

# COMPO\*SED: Composite Parallel Coordinates for Co-Dependent Multi-Attribute Choices

Lena Cibulski, Thorsten May, Johanna Schmidt, and Jörn Kohlhammer

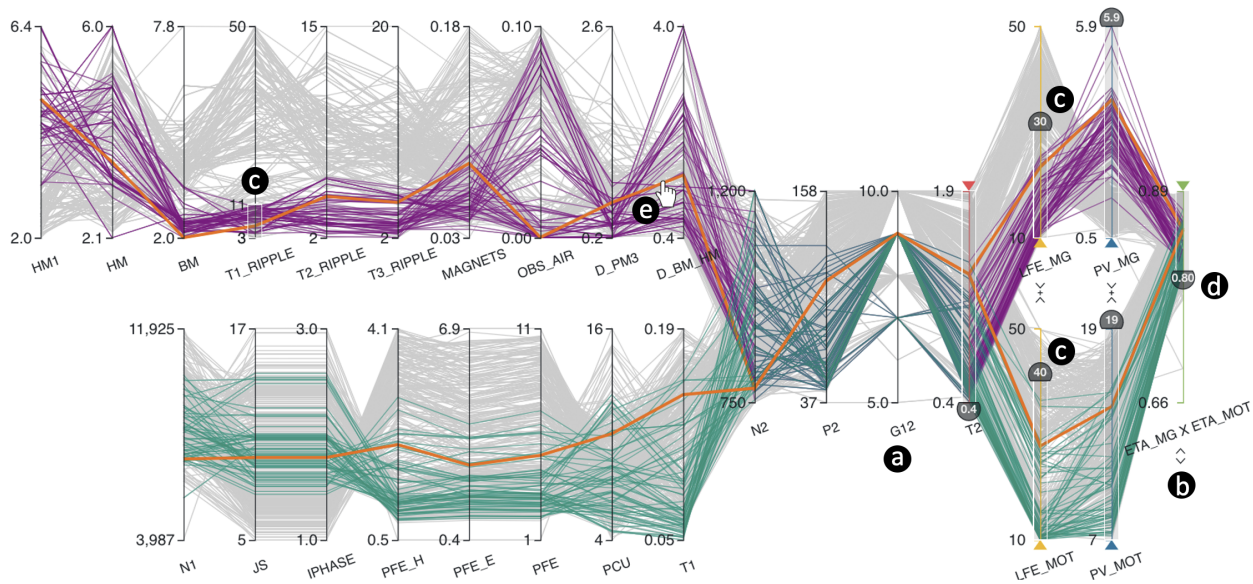


Fig. 1: Composite Parallel Coordinates help users make decisions about multi-component systems, e.g., choose one of the 392 combinations of gear (top) and motor (bottom) variants for an electric drive. Axes can be merged to represent shared properties like the gear ratio (a) or to compute system performance indicators from component criteria (b). Filters can be applied to single components (c) or to the entire system (d). Hovering a gear variant highlights it together with the compatible motor variant (e).

**Abstract**— We propose Composite Parallel Coordinates, a novel parallel coordinates technique to effectively represent the interplay of component alternatives in a system. It builds upon a dedicated data model that formally describes the interaction of components. Parallel coordinates can help decision-makers identify the most preferred solution among a number of alternatives. Multi-component systems require one such multi-attribute choice for each component. Each of these choices might have side effects on the system's operability and performance, making them co-dependent. Common approaches employ complex multi-component models or involve back-and-forth iterations between single components until an acceptable compromise is reached. A simultaneous visual exploration across independently modeled but connected components is needed to make system design more efficient. Using dedicated layout and interaction strategies, our Composite Parallel Coordinates allow analysts to explore both individual properties of components as well as their interoperability and joint performance. We showcase the effectiveness of Composite Parallel Coordinates for co-dependent multi-attribute choices by means of three real-world scenarios from distinct application areas. In addition to the case studies, we reflect on observing two domain experts collaboratively working with the proposed technique and communicating along the way.

**Index Terms**—Visual exploration, multi-criteria decision-making, systems engineering design, parallel coordinates

## 1 INTRODUCTION

Identifying the most-preferred solution among a number of multi-criteria alternatives is a common task in our everyday and professional lives. It is at the core of many real-world decisions related to buying a car, choosing a suitable maintenance strategy, or developing products.

- Lena Cibulski and Thorsten May are with Fraunhofer IGD, Darmstadt, Germany. E-Mail: {lena.cibulski | thorsten.may}@igd.fraunhofer.de.
- Johanna Schmidt is with VRVis Research Center, Vienna, Austria. E-mail: johanna.schmidt@vrvis.at.
- Jörn Kohlhammer is with TU Darmstadt and Fraunhofer IGD, Darmstadt, Germany. E-mail: jörn.kohlhammer@gris.tu-darmstadt.de.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxx/TVCG.201x.xxxxxx

Solution candidates are often generated by simulating the subject under investigation with varying design parameters. A lot of interactive visualizations have been proposed to help decision-makers deal with the resulting sets of alternatives, conflicting criteria, and informal subjective preferences [35], with the parallel coordinates plot [18] being an important representative. These visualizations address the analysis of single units. However, many subjects to decide upon are systems, which consist of multiple components being operated together. In this article, we propose a novel variant of parallel coordinates to enable co-dependent multi-attribute choices.

The key challenge in multi-attribute choices is the presence of conflicting criteria, which raises the need to decide on suitable trade-offs. This becomes even more challenging when targeted at a system rather than a single unit. With a system, the task of making a multi-attribute choice turns into a series of co-dependent choices, one for each component involved. At the same time, component-specific choices need to consider the side-effects on the entire system's operability and performance. Making trade-offs thus extends beyond one single unit.

For example, let us consider a professional photographer who wants to buy new equipment for taking portraits. She needs to decide on a camera body and a suitable lens such that their interplay satisfies her requirements regarding portrait photography. For this decision, she needs to be aware of two aspects: 1) deciding for a camera body restricts her lens choice due to different mount types, and 2) the quality of the camera body and lens only shows in their joint performance.

When facing such a task, deciding on each component independently is not an option because the resulting parts might not be interoperable. Even if they were, component-wise optimality would not guarantee a globally optimized system performance due to emergent effects. Both problems could be solved by using a multi-component model to represent the entire system. However, this means increased model complexity and computational efforts because unchanged partial simulation results cannot be reused [5].

We aim to avoid the drawbacks of these approaches while retaining their advantages. To strike a balance between these two ends of the spectrum, we propose to take into account both 1) the component level via providing the alternatives of individual components as well as 2) the system level via linking the component alternatives according to interoperability and performance aspects.

One approach considering both levels is an iterative optimization, where the decision-maker observes one system component at a time. However, iterative multi-criteria exploration of a system can become a tedious and, at times, frustrating trial-and-error process. In each iteration, the most-preferred alternative for the component under investigation is chosen (component level). Its properties are then considered in the subsequent iterations to evaluate the interoperability and overall performance (system level) for the multi-attribute choices of the remaining components. With an iterative optimization, decision-makers need to make their way through many back-and-forth iterations until they reach a desired system design. As the components are visited one after the other, decision-makers also need to think multiple steps ahead to anticipate the implications of their current choice in the following iterations. As a consequence, analysts might tend to proceed with the first working solution rather than striving to find better designs [5].

In this paper, we present Composite Parallel Coordinates, a technique built upon the well-known parallel coordinates plot [18] to help decision-makers choose the most-preferred design variant of a multi-component system. Unlike an iterative trial-and-error process, it supports a simultaneous exploration of multiple components. The visualization combines superimposition and juxtaposition strategies to depict both the individual properties of an alternative at the component level as well as the dependencies between alternatives of different components at the system level. Dedicated interaction patterns strengthen the perception of component relationships and support efficient navigation through the usually large solution space. In this way, Composite Parallel Coordinates enable decision-makers to perform a series of co-dependent component-specific choices simultaneously and directly take into account the effects of each decision on the system interoperability and performance. In particular, the contributions of this paper include:

- A characterization of co-dependent multi-attribute choices including an appropriate data model.
- COMPO\*SED, a novel parallel coordinates technique that enables a simultaneous exploration of linked alternative ensembles.
- Validation scenarios taken from three distinct fields of application that showcase the use of Composite Parallel Coordinates.

The remainder of this paper is organized as follows. We characterize the analytical problem associated with system design in Section 2. This includes a data model considering the interaction of components. Next, related work is discussed in Section 3, followed by the visualization design and interaction patterns in Section 4. In Section 5, we report on three validation scenarios that indicate the usefulness of the proposed technique. Finally, we discuss limitations in Section 6 and provide conclusions and directions for future research in Section 7.

## 2 PROBLEM CHARACTERIZATION

Many notions of what a *system* is have been expressed in the literature [29]. Most of them describe a system as a collection of components that jointly perform a function to achieve a common objective. Each component is typically designed by dedicated specialists who focus on optimizing its individual characteristics [29]. A central task in systems engineering is, therefore, to establish a balance or even a symbiosis [51] among the various components.

The essential – and challenging – characteristics of a system are its combinatorial nature and the interactions between its components that lead to emergent properties. Emergent properties are properties of the system that the individual components do not possess when acting separately [42]. The characteristics pose three major challenges for systems design:

- CH<sub>1</sub>: Combinatorial optimization** typically entails a huge solution space, even if restrictions apply. This prohibits an assessment of all possible system designs.
- CH<sub>2</sub>: Interoperability constraints** restrict how components can be connected in a system. Consequently, individually optimal component alternatives might not be interoperable.
- CH<sub>3</sub>: Emergent properties** make the system performance difficult to derive from individual component performances. In particular, local optimality might not yield a global system optimum.

With the term *systems design*, we denote the process of determining a combination of interacting components that optimizes the emergent system performance with respect to a number of objectives. Carlson-Skalak et al. introduced the term *catalog design* [5] for a two-stage process that consists of 1) specifying a system configuration [36], i.e., an arrangement of generic components, and 2) instantiating this configuration by selecting particular component variants from catalogs. In this work, we address the second stage, where the generic components are instantiated in a way that optimizes the system performance while the configuration does not change.

### 2.1 System-Oriented Data Model

The subject under investigation is a fixed set  $C = \{C_1, \dots, C_k\}$  of components that together form a system. Each component is optimized individually based on multi-run simulation, resulting in one ensemble per component. Each member of a component ensemble is a variant of this component. Although the components seem independent at first, they need to be integrated to achieve the purpose of the system. Thus, the ensembles cannot be analyzed separately. Instead, an optimal combination of component variants requires a consideration of two levels: the component and the system level (Figure 2).

#### 2.1.1 Ensemble Data at the Component Level

Independent of their role in a system, component variants are described by a set of properties. An ensemble of component variants results from multi-run simulation describing the behavior of a component under different input settings. The simulation model approximates a function that maps some design parameters  $X = \{X_1, \dots, X_n\}$  to some criteria  $Y = \{Y_1, \dots, Y_m\}$ . For each criterion, the desired direction of change (minimization or maximization) is given as metadata. We refer to the design parameters and criteria as variables. Sampling the design parameter space yields an ensemble of design options  $\mathbf{x} = (x_1, \dots, x_n)$ ;  $x_i \in X_i$ . For each design option, the simulation results in a performance vector  $\mathbf{y} = (y_1, \dots, y_m)$ ;  $y_i \in Y_i$ . We refer to the union  $\mathbf{v} = (\mathbf{x}, \mathbf{y})$  of a design option and its simulated performance as a *component variant*. All variants together form a component ensemble  $V$ .

#### 2.1.2 Dependencies between Ensembles at the System Level

At the system level, the variants available for each component are put into the system context to account for their interoperability (**CH<sub>2</sub>**), i.e., components can only be connected under certain conditions, and emergent properties (**CH<sub>3</sub>**), i.e., system performance as a synergy of component performances. The following formal description reflects these dependencies in the data model.

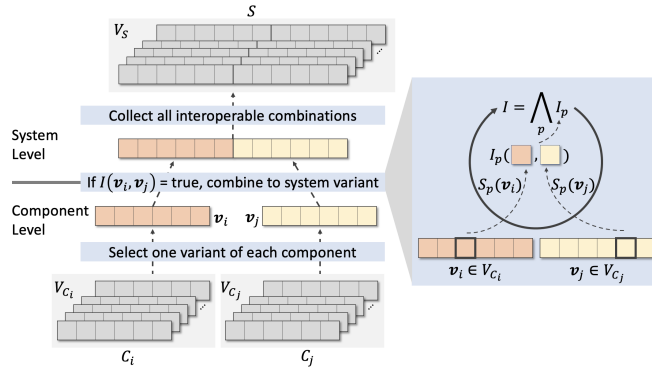


Fig. 2: The system-oriented data model, going from the component level (bottom) to the system level (top). Only interoperable component variants (right) can be combined. The resulting system variants are augmented with global criteria approximating their emergent performance.

**Interoperability** To form a smoothly operating system, each individual component needs to fit its neighbors, physically and functionally. Mittal and Frayman use the idea of ports to describe such intercomponent boundaries [36]. A port is where a component connects to other components. Since we do not assume arbitrary connectivity, a port is also associated with constraints. For example, a lens can only be mounted on a camera body with a fitting mount type.

To represent ports and their constraints in our data model, we looked at how data sets are joined in relational databases [10]. A join condition specifies whether items from different data sets can be combined into a single type. In our case, items are variants of different components that are combined into a system variant based on an interoperability condition. A system variant is valid if it contains exactly one variant of each component in the configuration. To avoid potentially incomplete system variants, we use inner joins to model the interoperability of components. Inner joins consider a tuple of component variants as a system variant, if and only if all variants match the given condition.

Consider two components  $C_i$  and  $C_j$  with variant ensembles  $V_{C_i}$  and  $V_{C_j}$ . The inner join computes an ensemble  $V_S$  of system variants, i.e., combinations of interoperable variants from  $C_i$  and  $C_j$  (Figure 2):

$$V_S = \{(\mathbf{v}_i, \mathbf{v}_j) \in V_{C_i} \times V_{C_j} \mid I(\mathbf{v}_i, \mathbf{v}_j) = \text{true}\} \quad (1)$$

The interoperability condition is represented by a generic predicate function  $I$ . As interoperability is concerned with design space restrictions,  $I$  is evaluated on the components' design parameters. We detail the definition of  $I$  based on the following assumptions:

- Interoperability might be constrained by more than one port. We distinguish the involved ports by an index  $p$ . All defined port constraints must be met for  $I$  to yield *true*.
- Each port constraint is described by a predicate  $I_p$  that is defined on two design parameters, one of  $C_i$  and one of  $C_j$ . We write  $S_p(\mathbf{v}_i)$  and  $S_p(\mathbf{v}_j)$  to select the values of the two design parameters from the respective component variants  $\mathbf{v}_i$  and  $\mathbf{v}_j$ .
- The predicate  $I_p$  of each port is selected from a class of boolean functions during the specification of the system configuration.

Given the first two assumptions, we define the predicate function  $I$  as the logical AND operation ( $\bigwedge$ ) of all individual port predicates  $I_p$ :

$$I(\mathbf{v}_i, \mathbf{v}_j) = \bigwedge_p I_p(S_p(\mathbf{v}_i), S_p(\mathbf{v}_j)) \quad (2)$$

By specifying the predicates  $I_p$  and the design parameters they operate on,  $I$  can be chosen to account for a variety of interoperability conditions in various domains. The possible functions for  $I_p$  can be of two different types: the two design parameter values are directly compared, or an aggregation is computed and compared to a constant  $c$ .

$$I_p(x, y) = \begin{cases} x \bullet y, & \text{(direct comparison)} \\ x \odot y \bullet c, & \text{(comparison with constant)} \end{cases} \quad (3)$$

Above,  $\bullet \in \{=, \neq, >, \geq, <, \leq\}$  and  $\odot \in \{+, -, *, /\}$ . Thus, Equation 3 describes 30 functions that can be used to define the most common port constraints during the system configuration. A natural join or a theta-join [10] is implemented through a direct comparison (natural join) or a comparison with a constant (theta-join) on the relevant design parameters. For example, for the natural join of camera bodies and lenses we can choose  $I_p$  to operate on the components' design parameters *mount type* using "=" as the comparison operator  $\bullet$ . Depending on the interoperability conditions for a system design problem, the definition can be extended with custom predicates if they are using exactly one design parameter of each component. In this way, our technique generalizes to a wide variety of applications.

**Emergent Properties** This dependency between components refers to the system performance as an emergent property of a combination of component variants. We distinguish between local criteria capturing the individual component performances and global criteria capturing the emergent system performance of valid system variants. Following the formal characterization of Weidele [50], this can be described as conditional data: if component variants meet the interoperability predicate, the resulting system variant can be augmented with details about the system performance. Instead of costly simulations, we approximate the system performance by computing global criteria from selected local criteria. A computation based on semantically related local criteria may involve simple mathematical operations, like adding up individual component costs to a system price but may also use more complex functions to address non-trivial compositions. However, the global criteria can also be computed by a conventional weighting approach to aggregate local criteria without any semantic relationship.

## 2.2 Task and Requirement Analysis

The design of engineered systems is relevant in a variety of domains, each providing different environments regarding specifications, domain knowledge, and existing workflows. In this work, we focus on the common difficulties and needs that are associated with system design problems where the arrangement of generic components, i.e., its configuration, is already known.

Wang and colleagues present six analytic tasks related to inferring meaningful information from ensemble data [48]. Their compilation does not cover the main task of system design, which lies in an informed *combination* of members of different ensembles. The primary goal of the analyst is to determine the most preferred combination of component variants, such that the resulting system's performance is optimized with respect to a number of criteria. As there is no inverted simulation model that tells analysts how to choose the design parameters to achieve the desired performance, the available combinations need to be explored [43]. The exploratory choice involves multiple co-dependent decisions: for each component involved in the system configuration, analysts need to select the best option among a finite number of variants, generally referred to as *multi-attribute choice* [12].

With a single unit, *best* depends on an alternative's Pareto dominance and the decision-maker's subjective preferences. When dealing with a system, the superiority of a component variant additionally depends on its interplay with the rest of the system. Herein lies the main challenge of system design. In fact, a central aspect of system design is the "*subordination of individual goals and attributes in favor of those of the overall system*" [29]. Such mutual dependencies significantly intensify the decision-making process. Deciding for the best combination turns into a series of co-dependent multi-attribute choices, whose potential side effects need to be considered when making trade-offs. An exploration across ensembles is thus guided by the following questions:

- T<sub>1</sub> Overview:** What are the value distributions of design parameters and criteria at the component level and at the system level?
- T<sub>2</sub> Competition:** What is the nature of conflicts among criteria? How important are component performances and system performance?
- T<sub>3</sub> Filter:** What is the effect of system and component constraints?
- T<sub>4</sub> Subjective Evaluation:** Is a variant feasible? Does it balance the criteria according to the stakeholder's preferences, tolerances, domain knowledge, and experience? Which variant is superior?

- T<sub>5</sub> Connectivity:** Given a (set of) component variant(s) of interest, which variants of the remaining components are interoperable?
- T<sub>6</sub> Navigation:** How does deciding for a component variant affect the availability of remaining variants and the system performance in reach? Which variant improves the emergent system properties?
- T<sub>7</sub> Key Component Variants:** Which component variants yield a good performance, while being interoperable with many other variants?
- T<sub>8</sub> Partial Choice:** Does a partial component choice, i.e., selecting a set of possible component variants, leave enough room for balancing the properties of the remaining components?
- T<sub>9</sub> Variant Replacement:** How does a component variant contribute to the system performance? What is the effect of replacing it? Which replacement increases component or system performance?

### 3 RELATED WORK

For depicting multi-attribute variants of a single component, we can draw from existing works on multivariate Pareto front visualization (Section 3.1). Jointly analyzing variant ensembles of more than one component relates to visualization approaches addressing a simultaneous investigation of different but related datasets (Section 3.2). Since we use parallel coordinates, we also investigate related approaches for organizing parallel coordinates axes (Section 3.3).

#### 3.1 Multivariate Pareto Front Visualization

Visualizations for multi-criteria decision-making can be categorized based on the cardinality of the result set, i.e., visualizing a single solution, a finite solution set, or an infinite set of solutions [27]. Our data model even involves multiple finite solution sets. Dimension reduction and lossless projection are two common strategies for visualizing multi-dimensional criteria in two-dimensional visual space [33].

Dimension reduction approaches provide a dense representation of a virtually arbitrary number of dimensions. They have been used to visualize Pareto fronts using self-organizing maps [7] or t-SNE [52]. However, they are usually hard to read and interpret, which impedes a considerable number of analysis tasks that rely on raw values.

In contrast, a lossless projection enables decision-makers to visually retrieve any criterion value of any alternative without interaction [12]. To simplify multi-attribute choice, tabular visualizations have been extended with weight-based ranking [15]. However, it is difficult to adequately capture preferences by weights. Scatterplots visualize trade-offs between two criteria [37] but can be extended to more criteria by composing them into a scatterplot matrix. Still, the perception of multi-criteria options is limited to pair-wise projections.

To date, the predominant technique to visualize Pareto frontiers are parallel coordinates [18]. They present a compact two-dimensional visual representation that allows for a comparison of alternatives across all design parameters and criteria. Different layouts [22], axis modifications [2], and extensions [6, 11] have been proposed. Among others, parallel coordinates have been applied to design problems in automotive and aerodynamic engineering [25], also in virtual reality settings [46]. In their survey, Heinrich and Weiskopf define a composite parallel coordinates plot as a composition of several visualization layers, e.g., axes and brushes [17]. In contrast, we propose a side-by-side composition of multiple parallel coordinates plots, which emphasizes the dependencies and emergent properties in multi-component systems.

#### 3.2 Visualization of Multiple Related Data Sets

System design requires an investigation of related data sets representing components and their interactions. Konyha and colleagues conclude that single table approaches are insufficient to describe such data and its dependencies [26]. This view is shared by Kehrer and Hauser, who identify multi-model scenarios resulting in two or more interacting data parts as a promising direction of visualization research [23]. A central question in such cases is how to investigate patterns across data sets.

Coordinated multiple views may link multiple tables via primary and foreign keys, for example, in *Snap-Together* by North and Shneiderman [39]. Liu et al. consider the relationships between data items as a graph and propose the system *Ploceus* as a network-based visual analysis [32]. With *Domino*, Gratzl et al. propose a meta-visualization technique

allowing users to create explicitly linked views to represent data subsets and four degrees of relationships between them [14]. Working with multi-resolution models, Splechna et al. address the complication of only partially overlapping parameter spaces as a key challenge [45].

The links between our component data sets are not defined by shared identifiers but by value predicates. Kehrer et al. propose a similar abstraction of the relation between two data sets, which they call *interface* [24]. Their abstraction addresses multi-model scientific data in a spatial domain, which can be exploited to describe the relations via co-location. Closest to our approach is the work by Splechna et al., who propose to relate items of different data sets based on their properties regarding one or multiple (common) attributes [44]. While we build upon such conditions to represent the interoperability of components in a system, their approach cannot be used for emergent properties.

The analysis of emergent properties plays a significant role for the optimization of complex engineered systems. Basole et al. propose a network-based visual analytics tool for system design that explicitly considers how intermediate decisions influence system-level properties [3]. While their approach focuses on an iterative reconfiguration of the system, our approach is based on systematic sampling of the design space upfront to gain a broad overview early in the process. Closest to our work, Marth et al. use scatter plots to evaluate the joint performance of a motor and a gear in a side-by-side arrangement [34]. They provide performance criteria for individual motor and gear variants as well as for their combinations by summing up the individual criteria (e.g., the sum of losses or lengths). We generalize the specification of interoperability and joint performance and visualize these system-level properties together with component-level properties. So far, no multi-ensemble approach has been proposed that allows for a simultaneous exploration and optimization of individual ensemble members as well as conditional combinations of ensemble members.

#### 3.3 Axis Configurations for Parallel Coordinates

The core challenge of our parallel coordinates composition is the arrangement of axes. In the conventional layout, every dimension has two direct neighbors. A strategy to overcome this limitation for visualizing many pair-wise relations is to replicate axes of individual dimensions. Lind et al. combine multiple axis orderings in a many-to-many parallel coordinates plot [31]. Replicated axes depicting the same dimension are arranged in polygons to communicate all pair-wise relations of dimensions in a non-overlapping way. Claessen and van Wijk propose a generalization where axes can be positioned freely and linked via scatter plots (orthogonal axes) or parallel coordinates (parallel axes) [9]. We did not consider such an approach because it results in a complex visualization layout, even for single components.

We want particular relations across components to stand out. Multiple strategies have been proposed to visually aggregate dimensions with similar semantic meanings. Andrews et al. introduce aggregate axes that replace related dimensions by substituting the dimension values with their mean [1]. Axes can be interactively collapsed and expanded. Bhattarai and colleagues use the sum to merge dimension axes for an exploration of material compositions [4]. Garrison et al. aggregate dimensions that contribute similarly to the variance of a dataset by mapping parallel coordinates axes to the first and second principal components of the dimensions [13]. In their product comparison tool *ConfigurationFinder*, Riehm and colleagues organize semantically related dimensions in groups that are represented by an expandable proxy axis [40]. The approaches mainly differ in how the related dimensions are identified (domain knowledge or automated analysis) and how the aggregations are calculated and presented. In our case, related dimensions of different components are derived from the analyst's knowledge. Their axes can be merged using different functions to depict system-level properties. A primary challenge with our approach is to represent the interoperability conditions.

*Conditional parallel coordinates* by Weidele use predicates to insert nested axes for conditional dimensions that apply only to data sets satisfying specific properties [50]. While this approach allows for representing items of different types in one view (e.g., cameras and lenses), it does not enable a combination of items of two or more types.

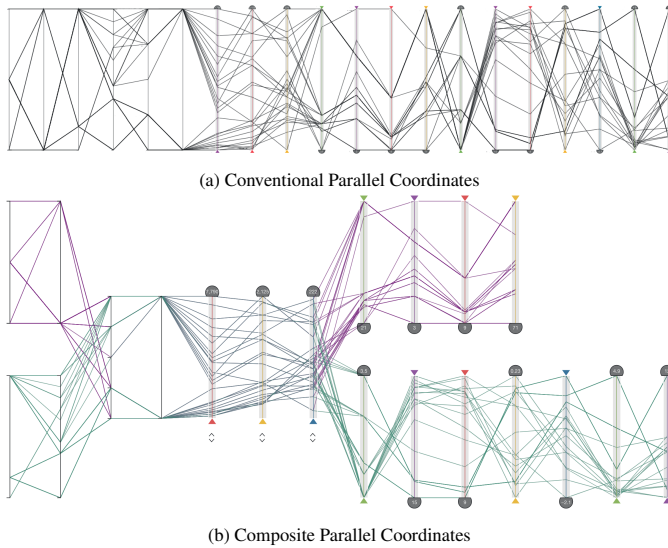


Fig. 3: Camera-lens systems depicted by (a) conventional parallel coordinates (PC) vs. (b) the proposed Composite Parallel Coordinates (ComPC). PC display the combinations over their joint variable space, while ComPC partially juxtapose cameras (purple) and lenses (green). In contrast to PC, they explicitly visualize the combinatorial problem.

In our approach, all dimensions can be shown from the beginning because the predicates imply combinatorial constraints, leaving the dimension schema unchanged. In particular, axes unique to one type (i.e., component) are shown at the same level of detail as shared axes.

Regarding the ambition to visualize multiple ensembles, most similar to our approach are the nested parallel coordinates proposed by Wang et al. [49]. They use nested axes to compare data distributions from multiple ensembles that originate from climate simulation at different resolutions. Our Composite Parallel Coordinates are inspired by their approach, i.e., to assemble juxtaposition and superimposition for analysis within and between different ensembles. However, while their ensembles provide different resolutions of the same subject, the components represented by our ensembles are different subjects.

#### 4 DESIGN OF COMPOSITE PARALLEL COORDINATES

Based on our formal characterization of system-oriented data, we introduce a novel variant of parallel coordinates for their visualization. It allows analysts to explore the possible combinations of component variants while taking into account both the individual component properties as well as their emergent system properties. Parallel coordinates offer a compact and lossless two-dimensional visual representation for multi-dimensional observations. We made them the basis of our visualization design primarily for their lossless mapping and flexible axis arrangements [9] but also for their simple applicability and wide-spread use in multivariate data exploration.

System design requires an observation of design options both at the component level and at the system level. This leads to three conflicting visualization design goals ( $G_1$ ,  $G_2$ ,  $G_3$ ) and one independent goal ( $G_4$ ):

- $G_1$ : Component-level analysis** requires a stand-alone observation of individual properties per component. Without prior relevance information, all components and properties are considered equally meaningful for analysis.
- $G_2$ : Context awareness** requires to relate observed component properties to semantically similar properties of other components.
- $G_3$ : System-level analysis** needs an observation of similar properties across components. The evaluation of interoperability and emergent performance benefits from explicit system properties.
- $G_4$ : Layout stability** is an overarching design goal. In contrast to open data exploration scenarios, where no analysis strategy is imposed, system design relies on a clear mental model of the

system structure and properties to investigate. A stable overview layout allows analysts to focus on trade-offs instead of adapting to varying positions of components and their properties.

Conventional parallel coordinates lack the ability to depict the dependencies between individual ensemble members (Section 4.1). We therefore propose Composite Parallel Coordinates, whose axis layout (Section 4.2) and interaction patterns (Section 4.3) reflect the notion of a system being a composition of interacting components.

#### 4.1 Reviewing Conventional Parallel Coordinates

A conventional parallel coordinates plot depicts a single multi-dimensional dataset, where all items are defined in the same variable space. To visualize multiple component ensembles, their different variable spaces need to be merged already during the data transformation step. This can be achieved by joining the component variants according to their interoperability. Due to the combinatorial nature, the number of items and variables to visualize increases drastically (upper bound  $n * m$  for items or  $n + m$  for variables). Conventional parallel coordinates then result in a plot with many side-by-side axes, where the polylines represent the ensemble  $V_S$  of complete system variants (Figure 3a).

In this plot, the subdivision of system variants into individual component variants is not obvious. This makes it difficult to perceive how the properties of the system originate from the interactions between the individual components. It complicates the central aspect of system design, being the subordination of component-specific characteristics in favor of the system performance [29]. The root of this complication is the axis layout being restricted to the horizontal direction.

On the one hand, the success of a choice at the component level is determined by the individual component properties ( $G_1$ ). This is easier to evaluate if all axes belonging to the same component  $C_i$  are placed directly next to each other, such that each component variant  $v$  is represented as a self-contained polyline. This axis order supports tasks like gaining an overview of component variants ( $T_1$ ), determining key component variants ( $T_7$ ), or replacing a component variant ( $T_9$ ).

On the other hand, determining the success of an intermediate choice builds upon an evaluation of emergent system properties ( $G_2$  and  $G_3$ ). This can only be achieved by placing the involved axes belonging to different components  $C_i$  and  $C_j$  directly next to each other. Such an axis order supports tasks like determining interoperability ( $T_5$ ), evaluating the system performance ( $T_4$ ), or navigating the combinatorial design space ( $T_6$ ). However, it contradicts the component-wise adjacency of axes required for evaluating the choice at the component level.

Using conventional parallel coordinates,  $G_1$ ,  $G_2$  and  $G_3$  can only be achieved if we allow the axis order to be interactively adjusted to the varying analysis focus. However, this would mean violating the requirement for layout stability ( $G_4$ ).

To summarize, conventional parallel coordinates are not suited to meaningfully represent both component membership ( $CH_1$ ) as well as interoperability ( $CH_2$ ) and system performance ( $CH_3$ ). This boils down to them being restricted to depicting a single ensemble of system variants using a one-directional axis layout. Parallel coordinates cannot communicate the dual role of component variables that contribute to both component-specific properties as well as system-wide properties.

#### 4.2 Layout of Composite Parallel Coordinates

As explained previously, conventional parallel coordinates do not allow for an understanding of how the components work together. This issue can be solved by visualizing the components individually instead while paying particular attention to the interfaces between them. We map the component ensembles to distinct parallel coordinates plots. Depicting their interfaces poses an inherent challenge when visualizing a system as a composition of components. It requires careful integration of the different parallel coordinates plots into the same view. Javed and Elmquist define this approach as *composite visualization* [21]. We make use of their design space to convey the idea of Composite Parallel Coordinates. Our visualization design was guided by the following question: how to make component properties ( $G_1$ ), their context ( $G_2$ ), and system properties ( $G_3$ ) equally accessible in a stable layout ( $G_4$ )?

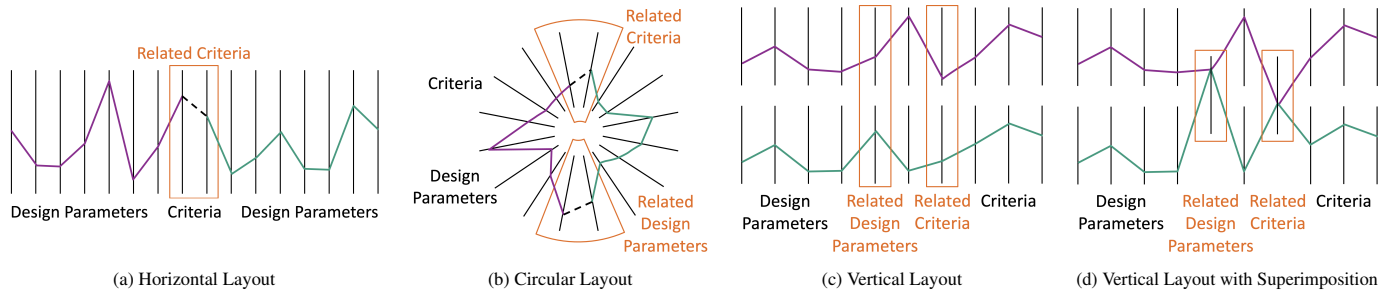


Fig. 4: Different layout strategies to represent two components and their dependencies. Related properties of different components should be depicted close to each other. (a) Concatenated PCPs allow for adjacency of one pair of related properties. (b) Bending them to a circle adds another adjacent pair of related properties. (c) A vertical arrangement allows for more than two pairs of properties with similar semantics to be placed close to each other. (d) Where applicable, shared and related component properties can be merged to explicitly reflect system properties.

#### 4.2.1 General Visualization Design

A composite visualization is a natural choice for depicting a system of interacting components. In contrast to the concept of coordinated multiple views [41], where different visualizations depict different aspects of the same data items, our composition involves multiple instances of the same visualization to depict different but related data items. The reason is that the primary task of making a multi-attribute choice is the same for each component in the system to be designed. Each system component ensemble  $V_{C_i}$  is depicted by one parallel coordinates plot.

How do these views become part of a composite visualization? Data-wise, they are independent because the variable spaces  $X \cup Y$  are different for each component. The views' dependency originates from the domain-specific semantics regarding interoperability and emergent properties (see Section 2.1.2). The visual design task is to communicate these implicit dependencies as explicitly as possible.

The design space of composite visualizations proposes two symmetric and two asymmetric composition strategies [21]. The asymmetric strategies, namely overloading and nesting, impose an imbalance between views, which does not match the inherent symmetry of the system design problem, where all components are considered equally important. Thus, we turn toward the two symmetric strategies: juxtaposition and superimposition (Figure 3b). Juxtaposed parallel coordinates plots address the component level by depicting the largely different variable spaces. Superimposed axes of different plots address the system level by communicating emergent properties like interoperability conditions and overall system performance.

Our strategy accounts for the pairwise relations between the  $k$  variable spaces of the components. It involves different visual mappings to communicate the following parts of a relation between variable spaces:

- *Shared*: two variable spaces share parts where they exhibit common variables. Typically, these variables are design parameters considered for the modeling of interoperability, e.g., *mount type*.
- *Related*: the related parts of two variable spaces contain those non-common variables that contribute to interoperability and emergent properties (see Section 2.1.2). Related variables can be design parameters or criteria, such as *camera price* and *lens price*.
- *Unique*: those parts of a variable space that neither involve common nor related variables are unique, e.g., *lens focal length*.

Below, we describe our design choices regarding these visual mappings.

#### 4.2.2 Juxtaposition for Component Level

At the component level, the decision-maker focuses on the individual properties of one component at a time ( $G_1$ ). Besides context awareness, considerations that involve other components, like interoperability and system performance, are of secondary importance. The simplest way of presenting an overview of all component ensembles is a juxtaposition of separate visualizations. Due to their independence, they allow analysts to focus on individual components without interference or distraction. As all components are considered to be equally important, we symmetrically divide the available visual space.

Juxtaposed views are generally highly flexible regarding their arrangement. However, in our case, the layout quality particularly depends on its ability to display semantically related properties of different components spatially close to each other in order to maintain the system context ( $G_2$ ). The value of juxtaposition then stems from the boundary between two views conveying shared semantics. In the following, we discuss different layout options in light of this aspect.

Let us consider the composite visualization of two component ensembles  $V_{C_i}$  and  $V_{C_j}$ . A naive approach would be to horizontally concatenate the two parallel coordinates plots (Figure 4a). Perceiving the system context requires the related properties of both components to be depicted close together. To achieve adjacency, the respective axes are placed at the inner ends of the plots. However, only a single pair of design parameters or criteria is adjacent, meaning that only one interoperability condition  $I_p$  or emergent property can be communicated.

The second option is a circular layout (Figure 4b). The related design parameters are placed at the inner boundary of the concatenated plots and the criteria contributing to emergent properties at their outer ends. The plots are then bent to a circle, such that the criteria, too, are adjacent. The result resembles a radar chart. Still, this layout conveys only one interoperability condition  $I_p$  together with one emergent property and cannot be extended to more than two components.

A third option is to arrange the parallel coordinates plots vertically (Figure 4c). The vertical distribution clearly separates the visual representations of individual components, thus enabling an efficient perception of the component level. The horizontal direction can then be exploited to position variables with similar semantic meanings but belonging to different components one below the other. In this way, the boundary between two views conveys shared semantics via multiple interoperability conditions and emergent properties and properly accounts for context awareness. In addition to that, the layout offers the potential to be extended to more than two components.

Based on the requirements imposed by the component-level analysis ( $G_1$ ) and context awareness ( $G_2$ ) together with the overarching layout stability ( $G_4$ ), the juxtaposition with vertical layout is the most promising option to proceed with. While the boundary between views conveys the system context via shared variable semantics, any linking between data items of different views is revealed only upon user interaction. Relations between variants  $\mathbf{v}_i \in V_{C_i}$  and  $\mathbf{v}_j \in V_{C_j}$  of different components are difficult to perceive. These relations describe the interoperability of component variants as well as their joint performance. They refer to a system-level analysis ( $G_3$ ), which is detailed in the following section.

#### 4.2.3 Superimposition for System Level: Composite Axes

Dependencies at the system level manifest in parts of the variable spaces being shared (common variables) or related (variables contributing to interoperability conditions or emergent properties). Superimposition means to overlay two plots in a single view [21]. We implement it by allowing the user to merge those axes that are associated with shared or related parts of the variable spaces (Figure 4d). As a result, the interface of two components is depicted by those polyline sections that

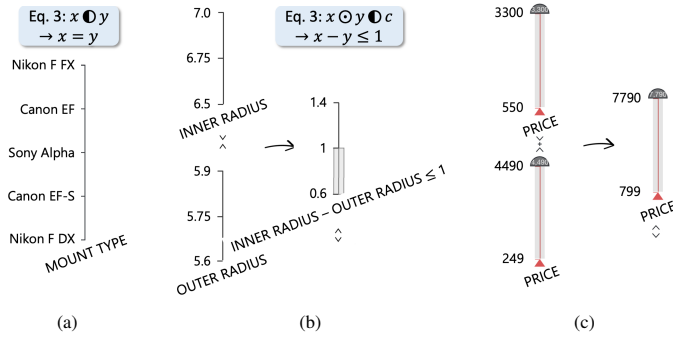


Fig. 5: Composite axes represent interfaces between components. (a) Shared design parameters can contribute to interoperability (e.g., mount types should be equal). (b) Related design parameters specify interoperability via inequality (e.g., radii of components should differ by one unit at most). (c) Related criteria are added up to a system criterion.

intersect the superimposed axes and thus share the same visual space at the boundary between the juxtaposed plots. Unlike any other layout that we considered, this strategy solves the component-level analysis ( $G_1$ ) with context awareness ( $G_2$ ) and the system-level analysis ( $G_3$ ) while providing a stable layout ( $G_4$ ).

The design parameters and criteria that are unique to individual components are not affected by superimposition. They are represented by the juxtaposed atomic axes of the individual parallel coordinates plots because they do not semantically relate to another component's properties. An example is the property *focal length* of camera lenses, which remains unaffected no matter which camera body is chosen.

For the shared and related parts of the variable spaces, we introduce *composite axes*. These axes can take different forms, depending on the represented type of dependency and the current analysis focus. What they have in common is that atomic axes of different plots might be collapsed into a derived axis. It is placed vertically in between the involved juxtaposed views and acts as an independent variable itself.

We distinguish three types of dependencies between components, which are represented by slight variations of composite axes:

- **Shared design parameters** are depicted using a permanently collapsed axis (Figure 5a). They are not expandable, as this would mean to duplicate the axis and thus add redundancy.
- **Related design parameters** are depicted using separate axes initially, but can be collapsed by applying a predicate function describing the interoperability condition  $I_p$  (Figure 5b).
- **Related criteria** are also depicted using separate component axes, but can be collapsed via a mathematical operation that maps the component criteria to a system criterion (Figure 5c).

Composite axes have a button located beneath them to collapse and expand the associated component variables. When two component axes are collapsed, the derived axis is inserted vertically centered between the two original axes, replacing them. The polylines of both involved plots are updated to intersect the collapsed axis. Where non-identical parameters are merged, this requires the creation of combinations. To keep complexity low, the range of the collapsed axis is computed naïvely from the extreme values of the original axes. When the collapsed axis is expanded again, the component axes are inserted at their original position in the plot, replacing the collapsed axis ( $G_4$ ). The polylines are updated again to intersect the separate component axes.

The dependencies, i.e., which parameters can be collapsed and how are prescribed by the particular application domain. The axis pairs and collapse functions are specified a priori by users in the form of metadata of the dataset to be analyzed. Up to now, the users have managed to do so without a dedicated user interface. Still, whenever needed, only development efforts would be required to provide a user interface to not only specify but also adjust the axis pairs and collapse functions.

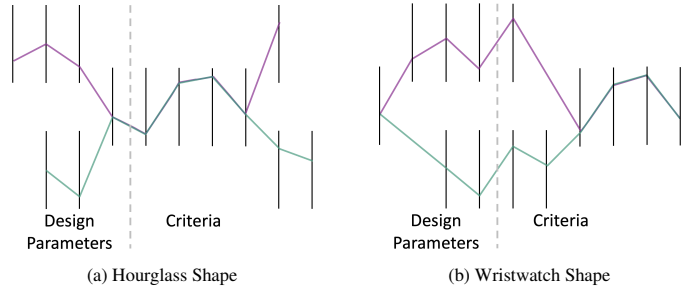


Fig. 6: The axis order follows the input to output mapping. It minimizes alternation between separate and collapsed axes. Unlike the wristwatch shape (b), the hourglass shape explicitly depicts the bottleneck (a).

**Value Mapping** The value mapping of a composite axis depends on its type and collapse function. Composite axes displaying shared design parameters do not require a dedicated value mapping. The permanently collapsed axis simply displays the original parameter values of the variants across both components (Figure 5a).

In contrast to shared parameters, related parameters have the same semantic meaning but are not identical. They are initially observed individually using separate component axes. Based on their semantic relationship, these axes can be collapsed to reflect properties at the system level. This requires the combination of component variants using a dedicated value mapping that derives an aggregated system value from the two original component values. These aggregated values are displayed on the collapsed axis. To specify the exact mapping, we need to distinguish related design parameters and related criteria.

For related design parameters, the collapse function is taken from the pool of interoperability predicates (Section 2.1.2). As an example, the outer radius of a motor rotor is related to the inner radius of the surrounding stator according to an inequality constraint. A predicate  $I_p$  is applied in two steps. First, the aggregate of the two component values is computed using the  $\odot$  operator. In case of a direct comparison, where no operator is involved, the predicate is rewritten to a comparison with constant 0, e.g.,  $x < y \rightarrow x - y < 0$ . For the rotor and stator,  $\odot$  might be defined as subtraction such that  $r_{inner} - r_{outer}$  describes the clearance between both mechanical parts. This aggregate value is displayed on the collapsed axis. Its range is derived from the aggregate values across all combinations of rotor and stator variants. In a second step, the comparison operator  $\bullet$  is applied as a filter. In this case, the clearance should not exceed the value one, so  $\bullet$  is  $\leq$  and any aggregate value less than or equal to one is brushed on the axis (Figure 5b).

As an example for related criteria, the individual prices of a camera body and a lens might be added up to reflect the system price. Criteria can only be collapsed when they are 1) both to be minimized or both to be maximized and 2) associated with the same or relatable units. Upon collapse, any meaningful mathematical operation might be applied to the original values of the two involved component criteria. The collapsed axis then depicts for example the total system costs as the sum of the two component prices (Figure 5c). Its range is computed by applying the same operation to the original minima and maxima of the component axes. This range covers all potential combinations of component variants but is not necessarily exploited.

**Axis Order** The initial order of composite axes is generated by mapping the input-output order of the data model to the reading direction from left to right. The design parameters (input) are placed on the left side of the visualization, while the criteria (output) go on the right side. We can further divide the input and output into unique and shared or related variables. It should be noted, however, that strong alternation between separate and collapsible composite axes is not desired due to the turbulent polyline courses this generates.

Considering this, we can place the separate design parameters left, then the collapsible design parameters and collapsible criteria in the middle, and the separate criteria to the right (Figure 6a). The resulting shape resembles an hourglass. Alternatively, we can place the

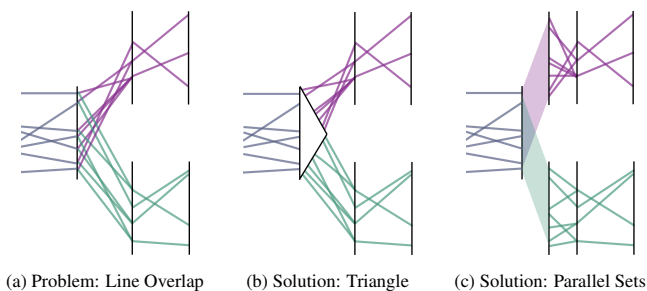


Fig. 7: Unwanted line crossings occur where separated polylines merge into one (a). Redirecting the polylines via duplicates of the collapsed axis using a triangle (b) or parallelograms (c) mitigates the overlaps.

collapsible design parameters to the left, then the separate design parameters and separate criteria in the middle and the collapsible criteria to the right (Figure 6b). Due to the closed ends, this shape looks more like a wristwatch. Within these constraints, the axes are initialized according to the order in which they occur in the datasets. We decide for the hourglass shape, because it explicitly communicates that the *bottleneck* of system design is the interaction between components.

**Handling Line Overlaps** The fusion of component polylines at collapsed composite axes provokes line crossings that can make the course of individual lines hard to trace (Figure 7a). We developed two strategies to cope with this phenomenon.

The first strategy is inspired by the many-to-many parallel coordinates proposed by Lind et al. [31]. They arrange replicated axes of the same variable in a triangular shape to visualize a one-to-three relationship. We implement this concept to handle line crossings by replicating the collapsed axis, arranging the three identical axes in a triangle, and redirecting the polylines from the original collapsed axes via the two replicated axes to the adjacent separate axes (Figure 7b).

The second strategy resembles the parallel sets originally introduced for categorical data [28]. We use parallelograms to show the connection between the original and replicated axes (Figure 7c). The vertical orientation of the replicated axes matches well with the general layout of the parallel coordinates. It explicitly communicates that the polylines split up into the two component levels. The values are easier to read from the axes, and the match between incoming polyline sections at the collapsed axis and outgoing sections at the replicated axes is easier to make. We thus opt for the parallel sets strategy to bypass the line crossings where separate and collapsible composite axes are adjacent.

### 4.3 Interaction Patterns

Interaction is essential for effective use of Composite Parallel Coordinates. It allows analysts to filter the available alternatives according to constraints and preferences as well as emphasize alternatives of interest. Our interaction mechanism involves three cascading selectors to gradually refine a selection of alternatives under focus.

The selectors are hierarchically structured: *filters* take precedence over *locks*, which in turn have priority over *mouseover* interactions. The outcome of any selector is a set of alternatives. Each selector operates on the outcome of the precedent selector: hovered alternatives are a subset of locked alternatives, which in turn are a subset of filtered alternatives. Note that this structure does not prescribe the order in which selectors have to be applied. Selectors are optional: if a selector is not active, the outcome of its precedent selector remains unchanged. At any time, the selection resulting from the cascade of currently active selectors is highlighted, while the remaining alternatives are depicted in grey for context. In the following, we describe the three selectors in the order of their precedence.

#### 4.3.1 Eliminating Undesired Variants Via Filters

We provide filters in the form of range brushes that can be applied to any design parameter or criterion axis in the Composite Parallel Coordinates. As alternatives need to be evaluated regarding multiple constraints

and preferences, multiple brushes can be combined into a composite brush using the logical AND operation. Where brushes represent interoperability conditions  $I_p$ , their composite brush corresponds to the overall interoperability  $I$ . Because it does not make sense to exclude variants that are located at the desired end of an axis, brushes on criteria axes are tied to the high-quality end of an axis [8].

In large parts, filtering works in a standard way: alternatives covered by a brush are included in the selection. However, the combinatorial nature of the optimization gives rise to some special considerations:

- A component variant is selected if it is brushed itself or can be combined with at least one brushed variant of another component.
- A component variant can be brushed itself on a unique axis or as part of a combination on a collapsed axis.
- When two axes are collapsed, the new brush slider position is determined by applying the collapse function to the original slider values. Upon expansion, the original slider values are restored.

Eliminating undesired alternatives via filtering results in a subset of acceptable options to proceed with (Figure 8a).

#### 4.3.2 Highlighting Desired Variants Via Locks and Mouseover

With a potentially large number of acceptable combinations remaining after filtering, users need support in scanning through the filtered alternatives to further refine the selection.

We provide locks and mouseover selection on polyline segments to convey interesting valid combinations of component variants. The atomic unit of an interaction is a system variant  $(v_i, v_j) \in V_{C_i} \times V_{C_j}$ , i.e., a one-to-one combination of component variants. Any interaction taking place on one part of a system variant also applies to the rest of the system variant. Due to the combinatorial nature, multiple system variants might pass through the same polyline segment, in particular where shared axes are adjacent. An interaction with a polyline segment can thus lead to more than one system variant being hit.

To specify the second selector, a filtered polyline segment can be clicked to lock the associated set of combinations, updating the selection to the respective subset of the filtered alternatives (Figure 8c). Only one segment can be locked at a time. A lock is active until it is unlocked (by clicking again) or moved to another polyline segment (by clicking the respective segment, see Figure 8e). Unlocking a lock makes the selection fall back to the superset of filtered alternatives.

To specify the third selector, the selection resulting from the lock can be refined via mouseover (Figure 8d). If there is no active lock, the mouseover operates on the set of filtered alternatives (Figure 8b). The mouseover interaction is temporary: when the cursor leaves the hovered polyline segment, the selection falls back to the set of locked alternatives or to the set of filtered alternatives if no lock is active.

Anything that remains in the selection after applying the current cascade of filters, lock, and mouseover is highlighted. At the end of an analysis, this is usually a unique combination of component variants, i.e., the final system choice (Figure 8f).

## 5 VALIDATION

Composite Parallel Coordinates provide a novel approach to a simultaneous exploration and analysis of multiple interacting datasets. To validate its domain usefulness in terms of problem-solving characteristics, we report on two usage scenarios and one case study [20] from distinct application domains. In the case study, we particularly reflect on observing an analysis conversation between two engineering experts. The results suggest that our technique supports the identified analysis tasks for making co-dependent multi-attribute choices.

These three real-world scenarios showcase how COMPO\*SED helps users simultaneously explore linked component ensembles for the analysis of complex systems. The datasets exhibit different properties regarding unique, shared, and related design parameters and criteria. In all three scenarios, the visualization enabled users to apply constraints and observe their combined effects on both the component and the system level. In particular, it supported decision-makers in investigating how a component-specific choice affects the system performance and the availability of interoperable component variants.



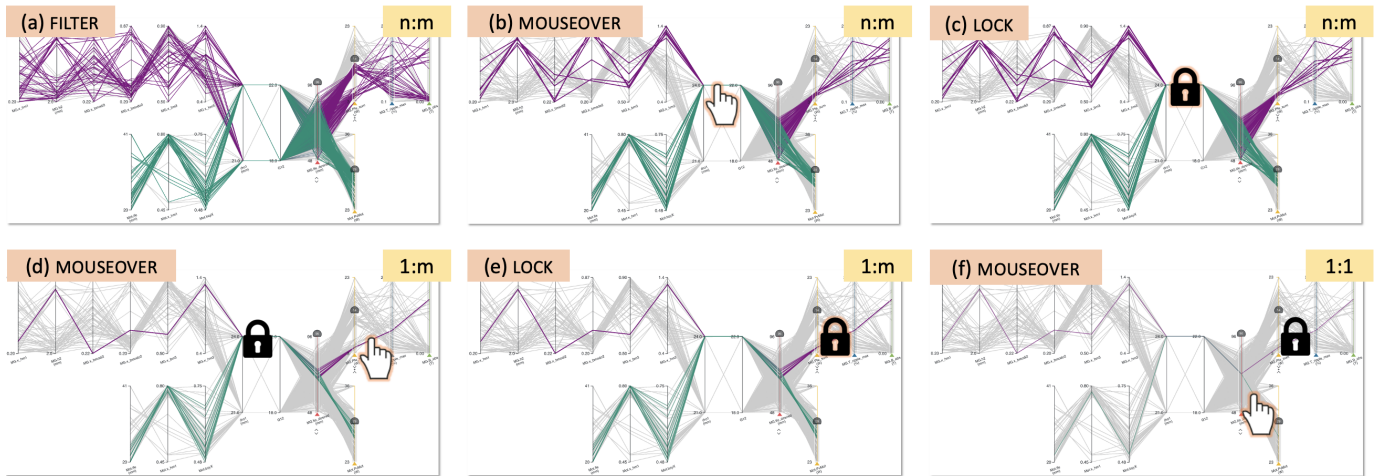


Fig. 8: Selectors are applied at three precedence levels: filter, lock, and mouseover. The resulting selections range from many-to-many over one-to-many to one-to-one combinations. (a) A filter results in a set of alternatives that is refined by (b) a mouseover selecting system variants with a particular diameter and gear ratio. (c) Locking them allows for (d) mouseover exploration of the involved component variants. (e) One gear variant is locked to explore the compatible motor variants. (f) One of them is finally chosen as the best fit.

### 5.1 Usage Scenario I: Power Plant Operation

District heating describes a method for delivering space heating or hot water to buildings via insulated pipe systems. Power plants that serve as heat suppliers mostly fire fossil fuel, biomass, or waste. This usage scenario stems from a long-term collaboration with engineers who are responsible for the operation of a district heating power plant. The power plant is located in Austria, and the data has been anonymized to not reveal ownership. The power plant consists of two neighboring blocks to burn different types of combustion material. One block uses water, while the other block uses air as a cooling material.

Rather than interoperability, it is the joint production of heat that prohibits an optimization of the blocks in an isolated manner. Operators of the power plant can use Composite Parallel Coordinates to decide which block to use to which extent in order to jointly produce a certain amount of heat requested by the operation plan. If the outcome of one block decreases, the outcome of the other block has to be increased. This is typically based on the domain knowledge of operation engineers. Using data analysis for power plant operation is still a novel approach. The engineers only have basic experience in data analysis and data visualization. With the Composite Parallel Coordinates, they can, for the first time, study 1) how environmental conditions influence the possible operation modes of the power plant and 2) how the parameters of one block influence the operation of the second block. The data was generated using a simulation model.

The heat produced by each block depends on various factors. Block-specific design parameters include combustion material, cooling parameters, and the number of active valves. Environmental conditions like temperature, humidity, and air pressure are shared by both blocks because the blocks are equally affected by their changes. The different consumptions and efficiencies associated with the produced heat in each block are related and can be composed into system criteria.

Engineers need to constantly regulate the operation of the blocks during the day. The main trade-off lies in producing a maximum outcome while consuming the least possible amount of combustion material. The goal is to distribute the production of the requested heat to both blocks such that the yield, i.e., the difference between the price for heat on the market and the operating costs, is maximized.

When loading the data into the Composite Parallel Coordinates, every polyline in the plot corresponds to one possible operation mode of the power plant (Figure 9). The two blocks are shown as separate pathways. First, engineers can study the influence of the shared environmental parameters on the operation of the power plant. Merging the blocks' individual consumptions via addition, they can observe that high outside temperatures and low air pressure both lead to higher

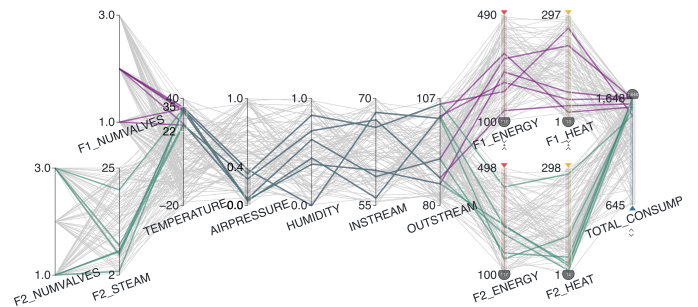


Fig. 9: Power plant operation: a certain amount of heat is to be produced by a combination of blocks (top and bottom) that share the same environment (center). High temperature and low air pressure lead to a high cumulated consumption of combustion material (right).

overall consumption of combustion material and therefore high costs. At the component level, only low number of valves is needed for the air-cooled block to reach a high outcome when outside temperatures are high. An analysis of the separated variable spaces of both blocks shows how the number of valves of one block influences the operation of the other block. Selecting operation modes with two active valves for the water-cooled block (since this ensures a low consumption of combustion material), a similar combustion-saving setting for the air-cooled block relies on high air pressure and high air humidity – thus, it highly depends on non-controllable environmental conditions.

The Composite Parallel Coordinates enable operators to see all involved parameters at a glance (**T<sub>1</sub>** Overview) and to understand the dependencies between parameter settings of different blocks (**T<sub>6</sub>** Navigation). With the novel representation, engineers are able to study the effect of outside temperatures on the needs of combustion material simultaneously for different blocks (**T<sub>5</sub>** Connectivity). They are also able to see how a block-specific decision for a number of valves affects the energy production in the other block (**T<sub>8</sub>** Partial Choice). In this sense, Composite Parallel Coordinates open up new possibilities for investigating multi-block power plants.

### 5.2 Case Study: Magnetic-Geared Motor Design

Magnetic-geared motors (MGM) are suited for industrial applications that require high power densities, e.g., wind energy or ship propulsion. To achieve the desired outcome, the driving motor and gear need to interact effectively. This case study was informed by an observational

study, where two engineers collaboratively worked with the tool. We report on their qualitative feedback at the end of this section. The one-and-a-half-hour remote study was recorded. Both engineers have a background in mechatronics and years of experience in electric drive design. Their daily work involves complex simulations and optimizations of geometries, magnetics, thermal conditions, and their interplay. They are familiar with basic visualizations and brushing techniques, including standard parallel coordinates. We primarily wanted to identify aspects of our technique that are particularly relevant to the engineers' decision-making. Thus, we did not impose a pre-defined setting but emphasized free discussions on one of their real-world design optimization use cases. We only prescribed the high-level task to analyze the data and choose the most preferred motor-gear combination. After a brief introduction to the functionality of the tool, the engineers explored the Composite Parallel Coordinates on their own.

**Data Analysis** Our domain experts used COMPO\*SED to analyze 392 combinations of motor and gear variants. The data result from optimizations they conducted to investigate a side-by-side arrangement of motor and gear [34]. The design parameters in each dataset represent geometric properties and operating conditions. The gear ratio ( $G12$ ) and output specifications ( $N2$ ,  $P2$ ,  $T2$ ) are common to both datasets (Figure 1a). Related criteria that might be assembled into system criteria are component lengths ( $LFE$ ), power losses ( $PV$ ), and efficiencies ( $ETA$ ).

The experts aim at parameter settings of motor and gear that lead to a high overall efficiency with a low construction volume and torque ripple. First, the experts observe how the motor and gear variants distribute along the unique and connected parts of the system ( $T_1$  Overview). Viewing the separate pathways, they notice a motor outlier with drastically lower efficiency ( $ETA_{MOT}$ ) than all other motor variants. They also recognize two clusters of gear variants that significantly differ with respect to efficiency ( $ETA_{MG}$ ) and torque ripple ( $T1\_RIPPLE$ ).

The primary objective is the efficiency of the entire system. At the motor level, they filter out the outlier with low efficiency ( $T_3$  Filter). Next, they merge the efficiencies of motor and gear using multiplication (Figure 1b,  $ETA_{MG} * ETA_{MOT}$ ). On the collapsed axis, they restrict the overall efficiency to high values ( $T_3$  Filter, Figure 1d). From the separate pathways, the experts notice that, to their favor, motors with high current density ( $JS$ ) and high copper losses ( $PCU$ ) are excluded.

The secondary objective is the length as an approximation of the construction volume. It needs to be filtered at the component level. Otherwise, the engineers could not recognize undesired combinations where the total length is acceptable, but motor and gear lengths differ significantly. Restricting the motor and gear variants to small lengths each ( $T_3$  Filter, Figure 1c,  $LFE_{MOT}$  and  $LFE_{MG}$ ), the experts notice that gear variants with preferred low torque ripple are not in the selection anymore. Undoing the previous filter actions one by one reveals that the previous system efficiency maximization excluded them. This correlation was not known before. The engineers expect it to originate from problem-specific boundary conditions of the optimization.

The experts now face a conflict between a system-level criterion and a gear criterion ( $T_2$  Competition). They decide to not sacrifice the gear criterion too early and rather investigate the trade-off from the reverse perspective. Brushing the cluster of gear variants with low values for their unique parameter torque ripple ( $T_3$  Filter, Figure 1c,  $T1\_RIPPLE$ ) leaves the engineers with about 50 MGMs still offering acceptable system efficiencies ( $T_6$  Navigation,  $T_8$  Partial Choice).

The current selection is associated with short gears. From their experience, the engineers anticipate that this could induce less output power ( $P2$ ) of the system ( $T_2$  Competition). However, the output power should not be too low. Brushing the upper half of the respective shared axis results in a dozen motor-gear combinations ( $T_3$  Filter). Merging the components' length axes via addition ( $LFE_{MG} + LFE_{MOT}$ ) reveals that the selection still contains short MGMs ( $T_6$  Navigation). Other properties offer potential for further drill-down.

Two clusters of gear variants can be observed for the unique parameter flux density ( $OBS\_AIR$ ): one with higher flux density and one with low flux density. Brushing the latter results in six selected MGMs, which still cover a wide range of system efficiencies ( $T_8$  Partial Choice). One outlier with significantly higher total length is excluded.

The remaining five magnetic-gear motors are on par with respect to their performance. Manufacturing benefits are thus pivotal. If gear magnets are nearly squared, their mounting direction might get mixed up, leading to wrong magnetization. If their distance is too low, they are difficult to mount. After a detailed comparison ( $T_4$  Subjective Evaluation), the MGM design offering the largest gear magnet rectangularity ( $D\_BM\_HM$ ) and distance ( $D\_PM3$ ) is chosen (Figure 1e).

COMPO\*SED allowed for constant switching between the component level and system level and between overview and detail. Unlike before, the experts did not have to go back and forth between individual component optimizations. Instead, the component variants and their dependencies could be explored simultaneously. At both levels, design parameters and criteria could be equivalently used for real-time filtering. This helped the engineers directly take into account the effects of each component decision on the system operation. Rather than choosing the first working solution, the engineers could learn which combinations and what level of performance were achievable under which conditions.

**Expert Conversation** Although field observations and think-aloud walkthroughs are common evaluation methods, performing them with pairs of domain experts (E1 and E2) is rather rare [30]. Our motives were slightly different from those of studies in computer-supported cooperative work [38] and collaborative visualization [19], which primarily aim to assess teamwork. First, the conversation resembled the day-to-day practice of our experts, who analyze and discuss complex optimization problems collaboratively. Second, we hypothesized that a natural conversation between like-minded colleagues yields more valuable insights than an artificial monologue of a single expert.

We found that the overview of all involved design parameters and criteria – in particular their different roles – is the primary advantage of Composite Parallel Coordinates: "A lot of information is presented in a clear and compact way" (E1). They also adequately support filtering both at the component level and the system level: "If you drag the slider slowly, you can easily trace which variants drop out and at which point they join back in" (E2). In fact, the ability to view and constrain individual components while also observing system-level properties was perceived as a significant advantage: "If you would restrict the system length, e.g., to 70 mm, you might end up with a 60 mm motor and a 10 mm gear, which would simply be useless" (E1). Although their routine involves making complex decisions collaboratively, multiple users interacting in realtime with the same visualization is not desired, as they can no longer understand what has led to the final outcome: "The collaborative decision-making is about considering the next steps together, not about speeding up the interaction" (E1). COMPO\*SED did not directly reduce the time required for a choice, but the experts reported that it helps avoid optimization iterations. With existing tools, their choice is based on a subset of the most important parameters. If inconsistencies arise during validation, they enter an additional iteration. Such iterations and the additional time are avoided by the more comprehensive picture our technique offers: "With COMPO\*SED, we can keep an eye on all parameters right from the beginning" (E1).

The collaborative analysis session was highly similar to pair-programming: one expert, the driver, interacted with the visualization, while the other, the navigator, kept an eye on effects and hinted at further aspects to address. The conversation was free-flowing, interrupted only by considerations of what to look at next. The engineers communicated by agreeing upon next steps, refining each other's explanations, and at times even correcting each other. They also drew the other's attention toward interesting regions in the visualization. Watching the engineers learn not only from the data but also from each other provided us with insights that we might not have gained otherwise.

### 5.3 Usage Scenario II: Camera-Lens System Purchase

The previous scenarios dealt with systems where one component variant was exclusively compatible with exactly one variant of another component. However, some system design problems might require the component variants to be combined more freely. An example are cameras and lenses, where one camera body can be equipped with different lenses and one lens can be mounted on multiple camera bodies. Aiming at optimized equipment, a photographer might use Composite

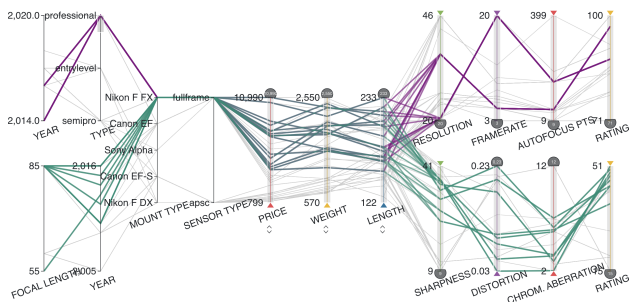


Fig. 10: A photographer can choose from combinations of three cameras (purple) and 13 lenses (green). Only components with the same mount type are compatible (third from left). Component prices, weights, and lengths are added up to system criteria (center).

Parallel Coordinates to decide whether to buy a lens to mount on her semi-professional camera body or upgrade to a professional camera body requiring new lenses due to a different mount type.

The former case requires the analysis of a one-to-many relationship. Brushing her existing camera body, the photographer compares the five compatible lenses ( $T_5$  Connectivity). They expand across a wide range of prices, weights, and lengths ( $T_8$  Partial Choice) while exhibiting similar ratings. The photographer excludes two lenses that are located towards the upper ends of the price, weight, and length ranges while not performing exclusively better in the remaining criteria. The remaining three lenses have similar prices. The final choice for one of them cannot be made at the component level. Instead, the photographer needs to consider how their characteristics like sharpness, distortion, etc. work together with the existing camera's resolution, framerate, and so on.

To further improve the performance, it might be beneficial to replace the camera body with a professional one ( $T_9$  Variant Replacement). This requires the analysis of a many-to-many relationship. Brushing the professional camera bodies, the photographer is left with combinations of three camera bodies and 13 compatible lenses (Figure 10). To not miss a preferable combination, the photographer first looks for variants that yield a good performance while being compatible with many other variants ( $T_7$  Key Component Variants). Applying her total budget as a constraint, the camera body with the highest rating and a high resolution remains. It is still compatible with six of the 14 lenses, leaving enough room to further exploit the optimization potential at the lens level.

## 6 DISCUSSION AND LIMITATIONS

Parallel coordinates are certainly one of the more complex visualization techniques. From the results of an earlier design study [8], we were confident that conventional parallel coordinates are accessible for analysts performing single-component multi-criteria optimization. Nevertheless, the question remains whether the added complexity of the novel parallel coordinates variant matches its increased usage value.

Composite Parallel Coordinates are not merely two linked visualizations. A distinctive contribution of our technique is the possibility to jointly explore the variants of multiple components, their combinations, and constraints. The relations between system and component properties can be perceived from the side-by-side layout and the interaction with composite axes. In particular, our approach shows one-to-many and many-to-many combinations of matching variants explicitly by extending the idea of linked axes across components. Our strategy follows the recommendation to integrate views with an explicit linking when relations between items of different datasets are of particular importance [21]. The cost of added visual complexity is mitigated by filters, which our experts perceived as a powerful tool. In contrast, the scenarios did not require the parallelogram strategy to avoid line crossings. The numbers of polylines seem to have been manageable, which leaves us with the open question at which point the strategy develops its full potential. A number of techniques have been proposed to address different issues associated with dense line charts, including density estimation [16], edge bundling [53], and importance-driven blending

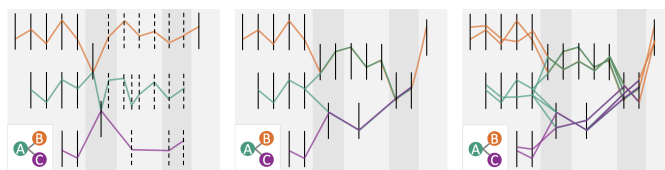


Fig. 11: A sample case with three components where A interacts with B and C. Left: collapsible axes are dotted. Center: all composite axes are collapsed. Right: three complete system variants shown.

order [47]. These techniques can be transferred to Composite Parallel Coordinates, potentially involving a particular treatment of component connections. To what extent the explicit but visually complex depiction of many-to-many relations simplifies the analysis is yet to be examined.

We demonstrate the working principle for two components. A major limitation of the composite layout is that it does not effortlessly scale to an arbitrary number of components. While the vertical layout is open to stacking multiple components, any component can only be directly connected to two neighbors above and below (see Figure 11). The vertical positioning according to component links is remarkably similar to the axis-ordering problem of conventional parallel coordinates. We hypothesize that existing solutions (e.g., linearization of node-link representations, interactive reordering, or aggregation) can be adapted to overcome this limitation on the vertical axis. Stacking more than two components leads to more complex branching of the polylines, requiring dedicated rendering and interaction techniques to trace individual polylines. Showing relations between non-adjacent components will likely introduce visual clutter. Additional ranges on the horizontal axes are needed to depict related attributes of non-adjacent components (see the second gray area in Figure 11, where parameters of all three components are totalized). With a certain number of components making up the system structure, a dedicated navigation strategy (e.g., using a minimap) might generally be required.

Another limitation is that composite axes represent only one-to-one mappings of related properties from different components. Where other constellations are required, e.g., one-to-many mappings of related properties, a more flexible representation of composite axes is needed.

The validation scenarios showcase systems with different relations between components that can be explored using COMPO\*SED. They indicate that our technique adapts to different types of data and tasks, providing an effective means for co-dependent multi-attribute choices.

## 7 CONCLUSIONS

We present Composite Parallel Coordinates for the visualization of distinct but related datasets to help decision-makers choose the most preferred design variant of a multi-component system. This requires working through a series of co-dependent multi-attribute choices. Variants of single components are depicted in juxtaposed parallel coordinates plots. Where design parameters or criteria are shared or related between components, the associated axes can be merged. Dedicated interaction patterns enable analysts to explore multi-component alternatives while considering unique component properties and emergent system properties. Qualitative feedback from three real-world scenarios and an observed expert conversation demonstrates the effectiveness of our technique for multi-component optimization. Our future research will focus on scaling the technique to more than two components, introducing more flexible specifications of system properties, and investigating the technique's effectiveness for multi-topology optimization.

## ACKNOWLEDGMENTS

We thank Hubert Mitterhofer, Edmund Marth, Gerald Jungmayr, and Siegfried Silber for their enthusiasm and insights into their daily work. We also thank Krešimir Matković for his early feedback on the idea. This work has been supported in part by the LCM – K2 Center within the framework of the Austrian COMET-K2 program. VRVis is funded by BMK, BMDW, Styria, SFG, Tyrol, and Vienna Business Agency in the scope of COMET (879730), managed by FFG.

## REFERENCES

- [1] K. Andrews, M. Osmić, and G. Schagerl. Aggregated parallel coordinates: Integrating hierarchical dimensions into parallel coordinates visualisations. In *15th Int. Conf. Knowl. Technol. and Data-Driven Bus.*, pp. 1–4, 2015.
- [2] G. Andrienko and N. Andrienko. Constructing parallel coordinates plot for problem solving. In *1st Int. Symp. on Smart Graphics*, pp. 9–14, 2001.
- [3] R. C. Basole, A. Qamar, H. Park, C. J. Paredis, and L. F. McGinnis. Visual analytics for early-phase complex engineered system design support. *IEEE Computer Graphics and Applications*, 35(2):41–51, 2015.
- [4] D. Bhattarai, J. Zhang, and B. B. Karki. Parallel coordinates-based visual analytics for materials property. In *10th International Conference on Information Visualization Theory and Applications*, pp. 83–95, 2019.
- [5] S. Carlson-Skalak, M. D. White, and Y. Teng. Using an evolutionary algorithm for catalog design. *Res. in Eng. Design*, 10(2):63–83, 1998.
- [6] C. Chang, T. Dwyer, and K. Marriott. An evaluation of perceptually complementary views for multivariate data. In *2018 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 195–204. IEEE, 2018.
- [7] S. Chen, D. Amid, O. M. Shir, L. Limonad, D. Boaz, A. Anaby-Tavor, and T. Schreck. Self-organizing maps for multi-objective Pareto frontiers. In *IEEE Pacific Visualization Symposium*, pp. 153–160, 2013.
- [8] L. Cibulski, H. Mitterhofer, T. May, and J. Kohlhammer. Paved: Pareto front visualization for engineering design. *Computer Graphics Forum*, 39(3):405–416, 2020.
- [9] J. Claessen and J. van Wijk. Flexible linked axes for multivariate data visualization. *IEEE Trans. Vis. Comput. Graph.*, 17(12):2310–2316, 2011.
- [10] E. F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, 1970.
- [11] T. N. Dang, L. Wilkinson, and A. Anand. Stacking graphic elements to avoid over-plotting. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1044–1052, 2010.
- [12] E. Dimara, A. Bezerianos, and P. Dragicevic. Conceptual and methodological issues in evaluating multidimensional visualizations for decision support. *IEEE Trans. on Vis. and Comput. Graphics*, 24(1):749–759, 2017.
- [13] L. A. Garrison, J. Müller, S. Schreiber, S. Oeltze-Jafra, H. Hauser, and S. Bruckner. Dimlift: Interactive hierarchical data exploration through dimensional bundling. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):2908–2922, 2021.
- [14] S. Gratzl, N. Gehlenborg, A. Lex, H. Pfister, and M. Streit. Domino: Extracting, comparing, and manipulating subsets across multiple tabular datasets. *IEEE Trans. Vis. Comput. Graphics*, 20(12):2023–2032, 2014.
- [15] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. Lineup: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2277–2286, 2013.
- [16] J. Heinrich and D. Weiskopf. Continuous parallel coordinates. *IEEE Trans. on Vis. Comput. Graph.*, 15(6):1531–1538, 2009.
- [17] J. Heinrich and D. Weiskopf. State of the art of parallel coordinates. In *Eurographics (State of the Art Reports)*, pp. 95–116, 2013.
- [18] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, 1985.
- [19] P. Isenberg, A. Tang, and S. Carpendale. An exploratory study of visual information analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1217–1226, 2008.
- [20] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller. A systematic review on the practice of evaluating visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2818–2827, 2013.
- [21] W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In *IEEE Pacific Visualization Symposium*, pp. 1–8, 2012.
- [22] J. Johansson and C. Forsell. Evaluation of parallel coordinates: Overview, categorization and guidelines for future research. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):579–588, 2015.
- [23] J. Kehrer and H. Hauser. Visualization and visual analysis of multifaceted scientific data: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):495–513, 2012.
- [24] J. Kehrer, P. Muigg, H. Doleisch, and H. Hauser. Interactive visual analysis of heterogeneous scientific data across an interface. *IEEE Transactions on Visualization and Computer Graphics*, 17(7):934–946, 2010.
- [25] T. Kipouros, A. Inselberg, G. Parks, and A. M. Savill. Parallel coordinates in computational engineering design. In *54th Structures, Structural Dynamics, and Materials Conference*, p. 1750, 2013.
- [26] Z. Konyha, K. Matkovic, and H. Hauser. Interactive visual analysis in engineering: A survey. *Posters at SCCC*, pp. 31–38, 2009.
- [27] P. Korhonen and J. Wallenius. Visualization in the multiple objective decision-making framework. In *Multiobjective Optimization*, pp. 195–212. Springer, 2008.
- [28] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):558–568, 2006.
- [29] A. Kossiakoff, W. N. Sweet, S. J. Seymour, and S. M. Biemer. *Systems Engineering: Principles and Practices*. John Wiley & Sons, 2nd ed., 2011.
- [30] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. Empirical studies in information visualization: Seven scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536, 2011.
- [31] M. Lind, J. Johansson, and M. Cooper. Many-to-many relational parallel coordinates displays. In *13th International Conference Information Visualisation*, pp. 25–31. IEEE, 2009.
- [32] Z. Liu, S. B. Navathe, and J. T. Stasko. Network-based visual analysis of tabular data. In *IEEE VAST*, pp. 41–50, 2011.
- [33] A. V. Lotov and K. Miettinen. Visualizing the pareto frontier. In *Multiobjective Optimization*, pp. 213–243. Springer, 2008.
- [34] E. Marth, G. Jungmayr, W. Amrhein, and F. Jeske. Magnetic-gear motor in side-by-side arrangement-optimization. In *IEEE International Electric Machines & Drives Conference*, pp. 839–846, 2019.
- [35] K. Miettinen. Survey of methods to visualize alternatives in multiple criteria decision making problems. *OR Spectrum*, 36(1):3–37, 2014.
- [36] S. Mittal and F. Frayman. Towards a generic model of configuration tasks. In *IJCAI*, vol. 89, pp. 1395–1401. Citeseer, 1989.
- [37] T. Mühlbacher, L. Linhardt, T. Möller, and H. Piringer. Treepod: Sensitivity-aware selection of Pareto-optimal decision trees. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):174–183, 2017.
- [38] D. C. Neale, J. M. Carroll, and M. B. Rosson. Evaluating computer-supported cooperative work: models and frameworks. In *Proceedings of the ACM conference on Computer supp. coop. work*, pp. 112–121, 2004.
- [39] C. North and B. Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *Working Conference on Advanced Visual Interfaces*, pp. 128–135, 2000.
- [40] P. Riehmann, A. Schädler, J. Harder, J. Herpel, and B. Froehlich. Configuration finder: A tidy visual interface for effective faceted search. 2020.
- [41] J. C. Roberts. On encouraging multiple views for visualization. In *IEEE Conference on Information Visualization*, pp. 8–14, 1998.
- [42] H. Sillitto, R. Griego, E. Arnold, D. Dori, J. Martin, D. McKinney, P. Godfrey, D. Krob, and S. Jackson. What do we mean by “system”?—system beliefs and worldviews in the incose community. In *INCOSE International Symposium*, vol. 28, pp. 1190–1206. Wiley Online Library, 2018.
- [43] B. Spence, L. Tweedie, H. Dawkes, and H. Su. Visualisation for functional design. In *Proceedings on Information Visualization*, pp. 4–10, 1995.
- [44] R. Splechna, M. Beham, D. Gračanin, M. L. Ganuza, K. Bühler, I. S. Pandžić, and K. Matković. Cross-table linking and brushing: Interactive visual analysis of multiple tabular data sets. *The Visual Computer*, 34(6):1087–1098, 2018.
- [45] R. Splechna, K. Matković, D. Gračanin, M. Jelović, and H. Hauser. Interactive visual steering of hierarchical simulation ensembles. In *IEEE VAST*, pp. 89–96, 2015.
- [46] S. K. Tadeja, T. Kipouros, Y. Lu, and P. O. Kristensson. Supporting decision making in engineering design using parallel coordinates in virtual reality. *AIAA Journal*, 59(12):5332–5346, 2021.
- [47] T. Trautner and S. Bruckner. Line weaver: Importance-driven order enhanced rendering of dense line charts. In *Computer Graphics Forum*, vol. 40, pp. 399–410, 2021.
- [48] J. Wang, S. Hazarika, C. Li, and H.-W. Shen. Visualization and visual analysis of ensemble data: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 25(9):2853–2872, 2018.
- [49] J. Wang, X. Liu, H.-W. Shen, and G. Lin. Multi-resolution climate ensemble parameter analysis with nested parallel coordinates plots. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):81–90, 2016.
- [50] D. K. I. Weidele. Conditional parallel coordinates. In *IEEE Visualization Conference*, pp. 221–225, 2019.
- [51] K. Zeman, F. Jungreitmayr, A. Khandoker, R. Scheidl, H. Wahl, T. Buchegger, R. Haas, J. Hoffelner, S. Boschert, R. Rosen, C. Aschpurwis, and B. Wanner. Symbiotic mechatronics: An alternative view on complex systems. In *Proc. of Tools and Methods of Competitive Engineering*, 2020.
- [52] X. Zhao, Y. Wu, W. Cui, X. Du, Y. Chen, Y. Wang, D. Lee, and H. Qu. Skylens: Visual analysis of skyline on multi-dimensional data. *IEEE Trans. on Visualization and Computer Graphics*, 24(1):246–255, 2017.
- [53] H. Zhou, X. Yuan, H. Qu, W. Cui, and B. Chen. Visual clustering in parallel coordinates. In *Computer Graphics Forum*, vol. 27, pp. 1047–1054, 2008.