Rotate or Wrap? Interactive Visualisations of Cyclical Data on Cylindrical or Toroidal Topologies

Kun-Ting Chen, Tim Dwyer, Benjamin Bach, and Kim Marriott

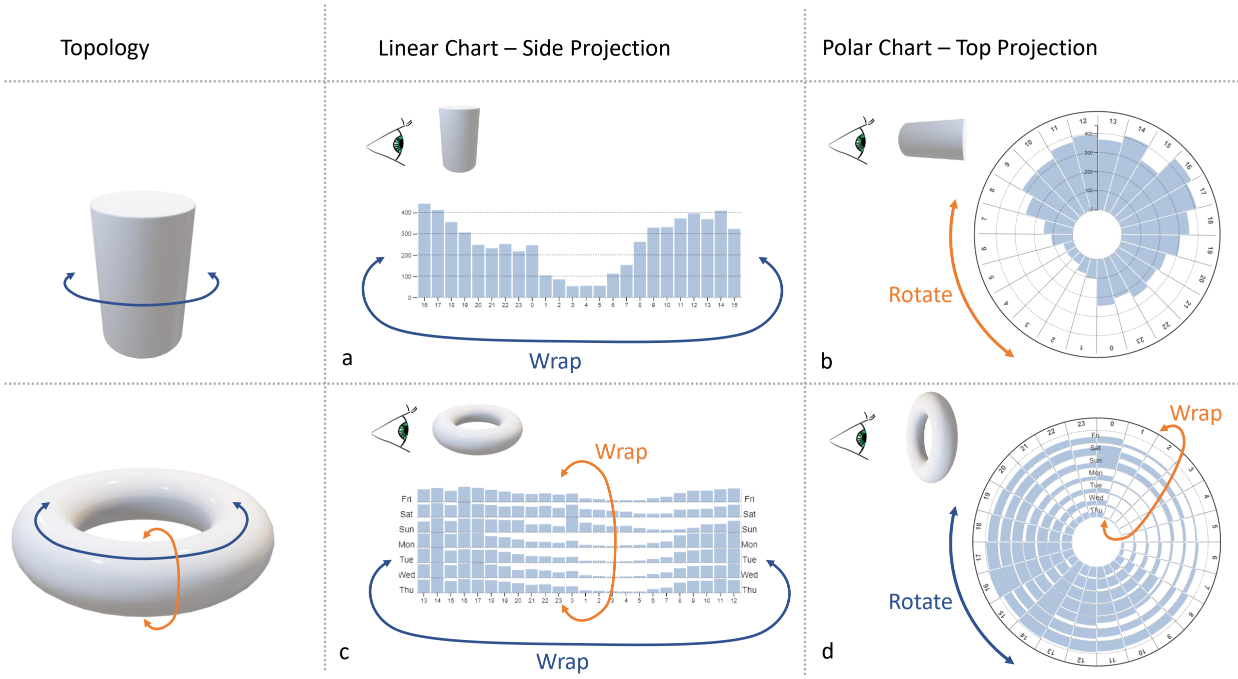


Fig. 1. Rotate or Wrap, centre-row: average traffic accidents per hour on Thursdays in Manhattan in 2016 [42]. bottom-row: average traffic accidents per hour across the week in Manhattan in 2016.

Abstract—In this paper, we report on a study of visual representations for cyclical data and the effect of interactively *wrapping* a bar chart ‘around its boundaries’. Compared to linear bar chart, polar (or radial) visualisations have the advantage that cyclical data can be presented continuously without mentally bridging the visual ‘cut’ across the left-and-right boundaries. To investigate this hypothesis and to assess the effect the cut has on analysis performance, this paper presents results from a crowdsourced, controlled experiment with 72 participants comparing new continuous panning technique to linear bar charts (*interactive wrapping*). Our results show that bar charts with interactive wrapping lead to less errors compared to standard bar charts or polar charts. Inspired by these results, we generalise the concept of interactive wrapping to other visualisations for cyclical or relational data. We describe a design space based on the concept of one-dimensional wrapping and two-dimensional wrapping, linked to two common 3D topologies; cylinder and torus that can be used to metaphorically explain one- and two-dimensional wrapping. This design space suggests that interactive wrapping is widely applicable to many different data types.

Index Terms—Cyclic temporal data, cylindrical topologies, toroidal topologies, interaction techniques, bar charts, polar charts, crowdsourced experiment

1 Introduction

While often perceived as linear, many temporal phenomena are intrinsically cyclical following cycles of day and night, seasons, or biorhythms. Common examples of such data include time series of traffic flow over a twenty-four hour period; average temperature or birth rate over 12 months; or electrical and sound wave amplitude profiles. Non-temporal data can also be cyclical, such as average wind strength from different compass directions.

Often, cyclical data is presented in traditional linear bar and line charts. However, this ignores the cyclic nature of the underlying data and so understanding of cyclical phenomena in these charts may be hindered by ‘cuts’ in the visualisation; i.e. the analyst needs to mentally ‘join’ the left and right sides of the visualisation together. This can make it hard to understand trends across the boundary of the chart as well as to compare bars that are far apart in the chart but whose data is temporally close, e.g. comparing hour 23 with hour 0 on a 24-hour chart (Fig. 1a) [38].

One common way to overcome this problem is to represent cyclical data in *polar* (a.k.a. radial) visualisations. Like an analogue clock, polar visualisations show time in a circle, allowing for continuous

• *Kun-Ting Chen, Tim Dwyer and Kim Marriott are with Monash University. E-mail: kun-ting.chen,tim.dwyer,kim.marriott@monash.edu.*
• *Benjamin Bach is with the University of Edinburgh. E-mail: bbach@inf.ed.ac.uk*

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.xxxx/TVCG.201x.xxxxxxx

representation of patterns and trends in any part of the cyclical data. Often, these time series are visualised using the length of bars, organised around the centre of the polar visualisation (Fig. 1b). However, previous studies [1, 11, 44] have shown that comparing lengths of bars in polar visualisation is less effective than comparing bar heights in traditional linear bar charts. Likely (and in accordance with position encoding differences identified by Cleveland and McGill [16]), this is due to bars in the linear charts being parallel and aligned to a common baseline, as opposed to being at different angles and aligned to a circular base.

In this paper, we explore the trade-off between the continuity provided by polar visualisations versus the ease of height comparison on linear charts, and—most noteworthy—the use of novel interactions for polar and linear chart representations of cyclical data, to work around their respective limitations. That is, for linear charts, we implement a "wrapped" panning interaction, such that panning left causes marks which disappear from the left side to reappear on the right (and *vice versa* when panning right). Thus, bars that might otherwise be at opposite ends of the chart can be brought closer together. We have not seen such "wrapped panning" applied to charts before, but it is inspired by recent promising results in wrapped presentations that untangle and facilitate the interpretation of the node-link representations [14, 15]. We also implement a similarly novel 'rotate' gesture for polar charts, which allows the user to bring any bar to the top or centre bars for more aligned comparison.

Our first contribution in this paper is to address the following open questions: [Q1] *Does adding interactive wrapping to linear bar charts improve their effectiveness?* [Q2] *How effective is interactive wrapping for linear bar charts compared with static polar bar charts?* [Q3] *Does adding interactive rotation to polar bar charts improve their effectiveness?* To investigate these questions, we report on a controlled user study with 72 participants comparing four different visualisations of cyclic data: (a) linear bar chart (without interactive wrapping); (b) linear bar chart with interactive wrapping; (c) polar bar chart (without interactive rotation); and, for completeness, (d) polar bar chart with interactive rotation. Our results show that for reading intervals or comparing values across the cut, interactive wrapping of linear bar charts significantly outperforms static linear or polar charts in terms of error.

Inspired by these results, our **second contribution is to investigate the design space** for (interactive) wrapped visualisations. We describe how wrapping can be applied to other visualisations for non-spatial cyclical data or abstract/relational data that have not previously been considered in this way, such as horizon graphs, Sankey diagrams, adjacency matrices, and multi-dimensional scaling. Our design space considers not only one-dimensional wrapping (as in bar charts) but also two-dimensional wrapping where a visualisation can be wrapped both horizontally as well as vertically. This is applicable, for example, to cyclical horizon charts. While the one-dimensional wrapping described above can be considered to exist in a cylindrical topology (before being projected to the screen), two-dimensional wrapping implies a toroidal topology (as per Fig. 1-bottom). This topological understanding of wrapping visualisations provides a unified view of wrap and rotate interactions with linear and polar charts and suggests new wrapped interaction designs (e.g. Fig. 1d).

The full study material, illustrative examples of interactive wrapping or rotation are available in the supplementary file as well as the Open Science Foundation: `https://osf.io/r8cw4/`.

2 Related Work

2.1 Visualising Temporal Data

Despite extensive consideration from designers of visualisation techniques, temporal data visualisation continues to pose challenges for effective representation. Various surveys of temporal data visualisation exist, citing general techniques [2], timelines [10], spatio-temporal data exploration techniques, for attributed trajectories [41], and more generally [4], and more abstract quantitative data that can be represented through space-time cubes [5]. Interactive techniques for exploring general temporal data (without any particular allowance for cycles) range from multilevel zooming to scale to large data or minute detail [46] to sophisticated visual organisation of sequences of state data (such as brain activity) [6].

Data for many temporal phenomena is characteristically cyclical, as nature follows the cycles of day and night, climate and seasonality, as well as biochemical processes that are continuous cycles. In the case of time series, polar visualisations have often been used either as simple cycles with bars or a line, multiple layers of bars or lines (e.g., silhouette graphs [22], overlaying multiple lines, e.g., one for every year, or a 'timeline' spiralling outside the centre of the polar chart [40]. A particular way to visualise cycles and repetition in temporal data is to abstract temporal change through multidimensional reduction methods [7].

2.2 Empirical Evidence

Several studies have investigated error performance with bar charts, finding that people are more accurate at comparing *adjacent* bars than bars that are further apart [16, 23, 38]. Comparing temporal data across 24 hours with static linear or polar bar charts, Waldner et al. [44] found that for low-level tasks (e.g., locating extrema, reading values and comparing values at fixed 12-hour separation) a linear layout was significantly faster than a polar layout. Other studies on time series visualisation also found that for positional and length judgements, people make less error using linear bar charts than when using polar charts for tasks of finding trends [1], locating extrema [33, 44], locating features at specific times [11, 44], comparing values [20], or proportion judgement [37]. However, there is no empirical evidence comparing the readability of cyclical temporal intervals split across the 'cut' on bar charts, compared with their corresponding polar bar charts showing the continuity.

To cope with the complexity of temporal data, interaction has been investigated. For example, pan and zoom along a linear timeline [35], mouse hover and showing hints on a linear chart or circular chart [1], and a study of interaction and single-scale selection techniques on a linear timeline [36] have been presented. However, we are not aware of wrapped panning (as defined in Sec. 3.1), being evaluated with cyclical time-dependent data.

In summary, while many tasks have been tested, such as trend detection [1, 45], pairwise group comparison [1], and pairwise single value comparison [44], no study has particularly focused on bars far apart but which could be brought closer through wrapping. Our contribution to this corpus of knowledge is the particular study of interaction for wrapping in linear bar charts and rotation in radial charts. This task suggests looking at intervals across both ends of a linear bar chart as well as bars that are 'far appart' in both chart types that could be brought 'closer together' in interactive linear bar charts.

2.3 Interaction and Wrapping

By wrapping, we refer to the fact that some visualisations can be perceived as continuous when connecting their left and right and/or their upper and lower boundaries (Fig. 1 top-centre). Perhaps, the best known example for wrapping is the game *Asteroids* where a player can leave the screen on, e.g., the right side, and reappear on the left. Similar to radial visualisations, a bar chart showing cyclical data, e.g., per month of the year, may well be understood as wrapping vertically placing the value for December adjacent to that of January. In our work, we investigate interactive panning as one possibility to wrap a linear visualisation. This notion is different from other notions of wrapping in visualisations, such as "*DuBois Wrapping*" [28], which wraps over-sized bars in bar charts into zig-zag lines.

For two-dimensional wrapping (wrapping vertically and horizontally), recent studies have shown benefits for exploring clusters in networks [14, 15]. These visualisation techniques can be understood as surface projections of a torus topology. The studies found that static wrapped visualisation of networks were more error-prone than static unwrapped network diagrams for edge and path following tasks but that interactive panning (wrapping) improved error rate over static and non-wrapped (traditional) node-link layouts. Inspired by these results, we aim to provide further evidence about the potential of both wrapping and interaction in exploring visualisations that can be seen as projections of cylinder (one-dimensional wrapping) and torus (two-dimensional wrapping). To that end, our paper concludes by describing a general design space for many types of visualisations for cylindrical and torodial topologies (Sec. 4).

3 User Study: Wrapped Time Series Readability

This section reports on a controlled user study with 72 participants investigating the performance of linear and interactive wrapped charts to understand time series data. The study aims to answer our questions Q1 to Q3 stated in the introduction. Apart from the interactive wrapping, our study differs from past work by focusing on tasks that require considering multiple bars: pair comparisons and aggregation across intervals (trend identification and average value estimation), as we expect these to be the tasks most affected by wrapping in bar charts and rotational centring in polar charts.

3.1 Techniques

In our study, we compare four techniques for the visualisation of cyclical data: *static bar*, *interactive bar*, *static polar* and *interactive polar*. A video included with the supplementary material demonstrates these interaction techniques.

- **StaticBar** represents a traditional static bar chart, as in Fig. 1a.
- **StaticPolar** represents a traditional polar bar chart, arranging bars in a radial fashion, as in Fig. 1b.
- **InteractiveBar** is a linear bar chart (Fig. 1a) where a user can drag horizontally using mouse or touch interaction to pan the visualisation, such that the chart wraps around from the left to the right and vice versa. This wrap interaction is intended to avoid the issue of bars at the extreme left of the chart being difficult to compare against bars at the extreme right, and making it difficult, for example, to detect trends that continue across this arbitrary cut. The user can simply bring such bars back together by re-centring.
- **InteractivePolar** is a polar bar chart like StaticPolar (Fig. 1b) while a user can drag on any part of the polar bar chart to rotate the chart. The chart spins around the centre in a clockwise or counterclockwise fashion as the user pans accordingly. This rotate interaction addresses a specific problem we came across when piloting our study, that bars in a polar chart seemed easier to compare when they could be centred around the vertical centre line. When a pair of bars is centred in this way, the two bars have the same vertical baseline, meaning their heights can be compared directly. When they are off-centre, they are at an awkward angle. Rotation allows the user to centre the two bars they are interested in.

3.2 Hypotheses

We group our hypotheses by our research questions Q1-Q3. Hypotheses have been preregistered with the Open Science Foundation: `https://osf.io/9k5bm`.

[Q1] Does adding interactive wrapping to linear bar charts improve their effectiveness?

- H1.1-error: InteractiveBar has less Error than StaticBar. This is observed by the existing study results that people perform more accurately when comparing adjacent bars than bars that are far apart [38]. With our interactive technique, it allows them to bring bars that are far apart closer.
- H1.1-time: StaticBar has less Time than InteractiveBar as the latter requires panning.
- H1.2-error: InteractiveBar has less Error than InteractivePolar.
- H1.2-time: InteractiveBar has less Time than InteractivePolar.

[Q2] How effective is interactive wrapping for linear bar charts compared with static polar bar charts?

- H2-error: InteractiveBar has less Error than StaticPolar.

[Q3] Does adding interactive rotation to polar bar charts improve their effectiveness?

- H3.1-error: InteractivePolar has less Error than StaticPolar since participants are able to rotate the chart to best solve the task.
- H3.1-time: StaticPolar has less Time than InteractivePolar since the latter potentially involves panning.

Our last hypothesis is about user preference (H4): user prefers InteractiveBar to StaticBar, StaticPolar, or InteractivePolar since InteractiveBar facilitates comparisons across the cut while allowing to best compare bar heights.

3.3 Tasks

We designed three tasks to compare our four techniques. The tasks are motivated by existing time series visualisation research (e.g., [1, 3, 11, 17, 20, 44]) and graphical perception task typology [12]. However, they did not look at the effect of interactive panning nor did they look at a group of bars that span across the cuts. We focus on the particular effect of analysing data across the cuts in linear static and interactive bar charts. To that end, we created two levels of difficulty for each task: *unwrapped* (for bar charts) or *centred* (for polar charts) and *wrapped* (for bar charts) or *uncentred* (for polar bar charts). In the *wrapped* condition, the user is required to analyse the data across the cuts, compared to the *unwrapped, centred, and uncentred* conditions. Examples for all tasks and conditions are shown in Fig. 2.

- **Trend Identification:** *"What describes the highlighted sequence of bars best, e.g., continuously increasing/continuously decreasing/neither?"* (Fig. 2-Trend Identification) The participants were asked to identify the trend of a highlighted set of adjacent bars. The timeout for this task was 5 seconds, inspired by our pilot studies. We recorded participants' response with multiple-choice questions with 4 options with *continuous decreasing, continuous increasing, neither*, or *unsure*. We created 10 trials for this task per technique, 3 of them were monotonically increasing, 3 were monotonically decreasing intervals, 4 were neither. We fixed the length of all intervals to be 6 bars.
- **Pairwise Group Comparison:** *"Which group of bars, A or B, has the higher total values?"* (Fig. 2-Pairwise Group Comparison) The timeout for this task was 10 seconds. We gave this task more time, as it required mental aggregation of a group of bar height. We recorded participants' response with multiple-choice questions with 3 options with *A*, *B*, and *unsure*. We created 10 trials for this task per technique, balancing the answers of *A* and *B*.
- **Pairwise Single Value Comparison:** *Which bar, A or B, has the higher value?* (Fig. 2-Pairwise Single Value Comparison) The timeout for this task was 5 seconds. We created 12 trials for this task per technique. We controlled for the distance between targeted bars, i.e., 6 trials per technique for *Short* (2 bars apart) and 6 trials per technique for *Long* (6 bars apart). We balanced the answers of *A* and *B*. We recorded participants' response with multiple-choice questions with 3 options with *A*, *B*, and *unsure*.

3.4 Visual Configuration

The width of the bar chart was fixed to 628 pixels, which was the same as the median grid line of the circumference of a polar chart with 200-pixels radius, as seen in Fig. 1-middle column and Fig. 1-right column. The y-axis scale of the bar chart was same as the y-axis scale in the polar chart, i.e., 200 pixels. This setting was similar to prior visualisation studies comparing polar and bar charts [44]. For the study, we removed all tick marks, grid lines and labels, such that visual comparison is based purely on bar height. We used a monochrome colour for the bars, as Adnan et al. [1] found colour visual encoding makes the effect of linear or polar chart negligible for time series visualisation. For the polar chart, we used an inner circle with a radius of 50 pixels for showing a curve. This marks the bars in each task, as shown in Fig. 2. The corresponding visual encoding was an underline for bar charts. To support Trend Identification, each chart was accompanied by a

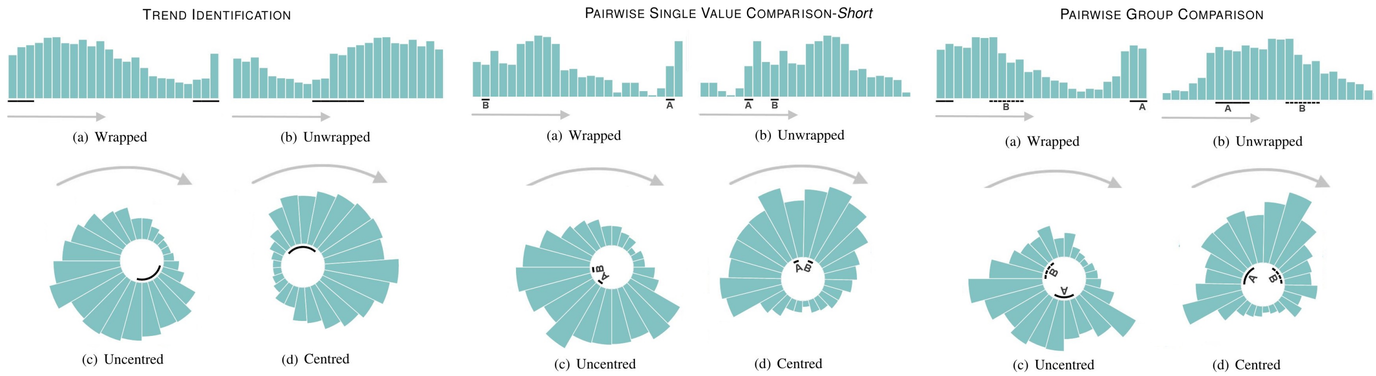


Fig. 2. Examples of data sets and visual stimuli for all three tasks used in the study: TREND IDENTIFICATION(left), PAIRWISE GROUP COMPARISON(centre) and PAIRWISE SINGLE VALUE COMPARISON(right). Sub figures (a)-(d) show the individual conditions for wrapped and unwrapped as well as centred and uncentred across both visualisation techniques.

grey arrow indicating the direction of time inside the visualisation. For the linear bar charts, the arrow was shown from the left to the right on the bottom of the chart. We implemented each technique using D3.js [9].

3.5 Data

Our stimuli were generated from New York hourly traffic accident datasets between 2013 and 2016 [42]. We selected Brooklyn, Bronx, Manhattan, Queens, and Staten Island borough data. These data show a non-smooth distribution of the aggregated number of traffic accidents over 24 hours, which we felt would provide a suitable and ecologically-valid data set for our study. The same data has been used in prior time series visualisation studies comparing linear and polar charts [44]. To generate cyclical temporal data for the study tasks, we first aggregated the hourly traffic accidents on a monthly basis for each borough. We then obtained average hourly traffic accident datasets in 24 hours (i.e., 24 data points) for a given borough, month and year to obtain hundreds of candidate sample data sets. We selected from these candidates for our task stimuli, controlling for difficulty based on pilot testing, as follows:

- **TREND IDENTIFICATION:** Across all samples we restricted the candidates to instances with a monotonically increasing or decreasing set of six adjacent bars, with a minimum height difference of 1-2% (i.e., small but apparent on the displays tested). For neither monotonically increasing nor decreasing instances, there is no other monotonic sequences of more than two bars. Furthermore, we added one quality control (obvious) trial per technique with a much larger minimum height difference of greater than 20%.
- **PAIRWISE GROUP COMPARISON:** The difference of total values between two groups was 5-15% of bar chart height. For the quality control trial, the difference is set to be greater than or equal to 20% of bar chart height.
- **PAIRWISE SINGLE VALUE COMPARISON:** The difference of heights between the two bars was 1-1.5% of bar chart height. For the quality control trial, the difference is set to be greater than or equal to 20% of the bar chart height.

3.6 Experimental Design

We decided on a within-subject design with 4 techniques $\times$ 3 tasks. We used 10 repetitions for TREND IDENTIFICATION, PAIRWISE GROUP COMPARISON, and 6 repetitions for PAIRWISE SINGLE VALUE COMPARISON-*Short* and 6 repetitions for PAIRWISE SINGLE VALUE COMPARISON-*Long*. We inserted a quality control trial for each technique per task to check participants' attention. We used 4 practice trials for each technique per task. This gives a total of 47 trials $\times$ 4 techniques = 188 trials per participant. We used a full-factorial design to counterbalance the learning effect of four techniques (24 orderings). Each recorded trail has a timer associated with the task type, as described in Sect. 3.3. We expected the study to complete within 40 minutes. The order of *wrapped*, *unwrapped*, *centred, and uncentred* questions for each task was randomised across tasks per technique. Each participant went through the same order of trials. Each trial used a randomly selected dataset from 5 borough data that satisfied the constraints as described in Sect. 3.5. Therefore, none of the same graphics appear twice throughout the study.

We used equal numbers of trials for each of these four groups. We recorded *task-completion time* (TIME), *task-error* (ERROR), and *subjective user preference* (PREF) as dependent variables across all the tasks.

3.7 Participants and Procedure

We crowdsourced the study via both convenience sampling and the Prolific Academic system [31]. The participants on Prolific Academic platform have been reported producing data quality comparable to Amazon Mechanical Turk [32]. Many time series visualisation studies were in fact crowdsourced [3,11,17,18,34,35,44]. Recent visualisation studies were deployed over the Prolific platform [34].

We hosted the study on our web server application. We set a pre-screening criterion on performance, i.e., minimum approval rate of 95%, and minimum number of previous submissions of 10. We also limited our study to desktop users with larger screens. We provided a payment of £5 (i.e. a rate of £7.5/h) to Prolific participants. This is considered good payment according to the Prolific Academics platform. We recorded 72 participants who passed quality control tasks and completed the study. This comprised 3 full counterbalanced blocks of participants. 48 participants from the Prolific group and 24 participants from local sampling (recruited via email). 23 were females, 47 were males and 2 preferred not to disclose their gender. The age of participants is between 18 and 50. Each participant went through all of the 4 techniques with the order assigned by the software with the following procedure. First, they completed a tutorial explaining the techniqueand task.

For TREND IDENTIFICATION, participants were instructed to read the trend from a set of adjacent bars highlighted by either an underline (for bar charts) or a curve (for polar charts). Examples of monotonically increasing, monotonically decreasing, and neither were given during the training as well as in practice trials. An example of neither is shown in Fig. 2-TREND IDENTIFICATION(a-d) and Sect.1.1 of the supplementary file. During the training, when the participant's answer was incorrect they would be shown the same practice trial again until a correct answer was given.

For INTERACTIVEBAR and INTERACTIVEPOLAR, an animated image demonstrating the wrapping or rotation interaction was shown for the task. Participants were encouraged to try the interaction themselves with the same example as the one in the animation. Following that, participants were required to successfully complete 4 practice trials before proceeding to the recorded trials. For recorded trials, each trial was first loaded on a participant's browser before the software started the timer. Each participant went through the same task order, i.e., Trend Identification→Pairwise Group Comparison→Pairwise Single Value Comparison-*Short*→ Pairwise Single Value Comparison-*Long*.

The web link to the experimental software is available online: `https://observablehq.com/@kun-ting/rotate-or-wrap`.

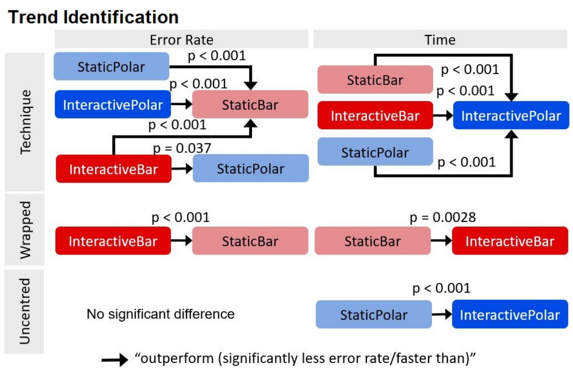


Fig. 3. Statistically significant results of Trend Identification

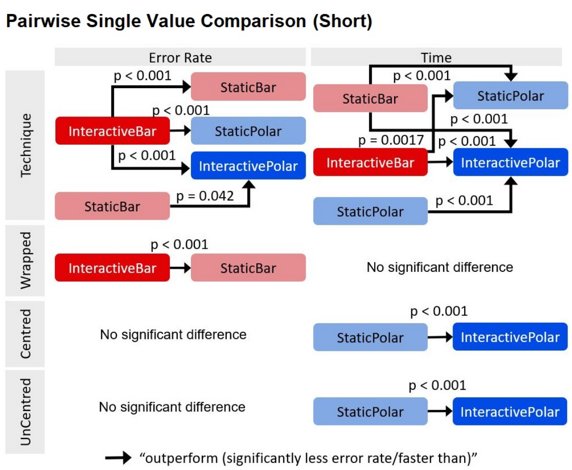


Fig. 4. Statistically significant results of Pairwise Single Value Comparison (Short distance - 2 bars apart)

3.8 Results

We report on the results of 72 participants from both Prolific group and convenience sampling group. All of these participants passed the attention check trials, completed the training and recorded trials. This resulted in 9,216 trials. We excluded attention check trials in the analysis, as they used a much looser constraint than recorded trials, as described in Sect. 3.5. Therefore, we have equal numbers of *wrapped*, *unwrapped* trials for bar charts, and equal numbers of *centred* and *uncentred* trials for polar charts for each task in the analysis. Since the distribution of Error and Pref of each technique did *not* follow a normal distribution, we used Friedman's non-parametric test and Tukey's posthoc pairwise comparison to identify significant differences between StaticBar, InteractiveBar, StaticPolar, and InteractivePolar. The confidence interval is 95%. Since Time of each technique was normally distributed, tested with Shapiro's normality test and visually checked by Q-Q plot, we used ANOVA repeated measures and Tukey's posthoc pairwise comparison to test significance. We report on the most significant findings for Trend Identification and Pairwise Single Value Comparison-Short task visually in Fig. 3 and Fig. 4. The results of error bar graphs are shown in Fig. 5 and Fig. 6. The detailed statistical results can be found in Sect. 1.3 of the supplementary file.

3.9 Qualitative User Feedback

The majority of participants reported more confidence in using InteractiveBar across all tasks. Some participants mentioned that panning the chart brings bars closer to one another, making it significantly easier to inspect bars and come to a decision. Some participants reported more confidence in StaticBar than StaticPolar, citing that the StaticPolar looked more confusing and was harder to read. Other participants mentioned they had to turn their head or neck to spot the answer and static polar charts caused confusion about comparing the bars, and therefore they ranked it the worst.

To our surprise, StaticBar was not significantly favoured over InteractivePolar. Some participants mentioned the ability to rotate allows them to see in different angles and make them feel more confident in the analysis. Some other participants mentioned the polar charts provided a more panoramic view. StaticBar made it hard to focus when seeing broken ranges or bars that are located far apart.

3.10 Study Conclusion

Overall, the results of our study indicate that interactive wrapping leads to significant improvements in error over static representations for Trend Identification and Pairwise Single Value Comparison, despite the additional time required to pan.

Returning to our questions from the Introduction, to answer **(Q1)** *"Does adding interactive wrapping to linear bar charts improve their effectiveness?"*, we begin by noting that contrary to findings from past studies [1, 11, 44] (that static linear bar charts outperform static polar bar charts), we found that for Trend Identification across the 'cut' in linear bar charts, the polar representation is actually significantly better in terms of error rate. However, introducing interaction to linear bar charts reverses this result. We can conclude that the InteractiveBar clearly outperformed StaticBar for Trend Identification and Pairwise Single Value Comparison tasks in terms of error, especially for trials where the bars being compared are separated by the 'cut'. These results allow us to reject the null-hypothesis for H1.1-error for some tasks. We found that InteractiveBar results in less errors than InteractivePolar for some tasks, rejecting the null-hypothesis for H1.2-error and sometimes is significantly faster, rejecting the null-hypothesis for H1.2-time.

While we found that StaticBar was faster than InteractiveBar across all tasks except Pairwise Single Value Comparison-Short (rejecting the null-hypothesis for H1.1-time for some tasks), the time spent actually moving the charts was a notable fraction of the trial time. We see from the textured parts of bars in Fig. 5-Trend Identification and Fig. 5-Pairwise Single Value Comparison-*Short* that for InteractiveBar the interaction time in Trend Identification and Pairwise Single Value Comparison-short tasks were greater than the time difference compared to StaticBar. We observed that most people use the interaction once, moving InteractiveBar or StaticPolar so that they could comfortably solve the task. While it is tempting to assume the time difference is due to the extra time spent performing the interaction, it is impossible to know whether people are able to actively reason about the visualisation during interaction, suggesting an interesting direction for future study.

We also found that InteractiveBar significantly outperformed StaticPolar in terms of error for Trend Identification and Pairwise Single Value Comparison **(Q2)**. We can thus reject the null-hypothesis for H2-error for these tasks. We conclude that adding interactive wrapping to bar charts—while incurring a cost in terms of the time spent interacting—decreases errors compared to static polar versions which technically do not require rotation to avoid the cut-problem. This trend is similar across all tasks but significant only for two.

Eventually, while we did not find any significant result that interaction reduces errors in polar charts **(Q3)**, we found StaticPolar was significantly faster than InteractivePolar across all tasks, we

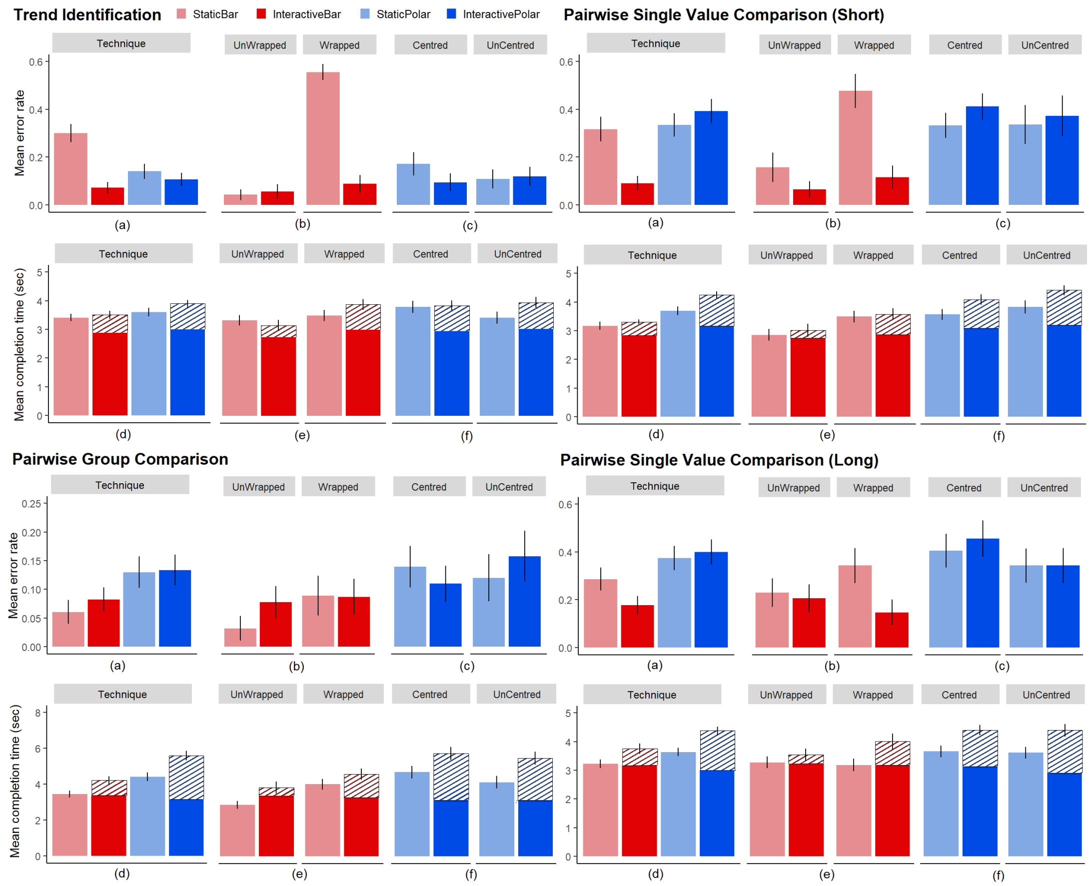


Fig. 5. Quantitative results for ERROR and TIME for all tasks and techniques: bar charts (red) and polar charts (blue). Static techniques are indicated in a lighter colour. Textured parts of some bars indicate the fraction of time used for panning (INTERACTIVEBAR) and rotating (INTERACTIVEPOLAR).The detailed statistical significant results are available in the supplementary file.

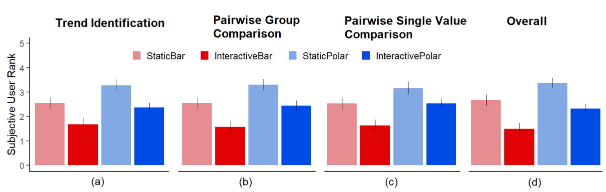


Fig. 6. Subjective User ranking for all tasks and techniques (lower is better). The statistical significant results are available in the supplementary file.

therefore accept the null-hypothesis for H3.1-error and reject the null-hypothesis for H3.1-time. We also found that INTERACTIVEBAR was significantly preferred to STATICBAR, STATICPOLAR, and INTERACTIVEPOLAR. We therefore reject the null-hypothesis for H4.

In summary, we take these results as strongly encouraging interactive wrapping for linear bar charts, especially since the upfront cost of adding an interactive wrapping technique is small and the interaction is *not* required to read the bar charts.

4 DESIGN SPACE OF WRAPPED DATA VISUALISATIONS

Encouraged by the results of our study, this section generalises the concept of interactive wrapping. First, we consider data types that are cyclical in more than one dimension, and hence have a fundamentally different topology to those considered so far. These include time series data that have multiple levels of periodicity (e.g. weekly as well as daily). Second, we consider quite different data types such as networks represented by node-link diagrams [14, 15] or matrices [21] and high-dimensional data visualised by self-organising maps [26, 43] or multi-dimensional scaling. We can describe these types of two-way wrapped visualisations as *toroidal*, since topologically the visualisations are connected not only at the sides, but also at the top and bottom. We now introduce a design space which maps the possibilities for cylindrical and toroidal *topology* against the possibilities for mapping the topology to a two-dimensional visualisation (*projection*) and the affordances for interactive panning of the resulting visualisation (*pannability*).

Thus, our design space is the composition of three dimensions:

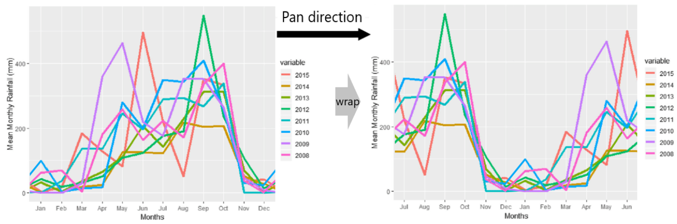
Topology describes whether a visualisation (technique) is mapped onto a cylinder (Fig. 1-top) or torus (Fig. 1-bottom).

Projection describes the projection method used to obtain a 2-dimensional representation from the 3-dimensional topology. A side projection results in a rectangular representation (Fig. 1-middle column), while a top projection results in a concentric radial (or "polar") representation (Fig. 1 right column).

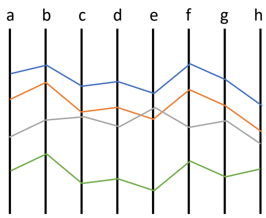
Pannability describes the directions which the user is able to pan the visualisation. To some extent, this is determined not only by the *topology* and *projection* dimensions but also by the type of data and conventions within the application domain.

In Sect. 4.3, we give a gallery of pannable visualisations that emerge from these design dimensions. These are described more fully in that section, but here we briefly illustrate how these examples relate to our design dimensions. For the *pannability* dimension, we consider a side-projected cylinder which is only pannable in one dimension, but the orientation may be horizontal (Fig. 7a-b, Fig. 8) or vertical (Fig. 9). On a top-projected cylinder (Fig. 7c), the user rotates the resulting circular visualisation, so the panning is rotational. A torus *topology* supports two-dimensional panning, either horizontal and vertical (Fig. 10) for a side projection, or rotational and radial panning for a top projection (Fig. 11). For some toroidal projections of certain data types, it makes sense to further constrain the *pannability*. For example, a symmetric matrix could be panned both horizontally and vertically, but it usually makes sense to keep the matrix diagonal centred (Fig. 12).

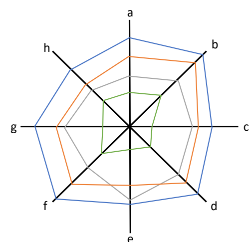
In the following, we motivate our design space by reflecting on the wrap and rotate interactions we evaluated in our user study. This *topo-*



(a) Time series, wrapped horizontally



(b) PCP side-projection



(c) PCP top-projection

Fig. 7. Cylindrical interaction for standard visualisations. (a) Bangkok's average monthly rainfall from 2008 to 2015: a timeseries wrapping horizontally and allowing to perceive patterns within individual time periods (e.g., year.) [27]; (b) a parallel coordinates plot (PCP) with cyclical dimensions in side-projection; (c) the PCP in top-projection radial view.

(a) Two connected lines from the Singapore metro map, causing a cycle.

(b) The map from above, wrapped around a cylinder topology and side-projected.

(c) Result of panning the map above to the left.

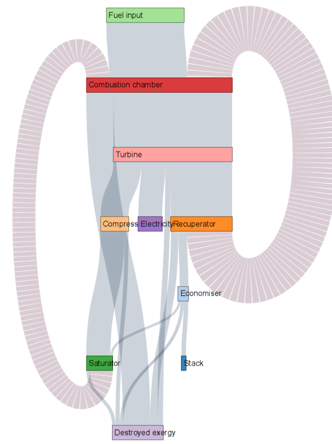
Fig. 8. Example of cylindrical wrapping: transportation map [30]

logical perspective allows us to see not only that these one-dimensional wrap and rotate operations share the same topology (a cylinder, see 4.1), but also that we can extend both the *linear wrap* and *polar rotate* interactions into two dimensions by seeing how they can be implemented in a toroidal topology (see Sect. 4.2). This topological understanding results in a very general design space that encompasses sophisticated wrapping interactions and can suggest novel applications.

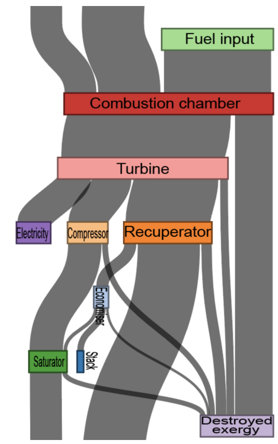
As our design space is meant as a conceptual model, rather than an affordance to 3D rendering, we focus exclusively on the 2D projections that can be created from these views. Our design space can be a powerful tool to: *(a)* understand relations between visualisations (e.g., linear and polar bar charts), *(b)* apply interactive wrapping to a range of visualisations, and *(c)* to create new visualisations by mapping existing visualisations onto cylinder and torus visualisations to obtain side and top projections. In Section 4.3, we explore other types of wrapping visualisations with underlying cylindrical or toroidal interaction topologies.

4.1 Cylindrical Interaction Techniques: Wrap or Rotate

The interaction techniques described in Section 3 can be thought of as stemming from the connectivity in cyclical data being mapped to a 3D



(a) Cyclical links are circular.



(b) Cylindrical projection to allow continuous vertical panning.

Fig. 9. A Sankey Diagram with cycles.

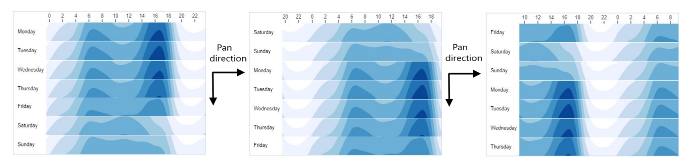


Fig. 10. Example for torus horizontal and vertical torus wrapping, individually.

topology, as per Fig. 1a-b. For the data with a single cyclical domain, the underlying topology we consider for both linear and polar charts is a cylinder. Both the wrap and rotate interactions can be thought of as a rotation of the 3D cylindrical surface before projecting back to a 2D visualisation along one of two possible view angles of the cylinder, as follows.

4.1.1 Side Projection: Wrap Interaction for Bar Charts

A cyclical bar chart can be wrapped around the cylinder such that the full time period covers the circumference of the cylinder, and the range of the bars representing values will extend to the height of the cylinder. A side-view of the cylinder would show half the chart (only the half of the time domain facing the viewer). We can rotate the cylinder to choose the time interval that is centred in the field of view. To recover a 2D view of the whole data domain, we can "slice" the cylinder on the side farthest from the viewer and flatten the resulting sheet, as per Fig. 1a.

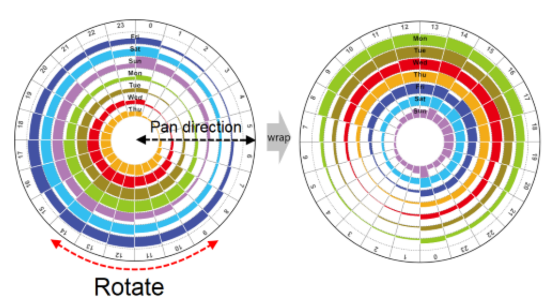


Fig. 11. Example of a top-down view onto a torus topology, allowing to move elements from the centre of the visualisation to its outer boundary.

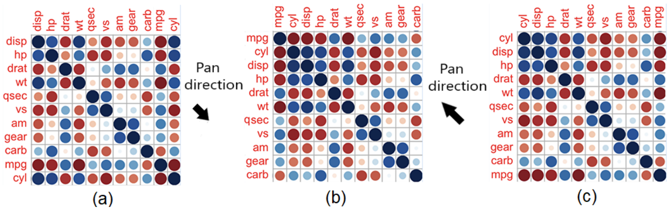


Fig. 12. Effects of diagonal panning on an torus-topology matrix. Using the same row/column ordering: (a) random start and end row/column shows two clusters (blocks along the diagonal), (b) horizontal panning now shows a single cluster (top-left), and (c) panning again highlights strongly correlated `mpg` row/column. Taken from `http://www.sthda.com/english/wiki/visualize-correlation-matrix-using-correlogram`

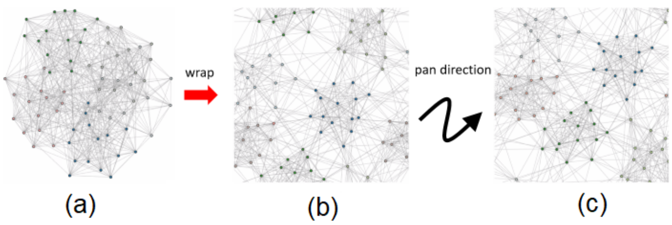


Fig. 13. Examples of torus panning on a network using the random partition nework data set [19] with already known clustering information: (a) traditional (unwrapped) network on a 2D-plane; (b, c) two configurations wrapped network laid-out on a 2D torus, using the algorithm presented in [14].

4.1.2 Top Projection: Rotate Interaction for Polar Charts

For a polar chart, imagine shrinking the base of the cylinder such that it forms a cone, with the bars extending from the narrow base, outward to the wider top. The polar chart is then simply a view of the cone with the point facing the viewer. Rotation of the cone now changes which bar is centred at the top.

4.2 Toroidal Interaction Techniques: Combining Wrapping and Rotation

4.2.1 Side Projection: 2D Wrapped Bar Chart Arrays

The concept of a visualisation that can wrap in two-dimensions is best understood, initially, by extending our earlier charts. In Fig. 1c, we further break down the hourly traffic accident data by day of the week. Each bar still represents one hour, but there are now seven small multiple visualisations, one for each day. Like the 24-hour day, the seven-day week is also cyclical. It's conventional to display the week from Sunday to Saturday, with the weekdays centred and consecutive. But what happens if an analyst particularly wants to compare Saturday activity against Sunday? Here is where we can introduce a second (vertical) dimension of interactive wrapping, such that vertical mouse or touch drags cause the day order to wrap, while still allowing horizontal drags to wrap the hours. Topologically, this can be seen as a side view of the torus, where now the hour dimension spans the long circumference of the torus, while the days progress around the circumference of a torus segment (as per Fig. 1-lower-left).

4.2.2 Top Projection: Polar chart arrays that wrap and rotate

In Fig. 1d, we introduce a new type of visual inspired by a top-down view of the torus. The result is a polar visualisation where, as before, the disc can be rotated through a mouse or touch drag tangential to the disc to change the hour that is centred at the top. In addition, a mouse drag outward from, or towards, the centre of the disc changes the day order.

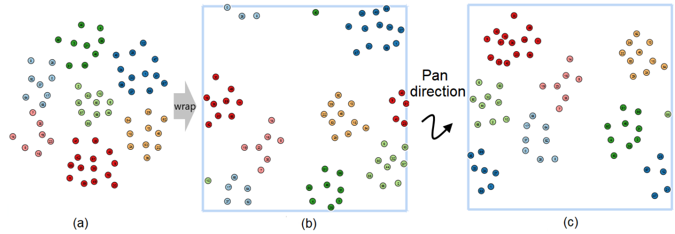


Fig. 14. Examples of torus panning on an MDS using the Hepta data set [39] with already known clustering information: (a) traditional (unwrapped) MDS plot on a 2D-plane; (b, c) two configurations wrapped MDS plot laid-out on a 2D torus, using the algorithm presented by Chen et al. [14].

4.3 Mapping and Extending other types of visualisations to Cylinder or Torus Interaction Topologies

In the following, we discuss each of the two topologies and give further examples. We show how each topology leads to interactive wrappable visualisations and for which kinds of data wrappable visualisations make most sense. To create wrappable visualisations, we implemented a small web tool[1] that can present arbitrary 2D visualisations with one- or two-dimensional wrap panning and allows for the selection of panning constraints in accord with the topology (cylinder or torus). The user can upload separate images for the visualisation and optional axes-labels that remain fixed at the sides while the visualisation is being panned.

4.3.1 Cylinder Topology Examples

Cylinder wrapping can be applied to any visualisation that benefits from panning continuously along one spatial dimension, just as we were able to demonstrate for the bar charts in our study.

Apart from bar charts, cylindrical wrapping can be applied to other types of visuals in a straightforward manner. Fig. 7a shows a set of average yearly rainfall data shown as lines plotted across 12 months. Similarly, parallel coordinates plots of multidimensional data, where the dimensions have a sensible cyclical ordering, can be mapped to the cylinder and be either side-projected (Fig. 7b), or top-projected in a radial form (Fig. 7c).

However, the concept can equally be applied to other types of data. For example, Fig. 8a shows a transportation map (two intersecting lines from the Singapore Metro) with a cycle. We show that by redrawing the map on a cylinder such that the loop is routed around the circumference, and then by a side-projecting, we can unwrap the cycle (Fig. 8b). The projected view can then be endlessly panned left or right (Fig. 8c). Such a representation could be used on (e.g.) a circle line train to continuously show the stations ahead in the order they will be visited. Such narrower maps can be shown more space-efficiently on, e.g., narrow static or digital displays above doors inside trains.

In a similar (cyclical network) vein, Fig. 9 shows a Sankey diagram depicting thermodynamic analysis of water injection in a micro gas turbine [13]. Sankey diagrams have become popular in information visualisation for showing movements of data elements between different groupings. For example, Google Analytics uses them to show click-through behaviour of users of web pages. However, in such abstract information visualisation they are rarely depicted with cycles, possibly because they start to look messy, as in Fig. 9a[2]. In Fig. 9b, we redraw the same Sankey diagram on a cylinder topology, and project from the side to afford vertical panning.

4.3.2 Torus Topology Examples

As we have seen, in a torus topology, a 2D visualisation may be mapped to a torus surface provided the top and bottom edges as well as the

[1] `https://github.com/Kun-Ting/WrappingChart`

[2] generated with a fork of the d3-Sankey software: `http://bl.ocks.org/soxofaan/bb6f91d57dc4b6afe91d`

left and right edges of the 2D visualisation meet up (Fig. 1). A side projection of the torus then leads to visualisations that can be panned in both dimensions at the same time. In user interface design, the earliest example of such torus wrapping may be the classic *Asteroids* game[3]. Torus topologies have also been explored for mouse pointers on computer screens [25]. However, in information visualisation, torus topologies have been used experimentally for network visualisation [15], but otherwise have not previously been systematically explored. Some algorithmic work has been applied to create self-organising maps (SOMs) [26, 43] on a torus, but visualisation has always happened in a static plane.

A top projection of a torus results in a polar visualisation, similar to top-down views on cylinder topologies. The difference is that panning can now happen in two ways: using rotation along one dimension and using wrapping to move elements from the *inside* of the polar visualisation to its outside (Fig. 11). This kind of panning can also overcome the common problem of polar visualisations that visual information at its centre is rendered smaller than on its outer boundary.

4.3.3 Bespoke layouts:

While wrapping some 2D-visualisations, such as heatmaps, matrices, or horizon graphs is straighforward, other visualisations can benefit from optimising their layout or embedding. Such examples include node-link representations for networks [14,15], multi-dimensional scaling (MDS), or SOMs. For node-link diagrams, previous work investigated drawing strictly crossing-free graphs over 2D torus topology with partial or full context of the layout [29]. Chen et al. later introduced a layout algorithm for network embeddings on a torus and showed how interactive toroidal wrappings improve understanding clusters in node-link diagrams [14]. They used a freely pannable 2D-plane to explore the layout (Fig. 13).

Using the same algorithm, Fig. 14 shows a multi-dimensional scaling (MDS) calculated on a torus topology. Fig. 14(a) shows an unwrapped original 2D MDS plot with 70 points and 7 classifications on a 2D-plane, using a standard force-directed layout algorithm. Fig. 14b and Fig. 14c show our torus layouts in two different panning configurations (free vertical and horizontal panning, no panning constraints).

4.3.4 Constrained panning

While panning direction is constrained naturally in cylinder topologies to only one dimension (e.g. horizontally or vertically), on toroidal topologies, users can freely pan along both spatial dimensions. Such free panning allows for rapid navigation, however, in many visualisations, spatial dimensions have meaning and constraining interactions to one dimension at a time facilitates exploration.

For example, Fig. 10 shows an horizon graph [24] for aggregated time series data over days (vertically) and hours (horizontally). Panning horizontally will cycle through the hours of the day (`0-24h`) and panning vertically will cycle through the days of the week (`Mon-Sun`). Panning in this example is restricted to one of these dimensions at a time to allow for exploring values across either days or hours without accidentally changing the context of the other dimension. For example, an analyst might be interested in exploring the exact time of peaks across days, and therefore requires the hour-dimension (horizontal) to remain fixed, while panning vertically through the days.

Another example for a torus topology using the constraint panning are adjacency matrices for network visualisation (Fig. 12). Some matrix orderings are in fact *cycles* such as those based on the travelling salesman problem [8]. These matrix orderings make the matrix a torus since their start and end point have to be taken randomly from the permutation and matrices can be panned along the two spatial dimensions. Which of the elements becomes the first and last row is usually determined using external heuristics such as the highest-degree node, however, there is no given starting point. Choosing a random starting point from a permutation can lead to arbitrary patterns in a matrix. For example, the matrix in Fig. 12a seems to show two strong clusters, one on the top-left and one on the bottom-right. The torus panning for matrices therefore can serve two purposes: *(a)* learning and communicating the idea of ordering rows and columns and *(b)* exploring a specific ordering to avoid, e.g., overlooking cells at the margins of the matrix or misinterpreting clusters cut in half by the ordering (Fig. 12a).

For undirected networks, matrices are symmetric to the diagonal (usually top-left to bottom-right diagonal, if both rows and columns run left-right and top-down respectively). To preserve this important feature of symmetry and to keep viewers' mental map of the matrix preserved, we can constrain the panning so that user can pan only along the diagonal, i.e., the diagonal is fixed while vertical and horizontal panning happen at the *same* time.

Panning the matrix diagonally (Fig. 12b) reveals that the two clusters visible in Fig. 12a are in fact one large cluster situated at the top-left of the matrix. Panning further along the diagonal for just *one* single row (Fig. 12c) highlights the `mpg` row/column by emphasising its strong connections (correlations) with any of the other nodes in the network. The same "pan-configuration" also highlights the red cells in the `mpg`-row/column, which depict negative correlations. In contrast, the configuration in Fig. 12b visually emphasises the *strength* of the correlation, rather than their *type*. In summary, torus wrapping can help exploration and scrutinising data as well as finding the views most appropriate for a given task or message.

5 Discussion and Future Work

In our study, we exclusively focus on evaluating cyclic temporal data in one spatial dimension to keep the crowdsourced study from becoming overly complicated. Based on the study results, we might hypothesise that, for more complicated case of 2D temporal data exploration in a 2D torus topology, interactive wrapping with two spatial dimensions may still outperform the rotational and wraparound polar chart. In future, we intend to further investigate this hypothesis. Another direction is to investigate if the performance benefits afforded by interactive wrapping across the boundaries also applies to other non-temporal cyclic data types, such as geographic maps or flow diagrams with cycles. We would also like to further investigate the aspects of the wrapping method that drive better performance. For example, we cannot say whether the interactive panning we provided is better than a passive animation of wrapping affording different views. Our feeling, however, is that the interaction gives users a better understanding of the paradigm as well as a sense of control.

Our design space exploration yielded a number of novel applications of toroidal wrapping, such as multidimensional scaling plots, and matrices. We would like to study these further and investigate if such interactive wrapped visualisations could be practically usable by domain experts.

6 Conclusion

We have presented a study comparing the effect of interactive wrapping of bar charts and rotation of polar charts on the readability of real-world cyclical data. Our study is the first to demonstrate that a pannable wrapped visualisation offers significant benefits in terms of error rate over the equivalent static visualisation for reading ranges and comparing values that are split across the edges of the chart. Specifically, our study indicates that standard bar charts with interactive wrap panning offers significant benefits over standard unwrapped bar charts in accuracy for reading ranges or value comparison tasks and overall user preference. However, interactive bar chart is significantly slower than standard bar charts. For polar charts, interactive panning makes them significantly slower and less accurate than static polar chart or bar chart, except for tasks of reading monotonic ranges. In the latter, standard bar chart is least accurate due to the need to mentally connect discontinued ranges across two ends of the charts.

Our design space opens a new class of wrappable visualisations and identifies data types that are wrappable. Cylindrical topologies are well suited to data that is cyclical in one dimension. Toroidal topologies are useful when the data has two cyclical dimensions and for relational data that can be arranged onto such a topology.

[3] https://en.wikipedia.org/wiki/Asteroids_(video_game)

REFERENCES

[1] M. Adnan, M. Just, and L. Baillie. Investigating time series visualisations to improve the user experience. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5444–5455, 2016.

[2] W. Aigner, S. Miksch, H. Schumann, and C. Tominski. *Visualization of time-oriented data*. Springer Science & Business Media, 2011.

[3] D. Albers, M. Correll, and M. Gleicher. Task-driven evaluation of aggregation in time series visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 551–560, 2014.

[4] N. Andrienko and G. Andrienko. *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Science & Business Media, 2006.

[5] B. Bach, P. Dragicevic, D. Archambault, C. Hurter, and S. Carpendale. A descriptive framework for temporal data visualizations based on generalized space-time cubes. In *Computer Graphics Forum*, vol. 36, pp. 36–61. Wiley Online Library, 2017.

[6] B. Bach, N. Henry-Riche, T. Dwyer, T. Madhyastha, J.-D. Fekete, and T. Grabowski. Small multipiles: Piling time to explore temporal patterns in dynamic networks. In *Computer Graphics Forum*, vol. 34, pp. 31–40. Wiley Online Library, 2015.

[7] B. Bach, C. Shi, N. Heulot, T. Madhyastha, T. Grabowski, and P. Dragicevic. Time curves: Folding time to visualize patterns of temporal evolution in data. *IEEE transactions on visualization and computer graphics*, 22(1):559–568, 2015.

[8] M. Behrisch, B. Bach, N. Henry Riche, T. Schreck, and J.-D. Fekete. Matrix reordering methods for table and network visualization. In *Computer Graphics Forum*, vol. 35, pp. 693–716. Wiley Online Library, 2016.

[9] M. Bostok. *https://d3js.org/*, 2011 (accessed 26 March, 2021).

[10] M. Brehmer, B. Lee, B. Bach, N. H. Riche, and T. Munzner. Timelines revisited: A design space and considerations for expressive storytelling. *IEEE transactions on visualization and computer graphics*, 23(9):2151–2164, 2016.

[11] M. Brehmer, B. Lee, P. Isenberg, and E. K. Choe. Visualizing ranges over time on mobile phones: a task-based crowdsourced evaluation. *IEEE transactions on visualization and computer graphics*, 25(1):619–629, 2018.

[12] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE transactions on visualization and computer graphics*, 19(12):2376–2385, 2013.

[13] M. M. Carrero, W. De Paepe, S. Bram, and F. Contino. Thermodynamic analysis of water injection in a micro gas turbine: Sankey and grassmann diagrams. *Energy Procedia*, 105:1414–1419, 2017. 8th International Conference on Applied Energy, ICAE2016, 8-11 October 2016, Beijing, China. doi: 10.1016/j.egypro.2017.03.527

[14] K.-T. Chen, T. Dwyer, B. Bach, and K. Marriott. It'sa wrap: Toroidal wrapping of network visualisations supports cluster understanding tasks. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2021.

[15] K.-T. Chen, T. Dwyer, K. Marriott, and B. Bach. Doughnets: Visualising networks using torus wrapping. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2020.

[16] W. S. Cleveland and R. McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American statistical association*, 79(387):531–554, 1984.

[17] M. Correll, D. Albers, S. Franconeri, and M. Gleicher. Comparing averages in time series data. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1095–1104, 2012.

[18] S. Di Bartolomeo, A. Pandey, A. Leventidis, D. Saffo, U. H. Syeda, E. Carstensdottir, M. Seif El-Nasr, M. A. Borkin, and C. Dunne. Evaluating the effect of timeline shape on visualization task performance. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2020.

[19] S. Fortunato. Community detection in graphs. *Physics reports*, 486(3-5):75–174, 2010.

[20] J. Fuchs, F. Fischer, F. Mansmann, E. Bertini, and P. Isenberg. Evaluation of alternative glyph designs for time series data in a small multiple setting. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 3237–3246, 2013.

[21] M. Ghoniem, J.-D. Fekete, and P. Castagliola. On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis. *Information Visualization*, 4(2):114–135, 2005.

[22] R. L. Harris. *Information graphics: A comprehensive illustrated reference*. Oxford University Press, USA, 1999.

[23] J. Heer and M. Bostock. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 203–212, 2010.

[24] J. Heer, N. Kong, and M. Agrawala. Sizing the horizon: the effects of chart size and layering on the graphical perception of time series visualizations. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1303–1312, 2009.

[25] S. Huot, O. Chapuis, and P. Dragicevic. Torusdesktop: pointing via the backdoor is sometimes shorter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 829–838, 2011.

[26] M. Ito. The characteristics of the torus self organizing map. In *Proceedings 16th Fuzzy Systerm Symposium Akita, 2000*. Japan Society for Fuzzy and Systems, 2000.

[27] R. Jain, S. Sontisirikit, S. Iamsirithaworn, and H. Prendinger. Prediction of dengue outbreaks based on disease surveillance, meteorological and socio-economic data. *BMC infectious diseases*, 19(1):1–16, 2019.

[28] A. Karduni, R. Wesslen, I. Cho, and W. Dou. Du bois wrapped bar chart: Visualizing categorical data with disproportionate values. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2020.

[29] W. Kocay and D. L. Kreher. *Graphs, algorithms, and optimization*. Chapman and Hall/CRC, 2016.

[30] M. of Singapore. *https://mapa-metro.com/en/Singapore/Singapore/Singapore-MRT-map.htm*, 2010 (accessed 26 June, 2021).

[31] S. Palan and C. Schitter. Prolific. ac—a subject pool for online experiments. *Journal of Behavioral and Experimental Finance*, 17:22–27, 2018.

[32] E. Peer, L. Brandimarte, S. Samat, and A. Acquisti. Beyond the turk: Alternative platforms for crowdsourcing behavioral research. *Journal of Experimental Social Psychology*, 70:153–163, 2017.

[33] B. Saket, A. Endert, and Ç. Demiralp. Task-based effectiveness of basic visualizations. *IEEE transactions on visualization and computer graphics*, 25(7):2505–2512, 2018.

[34] K. A. Satriadi, B. Ens, T. Czauderna, M. Cordeil, and B. Jenny. Quantitative data visualisation on virtual globes. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2021.

[35] M. Schwab, S. Hao, O. Vitek, J. Tompkin, J. Huang, and M. A. Borkin. Evaluating pan and zoom timelines and sliders. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2019.

[36] M. Schwab, A. Pandey, and M. A. Borkin. Evaluation of 1d selection techniques for mobile visualizations. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–7, 2021.

[37] D. Simkin and R. Hastie. An information-processing analysis of graph perception. *Journal of the American Statistical Association*, 82(398):454–465, 1987.

[38] J. Talbot, V. Setlur, and A. Anand. Four experiments on the perception of bar charts. *IEEE transactions on visualization and computer graphics*, 20(12):2152–2160, 2014.

[39] M. C. Thrun and A. Ultsch. Clustering benchmark datasets exploiting the fundamental clustering problems. *Data in Brief*, p. 105501, 2020.

[40] C. Tominski and H. Schumann. Enhanced interactive spiral display. In *SIGRAD 2008. The Annual SIGRAD Conference Special Theme: Interaction; November 27-28; 2008 Stockholm; Sweden*, number 034, pp. 53–56. Linköping University Electronic Press, 2008.

[41] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko. Stacking-based visualization of trajectory attribute data. *IEEE Transactions on visualization and Computer Graphics*, 18(12):2565–2574, 2012.

[42] P. Troshenkov. *https://www.kaggle.com/pavetr/nypdcollisions/version/1*, 2017 (accessed 26 March, 2021).

[43] A. Ultsch. Maps for the visualization of high-dimensional data spaces. In *Proc. Workshop on Self organizing Maps*, pp. 225–230, 2003.

[44] M. Waldner, A. Diehl, D. Gračanin, R. Splechtna, C. Delrieux, and K. Matković. A comparison of radial and linear charts for visualizing daily patterns. *IEEE transactions on visualization and computer graphics*, 26(1):1033–1042, 2019.

[45] J. Zacks and B. Tversky. Bars and lines: A study of graphic communication. *Memory & cognition*, 27(6):1073–1079, 1999.

[46] J. Zhao, F. Chevalier, E. Pietriga, and R. Balakrishnan. Exploratory analysis of time-series with chronolenses. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2422–2431, 2011.