity defined by the visual clustering process that assigns the cluster membership to each observation point a priori.

4.3 Emerging edge bundling

The following *emerging* edge bundling methods do not rely on a priori assigned cluster memberships; instead, they evolve the bundles dynamically during the bundling process.

4.3.1 Cost-based

This approach falls under the *emerging edge bundling methods*, which define bundles through the bundling algorithm itself, rather than relying on predefined data clusters. In this case Zhou et al. [16] employ an energy minimization framework to determine the optimal edge shape. Since visual visual clutter is generally difficult to measure, parallel coordinates are modeled as a system of force interactions between individual lines, where an energy function guides the bundling process

$$E = \alpha_c E_{curvature} + (1 - \alpha_c) E_{gravitation}$$

with α as a weighting coefficient, that adjusts the visual clustering, as depicted in Figure 10.
The system seeks to minimize energy levels, as higher curvature $E_{curvature}$ contributes to increased energy values. Simultaneously, gravitational energy $E_{gravitation}$ describes interactions between neighboring lines. As a result, lines that are initially closer together and more visually parallel become more tightly bundled throughout the process.
This results in an implicit visualization, seen in Figure 10.

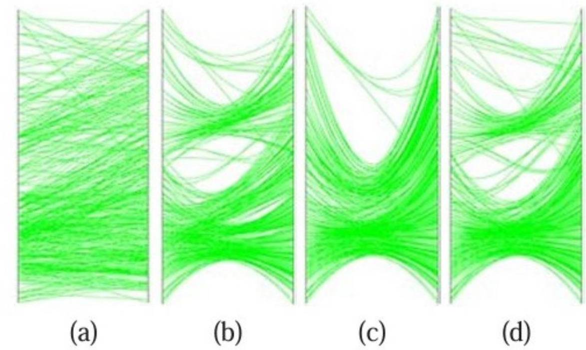

Figure 10. Cost-based approach resulting in implicit bundles. (a) no visual clustering; (b)-(c) effect of energy term weighting on visual clustering. Figure reprinted from [16]

Unlike *data-centric* approaches that rely on predefined cluster assignments, this method does not explicitly categorize data points. Instead, spatially close and visually aligned edges are dynamically grouped through the optimization process, leaving the identification of clusters to the user.

4.3.2 Angular-based

The angular-based approach focuses on preserving correlation information between adjacent data axes - something often lost in other bundling approaches. The intersection patterns between adjacent data axes are inherent to parallel coordinates and essential for analyzing relationships within data.
To address the issue that these intersection patterns are often not maintained when reducing visual clutter, Watanabe et al.[13] introduces an angular-based edge bundling technique. Their method extends the classical parallel coordinate plots by integrating an Angular Distribution Plot (ADP) between adjacent axes.
The method calculates the angle θ of each polyline segment $l^{(m)}$ as it intersects a data axis. Using this information, the mean $\overline{\theta}$ and variance $\hat{\theta}$ for each polyline are derived. The ADP then visually encodes these values:

- The x-axis represents the mean angle $\overline{\theta}$, with 0 positioned at the center. A negative value indicates more downward-sloping segments, while a positive value represents more upwards-sloping ones.

- The y-axis represents variance $\hat{\theta}$, with 0 at the bottom. A lower variance suggests fewer intersections (indicative of strong positive correlation), while a higher variance indicates more intersections (indicative of negative correlation).

Thus, the position of a point in the ADP encodes its correlation pattern, offering an alternative way to interpret relationships between dimensions.
To integrate this angular distribution information into the bundling process, the ADP is inserted between adjacent axes. Each polyline segment is linked to its corresponding ADP point by construction a Bézier curve between the original polyline's starting point $P^{(m)}_{x_j}$, its corresponding point in the ADP, and the segment's endpoint $P^{(m)}_{x_{j+1}}$ (Figure 11).

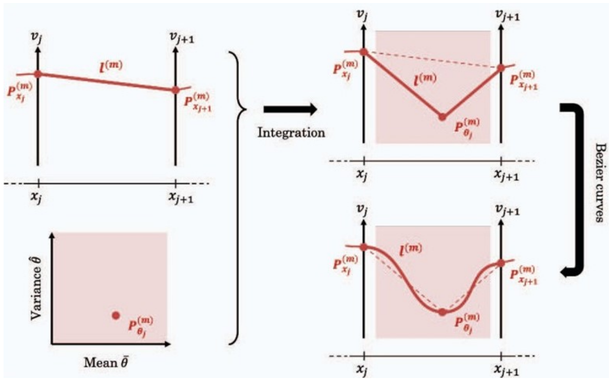

Figure 11. Constructing Bézier curves with three control points, including the starting and end point $P_{x_j}^{(m)}$, $P_{x_{j+1}}^{(m)}$ of the original polyline $l^{(m)}$ and $P_{\theta_j}^{(m)}$ as the position within the angular distribution plot. Figure reprinted from [13]

By applying this technique across all polylines, a bundling effect emerges based on the similarity of correlation patterns. The result is a visualization that reduces clutter while still preserving essential intersection information.

5 Evaluation

This section evaluates the previously introduced edge bundling methods for parallel coordinates, analyzing their effectiveness in reducing visual clutter, and supporting data interpretability, as well as their limitations.

The *data-centric* approach introduced by Heinrich et al. [4] effectively reduces visual clutter by transforming polylines into continuous, smooth curves. This transformation makes line tracing more visually pleasant for the eyes and leverages the Gestalt law of continuity and proximity, as curves assigned to the same cluster are bundled. A user study confirms its effectiveness in assisting cluster estimation tasks.
While this approach provides the user with a wide range of views from abstract high-level to detailed low-level representation, simply by adjusting the parameter value of β, we must note that strict edge bundling result in complete loss of pattern as seen in Figure 12.

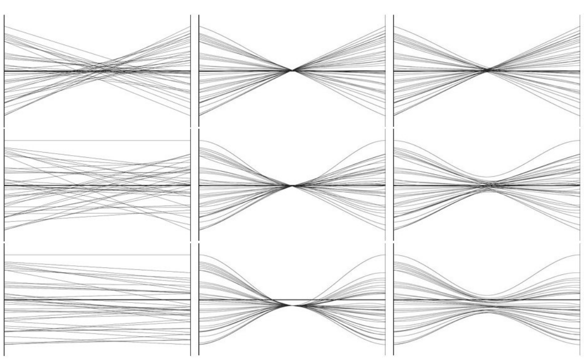

Figure 12. Strict edge bunding results in no visual difference for positive and negative correlation. first row: strong negative correlation; second row: no correlation; third row: positive correlation with the bundling parameter of 0; 1; 0.8 per column. Figure reprinted from [4]

Moreover, the quality of the visualization is restricted by the underlying clustering method. Choosing a suitable algorithm can be quite challenging and often needs expert knowledge. However, this issue can be generalized for all *data-centric* bundling approaches.
These limitations suggest that visual edge bundling may provide a more flexible alternative.

While McDonnel et al. [11] introduce various novel IPC rendering techniques for parallel coordinate, edge bundling is just one aspect of their approach. Their main focus is on creating aesthetically pleasing visualizations, rather than intuitive and analytically driven representations.
Although, B-splines ensure smooth transitions within the area between data axes, the transitions at the axes themselves remain sharp and less intuitive. The cross-over problem persists even after transforming polylines into polycurves. While the principle of continuity is not found here, this approach still adheres the Gestalt principles of proximity by bundling polylines based on their cluster assignment. The principle of similarity is reinforced through the use of color to differentiate clusters. Despite these advantages, the lack of smooth transitions across dimensions poses challenges for traceability, making the method less effective for analytical tasks.

Palmas et al. [12] propose a density-based visual bundling technique using cubic Bézier curves, ensuring C^1-continuous and G^1-continuous splines for smooth transitions. This method adheres to the Gestalt laws of continuity and proximity, improving readability. The intuitive clustering approach allows an easy interactive adjustment of cluster numbers, providing flexibility for data exploration. Additionally, the bundled parallel coordinate plots have shown to produce statistically significant reduction in correlations estimation errors.
However, there is no statistically proven improvement in subsets tracing tasks, unless color encoding is additionally used.

The cost-based method by Zhou et al. [16] leverages spatial relationships rather than predefined clusters. By treating PCPs as force interaction systems, this approach removes the need for explicit data classification, mitigation the risk of poor clustering algorithm selection. However, since the method relies on geometric relationships, the results may be inconsistent based on the order of dimensions.

Unlike other methods, the angular-based approach by Watanabe et al. [13] focuses on preserving correlation patterns. By introducing an *Angular Distribution Plot*, this method visually encodes intersection variance and mean slope to retain correlation relations, making this approach ideal for correlation analysis. However, this approach does not create smooth transition between dimensions which makes line tracing challenging. Consequently, a user must rely on interactive selection and color encoding to follow a data point.
This approach is great for visual clutter reduction while still preserving correlation information found in the intersection pattern.

Each bundling technique offers a unique balance between clutter reduction and information preservation. While *data-centric* approaches might be most suitable for cluster identification tasks, *emerging* edge bundling approaches as the cost-based approach might be best for general visual grouping, as it removes any dependencies on clustering algorithms. The angular-based bundling is superior for correlation analysis, as it maintains intersection patterns.

6 Discussion

Edge bundling for parallel coordinate plots has proven effective in reducing visual clutter, enhancing cluster identification and data interpretability, aligning well with Gestalt principles such as continuity and proximity. By transforming polylines into smooth curves, these techniques offer an intuitive representation of relationships across multiple dimensions. Additionally, many approaches support interactive adjustments, allowing users to switch between implicit views and broader explicit overviews.
However, transforming dense, cluttered visualization into structured, more readable representations, several limitations remain. These challenges must be carefully considered before applying edge bundling

techniques to PCPs.
In this section the general limitations of edge bundling for parallel coordinates will be discussed paving the way for possible future research directions.

Computational Complexity One of the greatest challenges of edge bundling is its computational complexity. By relying on geometric curve modeling and clustering algorithms, edge bundling has a complex nature, making them computationally expensive. Thus, many bundling techniques require significant processing power, limiting their scalability and interactive usage. While precomputation can be a clever way to handle high rendering time [12], the scalability of many methods is still limited. While explicit bundle representations as introduced by Heinrich et al.[4] are another approach to decrease the rendering time, scalability limitations and computational complexity still remain. Cui et al. [1] present a more scalable lightweight bundling method that uses a frequency-based approach to display clusters as histogram-like bundles to better visualize the data distribution, while still reducing overplotting. As demonstrated in a case study, when it comes to changing the number of clusters per dimension interactively, their method takes 1 second to cluster 10^6 data records with 4 dimensions, while the approach using KDE introduced by Palmas et al. [12] takes 60 seconds for one dimension with 10^5 data points, solely to precompute the required information. The lightweight method demonstrates high scalability, combined with user interactions it effectively assists visual analysis of multidimensional data in parallel coordinates without the need for precomputation. As seen with this approach, I believe there is potential for more scalable and computational effective edge bundling methods, possibly advancing real-time interaction with parallel coordinates for visual analytics.

Clustering Quality Clustering quality is another concern, as for many edge bundling techniques the chosen underlying clustering algorithm determines the cluster assignment, making the bundled visualization sensitive to the selection of a suitable clustering algorithm. Choosing the correct clustering algorithm requires expert knowledge, which - given that parallel coordinates find application in various non-computer science fields like finance or medicine - restricts usability for domain experts outside of the data visualization research. Future work could further explore techniques like force-based and cost-based methods that work without the need of a clustering algorithm, as we have seen in the energy minimization approach by Zhou et al. [16]. By reducing dependency on predefined clusters, such methods could make PCP bundling more accessible across diverse fields.

Edge bundling in parallel coordinates could further benefit from the integration of other visualization tools [11]. Existing research in graph visualization suggests promising interactive techniques, such as *EdgeLens* developed by Wong et al. [14] that could be adapted for PCPs.

While individual studies have analyzed the benefits of specific bundling techniques, the field lacks a standardized comparative framework, such as introduced by Lhuillier et al. [10] for graph visualization. Developing a similar framework could help formalize the comparison of different edge bundling techniques for parallel coordinates, making the decision of the edge bundling technique for a given application clearer and more transparent.

7 Conclusion

There are several different edge bundling techniques applied to parallel coordinates, each with its own specific focus. Many techniques leverage Gestalt principles such as continuity, proximity, and similarity by transforming polygonal strips into smooth curves to create well-structured and clear visualizations. However, with a clear taxonomy still missing, this work introduces a new way of categorizing edge bundling techniques, primarily based on the process of bundling and the similarity measure used to form them. *Data-centric* approaches typically rely on data clustering and focus on similarities within the data itself, while *emerging* bundling methods dynamically create bundles throughout the process. Depending on the specific application, the edge bundling technique can be chosen accordingly, and it is hoped that the overview provided by this work will aid in simplifying the decision-making process.

Looking into the future, I suggest that the visualization community rethinks how overplotting and visual clutter are approached. Rather than viewing visual clutter merely as something to eliminate, we should recognize it as an inherent part of data analysis and further develop techniques to unlock its full potential. Parallel Coordinate Plots, especially when enhanced with edge bundling, offer unique opportunities for the visual analysis of complex datasets. Edge bundling techniques, particularly when applied interactively, can help clarify relationships between dimensions and clusters, thereby enhancing users' understanding of the data and supporting the data analysis process. With further research and development, edge bundling could become a crucial tool for navigating through the overwhelming complexity of modern datasets, enhancing parallel coordinates with clarity and structure. Last but certainly not least, in the era of Big Data, establishing clarity isn't just helpful, it's essential.

References

[1] W. Cui, G. Strazdins, and H. Wang. Visual analysis of multidimensional big data: A scalable lightweight bundling method for parallel coordinates. *IEEE Transactions on Big Data*, 9(1):106–117, 2023.

[2] G. Farin. *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann, 2001.

[3] M. Graham and J. Kennedy. Using curves to enhance parallel coordinate visualisations. In *Proceedings on Seventh International Conference on Information Visualization, 2003. IV 2003.*, pages 10–16, 2003.

[4] J. Heinrich, Y. Luo, A. E. Kirkpatrick, H. Zhang, and D. Weiskopf. Evaluation of a bundling technique for parallel coordinates, 2011.

[5] J. Heinrich and D. Weiskopf. State of the art of parallel coordinates. *Eurographics (state of the art reports)*, pages 95–116, 2013.

[6] J. Heinrich and D. Weiskopf. Parallel Coordinates for Multidimensional Data Visualization: Basic Concepts . *Computing in Science & Engineering*, 17(03):70–76, May 2015.

[7] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):741–748, 2006.

[8] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1:69–91, 1985.

[9] S. G. Kobourov, T. Mchedlidze, and L. Vonessen. Gestalt principles in graph drawing. In E. Di Giacomo and A. Lubiw, editors, *Graph Drawing and Network Visualization*, pages 558–560, Cham, 2015. Springer International Publishing.

[10] A. Lhuillier, C. Hurter, and A. Telea. State of the art in edge and trail bundling techniques. *Computer Graphics Forum*, 36(3):619–645, 2017.

[11] K. T. McDonnell and K. Mueller. Illustrative parallel coordinates. *Computer Graphics Forum*, 27(3):1031–1038, 2008.

[12] G. Palmas, M. Bachynskyi, A. Oulasvirta, H. P. Seidel, and T. Weinkauf. An edge-bundling layout for interactive parallel coordinates. In *2014 IEEE Pacific Visualization Symposium*, pages 57–64, 2014.

[13] K. Watanabe, N. Sakamoto, J. Nonaka, and Y. Maejima. Angular-based edge bundled parallel coordinates plot for the visual analysis of large ensemble simulation data, 2022.

[14] N. Wong, S. Carpendale, and S. Greenberg. Edgelens: an interactive method for managing edge congestion in graphs. In *IEEE Symposium on Information Visualization 2003 (IEEE Cat. No.03TH8714)*, pages 51–58, 2003.

[15] H. Zhou, P. Xu, X. Yuan, and H. Qu. Edge bundling in information visualization. *Tsinghua Science and Technology*, 18(2):145–156, 2013.

[16] H. Zhou, X. Yuan, H. Qu, W. Cui, and B. Chen. Visual clustering in parallel coordinates. In *Computer graphics forum*, volume 27, pages 1047–1054. Wiley Online Library, 2008.