Analyzing High-dimensional Multivariate Network Links with Integrated Anomaly Detection, Highlighting and Exploration

Sungahnn Ko* Purdue University
Shehzad Afzal* Purdue University
Simon Walton† Oxford University
Yang Yang* Purdue University
Junghoon Chae* Purdue University
Abish Malik* Purdue University
Yun Jang‡ Sejong University
Min Chen† Oxford University
David Ebert* Purdue University

ABSTRACT

This paper focuses on the integration of a family of visual analytics techniques for analyzing high-dimensional, multivariate network data that features spatial and temporal information, network connections, and a variety of other categorical and numerical data types. Such data types are commonly encountered in transportation, shipping, and logistics industries. Due to the scale and complexity of the data, it is essential to integrate techniques for data analysis, visualization, and exploration. We present new visual representations, *Petal* and *Thread*, to effectively present many-to-many network data including multi-attribute vectors. In addition, we deploy an information-theoretic model for anomaly detection across varying dimensions, displaying highlighted anomalies in a visually consistent manner, as well as supporting a managed process of exploration. Lastly, we evaluate the proposed methodology through data exploration and an empirical study.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques; I.3.8 [Computer Graphics]: Applications—Visual Analytics

1 INTRODUCTION

The recent trend of increasing size, complexity, and variety in datasets (e.g., spatial, temporal, quantitative, qualitative, network data) makes analysis and decisions from these data more challenging, often called the *big data* problem [24, 34, 40]. One very challenging type of big data is multivariate network data, especially when there are multivariate values for both nodes and links. For example, transportation, shipping, logistics, commerce, trading, electricity and communication industries [8, 46] have many connected operational locations where multiple variables describe each location's operations. With flight delay network data, various multivariate operational aspects are considered simultaneously: types of delay, patterns based on airport location, trends in time, and relationships among the airports. To reduce the analysts' information overload and to enable effective planning, analysis and decision making, an interactive visual exploration and analysis environment is needed as traditional machine learning and big data analytics alone can be insufficient [10].

While various systems and techniques for network visualization have been proposed [22], few support analyzing both multivariate network data (e.g., [43] and [28]) and map-based spatial network data (e.g., [19] and [8]). There still remains a gap in effective multivariate spatial network data exploration and analysis to efficiently answer challenging questions such as the following: What are the patterns in multivariate variables on a node or among node-node pairs? Are the patterns relevant to specific regions and times? Is there any seasonality in the patterns? Can we verify the patterns on a map? Which network nodes and links could be anomalous?

In this work, we fill this gap by integrating a family of visual analytics techniques for exploring and analyzing such complex data. We employ multiple linked views [33] (see Fig. 1), two new multivariate visualization techniques, *petals* and *threads*, and an information-theoretic analytical backend engine for aggregate-level and detail-level network analysis.

Petals and *threads* efficiently present a simplified representation of many-to-many networks where multi-attribute vectors represent the size of attributes in different directions. Specifically, *petals* represent an aggregated summary view of directional data (Fig. 3) and *threads* encode multiple variables of links (Fig. 2). An information-theoretic model provides our analytical engine the ability to highlight anomalies in the data. The anomaly detection can be dynamically configured based on new contextual requirements that usually result from user-generated hypotheses stimulated from visualization and exploration of data. The analytical method provides the visualization with additional warning signals and enables users to prioritize their exploration strategy.

The contributions of our work in the multivariate spatiotemporal network visualization and analysis domain are 1) designing *petals* and *threads* for high-dimensional multivariate network link analysis, 2) evaluating *petals* and *threads* with a user study, 3) designing and implementing a visual analytics system using multiple coordinated views, 4) integrating an information-theoretic anomaly detection method in the interactive visualization analysis process, and 5) exploring complex data (e.g., flight delay network) to illustrate the use and potential of our designs in the multiple-coordinated views.

Our system can be applied to exploration of any multivariate spatiotemporal, network link data generated in transportation, shipping, logistics, commerce, trading, and communication industries (e.g., AT&T communication network data [8] and electric power grid data [46]).

2 RELATED WORK

While the research topics in network visualization are as numerous as the visualizations themselves [22, 38], in this work, we consider network visualization techniques and tools that are pertinent to multivariate geospatial network data. For multivariate network visualization research, Wattenberg [43] has designed PivotGraph, a software tool focusing on the relationships between node attributes and connections of multivariate graphs on a grid layout. Ploceus [28] enables multi-dimensional and multi-level network-based visual analysis on tabular data while Honeycomb [42] focuses on scalability (e.g., millions of connections) using a matrix representation that is also incorporated in our matrix view. Shneiderman et al. [38] visualize networks by semantic substrates and Selassie et al. [36] present an edge bundling technique for directed networks.

For geospatial network visualization, Guo [19] has developed an integrated, interactive visualization framework that visualizes

*e-mail: {ko|safzal|yang260|jchae|amalik|ebertd}@purdue.edu
†e-mail: {simon.walton|min.chen}@oerc.ox.ac.uk
‡e-mail:jangy@sejong.edu

IEEE Symposium on Visual Analytics Science and Technology 2014
November 9-14, Paris, France
978-1-4799-6227-3/14/$31.00 ©2014 IEEE

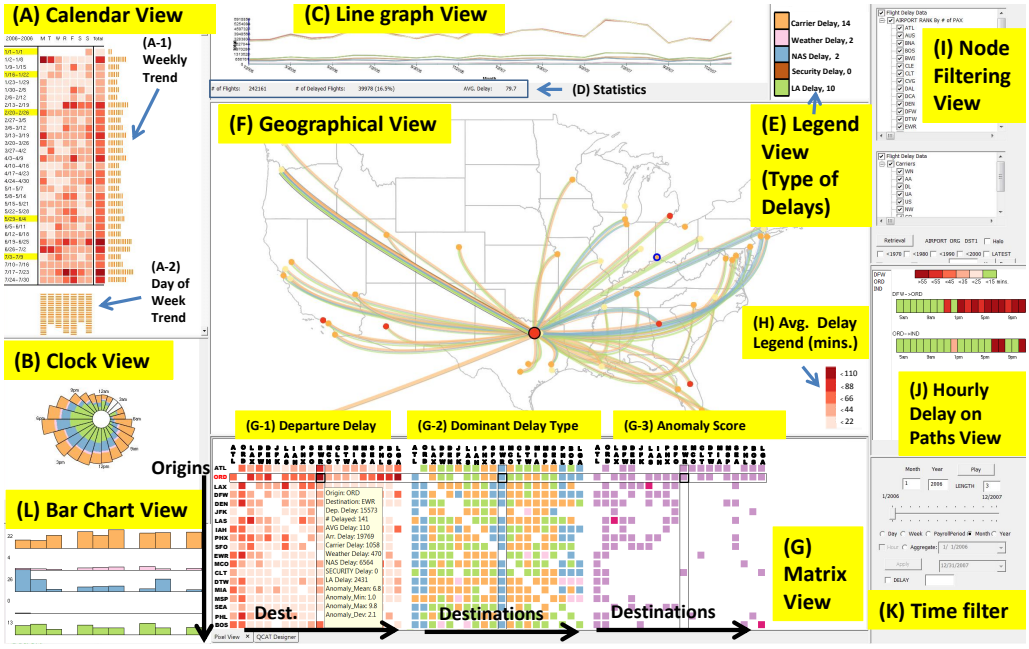


Figure 1: Our system consists of multiple coordinated and linked views: (A) Calendar view, (B) Clock view, (C) Line graph view, (D) Statistics, (E) Legend view for displaying types of delays, (F) Geographical view, (G) Matrix view, (H) Legend view for delay type and time, (I) Node filter, (J) Pattern on itinerary view, (K) Time and aggregation filter, and (L) Twitter tag cloud view. In the (H) legend, the darker the red, the longer the average delay is. A route from Dallas (DFW) to Portland (PDX) is specified in (F), and the top 20 airports in terms of delays are visualized in (G) for explanation. In (G-3), the red links have the highest level of Z-scores, while the purple links have the second largest level of Z-scores.

major flow structures and multivariate relations at the same time. SeeNet [8] visualizes geospatial network data in a communication industry; however, its visualization focuses on univariate data. In contrast to the previous work, our system allows users to analyze all combinations of spatial, temporal, multivariate, and network characteristics simultaneously. Herman et al. [22] surveyed other network visualization techniques beyond our paper's scope.

In order to visualize multivariate data, and to display the maximum amount of data relative to the available screen space, a pixel-based visualization was developed by Keim et al. [23]. In the pixel-based visualization, each data attribute is assigned to a pixel, and a predefined color map is used to shade the pixel to represent the range of the data attribute. Thus, the amount of information in the visualization is theoretically limited only by the resolution of the screen. Borgo et al. [9] present how the usability of the pixel-based visualization varies across different tasks and block resolutions while Ko et al. [25] demonstrate the effectiveness of pixel-based visualization in the task of analyzing corporate competitive advantages. Unlike the pixel displays, the matrix displays assign fewer nodes on both axes of a matrix, and the relational attributes of two nodes are visualized in a link location where the two nodes meet. Matrix displays have been widely used for network visualization due to their effectiveness in providing an overview of the connections in dense networks [14, 16, 21]. Our system utilizes both pixel and network displays, not only to visualize multivariate data (e.g., airports–airlines), but also to describe the network (e.g., airports–airports). We use the term "link" for matrix displays that corresponds to "a pixel" in the pixel displays. Heatmaps present attributes through different shadings of rectangular tiles in a data matrix [45]. We use the heatmap shading approach in the calendar representation [44] that is incorporated in our system.

To help users visually explore multivariate data, many systems have been developed in research and commercial areas [47] (e.g., Spotfire [6], QlikView [4], and Tableau [5]). Common among these systems is that they make extensive use of interactive techniques for brushing, linking, zooming, and filtering to refine the user's queries. Of the systems, Tableau [5], which has become popular due to its flexible operation, allows analysts to easily access and effectively analyze their data [47]. Although multivariate and time-series data analysis is possible in the tool, comparison among multivariate, spatial-temporal, and network-based attributes with geographical components is not well supported by Tableau. In our system, all attributes and characteristics in the data are incorporated and visualized using multiple linked views for simultaneous comparisons. For visualizing multivariate data, Duffy et al. [13] use a glyph encoding some 20 variables while Scheepens et al. [35] focus on a method for reducing visual clutter and occlusion among glyphs.

Lee and Ziang [27] provide an overview of using information-theoretical measures for anomaly detection, including entropy, conditional entropy, information gain, and information cost. A number of case studies are also provided in the domain of network security. Chandola et al. provided a comprehensive survey on methods for anomaly detection [11]. Arackaparambil et al. [7] use information theory to monitor network streams for anomalies in network traffic, and to explore the challenges of providing a scalable implementation using a distributed approach to computing entropy and conditional entropy. Kopylova et al. [26] investigate the use of mutual information in network traffic anomaly detection using Rényi entropy rather than the traditional Shannon entropy measure.

3 Multivariate Network Visualization

To effectively reveal as many aspects of the data characteristics as possible, we explore the data in a series of linked visualizations. Fig. 1illustrates how our system provides comprehensive multivariate network information in multiple linked views. For illustration, we use a flight delay network dataset [1] as an example of multivariate geospatial network data, but any multivariate network data can be populated into our system. Multivariate network information is provided in the geographical view (F) where any operational variable can be used for coloring the node (e.g., anomaly score). The user can explore the data in either a matrix view or a parallel coordinate view (G). Note that (G) has two tab views at the

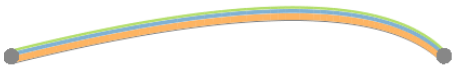


Figure 2: *Thread* example, showing a single link with multiple *threads*. Width of each *thread* within a link is adjusted based on the contribution of each variable. Contribution of each variable in this example is as follows: Variable 1 (Orange) = 0.5, Variable 2 (Blue) = 0.3, Variable 3 (Green) = 0.2.

bottom, and a parallel coordinate view example in (G) is shown in Fig. 7. Similarly, time-varying variables (e.g., delays) are presented in different linked visualizations for efficient exploration in the calendar view (A), clock view (B), and line graph view (C). In the bar chart view (L), the user can interactively compare 1) five delays in all petals, 2) five delays in a petal pie wedge, and 3) five delays for an origin-destination pair *thread*. The height of the bars is normalized and the numeric delay information (longest) is presented in (L). The hourly delay of paths view (J), is designed to allow users to explore attributes in a series of nodes on the paths that a user specifies. With the flight delay data, the user can compare the delays between a direct flight and stop-over flights. As an example, DFW–ORD–IND is shown in (J), where we see that a delay will possibly be maximized if a traveler leaves after 1pm from DFW and between 7pm–9pm from ORD. Users can select airports for analysis in (I) and choose the time in (K). In the system, the line graph view presents temporally aggregated data (e.g., weekly, monthly, yearly). The parallel coordinate view (discussed later in Section 5.2) can be be used to explore the attributes and their value distributions, as well as designing and selecting Query Conditional ATtributes (QCATs, discussed in Section 4) for anomaly detection. Based on characteristics of the data, perceptually appropriate color maps are chosen from both sequential and qualitative color maps from ColorBrewer [20].

3.1 Spatial Multivariate Network Visualization

Unfortunately, a barrier exists in analyzing multivariate network data because visual clutter and complexity often occur in visualizing multiple variables for a node with multiple links between nodes in the map. To reduce such clutter and complexity in the analysis, we design *threads* (see Fig. 2) and *petals* (see Fig. 3) for exploring multivariate link network data. *Threads* connect an origin to each destination and visualize multiple link variables. Because visual clutter around the origin is often generated by link visualization and our *threads*, we also design *petals* to present aggregated and simplified many-to-many network link data. *Threads* and *petals* are designed based on the following requirements for the visualization:

- R1 A visualization should present multiple variables describing the relationships between an origin and multiple destination nodes on the map. Here, users should be able to see an overview of the multivariate relationships and discern at least the largest variable in the visualization for both one-to-one and one-to-many relationships.
- R2 The visualization should provide simplified one-to-many multivariate spatial networks with minimum visual clutter. Use of node rearrangement techniques (e.g., force-based model algorithm [31]) is not allowed to maintain geospatial semantic meanings.
- R3 Users should be able to discern in the visualization for R2 which one-to-many network has the largest aggregate value and which variable has the largest contribution for the largest aggregate value of the one-to-many network.
- R4 Multiple variables describing the statistics for a node should be visually presented.

For goal R1, we design *threads*, and for goals R2–R4 we design *petals*. In the following sections, we explain their visual representations in detail.

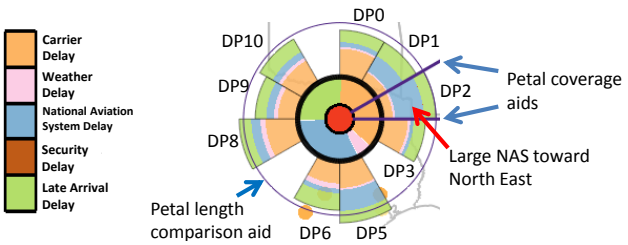


Figure 3: To show the *petal* coverage including destinations, *petal* coverage guide lines are provided (Other coverage aids examples are in Fig. 5 and Fig. 9). For comparison of *petal* lengths, equal-radius circles are drawn on all *petals* as shown. The radius of the circles is the length of the *petal* where a user's mouse is hovering.

3.1.1 Thread Visual Representation

We design the *thread* visualization for representing multiple link variables with a focus on the relationship in an origin-destination pair (R1). Each network link consists of multiple *threads*, and each *thread*'s width is scaled based on a link variable's value. Therefore, each link has the same number of *threads* as the number of link variables, but with varying *thread* widths. While GreenGrid [46] utilizes the force-directed layout [31] and presents a (combined) variable on its links, the *threads* are placed on physical locations and present multiple variables. Users can choose the node variables to be encoded in the *thread* link width. Fig. 2 illustrates an example presenting how link variables can be mapped to *threads*. In this work, we use the departure delay times for each cause of delay as the link variables. This visual representation helps users easily identify which link has the largest delay and which delay type contributes most to the delay. In addition, when a link is specified as an anomalous link, it is located on the top in the stack of *threads* and other links become transparent so that the anomalous link can be highlighted as shown in Fig. 1 (F). To show the direction, an origin node is larger than other destination nodes and has a black outline on the node. Note that Bezier curves are utilized for the link visualizations, and *threads* can be sorted (e.g., departure delays or anomaly scores in our implementation). We incorporate general Bezier curves [32] but we configure the control points of the curves so that long-haul flights tend to be straighter than short-haul curves. In addition, we invert the direction of the normal vectors of the curves alternatively to prevent the case that all control points are moved to one direction in each quadrant. To help user perception, our system provides zooming (with a mouse wheel) and allows users to select the *thread* base width.

3.1.2 Petal Visualization

We introduce *petals*, a new directionally-aggregated radial visual representation as shown in Fig. 3 (Dallas, TX). In this representation, we can provide aggregated directional multivariate network link visualization with minimal visual clutter because we avoid link crossings [8]. Moreover, the spatial and multivariate characteristics are preserved and emphasized. Each directional *petal* (DP) encodes various information between one origin and multiple destinations in a given aggregate direction. Many transportation and logistics problems do have variable variation that is directionally dependent due to transportation paths, weather, routing, etc. By radiating from the origin location to multiple directions (one- to-many), a *petal* presents the geospatial relationships (R2). The *petal* length encodes a selected variable value (R2). Additional variable information is then encoded as radial sections within each *petal* (R3). For example, with the flight delay network data, the average departure delay for the flights heading for airports in a certain radial direction is mapped to the length of the *petal*. Then, the five types of delays are encoded by length (i.e., a segment on a radius) inside the *petal* presenting the contributions of each delay type. Thus, we interpret

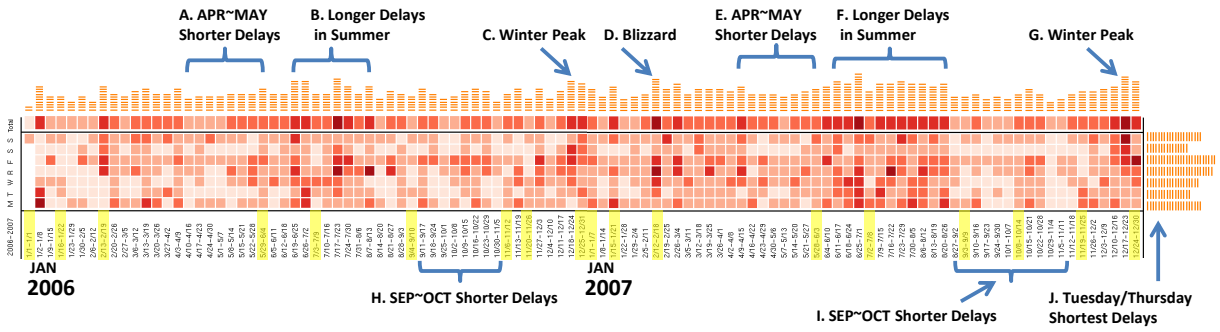


Figure 4: Calendar view showing delay patterns for 2006–2007. In general, there were long delays in the summer and winter seasons, while APR–MAY and SEP–OCT did not have as many delays. Some delays increased around the holidays (highlighted in yellow), but not all holidays had much impact on the delays.

that DP2 in Fig. 3, has a large NAS (National Aviation System, pointed by an red arrow) delay from Dallas. This indicates a large air traffic delay for the destinations, especially toward the airports in New York. Within a *petal*, we insert a pie chart visualization to show comprehensive overviews and comparisons among multiple variables in a node (R4). In the system, users can turn the *petal* display on and off. By default, we assign 12 *petals* for each origin, but users can change the number of *petals*, merge two adjacent *petals* or split one *petal* into many *petals* if necessary. To help users easily recognize the destinations included in a *petal*, our system provides *petal* coverage guide lines as shown in Fig. 3. In addition, when the mouse hovers on a *petal*, the destinations included in the *petal* turn red for better recognition. Lastly to ease comparison of *petal* lengths, equal-radius circles are drawn on all *petals*. The radius of the circle is the length of the *petal* where a user's mouse is hovering (e.g., the radius of the current circle in Fig. 3 is the length of DP2). A tooltip presenting numeric information is provided when a mouse hovers at the center of the *petals*. This can be used for comparing a variable at one location to a variable at another location. Note that the data for the destinations within a *petal*'s coverage are aggregated and visualized together in the corresponding *petal*.

3.2 Network Matrix Displays

Matrix displays have been adapted in various network visualizations because they are effective in providing an overview and relationships of nodes in a dense network. We utilize the matrix displays in our system to provide more complete multivariate network information, as shown in Fig. 1 *(G)*. Our system allows up to three matrices, where the *y*-axis of all matrices are the origins while the *x*-axis represent destinations. For example, for a flight delay network data for 20 airports, we place the departure delay matrix in *(G-1)*, the dominant delay type matrix (e.g., weather, security) in *(G-2)*, and the anomaly Z-score (or standard score, $z = \frac{x-\mu}{\sigma}$ where μ is mean and σ is standard deviation) matrix *(G-3)* from our information-theoretic model as discussed in Section 4. Note that a Z-score filter is applied so that red links have Z-scores larger than 2 (97.7%) and purple links have Z-scores between 1 and 2 (84.1%). In our implementation, users can optionally make *G-3* present additional delay information (e.g., delay by airplane ages and by airlines as shown in Fig. 6 (c) and (d)). When a mouse hovers on a link, a tooltip pops up to display detailed information including delays of different types, the number of flights, and the anomaly scores, as shown in Fig. 1 (G-1). This interaction method is useful when a user wants to find out whether a delay type presented as a dominant type in (G-2) is indeed dominant among all delay types.

3.3 Time Series Displays

In order to present temporal trends, our system provides various time-series views: a calendar view *(A)*, clock view *(B)*, and line graph view *(C)* in Fig. 1. With the calendar representation [44] that applies a calendar metaphor to effectively reveal seasonality and cyclic trends, our system presents the delays by using different shading levels. For instance, the longer delays are presented with darker red. In addition, to help users identify any holiday effect, the week including a holiday has a yellow background. In order to supplement the functionality of the monthly trend line graph, our calendar representation provides additional weekly information on the right side of the calendar *(A-1)* and day of weekly patterns at the bottom of the calendar *(A-2)* in Fig. 1. The clock representation *(B)* is an efficient tool to detect houly trends [17], and we encode variables using areas to enhance visual perception according to Stevens' power law [39]. The line graph view *(C)* presents the types of aggregated delays as well as statistics such as the number of total flights, delayed flights, and average delay time.

4 Anomaly Detection and Highlighting

The visualizations in our system are able to draw upon an information-theoretic model for anomaly detection in a context-sensitive manner, utilizing the anomaly data for a consistent highlighting strategy shown throughout the visualization pipeline. For example, while Fig. 1 *(G-3)* explicitly encodes the anomaly score as the primary visual attribute, Fig. 1 *(F)* focuses on highly anomalous routes with thin outlines. In this case, attribute $a_{origin} = DFW$ (Dallas) is set as the condition in the model. What defines an 'anomalous' record depends upon the user's design and definition of individual anomaly detectors, *QCATs*, discussed in detail in this section. From a visual analytical perspective, these QCATs provide an overview of records where important attributes deviate from usual for specific conditions.

4.1 Overview of Anomaly Detection Method

Chandola et al. provided a comprehensive survey on methods for anomaly detection [11], categorizing them based on the nature of inputs, instance types, algorithmic mechanisms, and forms of outputs. For multivariate network data, we are interested in methods that can:

- Handle multi-dimensional records – because the main flight data concerned is a structured data stream consisting of 29 attribute dimensions (e.g., ≥ 10);
- Address the need for detecting contextual anomalies – which can provide a high-degree of flexibility and accommodating dynamic data and task variations in different detection scenarios;
- Facilitate an unsupervised algorithmic mechanism – alleviating the lack of training data in many situations;
- Generate anomaly scores as outputs that can be effectively conveyed by most visualization techniques.

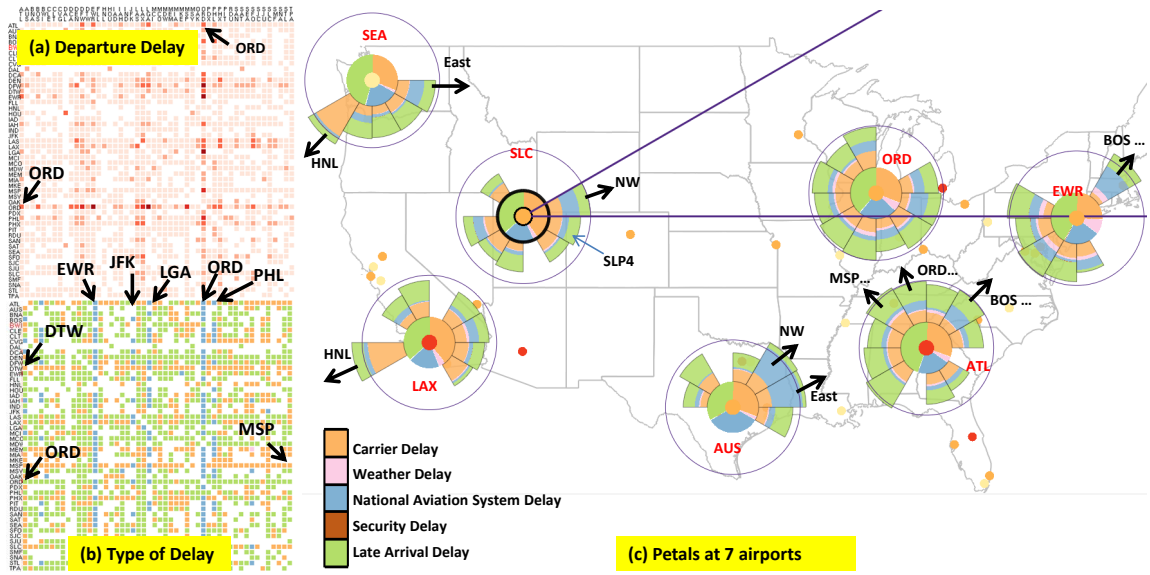


Figure 5: (a) ORD is the most congested airport for both in-bound (vertical) and out-bound (horizontal). It is notable that carrier delay is the prevalent (out-bound) delay for DTW and MSP while NAS delay is the prominent delay for the incoming flights (vertical) in at EWR, JFK and LGA (b). (c) Flights heading to Hawaii from west cost airports in winter had long delays. Flights heading for ORD, ATL and airports from mid-east and east usually suffer from NAS delays.

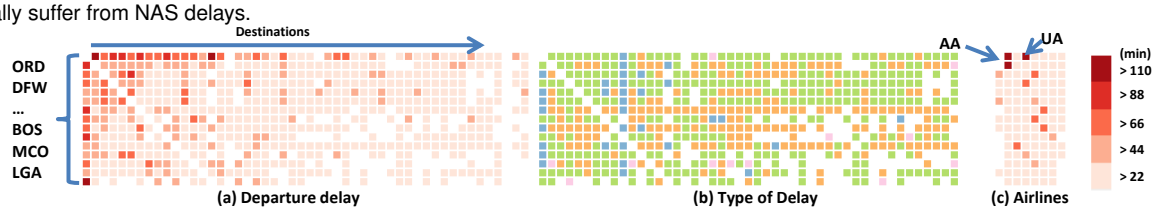


Figure 6: Airports are sorted by delays. ORD shows the longest delays in many out-bound flights in (a). The dominant type of delay was carrier delay and LAD in (b). UA and AA had the longest delays in ORD when ORD was top by delays in (c).

In general, the family of statistical and information-theoretic methods can address the above-mentioned requirements better than the families of classification-, nearest neighbor- and clustering-based methods. As information theory is fundamentally built on probabilistic and statistical measures, information-theoretic methods may also be considered as a subset of the family of statistical methods. In this work, we use an information-theoretic method because of advantages as highlighted in [11]. "(1) They can operate in an unsupervised setting. (2) They do not make any assumptions about the underlying statistical distribution for the data."

Let $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$ be a set of n variables. Each data record, $R = \{v_1, v_2, \ldots, v_n\}$ be a n-tuple, where v_i represents a valid value of attribute $\mathbf{a}_i$. In a practical scenario, an attribute, $\mathbf{a}_i$, may have a very large or infinite number of valid values. Binning is normally used to facilitate more accurate estimation of the probability of each valid value. In the following discussion, the probability distribution of an attribute, $p(\mathbf{a}_i)$, is assumed to be estimated in conjunction with an appropriate binning scheme.

The attribute set, $\mathbf{A}$, is divided into three mutually-exclusive subsets, $\mathbf{A}_{cnd}$, $\mathbf{A}_{von}$, and $\mathbf{A}_{ins}$. As anomalies are context-sensitive, $\mathbf{A}_{cnd}$ defines the context of a type of anomaly as a particular condition, such that all attributes in $\mathbf{A}_{cnd}$ are associated with specific values. For example, we may have $\mathbf{a}_4 = 1(Monday)$, $\mathbf{a}_{17} = JFK$, $\mathbf{a}_{18} = LHR$. The attributes in $\mathbf{A}_{cnd}$ are referred to as *conditional attributes*. In some situations, a conditional attribute may also take a range of values, e.g., $\mathbf{a}_4 = 1,2,3,4$ or 5 (Monday–Friday).

The attributes in $\mathbf{A}_{von}$ play the primary role in determining an anomaly score for each record that has met the condition defined by $\mathbf{A}_{cnd}$. These attributes are referred to as *Variants of Normality* (VON). The remaining attributes, which are grouped into $\mathbf{A}_{ins}$, are considered to have "insignificant" influence on the type of anomaly concerned and are therefore excluded in the computation. Such a decision is usually made based on some known factors or logical reasoning by the user.

A combined configuration of $\mathbf{A}_{cnd}$ and $\mathbf{A}_{von}$ in relation to the overall attribute set $\mathbf{A}$, subsequently, determines how anomaly scores are estimated for each record. Given a record R, we first retrieve all records that have the same conditional attribute values as R. Let this collection of records be $R_1, R_2, \ldots, R_W$, where W is usually a very large number. We now consider only the variants of normality defined by $\mathbf{A}_{von} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_j, \ldots, \mathbf{x}_s\}$. In conjunction with a binning scheme, each attribute, $\mathbf{x}_j$, may take valid values that are mapped to a set of t_j bins $B_j = \{b_{j,1}, b_{j,2}, \ldots, b_{j,t_j}\}$. For the s attributes in $\mathbf{A}_{von}$, there are a total of: $t_1 \times t_2 \times \ldots \times t_s$ different combinations of bins across different attributes. These combinations collectively define an alphabet $\mathscr{Z}$, and each unique combination is a letter $z \in \mathscr{Z}$.

The selection of an appropriate binning scheme for each attribute $\mathbf{x}_j$ is essential for ensuring that the total number of letters $|\mathscr{Z}|$ is smaller than the total number of records W. Ideally, we have $|\mathscr{Z}| << W$. We can, then, estimate the probability of each letter $z \in \mathscr{Z}$ based on the collection of records $R_1, R_2, \ldots, R_W$, resulting in a probability distribution function $p(z)$. For the given record R, we obtain its probability $p(R)$ by mapping it to its corresponding letter in $\mathscr{Z}$. The level of self-information is $I(R) = -\log_2(p(R))$, which is also called *surprisal*. We use this surprisal value as the anomaly score for the given record R. The level of uncertainty of this score can be defined as $H(\mathscr{Z})/\log_2(|\mathscr{Z}|)$, where $H(\mathscr{Z})$ is the entropy of the alphabet $\mathscr{Z}$.

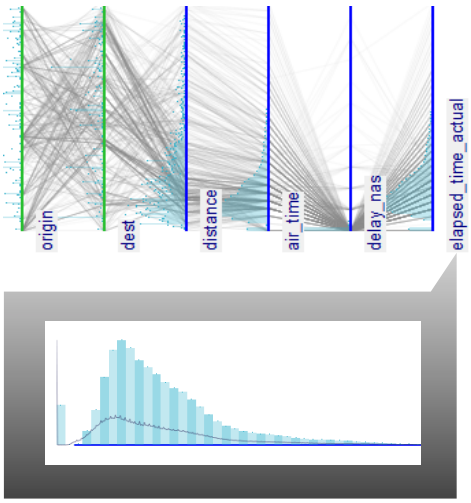
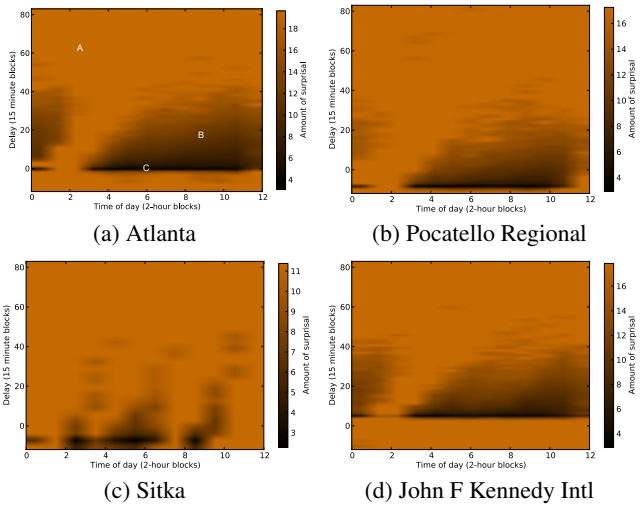


Figure 7: Using Parallel Coordinates to Design QCATs: *(top)* Exploring the attributes as a parallel coordinate plot; *(bottom)* Specifying an individual attribute's bin specification



(a) Atlanta (b) Pocatello Regional

(c) Sitka (d) John F Kennedy Intl

Figure 8: Heatmaps representing the surprisal spaces of flights leaving four different airports, with *(x)* Time of day *(bin size: 2 hrs)*, and *(y)* Departure delay *(bin size: 15 mins)*

It is necessary to emphasize that the anomaly score obtained for R reflects only the type of anomalies encoded by the specific configuration of $\mathbf{A}_{cnd}$ and $\mathbf{A}_{von}$. Hence, each configuration is only for queries of a specific type of anomaly in a particular context. We call each configuration a QCAT (**Q**uery **C**onditional **AT**tributes). It is not difficult to see that a visual analytics system can be equipped with one or more QCATs. For a given record, scores obtained using different QCATs can be aggregated, though it is necessary to understand the semantic implication of combining different QCATs and the difference between different aggregation methods (e.g., mean or max). Section 5.2 discusses the workflow for working with QCATs in a visual analytical system.

The information-theoretic method for anomaly detection is not an algorithm in a traditional sense. Using this approach, anomalies are defined mathematically based on the probability of events captured by the historical data. So in relation to this definition of anomaly, the probabilistic ranking of events using the method is always correct. On the other hand, machine learning methods mostly use a different definition, where an event is anomalous if it is subjectively annotated as an anomaly. So the goal of a learned algorithm is to mimic human perception of an anomaly. One cannot compare the accuracy of these two methods directly. For qualitative comparison, refer to the survey by Chandola et al. [11], where a few other approaches are considered. The mathematics is not new in this algorithm [12,41] but to the best of our knowledge, this kind of probabilistic measures have not been used in visualization, or for the flight data.

4.2 Implementation & Scalability

We have conducted a series of tests on the scalability of QCATs. Two implementations, client- and server-based, have been developed using PostgreSQL [3]. The former performs the grouping and aggregation on the client (i.e, in native code), and the latter uses a stored procedure hosted by the database server. Both server- and client-based implementations show that QCATs are linearly scalable in relation to the number of records used in the computation; the server-based implementation is about 2.5 times faster than the client-based implementation. Additionally, the client implementation is more sensitive to the network bandwidth and latency to the database server.

In our scalability tests, we have found that the performance of the server-based solution can be seriously affected by the number of VONs in $\mathbf{A}_{von}$, while the client-based implementation shows steady linear scalability in relation to the increasing number of VONs. The largest factor is the amount of shared buffers provided to PostgreSQL. The scalability of entropy computation is linear but does rely on recomputing past data due to updated probability masses. However, Arackaparambil et al. [7] show that a distributed method for conditional entropy computation is feasible, while Guba et al. [18] demonstrate entropy estimation in streaming insert-only datasets. In the following sections, we describe how our system presents multivariate network data and visualizes the detected anomalies.

5 Geospatial Multivariate Network Data Exploration

As an example, we will use US domestic flight delay data from the Bureau of Transportation Statistics (BTS) [1] where each data row provides information for an individual flight including origin, destination, day of week, day of month, scheduled (departure/arrival) time, and real (departure/arrival) time and type of delay. There are five types of delays. Carrier delay is a problem within the airlines' control including mechanical problems of aircrafts, while NAS delay is caused by the control of the National Aviation System (NAS) including heavy traffic volume. Late Arrival Delay (LAD) is caused by the late arrival of the same aircraft at a previous airport. Security delay includes re-boarding time due to security breach and waiting time at the screening equipment. Weather delay means delay caused by extreme weather conditions at point of departure or arrival. Note that NAS delay and Security delay might be caused by the government organizations, while Carrier delay and LAD are caused by the airlines. We use the top 50 airports according to the number of passenger boardings that encompasses FAA's OEP-35 (Operational Evolution Partnership 35) airports accounting for more than 70% of the entire number of passengers [2].

5.1 Flight Delay Network Exploration

In this section, we explore the flight delay network data from 2006-2007 and summarize delay patterns in terms of temporal (e.g., summer, winter, holidays, weekly, hourly, and day of week) and spatial effects including special conditions such as severe weather (e.g., blizzards). First, we use the calendar view to investigate data patterns. In Fig. 4, we can see long delays as prominent seasonal patterns in the summer (B, F) and winter (C, G), while shorter delays were recorded during April–May and September–October. Another visible pattern is that there were fewer delays on Tuesday and Saturday in (J). We find that the patterns are related to holidays that are

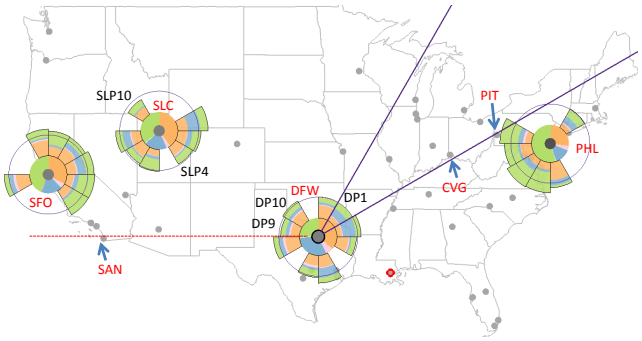


Figure 9: An example for a *petal* experiment. With the visual aid, users could better tell that CVG is included in DP1 while PIT is included in DP2.

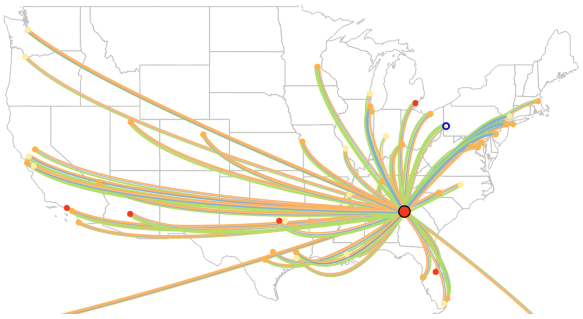


Figure 10: An example of a *thread* experiment. 40% of the participants answered incorrectly that green was the prevalent delay due to the severe color concentration around the origin.

concentrated in summer and winter (e.g., Independence Day in July, Christmas in December, personal vacations) but long delay patterns are not indicated for Martin Luther King Day in January and Labor Day in September. Moreover, long delay patterns tend to increase in 2007, especially in the summer (B and F). Also, there is a sudden spike (D) shown with the darkest red that might be another point for investigation.

Next, we can explore the aggregated delays for two years in the matrix view as shown in Fig. 5 (a, b), where we see some interesting patterns. The most prominent pattern is the series of horizontal and vertical dark red links (long delays) generated at the Chicago O'Hare Airport (ORD) in (a), which indicates that both in-bound (horizontal) and out-bound (vertical) flights were severely congested. We also observe that such delays in ORD were caused mainly by late arrivals of aircraft (horizontal green line) shown in (b). In addition, we notice that there are five distinguishable vertical blue lines in the matrix (b) and four of them (EWR, JFK, LGA, and ORD) were regulated by the High Density Rule (HDR) enacted in 1969 by the FAA due to severe congestion. This may indicate that the rule might not be strong enough to prevent such long delays. The delays in DTW (Detroit) and MSP (Minneapolis), which are two of the biggest hubs of Delta Airlines, are not very long compared to those in other top congested airports. However, it is interesting that the major type of delay is carrier delay (orange) caused by the airline itself.

Since one of the highest delays is observed in winter as shown in Fig. 4, we use our *petal* visualization with winter seasonal data for finding patterns and types of delays in the network as shown in Fig. 5 (c). We can select as many *petals* as designed for the exploration as long as minimal visual clutter is maintained. One interesting finding is that the flights heading for HNL (Hawaii) from the west coast airports (SEA and LAX) have relatively long delays (e.g., 120 minutes on average) and the prevalent cause for the delay is carrier delay. Moreover, those airports also have relatively long NAS delays for flights heading for north-east destinations (ORD, and airports around New York).

The next interesting aspect is the delay distribution by time as shown in Fig. 1 (B) in the proportional mode with area encoding for each delay type. Here, we see a trend showing that delays increased from 6 am and had a peak around 6 pm. It is noted that this is the same pattern shown in the late aircraft delay while other types retained their proportion. This suggests that delays propagate during the day, a problem that Mazzeo termed "cascading delays" [29]. Such trends may imply that delays might be effectively reduced because these delays can be controlled either by the airlines (carrier delay/late arrival delay) with enough of an interval or layover time between two consecutive flight schedules, or by a government agency (e.g., Federal Aviation Administration) with advanced systems for air traffic control. *Threads* can be a good means for understanding delay patterns, as well as the concentration of delays and complications at hub airports. Figure 1 (F) presents an example of the complicated network status at DFW, a hub of two major airlines in *Threads* (April 2012). Here we see that the airport has many connections with airports across the U.S., and the major type of delay is late arrival delay (green) and carrier delay (yellow). In addition, the flights heading for New York City suffer from NAS (National Aviation System) delay.

Of primary interest are the patterns in the length and types of delays that can be can be better explored by sorting airports. We see that the ranks change with little variation based on seasons, but most delays are caused by major airports including ORD (Chicago), ATL (Atlanta), LGA (New York City), EWR (New York City), DTW (Detroit), LAX (Los Angeles), LAS (Las Vegas), and DFW (Dallas) as shown in Fig. 6 (a). From the type matrix Fig. 6 (b), we notice that in many highly-ranked airports, the main type of delay is the late arrival delay in busy travel seasons while the NAS delay is dominant at other times. This implies that the NAS might not be properly adapting to the current increasing traffic in terms of delays. On the other hand, we notice that the two distinguishable airlines causing delays are AA (American Airline) and UA (United Airline) in the two most delayed airports as shown in Fig. 6 (c). The dominant delay type matrix in Fig. 6 (b) indicates that the airlines are responsible for solving the delay problem because the dominant types of delays were carrier delay and late aircraft arrivals.

5.2 QCAT Workflow

As discussed in Section 4, our system features an information-theoretic anomaly detection system that is comprised of a set of user-defined QCATs. The design of a QCAT can be based on a specific hypothesis, or as a more general monitoring system for one or more attributes. Ideally, in a deployed system, the roles of QCAT *designer* and overall *analyst* would be disparate, with the analyst analyzing the data for anomalies and reporting back to the designer to refine the QCATs based on new trends.

To assist the user in defining the QCATs in the system, we provide a design tool based on parallel coordinates (see Fig. 7 *(top)*) while the user is able to explore the attribute space by adding/removing attribute dimensions, observing their value distributions (e.g., probability mass functions), as well as viewing the record relationship between attributes afforded by a standard parallel coordinates representation. The role of an attribute can be toggled between conditional (green) and VON (black) using the right mouse button. The user is also able to explore an individual attribute in more detail by clicking the left mouse button that expands the attribute to the full view to show its distribution in more detail (Fig. 7 *(bottom)*). The detail view also shows the attribute's bin width specification, which can be modified per QCAT. The user's choice of bin width has an effect on the anomaly results and reflects the user's knowledge of the attribute's semantic meaning. The system maps data types to suitable bin width granularities automatically. For example, timestamp datatypes are divided into bins of *n*

Petal Index	Difference (%)	Accuracy (%)	Time (s)
DP1	4 (Small)	76.7	8.2
DP10	21 (Large)	100	3.7

Table 1: Participants found the longest delay inside a *petal* more accurately in less time as the difference became larger (HP1).

Petal Index	Difference (%)	Accuracy (%)	Time (s)
DP9	3 (Small)	46.7 (83.3)	6.6 (5.4)
SLP10	12 (Large)	96.7 (100)	3.9 (2.6)

Table 2: As the difference became larger, the participants better detected the shortest delay (HP2). Visual aids improved both the accuracy and efficiency (HP5).

minutes; categorical data such as strings are unbinned. Since integer types may represent categorical, interval or ratio measurements, we assume a default bin width of 1 and let the user decide upon a more suitable width.

Once the user has defined a QCAT, it can be saved to the QCAT library and selected as the active QCAT. Anomaly-supporting visualizations in our system such as the network matrix view update to reflect the anomaly scores by completing the relevant conditionals in the QCAT (i.e., origin and destination pairs) and executing the QCAT on the data to obtain statistics (i.e., mean, max, variance) on the surprisal values for records matching the conditionals. Our system by default displays the maximum surprisal value as the anomaly value mapped to a visual attribute (i.e, outline on *threads*) in the visualization. The anomaly values in the visualizations guide the user to identify abnormal flights based on their own criteria specified in the design of each QCAT. Anomalous results can then be explored further using the available visual analytical tools to understand why the anomaly value was high and report these findings to the QCAT's designer.

For a QCAT consisting of two VONs, we can illustrate the anomaly distribution using a heatmap. Fig. 8 shows the anomaly space for flights leaving four different airports for the years 2006 and 2007. The x-axis shows the time of day (morning) divided into two-hour blocks, and the y-axis shows the amount of delay in 20-minute blocks (notice that flights can leave early). Areas of low surprisal value are black and become amber with higher surprisal values. It is clear that for this airport, flights around 4AM are uncommon, and the amount of delay seems to increase steadily throughout the day until late afternoon before leveling out. For the Atlanta airport, three example records, A, B and C are shown of high ($\approx$19.68), slightly above average ($\approx$14.36), and low ($\approx$3.478) surprisal values, respectively. Investigating these flights using *threads* shows that late aircraft were largely to blame for both A and B; however, in the case of A the high surprisal value indicates that such a large delay is unusual at this time of the morning. At C, we find ourselves in the 'usual' low-anomaly area for this airport, where delays are close to zero for most of the day.

A professional analyst from an industry-leading company that deals with flight delay data evaluated our system and our approaches used in this work. The analyst mentioned that, at this company, they do not have such visual tools that can enable visual analysis of multiple variables at different locations and different times. Therefore, our system is excellent for dealing with challenging data in the flight delay domain, and it is cutting-edge work for the industry. In particular, the information theory based anomaly detection approach is very intriguing, and it has not been applied to analyses in the industry as of today. Lastly, the analyst suggested visualizations of correlations and propagation of delays (or cascading delay) as key properties of an interconnected network to enhance our system because such visualizations allow for a form of root-cause analysis to help analysts see what is driving delays in the network and what is happening to the delay debt.

Petals	Diff. (%)	Accuracy (%)	Time (s)
SLC+SFO+PHL	2 (Small)	36.7 (73.3)	10.9 (6.8)
SLC+SFO+PHL	12 (Large)	96.7 (96.7)	4.7 (5.1)

Table 3: Users had difficulty finding the longest delay among distant *petals* with a small (2%) difference (HP3). The visual aid helped the users better answer with a small difference (HP5).

6 User Study

In order to evaluate the *petal* and *thread* designs, we performed a user study with 30 participants recruited from various majors at our university. In the study, the participants were given computer-based tasks for verifying hypotheses. Various difference levels in the flight delay network data were used in the tasks. Note that the *difference level* in this section means the difference between the longest (shortest) and second longest (shortest) delays. Note that the numbers in parentheses in Table 2, Table 3, and Table 4 are the results with visual aids. We use a paired t-test to check if our experimental result obtained is significant (p-value < 0.05) within a 95% confidence interval.

6.1 Petal User Study Results

We first set up the following hypotheses for the *petals* visualization as follows:

HP1 As the difference becomes larger, users will show high accuracy and speed in detecting the longest delay inside a *petal*,

HP2 As the difference becomes larger, users will show high accuracy and speed in finding the shortest (or longest) delay among *petals* for one operational place (e.g., airport),

HP3 Users will show lower accuracy in finding the shortest (or longest) *petal* among the *petals* at multiple operational places,

HP4 Users will show low accuracy and speed in finding whether an airport is included in *a petal* as the distance between the *petal* and the airport becomes longer and as an airport is close to the boundary of the *petal*, and

HP5 Visual aids will improve accuracy and speed.

TASK1 for verifying HP1 asked the participants to choose the longest delay inside a *petal* in 2 locations: DP1 (delay difference: 4%) and DP10 (21%), as shown in Fig. 9. The participants showed higher accuracy and speed as the difference increased (Table 1, p-value < 0.05). In TASK2, for verifying HP2, the participants were asked to select the shortest *petal* in 2 locations: DFW (3%) and SLC (12%). For a small difference (3%), 46.7% of the participants answered correctly. As the difference became larger and the visual aid (circle) was provided (HP5), both accuracy and speed were improved (Table 2, p-value < 0.05). TASK3 was the same as TASK2 but multiple *petals* at Salt Lake City (SLC), San Francisco (SFO), and Philadelphia (PHL) were presented concurrently. Here, the participants showed lower accuracy (from 46.7% to 36.7%) and slower speed (from 6.6s to 10.9s) compared to the results in TASK2. The visual aid (HP5) improved both accuracy and speed in the lower difference (Table 3, p-value < 0.05). In order to evaluate if users accurately recognized the coverage of each *petal* (HP4), TASK4 asked the participants to select airports that were included in DP1 and DP9, as shown in Fig. 9. As summarized in Table 4, the participants showed low accuracy (23.3% and 60%). The main reason for such low accuracy was that it was hard for them to find whether CVG (Cincinnati) and PIT (Pittsburgh) were included in DP1. In the same context, only 60% of the participants correctly found that SAN was not included in DP9. However, with the visual coverage line (HP5), both the accuracy and speed improved.

Petal Index	# of Airports	Accuracy (%)	Time (avg.)
DP1	8	23.3 (83.3)	1.96 (1.18)
DP9	8	60.0 (93.3)	2.3 (1.3)

Table 4: The participants had difficulty finding whether CVG and PIT were included in DP1 (HP4). The visual aid helped the users better recognize if an airport was included in a *petal* or not (HP5).

Difference (%)	Accuracy (%)	Time (s)
3.1 (Small)	66.7	5.2
28 (Large)	100	2.9

Table 5: The participants made errors and spent more time finding the longest delay when the difference in *threads* was small (HT1).

6.2 Thread User Study Results

Next, we set up hypotheses for the *threads*. As the difference between the longest and the second longest delays becomes larger, the users will produce better results in HT1) detecting the longest delay inside the *threads*, and in HT2) choosing the most prevalent delay among all the *threads*. TASK5 for verifying HT1 asked the participants to select the thickest *thread* for small (3.1%) and large (28%) difference levels. As summarized in Table 5, when the difference was small (3.1%), it was hard for the participants to tell the longest delay (66.7% accuracy). On the other hand, when the difference became larger, they answered very accurately and spent less time (p-value < 0.05). TASK6 for verifying HT2 asked the participants to tell the longest delay when all the *threads* were considered. Here we see a similar result as in TASK5: the larger the difference, the higher the accuracy and the slower the speed (Table 6). In TASK6, we had an interesting result showing that special concentration of a color may interfere with accurate visual perception. For example, we can see LAD (green) is concentrated on short-haul routes as shown in Fig. 10. In this case, 40% of the participants thought that LAD was the longest delay for flights leaving from Atlanta, but in fact the carrier delay was 23% larger than LAD. This error rate is unexpected compared to the result in TASK5 where the participants showed higher accuracy and speed with a similar difference (28.6%). Conversely, we think it is possible that users could assume that the color on long-haul routes has the largest value if the color is concentrated in long-haul *threads*. To prevent this, our system provides numeric information in the legend view that users can refer to, as shown in Fig. 1 (E).

7 LIMITATIONS AND DISCUSSION

Petals have a similar appearance to the rose or sunburst diagrams that have been adapted in various contexts [15, 30, 37]. The contribution of *petals* lies in extending the usability of the family of the rose diagram by allowing geographically-directional, multivariate, and aggregated network analysis simultaneously. Discerning widths of *thread* can be hard when each variable has similar values or when a unit *thread* within a route is not thick enough for visual perception. In addition, when a color is concentrated on long-haul or short-haul routes, it could be hard to select the largest value among all *threads*. In these cases, a line with a superimposed histogram can be utilized. To help users with these issues with threads, our system provides interactive bar charts and the numeric variable information in the legend view when a user specifies an area of *threads* (aggregated) and in a tooltip when the user's mouse hovers over an airport (origin to destination). The tooltip in the matrix view can be used for verifying that the presented dominant delay (Fig. 1 (G-2)) is indeed dominant compared to others. The scalability of threads can be limited by two factors: the number of variables and the links. In our system, the number of links can be adjusted by the on/off function in threads and the provided network matrices can complement the link analysis. Our user study implies that a thread with 30% larger value than the others can be distinguished from the others. However, a fundamental issue when a large number of variables is used becomes how many colors a human can distinguish and which colors should be used. Harrower et al. suggest 12 distinguishable colors [20] but a lower number of colors would be effective for the *threads* due to the difficulty in comparing widths.

Difference (%)	Accuracy (%)	Time (s)
11 (Small)	90	5.3
28.6 (Large)	100	2.9
23 (Large)	60	5.1

Table 6: 40% of the participants answered incorrectly with a large difference (23%) in finding the prevalent delay among all *threads*. This may indicate that color concentration on long-haul or short-haul *threads* interferes with visual perception.

8 CONCLUSION AND FUTURE WORK

We have explored complex multivariate network links with multiple tightly-integrated interactive visualizations. We have introduced two new visual representations, *petals* and *threads*, for spatial multivariate link visualization. Our sortable matrix displays have the ability to represent multiple origin and destination pairs, while the linked line graph, calendar, and clock views give opportunities to find temporal characteristics. An information-theoretic anomaly detection model was introduced based on conditional attributes, with the visualizations in the system utilizing the surprisal values for visual highlighting of anomalies in multiple visualization components in a unified manner.

It has several benefits compared to previous systems. Our system allows users to investigate the data status of a large number of operational locations by simultaneously observing various data characteristics at both aggregate (entire network) and detailed levels (e.g., origin-destination pairs) using our multiple linked view. Our new visual representations, *petals* and *threads*, help users find features of multiple spatial network variables with minimum visual clutter; the network matrices aid in analyzing the entire network in terms of multiple origin-destination pairs as well as origin-attribute pairs. Seasonal and cyclical trends can be efficiently detected in the calendar, line graph, and clock visualizations from our system. Lastly, our system provides an information-theoretic model for detecting anomalies based on conditions. For the evaluation of our system, we presented an example using flight delay network data from the top 50 airports to illustrate the use and potential of our designs and the user study results.

Our system can be easily applied to analysis with any other multivariate spatiotemporal, network-based data such as transportation and logistics, trading, and communication industries [8]. As a future work, we plan to incorporate the ability to help users find correlations using *petals* and *threads*. The capability for visualizing cascading effects and clusters of operational places that have the same characteristics will also be investigated. We also plan to use actual routes to enable comparison with length of flights. In addition, we would like to explore our anomaly detection more by investigating methods of combining the anomaly values for groups of QCATs.

ACKNOWLEDGEMENTS

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 2009-ST-061-CI0001-06. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security. Jang's work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2013R1A1A1011170).

REFERENCES

[1] Bureau of Transportation Statistics (Accessed 20 Mar 14. http://www.rita.dot.gov/.

[2] Operational Evolution Partnership 35. http://aspmhelp.faa.gov/index.php/OEP_35.

[3] PostgreSQL (Accessed 11 Jun 14. http://www.postgresql.org/.

[4] Qlikview. http://www.qlikview.com/.

[5] Tableau. http://www.tableausoftware.com.

[6] C. Ahlberg. Spotfire: An information exploration environment. *ACM Special Interest Group on Management of Data Record*, 25(4):25–29, 1996.

[7] C. Arackaparambil, S. Bratus, J. Brody, and A. Shubina. Distributed monitoring of conditional entropy for anomaly detection in streams. In *Parallel Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on*, pages 1–8, 2010.

[8] R. A. Becker, S. G. Eick, and A. R. Wilks. Visualizing network data. *IEEE Transaction on Visualization and Computer Graphics*, 1(1):16–21, Mar. 1995.

[9] R. Borgo, K. Proctor, M. Chen, H. Janicke, T. Murray, and I. Thornton. Evaluating the impact of task demands and block resolution on the effectiveness of pixel-based visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):963–972, 2010.

[10] D. Brooks. What data can't do. *The New York Times*, Feb. 2013.

[11] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys.*, 41(3), July 2009.

[12] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.

[13] B. Duffy, J. Thiyagalingam, S. Walton, D. J. Smith, A. Trefethen, J. C. Kirkman-Brown, E. A. Gaffney, and M. Chen. Glyph-based video visualization for semen analysis. *IEEE Transactions on Visualization and Computer Graphics*, 99:1, 2013.

[14] N. Elmqvist, T.-N. Do, H. Goodell, N. Henry, and J.-D. Fekete. ZAME: Interactive large-scale graph visualization. In *PacificVis*, pages 215–222. IEEE, 2008.

[15] N. Elmqvist, J. Stasko, and P. Tsigas. Datameadow: A visual canvas for analysis of large-scale multivariate data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 187–194, 2007.

[16] J.-D. Fekete. Visualizing networks using adjacency matrices: Progresses and challenges. In *CAD/Graphics*, pages 636–638. IEEE, 2009.

[17] J. Fuchs, F. Fischer, F. Mansmann, E. Bertini, and P. Isenberg. Evaluation of Alternative Glyph Designs for Time Series Data in a Small Multiple Setting. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*. ACM, 2013.

[18] S. Guha, A. Mcgregor, and S. Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 733–742, 2006.

[19] D. Guo. Flow mapping and multivariate visualization of large spatial interaction data. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1041–1048, 2009.

[20] M. A. Harrower and C. A. Brewer. Colorbrewer.org: An online tool for selecting color schemes for maps. *Cartographic Journal*, 40(1):27–37, 2003.

[21] N. Henry, J.-D. Fekete, and M. J. McGuffin. Nodetrix: a hybrid visualization of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, 2007.

[22] Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. In *IEEE Transactions on Visualization and Computer Graphics*, volume 6 (1), pages 24–43. 2000.

[23] D. A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):59–78, 2000.

[24] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

[25] S. Ko, R. Maciejewski, Y. Jang, and D. S. Ebert. Marketanalyzer: An interactive visual analytics system for analyzing competitive advantage using point of sale data. *Computer Graphics Forum*, 31(3):1245–1254, 2012.

[26] Y. Kopylova, D. Buell, C.-T. Huang, and J. Janies. Mutual information applied to anomaly detection. *Communications and Networks, Journal of*, 10(1):89–97, 2008.

[27] W. Lee and D. Xiang. Information-theoretic measures for anomaly detection. In *Security and Privacy, 2001. S P 2001. Proceedings. 2001 IEEE Symposium on*, pages 130–143, 2001.

[28] Z. Liu, S. B. Navathe, and J. T. Stasko. Network-based visual analysis of tabular data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 41–50, 2011.

[29] M. Mazzeo. Competition and service quality in the u.s. airline industry. *Review of Industrial Organization*, 22(4):275–296, June 2003.

[30] F. Nightingale. *Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army*. Harrison and Sons, 1958.

[31] A. Noack. An energy model for visual graph clustering. In *Graph Drawing*, volume 2912 of *Lecture Notes in Computer Science*, pages 425–436. Springer, 2003.

[32] S. M. Peter Shirley, Michael Ashikhmin. *Fundamentals of Computer Graphics*. A K Peters, 2009.

[33] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*, pages 61–71, 2007.

[34] A. Z. Santovena. Big data : evolution, components, challenges and opportunities. Master's thesis, Massachusetts Institute of Technology, Sloan School of Management, 2013.

[35] R. Scheepens, H. van de Wetering, and J. J. van Wijk. Non-overlapping aggregated multivariate glyphs for moving objects. In *IEEE Symposium on Pacific Visualization*, pages 17–24, 2014.

[36] D. Selassie, B. Heller, and J. Heer. Divided edge bundling for directional network data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2354–2363, 2011.

[37] Z. Shen and K.-L. Ma. Mobivis: A visualization system for exploring mobile data. In *IEEE Symposium on Pacific Visualization*, pages 175–182, 2008.

[38] B. Shneiderman and A. Aris. Network visualization by semantic substrates. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):733–740, 2006.

[39] S. S. Stevens. *Psychophysics: Introduction to Its Perceptual, Neural, and Social Prospects*. Wiley, 1975.

[40] J. J. Thomas and K. A. Cook, editors. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[41] M. J. Usher. *Information Theory for Information Technologists*. Computer Science. Macmillan, 1984.

[42] F. van Ham, H.-J. Schulz, and J. M. DiMicco. Honeycomb: Visual analysis of large scale social networks. In *Proceedings of International Conference on Human-Computer Interaction*, volume 5727 of *Lecture Notes in Computer Science*, pages 429–442. Springer, 2009.

[43] Wattenberg, Martin. Visual exploration of multivariate graphs. In *Proceedings of ACM CHI 2006 Conference on Human Factors in Computing Systems*, volume 1 of *Visualization 1*, pages 811–819, 2006.

[44] J. V. Wijk and E. V. Selow. Cluster and calendar based visualization of time series data. In *1999 IEEE Symposium on Information Visualization (INFOVIS '99)*, pages 4–9, Oct. 1999.

[45] L. Wilkinson and M. Friendly. The history of the cluster heat map. *The American Statistician*, 63(2):179–184, 2009.

[46] P. C. Wong, K. Schneider, P. Mackey, H. Foote, G. Chin, Jr., R. Guttromson, and J. Thomas. A novel visualization technique for electric power grid analytics. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):410–423, May/June 2009.

[47] L. Zhang, A. Stoffel, M. Behrisch, S. Mittelstädt, T. Schreck, R. Pompl, S. Weber, H. Last, and D. A. Keim. Visual analytics for the big data era - A comparative review of state-of-the-art commercial systems. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 173–182. IEEE Computer Society, 2012.