ARTICLE TEMPLATE

Evaluating Graphical Manipulations in Automatically Laid Out LineSets

ARTICLE HISTORY
Compiled August 22, 2019

ABSTRACT
This paper presents an empirical study to determine whether alterations to graphical features (colour and size) of automatically generated LineSets improve task performance. LineSets are used to visualize sets and networks. The increasingly common nature of such data suggests that having effective visualizations is important. Unlike many approaches to set and network visualization, which often use concave or convex shapes to represent sets alongside graphs, LineSets use lines overlaid on a graph. LineSets have been shown to be advantageous over shape-based approaches. However, the graphical properties of LineSets have not been fully explored. Our results suggest that automatically drawn LineSets can be significantly improved for certain tasks through the considered use of colour alongside size variations applied to their graphical elements. In particular, we show that perceptually distinguishable colours, lines of varying width, and nodes of varying diameter lead to improved task performance in automatically laid-out LineSets.

KEYWORDS
Set visualization; LineSets; graphical properties

1. Introduction

A large number of techniques have been devised for visualizing data whose items are grouped into sets and are related to each other in some way (B.Alsallakh et al. 2014). These techniques often use closed curves, or variations thereof, to represent the groups whereas the network (corresponding to the items and their connections) is represented by a graph; such techniques include BubbleSets (Collins, Penn, and Carpendale 2009), KelpFusion (Meulemans et al. 2013), and EulerView (Simonetto, Auber, and Archambault 2009). An alternative technique, called LineSets, was proposed by Alper et al. (Alper et al. 2011), which instead uses lines to represent sets, overlaying them on an already drawn graph. LineSets are not alone in representing sets using lines (Alper et al. 2011; Cheng 2011; Gottfried 2015; Rodgers, Stapleton, and Chapman 2015), but are the only technique of which the authors are aware that combine lines and graphs to represent set and network data. This paper sets out to shed light on how to manipulate selected graphical properties of LineSets which have been drawn automatically by software, as opposed to having been drawn by hand. Domains such as social media generate vast amounts of data that needs to be filtered, queried and presented in dynamic contexts. Thus, studying the effect of graphical features in LineSets that have been algorithmically generated is essential if we are to improve their effectiveness as a visualization technique for these kinds of data.

An example can be seen in Figure 1. It represents four sets, Apple, Dell, Lenovo and

Sony, using four (coloured) set-lines. The nodes passed through by the set-lines are elements of the respective sets. So, for instance, the set Apple contains two elements that are also in the set Lenovo. Taking the elements to be people who bought the respective brands, and the black lines in the underlying node-link diagram to indicate friendship, we can see also that the two people who bought Apple products are friends. In summary, set-lines are coloured and represent sets. The set-lines are overlaid on the underlying network, drawn in black. As LineSets is an overlay technique (the set-lines are drawn after – overlaid on – the network), it can be applied to many different data sets.

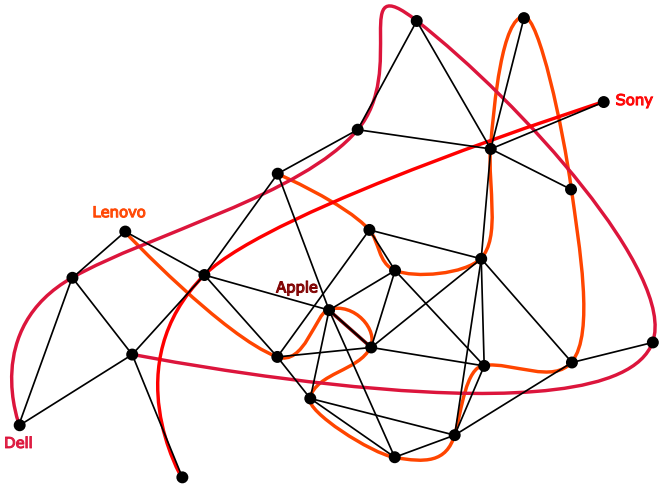

Figure 1.: Using LineSets to visualize grouped network data.

In contrast to LineSets, many visualizations of sets use closed curves instead of set-lines. Euler diagrams are an example which exploits overlapping regions, surrounded by closed curves, to convey information about sets (B.Alsallakh et al. 2014). Given an Euler diagram, when overlaying a network, the closed curves constrain the location of the nodes (data items) and can therefore lead to a less than ideal layout of the network's edges. By contrast, when using Linesets the network is drawn first, with the set-lines subsequently drawn over the top of the network. Even though closed curves can appear to be a natural representation of sets, recent evidence suggests that using lines can be more effective (Rodgers, Stapleton, and Chapman 2014). In addition, compared to other techniques that overlay group information on node-link diagrams, such as GMap and simple node colouring, Jianu et al.'s study found LineSets to be a promising alternative to GMap diagrams for a range of task types: group-only tasks, network-only tasks and group-network tasks (Jianu et al. 2014). This evidence, together with the fact that LineSets do not compromise the layout of the network, leads us to conjecture that improving the design of LineSets could make them more effective than techniques based on closed curves when visualizing grouped network data. So it is important to understand how the choice of their graphical properties impacts on task performance. The intention of the work in this paper is to improve the visualization of LineSets so that users can better comprehend the underlying data.

Alper et al.'s LineSets paper, which introduced the technique, focussed on exploring the potential of LineSets (Alper et al. 2011). They found that LineSets outperformed BubbleSets (Collins, Penn, and Carpendale 2009) when people performed set membership and intersection tasks. Their studies also evaluated how to best draw the set-lines in order to connect elements. They established two simple design guides: LineSets

should be generated with set-lines that (a) are as linear as possible and (b) smooth. The former guide is related to the position of the set-lines in the plane: specifically it suggests that the relative *position of the points* on the set-line *in the plane* should be close to linear. The latter guide is loosely related to the *graphical property* of the shape of the set-lines: whilst not prescribing a particular shape, this guide indicates that smoothness should be prioritised over jagged lines with sharp turns, for instance. Both guides relate to the routing of the set-lines, which is an integral part of the layout algorithm produced by Alper et al. to automatically draw LineSets.

The shape of the set-lines and position of the points on the line are just two features of LineSets that can be altered. Bertin, in his Semiology of Graphics, divides graphical features into elements and their properties, highlighting our perceptual sensitivity to them (Bertin 1983). In LineSets, the graphical elements are the set-lines and the nodes and edges of the underlying network. Graphical properties include their size, colour, shape, and orientation. We exploit Bertin's seminal insights when identifying graphical properties that can be manipulated in LineSets in order to, potentially, impact their effectiveness for users. Since we are concerned with the use of LineSets as a visualization technique, we will focus on the graphical properties that can be readily altered in the implemented software developed by Alper et al. (as opposed to those that are integral parts of the layout algorithm). This will lead to results that can be readily included in existing software, maximising the impact of the results in a computer-human interaction context. The specific contributions of this paper are as follows:

- We apply Bertin's Semiology of Graphics to LineSets, identifying their graphical properties to which we are perceptually sensitive. In this context, we identify existing work that has previously evaluated such properties and highlight its limitations. In particular, we demonstrate that whilst some graphical properties have been shown to impact cognition, as measured by user task performance, no research has been undertaken to evaluate the combined effect of these properties, their effect when used with real-world data, or their incorporation in an automated layout setting. This is covered in section 2.
- We performed an empirical study to compare the combined effect on task performance of selected graphical properties, known to individually enhance the understandability of LineSets, against LineSets as currently produced by Alper et al.'s software. The study compared automatically generated LineSets, produced using the default settings in the software, against the same diagrams but with selected graphical properties altered, namely: the colour of set-lines, the thickness of set-lines, and the diameter of nodes in the underlying network. The data used to produce the LineSets diagrams which were stimuli the study is derived from freely available, real-world data. The study design, execution and results are covered in sections 3 to 5. In addition, we gathered information on user preference which is discussed in section 6.
- The results of the empirical study, discussed in section 7, suggest that the LineSets software can be improved by (a) modifying how colours are assigned to set-lines, (b) prescribing varying set-line thickness to convey cardinality information, and (c) varying node sizes to indicate degree of connectivity. These improvements lead to significantly better task performance and respect the dominantly held user view, established in section 6, that the 'graphically improved' LineSet diagrams are preferable to those drawn using the default software settings.

The threats to validity are presented in section 8, indicating the extent to which

are results are valid and we conclude in section 9. As a consequence of our work, human-computer interaction can be improved since the graphical properties of colour, set-line thickness and node diameter can readily be altered in automatically produced LineSets. The resulting visualizations will lead to significantly improved user task performance and better reflect user preference. All of the diagrams used in the studies, along with the questions and details of the real-world data from which the LineSets for our study were derived can be found in the supplementary material associated with the paper. The anonymized data collected during the study is also included with the supplementary material.

2. Research Motivation and Questions

The choice of graphical properties can affect the comprehension of LineSets. As noted in the introduction, Bertin divides graphical features into elements and their properties. The elements of LineSets are set-lines, nodes and edges. Since we are perceptually sensitive to the properties of these elements, such as their colour, it is important to understand their impact on user task performance. Bertin's *retinal variables* include size, colour value, colour hue, texture, shape, and orientation; particular choices of these correspond to graphical properties, such as the particular size or colour of a graphical element. These are complemented by planar variables, which correspond to the relative position of graphical elements. Since LineSets are an overlay technique, the position of the nodes in the network is predefined and not part of the LineSets layout algorithm per se. Related work includes studies on the design of scroll bars (Alexander et al. 2009) and sliders (Schoeffmann et al. 2010) which demonstrate that increasing detail on interfaces can improve usability. A very significant body of work exists on laying out networks effectively (i.e. choosing *positions* for the nodes and *routes* for the edges) and is not the focus of this paper. In addition, the position of the set-lines (the points in the plane through which the set-lines pass) is not our concern, since the choice of route is pre-determined by the software. We are, though, concerned with those retinal variables which can be readily altered in drawn LineSets. We now consider each of these in turn.

Firstly, we observe that the use of texture does not readily apply to LineSets: texture is normally recognised as useful when representing both qualitative and quantitative differences. Shape, as well as being covered by existing guidance, is (as with the planar variables) an integral part of Alper et al.'s layout algorithm: the heart of the layout algorithm is to determine suitable routes for the set-lines to take. Regarding orientation, it is possible to alter the orientation of the underlying network before routing the set-lines but, again, given the substantial body of work on effective layout of networks, orientation is not a major focus for us. So, we are primarily concerned with the remaining retinal variables: size, colour value, and colour hue. The particular choice of graphical properties corresponding to these variables can be easily altered post-layout and, thus, understanding their effect on user task performance (and indeed user preference) can lead to readily achievable improvement to the layout of LineSets.

Size is recognised as a powerful variable to control when visualizing quantitative differences, yet the current implementation of the LineSets software does not exploit it. In LineSets, the sets have varying cardinalities and the nodes have varying degrees of connectivity. Thus, both of these graphical features could benefit from controlled manipulation of their sizes. Colour hue (different hues have different colours) is recognised as important when visualizing *qualitative* differences (Card, Mackinlay, and Shneider-

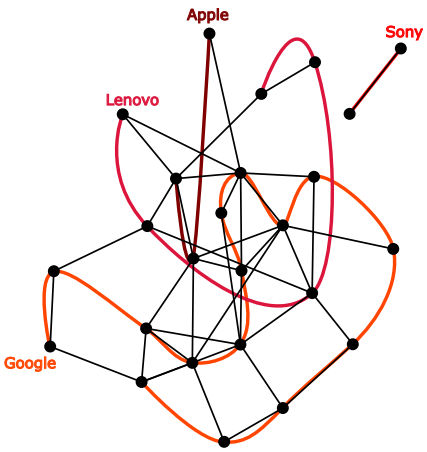

Figure 2.: Default LineSet.

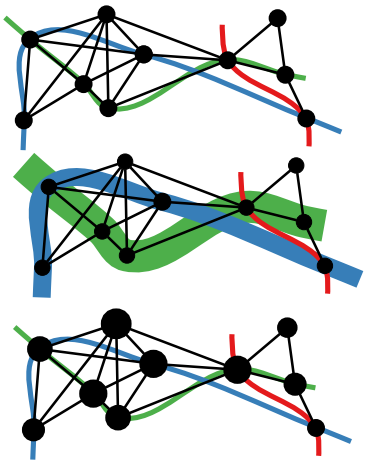

Figure 3.: LineSets: standard (left), varying line thickness (middle), varying node diameters (right).

man 1999; Leborg 2006; Mazza 2009), which is reinforced by the Gestalt principle of good form (Koffka 1935)[1]. In LineSets, if set-lines adopt different colour hues this is a visual indication that they represent different categories. In addition, if two nodes are passed through by a coloured line, this visually indicates the common property of both belonging to the respective set. By contrast, colour value (brightness) is seen as important when representing *quantitative* data; the way in which the LineSets software assigns colours to set-lines, by default, is by varying colour value but not in a way that reflects quantitative information (such as relative set cardinality).

Figure 2 shows a LineSet diagram as drawn by Alper et al. (Alper et al. 2011) where the nodes and set-lines are all of an equal thickness. But different graphical choices can be made which may aid (or hinder) user task performance. For example, we could instead use set-lines that are assigned uniformly distinct colour hues, see the top of Figure 3. This graphical choice was found to significantly improve user task performance, as compared to alternative colour treatments (e.g colour value, as in Figure 2) (Tranquille et al. 2016). Secondly, the thickness of the set-lines could be made relative to the number of elements within a set, in that if a set has more members it will be thicker than the other set-lines; see the middle of Figure 3. Therefore, the thickest set-line represents the largest set displayed in the diagram. This graphical choice, varying line thickness, also significantly improves user performance (Tranquille et al. 2017). Finally, the diameters of the nodes can also vary depending on the number of edges to which they are connected. The more edges to which a node is connected, the larger the diameter of the node; see the bottom of Figure 3. Again, this graphical choice was found to significantly improve user performance (Tranquille et al. 2017). However, this prior work evaluated the graphical choices of set-line colour, set-line thickness and node diameter individually. It is possible that the emphasis or variation of one graphical feature may act as a distractor for other features. For example, exploiting node size variations to reflect node degrees will mean that nodes of low degree are small. A consequence of varying node sizes is that small nodes are more likely to be obfuscated by set-lines, which is more likely when thick lines are present. In addition,

[1]The principle of good form indicates that people group graphical objects together if they have a common feature.

thick lines are more likely to pass through or very close to nodes than thinner lines, creating more potential for ambiguity to be present, particularly when nodes are large. It is not yet known whether applying all three treatments in combination, as seen in Figure 4, yields significant performance benefits over the existing graphical choices shown in figure 2.

Understanding whether the three treatments are effective in combination *and* for real-world data is important. In part, this is because LineSets are intended for visualising social networks in order to facilitate a user's ability to interpret and reason about datasets. The prior results, on the effects of manipulating colour, set-line thickness and node diameter, were collected using diagrams portraying synthetic data and manually produced layouts. These choices allowed the visual appearance and layouts of the diagrams to be carefully controlled to suit the hypotheses that were being tested. Because of this, the degrees of connectivity and set memberships were selected to ensure controlled variability to the LineSets used in the tests. By contrast, real-world data can vary in how many graphical elements are displayed.

As such, real-world data should be used to further evaluate LineSets because real-world applications would include varying amounts of data rather than strictly controlled amounts. It is certainly possible that combinations of graphical choices may be ineffective with real-world data if degrees of connectivity or set memberships are too great. For example, increasing sizes of some graphical elements could, say, occlude other elements. Furthermore, using manual layouts instead of automated layouts gave full control of the position of the nodes and the paths of the set-lines. This control is lost to a great extent when using automated layout tools. As such, it is important to evaluate an automated layout software environment as predetermined node positions and set-line paths may render certain graphical choices ineffective.

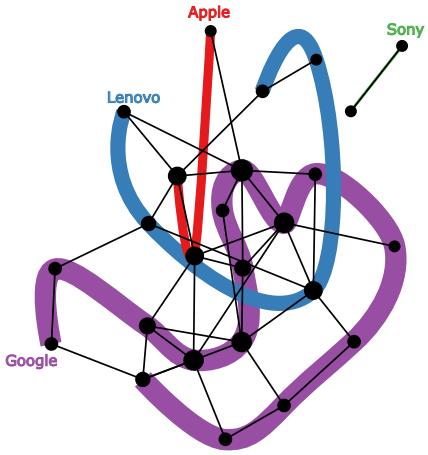

Figure 4.: Treated LineSet.

Consequently, the results in (Tranquille et al. 2016, 2017) should be built upon using real-world data and automated layouts as well as combining the use of size and colour. This would allow us to make inferences about the effects of manipulating graphical properties in LineSets and establish if certain choices are effective in automated layouts. This leads to our overarching question, the novelty and significance of which is explained above: *do graphical manipulations affect the comprehension of automatically laid out LineSets?* By answering this question, we can understand whether different graphical choices to those made by the current LineSets generation software, improve or degrade user performance when they are applied to automatically produced layouts.

To help to answer the overarching question, we specifically consider the following research questions in this paper:

(1) Do the colour, set-line thickness and node diameter treatments combine effectively to support task performance?
(2) Do the three treatments aid task performance when used with real-world data?
(3) Do the three treatments aid task performance when applied to automatically generated LineSets?

This paper sets out to address these three questions. Based on the previous findings just discussed, we establish two treatments to evaluate:

(1) *Default*: set-lines and nodes are all of an equal thickness and diameter, set-line colours are determined by the original LineSet software (Alper et al. 2011) (shown in Figure 2).
(2) *Treated*: the thicknesses of the set-lines vary according to the cardinality of the represented sets, the diameter of the nodes vary according to the number of edges connected to them, and each set-line is drawn in a unique colour hue (shown in Figure 4).

This paper establishes whether there are significant user performance differences between *Default* and *Treated* diagrams when the layout of the graph and set-lines are determined by automated layout software using real world data. If graphically treating the set-line colours, set-line thicknesses and node diameters in combination has a significant effect, we can expect higher rates of accuracy or lower mean task completion times from users interpreting *Treated* diagrams. The work in this paper, therefore, significantly advances the prior work just described by evaluating the three graphical choices in *combination*, using *real-world data* and *automatically laid out* LineSets.

3. Methodology

A between-group experiment design was adopted for performance data collection with one group for each diagram treatment, *Default* and *Treated*. The Default group performed tasks using automatically drawn LineSets whereas the Treated group used the same LineSets but with different set-lines colours, set-line thicknesses and node diameters. We recorded two dependent variables: the time taken to answer each question and whether each question had been answered correctly. Preference data was also collected to build an understanding of the perceived differences between the two treatments with respect to the types of questions asked.

3.1. *Tasks for Gathering Performance Data and Hypotheses*

In common with our earlier studies of LineSets, the tasks performed by participants are based on Sahket, Simonetto and Kouborov's group-level graph visualization taxonomy (Sahket et al. (2014)). The taxonomy consists of *group only* tasks, *group-node* tasks, *group-link* tasks and *group-network* tasks. Because the experiments were intended to test the effects of graphical alterations, the actual tasks selected reflected the hypotheses that were being tested and the expected advantages of the graphical properties under manipulation. These properties include selecting set-line colours to

improve set distinguishability and visibility, set-line thicknesses to reflect set cardinality, and node diameters to reflect nodes' degrees of connectivity. With these considerations, tasks included questions requiring users to

- identify sets or their elements based on intersections, potentially aided by colour,
- identify sets based on their size, potentially aided by set-line thickness, and
- identify nodes based on their degrees of connectivity, potentially aided by node diameter.

Thus, each task type, described in the following subsections, required attention to be placed on different components of LineSets to elicit the answer. To this end, we had three types of task: Set Intersection (SIn), Extreme Set Size (ESS) and Extreme Node Degree (END), each of which had two variants. These will be explained in sections 3.1.1 to 3.1.3.

In order to pose questions, we needed a context for the data being represented. The LineSets used in this study illustrated social networks with nodes representing people, edges representing friendship links between people, and sets representing brands of products that people have bought. Set names, which label the set-lines, were derived from a selection of brands. Each question was multiple choice, with participants asked to select the one correct answer from four possible answers. We note that for END tasks only, there could be multiple correct answers to the question, but only one of these was included in the four options. This aspect was made clear to participants in the training phase of the experiment. The feature arose because END tasks required the identification of a node with highest or lowest degree and it was not realistic to have a single such node. In what follows, the figures used to illustrate the example questions were all drawn using automated layout software, and some of them were manually altered to reflect the graphical properties we are testing. In all cases, the set names were manually added, not automatically positioned.

3.1.1. Set Intersection Tasks

The empirical study presented in (Tranquille et al. 2016) found that some tasks were performed significantly better when different colour hues were applied to the set-lines, rather than different colour values. Such tasks included computing information about the cardinality of intersecting sets, as such we call them set intersection (SIn) tasks. The selected SIn tasks are based on those which yielded significant performance differences in (Tranquille et al. 2016):

- *Count the total number of items that belong to both set x and set y.* We call this task type *SIn Elements*.
- *Count the number of sets that intersect with a given set.* We call this task type *SIn Sets* (reflecting the wording used in the task, see below).

The questions were worded to the participants in the following way:

- *SIn Elements:* How many people who bought Lenovo also bought Dell?
- *SIn Sets:* How many other brands were bought by people who have Lenovo?

With reference to Figure 5, an example *SIn Elements* task is 'How many people who bought Sony also bought Dell?'; the answer is 1.

So, tasks in this category required users to count nodes belonging to multiple sets. SIn Sets tasks required users to count how many sets intersected with a specified set. Since both tasks required the identification of multiple sets, performance could be as-

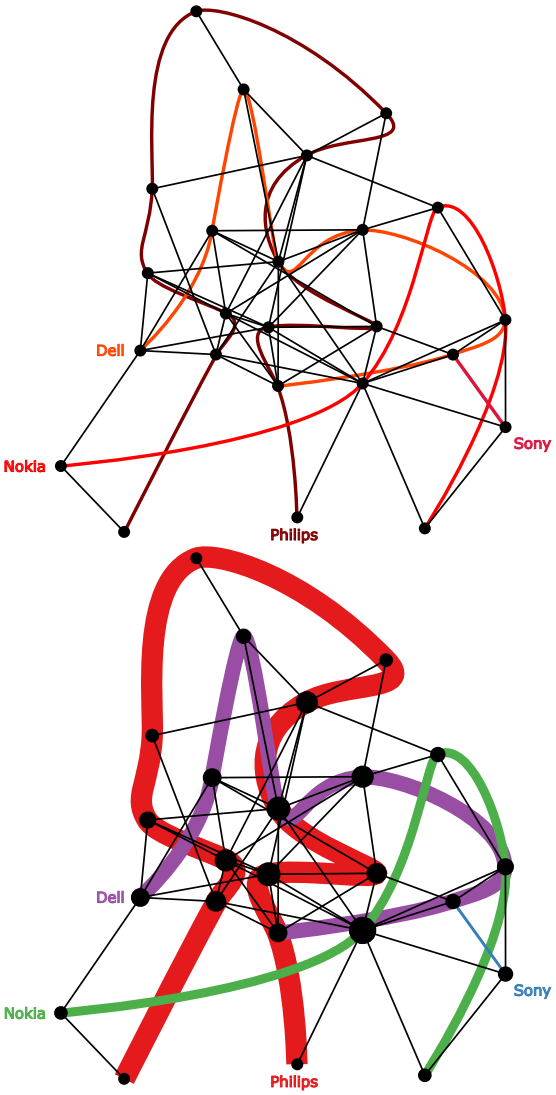

Figure 5.: Example Set Intersection Task

sisted by the distinguishability and visibility of the set-lines. Bertin's work on retinal variables suggests that assigning uniquely distinguishable colour hues, as used in the Treated group, can improve users' ability to isolate individual entities (Bertin 1983). Specifically for LineSets, this is supported by Tranquille et al. (Tranquille et al. 2016) who suggested that set-lines treated with uniquely distinguishable colour hues significantly improve user performance. We hypothesize that SIn tasks will be performed significantly better (i.e. either more accurately or faster) using Treated LineSets than Default LineSets.

3.1.2. Extreme Set Size Tasks

The empirical study presented in (Tranquille et al. 2017) found that some tasks were performed significantly better when different thicknesses were applied to the set-lines, rather than the same line thickness for all the set-lines. Importantly, the variations in set-line thicknesses reflected the different set cardinalities: one line is thicker than

another if and only if the set represented by the thicker line contains more elements than the other set. The associated tasks required identifying either the largest or the smallest set, so we call them extreme set size (ESS) tasks. Again, we selected ESS tasks based on those which yielded significant performance differences in (Tranquille et al. 2017):

- *Identify the largest set.* We call this task type *ESS Max.*
- *Identify the smallest set.* We call this task type *ESS Min.*

The questions were worded to the participants in the following way:

- *ESS Max:* Which product was bought the most times?
- *ESS Min:* Which product was bought the fewest times?

With reference to Figure 6, an example *ESS Min* task is 'Which product was bought the fewest times?'; the answer is Philips.

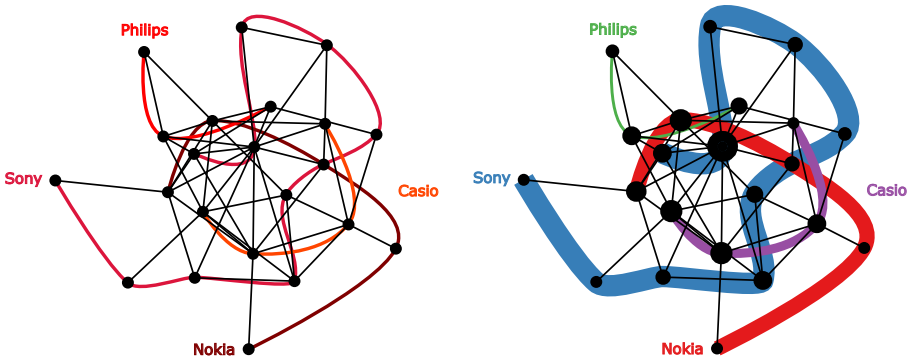

Figure 6.: Example Extreme Set Size Task

Since ESS tasks require users to identify the largest or smallest set, the thickness of the set-line could improve task performance. We appeal to work by Healey which suggests that people can identify size variations when exposed to an array of items of different sizes on a display (Healey and Enns 2012). Healey specifically refers to an experiment asking users to estimate which group of circles, each with their own unique colour, has the larger average size. Although Healey's work was in a different context, it indicates that the varying the size of graphical objects is preattentively processed. This suggests that, in the case of LineSets, varying size may assist people with identifying the largest (or smallest) set. Results by Tranquille et al. (Tranquille et al. 2017) support this insight, also suggesting that varying the thickness of set-lines significantly improves user performance, specifically for tasks where users have to identify a set of an extreme size. Therefore, we hypothesize that ESS tasks will be performed significantly better using Treated LineSets than with Default LineSets.

3.1.3. Extreme Node Degree Tasks

The empirical study presented in (Tranquille et al. 2017) found that some tasks were performed significantly better when varying node diameters were used in the graph underling the set-lines. Similarly to set-line thickness, the variations in node diameter reflected the different degrees of connectivity: one node diameter is larger than another if and only if the data item represented by the larger node is connected to more data items than the smaller node. This time, the associated tasks included finding a node

with largest or smallest degree and counting its incident edges, so we call them extreme node size (END) tasks. The selected END tasks are based on those which yielded significant performance differences in (Tranquille et al. 2017):

- *Identify a node with the highest degree of connectivity and count its incident edges.* We call this task type *END Max.*
- *Identify a node with the lowest degree of connectivity and count its incident edges.* We call this task type *END Min.*

The questions were worded to the participants in the following way:

- *END Max:* How many friends does the person with the most friends have?
- *END Min:* How many friends does the person with the fewest friends have?

With reference to Figure 7, an example *END Max* task is ‘ How many friends does the person with the most friends have?’; the answer is 10.

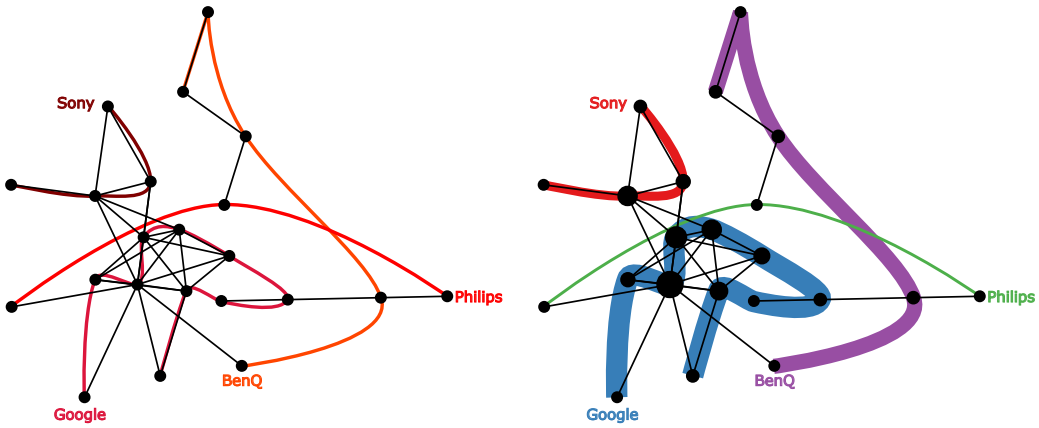

Figure 7.: Example Extreme Node Degree Task

Users are required to identify a node with highest or lowest degree of connectivity and to count the number of incident edges. Participants in the Treated group had the option of using the node diameters to identify an appropriate node before counting the edges. Healey’s work, on preattentive processing of size variations, supports the hypothesis that varying node sizes could be beneficial for task performance. This is further supported by Ariely’s research, on representing sets with statistical properties, which suggests proportional node diameters may allow users to preattentively identify extreme values (Ariely 2001). Specifically for LineSets, the results in Tranquille et al. suggested that varying the diameter of the nodes significantly improved user performance for tasks where users had to find a node of an extreme size (Tranquille et al. 2017). On this basis, we hypothesize that END tasks will performed significantly better using Treated LineSets than Default LineSets.

3.1.4. Number of Tasks and their Complexity

Considering the potential generalizability of our results, we opted to include LineSets with different characteristics whilst being careful to control the associated variability. This was achieved by introducing two types of complexity through the data being visualized by the diagrams. The task complexity was determined by the number of data items (nodes), connections (edges), and sets (set-lines). To this end, diagrams conformed of two characteristic types:

(1) Type-I diagrams: these diagrams represented 4 sets and included between 14 and 26 nodes and between 30 and 60 edges.
(2) Type-II diagrams: these diagrams represented 8 sets and included between 34 and 46 nodes and between 70 and 100 edges.

Figures 8 and 9 show examples of type-I diagrams and figures 10 and 11 show type-II diagrams. The study included 12 Type-I diagrams and 12 Type-II diagrams, giving 24 questions in total. The 12 Type-I diagrams were distributed across to the three task types (two diagrams for each task variant), as were the 12 Type-II diagrams.

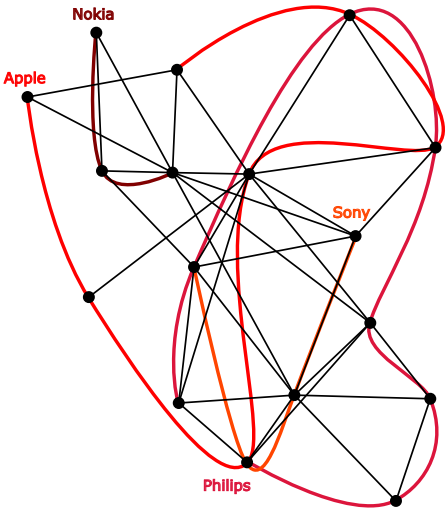

Figure 8.: A Type-I Default LineSet.

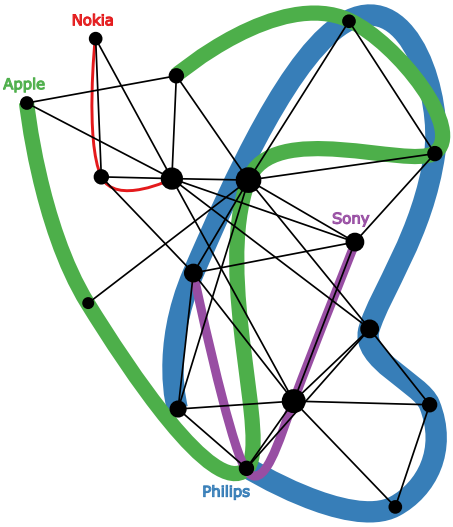

Figure 9.: A Type-I Treated LineSet

3.1.5. Preference Data Collection

For this stage of the study, a Type-II diagram was presented to participants in both Default and Treated states, as seen in figures 12 and 13 (scaled to 40% of their actual size). Participants were presented with the following seven questions:

(1) Which diagram do you prefer in terms of your aesthetic preference?
(2) Which product was bought the most times?
(3) Which product was bought the fewest times?
(4) How many friends does the person with the most friends have?
(5) How many friends does the person with the fewest friends have?
(6) How many people who bought Nokia also bought Sony?
(7) How many other brands were bought by people who have Apple?

Participants were asked to answer question 1 and, for the remaining six questions, they were asked to indicate which of the diagrams they preferred for answering it and to give a justification for their choice. The expectation was that this would allow us gain further insight into significant performance differences, or lack thereof. Participants could jointly rank the two diagrams if they so wished, when they had no preference between them.

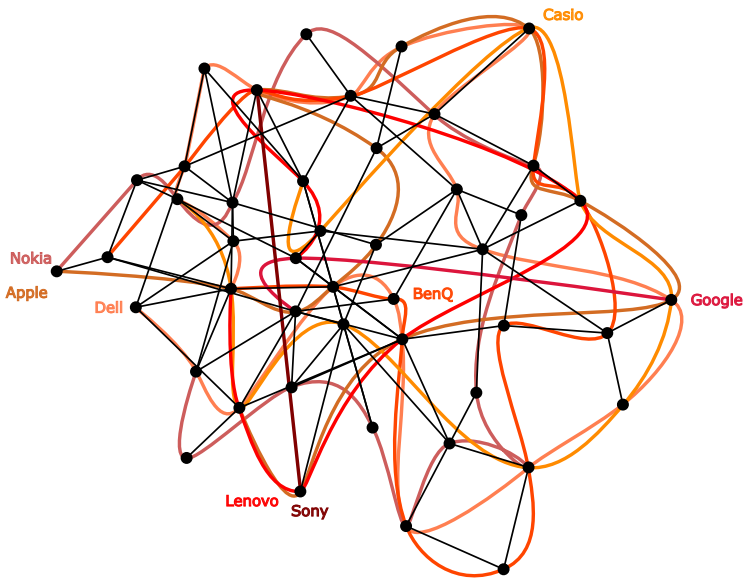

Figure 10.: A Type-II Default LineSet.

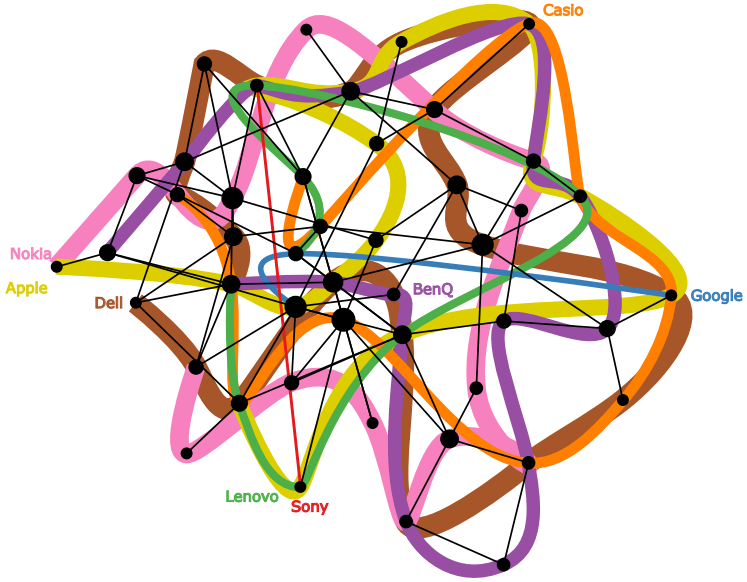

Figure 11.: A Type-II Treated LineSet

3.1.6. Summary

To reiterate, our study used three task types – SIn, ESS, and END – each with two variations. By appealing to prior work, as well as insight into preattentive properties of graphical entities, we hypothesize that all three task types will be preformed significantly more accurately or, if no significant difference in accuracy exists, significantly more quickly by participants using Treated LineSets as compared to Default LineSets. In addition, we also hypothesize that overall, irrespective of task type, Treated LineSets will support significantly better task performance. To aid generalizability, two levels of complexity were used for the tasks, reflecting the number of sets, data items and connections between the data items. Preference data was also collected and the design of this part of the study was geared towards revealing insights about any

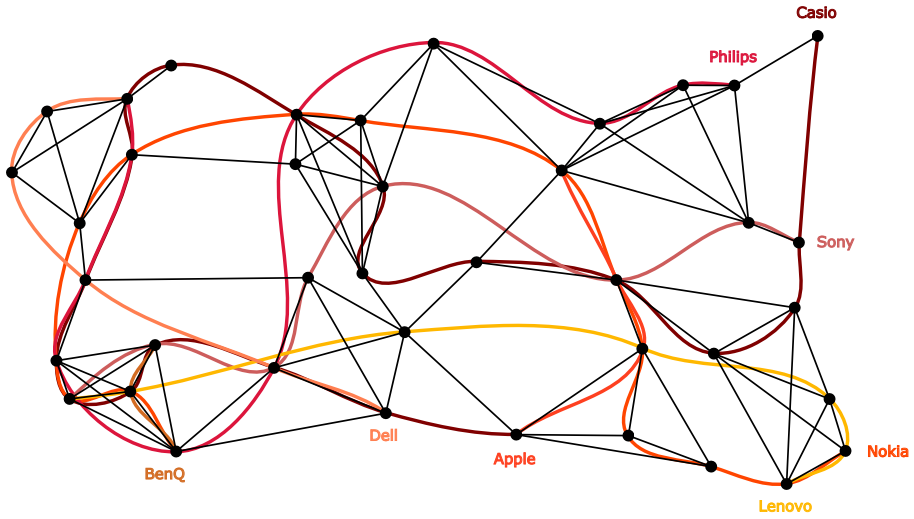

Figure 12.: Default diagram used to collect preference data

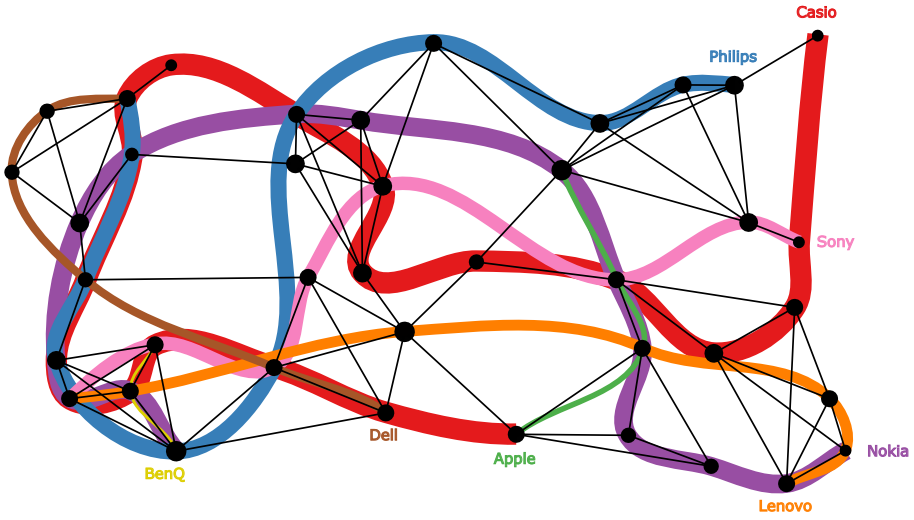

Figure 13.: Treated diagram used to collect preference data

significant performance differences.

3.2. *SNAP Data Sets for Visualization*

The information presented in the LineSets used in this experiment was derived from the SNAP Data Twitter social circles (Leskovec and Krevl 2014); SNAP contains over 970 data sets. Some networks contained over 68000 nodes and 1680000 edges, as well as many sets, which makes them too complex to visualize in a controlled empirical

study. It was necessary to filter the data in order to create data sets that gave rise to LineSets that were either Type-I or Type-II. While this filtering process reduces our data set to a fragment of the original social network, the remaining data points and sets are real world data. As we explain below, we choose a number of strategies for reducing the data to manageable size and for facilitating our tasks, such as the removal of sets with the same number of data items. In real-life, there is a high possibility that a data set will contain two sets with the same cardinality but we need to ensure that tasks such as those that involve identifying minimal and maximal sets have an unique answer. In a real world scenario, researchers may pre-process data in similar ways before investigating it. Our goal, therefore, was to simplify the SNAP data sets whilst limiting the loss of complexity and authenticity. As such, we removed sets, nodes, and edges from SNAP data sets to ensure that the resulting simplified data sets had the following characteristics:

- *No two sets contained exactly the same data items.* Such data sets would give rise to LineSets with concurrent set-lines and, thus, at least one set-line would be obfuscated. When such sets were identified, all but one of the identical sets were removed, thus avoiding complete concurrency between set-lines.
- *No two sets contained exactly the same number of data items.* Obviously this is a stronger condition than that just given and it allowed us to ensure that each set-line had a different thickness to the other set-lines in the Treated LineSets. As with identical sets, we removed all but one of the sets with an identical number of items.
- *Every node was in at least one set.* This ensured that every node was passed through by at least one set-line. Thus, removing such nodes did not alter the complexity of the visualized sets, but did reduce the complexity of the network.
- *No node had zero degree.* This ensured that every node was connected to at least one other node in the network; removing nodes with no incident edges did not substantially alter the complexity of the network.
- *There were no duplicate edges.* This ensured that any pair of nodes were connected by at most one edge. Duplicate edges, if visibly shown in graph underling the set-lines, would increase the complexity of the diagrams and potentially clutter the user's view. However, as is typical, we used a straight line embedding for the edges, so duplicate edges would not be visible anyway. The inclusion of duplicate edges would therefore render the visualization inaccurate, compared to the data being visualized, justifying their removal.

After this process, there were insufficient data sets that conformed to type-I and type-II – we wanted 12 of each. Many of the data sets were still too complex, so we randomly removed sets, node and edges, whilst ensuring that the above properties were maintained (if necessary, removing further items) to simplify the data further. At any stage, data sets that could never conform to type-I or type-II diagrams were discarded, for instance if they had too few edges. The final 24 data sets were selected randomly from the resulting simplified data sets, 12 each of type-I and type-II.

3.3. *Creating Diagrams for the Study: Conventions and Characteristics*

It was important that the diagrams across the treatment groups only differed by the independent variables that were being tested. To this end, we adhered to a series of drawing conventions and characteristics would prevent unwanted variations between groups. In particular, diagrams for each group were semantically identical and varied

syntactically only by the graphical features being manipulated for the study.

3.3.1. Drawing Default LineSets

Initially, we created 24 Default LineSets from the 24 simplified SNAP data sets using the LineSets software (Alper et al. 2011). We chose to use the default settings as these have been provided by the software designers, informed by usability studies. This software overlays lines on an already drawn graph and, so, the networks had to be drawn first. Edge lists, derived from real data following the process described in section 3.2, were imported into GraphViz (Gansner and North 2000) in order to create a node graph in SVG format along with a text file containing coordinates of each node in each diagram. The coordinates of each node were then exported to an XML file format which could be read by the automated LineSet layout software (Alper et al. 2011) specifying where to plot the points along each set-line. Set memberships for each node were specified in the XML file so that the set-lines could be generated. Where a node belonged to two or more sets, an instance of the node had to be created for each set to which it belonged so that the lines crossed at that point. As the LineSet software does not facilitate file exports, screenshots of the generated set-lines were imported into Inkscape (Gould et al. 2003). This allowed the set-lines to be manually traced over using the Bezier curves tool, which was a necessary step in order to generate the Treated LineSets and high-resolution images. The set-lines, with 4 pixels thickness, were then overlaid on top of the graph generated in step 1; we emphasize that this process preserved the paths followed by the set-lines as produced by the LineSets software. In each diagram, the LineSets software allocated each set-line a colour, which was also preserved when we re-drew the set-lines. The colour palletes used by the two types of Default diagrams can be seen in Figures 14 and 15. In all cases, the diagrams were drawn in an area no larger than 1280×720 pixels.

Figure 14.: Palette for *Default* Type-I diagrams.

Figure 15.: Palette for *Default* Type-II diagrams.

Figure 16.: Palette for *Treated* Type-I diagrams.

Figure 17.: Palette for *Treated* Type-II diagrams.

For the 24 Default diagrams, small adjustments were made to the graphs produced by GraphVis at this stage. In particular, the nodes were too small and the edges too thin to be readily visualized. So the Default group used nodes with a diameter of 16 pixels. The graph edges were drawn with 2 pixel thickness. Set-labels (i.e. names) were manually added adhering to the following characteristics:

(1) set-labels were written in sans-serif font at 14 pixels size with a 1 pixel stroke,
(2) set-labels were written in lower-case with an initial capital letter,
(3) set-labels were written at one end of the corresponding line whilst being spread out as much as possible and not obscuring any surrounding elements. Where this was not possible, the set-label was placed as close to the end of the corresponding

line as was practically possible, and
(4) the set-labels were assigned the same colour as the corresponding set-line.

3.3.2. Drawing Treated LineSets

To create Treated diagrams from the Default diagrams, we had to alter the set-line colours, the set-line thickness and the node diameters. Set-line colours for the Treated diagrams were derived from ColorBrewer (Brewer et al. 2013) as these colours were found to be an effective colour scheme (Tranquille et al. 2016). The colours used for each diagram type are shown in Figures 16 and 17. The colours were randomly assigned to the set-lines, thus allocating a colour to each set.

The set-line thicknesses were determined following a previously used specification (Tranquille et al. 2017), as illustrated in Table 1 (here with the set-line drawn in red). The thinnest set-line was 3.43 pixels and the thickest was 25 pixels, giving an effective difference of thickness ratio of approximately 1:7.2 between adjacent-sized sets in Type-I diagrams and of 1:3.08 between adjacent-sized sets in Type-II diagrams. Node diameters were also determined following a previously used specification (Tranquille et al. 2017), as shown in Figure 18. The smallest node size was 12 pixels and nodes increased in diameter by 2 pixels for each edge that was connected.

Type-I	Type-II

Table 1.: The thickness of each set-line from thinnest to thickest for both Type-I and Type-II diagrams.

Figure 18.: Nodes sorted by size.

3.4. *Collecting Performance and Preference Data*

A software tool developed for the purpose of conducting empirical studies (Blake et al. 2012, 2014a,b) was used to collect performance data, specifically time and accuracy data. Initially, the software collected demographic information about the participants, as well as an experimental reference which recorded the group to which the participant belonged, and a participant reference which allowed us to mark the data collected with the respective participant. Demographic data included the participant's gender, age, and whether they have any sight-based disabilities.

The software's design allowed us to display diagrams and their corresponding questions to participants whilst also capturing the answers submitted for each question and the time taken to complete each question. The time taken to complete a question was determined from the instant a question was presented until the instant a

participant had submitted the answer by clicking 'Submit'. Figure 19 shows how the diagram, question and available answers were presented on screen. A pause screen was presented to the participant prior to starting the next question . The 'continue' button allowed them to choose when they wanted to proceed to the next question. This ensured that they were ready to start the next question when it was displayed. Each question was subject to a two minute time limit in order to ensure that the experiment sessions finished within a reasonable time. If the time limit was reached, no answer was submitted and the software moved to the pause screen, allowing them to control when they started the next question. For each participant, the software randomised the order in which the questions were presented, to reduce the impact of any learning effects as compared to using a fixed predetermined question order.

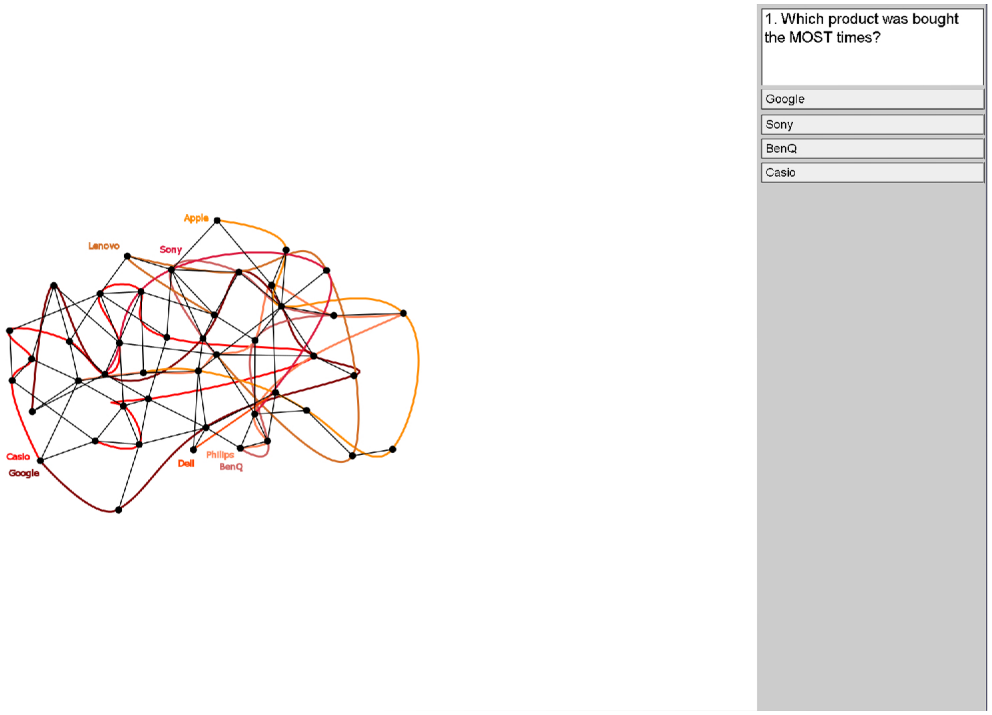

Figure 19.: Example of how questions were presented to users.

4. Study Phases and Execution

This section outlines the various phases we undertook to collect and analyse the data we captured.

4.1. *Training Phases*

In order to collect performance data, the participants first had to undertake two phases of training, one focusing on how to interpret LineSets and the other on how to use the data collection software. These two phases were the first of a four phase experiment.

The first phase of training introduced the participants to the concept of LineSets, the treatment that they would be exposed to, and the types of questions that they would be answering in the experiment. Hard-print copies of twelve example diagrams,

distinct from those used for performance and preference data collection, were shown to the participant, two for each task type and its variants. For each task type variant, the facilitator first explained how to answer the question with one of the diagrams. The participant was then shown the other diagram and asked to answer a question by themselves. They were given the time to study the diagrams carefully and thoroughly as well as being given the opportunity to ask questions if any aspect of the diagram or question was unclear. If they answered a question incorrectly, the facilitator would explain to the participant where they went wrong. If the facilitator was satisfied that the participant understood how to correctly derive the answer, the participant could proceed.

The second training phase introduced the data collection software in order to allow participants to familiarise themselves with the interface. A total of six questions, one of each task type variant, was asked. No time limit was imposed on the questions in this phase. Upon completion of the final training question, participants could proceed to the first of two data collection phases if they wished to do so.

4.2. *Data Collection Phases*

Performance data was collected in the third phase of the experiment process. Answers for each of the 24 questions were recorded by the data collection software as well as the time taken to complete the task. In this phase, the software incorporated a two minute time limit for each question to ensure that the study did not proceed indefinitely. Upon the completion of the final question, participants were asked to rate how difficult they found the questions. A series of seven intervals, ranging from very difficult to very easy, was used to indicate the participants' answers.

Preference data was collected in the fourth and final phase of the experiment process. Participants were presented with printed copies of LineSets drawn to reflect the treatments of the experiment and asked to rank them against the series of questions, as given in section 3.1.5. Participants could jointly rank the two diagrams if they so wished. The diagrams were affixed at eye level to a wall directly in front of the participant and their answers were recorded. The end of the experiment session was marked when the participant answered the last question.

4.3. *Study Execution*

We conducted a pre-pilot study before the pilot study to ensure that the design was not too difficult or time consuming to complete. The pre-pilot involved members of the **omitted for anonymity in the review process** who are expert users. When satisfied with our initial design, we then carried out a pilot test with participants who were representative of those we would recruit for the main study. Six participants were initially recruited for this phase of the process. Data collected from the pilot study found unexpected results for three questions. A low accuracy rate for question 17 exposed a mistake in the way this question had been encoded in the research software. This was subsequently revised. Although question 18, 'How many friends do the people with the most friends have?', did not accrue a low rate of accuracy, the diagrams did however display the node in question with the wrong size applied to it. The node was consequently changed to reflect the correct size for the number of edges connected to it. Question 24, 'How many people who bough Nokia also bought Apple?', displayed diagrams that showed two sets with an Apple label. This was rectified to only show

the correct set as Apple and the other was renamed. When we were satisfied with our experimental design, we proceeded to recruit 60 participants for the main study.

All participants were provided with a consent form and debrief sheet which provided them with all of the necessary details about the session in which they took part and how they would be able to access the results when they were made available. To ensure all participants were equally treated during the sessions, a script covering the introduction, training, data collection and debrief phases was generated and followed throughout each session. The material helped ensure that participants were informed consistently of the syntax and semantics of LineSet diagrams, the purpose of the study, their role during the study and their responsibilities after the experiment.

All participants were randomly allocated to one of the treatment groups; these groups were of equal size. All participants were students from the **omitted for anonymity in the peer review process** consisting of both undergraduates and postgraduates. The experiment took place within a usability laboratory between the hours of 9am and 6pm on weekdays. Participants were free from noise and distraction in this environment. The same equipment and room layout was used throughout the study in order to ensure that sessions were carried out under the same conditions. We limited each session to one participant at a time in order to avoid possible distractions, the only other person present was the study facilitator. The sessions lasted approximately 30 to 60 minutes and all participants were compensated with a £6 canteen voucher. Participants were also asked not to discuss the details of the study with those yet to participate.

5. Results: Performance Data

In this section, the collected performance data and preferential data gathered during the study is analysed. Data were collected from 60 participants (48 M, 11 F, one participant chose not to disclose their gender). Their average age was 23 (age range 18 to 36) and no participants identified as suffering from colour blindness. Accuracy was considered to be more important than time in terms of a performance indicator. This is because the time to complete a task is redundant if the answer is ultimately wrong. Leading on from this, we only analyse time data for responses where a correct answer was provided. As is typical, differences were considered to be significant if $p \leqslant 0.05$.

Prior to the main analysis, a preliminary analysis was performed to identify any outliers in the data. Participant 23, who was in the Treated group, had a noticeably lower level of accuracy than the other participants with just 7 correct answers, whereas the lowest number of correct answers accrued by any other participant was 13. The accuracy rate of participant 23 was 25%, and we therefore posit that this person was merely guessing the answer to each question. We must be mindful of the impact of the outlier on the statistical analysis. Thus, after performing our statistical analysis, we will determine whether any result is altered by the removal of participant 23's data. If the original result changes (e.g. a significant result becomes not significant) then this will call into question the robustness of our original result.

5.1. *Overall Analysis*

A total of $1,054$ correct answers were accumulated across both treatment groups of which the Default treatment accrued 511 (71.0%) and Treated group accrued 543 (75.4%); there were 369 incorrect responses and a total of 17 timeouts (questions that

were not answered within the two minute limit) with the Default group accruing 15 of them, leaving 2 accrued by the Treated group. These accuracy rates indicate that there was not a ceiling or floor effect. This suggests that the tasks were not trivial (the accuracy rate is not close to 100%) and not overly hard (the accuracy rate was not close to 25% which would be expected if participants were guessing the answers). These data thus suggest that the tasks required some cognitive effort to perform, as is required for a study such as this. Conducting a Kruskal-Wallis test showed that there were no significant differences in overall accuracy ($p = 0.057$) between the Default and Treated groups. The significance of the overall accuracy result changed as a result of the outlier being removed ($p = 0.010$), which suggests that the Treated group may, in fact, have performed significantly more accurately.

Of the $1,054$ correct answers that were recorded, the overall mean task completion time was 24.66 seconds with a standard deviation of 18.11. The overall mean times for the Default group and Treated group were 31.20 seconds (s.d. 20.88) and 18.50 seconds (s.d. 12.19) respectively. The time data had to be checked for normality prior to any ANOVA analysis. As expected with time data, they were significantly different from normal. A base 10 log transformation was subsequently applied, resulting in data that were not significantly different from normal ($p = 0.202$; skewness 0.11), which rendered the data suitable for analysis. In order to determine whether significant differences in task completion time existed between the two groups, we proceeded to perform an ANOVA test. The results gave $p = 0.000$, indicating that a significant difference existed between the two treatment group's mean times. The significance of this result did not change when the outlier was removed ($p = 0.000$), so the result can be considered robust.

In summary, the time analysis supports our hypothesis that Treated LineSets support significantly better task performance than Default LineSets: users perform tasks significantly more quickly using Treated LineSets, irrespective of the task type; see Table 2. A notable effect size is seen with the time data: participants in the Default group took, on average, 12.7 seconds longer to answer the questions correctly in comparison to the Treated group, an increase of approximately 69%. Regarding overall accuracy, if we accept that the outlier impacted robustness and, in fact, there is a significant difference then we would expect at least four more correct answers per 100 questions from participants using Treated LineSets than when using Default LineSets. In conclusion, this study suggests that LineSets facilitate significantly improved task performance when treated with uniformly distinct set-line colours, set-lines of varying thickness and nodes of varying diameter.

5.2. *Analysis by Task Type Variants*

The analysis conducted for each task type follows the same structure as the overall analysis and is summarised in Table 2. We found that four out of our six task-level hypotheses are supported: for SIn Elements, ESS Max, ESS Min, and END Max. In these four cases, the effect sizes for time were substantial with increases ranging from 44% to 193%. For SIn Sets there was no significant difference in either accuracy or time performance. An interesting feature of our data is that the SIn Sets tasks had a low accuracy rate of 45.8% for both groups. This unusual observation will therefore be further discussed in section 7 to understand why this phenomena manifested. For END Min tasks, Default LineSets were, unexpectedly, significantly more accurate.

Table 2.: Summary of the task-level accuracy (A) and time (T) results.

Question Type	With Outlier				No Outlier		Results with Outlier	
	Default	Treated	F-Statistic	p-value	F-Statistic	p-value	Best Performance	Effect Size
Overall (A)	$\frac{511}{720} = 71.0\%$	$\frac{543}{720} = 75.4\%$		0.057		0.010	–	–
Overall (T)	31.20 (20.88)	18.50 (12.19)	$F_{1,948} = 55.42$	0.000	$F_{1,942} = 54.74$	0.000	Treated	12.7 (69%)
SIn Elements (A)	$\frac{80}{120} = 66.7\%$	$\frac{89}{120} = 74.2\%$		0.204		0.160	–	–
SIn Elements (T)	34.71 (22.55)	24.03 (10.83)	$F_{1,103} = 25.91$	0.000	$F_{1,102} = 26.00)$	0.000	Treated	10.7 (44%)
SIn Sets (A)	$\frac{55}{120} = 45.8\%$	$\frac{55}{120} = 45.8\%$		1.000		0.808	–	–
SIn Sets (T)	30.80 (12.57)	30.72 (19.63)	$F_{1,50} = 0.02$	0.893	$F_{1,50} = 0.02$	0.893	–	–
ESS Max (A)	$\frac{101}{120} = 84.2\%$	$\frac{116}{120} = 96.8\%$		0.001		0.000	Treated	$\frac{13}{100}$
ESS Max (T)	38.88 (26.59)	13.29 (10.00)	$F_{1,151} = 209.33$	0.000	$F_{1,150} = 203.39$	0.000	Treated	25.6 (193%)
ESS Min (A)	$\frac{89}{120} = 74.2\%$	$\frac{100}{120} = 83.3\%$		0.083		0.033	–	–
ESS Min (T)	33.40 (23.74)	11.80 (6.56)	$F_{1,123} = 118.81$	0.000	$F_{1,123} = 118.03)$	0.000	Treated	21.6 (183%)
END Max (A)	$\frac{67}{120} = 55.8\%$	$\frac{73}{120} = 60.9\%$		0.433		0.403	–	–
END Max (T)	34.19 (16.82)	22.52 (10.73)	$F_{1,76} = 4.38$	0.039	$F_{1,77} = 4.72$	0.033	Treated	11.7 (52%)
END Min (A)	$\frac{119}{120} = 99.2\%$	$\frac{110}{120} = 91.7\%$		0.006		0.050	Default	$\frac{8}{100}$
END Min (T)	19.19 (9.01)	16.83 (6.99)	$F_{1,164} = 1.42$	0.238	$F_{1,164} = 1.42$	0.238	–	–

5.3. *Results By Complexity Level*

Table 3 further breaks down the analysis by the two task complexity levels. For low-level complexity, Type I, we can see that Treated performed significantly faster, but never more (or less) significantly accurately than Default, except for END Min tasks. In this case, Default was significantly more accurate than Treated. However, when the outlier is removed from the analysis, the significant difference is eliminated, with the p-value increasing to 0.076. This leads us to suggest that the result – of Default being more accurate than Treated for END Min tasks at the low complexity level – is not robust.

For the high complexity tasks, Treated was always significantly better (more accurate or faster) or not significantly worse (less accurate or slower) than Default. For ESS Min tasks we found no significant difference at the high complexity level, but this alters when the outlier is removed: the p-value reduces to 0.026. Therefore it is likely that, in fact, Treated is significantly more accurate than Default for these tasks.

5.4. *Summary of Results*

Of our seven initial hypotheses – that overall and for the six task type variants, Treated LineSets would support better performance than Default LineSets – five were supported by the analysis in table 2 with END Min being the exception. The result summarised here:

Overall, participants performed significantly significantly faster using LineSets with set-lines of varying colours and thicknesses, and nodes with varying diameter. For high complexity tasks, there was also a significant accuracy benefit to using Treated LineSets. These results suggests that manipulating the graphical properties of set-line colour, line thickness and node size is beneficial for general task performance. The effect size was large for time, with the Default group taking approximately 69% longer overall than the Treated group overall. The effect size was notably higher for high complexity tasks (82%) than for low complexity tasks (65%).

For tasks that required counting elements common to two sets (SIn Elements), participants performed significantly faster using LineSets with set-lines of varying colours

Table 3.: Summary of the task-level accuracy (A) and time (T) results per level of complexity.

Question Type	Type I: Low Complexity								Type II: High Complexity							
	With Outlier			No Outlier		Results with Outlier			With Outlier			No Outlier		Results with Outlier		
	Default	Treated	F-Statistic	p-value	F-Statistic	p-value	Best	Effect	Default	Treated	F-Statistic	p-value	F-Statistic	p-value	Best	Effect
Overall (A)	$\frac{322}{360} = 89.4\%$	$\frac{313}{360} = 86.9\%$		0.229		0.605	–	–	$\frac{189}{360} = 52.5\%$	$\frac{230}{360} = 63.9\%$		0.002		0.000	Treated	$\frac{11}{100}$
Overall (T)	24.92 (12.52)	15.14 (8.20)	$F_{1,553} = 53.31$	0.000	$F_{1,548} = 52.99$	0.000	Treated	9.8 (65%)	41.92 (27.06)	23.07 (14.96)	$F_{1,337} = 34.70$	0.000	$F_{1,337} = 33.30$	0.000	Treated	18.9 (82%)
SIn Elements (A)	$\frac{57}{60} = 95.0\ \%$	$\frac{53}{60} = 88.3\%$		0.188		0.170	–	–	$\frac{23}{60} = 38.3\%$	$\frac{36}{60} = 60.0\%$		0.018		0.010	Treated	$\frac{22}{100}$
SIn Elements (T)	27.79 (15.29)	19.15 (7.17)	$F_{1,49} = 13.64$	0.000	$F_{1,48} = 13.58$	0.000	Treated	8.6 (45%)	52.09 (28.06)	31.22 (11.39)	$F_{1,12} = 14.46$	0.000	$F_{1,12} = 14.46$	0.000	Treated	20.9 (67%)
SIn Sets (A)	$\frac{48}{60} = 80.0\%$	$\frac{39}{60} = 65.0\%$		0.067		0.117	–	–	$\frac{7}{60} = 11.7\%$	$\frac{16}{60} = 26.7\%$		0.038		0.030	Treated	$\frac{15}{100}$
SIn Sets (T)	29.82 (11.68)	21.38 (7.30)	$F_{1,31} = 8.01$	0.006	$F_{1,31} = 8.01$	0.006	Treated	8.4 (39%)	37.50 (17.13)	53.50 (21.75)	$F_{1,19} = 0.78$	0.387	$F_{1,19} = 0.78$	0.387	–	–
ESS Max (A)	$\frac{58}{60} = 96.7\%$	$\frac{60}{60} = 100.0\%$		0.156		0.163	–	–	$\frac{43}{60} = 71.7\%$	$\frac{56}{60} = 93.3\%$		0.002		0.000	Treated	
ESS Max (T)	22.96 (11.94)	8.47 (4.36)	$F_{1,56} = 109.50$	0.000	$F_{1,55} = 104.95$	0.000	Treated	14.5 (171%)	60.36 (25.83)	18.45 (11.68)	$F_{1,40} = 142.88$	0.000	$F_{1,40} = 142.88$	0.000	Treated	41.9 (227%)
ESS Min (A)	$\frac{54}{60} = 90.0\%$	$\frac{55}{60} = 91.7\%$		0.753		0.547	–	–	$\frac{35}{60} = 58.3\%$	$\frac{45}{60} = 75.0\%$		0.054		0.026	–	–
ESS Min (T)	20.65 (10.01)	8.39 (2.78)	$F_{1,47} = 87.13$	0.000	$F_{1,47} = 85.79$	0.000	Treated	12.2 (146%)	53.07 (25.42)	15.97 (7.42)	$F_{1,27} = 78.06$	0.000	$F_{1,27} = 78.06$	0.000	Treated	37.1 (232%)
END Max (A)	$\frac{45}{60} = 75.0\%$	$\frac{51}{60} = 85.0\%$		0.173		0.126	–	–	$\frac{22}{60} = 36.7\%$	$\frac{22}{60} = 36.7\%$		1.000		0.959	–	–
END Max (T)	30.82 (11.99)	19.18 (8.38)	$F_{1,37} = 22.6$	0.000	$F_{1,37} = 25.89$	0.000	Treated	11.6 (61%)	41.10 (22.67)	30.29 (11.69)	$F_{1,4} = 2.34$	0.154	$F_{1,4} = 2.62$	0.134	–	–
END Min (A)	$\frac{60}{60} = 100.0\%$	$\frac{55}{60} = 91.7\%$		0.023		0.076	Default	$\frac{8}{100}$	$\frac{59}{60} = 98.3\%$	$\frac{55}{60} = 91.7\%$		0.095		0.295	–	–
END Min (T)	19.65 (9.18)	17.13 (6.99)	$F_{1,54} = 0.94$	0.335	$F_{1,54} = 0.94$	0.335	–	–	18.72 (8.89)	16.53 (7.03)	$F_{1,54} = 1.38$	0.244	$F_{1,54} = 1.38$	0.244	–	–

and thicknesses and nodes with varying diameter. This time benefit was also seen at both complexity levels, where Treated outperformed Default. At high complexity, there was also a significant accuracy benefit to using Treated LineSets. With regard to the effect size for time, the Default group took approximately 44% longer overall than the Treated group. The effect size was notably higher for high complexity tasks (67%) than for low complexity tasks (45%).

For tasks that required counting the number of sets that shared elements with a given set (SIn Sets), no significant differences were observed overall. However, Treated Linesets gave rise to significant time benefits for low complexity tasks. At the high complexity level, a significant accuracy benefit was found when using Treated LineSets. This suggests that, whilst the graphical manipulations are, unexpectedly, not beneficial *overall* for these tasks, they do not hinder task performance when counting common elements in sets. In addition, there is evidence to support the hypothesis that Treated LineSets are more effective for SIn Sets tasks, due to the low complexity time result and the high level complexity accuracy result.

For tasks that required the identification of the largest set (ESS Max), participants performed significantly more accurately and significantly faster using LineSets with set-lines of varying colours and thicknesses and nodes with varying diameter. The time results are reinforced at both task complexity levels and the accuracy results are also supported at the high task complexity level. The overall accuracy effect size was 13 (i.e. 13 more correct answers would be expected from the Treated group per 100 questions). In the case of time, the overall effect size was very notable, with the Default group taking approximately 193% longer overall than the Treated group. Examining the two complexity levels, the time effect size is unsurprisingly lower, at 171%, for low complexity tasks and much higher at 227% for high complexity tasks.

For tasks that required the identification of the smallest set (ESS Min), participants performed significantly faster using LineSets with set-lines of varying colours and thicknesses and nodes with varying diameter. Time benefits are also seen at both task complexity levels. The overall effect size time was again notable, with the Default group taking approximately 183% longer overall (146% for low complexity and 232% for high complexity) than the Treated group.

For tasks that required counting number of edges connected to the node with highest degree (END Max), participants performed significantly faster using LineSets with set-lines of varying colours and thicknesses and nodes with varying diameter. They also performed significantly faster using Treated LineSets for the low complexity tasks but there was no significant difference for the high complexity tasks. With regard to the overall effect size for time, the Default group took approximately 52% longer than the Treated group overall (61% at the low complexity level).

For tasks that required counting number of edges connected to the node with smallest degree (END Min), an overall significant difference in accuracy only was observed. We found that Default LineSets were significantly more accurate both overall and for low complexity END Min tasks, with the latter becoming insignificant after removal of the outlier. This suggests that whilst the graphical manipulations are, unexpectedly, not beneficial for these tasks, there is evidence to suggest that they hinder task performance when counting edges connected to a node of low degree. The accuracy effect sizes both indicate, overall and at low complexity, that we would expect 8 more correct answers per 100 tasks using Default LineSets.

The strong take-away message is that Treated LineSets typically performed better

than Default LineSets. This indicates that varying colour hues, varying set-line thickness and varying node diameter can all be manipulated to LineSets' advantage when diagrams are generated automatically. Therefore our results lead to the recommendation that the LineSets software is extended to incorporate the use of varying colours, line thicknesses and node sizes.

6. Results: Preference Data

Recall that participants were asked which of the two diagrams in figures 12 and 13, reflecting the two treatments, was aesthetically preferable. They were also asked to indicate their preferred diagram in respect of six questions corresponding to the six different types of tasks they undertook in the performance phase of the study; details were given in section 3.1.5. The data were subsequently analysed by performing a series of Kruskal-Wallis test in order to rank the treatments for each of the seven preference tasks. For all seven tests the p-value was 0.000, with Treated being highest ranked (most preferred). These results show a clear significant preference for treated LineSets both overall, in terms of aesthetics, and as the preferred option for answering the six questions corresponding to the study tasks.

Of course, we must be mindful of the fact that we conducted a between group study, and participants had just been exposed to exactly one of the two treatments. With any between group study design, a preference phase such as this could be impacted by the treatment by the treatment to which the participant had been exposed. Given the overwhelming preference for the Treated diagrams, overall, it is not clear that there was any significant influence arising from treatment exposure in the performance phase. The bar chart in Figure 20 shows the breakdown of preference data across treatment groups. Bar charts for the remaining six preference questions also show the breakdown across groups.

The rankings were supported by written answers where participants explained their choices. Firstly we focus on comments concerning *overall aesthetic preference*. Here 53 participants (of these, 28 were in the Treated group) preferred the treated diagram (Figure 13), with the remaining 7 (of these, 5 were in the Default group) preferring the default diagram (Figure 12); the data are illustrated in Figure 20. One theme that emerged was that the treated diagram helped participants see the differences between the sets more easily because of the uniquely distinguishable colours and the varying sizes of the set-lines and nodes. This sentiment was expressed by about 45% of participants. Some participants (10%) commented only on the colours of the treated set-lines being helpful in distinguishing the sets. Other comments were less specific, with 15% of participants noting that the treated diagram was easier to interpret at a glance or clearer, with two participants stating that the treated diagram was easier to read.

In cases where participants preferred the default diagram, 5% commented that it was clearer because the varying thickness of the set-lines makes the treated diagram harder to interpret. One participant commented that the default diagram was not impeded by the varying set-line thickness seen in the treated diagram, which they felt made the set intersections difficult to interpret. Lastly, 5% of participants found the consistent use of size in the default diagram was visually neater.

These comments on the default diagram, combined with those for the treated diagram, may suggest that varying set-line thickness is not always seen as beneficial or aesthetically pleasing, by participants. We observe that, whilst participants were asked

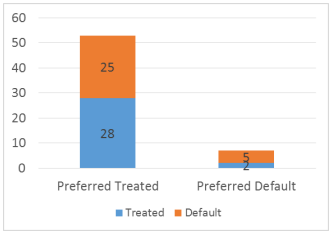

Figure 20.: Overall preferences.

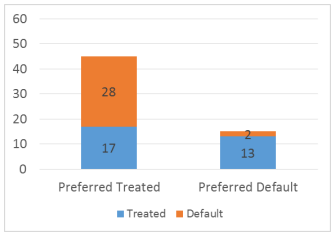

Figure 21.: SIn Elements preferences.

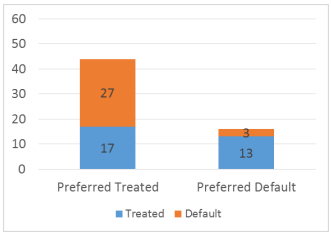

Figure 22.: SIn Sets preferences.

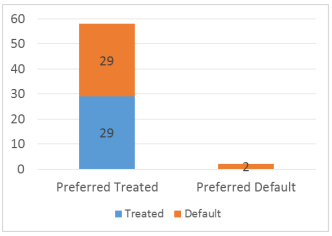

Figure 23.: ESS Max preferences.

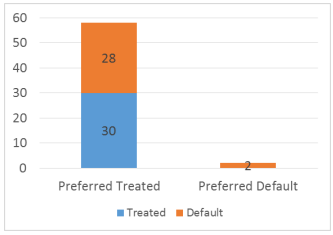

Figure 24.: ESS Min preferences.

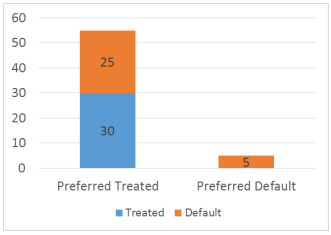

Figure 25.: END Max preferences.

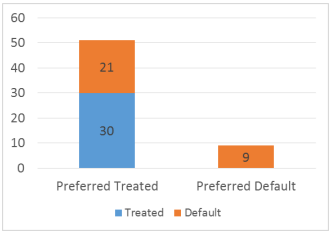

Figure 26.: END Min preferences.

for comments on aesthetic preference, many of the response themes above relate to using the diagrams for information extraction. We posit that this could be because they were asked about aesthetic preference after undertaking the performance phase of the study. Different preferences or comments might have been obtained if the participants were about aesthetic preference before collecting performance data.

The next preference task was related to an *SIn Elements question*, where 45 participants (17 in the Treated group) expressed a preference for using the treated diagram, with the remaining 15 participants (2 in the Default group) preferring the default diagram; the data are illustrated in Figure 21. Nearly half of the participants commented that the reason for preferring the treated diagram was because the colours helped to distinguish the sets more easily; interestingly, one participant preferred the default diagram for the same reason. Varying set-line thickness in the treated diagram was noted by 15% of participants to help them to differentiate the sets more easily. Contrastingly, 12% preferred the default diagram because they believed it was easier to see the set intersections with consistent set-line thicknesses and a further 3% felt that the consistent set-line thickness helped them to see differences between the sets.

For the *SIn Sets question*, 44 participants (17 in the Treated group) expressed a preference for using the treated diagram, with the remaining 16 participants (3 in the Default group) preferring the default diagram: the data are illustrated in Figure 22. The participants' reasons followed similar themes to those for SIn Elements. Com-

ments, by 53% of participants, stated a preference for the treated diagram due to its use of colour and set-line thickness, which they believed helped them to distinguish the sets more easily. Two participants stated that they found the treated colour scheme easier to follow. Participants also preferred the treated diagram because they said it was clearer (8% of participants) or because the varying thicknesses of the set-lines helped them to see the differences between the sets more easily (8% of participants). A further 15% found that the thickness of the set-lines grabbed their attention. Also relating to set-line thickness, 12% felt that thickness was an indication of the answer, therefore removing the need to count nodes along the set-line. Other participants found that it was easier to just see the answer rather than count the nodes, the sets were more distinguishable, there was no need to count the nodes, the colours made it easier to distinguish between the sets and it was easier to find the answer (1 participant each respectively).

The default diagram was preferred for a number of reasons: 11% of participants found that it was easier to see the intersections between the sets with the consistent set-line thickness; two participants believed that the default diagram was clearer and a further two thought that the consistent thickness showed the differences between the sets more easily. Interestingly, 5% of participants said that the default colour scheme made it easier to distinguish the sets. One participant said that the default diagram was clearer and another said that there was no benefit to using the treated diagrams for these questions as the set-line thickness and node diameters were not useful for finding the answer. One participant said that they felt the answer stood out more in the default diagram and another participant' gave this preference because they found counting [sets] was easier than estimating.

The next preference task was related to an *ESS Max question*, where 58 participants (29 in the Treated group) expressed a preference for using the treated diagram, with the remaining 2 participants (1 in the Default group) preferring the default diagram; the data are illustrated in Figure 23. There was a key theme in the comments associated with this task: 50% of participants found the colours and thicknesses used in the treated diagram helped them distinguish between the sets more easily. Focusing on the thickness of the set-lines, 15% commented that the thickness grabbed their attention and a further 12% found that thickness to give them an indication of the answer. A further 8% of participants found the treated diagram to be clearer. Two participants commented that the treated diagram was easier to interpret at a glance. Lastly, the following comments were written by one participant in each case: it was easier see and estimate the answer, it was simply easier to find the answer, the set-lines in the treated diagram were more distinguishable, the colours made it easier to distinguish between sets and, finally, that this treatment negated the need to count the nodes on the set-lines. Where participants preferred the default diagram, two reasons were included: the answer stood out to them more and they found it easier to count the nodes along the set-lines.

For the *ESS Min question*, 58 participants (30 in the Treated group) expressed a preference for using the treated diagram, with the remaining 2 participants (2 in the Default group) preferring the default diagram; the data are illustrated in Figure 24. 73% of participants preferred the colours and thicknesses of the treated diagrams because it helped them distinguish between the sets more easily. Only one participant found the treated colour scheme to be the only factor in them preferring treated diagrams for this question or found the treated diagrams to be easier to interpret at a glance. 13% of participants found the set-line thickness to be the most useful aspect of the treated diagram. A further 7% of participants simply found the data to be more

clear in the treated diagrams. Only two participants preferred the default diagram for the ESS Min task, saying that it was easier to count the points of intersection with the consistent thicknesses.

For the *END Max question*, 55 participants (30 in the Treated group) preferred using the treated diagram, with the remaining 5 participants (5 in the Default group) preferring the default diagram; the data are illustrated in Figure 25. Regarding preference for the treated diagram, 48% of participants commented that the diameters of the nodes grabbed their attention, so they did not need to spend time scanning the diagram for the answer. A further 32% preferred the treated diagram because they could more easily compare the differences in the nodes. One participant felt that they did not need to count the edges with the treated diagram, and another three found them easier to interpret at a glance which is why they preferred them for this reason. Focusing on preference for the default diagram, 5% of all participants said they preferred it because the felt it was easier to just count the edges or because the (non-varying thickness of) the set-lines did not obscure the nodes. One participant said that they preferred the default diagram because there was less interference with the other items in the diagram.

The final preference question concerned an *END Min question*. Here, 51 participants (30 in the Treated group) preferred using the treated diagram for the *END Max question*, with the remaining 9 participants (9 in the Default group) preferring the default diagram; the data are illustrated in Figure 26. It was observed, by 40% of participants, that the diameters of the nodes in treated diagrams grabbed their attention. Like END Max tasks, 28% of participants said the treated diagram allowed them to more easily compare the sizes of the nodes. Other participants preferred the treated diagram because it was clearer (3%), easier to interpret at a glance (8%), that they perceived the diagram to be bigger (3%), or found the colours useful to distinguish the sets (8%). Concerning preference for the default diagram, 3% of participants said it was because they found the presentation more clear, could see the differences between the sets more easily or because the set-lines did not obscure the nodes. Interestingly, one participant found it was easier to count the edges in the default diagram, and another found that there was not enough difference between the sizes of the nodes in the Treated diagram to allow them to effectively answer the question. A further participant preferred the default diagram because it was more aesthetically pleasing.

7. Discussion

Our understanding of graphical manipulations in LineSets led to the hypothesis that treating the set-lines with uniquely distinguishable colour hues, varying line thicknesses and nodes with varying diameters helps users interpret information more quickly and accurately. For most task types this was supported. When it was not, there was no significant difference between the two treatment groups. We start by recalling that the graphical choices were implemented because it was felt that they could facilitate different types of tasks. We will consider the three categories of task in turn.

7.1. *Set Intersection Tasks*

Set-line colours were chosen to help users differentiate between sets (SIn tasks). Both these tasks required nodes to be identified that were passed through by two (or more) lines. For SIn Elements tasks, users had to count nodes that were passed through by two

particular lines. For SIn Sets tasks, users had to count set-lines that shared a node with a specified set-line. For SIn Sets tasks, low levels of accuracy were seen in particular for two questions (numbers 17 and 22), where only 19 and 4 correct answers were collected respectively, each out of 60 responses. Qualitative data collected through the preference questionnaires suggested that participants sometimes found that the diagrams were difficult to interpret as set-lines brushed nodes that were not part of that set. This may therefore explain the very low levels of accuracy that were observed. Indeed, this issue is exacerbated in Treated LineSets as the nodes are larger, therefore sometimes making it difficult to determine whether a node belongs to a set. It is possible that the accuracy advantage of making the set-lines distinguishable through the use of colour is eliminated because of the increased difficultly of determining on which set-lines nodes lie when the set-lines have varying thicknesses (but we are mindful that for SIn Elements tasks, Treated LineSets still had a significant time advantage).

7.2. *Extreme Set Size Tasks*

Set-line thicknesses were chosen to help users compare set sizes (ESS tasks). When interpreting Treated LineSets, for ESS tasks, we have turned what would be *counting* tasks with the Default treatment into *target detection* tasks. These differ in that the former requires users to count individual elements to elicit the answer whereas the latter requires the user to identify a target element with a unique visual feature (Ariely 2001; Healey and Enns 2012); i.e., the user needs to identify the thickest or thinnest set-line. Treated diagrams were found to be significantly more accurate than Default for both ESS tasks. Preattentive properties are those visual stimuli that the eye and the brain process in less than 250 milliseconds, and size is one such property (Healey and Enns 2012): in our case, the visual targets can be found preattentively when their dimension is varied. Therefore, these results may have been observed because the differences between the set-line thicknesses were sufficiently salient that the user was able to make a more accurate estimate, more quickly, of size differences in a significantly quicker time instead of counting the elements along the set-lines. That is, by making it possible to find visual targets preattantively, we have increased accuracy and reduced time performance.

7.3. *Extreme Node Degree Tasks*

Node diameters were chosen to help users compare nodes' degrees of connectivity (END tasks). Just as with ESS tasks, END tasks can, in part, be considered target detection tasks in which the user needs to find the largest or smallest node. However, in this case even the Treated LineSets still required participants to count graphical elements – they had to establish how many edges were incident to the target node. Moreover, it could be considered that nodes with extreme degree still had some degree of saliency, and could possibly be idenfitied pre-attentively as well; at the very least, it would be clear that many nodes could be discounted as not necessary for completing the task due to their non-extreme degree. Thus, in both cases, we could consider the tasks to require both pre-attentive processing and counting in order to provide a solution. The results for these tasks were varied. We found no overall significant difference in accuracy between treatments for END Max tasks, but users performed significantly faster with Treated LineSets. For END Min tasks we found, surprisingly, that the Default treatment gave users a significant advantage in accuracy (but not

time), with or without the presence of the outlier. Users in both groups had to count to find the right answer, and counting is a potentially error prone task which we do not believe is likely to be affected by varying node size, other than by the potential for very small nodes to make the task more difficult. The smallest node diameter in a Treated LineSet was 12 pixels, compared to 16 pixels for the Default. Equally, the significant time performance benefit of Treated LineSets for the END Max task may suggest that large nodes are particularly salient, making them faster to identify than in the Default case. This would also help to explain the lack of significant time performance benefit for the ESS min tasks, since small nodes are likely to be less salient.

8. Threats to validity

We identified the following threats to the validity of our study.

Carry-over effect occurs when the exposure to one treatment affects the results obtained for another. We therefore used a between-group design in order to mitigate these effects. Moreover, only participants who had not already taken part in a study relating to this research (i.e. (Tranquille et al. 2016, 2017)) were invited to take part.

Learning effect was considered to be a threat if participants did not receive sufficient and appropriate training prior to the data collection phase. All participants were trained using both paper-based material and the software tool. Additional material was added to the training script for participants in the Treated group so that they were informed about the semantics associated with the graphical properties to which they were exposed. Participants of the Default group were informed of the semantics of the Treated graphical properties in phase 4 prior to preference data collection (the only time they were exposed to this treatment). It was necessary to expose participants the software tool during their training in order to ensure that they were familiar with the environment prior to data collection. Questions similar to the performance and preferential data collection phases were shown to participants during training to ensure that they fully understood each type of question. Although each participant group was shown diagrams that were topologically identical, they were exposed to the treatment that corresponded to their group. Questions order was considered a threat as a predetermined order to the questions could potentially afford a learning effect and therefore bias the results. In order to reduce this, the research software presented the questions to the participants in a random order.

The style of the question used in the study indicates, in part, the extent to which the results generalize. We included a two variants of each task type from each category to ensure that the type of questions were balanced within each category, as specified in section 3.1. This ensured we were able to measure the effectiveness of both treatments when finding the highest extreme value and the lowest extreme value, or counting intersections between two or more sets, which could have had different levels of impact.

Fatigue was considered to be a threat as participants were required to repeatedly answer questions for a considerable amount of time. We therefore designed the study to last approximately 1 hour. This was deemed sufficient to train participants and capture meaningful performance and preferential data without causing undue fatigue. Participants were also given the possibility of resting between phases as well as between individual questions, as specified in section 3.4

Motivation could be a threat if participants did not freely volunteer to take part in the experiment. Consequently all participants were recruited on a completely self-selecting basis. All participants were compensated with a £6 canteen voucher for their

time and participation. They were also given the possibility of consulting the results when published. Participants were advised that they could abandon the session at any time if they so wished.

It was also felt important that the participants undertook the study same environment, reducing variability, so the study took place in a dedicated laboratory that was free from noise and distraction. The participants were exposed to the same hardware, software and room configuration. They also took part one at a time in order to eliminate the behavioural influences of having multiple participants at one time.

It was considered a threat if participants were not familiar with the context of the information conveyed by the diagrams. In addition to the effort required to learn how to interpret the LineSets, cognitive effort would be required understand the context of the information being presented. We therefore chose an information context that participants would be familiar with, as explained in section 3.1.

It was considered to be a substantial threat if any participants had pre-exposure to the information conveyed in the diagrams when answering the questions. Therefore it was important that all diagrams and questions were unique and not derived from information available outside of the study. Similarly, it was considered to be a threat if any participants discussed details of the study with those yet to participate: such a discussion could afford them an advantage over the other participants and therefore bias the results. To manage this, participants were asked not to discuss the study with those yet to participate as explained in section 4.3. Materials used in the study were only accessible during the experiment sessions.

Thus, our results should be taken to be valid within the constraints imposed on the study design and related considerations.

9. Conclusion

Using Bertin's perceptual theories relating to graphical choices, we identified both colour and size as retinal variables can be controlled in LineSets to improve user task performance. In particular, we found that controlling these variables in combination was effective for a range of task types requiring participants to be sensitive to intersections between sets, set cardinalities, and degrees of connectivity in the underlying network. The result leads us to suggest that the default colours used in LineSets should adopt different hues, not different values as is currently the default case. In addition, the thicknesses of the set-lines should be varied so that they reflect set cardinalities and, likewise, the diameters of the nodes should reflect degrees of connectivity. Since these results were obtained using SNAP data and automatically generated LineSets, we have confidence that the results are valid in real-world setting where automated layouts are necessary. Hence, we show that varying visual properties that were previously single valued improve the effectiveness of LineSets.

Future work could consider how to best layout the underlying network in the context of LineSets. At present, LineSets overlay set-lines on a pre-drawn network that is produced without regard to the grouping of the nodes into sets. This gives primary spatial rights (Collins, Penn, and Carpendale 2009) to the network and could compromise the layout of the set-lines. By considering, for example, the thickness of the lines to pass through each node, more space could be made available around nodes so that thick set-lines are less likely occlude nodes. In addition, the nodes could be placed to allow effective routing of the set-lines, following Alper et al's guide of making them close to linear. We conjecture that linear set-lines are less likely to cause occlusion,

which was alluded to as being problematic in the comments made in the preference phase of our study. More generally, our studies show that varying these visual properties can help users understand data, and implies that for other visualization methods (for instance, node-link diagrams or Euler diagrams), varying visual properties that are currently single valued (such as line thickness or circle diameter) may be a fruitful avenue of further research.

References

Alper, B., N. Henry Riche, G. Ramos, and M. Czerwinski. 2011. *Design Study of LineSets, a Novel Set Visualization Technique*, Vol. 17, 2259–2267. IEEE Educational Activities Department.

Alexander, J., A. Cockburn, S. Fitchett, C. Gutwin, and A. Greenberg. 2009. "Revisiting Read Wear: Analysis, Design, and Evaluation of a Footprints Scrollbar." In *SIGCHI Conference on Human Factors in Computing Systems*, pp. 1665–1674 ACM.

Ariely, D. 2001. "Seeing Sets: Representation by Statistical Properties." *Psychological Science* 12 (2): 157–162.

B.Alsallakh, L. Micallef, W. Aigner, H. Hauser, S. Miksch, and P. Rodgers. 2014. "Visualizing Sets and Set-typed Data: State-of-the-Art and Future Challenges." In *Eurographics Conference on Visualization*, 1–21. The Eurographics Association.

Bertin, J. 1983. *Semiology of Graphics*. University of Wisconsin Press.

Blake, A., G. Stapleton, P. Rodgers, L. Cheek, and J. Howse. 2012. *Does the Orientation of an Euler Diagram Affect User Comprehension?*, Vol. 18, 185–190. Springer.

Blake, A., G. Stapleton, P. Rodgers, L. Cheek, and J. Howse. 2014a. "The Impact of Shape on the Perception of Euler Diagrams." In *Diagrammatic Representation and Inference*, July-August, 123–137. Springer.

Blake, A., G. Stapleton, P. Rodgers, and J. Howse. 2014b. "How Should We Use Colour in Euler Diagrams?" In *Proceedings of the 7th International Symposium on Visual Information Communication and Interaction*, August, 149–159. ACM.

Brewer, C., M. Harrower, D. Heyman B., Sheesley, and A. Woodruff. 2013. "Color Brewer." http://colorbrewer2.org/.

Card, S.K, J.D Mackinlay, and B. Shneiderman. 1999. *Readings in Information Visualisation: Using Vision to Think*. Academic Press.

Cheng, P. 2011. "Probably Good Diagrams for Learning: Representational Epistemic Recodification of Probability Theory." *Topics in Cognitive Science* 3: 475–496.

Collins, C., G. Penn, and M. Sheelagh T. Carpendale. 2009. "Bubble Sets: Revealing Set Relations with Isocontours over Existing Visualizations." *IEEE Transactions on Visualization and Computer Graphics* 15 (6): 1009–1016.

Gansner, E. R., and S. C. North. 2000. "An open graph visualization system and its applications to software engineering." *SOFTWARE - PRACTICE AND EXPERIENCE* 30 (11): 1203–1233.

Gottfried, B. 2015. "A Comparative Study of Linear and Region Based Diagrams." *Journal of Spatial Information Science* 2015 (10): 3–20.

Gould, T., B. Harrington, N. Hurst, and *MenTaLguYodruff*. 2003. "Inkscape." http://inkscape.org/.

Healey, C. H., and J. T. Enns. 2012. *Attention and Visual Memory in Visualization and Computer Graphics*, Vol. 18, 1170–1189. IEEE Educational Activities Department.

Jianu, R., A. Rusu, Y. Hu, and D. Taggart. 2014. *How to Display Group Information on Node-Link Diagrams: An Evaluation*, Vol. 20, 1530–1541. IEEE Educational Activities Department.

Koffka, K. 1935. *Principles of Gestalt Pschology*. Lund Humphries.

Leborg, S. 2006. *Visual Grammar*. Princeton Architectural Press.

Leskovec, J., and A. Krevl. 2014. "SNAP Datasets: Stanford Large Network Dataset Collection." http://snap.stanford.edu/data, June.

Mazza, R. 2009. *Introduction to Information Visualisation*. Springer.

Meulemans, W., N. Henry Riche, B. Speckmann, B. Alper, and T. Dwyer. 2013. *KelpFusion: A Hybrid Set Visualization Technique*, Vol. 19, 1846–1859. IEEE Educational Activities Department.

Rodgers, P., G. Stapleton, and P. Chapman. 2014. *Visualizing Sets with Linear Diagrams*, Vol. 22, 27:1–27:39. ACM.

Rodgers, P., G. Stapleton, and P. Chapman. 2015. "Visualizing Sets with Linear Diagrams." *ACM Transactions on Computer-Human Interaction* 22 (6): article 27 (28 pages).

Sahket, B., Simonetto, P. and Kouborov, S. 2014. "Group-level graph visualization taxonomy" Arxiv eprint http://arxiv.org/abs/1403.7421v1.

Schoeffmann, K., M. Taschwer, and L. Boeszoermenyi. 2010. "The video explorer: a tool for navigation and searching within a single video based on fast content analysis." In *Proceedings of the first annual ACM SIGMM conference on Multimedia systems*, 247–258. ACM.

Simonetto, P., D. Auber, and D. Archambault. 2009. "Fully Automatic Visualisation of Overlapping Sets." Vol. 28, 967–974. Blackwell Publishing Ltd.

Tranquille, D., G. Stapleton, J. Burton, and P. Rodgers. 2017. "Evaluating the Effects of Size in Linesets." In *Proceedings of the 10th International Symposium on Visual Information Communication and Interaction*, VINCI '17, 121–128. ACM.

Tranquille, D., G. Stapleton, Jim Burton, and P. Rodgers. 2016. "Evaluating the Effects of Colour in LineSets." In *Diagrammatic Representation and Inference: 9th International Conference*, 283–285. Springer.