#### A Search Set Model of Path Tracing in Graphs

by

Jessica Quinn Dawson

B.A., The University of British Columbia, 2011

#### A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

#### **Master of Science**

in

# THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES (Computer Science)

The University Of British Columbia (Vancouver)

#### October 2013

© Jessica Quinn Dawson, 2013

### Abstract

Path tracing is a common task in many real world uses of graphs that display networks of relationships. Despite previous work in the evaluation of how factors, such as edge-edge crossings, impact the readability of graph layouts, what makes one path-tracing task more difficult than another is not well understood.

To address this question we conducted an observational user study with 12 participants completing a path-tracing task. Our extensive qualitative analysis of the study data led to a detailed characterization of common path-tracing behaviours. We then created a predictive model of the paths that users are most likely to search, which we name the *search set*, based on the behaviours we observed. To validate our predictive behavioural model, and to demonstrate how the search set could be used, we conducted a careful comparison of graph readability factors through a hierarchical multiple regression analysis.

## Preface

The work presented in this thesis was completed under the supervision and guidance of Dr. Joanna McGrenere and Dr. Tamara Munzner. I took the lead on each stage of the work, including designing, implementing and conducting the user study presented in Chapter 3, developing the visualization tools and conducting the qualitative analysis presented in Chapter 4, developing, implementing and validating the model of the search set presented in Chapter 5, performing the statistical analysis presented in Chapter 6, and the writing of this thesis. The user study presented in Chapter 3 was conducted with the approval of the UBC Behavioural Research Ethics Board (BREB) under certificate H10-03336.

Parts of this thesis appear in a journal manuscript, currently in preparation, on which I am the first author: Dawson, J.Q., McGrenere, J., Munzner, T. A Search Set Model of Path Tracing in Graphs.

## **Table of Contents**

Abstrac	t ii
Preface.	iii
Table of	Contentsiv
List of <b>T</b>	ables viii
List of F	ïguresix
Acknow	ledgements xii
Dedicati	on xiii
Chapter	1 Introduction1
1.1	Motivation1
1.2	Summary of contributions
1.3	Thesis organization
Chapter	2 The Basis for a Search Set Model5
2.1	Related work
2.2	Geodesic tendency
2.3	Search set utility
2.4	Research questions and approach
Chapter	3 User Study Design
3.1	Piloting
3.2	Participants
3.3	Dataset and graphs
3.4	Task
3.5	Interface
3.6	Apparatus

3.7	Proc	edure	
Chapte	er 4 Qu	alitative Analysis of Path Tracing Behaviours	15
4.1	Preli	minary node-based analysis of search set	15
	4.1.1	Visualization of node hover overlap	16
4.2	Qual	itative analysis method	
	4.2.1	Data sample	19
	4.2.2	Data preparation and visualization	19
	4.2.3	Coding process	22
	4.2.4	Example of a coded trial	
	4.2.5	Final coded data set	
4.3	Resu	lts	
	4.3.1	Choice and use of anchors for searching	
	4.3.2	Prevalence of the closest to geodesic tendency	
	4.3.3	Likely directions of search	
	4.3.4	Use of apparent and topological paths	
	4.3.5	Revisitations	30
	4.3.6	Path stopping conditions	30
	4.3.7	Continuity and geodesic tendency	
4.4	Sum	mary	
Chapte	er 5 A I	Sehavioural Model to Predict the Search Set	33
5.1	The	search set model	
5.2	Algo	rithmic implementation and validation	
5.3	Usin	g the search set to predict task difficulty	
5.4	Sum	mary	39
Chapte	er 6 Me	asuring Graph Readability Factors Using the Search Set	40
6.1	Rela	ted work	40
6.2	Meth	od	43
	6.2.1	Data sample	
	6.2.2	Dependent measures	44
	6.2.3	Predictor variables	44

	6.2.4	Hypotheses	
6.3	Results		
	6.3.1	Linear correlations for individual effect	
	6.3.2	Multicollinearity between factors	
	6.3.3	Hierarchical multiple regression analysis	
6.4	Sum	nary	
	6.4.1	Summary of hypotheses	
Chapter	·7 Dis	cussion and Future Work	53
7.1	Contr	ibutions and future work	
	7.1.1	The characterization and the predictive behavioural model	
	7.1.2	The search set concept	
	7.1.3	Factor measurement for model validation	
	7.1.4	Regression versus ANOVA for factor characterization	
7.2	Limit	ations	
7.3	Conc	lusion	57
Bibliogr	aphy .		59
Bibliogr Append	aphy . ix A U	Jser Study Materials	59
Bibliogr Append A.1	aphy . ix A U Recru	ser Study Materials	<b>59</b> <b>63</b>
Bibliogr Append A.1 A.2	<b>ix A U</b> Recru	ser Study Materials	<b>59</b> <b>63</b> 
Bibliogr Append A.1 A.2 A.3	<b>ix A U</b> Recru Conse Expen	Ser Study Materials	<b>59</b> 63 63 65 68
Bibliogr Append A.1 A.2 A.3 A.4	<b>ix A U</b> Recru Conse Exper Backs	Ser Study Materials itment poster ent form rimenter script ground questionnaire	<b>59</b> 63 63 65 68 76
Bibliogr Append A.1 A.2 A.3 A.4 A.5	raphy . ix A U Recru Conse Exper Backs Post-s	Jser Study Materials itment poster ent form rimenter script ground questionnaire session questionnaire.	<b>59</b> 63 63 65 68 76 78
<b>Bibliogr</b> <b>Append</b> A.1 A.2 A.3 A.4 A.5 A.6	raphy . ix A U Recru Conse Exper Backs Post-s End c	User Study Materials itment poster ent form rimenter script ground questionnaire session questionnaire of study interview questions	<b>59</b> 63 63 65 68 76 78 80
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7	raphy . ix A U Recru Conse Exper Backs Post-s End c Task	User Study Materials	
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7 Append	raphy . ix A U Recru Conse Exper Backs Post-s End c Task ix B V	User Study Materials	
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7 Append B.1	raphy . ix A U Recru Conse Exper Backs Post-s End c Task ix B V Explo	User Study Materials itment poster	
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7 Append B.1 B.2	raphy . ix A U Recru Conse Exper Backs Post-s End c Task ix B V Explo Explo	User Study Materials itiment poster	
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7 Append B.1 B.2	raphy . ix A U Recru Conse Exper Backs Post-s End c Task ix B V Explo Explo B.2.1	User Study Materials	
Bibliogr Append A.1 A.2 A.3 A.4 A.5 A.6 A.7 Append B.1 B.2	raphy . ix A U Recru Conse Exper Backs Post-s End c Task ix B V Explo Explo B.2.1	User Study Materials	

	B.2.4 Small multiples	93
B.3	Visualization for qualitative analysis of path-tracing behaviours	95
	B.3.1 Steps 13 – 20 for example of a coded trial	97

## List of Tables

Table 5.1	Summary of our behavioural model for predicting a search set	34
Table 5.2	Parameters used in the algorithmic implementation of the final search	
	set model	36
Table 6.1	Descriptive statistics for predictors and for the dependent variables of	
	response time (RT) and error. Predictors are grouped by level of	
	measurement; those that our study is the first to evaluate are shaded	46
Table 6.2	Pearson correlation coefficients $(r)$ between predictor variables and the	
	dependent variables of response time (RT) and error. Predictors are	
	grouped by level of measurement; those that our study is the first to	
	evaluate are shaded	46
Table 6.3	Summary of results from the hierarchical multiple regression analysis of	
	measured factors on response time and error. Predictors that our study is	
	the first to evaluate are shaded.	49

### **List of Figures**

- Figure 3.1 Example of the user study interface. Graphs were displayed on a Wacom Cintiq tablet screen, and participants hovered over nodes with the pen to demonstrate their search process as they completed the study task. The FOUND IT and OK buttons are indicated with the pink labels, and are present on both sides of the screen. 12
- Figure 4.1 Example of small multiple visualizations of all of the hovered-over nodes for one graph trial. There is one small multiple per participant, labeled by the participant number in the top left. Hovered nodes are coloured in orange, with the remaining nodes shown in white. The source and goal nodes are coloured red and blue respectively, and hovers are not shown.
  17
- Figure 4.2 Example of an aggregate visualization showing all nodes hovered over by all participants on one graph trial. Node colour changes from light grey to dark as the frequency with which the node was hovered increases. The source and goal nodes are coloured red and blue, respectively. The convex hull of the source node and goal node and

	their one-hop neighbors (annotated with purple circles) is shaded in	
	light green	18
Figure 4.3	Example of the small-multiple visualization of discrete steps used to	
	support the qualitative coding process; each step is labeled in the top	
	left. The first node in an automatically determined sequence is coloured	
	light orange, and subsequent nodes coloured dark orange; edges along	
	the topological path between them are also coloured orange. Only the	
	first 12 steps are shown. The remaining steps can be found in Appendix	
	B.3.1	21
Figure 4.4	PATHS $1 - 6$ extracted from the steps $8 - 13$ and collapsed into single	
	images. Steps 8 - 12 are shown in Figure 4.3, while step 13 can be	
	found in Appendix B.3.1	24
Figure 4.5	PATHS 7 $-$ 10, extracted from the steps 14 $-$ 19, which can be found in	
	Appendix B.3.1. Where a path spanned multiple steps, it has been	
	collapsed into a single image	25
Figure 4.6	Example of a portion of a graph from the study, with a 2-hop solution,	
	where most participants did not follow the closest-to-geodesic branch in	
	either direction (B-A or R-A) for their first hop at the beginning of the	
	trial (the red and blue nodes are labeled R and B respectively). We	
	attribute this divergence from the geodesic tendency to the large angle	
	size approaching 90°	27
Figure 4.7	Illustration of the ordered groups of similarly likely candidates for the	
	first hop, coloured in grey-scale in decreasing order of likelihood and	
	named by their directionality with respect to the target: directly toward,	
	toward, away, and directly away.	29
Figure 4.8	Example of the past the target stopping condition. Left: A user would	
	typically stop a path at the first node past the target with respect to the	
	starting anchor, where past the target is the line through the current	
	target that was perpendicular (solid green line) to the geodesic straight	
	line between anchor and target (dashed green line). A user tracing from	
	red (R) to blue (B) would likely trace R-C-D and stop without going to	

	E. Right: Exceptions to this condition were when the next hop went	
	directly towards the target (D-G) or in a straight line from the previous	
	hop (D-F)	. 31
Figure 4.9	Example where a straight line appeared to interfere with geodesic	
	tendency (the red and blue nodes labeled R and B respectively). Some	
	participants followed R-A-C, and missed the solution path of R-A-B	. 32
Figure 5.1	Scatterplots of linear relationships between search-set edge-edge	
	crossings and our dependent variables. Top: Average response time (s)	
	and search-set edge-edge crossings. Bottom: Total errors by participants	
	and search-set edge-edge crossings. Both plots show a linear line of best	
	fit	. 38
Figure B.1	Screen shot of version 1 of the exploratory visualization used in the	
	node-based analysis.	. 84
Figure B.2	Screen shot of version 2 of the exploratory visualization used in the	
	preliminary node-based analysis.	. 86
Figure B.3	Screen shot of Version 2.1 of the aggregate view	. 88
Figure B.4	Screen shot of Version 2.2 of the aggregate view	. 90
Figure B.5	Screen shot of Version 2.3 of the aggregate view	. 92
Figure B.6	The small multiples view used in the second version of the exploratory	
	visualization for the node-based analysis	. 94
Figure B.7	Screenshot of the visualization used for the qualitative analysis of path-	
	tracing behaviours	. 96
Figure B.8	Steps $13 - 20$ for the example of a coded trial discussed in Section 4.2.4	. 98

### Acknowledgements

First and foremost, I would like to thank my supervisors, Dr. Joanna McGrenere and Dr. Tamara Munzner. Their constant motivation, high standards, firm support, and good humour ensured that the challenging experience of completing this work was also rewarding and fun. I feel very fortunate to have had them as mentors, and to have had the opportunity to get to know them. The members of the MUX lab and InfoVis group are a diverse and thoughtful crowd, and I am very thankful for their assistance with piloting, their critical and insightful feedback, and their friendship. I would also like to thank Dr. Kellogg S. Booth for being my second reader, and for the detailed feedback that he provided on this work.

My friend Alexander Boswell was a great source of help with the basics of linear algebra, a topic I somehow managed to completely avoid until beginning this thesis, and he answered more questions over text message than I should probably admit to. I am thankful for the love and support of my parents, especially my mother, Wendy Dawson, who has been an inspiration in hard work and dedication. And last, but not least, I am overwhelmingly grateful for my best friend and partner, David Myers, who has provided me with unwavering love and encouragement. His efforts to keep me going behind the scenes – be it with a hug, a tasty meal, a freshly poured mug of coffee, an expertly mixed adult beverage, or a well-timed pun – have ensured my present sanity. Thank you all.

The research in this thesis was supported by funds from AT&T Research and the Natural Sciences and Engineering Research Council of Canada (NSERC).

To Dave and Bokeh,

ambassadors for silliness.

## **Chapter 1**

## Introduction

The creation and evaluation of techniques for the visual analysis of graphs is a heavily studied visualization problem. In this thesis we focus on understanding the graph reading behaviours and factors that impact *path-tracing* [23] performance by users. For a concrete example of path tracing, consider a medical investigator exploring potential disease transmission paths in a graph where nodes represent people and edges represent known contact between them. Specific path-tracing tasks supporting this goal would include finding the number of paths connecting clusters of people, and identifying the shortest paths between two infected individuals. As an abstract task [3], path tracing has been widely studied in related work [5,13,17,18,21,22,28,29,31,35] because it underlies many such real-world use cases that utilize visual representations of graphs [7,23,36]. However, what makes one particular path more difficult to find or follow than another path is still not well understood.

#### 1.1 Motivation

In graph drawing, where the goal is to create spatial layouts of nodes connected by edges, extensive attention has been devoted to understanding what factors affect the quality of the layout. Many factors have been proposed, such as the number of edge-edge crossings, the total curvature of edge bends, and the total area of the drawing. Early work simply proposed factors and then immediately incorporated them into optimization-based layout

algorithms. The factors were considered as constraints to minimize or maximize, and thus a major emphasis was on factors that are amenable to automatic computation. Subsequent work has since begun to determine *graph readability* – whether and how these proposed factors directly affect human understanding in the context of specific kinds of tasks such as path tracing – and to identify new factors through observations of human *graph-reading behaviour*.

The common evaluation of factors for laying out a graph is *global*; that is, one or more factors would be measured across the entire graph. Recent work suggests that for path-tracing tasks a *local* approach is better. Ware et al. investigated the effect of factors measured along the *solution path*; that is, the specific path that is the correct solution for a specific path-tracing problem [35]. They showed that this approach was effective for predicting path-tracing difficulty, and found no additional benefit from including globally measured factors. The intuition behind this result is that global measurements take into account too much of the graph: a graph that scores poorly globally for a set of factors may nevertheless have paths that are easy to trace in regions of minimal clutter.

However, just as global measurement may consider too much of the graph, we suspect that measuring local factors only on the solution path does not take enough of the graph into account. We hypothesize that an even better solution lies between these two extremes, where the full subset of the graph that is relevant for the task at hand is considered; we name this subset the *search set*. Specifically, we propose that the prediction of path-tracing difficulty can be improved by accounting for the impact of important factors on the search set: in this case, all of the paths that a user follows during the tracing task before encountering the solution. Our logic is that if edge-edge crossings on the solution path slow a user down, then the other edge-edge crossings that a user encounters, on paths that are investigated before the solution path is found, must have a similar impact.

Our original motivation for this investigation was the need to both measure and control the difficulty of tracing a path through a graph when designing empirical studies to test visualization techniques. Path difficulty should not be a confounding variable that distorts experimental results. In addition to its obvious value in experimental design, a predictive model of the search set may also prove useful in other applications, such as the design of interaction techniques that modify layouts on the fly. The goal of predicting the search set led us to move beyond factor measurement alone, and address the more general problem of characterizing human path-tracing behaviour when interacting with visual representations of graphs as an end in and of itself.

We conducted an observational user study with 12 participants completing a pathtracing task. Our extensive qualitative analysis of the study data led to a detailed characterization of common path-tracing behaviours. We then created a predictive model of the paths that participants are most likely to search that is based on the observed behaviours. As a demonstration of how the search set could be used and as a preliminary validation of this model we conducted a careful comparison of graph readability factors through a hierarchical multiple regression analysis. Our results show a modest benefit of measuring factors on the search set over the solution path.

#### **1.2 Summary of contributions**

The contributions of this thesis are four-fold: (1) the introduction of the concept of the search set itself, an intermediate level between just the solution path and full global graph; (2) a detailed characterization of path-tracing behaviours based on human-subjects data; (3) the predictive model of the search set; and (4) the regression analysis that provides preliminary support for the model. Of these, (2) and (3) are our primary contributions. A more detailed articulation of each of these contributions is provided at the end of the thesis in Section 7.1.

#### **1.3** Thesis organization

We begin in Chapter 2 with a brief discussion of related work on using human behaviour to understand factors that affect graph readability, and then we introduce the search set concept. Next, we describe our user study in Chapter 3, which included observation of users completing a path-tracing task. For clarity, we present our analysis in three separate chapters. In Chapter 4 we present our qualitative analysis approach, and provide descriptions of common human path-tracing behaviours that we identified. In Chapter 5, we present our predictive behavioural model for the search set. Finally, in Chapter 6 we address the related work on evaluating factors for graph readability, and validate our search set model by comparing the effectiveness of measuring factors at the solution-path, search-set and global levels for predicting path-tracing difficulty. We conclude in Chapter 7 with a discussion of the implications of our findings regarding human path-tracing behaviour and the search set concept, the value of our methodology for untangling the importance of different graph readability factors, the limitations of our study and analyses, and plans for future work.

## **Chapter 2**

## The Basis for a Search Set Model

We first discuss the previous work most closely related to the concept of a search set and behavioural analysis. We leave the related work on graph readability factors for a later discussion, as part of the factor-based analysis presented in Chapter 6. We then provide a definition of the geodesic tendency, explain how it motivates our proposal of a search set, and discuss the utility of the concept. We discuss the motivation for our focus on path tracing as an abstract task, and then explain our approach in terms of our research questions, and how they drove our experimental design and analysis decisions.

#### 2.1 Related work

Our work is situated within two veins of evaluative studies that have looked to human behaviour to identify and explain important factors for judging graph layout quality [5,8,13,32]. One is the behavioural approach employed by van Ham and Rogowitz [8], Dwyer et al. [5], and Purchase et al. [32,33], where users are asked to manually generate or arrange graphs, and then their behaviour is analysed in order to understand what factors and criteria humans naturally rely on. Our work is similar in its emphasis on behavioural analysis through observational studies. While the previous work is an important step in our understanding of factors because it led to both the introduction of new factors and a refined understanding of the impact of existing factors, it does not provide a complete model of how humans trace paths in graphs. The other relevant thread of work includes several studies that use eye tracking with the goal of understanding how users actually read graphs. From a detailed analysis of eye-tracking data, Körner developed a sequential model of graph comprehension that involves separate search and reasoning stages, and then examined the impact of factors on these different stages [21,22]. The outcome of this work is a cognitive model intended to disambiguate between potential underlying mechanisms of visual cognition within a high-level framework. In contrast, our work provides a behaviour-based model specific enough to be used for measuring factors, but we do not attempt to provide any explanations for the cognitive mechanisms.

Huang et al. used eye tracking to study users completing path-tracing tasks, with the goal of actually observing the effect of edge-edge crossings on the user's gaze [13,18]. This work is the most similar to our own in terms of goals: they identify and provide evidence for the existence of a specific behavioural tendency, the geodesic tendency, which strongly affects path tracing. However, this tendency has yet to be put in the context of a full model of path tracing that accounts for other tendencies. We build on Huang et al.'s observational approach to complete a deeper study and characterization of human graph reading behaviours, including the geodesic tendency, than has been done previously. Through this work we show that, using a behaviour-based model of path tracing, it is possible to predict a set of paths that a group of users is likely to follow.

#### **2.2 Geodesic tendency**

A crucial result of the studies from Huang et al. is the identification of the *geodesic tendency* [13]; that is, when attempting to trace a path from a source node to a goal node, people have a tendency to follow the branch that is the *closest-to-geodesic* from the current node to the goal. A *geodesic* is the geometric straight line between two points. Figure 2.1 shows an example: users looking for the path from A to G would likely start by following the series of closest-to-geodesic branches from A to B to C to D. They would next deviate from the closest-to-geodesic and follow A to E, before returning to following the closest-to-geodesic branches from E to F to G. In this case, the solution path was not the first path explored, but rather the second one.



**Figure 2.1** –When tracing a path from A to G, the geodesic tendency predicts that a user would first follow the incorrect path ABCD that is closer to the straight line between A and G, before following the solution path AEFG. Redrawn from Huang et al. [13].

A corollary of the geodesic tendency is that certain paths are unlikely to be followed, either because they are not on a closest-to-geodesic branch, or because they would naturally fall after the solution path in the sequence of paths that a user explores. For example, in Figure 1 someone is not likely to follow AEFH because they will have already found the solution path AEFG. Our definition of the search set is exactly the likely set of paths that a person would search along the way to finding the solution path. In this case, the set is the paths ABCD and AEFG.

#### 2.3 Search set utility

We argue that a behavioural model of the search set is a useful result in its own right, because it is an observation-based characterization of human behaviour when interacting with visual representations of graphs. Our proximate motivation for creating it was to have a way to control for path-tracing difficulty in the design of experiments by having a better measure of factors that affect graph readability. We focus on this application of the search set in the thesis. We conjecture that a predicted search set may also be useful in the design of interaction techniques to support path-tracing tasks in visualizations, or

even of new automatic graph layout algorithms, but we leave these questions for future work.

We expect that measuring factors on the search set will improve prediction of pathtracing difficulty because the factors measured should be those most relevant to the task. Search set factors may also be more broadly applicable than solution-path factors because some instances of path-tracing tasks do not have solutions. In a disconnected graph, for example, a solution path between two points may not exist, but it is possible to calculate a set of paths that a user is likely to follow while making that determination.

#### 2.4 Research questions and approach

The research presented in this thesis was guided by a set of initial questions about the plausibility of the search set concept, and its use as a basis for measuring factors that impact path-tracing difficulty: (Q1) can we identify distinct path-tracing behaviours?; (Q2) how common are these path-tracing behaviours?; (Q3) can we predict the search set based on observed path-tracing behaviours?; and (Q4) how much improvement over previous work is gained by calculating factors for graph layout on a predicted search set?

To answer these questions, we conducted a user study and performed an extensive analysis, which we present in multiple parts. Chapter 3 outlines the design of the user study. Chapter 4 focuses on answering Q1 and Q2 through observation and characterization of path-tracing behaviours. Chapter 5 explores Q3 by incorporating these observed behaviours into the development of a simple predictive behavioural model for the search set. Finally, Chapter 6 focuses on answering Q4 through a hierarchical multiple regression analysis to compare factors measured on the solution-path, search-set, and global levels.

## **Chapter 3**

## **User Study Design**

We collected data through a lab-based observational user study with 12 participants, who were asked to complete a number of path-tracing trials over two sessions, while using a Wacom Cintiq tablet to demonstrate the paths that they followed. Our intention to observe and characterize path-tracing behaviours to explore the search set concept in Chapter 4 guided our choice of a tablet interface for recording the participants' search process. The study was predominantly designed to support the planned hierarchical multiple regression presented in Chapter 6, which influenced the design of the task, the graphs used, the procedure, and the number of trials. Materials for the study can be found in Appendix A

#### 3.1 Piloting

In designing the study, we investigated having users physically trace their search process as opposed to using eye tracking to record the paths that they searched. Eye tracking has a high analysis overhead, and we were concerned that it would not provide sufficient resolution to clearly identify the nodes users were considering in very dense regions of the graph.

We began by piloting the study with 6 participants recruited from the authors' research groups. The sessions lasted about 30 minutes. Participants were tasked with finding the shortest path (of length 2 to 5 hops) between two nodes in graphs printed out

on paper. During the task, participants were asked to trace their search progress by pointing at the nodes that they considered with a capped pen. One of the authors observed participants during the session, and also videotaped the pen movements from above for later review.

We observed from these sessions that, with a small amount of practice, participants became accustomed to moving the pen at the same time that they were searching. We also noted that participants did not always hover over nodes that they could reason about with their peripheral vision. For example, after following a path for four hops, a participant might follow a branch in the direction of the goal, but then realize that the edge was not connected without fully having to look at the node at the other end of the branch. The inability to pick up peripheral vision is also a known constraint of eye tracking [20], so we anticipated that this pointing method would suffice for the exploratory nature of our investigation. For the final design of the observational sessions we chose to display the graphs on an interactive tablet screen in order to support data logging for later analysis.

#### **3.2** Participants

We recruited 12 participants using fliers posted on campus (4 female, aged 20 - 33, M = 23.4). All were students with normal or corrected-to-normal vision and regular colour vision. They each received \$10 per hour of participation, and a bonus \$5 for returning to complete both sessions.

#### **3.3 Dataset and graphs**

We generated 144 graphs for use in the user study and subsequent analysis. We also generated an additional 9 practice graphs, which were only used for practice by participants and were not included in later analysis. The sample size of 144 graphs was chosen to provide enough graphs to create two discrete subsets of the total set for the analysis of human path-tracing behaviours and development of a predictive model in Chapter 4 and Chapter 5 respectively, and the validation of the predictive model in Chapter 6. We conducted a power analysis, which we report on in Chapter 6, and which

determined the appropriate sample size for the planned regression analysis to be 120 graphs. We then added an additional 24 graphs for use in the qualitative analysis and model development in Chapter 4 and Chapter 5.

We used the Watts-Strogatz model [37] to create graphs with small-world properties, as these are considered to be good models of networks from a range of application domains [1]. We selected a graph size of 75 nodes and 150 edges to balance density and difficulty [25]. To support both reproducibility and analysis, we generated layouts in advance of the experiment and saved the coordinates for later use. We did so by running the force-directed placement included in the Prefuse toolkit for 5 seconds to lay out each graph, and saved only graphs with an aspect ratio of 0.8–1.12 to ensure that nodes would appear at a similar size on the screen. Each of the graphs was randomly assigned a source node, and then breadth-first search was performed to assign a goal node to create a single shortest path of 2, 3 or 4 hops. An equal number of graphs were stored as XML files.

#### 3.4 Task

We used a shortest-path identification task. Participants were shown a graph with the source and goal nodes coloured red and blue respectively, and were asked to find the shortest path through the graph from the red node to the blue node as quickly and accurately as possible. The remaining nodes were coloured black. Participants were told explicitly that the path would always be 2, 3, or 4 hops in length. While searching for the path, participants were asked to use the tablet pen to hover over the nodes in the paths that they considered. Nodes became highlighted when hovered over by the tablet pen. Figure 3.1 shows an example of a graph displayed on the tablet.

Each trial consisted of two phases. In the search phase, participants were given a maximum of 90 seconds to find the shortest path and then press a button labelled FOUND IT! located on the side of the screen. In the answer phase, participants were given 20 seconds to demonstrate the path that they had found by selecting each node in the path with the tablet pen, and then press OK to submit their answer. To select a node, users were required to hover over it and then press a button on the side of the pen. Time remaining in



**Figure 3.1** – Example of the user study interface. Graphs were displayed on a Wacom Cintiq tablet screen, and participants hovered over nodes with the pen to demonstrate their search process as they completed the study task. The FOUND IT and OK buttons are indicated with the pink labels, and are present on both sides of the screen.

each phase was displayed at the top of the screen, and the colour of the node highlighting changed depending on the phase: orange highlighting for the search phase, green highlighting for the answer phase. If participants ran out of time in the answer phase, the nodes selected at the time-out point were automatically taken as the participant's answer.

Participants were asked to limit their search for the answer to the search phase. During piloting we noted that participants sometimes realized they made a mistake during the answer phase. To address this issue, we instructed participants to select the nodes they had originally thought made up the answer if this occurred. Finally, participants were shown the correct answer to the trial before beginning the next trial.

#### 3.5 Interface

The Cintiq tablet was inclined to a slight angle, about 25 degrees from the horizontal. The OK and FOUND IT! input buttons used in the task appeared on both sides of the screen, to



**Figure 3.2** – Example of a halo drawn around a node to support identification of node-edge crossings. The near-horizontal edge crosses the node. The four other edges, which pass through the halo, connect to the node.

support both left handed and right handed users. These buttons were configured to accept input only during the relevant stages of the trial to reduce the incidence of mistakes.

To limit the confounding effects of interaction time, we did not provide any complex interaction capability such as zooming or panning the graphs, or moving nodes. To resolve the ambiguity caused by node-edge crossings, each node was drawn with a small white halo around it, as shown in Figure 3.2: edges terminating at that node would pass on top of the halo and connect directly to the node, but unconnected crossing edges were drawn underneath the halo, resulting in a small gap between the edge and the node.

#### 3.6 Apparatus

The experiment was conducted on a Wacom Cintiq 12WX direct input pen tablet connected to a 13" 2.7 GHz Intel Core i7 Mac Book Pro with 8 GB of RAM and Mac OS X Lion 10.7.2. The experiment software was coded in Java using the Prefuse toolkit [10]. For each trial the system recorded a log of the participants' mouse movements, the graph nodes that had been hovered over (computed as any intersection between the cursor position and the node geometry), the task completion time, and the final answer.

#### 3.7 Procedure

The total experiment length was over 2 hours and thus was split across two sessions to avoid participant fatigue. The first session took between 1-1.5 hours, and the second session took  $\sim$ 1 hour. Participants were able to complete the experiment on the same day, but were required to wait a minimum of one hour between sessions.

In the first session, participants completed a brief questionnaire on their background before the experimenter demonstrated the tablet and the task. The tablet was configured to use the participant's dominant hand. Participants then walked through the built-in calibration utility until both the experimenter and participant were satisfied with the pen tip cursor alignment. Participants repeated the calibration and were reminded of these instructions for the second session. Before starting experimental trials, participants completed an equal number of practice trials of each possible solution-path length – 6 practice trials in the first session (two of each length), 3 in the second session (one of each length) – and the experimenter provided feedback to ensure they understood the task.

For each trial, participants completed the task with one of the 144 pre-generated graphs. The presentation order of the graphs was randomized across both sessions, while the practice graphs were shown in the same order. Participants completed 6 blocks of 12 trials at a time, for a total of 72 trials per session. Each session contained an equal number of graphs with each possible solution-path length – 24 each of the 2, 3 and 4-hop graphs – but these were not controlled for within blocks. The use of blocks in the experiment was only to ensure that users took consistent breaks.

After each session the participants rated the task on a Likert scale from 1-low to 7high according to the overall difficulty, and the mental and physical effort required. A post-experiment interview followed the second session. The Likert scale data did not reveal any interesting trends and thus did not factor into any of our analyses, so we do not report on it any further in this thesis.

### **Chapter 4**

## Qualitative Analysis of Path Tracing Behaviours

The focus of the first part of our analysis for the user study pertains to our first two questions concerning the search set concept: (Q1) can we identify distinct path-tracing behaviours?, and (Q2) how common are these path-tracing behaviours?

We began with a preliminary analysis of the nodes hovered over by participants during the study trials, and visualized that data to explore what each participant's search set looked like. This early exploration motivated the central qualitative analysis described in this chapter. First, we manually identified and described paths from the hovered-over nodes in a subset of the study trials. From that analysis, we characterized a number of common path-tracing behaviours.

Once we were able to describe how participants traced paths, we could then develop a predictive behaviour model of the search set, which we present in Chapter 5.

#### 4.1 Preliminary node-based analysis of search set

We began with a preliminary analysis of data collected, focusing on the overlap between the total set of nodes that each participant hovered over at least once during the trial for each of the 144 graphs.

The success rate for the user study trials was low. On average, participants successfully completed 58.7% of the trials (SD = 11.7%, min = 34.7%, max = 79.2%).

However, we felt that this level was appropriate for our study. In order to fully understand human behaviour on the path-tracing task it was important to look at a range of cases from easy to hard, including both successful and unsuccessful attempts.

On average, only 6.1% of the nodes for a given graph that were hovered over by at least one participant were also hovered over by all 12 participants (min = 0%, max = 25%). While we had expected to see some individual variability, we were nevertheless surprised by the extent of this apparent lack of commonality. The suggestion that there was almost no overlap in the paths explored by all users seemed unlikely given the previous work on the geodesic tendency, and so we chose to dig a bit deeper.

#### 4.1.1 Visualization of node hover overlap

We generated a number of visualizations of the node hover data to further explore how the overlap varied. In this section we present views from one of the visualizations that we created. Additional details about these views, and all the visualizations developed for this analysis, can be found in Appendix B.1 and Appendix B.2. The visualization we discuss here showed a graph and all of the nodes that were hovered over by participants for the corresponding experimental trials. The visualization included twelve small multiple views, each of which displayed the nodes hovered over by a single participant in the trial. The graph for a trial was laid out as in the experiment, with the nodes in each small multiple coloured according to whether or not they had been hovered over by that particular participant, as shown in Figure 4.1. A different view, shown in Figure 4.2, aggregated the hovers of all participants onto a single view of the graph, with the frequency of hovers encoded in grey-scale.

By examining these visualizations we noticed that subsets of the participants' hovered-over nodes would often overlap, even though the total overlap across all participants was small. When we incorporated the frequency with which each node was visited, we saw that the most frequently hovered-over nodes tended to fall in a convex hull around the red and blue nodes and their respective 1-hop neighbours, as shown in Figure 4.2. Three participants alluded to this convex hull behaviour in the post-experiment interview, stating for example that they "often tried to look in the area



**Figure 4.1** – Example of small multiple visualizations of all of the hovered-over nodes for one graph trial. There is one small multiple per participant, labeled by the participant number in the top left. Hovered nodes are coloured in orange, with the remaining nodes shown in white. The source and goal nodes are coloured red and blue respectively, and hovers are not shown.



**Figure 4.2** – Example of an aggregate visualization showing all nodes hovered over by all participants on one graph trial. Node colour changes from light grey to dark as the frequency with which the node was hovered increases. The source and goal nodes are coloured red and blue, respectively. The convex hull of the source node and goal node and their one-hop neighbors (annotated with purple circles) is shaded in light green.

between the red and blue nodes" (P11). On average, 93% of the total node hovers for a given graph fell inside the convex hull (min = 73.1% max = 100%). This consistency suggested to us that although the participants' approaches were not identical, there were in fact some similarities in how participants were tracing paths.

#### 4.2 Qualitative analysis method

Motivated by the findings in the preliminary node-based analysis, we moved from considering the data simply in terms of node hovers to reconstructing the paths searched by the participants. By looking at the progression of paths over time, we hoped to characterize common human path-tracing behaviours.

Given the variability we observed in the node-based analysis, we did not feel that we had a deep enough understanding to extract the paths through computation alone. Instead, we chose to manually extract paths from the node hover data using qualitative coding after applying some algorithmic filtering. The data was transformed from *hovers* to *steps*, which were then coded as *paths*. One investigator performed all of this qualitative analysis.

#### 4.2.1 Data sample

At this stage we chose to split our collected data into two groups, a training set (24 graphs) and a validation set (120 graphs). We used the training set as the basis for the qualitative analysis in this chapter, from which we characterized path-tracing behaviours. We reserved the larger validation set for a hierarchical regression analysis that served as a validation of our predictive behavioural model, described in Chapter 6. This type of approach, using a training set and a disjoint validation set, is commonly used in the machine learning communities for model selection and validation [2], and was intended to support testing whether or not the model derived from the training set generalized to the validation set.

The training set was made up of 8 graphs for each of the 3 possible hop lengths, for a total of 24 graphs. These were selected randomly from the total set of 144 graphs. We analyzed all 12 participant trials from the user study for each of these graphs, for a total of 288 trials.

#### 4.2.2 Data preparation and visualization

The raw data in the log files were *hovers* over a node, as described earlier. Some of these hovers were deemed to be spurious and were eliminated from further consideration after manual inspection. Some were automatically filtered out based on a quantitative threshold, while others were discarded as a result of the qualitative analysis process described below. In the automatic filtering, hovers lasting less than 5ms were discarded. This threshold was derived from a combination of quantitative analysis and observation while building the visualization; we found that less than 5ms was an unrealistically short length of time to hover over a node while actually tracing a path. Most discarded hovers

seemed to be caused inadvertently when participants transitioned their search from one area of a graph to another.

After the initial filtering, we transformed the hover data into a sequence of *steps*. Initially a step was created for each individual node hover, in temporal order. From these steps, the investigator could then compose a *path*: a complete sequence of nodes that constituted an intended single path-tracing attempt on the part of the participant. In order to assist the investigator in identifying topological paths, we automatically consolidated two or more successive node hovers into the same step when they were connected by edges.

Despite this automatic process, a path could still be split across multiple steps for several different reasons. First, some paths consisted of a combination of topological and apparent connections between nodes. We saw many examples where participants followed *apparent* paths that were not true topological connections (there was no edge in the graph between consecutive hovers), but these were mistaken for an apparent topological connection because of node-edge crossings. Second, some paths were split across multiple steps because of spurious hovers in the log that the investigator judged to be *incidental* to what the user was actually considering at the time. These were typically nodes crossing or near to a path that the user followed repeatedly, or nodes hovered over during transitions between different parts of the graph.

To support our analysis, we designed a visualization in which the first 20 steps were directly visible as small-multiple views of the trial graph with some nodes and edges coloured to show hover activity. Figure 4.3 shows an example of this visualization, with only the first 12 of the 20 steps shown for space reasons. The first node in an automatically determined topological sequence was coloured light orange, with subsequent nodes coloured dark orange, and edges along the topological path between them also coloured orange. As additional visual support for the analysis, we included an aggregate view similar to the one used in the preliminary analysis (Figure 4.2), and when the investigator hovered over a node, it was highlighted in every small-multiple view and its node ID was shown in a tooltip. An example of the entire visualization can be found in Appendix B.3. We chose to stop analysis after a maximum of 20 steps because our initial



**Figure 4.3** – Example of the small-multiple visualization of discrete steps used to support the qualitative coding process; each step is labeled in the top left. The first node in an automatically determined sequence is coloured light orange, and subsequent nodes coloured dark orange; edges along the topological path between them are also coloured orange. Only the first 12 steps are shown. The remaining steps can be found in Appendix B.3.1.

exploration showed that later steps tended to be more chaotic and less representative of common behaviours.

#### 4.2.3 Coding process

All of the steps in a trial, up to the maximum of 20 visualized, were described with at least one code. The paths that the investigator identified were coded as a sequence of node IDs, in addition to a number of other attributes, which we describe next.

First, a path could be either a true *topological* path or an *apparent* path. Second, the investigator coded the *target* node that the path was going towards, which could be the *source* node (red), the *goal* node (blue), or some *other* node in the graph. Third, the *anchor* node that the path started from was identified, which again could be the source node (red), goal node (blue), or some other node in the graph.

The final two attributes for a path were used to describe the branches that the participant followed for each hop of that path. One was the *direction* (forward, right angle, or backward) with respect to the target, of each branch in a path. The investigator used her approximate judgement rather than exact angles in determining whether a branch went towards the target, at a right angle from it, or away from it (i.e., whether the branch went closer to the target, kept roughly same distance from it, or went away from it). The last attribute was whether the branch at a particular hop was the *closest-to-geodesic* branch from the associated node to the current target. We did not expect participants to be skilled at judging very small differences in angles, and observed this to be the case in early exploration of the data. Thus, when the difference between two branches overlapped, the investigator recorded both as having the closest-to-geodesic property.

In addition to describing paths, the investigator generated codes for other types of movements by participants that emerged during the coding process as being potentially important. These were *jumps* between nodes, *switches* of a *target* and/or *anchor*, *checks* of nodes or node-edge crossings, and *doublebacks* over paths just traced. The investigator used the same attributes described above in these other codes as appropriate.

Finally, the investigator also coded *incidental* node hovers. These were hovers over nodes that occurred between two nodes in a coded path or during some other movement, but were judged to not signify the node a user was actually considering at the time.

#### 4.2.4 Example of a coded trial

To illustrate the coding process, we next walk through one example of the codes and the attributes used to code one participant trial. Figure 4.4 and Figure 4.5 demonstrate the paths identified from all of the steps in the example. Where a path spanned multiple steps, we show these steps collapsed into a single image; the red and blue nodes are labelled R and B respectively. The first 12 steps of this trial are also shown without annotation in Figure 4.3, as the investigator saw them during the annotation process. The remaining un-annotated steps can be found in Appendix B.3.1.

Figure 4.4 shows the first six paths. In PATH 1 (R-C-B, steps 1 - 2, anchor = red, target = blue), the participant follows two hops in the closest-to-geodesic direction; the hop from R-C is a true topological path, but the second hop is apparent, because no edge exists between C-B. In PATH 2 (B-D, steps 2 - 3, anchor = blue, target = red) the participant follows one hop in the closest-to-geodesic direction. The node I1 is coded as incidental even though it connects to D with an edge, because B is also connected to D. In PATH 3 (R-C, steps 4 - 7, anchor = red, target = blue) the participant again follows one hop along the closest-to-geodesic branch, repeating part of PATH 1. C1 and C2 from steps 5 and 6, respectively, are examples of checks around node C, to which the participant returns in step 7.

PATH 4 (B-E-F-G-R, steps 8 - 9, anchor = blue, target = red) is another example of an apparent path, because there is no edge between F-G. The first hop B-E goes in a right angle direction. The remaining hops all take the closest-to-geodesic branch. PATH 5 (R-G-F-E, steps 9 - 11, anchor = red, target = blue) is a doubleback of the previous path, PATH 4. In PATH 6 (E-F-H, steps 11 - 13, anchor = E, target = red) the participant again retraces part of PATH 4, but deviates to follow a true topological connection between F-H, which is in the closest-to-geodesic direction. Step 12 shows another incidental hover, I2; it seems clear that the participant is following the E-F edge, and thus would probably not think I2 is connected to either E or F.


**Figure 4.4** – PATHS 1 – 6 extracted from the steps 8 – 13 and collapsed into single images. Steps 8 – 12 are shown in Figure 4.3, while step 13 can be found in Appendix B.3.1.



**Figure 4.5** – PATHS 7 – 10, extracted from the steps 14 - 19, which can be found in Appendix B.3.1. Where a path spanned multiple steps, it has been collapsed into a single image.

PATHS 7 – 10 are shown in Figure 4.5. In PATH 7 (R-I, step 14, anchor = red, target = blue), the participant follows one hop in the backward direction. In PATH 8 (R-J-G, step 15, anchor = red, anchor = blue), the participant follows a different branch in the backward direction for the first hop, and then follows the closest-to-geodesic branch for the second hop. Between PATHS 8 and 9, another incidental hover occurs in step 16, which is not shown. In PATH 9 (R-K, steps 17 - 19, anchor = red, target = blue), the participant follows another branch in the backward direction. Finally, in PATH 10 (R-I-L, step 19, anchor = red, target = blue), the participant follows the closest-to-geodesic branch of the second hop. Between each of the second hop. Between each of the paths from PATH 1 to PATH 6, we observe *switching*, whereas for PATHS 7 – 10, the participant continues to search around the red source node.

#### 4.2.5 Final coded data set

We eliminated 11 of the 288 trials during the coding process because participants entered the answer phase of the trial without hovering over any nodes in the search phase. The investigator ultimately classified 95.8% of the steps in the remaining 277 trials with at least one code.

The remaining 4.2% of steps could not be made sense of in the context of our coding scheme, and were coded as *unclassified*. We suspect that some of these unclassified trials were caused by incomplete data, for example, if a participant missed a node with the pen tip despite looking at it. We had anticipated this limitation of the tablet, but decided that this number was small enough to be acceptable. In addition, some of the unclassified steps may have been deliberate but uncommon types of movements that we simply did not see often enough to classify with a unique code.

#### 4.3 Results

We now describe the behaviours that emerged during the coding process, and from subsequent analysis of the final code set. Some of these findings stem from differences between specific graphs and the common cases we observed, whereas others hold across all graphs.

#### 4.3.1 Choice and use of anchors for searching

Although participants were instructed to search from the red node to the blue node, we found that they often searched from blue to red, especially when the task was more difficult. On average, the majority of paths coded across the 24 training graphs used either red or blue as the starting, or anchor node (M = 86.9%, SD = 10.4%, min = 64.8%, max = 100.0%). Participants were unlikely to search out from intermediate nodes that were part of the way along promising paths. The extracted paths in Figure 4.4 demonstrate this behaviour; the participant switches anchor and target before beginning each path from PATHS 1 - 6, searching back and forth in alternating directions between red and blue. The participant only once uses an intermediate node as an anchor, in PATH 6.



**Figure 4.6** – Example of a portion of a graph from the study, with a 2-hop solution, where most participants did not follow the closest-to-geodesic branch in either direction (B-A or R-A) for their first hop at the beginning of the trial (the red and blue nodes are labeled R and B respectively). We attribute this divergence from the geodesic tendency to the large angle size approaching 90°.

#### 4.3.2 Prevalence of the closest to geodesic tendency

Participants preferentially followed paths along nodes forming closest-to-the-geodesic branches, suggesting strong evidence of a geodesic tendency. On average, the majority of the identified paths for a given graph (M = 65.5%, SD = 9.7%, min = 49.0%, max = 81.3%) fell along the closest-to-geodesic branches either for all hops (M = 39.4% SD = 10.7%, min = 15.8%, max = 56.3%), or for all but the first or last hop in the path (M = 26.2%, SD = 8.2%, min = 15.8%, max = 47.4%). In the post-experiment interviews, eight participants explicitly described strategies involving the closest-to-geodesic path.

We also examined how common it was for the very first branch followed in a trial to be the closest-to-geodesic branch for the starting anchor. The majority of participants began trials with the closest-to-geodesic branch (M = 60.3%, SD = 25.2%, min = 16.7%, max = 91.6%), which again points to the strength of the geodesic tendency. However, for six of the graphs in the training set, this number was well below 50%, and as low as 16.7%, which suggests that other factors could override this tendency. In particular, we noted that as the angle between the closest-to-geodesic branch and the straight line to the target increased to 90° or larger, it became more likely that the participant would pick a different branch.

Figure 4.6 shows an example. For this graph, only three participants started by following the closest-to-geodesic branch from red or blue, which in both cases went to A. Another interfering factor was the length of the closest-to-geodesic branch with respect to

the target distance; if the target was far away and the closest-to-geodesic branch was much shorter than surrounding branches, or if the target was very close and the closestto-geodesic branch went past the target, then the closest-to-geodesic branch seemed less likely to be searched at all.

#### 4.3.3 Likely directions of search

Despite the prevalence of the geodesic tendency, participants did spend considerable time searching along other branches. Typically, the likelihood of expanding to nodes that were not along the closest-to-geodesic branch increased with the amount of time a participant spent on a trial. We saw the largest divergence from the geodesic tendency for the first hop of paths emanating from red or blue. However, participants were likely to return to the closest-to-geodesic branch for subsequent hops.

Our analysis did not suggest that there was a fully systematic ordering of the likelihood of searching in a particular direction for the first hop. For example, branches did not simply become decreasingly likely as the size of the angle with the geodesic straight line to the target increased. However, the order was also far from random.

As shown in Figure 4.7, we loosely grouped directions of the first hop into four ordered groups more specific than those we used in coding. The first is *directly towards*, meaning a small angle with respect to a line straight towards the target; next is *towards*, up to slightly beyond a right angle. The third group is *away*, for even larger angles; the last and least likely category is *directly away*, for angles that were essentially in the opposite direction from the target. Participants had a roughly similar likelihood of selecting a branch within each group, starting with *directly torward*, and earlier groups were searched to exhaustion before later ones were begun. This also extends to intermediate nodes along promising paths, but the range of angles describing similarly likely branches at such nodes was much larger. The second hop in a path was more likely to go towards or directly towards the target were uncommon.

It appeared that participants tended to exhaust the options around red and blue before exhausting the options around nodes two or three hops along a path. This phenomenon partially explains our observation that participants tended to return to closest-to-geodesic candidates for subsequent hops in paths. It also provides some explanation for the



**Figure 4.7** – Illustration of the ordered groups of similarly likely candidates for the first hop, coloured in grey-scale in decreasing order of likelihood and named by their directionality with respect to the target: *directly toward, toward, away*, and *directly away*.

relationship between the angle of the closest-to-geodesic branch and the likelihood that it would be followed first in a trial. When the closest-to-geodesic branch goes directly towards the target, it becomes very likely it will be followed first. But as the angle increases to 90° or larger, it becomes more likely that the participant would follow any other branch in the same group. We suspect that in these instances, other factors, such as path straightness, begin to take precedence.

#### 4.3.4 Use of apparent and topological paths

Participants primarily followed topologically connected paths, but apparent paths created by node-edge crossings could cause significant distraction. Despite the fact that all users were trained to use the halos to identify node-edge crossings, some reported that it required extra effort to realize that they were looking at a crossing. Such paths were a common source of error, especially when they lay on top of a branch directly connected to the red or blue nodes. We see examples of this in PATHS 1 and 3, and PATHS 4 and 5 in Figure 4.4. The participant examines node C in both PATHS 1 and 3, even though it creates an apparent path, presumably because it seems so promising. Similarly in PATHS 4 and 5, the participant repeatedly follows the apparent path between nodes G and F, taking considerable time to determine that there is a node-edge crossing before trying a different route from F in PATH 6.

#### 4.3.5 Revisitations

We observed that participants often revisited the same path again and again. This repetitive behaviour took two forms. We saw many instances of *doublebacks*, where participants would retrace a path one or more times immediately after tracing it the first time. We also saw that participants would return to a path after tracing others, even if they had followed it multiple times before. This finding is not surprising in light of the known limits of working memory for remembering the results of previous searches [36]. P2 admitted that he would often "look at a path more times than was helpful." Some participants also related this behaviour to the tendency to search within the convex hull and along closest-to-geodesic path. P6 explained that "I would try to counteract and look for different paths, but the [closest-to-geodesic path] was more natural, and it was harder to force myself to look away."

#### 4.3.6 Path stopping conditions

Contrary to what might be expected, participants often did not follow every path that they started until they reached the maximum length allowed by the trial. However, we did observe some commonalities in when participants were likely to stop following a path. Some stopping conditions were largely dictated by the experimental tasks and common sense: participants typically stopped a path when the number of hops equalled the maximum of four, they had cycled to reach a node already in the current path, or they had reached the target. Other stopping conditions were less obvious. We found that participants often stopped when the number of hops was one less than the maximum path length in the task. We also saw that they frequently stopped when their current path took them past, or nearly past, the target with respect to the starting anchor. We defined *past* the target as the line through the current target that was perpendicular to the geodesic straight line between the path anchor and the current target, as illustrated in Figure 4.8 – Left, where the geodesic is the dashed green line and the perpendicular through the blue target is the solid green line; a user tracing from red (R) to blue (B) would likely trace R-C-D, stopping at D, which is just past the blue node, and not get to E. There were two exceptions to the final two conditions of stopping at the maximum less one or going past the target. Figure 4.8 – Right illustrates an example of these exceptions for the past the target condition. A participant tracing from R to B along the path R-C-D would be less



**Figure 4.8** – Example of the past the target stopping condition. Left: A user would typically stop a path at at the first node past the target with respect to the starting anchor, where past *the target* is the line through the current target that was perpendicular (solid green line) to the geodesic straight line between anchor and target (dashed green line). A user tracing from red (R) to blue (B) would likely trace R-C-D and stop without going to E. Right: Exceptions to this condition were when the next hop went directly towards the target (D-G) or in a straight line from the previous hop (D-F).

likely to stop at D as in the previous example if: the next hop formed a nearly straight line with the previous hop (as in D-F), or the next hop went directly towards the target (as in D-G).

#### 4.3.7 Continuity and geodesic tendency

In previous work, Ware et al. found continuity, namely the straightness of the path, to be a very important factor [35]. Huang et al. avoided variation in continuity in order to avoid confounding their results on the geodesic tendency, and suggested based on their results that geodesic tendency took precedence over path continuity [13]. However, we often observed the opposite effect, suggesting that interaction between path continuity and geodesic tendency is quite complex.

We found that in many instances participants would follow straight paths for more hops than they would "bendy" paths, and that straight paths could distract participants by causing them to miss a branch connecting to the solution. Figure 4.9 shows an example, where R-A-B was the solution. In this graph, three participants who followed the branch from R to A, next followed the branch from A to C, and missed the branch from A to the blue node (B). Only one of the participants detected the solution in the steps that immediately followed. In such instances, we suspect that the Gestalt principle of



**Figure 4.9** – Example where a straight line appeared to interfere with geodesic tendency (the red and blue nodes labeled R and B respectively). Some participants followed R-A-C, and missed the solution path of R-A-B.

continuity [36] sometimes contributes to participants perceiving the straight line formed by multiple nodes as a single hop, causing them to skip over interconnected nodes without considering their branches.

# 4.4 Summary

Through the coding process described in this chapter, we determined that it is possible to identify distinct path-tracing behaviours, addressing Q1. Further, we were able to characterize and describe a number of common path-tracing behaviours exhibited by our participants, addressing Q2. The behaviours include the use of both topological and apparent paths, the conditions under which participants stop following paths, the use of anchors in search strategies, the likely directions for the first hop in a path, and the tendency to revisit previously followed paths. We also verified the prominence of the geodesic tendency, and its interaction with path straightness and other behaviours that we observed.

All of these findings are useful in their own right as descriptions of human pathfollowing behaviours when interacting with visual representations of graphs. They were also crucial in helping to develop a predictive behavioural model of search set, which we present in the next chapter. While many of the behaviours that we observed could play out in different ways for different participants, enough commonalities exist to allow us to make informed guesses about the likely set of paths that a group of users may search.

# **Chapter 5**

# A Behavioural Model to Predict the Search Set

This chapter is devoted to our third research question: (Q3) can we predict the search set based on observed path-tracing behaviours? To explore this question, we developed a simple predictive model of the search set based on the strongest common behaviours that we described in Chapter 4. In this chapter we describe the components of the model in detail, and discuss our preliminary validation of its effectiveness in predicting the search set and as a basis for measuring factors for predicting task difficulty. We look at further validation approaches in Chapter 6.

### 5.1 The search set model

Table 5.1 shows a summary of our predictive behavioural search set model. The model takes as input a network graph with a defined solution between two points, which are used as anchors to explore likely paths. The model is designed to predict the set of paths that a *group of users* would be likely to search, rather than the set of paths that *one individual user* would use. The output from the model is ordered batches of unique paths, where within each batch a set of paths is unordered and considered to be similarly likely; together, the paths in these batches compromise the search set.

The model searches out from both anchors. When searching out from one anchor, the other anchor is used as the target for paths, and vice versa. This choice is based on our observations that participants regularly switched between using the red and blue nodes as

# A 3-step predictive behavioural model of the search set

Input: a connected network with a unique solution between a source node and a goal node.

Anchors: source node, goal node Target: target = goal (when anchor = source), and vice versa

3 –step model:

- 1. Generate Batch: from each anchor, generate a batch of likely candidates branches for the first hop in a path:
  - i. *If all first hop branches have been considered*: revisit each batch in sequence and generate likely branches for second hop;
- 2. Follow paths from batch candidates: for each candidate, follow along the closest-to-geodesic branch towards the target until a stopping condition has been met:
  - i. At each hop, add the current path, if not yet contained, to the search set;
- 3. When out of candidates: check if the solution has been found from both anchors:
  - i. *If yes*: Stop the search;
  - ii. *Else:* repeat from step 1 with the next most likely batch of candidate branches.

Output: Ordered batches of paths, where the paths are unordered within each batch, comprising the predicted search set between the source and goal nodes.

Stopping Conditions:

S1) reach the target

S2) number of hops = solution-path length minus one \*

S3) reach a node already in the current path (cycle)

S4) pass target with respect to anchor\*

\*Exception to allow one additional hop if:

- i. the hop forms a straight line
- ii. the hop goes *directly towards* target

 Table 5.1 – Summary of our behavioural model for predicting a search set.

anchors and targets in our study task. To begin in Step 1, we select a batch of likely candidate branches from each anchor to comprise the first hop in a path. The batches correspond to the groups of likely directions of search described in Section 4.3.3: *directly towards, towards, away,* and *directly away*. To start we create a batch comprising the most likely candidate branches, those that are in the directly towards group. All the candidates in one batch must be exhausted before generating the next batch, because we expect each candidate in the same group to have a similar likelihood.

Once the first batch is generated, Step 2 of the model iterates through each candidate, and from this candidate follows the closest-to-geodesic branch between the candidate and the target. If two potential branches have very similar angles or overlap, then both are saved as the closest-to-geodesic branch. This choice is based on our observation that real users make imprecise rather than precise angle judgements. At each hop in the path, we add the path so far to the search set. Thus, if the model follows a path for four hops, we would add four paths to the search set: the first path would contain the first hop, the second path the first and second hops, and so on. This choice is based on the observation that participants often did not simply search all the way to the maximum number of hops allowed by a trial, but instead they revisited parts of paths again and again. The search set contains only one copy of each path, even if that path is encountered multiple times. To determine when the model stops following a particular path, we constructed four different stopping conditions, S1 – S4, which are directly based on the common stopping patterns that we characterized in Section 4.3.6.

Finally, in Step 3 we check to see if the search set contains the solution path for the graph from both the source to the goal, and from the goal to the source. This is to account for our observation that many participants use the goal as an anchor for search, and the fact that a single individual might find the solution in either direction. If the solution path has not been found in both of these directions, we return to Step 1 and generate a new batch of candidates using progressively larger angles from the straight line to the target than were used in the previous batch. Once all of the one-hop branches around both anchors have been considered in previous iterations, we expand the candidates to include two hops around the anchor. To do this, for each subsequent batch we revisit a previous batch, starting with the first batch. For each one-hop candidate in the revisited batch, we

Search Set Algorithm Parameters							
Sizes fo	r Groups of Directions						
From ea	ch anchor, select one-hop candidates:						
Batch	Description	Range					
1	Directly towards target	0° - 50°					
2	Towards target	50° - 100°					
3	Away from target	100° - 165°					
4	Directly away from target	165° - 180°					
From ea	ch candidate in batch, select two-hop	candidates:					
Batch	Description	Range					
5 - 8 <del>5 - 8</del>	<i>Towards</i> target: - Batch 5 from batch 1 candidates - Batch 6 from batch 2 candidates - Batch 7 from batch 3 candidates - Batch 8 from batch 4 candidates	0° - 100°					
9 - 10	<i>Towards</i> target: Away from target - Batch 9 from batch 1 - Batch 10 from batch 2	100° - 165°					
Thresh	old for multiple closest-to-geodesic	branches					
Angular divergence from straight line $< 13^{\circ}$							
Stoppin	g condition angle definitions						
S4-i) S4-ii)	Straight line (angle between hops) Directly towards	165° - 180° 0° - 50°					

 Table 5.2 – Parameters used in the algorithmic implementation of the final search set model.

select all of the likely second hop candidates, again relying on the concept of groups of likely directions. If the solution path cannot be found in both directions after iterating on all of these batches, we judge the task to be very difficult and stop after the last batch has been processed.

#### 5.2 Algorithmic implementation and validation

We programmed an algorithmic implementation of our model so we would be able run it on the graphs in our data sample. To implement the model we had to assign specific parameters for the angle boundaries of each batch, as well as for the stopping conditions and the choice of geodesic shortest branches.

Our final parameters are shown in Table 5.2. We iterated on these parameters extensively before settling on the final choices. In order to measure the fit of the model using different parameters, we ran the algorithm on the 24 training set graphs and observed the overlap between the predicted search set and the data collected during the study. We found that as long as the construction of the batches followed a general separation of directions for first hop candidates into the four groups of *directly toward*, *toward*, *away* and *directly* away, the fit did not change dramatically in response to small changes in the exact parameters. The parameters for the second hop candidates, for which we used broader groups of toward (inclusive of directly toward) and away, were similarly robust. Our final parameters were selected to be consistent and generalizable, rather than being overly fit to our particular data set.

Once we had settled on these final parameters, we ran the algorithm implementation on each of our 144 study graphs. The predicted search set produced by the algorithm contained, on average for each graph, 86% of all of the nodes hovered over by participants in the study. Coincidentally, on average for each graph, 86% of the predicted nodes were also hovered over at least once during the study. We consider these results to be a very appropriate fit with the study data, and adequate for a first attempt at developing a predictive behavioural model.

# 5.3 Using the search set to predict task difficulty

Finally, we conducted a preliminary exploration into whether or not factors measured on the search set would be effective predictors of path-tracing task difficulty. We selected



Figure 5.1 – Scatterplots of linear relationships between search-set edge-edge crossings and our dependent variables. Top: Average response time (s) and search-set edge-edge crossings. Bottom: Total errors by participants and search-set edge-edge crossings. Both plots show a linear line of best fit.

the factor of edge-edge crossings, given its prevalence in previous work and our intention to use it in the hierarchical regression analysis described later in Chapter 6. We measured search-set edge-edge crossings for each of the 24 training set graphs by summing the crossings on each path in the set (M = 336.4, SD = 298.7, min = 37, max = 1212) and we measured difficulty both by response time and by total errors.

We used bivariate Pearson correlations to examine the individual effect of search-set edge-edge crossings on average participant response time in seconds (M = 42.54, SD = 22.5), and total errors (M = 5.5, SD = 3.3) for the training set graphs. Response time was measured as the time the participant spent in the search phase, before pressing FOUND IT!. We found strong positive correlations of search-set edge-edge crossings with response time (r = 0.772, p < 0.01), and with error (r = 0.582, p < 0.01). In general, |r| > .10 is considered to be a weak correlation, |r| > .30 a moderate correlation, and |r| > .50 a strong correlation [4]. Scatterplots of these relationships are shown in Figure 5.1, along with the line of best fit.

### 5.4 Summary

With respect to our third research question, Q3, our results suggest that by using the human path-tracing behaviours that we characterized previously in Chapter 4, it is possible to accurately predict the search set for a group of users. Further, our exploration into the use of the factors measured on the search set for predicting response time and total errors yielded promising results, encouraging us to perform the more in-depth validation that we present next in Chapter 6.

# **Chapter 6**

# Measuring Graph Readability Factors Using the Search Set

The focus of the last stage of our analysis is on answering our final question: (Q4) how much improvement over previous models is gained by calculating factors for graph layout on a predicted search set? This analysis was intended to serve as a validation of our predictive behavioural model as well as an example of how the search set might be used. To do this, we compared the relative importance of factors measured at three levels: the solution path, the search set, and globally. As a part of our analysis, we also evaluated the impact of node-edge crossings, which had not been previously investigated in a user study.

We begin with a discussion of previous work on evaluating factors that effect graph readability, including methodologies for evaluation. We present results that show a modest improvement of measuring factors on the search set over measuring on just the solution path. More crucially, we also identify important differences in the relative contributions of these factors in predicting response time and error.

# 6.1 Related work

Starting with Purchase et al. [31], a number of studies have sought to evaluate the impact of many different factors on human understanding of graphs. Factors studied include edge bends [28,31], edge length [8,35], orthogonality [5,28,33], angular node resolution [14,28], edge crossing angles [16,17], clustering [8], node-spacing [5,14] and edge stress [5], and edge-edge crossings [5,8,14,17,21,22,28,33,35]. More recently, Purchase et al.

have also explored the effects of visual features on memorability [24] and on mental map preservation using dynamic layouts [27]. However, many factors that are commonly incorporated into layout algorithms remain unexamined by controlled experiments. One such factor is node-edge crossings, which we evaluate for the first time in our analysis.

Evaluative studies of factors affecting graph readability have predominantly relied on significance testing to conclude that a factor is important or not. A handful of studies have further attempted to create priority lists of factors based on their relative importance, but these have largely been based on significance testing [14,28] or human judgements [11,30]. However, in these examples the magnitude of the effects are rarely reported, and such approaches are limited in their ability to untangle how different factors interact [34]. In their study of factors on the solution path, Ware et al. introduced the use of multiple regression for evaluating the impact of factors on task difficulty [35], and argued for its further use in evaluation of graph readability factors. Multiple regression inherently provides a measure of effect size, and has the additional benefit of assigning quantitative weights to factors, which can be useful when considering the importance of one factor over another.

To our knowledge, only one other study, by Huang and Huang, has used multiple regression for untangling the relative contributions of factors [17]. We adopted this methodology to compare the relative importance of factors measured at the solution-path, search-set, and global levels. We differ from previous work in our focus on the incremental validity of these factors. *Incremental validity* is a concept from clinical psychology that focuses on "the degree to which a measure explains or predicts a phenomenon of interest, relative to other measures" [9], and on the utility of variables in terms of cost and efficiency [19]. In particular, we show that factors measured on the search set show modest improvements over what can be explained with previously studied factors measured on the solution-path and global levels.

Some studies have suggested that some factors may also have a varying impact depending on the task that a user must perform with a graph. Purchase used a shortest path identification task, as well as two tasks related to graph connectivity, in a study that concluded that edge-edge crossings are the most important factor [28]. However, in a study of factors impacting sociogram use, Huang et al. concluded that edge-edge crossings are only important for path-tracing tasks [14]. Similarly, in a study of eye movements, Körner found evidence that edge-edge crossings have no impact during 'search' tasks, but do have significant impact during the 'comprehension' tasks that involve considering the edges between nodes [22]. Conversely, Dwyer et al. found no effect of edge-edge crossings for either path-tracing or connectivity tasks [5]. In our study, we focus specifically on untangling the factors that effect path-tracing difficulty. The concept of search set may shed some light on the underlying reasons for the mixed results in previous work, as we discuss later in Chapter 7.

In examining different tasks, the most common measure of graph readability used in previous work has been response time. Some studies have also examined the relationships of factors to measures such as error, user preference, and (less commonly) cognitive load [15]. While previous work has recognized that, for example, the factors that make a task take longer do not necessarily correspond to those that increase the likelihood of error, little work has examined the relative difference of factor importance for different measures of difficulty. Such an approach is crucial for understanding how factors might need to be traded off in order to achieve the best balance for all measures. A notable exception comes from Huang and Huang [17], who also used a regression approach to examine the relative impact of global edge-edge crossings and global crossing angles on four different measures of task difficulty. In our study, we provide a nuanced discussion of the ways that several factors affect response time and error in varying ways.

The prior studies that we have discussed thus far have all focused on globally measured factors, with only one exception; Ware et al. [35] studied factors measured on the solution path. In exploring the effectiveness of factors for predicting response time in a shortest path identification task, they identified a new factor of path continuity, and found significant effects of solution-path length, path continuity, total and average line length of the path, total branches on the path, average crossing angle on the path and total edge-edge crossings on the path. Further, they showed that with only four of these factors – solution-path length, continuity, edge-edge crossings and branches - they could account for 78.4% of the variance of response time in their study, and they identified path length and continuity as having the largest contributions. A crucial finding was that the globally

measured edge-edge crossings did not account for any additional variance on top of the solution-path factors. To our knowledge, ours is only the second study to compare the effect of graph readability factors measured locally in addition to globally on response time. We are the first to do so for error. Our study is also the first investigation of factors for either time or error at three different levels: we introduce the search-set level in addition to the global and solution-path levels.

# 6.2 Method

Our methodology follows directly from Ware et al. [35] and Huang and Huang [17]. We measured a selection of factors on different levels of the graph, which we call predictors. We use bivariate correlations to examine the individual effects of these predictors on user performance, and to determine which factors have any significant impact. We then use hierarchical multiple regression to factor out the internal relationships between the predictors, in order to examine the relative contributions of each factor in predicting performance. This approach allows us to examine the total percentage of variance in performance accounted for by the predictors, as well as any overlaps in what the predictors explain. The use of hierarchical regression follows recommendations from the literature on incremental validity [19] and on the benefit of requiring the researcher to reason about and justify the order in which variables are entered into a regression model.

# 6.2.1 Data sample

Our sample consisted of the 120 graphs in the validation set, which were those that that remained after we removed the 24 graphs for the training set used in Chapter 4. This number was determined through a power analysis using the following parameters:  $R^2 =$ 0.13, alpha = 0.05, and 9 independent variables. This analysis gave us a power level > 0.80 for 120 graphs, which is conventionally considered to be an acceptable level [4]. From this level of power we expected to be able to detect medium and large effects, where  $R^2 = 0.02$  is a small effect,  $R^2 = 0.13$  is a medium effect, and  $R^2 = 0.26$  is a large effect [4].

The sample consisted of an equal number of graphs with each possible solution-path length: 40 graphs each of lengths 2, 3 and 4 hops.

#### 6.2.2 Dependent measures

We measured user performance on each of the 120 graphs with two dependent variables: average response time (RT) and the number of incorrect user responses (error). We chose these measures because we were interested in the impact of the predictors on both correct and incorrect answers – it is important to understand how long a user might spend only to find an incorrect answer.

Response time (RT) was recorded as the average time to complete the search phase for each trial for all 12 participants, between 0 - 90 seconds. Error was calculated as the total number of incorrect responses by participants for each graph, between 0 - 12.

#### 6.2.3 Predictor variables

We selected nine different factors to measure on each of the 120 graphs in the validation set, which we used as predictor variables. A subset of our predictors were those found to by most important by Ware et al, all of which were measured on the solution path [35]: the length of the path in hops (sp-ln); the continuity of the path, calculated as the sum of the angles in degrees at each step (sp-cn); the total edge-edge crossings on the path (sp-ex); and the sum of the branches on each node on the path (sp-br).

For comparison, our analysis also looked at factors that were not measured on the solution path. We selected edge-edge crossings as a factor to measure on the search-set and global levels because edge-edge crossings are often cited as the most important metric. We measured the sum of the edge-edge crossings on the entire graph (gl-ex) and on each path on the search set (ss-ex).

We chose node-edge crossings as a second factor to compare at our three levels of interest. Node-edge crossings are widely allowed in many layout algorithms, but to our knowledge have not previously been evaluated with user studies. Our qualitative analysis results regarding apparent paths indicate that node-edge crossings might also be important for understanding errors. We measured the sum of the node-edge crossings at each level of interest: on the solution path (sp-nx), on each path in the search set (ss-nx), and globally across the entire graph (gl-nx).

#### 6.2.4 Hypotheses

Our hypotheses were as follows:

- H1. Solution path node-edge crossings (sp-nx) will account for additional variance in performance. We expected that solution-path node-edge crossings would explain variance not accounted for by the other factors of length, continuity, branches, and edge-edge crossings measured on the solution path.
- H2. Search set (ss-) factors will account for additional variance in performance. We expected that search-set factors would explain additional variance beyond the solution-path factors (sp-), because they account for factors on all the paths that a user might search.
- H3. Search set (ss-) factors will predict performance more efficiently than solution-path (sp-) factors. The search set typically overlaps the solution path, so we suspected that search-set factors might predict more with fewer variables.
- H4. *Global factors* (gl-) *will not account for additional variance in performance*. We expected from previous work that global levels would not add additional explanation to what can be explained by the more task-relevant levels.

Our hypotheses are focused on the incremental validity of factors measured on the search set, and on node-edge crossings on the solution path, neither of which had been evaluated by previous research. Although we do not make formal hypotheses about the individual effects of the factors, we expected to see some positive correlation of all factors with both dependent variables. This expectation includes replicating the results of Ware et al. that global edge-edge crossings, and solution-path length, continuity, branches and edge-edge crossings would be positively correlated with response time [35]. We also expected to find significant contributions of the factors studied by Ware et al. when used in regression models.

## 6.3 Results

Descriptive statistics for response time (RT) and error, as well as the predictor variables, are shown in Table 6.1. Upon inspection, the distributions for response time and error were found to have a positive skew, so we performed square root transformations [6] on both variables to improve their distribution.

Descriptive Statistics							
	М	SD	Min	Max			
RT	38.7	22.8	6	85			
Error	4.8	3.7	0	12			
gl-ex	304.6	51.7	195	434			
gl-nx	71.3	14.0	35	108			
ss-ex	388.3	299.2	6	1382			
ss-nx	173.8	138.3	0	684			
sp-ex	14.1	7.5	1	41			
sp-nx	5.7	3.0	0	15			
sp-ln	3.0	.82	2	4			
sp-cn	159.1	94.1	1	422			
sp-br	17.0	3.9	11	26			

**Table 6.1** – Descriptive statistics for predictors and for the dependent variables of response time (RT) and error. Predictors are grouped by level of measurement; those that our study is the first to evaluate are shaded.

Pearson Correlation Coefficients (r)									
	gl-ex	gl-nx	ss-ex	ss-nx	sp-ex	sp-nx	sp-ln	sp-cn	sp-br
RT	009	017	.759*	.723*	.543*	.505*	.808*	.711*	.813*
Error	.033	.042	.678*	.666*	.424*	.476*	.655*	.657*	.646*
* <i>p</i> < 0.01									

**Table 6.2** – Pearson correlation coefficients (r) between predictor variables and the dependent variables of response time (RT) and error. Predictors are grouped by level of measurement; those that our study is the first to evaluate are shaded.

#### 6.3.1 Linear correlations for individual effect

We report on the Pearson correlation coefficients (r) between predictor variables and the dependent variables in Table 6.2. We found significant positive correlations between all predictors and the dependent variables, with the exception of those at the global level.

We were a bit surprised by the lack of any significant individual effect at the global level for edge-edge (gl-ex) and node-edge (gl-ex) crossings, because Ware et al. [35] had measured a small correlation of global edge-edge crossings with response time in their study. (In contrast to this observation about amount of correlation, our hypothesis H4 pertains to the amount of additional variance explained by factors rather than to individual effects.)

Based on previous work, we had expected to see the strong positive correlations of response time with the solution-path factors of length (sp-ln), continuity (sp-cn), branches (sp-br), and edge-edge crossings (sp-ex) that we found. We also found effects of these factors on error, with moderate correlations with error for solution-path edge-edge crossings (r = .424) and node-edge crossings (r = .476), and strong correlations with error for solution-path length (r = .655), continuity (r = .657) and branches (r = .646). In other words, as any of these factors increases in number for a particular graph, so do the average response time and the total number of errors made by participants.

In terms of node-edge crossings, which have not been previously evaluated in user studies, we found strong positive correlations for node-edge crossings on the solution path (sp-nx) and for node-edge crossings on the search set (ss-nx) between both the dependent variables of response time (r = .505 and r = .723 respectively) and error (r = .476 and r = .666 respectively). We also found strong positive correlations between search-set edge-edge crossings (ss-ex) crossings and response time (r = .759) and error (r = .678). These results show that each of these factors is a strong individual predictor of response time and error, with the search-set level factors having a particularly strong effect.

#### 6.3.2 Multicollinearity between factors

We also inspected the correlations between all of the predictor variables (not shown) for multicollinearity (two or more highly correlated predictors). Collinearity between two

predictors prevents us from understanding the degree to which either of the two predictors entered into the model impacts the dependent variables. Choosing to omit some of these predictors allows us to better examine the extent of the contributions of the remaining predictors, but leaves questions surrounding the omitted variables to future work.

We identified two pairs of highly correlated predictors (r > .90) that were cause for concern: search-set edge-edge crossings (ss-ex) correlated with search-set node-edge (ss-nx) crossings, and solution-path length (sp-ln) correlated with solution-path branches (sp-br). We omitted solution set node-edge crossings, because the correlation with each dependent variable was weaker. We suspect that the relationship between solution-path length and branches stems from our graph generation model, so we would not necessarily expect to see it in other types of graphs. We chose to keep solution-path length because previous work suggests that it more commonly accounts for a larger variance in performance than does the number of branches [35].

#### 6.3.3 Hierarchical multiple regression analysis

We constructed two separate hierarchical multiple regression models, one for response times and one for errors, the results of which are shown in Table 6.3. We included all of the predictors that significantly correlated with our dependent variables, but excluded solution-path branches (sp-br) and search-set node-edge (ss-nx) crossings because of multicollinearity. We entered these predictors into each model in three steps<sup>1</sup>. For each regression model we then inspected histograms and probability plots of the residuals to confirm that the assumptions of homoscedasticity (similar variance in the dependent variables) and linearity were met.

We report on the standardized beta coefficients ( $\beta$ ) at each step, which indicate the individual contribution of each predictor to the model – they are measured in standard deviation units and thus are directly comparable measures of importance. We also report on  $R^2$ , a measure of the amount of variation accounted by the predictor(s) included in the

 $<sup>^{1}</sup>$  - The predictors were blocked as follows, with each block entered in a subsequent step: block one contained solution-path length (sp-ln), continuity (sp-cn), and edge-edge crossings (sp-ex), block two contained solution-path node-edge crossings (sp-nx), and block three contained search-set edge-edge crossings (ss-ex).

Standardized Beta Coefficients (β values)							
	RT			Error			
	Step 1	Step 2	Step 3	Step 1	Step 2	Step 3	
sp-In	.487*	.458*	.392*	.303*	.267*	.166	
sp-cn	.359*	.358*	.297*	.441*	.440*	.347*	
sp-ex	.160*	.083	.028	.101	.004	079	
sp-nx		.171*	.096		.217*	.104	
ss-ex			.239*			.364*	
Adj. $R^2$	.745	.764	.781	.533	.563	.603	
$R^2$	.752	.772	.790	.545	.578	.620	
$\Delta R^2$		.020*	.018*		.033*	.042*	
* <i>p</i> < 0.01							

**Table 6.3** – Summary of results from the hierarchical multiple regression analysis of measured factors on response time and error. Predictors that our study is the first to evaluate are shaded.

model at each step, and adjusted  $R^2$ , which takes into account the number of predictors in the model. All significant results were p < 0.01. For those who need additional guidance in understanding the statistics, we recommend Field [6] for an entertaining introduction to interpreting the results of multiple regression analyses.

#### **Response time**

We began by entering solution-path length (sp-ln), continuity (sp-cn), and edge-edge crossings (sp-ex) into the regression model. After this first step, the regression model accounted for 75.2% of the variance ( $R^2 = 0.752$ ). The relative contributions of the three predictors can be further understood by examining their individual beta values, the highest of which came from solution-path length (sp-ln) ( $\beta = 0.487$ ), followed by continuity (sp-cn) ( $\beta = 0.359$ ) and edge-edge crossings (sp-ex) ( $\beta = 0.160$ ). These results replicate the relative importance of these factors found by Ware et al. [35].

In step 2, we added solution-path node-edge crossings (sp-nx) in order to examine the incremental validity of this factor, which had not been evaluated in previous research, over other factors that we measured at the same level. Solution path node-edge crossings (sp-nx) accounted for an additional 2% of the variance ( $R^2 = 0.772$ ,  $\Delta R^2 = 0.020$ ).

In step 3 we added search-set edge-edge crossings (ss-ex) in order to examine the incremental validity of measuring a factor at the search-set level compared to factors

measured at the solution-path level. Search set edge-edge crossings (ss-ex) accounted for an additional 1.8% of the variance ( $R^2 = 0.790$ ,  $\Delta R^2 = 0.018$ ). The final regression model accounted for 79% of the variance in response time, and contains three statistically significant variables: Solution path length (sp-ln) had the highest beta value ( $\beta = 0.392$ ), followed by continuity sp-cn ( $\beta = 0.297$ ) and search-set edge-edge crossings ss-ex ( $\beta =$ 0.239). Solution path node-edge crossings (sp-nx) ( $\beta = 0.096$ ) and solution-path edgeedge crossings (sp-ex) ( $\beta = 0.028$ ) were not significant contributors to the final model.

#### Error

We again began by entering solution-path length (sp-ln), continuity (sp-cn), and edgeedge crossings (sp-ex) into the regression model. After this first step, the model accounted for 54.5% of the variance ( $R^2 = 0.545$ ). Only solution-path length (sp-ln) ( $\beta = 0.303$ ) and continuity (sp-cn) ( $\beta = 0.359$ ) made significant contributions.

We next examined the incremental validity of solution-path node-edge crossings (spnx) and search-set edge-edge crossings (ss-ex), based on the same reasoning used for the response time regression model. Adding solution-path node-edge crossings (sp-nx) in step 2 accounted for an additional 3.3% of the variance ( $R^2 = 0.578$ ,  $\Delta R^2 = 0.033$ ). Adding search-set edge-edge crossings (ss-ex) in step 3 accounted for an additional 4.2% of the variance ( $R^2 = 0.620$ ,  $\Delta R^2 = 0.042$ ).

The final model accounted for 62% of the variance in error. Only search-set edgeedge crossings (ss-ex) ( $\beta = 0.364$ ) and solution-path continuity (sp-cn) ( $\beta = 0.347$ ) were significant contributors to the final model.

#### 6.4 Summary

Our results replicate previous findings in the literature that the factors of path length, continuity, edge-edge crossings, and branches have a significant individual effect on response time when measured on the solution path [35]. We further found significant individual effects for node-edge crossings measured on the solution path, and both node-edge and edge-edge crossings measured on the search set. Through regression modelling we showed that we can predict 79% of the variance in response time using only three predictors: solution-path length is the most important, followed by solution-path continuity, and then search-set edge-edge crossings. Our results from the final step of the

regression model for response time suggest that measuring crossings on the search set has incremental validity over measuring them on just the solution path – search-set edge-edge crossings added only an additional 1.8% to the total variance explained, a small effect, but it also removed the need for solution-path edge-edge and node-edge crossings, making for a more efficient model.

We found that the relative importance of the factors differed quite dramatically for error from what we found for response time. Our results showed that all of the factors we measured on the solution-path and search-set levels had strong individual effects on error. Similar to our results for response time, our results in the final step of the regression model for error suggest that measuring crossings on the search set has incremental validity over the solution path, explaining an additional 4.2%, which is a small effect. The final regression model accounted for 60.3% of the variance in error using only two predictors, with search-set edge-edge crossings being the most important, followed closely by solution-path continuity.

We found some evidence that, at the solution-path level, node-edge crossings may be more important than edge-edge crossings. Adding solution-path node-edge crossings in step 2 of both models had a small effect, explaining an additional 2% of variance in response time, and 3.3% more in error, but it also reduced the contributions of solutionpath edge-edge crossings to insignificant levels. These results suggest that for layouts that allow node-edge crossings, this factor may be the more important of the two to control for at the solution-path level. We were not able to examine the relative effects of nodeedge crossings at the search-set level due to the multicollinearity with search-set edgeedge crossings, but our results about the individual effects of the factor suggest that it may be of similar importance. This conjecture is further evidenced by our observations of the difficulty that apparent paths caused for participants during the study, as discussed in Section 4.3.4.

#### 6.4.1 Summary of hypotheses

All four of the hypotheses were supported, although one was only partially explored because of limitations in our study. We summarize the outcomes for each.

- H1. Solution path node-edge crossings will account for additional variance in performance. Supported. Solution path node-edge crossings explained additional variance for both dependent measures.
- H2. Search set (ss-) factors will account for additional variance in performance.Supported for search-set edge-edge crossings, but we were not able to include search-set node-edge crossings in this analysis so future research will need to be conducted to fully understand the search-set factors.
- H3. Search set (ss-) factors will predict performance more efficiently than solution-path (sp-) factors. Supported. The overlap between the search set and solution path considerably reduced the relative contributions of node-edge and edge-edge crossings measured on the solution path.
- H4. Global factors will not account for additional variance in performance.Supported. We found no significant relationship of node-edge or edge-edge crossings measured globally with either dependent measure.

# **Chapter 7**

# **Discussion and Future Work**

The main goal of the research presented in this thesis was to dig deeper into what makes path tracing in graphs difficult. We did this by characterizing human path-tracing behaviour as a worthwhile pursuit in its own right, and in service of developing a predictive model of the search set. In this chapter, we discuss how our research has addressed our goals, and possible routes for future work. We also discuss the limitations of the user study and the qualitative analysis. We conclude with a summary of our contributions.

# 7.1 Contributions and future work

We discuss two primary contributions of our work, the characterization of human pathtracing behaviours and a predictive behavioural model of a search set based upon them. We also discuss the two secondary contributions, the concept of the search set itself, and the validation of the predictive behavioural model developed through multiple regression analysis of graph readability factors.

### 7.1.1 The characterization and the predictive behavioural model

Our characterization of path-tracing behaviours in graphs extends beyond the previously proposed geodesic tendency [13]. While we did find strong supporting evidence for this tendency, we also found many situations in which it falls short for explaining what people do. We sharpened the description and shortened the term that was used in previous work, where this phenomenon has been called the *geodesic path* tendency. Our discussion emphasizes that it entails following the *closest-to-geodesic branch*. We find this

description more evocative because it emphasizes that a decision is made many times along a path, once for each perceived hop, rather than only once for the entire path.

Our observations revealed a more complex behavioural framework, within which the geodesic tendency plays a major role but can be overridden by other tendencies: the tendency to continue following straight lines, the tendency to avoid directions that point away rather than towards the target, and the tendency to be misled into tracing apparent branches that are not in fact true topological connections. Moreover, a full model of path-tracing behaviour requires understanding when people stop tracing one path in order to try another, and where they begin their next tracing attempt.

Our predictive behavioural model allowed us to predict a set of paths that users were likely to follow to fairly high degree of accuracy (86%). We consider this model to be a good first step because it captures most of the behaviours that we observed in a robust way that avoids over fitting to the training set that we used in our first phase of analysis. We encourage future research on search set models to strive for further breadth, completeness, and accuracy.

#### 7.1.2 The search set concept

A secondary contribution of the research reported in this thesis is the concept of a search set. The search set concept appears to be an apt model for real human behaviour. We have shown it to be predictable when applied to path-tracing tasks. We argue that the idea of a search set promotes analysis of exactly the subset of a graph that is relevant to a particular task, at an intermediate level between completely global and the strictly local single path that is the answer to a specific query.

The concept of a search set could help to explain aspects of human behaviour that have been difficult to unravel thus far. For example, the search set concept provides one possible explanation for the variation in results on global edge-edge crossings found in previous research. Evaluation results for this factor have been very mixed [5,14,17,31]; our own study was one of several to find a lack of effect of global edge-edge crossings on performance. Our conjecture considers the size of the search set in relation to the size of the full graph. In a small graph, a user may search most of the graph in completing a task, so global measurements of factors will heavily overlap with the search set. Our study used slightly larger and denser graphs than have typically been in used in previous work.

For example, Ware et al. used graphs of 42 nodes (compared to our 75-node graphs), and although the number of edges varied, many nodes in their study had only two branches, resulting in a comparatively smaller number of paths that participants could search. In a larger graph, such as the ones we used, we would expect to see less overlap with the search set, which could explain the lack of any significant relationship between the global factors and our dependent variables.

While we focused on one specific application of the search set concept in our research, of how the search set can improve measurement of factors that affect graph readability, we believe that a predicted search set could be useful in many other applications. Future research could explore the design of new interaction techniques to support path-tracing tasks based on the search set concept, or the implementation of new automatic graph layout algorithms for preserving spatial consistency, which could use factors measured on the search set to make subtle improvements to the layout based on a user's task.

#### 7.1.3 Factor measurement for model validation

Although factor measurement was the initial impetus for our investigation, in the end it was relegated to a supporting role in validating our predictive behavioural model. We applied our predictive model of a search set to the problem of factor measurement both as validation that the model itself is a reasonable approximation of the human behaviour we had observed, and as an example of how the technique can be used. We consider the results of the regression analysis to be encouraging evidence that the concept of a search set is on target; indeed, we see a modest quantitative improvement for even this first attempt at a predictive model.

Our findings pertain specifically to path-tracing tasks in graphs, but real-world uses of graph visualizations also involve other abstract tasks. It would be useful to understand how the relative importance of the factors we examined in our study differs for different types of tasks. Future research could also seek to explore whether or not the incremental improvements seen by extending measurement of edge-edge crossings from the solution path to the search set also hold true of other factors. The differences we found in our analysis between how the various factors influence response time and error further leads us to believe that there is no single factor that most impacts graph readability, and that we should seek to understand a factor's priority or importance in a specific context. This is not a new idea, but it has received limited practical attention. An exception comes from Eades et al., who showed that making compromises between factors based on their relative importance can be utilized to create to better graph layouts [12]. Future research should continue to examine how factors might be traded off to provide the best support for particular user tasks or priorities when implementing layouts and techniques.

#### 7.1.4 Regression versus ANOVA for factor characterization

We echo and emphasize the call of Ware et al. [35] for the benefits of regression analysis over simply testing for statistical significance with methods such as analysis of variance. Although Ware et al. published their work over a decade ago, to the best of our knowledge only one subsequent paper in the visualization literature has followed the recommendation to adopt this two-step methodology of linear correlation analysis followed by multiple regression analysis [17]. Analysis of the linear correlations allows us to account for the individual effects of factors, while the second step of multiple regression analysis allows us to analyse the relative importance of the factors, for a much richer and deeper understanding of inter-factor relationships. We believe that this style of analysis can and should become increasingly important in the visualization literature, because it is a very good match for the long-term goal of fully characterizing both techniques and algorithms. Specifically, untangling the relationships between factors will help characterize the algorithms that use these factors, and it will also help develop guidelines of how to map between algorithms and the requirements of specific visual encoding and interaction techniques [26].

A small methodological contribution in this thesis is that we advocate for hierarchical rather than stepwise multiple regression, based on recommendations from the clinical psychology literature on incremental validity [19]. Although the term *incremental* typically has negative connotations in discussions of research contributions, we note that here it is being used in a specific technical sense of determining whether useful information has been added beyond what is already available.

### 7.2 Limitations

Our study design and our analytic approach were necessarily limited by the time allowed for a master's thesis. We balanced precision and completeness against the normal amount of time expected for completing the research appropriate for a degree.

While we found the recorded path-tracing data from the Cintiq tablet to be quite rich, and sufficient for our study, we know that it did not capture the complete picture of what users were doing. Our logged node hover data was not without noise, and surely some of the noise can be attributed to instances where users visually examined nodes but forgot to point at them with the pen. We chose not to use eye tracking in our study primarily because of its high overhead for analysis. A reasonable follow-up could combine the tablet approach with eye tracking, in order to examine how well pointing and eyes match up, and to potentially capture some of what the tablet misses. Tools to automatically compare or correlate the node hover data with eye tracking data might be of significant help in reducing the human time required to fully utilize eye tracking in this type of research.

Our qualitative analysis used only one investigator for the coding, which limits the reliability of our findings. Coding analysis always involves a degree of subjectivity. We would expect that a different investigator might describe some of the path-tracing behaviours that we identified in a different way, or even identify other behaviours that we did not. A useful follow-up analysis would employ additional coders to examine the reliability of the original investigator's codes, and potentially expand upon our findings.

Finally, and related to the previous point, our qualitative analysis and our characterization of behaviours was subject to the inherent limitations of human investigation. Quantitative or computation-based methods such as machine learning might potentially pick up on patterns that human judgement cannot, and thus could be a promising avenue for future research to explore.

## 7.3 Conclusion

In this thesis, we proposed the concept of the search set: the subset of the graph that is likely to be carefully investigated by a user in carrying out a path-tracing task in a graph. The two primary contributions of the thesis are a characterization of human behaviour in using visual representations of graphs for path tracing, and an initial predictive behavioural model of the new concept of a search set that is based on these observed behaviours. We validated the search set model by measuring graph readability factors on this set, in comparison with measuring them globally on the entire graph or very locally on only the single path that is the correct solution. The success of this approach in modestly improving predictions of response time and error, despite our model being only a first attempt at algorithmic instantiation of complex human behaviour, is encouraging evidence that the concept of a search set has merit. A secondary contribution of the thesis is the careful comparison of the relative importance of factors measured at these three levels of a graph through multiple regression analysis. We found key differences in the relative weighting of the importance of the factors that affect response time versus error, which further illustrates the explanatory power of regression analysis

# **Bibliography**

- 1. Auber, D., Chiricota, Y., Jourdan, F., & Melancon, G. Multiscale visualization of small world networks. *Proc. of Information Visualization (InfoVis)* 75 84, 2003.
- 2. Bishop, C.M. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, New York. 2006.
- 3. Brehmer, M. and Munzner, T. A Multi-Level Typology of Abstract Visualization Tasks. *To Appear in IEEE Trans. on Visualization and Computer Graphics (Proc. InfoVis)*, 2013.
- 4. Cohen, J. Statistical Power Analysis for the Behavioral Sciences. L. Erlbaum Associates. 1988.
- 5. Dwyer, T., Lee, B., Fisher, D., et al. A Comparison of User-Generated and Automatic Graph Layouts. *IEEE Trans. on Visualization and Computer Graphics* (*Proc. InfoVis*) 15(6):961–968, 2009.
- 6. Field, A. *Discovering Statistics Using SPSS*. SAGE Publications. 2009.
- 7. Ghoniem, M., Fekete, J.-D., and Castagliola, P. A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations. *Proc. IEEE Symposium on Information Visualization (InfoVIs)*, pages 17–24, 2004.
- 8. Ham, F. van and Rogowitz, B. Perceptual Organization in User-Generated Graph Layouts. *IEEE Trans. on Visualization and Computer Graphics (Proc. InfoVis)* 14(6):1333–1339, 2008.
- 9. Haynes, S.N. and Lench, H.C. Incremental Validity of New Clinical Assessment Measures. *Psychological Assessment* 15(4):456–466, 2003.
- 10. Heer, J., Card, S., and Landay, J. Prefuse: A Toolkit for Interactive Information Visualization. *Proc. SIGCHI Conference on Human Factors in Computing Systems* (*CHI*), pages 421–430, 2005.
- 11. Himsolt, M. Comparing and Evaluating Layout Algorithms within GraphEd. Journal of Visual Languages & Computing 6:255–273, 1995.
- 12. Huang, W., Eades, P., Hong, S.-H., and Lin, C.-C. Improving multiple aesthetics produces better graph drawings. *Journal of Visual Languages & Computing 1–11*, 2012.
- 13. Huang, W., Eades, P., and Hong, S. A Graph Reading Behavior: Geodesic-Path Tendency. *Proc. Pacific Visualization Symposium*, pages 137–144, 2009.
- 14. Huang, W., Hong, S., and Eades, P. Layout Effects on Sociogram Perception. *Proc. Graph Drawing (GD)*, pages 262–273, 2005.
- 15. Huang, W., Hong, S., and Eades, P. Predicting Graph Reading Performance: A Cognitive Approach. *Proc. Asia-Pacific Symposium on Information Visualisation* (*APVis*), pages 207–216, 2006.
- 16. Huang, W., Hong, S., and Eades, P. Effects of Crossing Angles. *Proc. Pacific Visualization Symposium*, pages 41–46, 2008.
- 17. Huang, W. and Huang, M. Exploring the Relative Importance of Number of Edge Crossings and Size of Crossing Angle: A Quantitative Perspective. *International Journal of Advanced Intelligence* 3(1):25–42, 2011.
- Huang, W. Establishing Aesthetics Based on Human Graph Reading Behavior: Two Eye Tracking Studies. *Personal and Ubiquitous Computing* 17(1):93–105, 2013.
- 19. Hunsley, J. and Meyer, G.J. The Incremental Validity of Psychological Testing and Assessment: Conceptual, Methodological, and Statistical issues. *Psychological assessment* 15(4):446–455, 2003.
- 20. Kim, S.-H., Dong, Z., Xian, H., Upatising, B., and Yi, J.S. Does an Eye Tracker Tell the Truth about Visualizations?: Findings while Investigating Visualizations for Decision Making. *IEEE Trans. on Visualization and Computer Graphics* (*Proc. InfoVis*) 18(12):2421–2430, 2012.
- 21. Körner, C. Sequential Processing in Comprehension of Hierarchical Graphs. *Applied Cognitive Psychology 18(4):467–480*, 2004.
- 22. Körner, C. Eye Movements Reveal Distinct Search and Reasoning Processes in Comprehension of Complex Graphs. *Applied Cognitive Psychology* 25(6):893–905, 2011.

- 23. Lee, B., Plaisant, C., Parr, C.S., Fekete, J.-D., and Henry, N. Task Taxonomy for Graph Visualization. *Proc. Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization (BELIV)*, pages 1–5, 2006.
- 24. Marriott, K., Purchase, H., Wybrow, M., and Goncu, C. Memorability of Visual Features in Network Diagrams. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis)* 18(12):2477–2485, 2012.
- 25. Melancon, G. Just How Dense Are Dense Graphs in the Real World? *Proc. Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization (BELIV)*, pages 1–7, 2006.
- 26. Meyer, M., Sedlmair, M., and Munzner, T. The Four-Level Nested Model Revisited: Blocks and Guidelines. *Proc. Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization (BELIV)*, 2012.
- 27. Purchase, H., Hoggan, E., and Görg, C. How Important is the "Mental Map"? An Empirical Investigation of a Dynamic Graph Layout Algorithm. *Proc. of Graph Drawing (GD)*, pages 184–195, 2006.
- 28. Purchase, H. Which Aesthetic has the Greatest Effect on Human Understanding? *Proc. of Graph Drawing (GD)*, pages 248–261, 1997.
- 29. Purchase, H. Performance of Layout Algorithms: Comprehension, not Computation. *Journal of Visual Languages & Computing 9(6):647–657*, 1998.
- 30. Purchase, H.C., Allder, J., and Carrington, D. Graph Layout Aesthetics in UML Diagrams: User Preferences. *Journal of Graph Algorithms and Applications* 6(3):255–279, 2002.
- 31. Purchase, H.C., Cohen, R.F., and James, M.I. Validating Graph Drawing Aesthetics. *Proc. of Graph Drawing (GD)*, pages 435–446, 1995.
- 32. Purchase, H.C., Pilcher, C., and Plimmer, B. Graph Drawing Aesthetics: Created by Users not Algorithms. *IEEE Trans. on Visualization and Computer Graphics* (*Proc. InfoVis*) 18(1):81–92, 2012.
- 33. Purchase, H.C., Plimmer, B., Baker, R., and Pilcher, C. Graph Drawing Aesthetics in User-Sketched Graph Layouts. *Proc. of the Australasian Conference on User Interface (AUIC)*, pages 80–88, 2010.
- 34. Vicente, K. and Torenvliet, G.L. The Earth is Spherical(p < 0.05): Alternative Methods of Statistical Inference. *Theoretical Issues in Ergonomics Science* 1(3):248-271,2000.

- 35. Ware, C., Purchase, H., Colpoys, L., and McGill, M. Cognitive Measurements of Graph Aesthetics. *Information Visualization 1(2):103–110*, 2002.
- 36. Ware, C. *Information Visualization: Perception for Design*. Morgan Kaufman, San Francisco. 2000.
- 37. Watts, D.J. and Strogatz, S.H. Collective Dynamics of 'Small-World' Networks. *Nature 393(6684):440–442*, 1998.

## Appendix A

# **User Study Materials**

This appendix contains materials from the user study presented in Chapter 3.

## A.1 Recruitment poster

The following study recruitment poster was poster across the UBC campus. The text from the poster was included in a website, along with a calendar showing available study time slots, and in an email sent to the CS Department graduate student mailing list.

Version: Nov 14, 2012

#### THE UNIVERSITY OF BRITISH COLUMBIA

Department of Computer Science University of British Columbia Vancouver, BC, V6T 1Z4



## Call for Participation: Human Factors in Visualization Study

## **Starting November 2012**

## Pay: \$10/hour (1 – 1.5 hours per session; 2 sessions total)

You are invited to participate in a 2 session study on visual factors that impact visualization use, being run by the Department of Computer Science at the University of British Columbia.

We are looking for participants who:

- Are 19 years of age or older
- Speak English fluently
- Have normal/corrected-to-normal vision
- Have no colour vision deficiencies

#### What is involved?

You will be asked participate in two sessions within a 2-week period. During each session, you will be asked to perform several tasks using a visualization on a desktop computer, and we will record your speed and accuracy. You will also be asked to complete a questionnaire and we may interview you about your experiences with the experimental system. You will receive an additional \$5 bonus for participating in both sessions.

#### Interested in participating?

If vou are interested in participating, please visit our website at

or send e-mail to

## A.2 Consent form

Participants were required to sign the following consent form before participating in the user study. Participants were emailed a copy of the consent form after they had scheduled their first session.



Participant Consent Form

#### Human Factors in Information Visualization

You are being invited to participate in a study entitled "Human Factors in Information Visualization" that is being conducted by Jessica Dawson, Tamara Munzner and Joanna McGrenere.

Jessica Dawson is an MSc student in the department of Computer Science at the University of British Columbia. You may contact her if you have further questions by email at

#### **Study Purpose and Procedures**

The purpose of this research project is to investigate how people analyze complex data, and how statistical analysis techniques and visual representations are used to make sense of this data. Research of this type is important because it allows us to design better data displays to allow more effective, efficient, and enjoyable analysis of data in a variety of applications.

If you agree to voluntarily participate in this research, your participation may include:

- Participating in a verbal interview.
- Filling out a background questionnaire that may ask about your experience with computer technology and data analysis applications, as well as personal characteristics such as age and gender.
- Completing computer-based tasks.
- Filling out a questionnaire about the computer-based tasks and tools you experienced.
- Being audio-taped.
- Being video-taped
- Being watched by live observers.
- Using software with automatic data logging techniques (e.g., screen recording, mouse recording).

You will be asked to participate in 2 sessions, which will be scheduled within a 2-week period. Each session is expected to take approximately 1-1.5 hours and will take place at UBC Computer Science Department. There are no known or anticipated risks to you by participating in this research.

This research is funded by NSERC. Portions of this research may be used in published articles, conference presentations, theses, and internet project descriptions. Raw data related to your performance and preference on computer-based tasks may be made publically available for use by the research community.

#### Confidentiality

The identities of all people who participate will remain anonymous and will be kept confidential. Data will be kept in a locked metal filing cabinet in the offices of the principal and/or co-investigators or as password-protected computer files. All data from individual participants will be coded so that their anonymity will be protected in any reports, papers, presentations and publically available raw data that result from this work. Data made publically available will NOT include any video recording, audio recordings or other information that could identify you.

**Remuneration/Compensation** 

Page 1 of 2

Version: Nov 14, 2012

For each session, you will receive an honorarium of \$10 per 1 hour of participation. In addition, if you participate in both sessions you will receive a \$5 bonus.

#### Consent

Your participation in this research is completely voluntary. If you do decide to participate, you may withdraw at any time without any consequences or any explanation. If you do withdraw, we will ask whether we may use your data for data analysis. If you decline, your data will be destroyed.

In addition to being able to contact the researcher at the above email address, you may verify the ethical approval of this study, or raise any concerns you might have, by contacting the University of British Columbia Office of Research Services (ORS) at 604-822-8581.

Your signature below indicates that you understand the above conditions of participation in this study and that you have had the opportunity to have your questions answered by the researchers.

Name of Participant

Signature

Date

A copy of this consent will be left with you, and a copy will be taken by the researcher.

Page 2 of 2

Version: Nov 14, 2012

## A.3 Experimenter script

To ensure consistency, the experimenter read from the following script when conducting all user study sessions.

#### **Metrics Study 1 Script**

#### **Materials Checklist**

- 2 pens Setup Printout of Subject Package Presentation Order List
- Laptop setup with Cintiq tablet
- Example image for demonstration

#### **Before Experiment**

- 1. Disconnect computer from network (including wireless).
- 2. Make sure the DOCK isn't at the bottom of the tablet screen
- 3. Setup video camera, and check memory space
- 4. Run the study jar, Study1\_Experiment.jar (in Users\Jessica\Workspace\graphmetrics\_WC)
- 5. Select 'Run Script', and choose the script specified in the presentation order list
- 6. Enter subject ID when prompted

#### **Immediately After Experiment**

- 1. Elaborate on notes made during experiment
- 2. Create a session folder in the Archive folder, and copy the log files into it (from Users/Jessica/Workspace/graphmetrics WC/Study1 LogFiles).
- 3. Email copy of log files to self.
- 4. Upload video files to computer

#### **Experimenter Script (PART 1)**

#### 1. Introduction and consent form:

- Thank you for coming in today. Today you will be doing some tasks with node-link graphs using a Cintiq tablet. The first part of the study will take place today, and should take about 1.5 hours.
- Before we get going, I have a **consent form** for you to look over and sign if you agree to participate. You'll notice that we have a video camera set up. We will be recording video of your hands and the tablet screen during the study, but not your face.
- If at any point during the study you feel that you do not wish to proceed, just let me know and we will end it.

Graph Readability Metrics Projects - Nov 1, 2012

Give consent form, and allow them to take as much time as they wish with it, and only proceed with the study if they sign it. Also, offer the second copy of the consent form to them if they wish to keep it.

• Thanks, Now we're going to get started with the tasks. Because we're running so many participants and we need to make sure we give everyone the same instructions, I'm going to read from a script sometimes.

#### 2. Background questionnaire:

• To start, we're asking all participants to provide us with some background information, which will help us understand our results. Could you please complete this questionnaire?

Give background questionnaire.

#### 3. Introduction and Practice

Intro and practice.

- You will be doing a path-tracing task on node-link graphs using this tablet. The tasks will start with training, so you'll have an opportunity to learn how to do the task and ask me any questions.
- Now I'll introduce you to the tablet. The graphs will be displayed on the screen, and you'll be able to interact with graphs using this pen.
- Let's start by calibrating the tablet input.
  - To CLICK with the tablet pen, you need to hold down or press this button on the side and actually press the pen tip to the screen.

Set the input to the participants correct hand (left or right). Run the tablet calibration utility until you AND the participant are happy with it.

- There are a couple thing you should know about how the tablet works: 1) You can rest your hand on the tablet screen without interfering with the pen input, so you can use your other hand to point to the screen or interact in other ways that you would like; 2) the computer can track the pen when the pen is touching the screen and when it is slightly above it; 3) the on screen cursor shows you feedback of where the pen cursor is pointing.
- Now I'll explain the task with a demonstration of the tablet.

Direction attention to the tablet with demo graph displayed (Explain the task as you go):

- Are you familiar with node link graphs? If not, explain briefly.
- The primary goal of this task is to find the shortest path from the RED node to the BLUE node as quickly and accurately as possible. It will always be 2,3 or 4 hops.
- A hop is what we call the distance from one node to the next along a connecting edge. *Demonstrate tracing a path with tablet pen.*

Graph Readability Metrics Projects - Nov 1, 2012

	Sometimes you will see nodes on top of edges that they aren't actually connected to. These nodes will have a small white halo around them. Only edges that go through the halo and actually touch the node are connected. <i>Point out an example.</i>					
•	EDGES tips:					
•	The edges always go in straight lines					
	• You'll notice that when an edge is attach to a node, it always goes towards the center of the node – it doesn't attach at the side <i>Point out an example</i> .					
	• Sometimes when it's cluttered, it's easier to tell if a node is attached if you look at both sides of the node – if you can see a straight edge coming from a halo on the other side, that's a clue. <i>Point out an example</i> .					
•	The secondary goal of this task is to use this tablet pen to actively trace the paths you're searching by pointing to the nodes you are looking at as you try to determine the answer.					
	0 Demonstrate.					
	• You won't be able to show me everything you glance at, but you should try to point out things you're actively searching.					
Ex	plain Tablet tracing to participant on the demo mode					
•	When you hover over a node with the tablet pen, it will be highlighted in ORANGE. Do your best to use the pen to mimic where you're looking and searching. When you're focusing on something, you want it to turn green (even if just for a second).					
•	You're looking for the shortest path between the RED node and the BLUE node. It will always be 2, 3 or 4 hops.					
•	Once you've found the answer, press the 'FOUND IT' button. You'll only have 90 seconds to find the answer (point to top where count down will be).					
•	Once you press the found it button, you then need to demonstrate the path you found. You'll know that you're in this mode because the RED and BLUE nodes will be surrounded in these green halos. The prompt at the top will also change, and the color of the nodes when you hover will also change to green.					
•	To do this, hover over on EACH NODE in the path (except the RED and BLUE nodes) and CLICK the node. Remember that to click you need to press the button on the pen and actually press down against the tablet. If you click is successfully, the node will change to a darker GREEN and stay highlighted. Click again to deselect the node if you make a mistake. When you've highlighted the path, press OK.					
	If you realize that you don't actually know the answer, please select the nodes that you thought formed the path. Then press the OK button to move on.					
•						
•	Once you've finished the trial, you'll be shown the answer.					
•	Once you've finished the trial, you'll be shown the answer. If you need to take a break, please try to wait until you have completed the current block of trials.					

• Do you have any questions? Are you ready to try it out?

#### Start the training

• We will start with 6 training trials now. These trials will be just like the real trials – this is a good opportunity for you to ask questions about anything that isn't clear, and to get used to the task.

#### 4. Experimental trials

After practice, but before block 1.

- We're ready to start the experiment. You will complete 6 blocks of trials. There will be a break between each block of trials, with a longer break about halfway through the experiment.
- Again, I want to stress that the main goal of the task is to go as quickly and accurately as possible. It's very important that you trace where you're looking and what you're focusing on with the tablet pen as best you can.

#### At halfway block

• You can take a break from the computer for a few minutes.

#### 5. Final Questionnaire / Debriefing:

To finish things up here is a final questionnaire.

Graph Readability Metrics Projects - Nov 1, 2012

#### **Experimenter Script (PART 2)**

#### 1. Introduction and consent form:

- Thank you for coming in for the second part of the study. Today you will be more of the same tasks using the tablet as last time. This session should take about 1 hour.
- Before we get going, I have a copy of the **consent form** that you signed last time. If you would like to look over it you can do that now. Remember that if at any point during the study you feel that you do not wish to proceed, just let me know and we will end it.
- Thanks, Now we're going to get started with the trials.

#### 2. Introduction and Practice

#### Intro and practice.

- To CLICK with the tablet pen, you need to hold down or press this button on the side and actually press the pen tip to the screen.
- Let's start by calibrating the tablet input.
  - Set the input to the participants correct hand (left or right). Run the tablet calibration utility until you AND the participant are happy with it.
- A reminder about how the tablet works: 1) You can rest your hand on the tablet screen without interfering with the pen input, so you can use your other hand to point to the screen or interact in other ways that you would like; 2) the computer can track the pen when the pen is touching the screen and when it is slightly above it; 3) the on screen cursor shows you feedback of where the pen cursor is pointing.

#### Show demonstration graph

- Now lets go through the demo again for a brief reminder of the study protocol.
- The primary goal of this task is to find the shortest path from the RED node to the BLUE node as quickly and accurately as possible. It will always be 2,3 or 4 hops.
- A hop is what we call the distance from one node to the next along a connecting edge.
- Keep in mind that some nodes which aren't actually connected to an edge will have these halo around them.
- EDGES tips:
- The edges always go in straight lines:
  - You'll notice that when an edge is attach to a node, it always goes towards the center of the node it doesn't attach at the side *Point out an example*.
  - Sometimes when it's cluttered, it's easier to tell if a node is attached if you look at both sides of the node if you can see a straight edge coming from a halo on the

Graph Readability Metrics Projects - Nov 1, 2012

#### other side, that's a clue. Point out an example.

#### Introduce participant to the tablet

- The secondary goal of this task is to use this tablet pen to actively trace the paths you're searching by pointing to the nodes you are looking at as you try to determine the answer. When you hover over a node with the tablet pen, it will be highlighted in ORANGE. Do your best to use the pen to mimic where you're looking and searching. When you're focused on something, try to make sure it's actually highlighted.
- Once you've found the answer, press the 'FOUND IT' button. You'll only have 90 seconds to find the answer (point to top where count down will be).
- Once you press the found it button, you then need to demonstrate the path you found.
- To do this, hover over on EACH NODE in the path (except the RED and BLUE nodes) and CLICK the node. Remember that to click you need to press the button on the pen and actually press down against the tablet. If you click is successfully, the node will change to a darker GREEN and stay highlighted. Click again to deselected the node if you make a mistake. When you've highlighted the path, press OK.
- If you realize that you don't actually know the answer, please select the nodes that you thought formed the path. Then press the OK button to move on.
- Once you've finished the trial, you'll be shown the answer.
- If you need to take a break, please try to wait until you have completed the current block of trials.
- **Do you have any questions?** Are you ready to do the study?

#### Start the training

• You'll do just 3 training trials now, to get back up to speed.

#### 3. Experimental trials

After practice, but before block 1.

- We're ready to start the experiment. You will complete 6 blocks of trials. There will be a break between each block of trials, with a longer break about halfway through the experiment. We will finish up with a short interview.
- Again, I want to stress that the main goal of the task is to go as quickly and accurately as possible. It's very important that you trace where you're looking and what you're focusing on with the tablet pen as best you can.

#### At halfway block

• You can take a break from the computer for a few minutes.

Graph Readability Metrics Projects - Nov 1, 2012

#### 4. Final Questionnaire / Debriefing:

To finish things up here is the same questionnaire from last time.

Lastly, I just want to ask you a few questions about your experience doing the study.

Graph Readability Metrics Projects - Nov 1, 2012

## A.4 Background questionnaire

Participants were asked to complete the following background questionnaire during the first session of the user study, immediately after signing the consent form and agreeing to participate in the experiment.

Participant # Part
Background
Do you have normal/corrected-to-normal vision?  Yes No
Do you have a color vision deficiency?  Yes  No
What is your gender?  Male  Female
What is your age?
Currently, I am
oUndergraduate student, in the department of
oGraduate student, in the department of
oPhD student, in the department of
oUniv. Faculty, in the department of
oEmployee at
oother:
What is your computer expertise?  Novice  Intermediate  Expert
On an average day (or week), approximately how many hours do you spend using a computer §?
hours per Day Week
Page 1 of 4

## A.5 Post-session questionnaire

Participants were asked to complete the following questionnaire twice, once at the end of each of the study sessions.

### Participant # \_\_\_\_\_ Part \_\_\_\_\_

Rate the study task according to the following criteria:

Overall Difficulty	Very easy	1	2	3	4	5	6	7	Very difficult
Mental Effort	Low effort	1	2	3	4	5	6	7	High effort
Physical Effort	Low effort	1	2	3	4	5	6	7	High effort

## A.6 End of study interview questions

At the end of the second user study session, the experimenter asked the participant the following set of questions.

Interview Questions

1. Did you use any strategies to help you trace the paths? These could be strategies that worked well, or strategies you tried but didn't find useful.

- Can you describe the strategy/strategies? What made it effective/not effective?

- 2. How well did you feel you were able to show where you were searching using the pen?
- 3. Were there any features or other things that you noticed during the tasks that made the trials harder or easier?

## A.7 Task reference sheet

The following reference sheet for the study task was taped down on the table next to the Cintiq tablet during the user study. It contains a brief description of the three main components of the study task.

#### **Task Reference Sheet**

1) Find the shortest path from the RED node to the BLUE node. Path is always 2, 3 or 4 hops

Press 'FOUND IT' when you find path

#### 2) Demonstrate your path by CLICKING each node in the path

Press 'OK' when you've selected the path

#### 3) View the answer

Press OK to move on to next trial

## **Appendix B**

# Visualizations for Qualitative Analysis

This appendix contains examples of the visualizations used for the qualitative analysis presented in Chapter 4. All of the visualizations were programmed in Java using the Prefuse toolkit [10].

### **B.1** Exploratory visualization for preliminary analysis (Version 1)

This visualization was the first that we developed to support exploration of the node hover data for the preliminary node-based analysis presented in Section 4.1.

One static image was generated per graph trial. An example of one of these images is shown in Figure B.1. The visualization displayed the node hovers from all participants for one graph trial, aggregated onto a single graph image. The size of a node encodes the total number of times it was hovered over by all participants; in other words, a node would become larger if multiple participants hovered over it, and/or if one participant revisited the node repeatedly. The graph id is shown in a label on the top left. Grey-scale encodes the number of participants that hovered over a particular node at least once; white nodes were not hovered over by any users. The source and goal nodes are indicated through the colour of their outlines, red and blue respectively.



Figure B.1 – Screen shot of version 1 of the exploratory visualization used in the node-based analysis.

## **B.2** Exploratory visualization for preliminary analysis (Version 2)

We developed a second visualization to support the preliminary node-based analysis presented in Section 4.1. This version improves over version 1 by supporting interaction and providing multiple views of the data.

The visualization displayed the node hovers from all participants for one graph trial at a time. It consisted of an aggregate view and a series of small multiples showing the data from each participant. During the preliminary analysis we tried three different ways of presenting the aggregate view: versions 2.1 and 2.2 were primarily used for exploration of the data, while version 2.3 was used for confirmation of the convex hull pattern. The following appendices in this section detail these iterations.

The visualization supported navigation between graphs using the arrow keys on the keyboard. The investigator could jump to a particular graph by pressing the space bar, and then entering the graph id when prompted. When the investigator hovered over a node, it was highlighted in every small-multiple view and its id was shown in a tooltip.

A screenshot of the visualization as it appeared on the investigator's monitor is shown in Figure B.2. The aggregate view is on the left (version 2.1 shown), and the small multiples view is shown on the right.

.



Figure B.2 – Screen shot of version 2 of the exploratory visualization used in the preliminary node-based analysis.

## **B.2.1 Aggregate view (Version 2.1)**

The first version of the aggregate view used grey-scale to encode the number of participants that hovered over a particular node at least once, as in version 1 of the exploratory visualization discussed in Appendix B.1. The label on the top left shows the graph id. Figure B.3 shows a close-up screenshot of this view.



Figure B.3 – Screen shot of Version 2.1 of the aggregate view

#### **B.2.2 Aggregate view (Version 2.2)**

The second version of the aggregate view used grey-scale to encode the frequency of node hovers across all participants. We made this change from the previous version of this view (version 2.1 shown in Appendix B.2.1) because we thought that the frequency of hovers was more interesting than whether or not participants just visited a node, as the hover frequency says more about a node's importance to the task. The label on the top left shows the graph id as in the previous version. An example of this version of the view is shown in Figure B.4.

In the very first visualization we built we used a size encoding for frequency, as in Figure B.1. However, the distortion caused by the change in size made the graph more difficult to analyze, and so we changed to using grey-scale for this value.

In the end, the difference between this version and version 2.1 (Appendix B.2.1) was subtle for many of the graphs, but allowed us to identify additional areas of the graph that were heavily used. For example, in Figure B.4 we note that a couple of the nodes on the right hand side of the graph, including a 1-hop neighbour of the red node, are dark and were therefore hovered frequently. More frequently hovered over nodes suggests that participants who searched in this area spent quite a bit of time. By comparison, version 2.1 shown in Figure B.3 does not capture the importance of these particular nodes because only a couple of participants hovered them.



Figure B.4 – Screen shot of Version 2.2 of the aggregate view

### **B.2.3** Aggregate view with convex hull (Version 2.3)

We created the third and final version of the aggregate view much later, after we had completed the bulk of the preliminary analysis. The view shows the hover frequency data, encoded the same way as in version 2.2 (Appendix B.2.2). The only change in this version is the addition of a visual representation of the convex hull around the one-hop neighbours of the source (red) and goal (blue) nodes, which is shaded in green. A close-up screenshot of this view is show in Figure B.5.

This visualization was primarily used for visual confirmation of the convex hull behaviour, which we had detected using the previous versions of the visualizations. We discuss the convex hull behaviour in detail in Section 4.1.



Figure B.5 – Screen shot of Version 2.3 of the aggregate view

#### **B.2.4 Small multiples**

Figure B.6 shows a close up of the small multiples used in the second version of the exploratory visualization. Each individual small multiple displays the hovers from one participant onto a graph image of the trial being visualized. The participant id is displayed in a label at the top left of each small multiple. Nodes that the participant hovered over at least once are coloured orange, while white nodes were not hovered over.

We found the hover frequency data displayed in version 2 of the aggregate view (Appendix B.2.2) to be sufficient for our exploration during the preliminary node-based analysis, and thus did not make a frequency version of the small multiples.



Figure B.6 – The small multiples view used in the second version of the exploratory visualization for the node-based analysis

### **B.3** Visualization for qualitative analysis of path-tracing behaviours

The following appendix provides more detail on the visualization developed to support the qualitative analysis of path-tracing behaviours presented in Section 4.2.

Figure B.7 shows a screenshot of the entire visualization used in the qualitative analysis as it appeared on the investigator's monitor. The visualization displays the data from one participant trial at a time.

The small multiples on the right visualized a maximum of 20 steps. These were the primary views used by the investigator in performing the qualitative coding. The hovered node(s) in each step are coloured orange. As discussed in Section 4.2.2, the first node in a collapsed topological sequence is coloured light orange, with subsequent nodes coloured dark orange, and edges along the topological path between them also coloured orange.

The aggregate view on the left was similar as that used in version 2 of the exploratory visualization for the preliminary node-based analysis (Appendix B.2.2). For the qualitative analysis, the aggregate view provided a useful overview by showing the nodes that the participant hovered over at least once, which are coloured in black. As additional support, when the investigator hovered over a node, it was highlighted in every small-multiple view and its id was shown in a tooltip. The graph id and participant id for the trial is shown in a label at the top left of the aggregate view, while the small multiples are labeled with the steps that they show. The small multiples view is described in detail in Section 4.2.2.

The visualization also supported navigation between trials using the arrow keys on the keyboard: up/down arrow keys moved between participants, while left/right arrow keys moved between graphs. The investigator could jump to a particular graph by pressing the space bar, and then entering the graph id when prompted.


Figure B.7 – Screenshot of the visualization used for the qualitative analysis of path-tracing behaviours.

## B.3.1 Steps 13 – 20 for example of a coded trial

Figure B.8 shows the remaining steps 13 - 20 for the example of a coded trial discussed in Section 4.2.4.



Figure B.8 – Steps 13 – 20 for the example of a coded trial discussed in Section 4.2.4