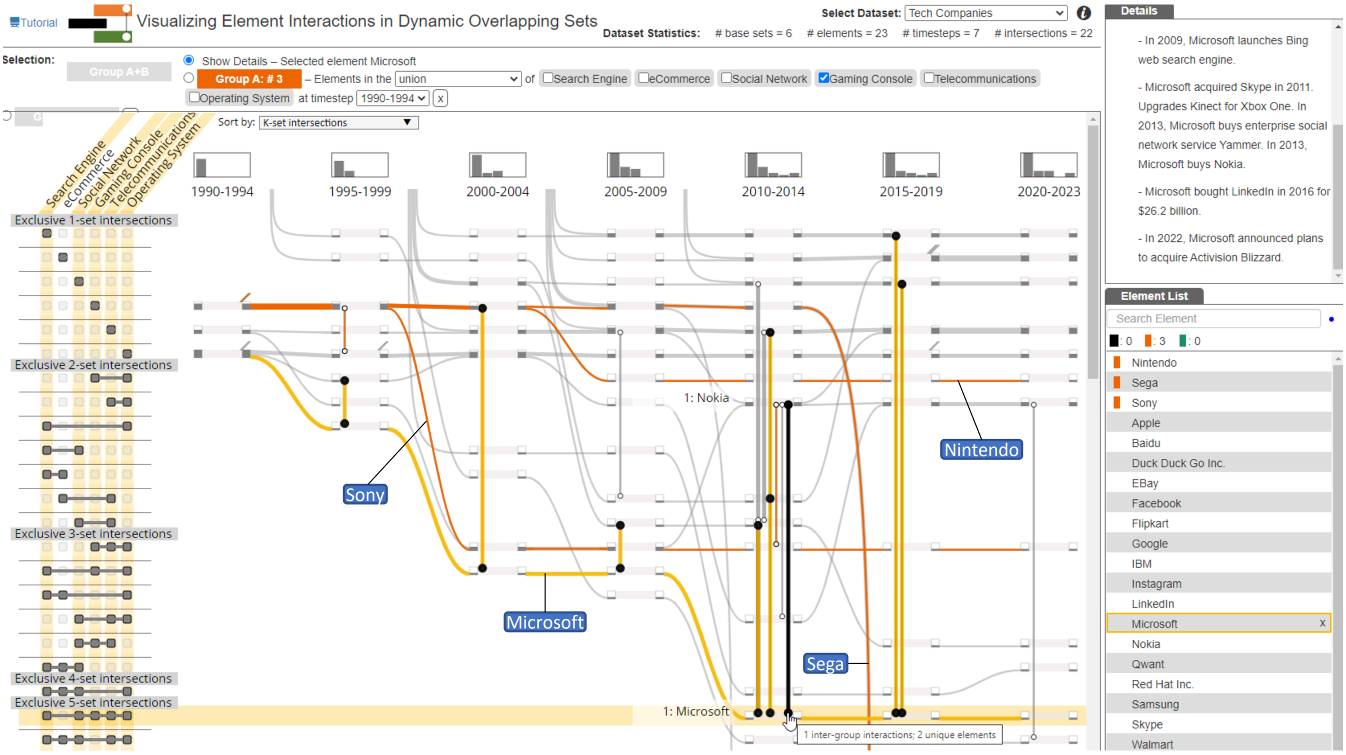
Visualizing Element Interactions in Dynamic Overlapping Sets

Shivam Agarwal[†]

University of Bamberg, Germany



Select Dataset: Tech Companies

Dataset Statistics: # base sets = 6 # elements = 23 # timesteps = 7 # intersections = 22

Selection: ● Show Details – Selected element Microsoft

○ Group A: # 3 – Elements in the union of □Search Engine □eCommerce □Social Network ☑Gaming Console □Telecommunications □Operating System at timestep 1990-1994

Sort by: K-set intersections

Details

- In 2009, Microsoft launches Bing web search engine.

- Microsoft acquired Skype in 2011. Upgrades Kinect for Xbox One. In 2013, Microsoft buys enterprise social network service Yammer. In 2013, Microsoft buys Nokia.

- Microsoft bought LinkedIn in 2016 for $26.2 billion.

- In 2022, Microsoft announced plans to acquire Activision Blizzard.

Figure 1: *A screenshot visualizing interactions between elements in dynamic sets. Yellow lines show* Microsoft*'s expanding business portfolio and interactions with other companies (acquisitions and partnerships). Four streams have been annotated with element names.*

Abstract

Elements—the members in sets—may change their memberships over time. Moreover, elements also directly interact with each other, indicating an explicit connection between them. Visualizing both together becomes challenging. Using an existing dynamic set visualization as a basis, we propose an approach to encode the interactions of elements together with changing memberships in sets. We showcase the value in visually analyzing both aspects of elements together through two application examples. The first example shows the evolution of business portfolio and interactions (e.g., acquisitions and partnerships) among companies. A second example analyzes the dynamic collaborative interactions among researchers in computer science.

1. Introduction

A data item belonging to multiple categories simultaneously may be modeled as an element in different overlapping sets. An element's set membership may also change, resulting in dynamic overlapping sets. Although a few visualizations have been proposed [FFKS21], no technique exists that shows interactions between elements together with their dynamic set memberships. Analyzing elements sharing the same attributes and hence having membership in specific sets, helps to understand the overlap among sets, which reveals useful insights about the dataset. Moreover, the interactions are an explicit connection between elements which could also affect the set membership of elements. Hence, a joint

[†] email: shivam.agarwal@uni-bamberg.de

DOI: 10.2312/evs.20231050

analysis of both aspects, although challenging, could help to understand temporal behavior better. For example, a company (element) acquiring another company to diversify its portfolio of offered products or services (sets), or collaborations between interacting members (elements) of research communities (sets).

We take inspiration from Set Streams [AB20], a dynamic set visualization technique showing timesteps as columns and exclusive intersections as rows. We propose an enriched design by encoding interactions of elements as vertical lines between rows and skewed rectangles on top of a cell. As shown in Figure 1, the evolving membership of a highlighted element *Microsoft* is shown in streams going from left to right, while the vertical lines in a timestep represent its interactions with elements in a different set. The supplementary material includes the prototype and a video. Please note that we aim to visualize the interactions between elements in the set-typed data, which is different from the general usage of the word to represent exchange between humans and computers.

2. Related Work

Among the existing dynamic set visualizations, Bubble Sets [CPC09] places the elements in a timeline, while colored overlays represent their membership in a set. The layout restricts the scalability of set overlaps as areas between overlays. Hence, in Bubble Sets, an element belongs only to one set at a time. Extending the idea further, TimeSets [NXWW16], shows color-coded sets as an overlay on the horizontal timeline and vertically positions the elements in set intersections at the boundaries of the respective sets. However, it becomes difficult to represent the intersections of more than two sets. Vehlow et al. [VBAW15] plots time on the horizontal axis, while rows represent color-coded groups in dynamic graphs. The change in membership is marked by the ribbons (or streams) connecting adjacent timesteps. However, a node in the graph may belong to only one group at a time.

Interactions between a pair of entities can be modeled as edges in a graph. For instance, MOSAIC Viewer [BRH*20] visualizes the communication among cooperative autonomous robots through an aggregated node-link diagram together with a timeline to mark important events. Animation-based dynamic set visualization is useful to convey the changes incrementally (e.g., [MWTI19]), but is not suitable to provide a static overview. Hence, we look towards other ways of visualizing element interactions with more flexibility on the underlying layout for positioning sets. Encoding a hyperedge as a vertical line to connect multiple elements is a simple and intuitive representation of interactions, as demonstrated in PAOHVis [VBP*20]. It is also generalizable for showing element interactions in other domains, e.g., between entities in a mixed-reality session [AASB20], or between game-playing AI agents for cooperation and competition [AWB20]. Therefore, we use and extend the encoding in our approach.

3. Visualization Approach

A survey organized the existing static set visualizations as Euler-based, overlays, node-link, matrix, aggregated, and scatter-based techniques [AMA*16]. Since they do not explicitly represent time, extending them to include dynamic set memberships along with element interactions is not straightforward and may not always be feasible. Hence we base the design of our approach on Set Streams [AB20], a dynamic set visualization with a matrix layout where rows represent partitions of overlapping set regions as exclusive set intersections and columns show the timesteps. Considering an example, if set $\mathbf{A} = \{x,y\}$ and $\mathbf{B} = \{y,z\}$, then there are three exclusive intersections with elements: [only in $\mathbf{A}$] $= \{x\}$, [only in $\mathbf{B}$] $= \{z\}$, and [only in $\mathbf{A} \cap \mathbf{B}$] $= \{y\}$. The design of partitioning was initially proposed in UpSet [LGS*14], a static set visualization, which avoids showing elements in overlapping regions multiple times, with each exclusive intersection in a row. Since the design was efficiently extended in Set Streams to show dynamic set memberships, we selected this representation. The highlighted row in Figure 1 shows the exclusive intersection of sets *Search Engine*, *Social Network*, *Gaming Console*, *Telecommunications* and *Operating System*. The streams connecting adjacent columns encode the change in set memberships of elements, while its width shows the number of elements. The streams coming from the top at a particular timestep to a row represent the introduction of new elements in sets, while downward streams indicate the elements do not belong to any set in further timesteps [AB20].

To show the element interactions, we modify the design of cells representing an exclusive intersection at a specific timestep in the matrix layout. In each cell, we put bars at the two ends to encode the number of contained elements ( ). Modeling an interaction as a hyperedge, inside the empty region of a cell, we draw a vertical line connecting rows (by small circles) showing aggregated interactions between elements in the respective exclusive intersection rows. Interactions between elements within the same exclusive intersection are aggregated and shown as a skewed rectangle on the top right border of a node. The width of the skewed rectangles and vertical lines encode the number of interactions, within the same and between different exclusive intersections, respectively.

Set Streams has options to sort the rows, e.g., by exclusive ($k-$) set intersections groups (default), decreasing order of cardinality in a timestep, etc. In addition, we implemented a row sort option by the sum of the number of interactions across all timesteps. Within a timestep, the vertical lines are packed using first-fit greedy algorithm [CGJ96], to reduce the required horizontal space. Adjusting the column width for each timestep based on the number of interactions would save even more space. However, we chose to have a fixed column width for all timesteps to avoid confusion.

Hovering over an aggregated interaction hyperedge temporarily shows the labels of participating elements in the first five interactions, in the respective rows. For instance, the hovered vertical line in Figure 1 shows an interaction between *Microsoft* and *Nokia*. To show details, e.g., the involved elements in an intersection, we add a *Show Details* option as a radio button (Figure 2 top). Once checked, on the left-click of a hyperedge, the details are shown in a panel on the top right (Figure 1). On selecting an element from the *Element List*, the yellow-colored streams and hyperedges are emphasized (by setting their width as 5 pixels) to highlight its memberships and interactions. Hovering over a row emphasizes the respective interaction hyperedges, as shown in Figure 2 for exclusive intersection [*NLP*, *AI/ML*]. Encoding the selection of two groups of elements works as proposed in Set Streams: orange color shows elements in group A, green for group B, and black for common elements. The element list is sorted alphabetically by default and on selecting a group, the corresponding elements are reordered: first elements in both groups A and B, then those in group A, then B, followed by the remaining elements. The visual query selection mechanism of the groups is extended to include the interaction hyperedges.

4. Application Examples

Next, we discuss insights from two application examples.

4.1. Evolving Business and Interactions among Companies

Interactions (acquisitions and partnerships) are common business strategies that affects the company's portfolio. For this example, we manually collected a dataset of 23 companies (elements), that offer products or services in six categories, namely, *Search Engine*, *eCommerce*, *Social Network*, *Gaming Console*, *Telecommunications*, and *Operating System*. We collect the information from 1990–2023 and divide the duration into seven timesteps, each representing a period of five years. It should be noted that the dataset has been checked for its correctness, but it is not a complete record of offered products or services through acquired companies, or all interactions between the included companies.

Horizontal downward streams (in default sorting), connecting rows from different $k-$set intersections, indicate the expanding portfolio of companies. As shown in Figure 1, the expanding business of a selected company *Microsoft* is visible through yellow colored edges going down with a summary in the *Details* view. Until the last timestep, *Microsoft* offered products and categories across all the six categories, except *eCommerce*. Highlighted hyperedges show the interactions of *Microsoft* with other companies. For instance, the hovered line in the timestep 2010–2014 shows an interaction between *Microsoft* and *Nokia*. The details of the interaction reveal that *Nokia*'s mobile and devices division was acquired by *Microsoft* in 2014, through which it ventured into the *Telecommunications* business. Also, in the same timestep (2013), *Microsoft* acquired *Yammer*, an enterprise social network service, and started its business in *Social Network* market (as seen from the details in Figure 1 top right). Investigating interactions between other companies in a similar way reveals similar insights, such as, *EBay* an *eCommerce* company expanded its business by acquiring stakes in *Skype* (who made *Telecommunications* software) in 2005, but later sold the shares to Microsoft in 2011, and narrowed its focus back to the original business of *eCommerce*.

Focusing on companies who initially made *Gaming Console*, we specify a query in selection A to show the elements in the union of the set at timestep 1990–1994 (Figure 1 top). The resulting three companies are shown in orange-colored streams. *Sony*, who made *Gaming Console* only, expanded its business in the timestep 2000–2004 by making *Operating System* (*Orbis OS*, for *PlayStation 4*) and *Telecommunications* devices. Similarly, *Nintendo* started making *Operating System* for its gaming console in the timestep 2005–2009, called *Nintendo DSi system software*, followed by *Nintendo 3DS system software* in 2011 and *Nintendo Switch system software* in 2017. An orange-colored vertical line in 1995–1999 shows a partnership with *IBM* to make processors for *Nintendo*'s consoles. On the other hand, *Sega* used to make only *Gaming Console*, but stopped doing so after 2014, and did not offer any products or services in the included categories.

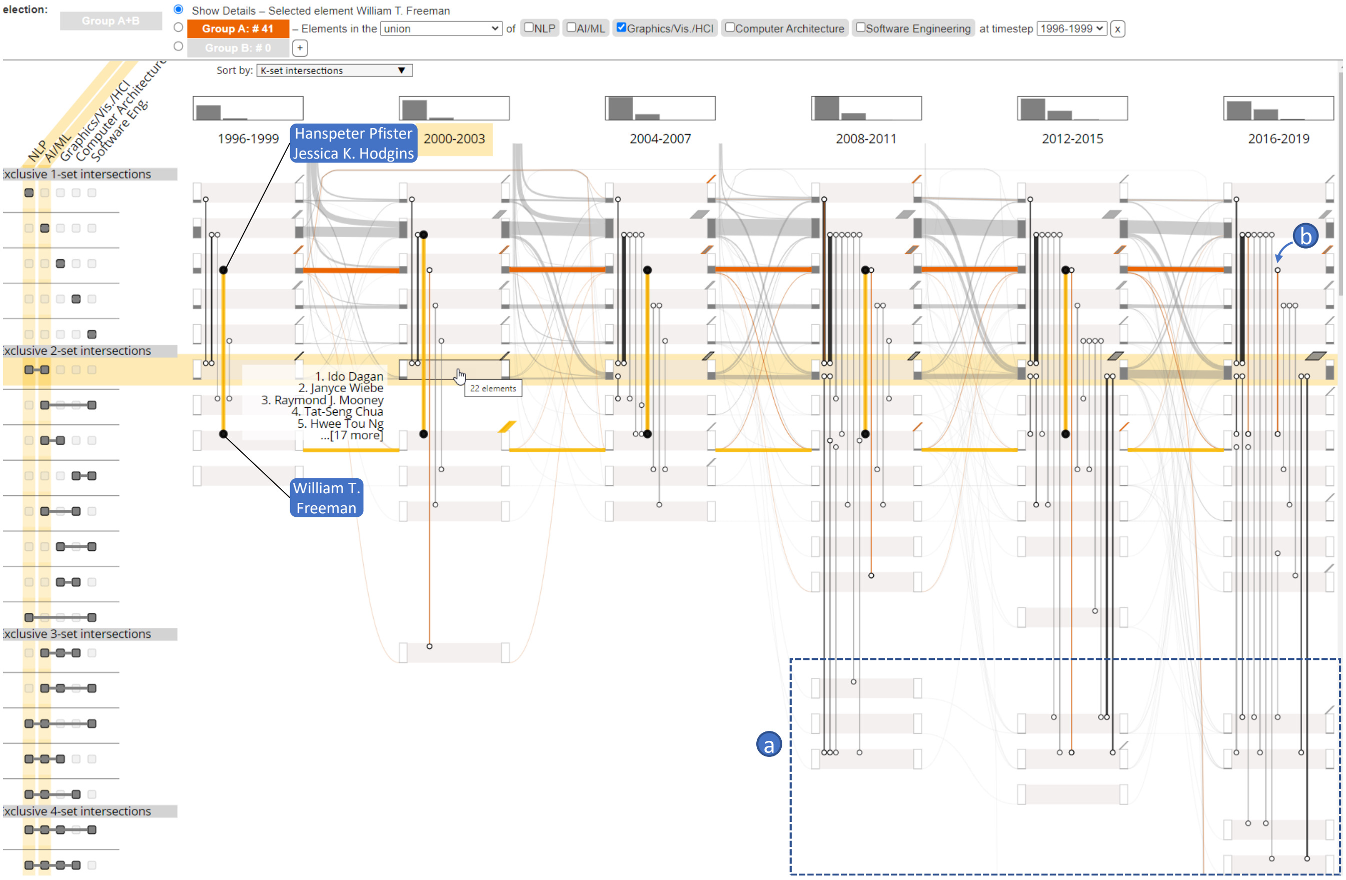
4.2. Dynamic Collaborations among Researchers

We collect the dataset of scientific publications in five areas of computer science and model them as sets: *NLP*, *AI/ML*, *Graphics/Vis./HCI*, *Computer Architecture*, and *Software Engineering*. There are 380 experienced researchers in the filtered dataset with at least 30 publications between 1996 and 2019. The duration is divided into six timesteps, each showing a range of four years. We model the co-authorship in a publication as the interaction hyperedge between involved researchers in a timestep.

In exclusive 1-set intersections across all timesteps (the first five rows in Figure 2), we observe a steady presence of the skewed rectangles on the top right of a node. It indicates that authors, publishing exclusively in one field, have a stable record of co-authorship interactions within the community. Additionally, we see a drastic rise in the number of interactions between authors publishing exclusively in the fields of both *NLP* and *AI/ML* (increasing width of skewed rectangles in the sixth row of Figure 2).

Being interested in *AI/ML* and *Graphics/Vis./HCI*, we wanted to explore who published early in both fields. Hovering over the node of the first timestep in the exclusive intersection (the highlighted row in Figure 2), we find there is only one such researcher, *William T. Freeman*. We selected the author from the element list, which highlighted the author's journey with yellow colored edges, as shown in the Figure 2. The horizontal yellow lines indicate that he consistently published in both research fields. The highlighted hyperedges mark his collaborations. Hovering over one such hyperedge in the first timestep reveals the names of collaborating authors (blue annotations in Figure 2). The interaction was between *Hanspeter Pfister* and *Jessica K. Hodgins* publishing exclusively in [*Graphics/Vis.HCI*] and *William T. Freeman* in exclusive intersection of [*AI/ML*, *Graphics/Vis./HCI*]. It indicates that the researchers co-authored a paper that was published in the field of [*Graphics/Vis.HCI*]. Since *William T. Freeman* is in the exclusive intersection of the two fields, it also means that apart from this collaboration, in the same timestep: (a) he published at least one paper in *AI/ML* venue not co-authored with either of the two researchers and (b) the authors did not publish in any other research fields. Such interactions may indicate diverse expertise of individual researchers, or that in interdisciplinary projects, the required skill set in a different field of research is fulfilled by inviting the experts from other fields (e.g., *William T. Freeman* being an expert in *AI/ML* consistently contributed to the projects published in *Graphics/Vis.HCI* venue).

We observe that initially there were no exclusive high-order intersections, but later, some researchers started publishing in multiple fields and interacted with others (Figure 2a). The width and the number of vertical lines in each timestep indicate that the number of interactions between researchers has been steadily increasing. Hovering over exclusive intersection [*NLP*, *AI/ML*] (Figure 2), we see the increasing width of emphasized vertical lines with exclu-



election: Group A+B

● Show Details – Selected element William T. Freeman

○ Group A: # 41 – Elements in the union of □NLP □AI/ML ☑Graphics/Vis./HCI □Computer Architecture □Software Engineering at timestep 1996-1999 x

○ Group B: # 0 +

Sort by: K-set intersections

Figure 2: *The main view shows dynamic collaborations among researchers (elements) as interactions publishing in different fields of study (sets). The blue annotations show the names of involved researchers in an interaction.*

sive intersection [*AI/ML*], suggesting a steady growth of collaborations between researchers publishing in the two fields. On the other hand, there are only a few vertical lines connecting rows involving *Computer Architecture*, suggesting minimal inter-disciplinary interactions with experienced researchers in this field.

To explore the interactions of early *Graphics/Vis./HCI* researchers, we specified a query to select all the elements that belonged to the set in the first timestep (Figure 2 top). The query returned 41 such researchers, shown in orange color. The orange-colored area in skewed rectangles on the nodes in the row and thin colored vertical lines (e.g., Figure 2b) indicate that the researchers collaborated not only with others from the same community but also with those from other communities.

5. Discussion and Future Work

The insights indicate the value in analyzing element interactions along with the changing memberships in overlapping sets. Among limitations, the approach does not show the membership weight of an element in a set, e.g., the market share of a company's product in a category or number of papers published by a researcher [ATWB20]. Similarly, there could be other interaction attributes that could be important for the analysis, e.g., location, as shown in a static set visualization technique *LineSets* [ARRC11].

The scalability of our approach is almost similar to Set Streams ($\sim$ 400 elements, max 6 sets, 6-7 timesteps, and $\sim$120 interactions). Since the nodes are split and widened, fewer timesteps could be shown. Horizontal scrolling could partly help. Moreover, due to the increase in information (interactions), as compared to Set Streams, the visual analysis becomes complex. The approach tackles this by abstractiong and providing relevant details on demand, but other solutions may be explored. For instance, providing a visual summary of the interactions through short natural language text templates and inline graphics, e.g., as proposed in VIS Author Profiles [LB19] for individual researchers. Although the vertical lines do not overlap, the dense representation affects the legibility. Also, the number of intersections grows exponentially with more sets, affecting the approach's scalability. Both could be partly addressed by aggregation (e.g., one row for all 3-set intersections) or hiding the unimportant intersections. Finally, including element interactions in the analysis of set-typed data opens up an interesting question for the research community: how to model and visualize multiple and generic relations (e.g., similarity) or attributes between data items in the set-typed data, together with their dynamic overlapping set memberships?

References

[AASB20] Agarwal, Shivam, Auda, Jonas, Schneegass, Stefan, and Beck, Fabian. "A Design and Application Space for Visualizing User Sessions of Virtual and Mixed Reality Environments". *Vision, Modeling, and Visualization*. 2020. doi: 10.2312/vmv.20201194 2.

[AB20] Agarwal, Shivam and Beck, Fabian. "Set Streams: Visual Exploration of Dynamic Overlapping Sets". *Computer Graphics Forum* 39.3 (2020), 383–391. issn: 1467-8659. doi: 10.1111/cgf.13988 2.

[AMA*16] Alsallakh, Bilal, Micallef, Luana, Aigner, Wolfgang, et al. "The State-of-the-Art of Set Visualization". *Computer Graphics Forum* 35.1 (2016), 234–260. doi: 10.1111/cgf.12722 2.

[ARRC11] Alper, Basak, Riche, Nathalie, Ramos, Gonzalo, and Czerwinski, Mary. "Design Study of LineSets, a Novel Set Visualization Technique". *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), 2259–2267. doi: 10.1109/TVCG.2011.186 4.

[ATWB20] Agarwal, Shivam, Tkachev, Gleb, Wermelinger, Michel, and Beck, Fabian. "Visualizing Sets and Changes in Membership Using Layered Set Intersection Graphs". *Vision, Modeling, and Visualization*. 2020. doi: 10.2312/vmv.20201189 4.

[AWB20] Agarwal, Shivam, Wallner, Günter, and Beck, Fabian. "Bombalytics: Visualization of Competition and Collaboration Strategies of Players in a Bomb Laying Game". *Computer Graphics Forum* 39.3 (2020), 89–100. issn: 1467-8659. doi: 10.1111/cgf.13965 2.

[BRH*20] Bae, Suyun Sandra, Rossi, Federico, Hook, Joshua Vander, et al. "A Visual Analytics Approach to Debugging Cooperative, Autonomous Multi-Robot Systems' Worldviews". *IEEE Conference on Visual Analytics Science and Technology*. 2020, 24–35. doi: 10.1109/VAST50239.2020.00008 2.

[CGJ96] Coffman, E. G., Garey, M. R., and Johnson, D. S. "Approximation Algorithms for Bin Packing: A Survey". *Approximation Algorithms for NP-Hard Problems*. USA: PWS Publishing Co., 1996, 46–93. isbn: 0534949681 2.

[CPC09] Collins, Christopher, Penn, Gerald, and Carpendale, Sheelagh. "Bubble Sets: Revealing set relations with isocontours over existing visualizations". *IEEE Transactions on Visualization and Computer Graphics* 15.6 (2009), 1009–1016. doi: 10.1109/TVCG.2009.122 2.

[FFKS21] Fischer, Maximilian T., Frings, Alexander, Keim, Daniel A., and Seebacher, Daniel. "Towards a Survey on Static and Dynamic Hypergraph Visualizations". *IEEE Visualization Conference (VIS)*. 2021, 81–85. doi: 10.1109/VIS49827.2021.9623305 1.

[LB19] Latif, Shahid and Beck, Fabian. "VIS Author Profiles: Interactive Descriptions of Publication Records Combining Text and Visualization". *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), 152–161. doi: 10.1109/TVCG.2018.2865022 4.

[LGS*14] Lex, Alexander, Gehlenborg, Nils, Strobelt, Hendrik, et al. "UpSet: Visualization of intersecting sets". *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), 1983–1992. doi: 10.1109/TVCG.2014.2346248 2.

[MWTI19] Mizuno, Kazuyo, Wu, Hsiang-Yun, Takahashi, Shigeo, and Igarashi, Takeo. "Optimizing Stepwise Animation in Dynamic Set Diagrams". *Computer Graphics Forum* 38.3 (2019), 13–24. doi: 10.1111/cgf.13668 2.

[NXWW16] Nguyen, Phong H, Xu, Kai, Walker, Rick, and Wong, BL William. "TimeSets: Timeline visualization with set relations". *Information Visualization* 15.3 (2016), 253–269. doi: 10.1177/1473871615605347 2.

[VBAW15] Vehlow, Corinna, Beck, Fabian, Auwärter, Patrick, and Weiskopf, Daniel. "Visualizing the Evolution of Communities in Dynamic Graphs". *Computer Graphics Forum* 34.1 (2015), 277–288. doi: 10.1111/cgf.12512 2.

[VBP*20] Valdivia, Paola, Buono, Paolo, Plaisant, Catherine, et al. "Analyzing Dynamic Hypergraphs with Parallel Aggregated Ordered Hypergraph Visualization". *IEEE Transactions on Visualization and Computer Graphics (To appear)* (2020). doi: 10.1109/TVCG.2019.2933196 2.

Proceedings published by Eurographics - The European Association for Computer Graphics.