Parallel Arc Diagrams:
Visualizing Temporal Interactions

Peter Hoek, p.hoek@student.adfa.edu.au
School of Engineering and Information Technology, University of South Wales at ADFA

Abstract

This paper introduces a new computer-based visualization method, the parallel arc diagram (PAD), which is capable of uniquely representing 2-mode temporal relationships in a manner that assists in highlighting simple features of the network. The PAD approach relies on a computer's ability to render link lines adjacent to each other with orderly precision, resulting in features that facilitate preattentive processing of simple network characteristics and providing the ability to discern patterns of interactions over time. PADs supplement existing methods such as node-link diagrams by offering a simple alternative visualization without the computational complexity of graph layout algorithms and the additional issues that animation introduces. This paper subjectively evaluates the PAD approach using low level task taxonomies developed for assessing adjacency matrix and node-link visualization effectiveness. We argue based on those taxonomies that the PAD approach is as effective or in some cases more effective than existing approaches except for tasks requiring the identification of structural groups or middle-man nodes. This paper also demonstrates how the PAD approach can be utilized in a software application. The TIPAD (Temporal Interactive Parallel Arc Diagram) uses character participation in movie scenes as a test-bed for exploring social interactions over time and provides the ability to compare a PAD based visualization with traditional visualizations of the same network.

Key Words

Dynamic Network, Visualization, Patterns of Interaction

Acknowledgements

Background

For as long as we have sought to understand complex sequences of events in nature and society we have used visual methods to help us explore and explain underlying cause and effect relationships. One might argue that statistics and or other mathematical techniques could provide answers to questions more efficiently and accurately. However, a mathematical approach is much more effective if you know what questions to ask a priori. Knowing what questions to ask often comes from insight gained through visual representations. The human brain is particularly effective at processing visual information and consequentially images and graphics have greatly assisted in advancement of many branches of science. Social network analysis is no exception, (Freeman 2000) suggests it has been central to the growth of social network analysis. The challenge now is to determine what can be improved in the current approaches to social network visualization.

Often the phenomena we seek to explain are contained in data sets which consist of entities and associated relationships which can be expressed formally as a graph. The resulting graph is almost always represented as a node-link diagram (Freeman 2000). Computer-based visualization using the node-link representation is very common, Henry's (2008) thesis points out that the vast majority of visualization software found in the INSNA software repository use node-link diagrams and Kan, Plaisant et al. (2007) also note that nearly all the submissions to the InfoVis 2004 contest used displays of node-link diagrams.

Sociologists and other analysts undertaking social network analysis are increasingly making use of node-link visualizations to analyze and highlight important relationships between the elements of the network. There are a number of software packages that offer the ability to present the network using node-link diagram, some of which are listed in (Bender-deMoll and McFarland 2006). All these software packages generally rely on the large amount of work that has been devoted to improving the algorithms responsible for the visual layout of graphs. However finding an optimal layout is a difficult problem, considering the almost infinite possibilities of laying out the nodes and arcs of a graph in 2- or 3-dimensional space.

Introduction

Layout algorithms of node-link diagrams are designed to optimize the positioning of nodes and edges with respect to various aesthetic criteria such as edge crossings and symmetry, etc. (Battista, Eades et al. 1998). However, the optimization of some criteria is intractable (Garey and Johnson 1983) and, in addition, the optimization of one criterion may impact adversely with the optimization of another causing unexpected consequences(Huang, Hong et al. 2007). Moreover, the number of studies evaluating the aesthetic and perceptual criteria on which those algorithms are based is relatively small (Dwyer, Lee et al. 2009) and there are few well-defined criteria for assessing how "good" a graph layout is at the interpretation level (Bender-deMoll and McFarland 2006).

Studies conducted by (McGrath, Blythe et al. 1997; Huang, Hong et al. 2007) find that the perceptions of structural features of the network changed as the spatial arrangement of the network changed. As Figure 1 illustrates, even at the most fundamental level, techniques used to render a graph can emphasize different network paths (Ware 2000). In this regard node-link visualizations present great challenges in providing a representation of network relationships that could be relied upon to be interpreted in a consistent manner (Huang, Hong et al. 2006). In addition, as noted by Brandes, Raab et al. (2001) the current approaches to node-link drawing

focus on readability rather than visual communication of substantive content. In view of the many issues node-link diagrams pose, some authors have proposed that we move beyond the graph paradigm or at least use graphs in concert with other visualization approaches (Viégas and Donath 2004; Karahalios and Viégas 2006).
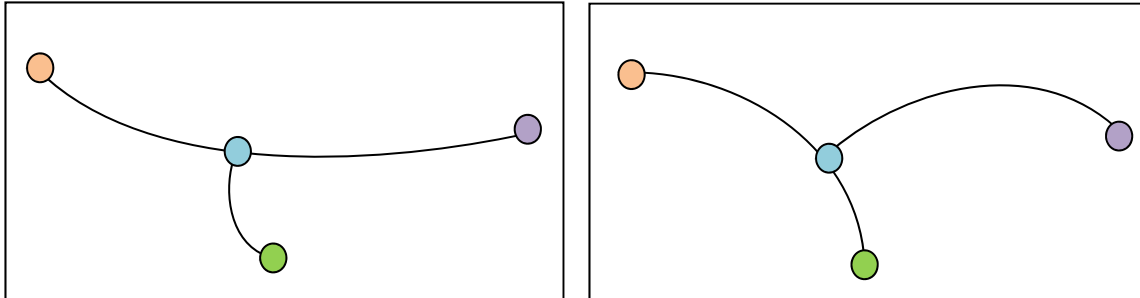

Figure 1 – Different paths are emphasized despite nodes positioned in the same location

Alternatives and Improvements to Node-link Diagrams

A common criticism of node-link diagrams is that, as the number of nodes increases they quickly become cluttered and conceal nodes and relationships making it difficult to discern any information, even when the graph contains as few as 20 nodes (Ghoniem, Fekete et al. 2005). Accordingly, a number of authors have offered alternatives to node-link diagrams. In general these alternatives consist of variations on the node-link diagram or matrix representations or an amalgamation of the two in an attempt to leverage the advantages of both. These advantages are broadly described in a number of studies which found that matrix-based representations were better suited for large and dense graphs and node-link diagrams were more suitable for small and sparse graphs (Ghoniem, Fekete et al. 2005; Keller, Eckert et al. 2006). In fact, it is quite surprising the node-link representations are used as extensively as they are, given that these studies noted that the matrix approach performed better in almost all low level tasks (described in the next section) with the exception of path finding for graphs with more than 20 vertices.
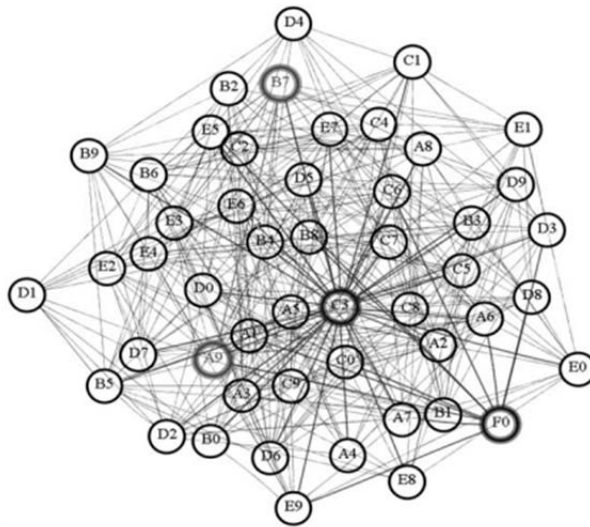
Figure 2 — Visual Complexity of a Network with 50 Nodes and 400 Edges (Ghoniem, Fekete et al. 2005)

Some researchers have attempted to increase the utility of their particular approach by enhancing the visualization so the range of low level tasks that can be accomplished is increased. Henry (2008) for example, added arcs to a matrix representation to facilitate easier path following in response to the aforementioned studies. Munzner (2000) attempted to overcome the clutter of node-line diagrams by adopting hyperbolic space and interactive techniques to explore the graph with the inclusion of interactive highlighting. Others such as Kang, Plaisant et al. (2007) shun the use of a node-link representation. Instead they opt for histograms and simple coordinated views of ordered lists.

One area of improvement which is becoming increasingly important is dynamic or temporal network visualization. Visualization approaches that are most often applied are animations of node-link representations over time or what is termed "static flip books" (Gloor and Yhao 2004; Gloor, Laubacher et al. 2004; Moody, McFarland et al. 2005). The general intent of animations is to help users understand the dynamics of a network over time which may not be easily understood using static representations. Some authors suggest that the use of animation can aid the comprehension of complex data (Kadaba, Irani et al. 2007). However others such as (Tversky, Morrison et al. 2002) have shown that well designed static representations can effectively replace animated diagrams.

In the following section I describe a new visualization method, the parallel arc diagram (PAD) which goes some way to meeting the desire for alternative representations. This visualization method was born out of the need for a simple and informative method of displaying a network and its relationships and one which offers the ability to visualize patterns of interaction over time or discrete periods without animation. It is worth noting here, that in adopting the PAD approach we do not mean to suggest that node-link diagrams have no value. Indeed, node-link diagrams have many advantages; however they do have deficiencies which PADs might go some way to overcoming by being a complementary visualization approach.

The Parallel Arc Diagram (PAD) Described

The concept of the PAD is a simple one; in some respects one could view it as a merging of a matrix representation with that of a node-link diagram and the concept of histograms. Node-link drawings are well suited to being drawn by hand, indeed it is often claimed that is their origin with the sociograms drawn by Moreno (1953). However, a computer can render the same network much more precisely with orderly precision.

The PAD approach relies on the rendering precision that computer technology provides. Consider the simple 2-mode network shown on the left in Figure 2. This graph representation shows relational links that have some temporal ordering associated with them. This network could be redrawn as shown on the right by laying out the links in a parallel fashion immediately adjacent to each other (see animation, http://users.tpg.com.au/pjhoek/PAD_explained.html). It might be argued that in such an approach the link relationships are harder to discern, however this is not the case and will be explained later in this paper. Moreover, according to studies previously described, the typical way of representing a relationship with a link line connected to a node at an arbitrary angle is only to be useful for path-following tasks.
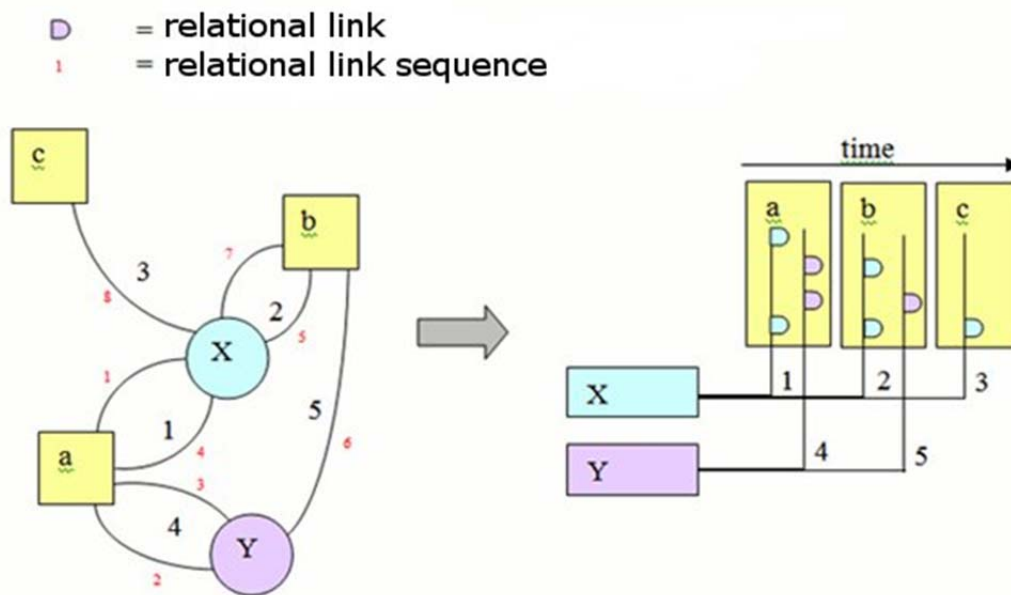

Figure 3 – Node-link layout vs. PAD layout

The key aspect of a 2-mode social network is that it does not record direct relations between social actors but rather via collective entities conventionally termed 'events' (Alexander 2005). The classic dataset from Davis, Gardner et al. (1941) is an illustration of such data and is shown in Figure 5. Other examples of 2-mode data include character participation in movies scenes, staff participation in business meetings, membership of corporate boards, and individual participation in online forums and so on. Often the data has a temporal nature to it and commonly additional data is available related to the actors in a particular event, for example, characters in a movie might have a sequence of alternating dialogue in a movie scene.

Direct relations (1-mode projections) can be derived from 2-mode data; however the process of transforming the data discards some information. In particular, which actors are related to which event and how they are related in time. Furthermore, the data often contains additional detail describing the relationship itself. This could include timing or sequencing of relational links within the event or contextual information such as questioning or responding actions in forum participation for example. The value of the PAD approach is that it attempts to preserve and visualize all aspects of the data including the temporal ordering of relational links and the contextual data.
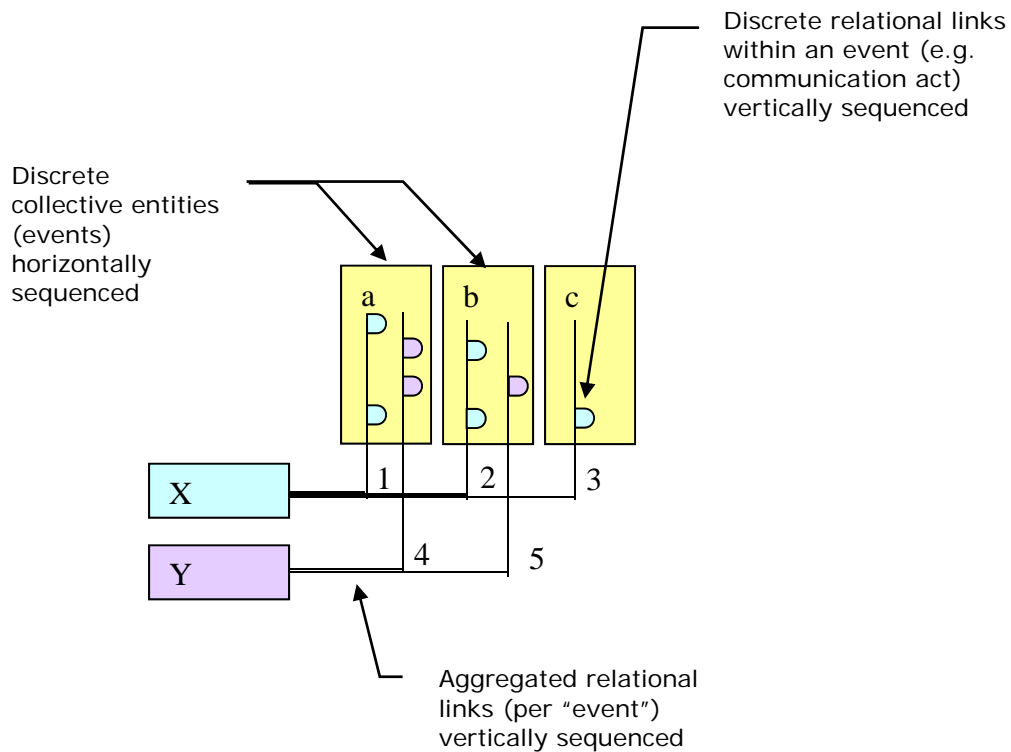

Figure 4 - PAD layout detail

Collective entities (event nodes) are placed horizontally across the top in time ordered sequence and actors, the origin of the data points, are placed vertically down the left. We propose the ordering and indeed the inclusion of actors not be fixed but based on the requirements of the user. The order of actors could be determined by filters and be of any nature desired by the user. For example, the ordering could be based on graph theoretic measures or thresholds. In the examples shown in this paper the actors are first ordered by degree and, second, by time of participation. Actors are assigned a color from a predetermined set and the color sequence is repeated if the number of actors exceeds the available colors in the set.

Link lines representing the relationship from actor to event form horizontal 'bars' of varying thickness due to multiple adjacent lines (shown in Figure 14). In node-link drawings these relational events would normally be represented via aggregated or multiple arcs, as shown in Figure 3. The 'bars' resulting from the PAD approach allow for easy visual determination of relative degree of each node. As the horizontal collective entities (event nodes) are arranged in

sequential order, when a filter is applied controlling which link lines are rendered, patterns emerge based on the filter and the temporal ordering of nodes (demonstrated later in this paper).

This type of layout also offers the ability to include contextual or attribute data in the visualization. In this paper we describe the inclusion of ordered dialogue, indicated by small semi-circles (bubbles) shown in the diagram above. These utterances are ordered vertically allowing the user to view the progression of conversation. Although we see this as a useful addition in this particular example, the point is, as well as preserving the temporal relationship the PAD layout affords the opportunity to preserve and display other elements of the data which normally would be difficult to include.

We consider the PAD approach to be useful in visualizing 2-mode data that has a sequential or temporal dimension to it and can be divided into discrete sessions or clusters. In addition, the data ideally consists of actors that are the origin of data points that relate to the temporally ordered   relational events.
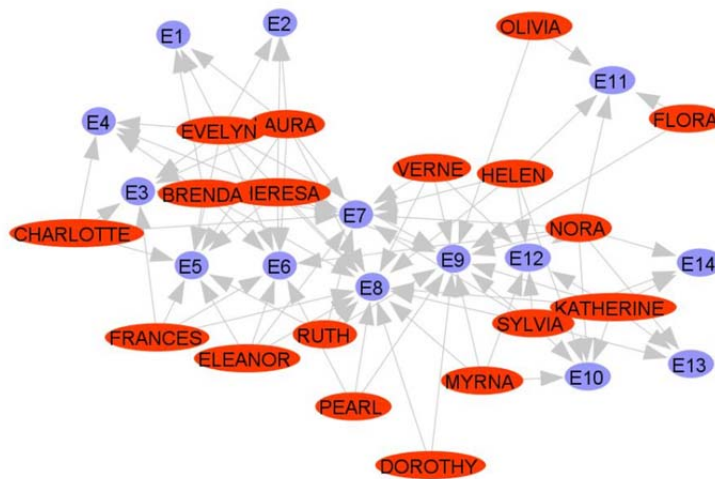

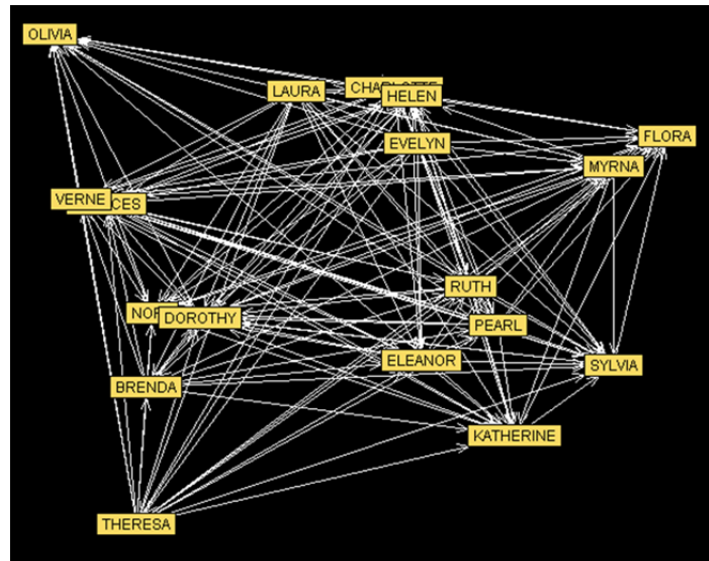Figure 5 – Davis Gardner et al. Dataset displayed as 2-mode node-link

Figure 6 - Davis Gardener et al. Dataset displayed as 1-mode
projection node-link diagram (Freeman 2000)

As a simple introductory example, Figure 7 shows the Davis, Gardner et al. Dataset displayed using the PAD approach. You will note that within each social event or activity (boxes at the top) there are no 'bubbles' indicating individual actor relational events, as this information is not available in the data.

In this example Nora is selected and the social events Nora attended are highlighted. What amounts to a count of co-attendance is displayed next to the actors' names and their co-attendance is highlighted. The pattern of co-attendance of each individual is displayed and highlighted via their colored arcs contrasted against other interactions shown as white lines.
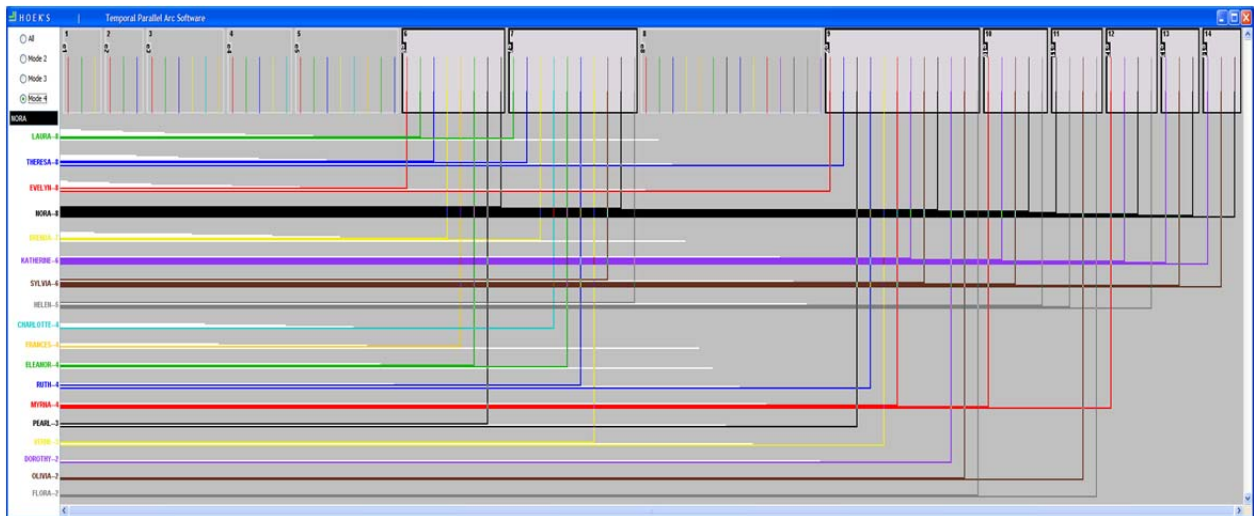

Figure 7 - Davis Gardener et al. Dataset displayed as 2-mode PAD layout

Comparative Evaluation Method

Plaisant (2004) describes information visualization as a way to answer questions you did not know you had. She goes on to point out that this presents challenges in designing tasks to measure the effectiveness of a visualization. In this paper we use task taxonomy derived from various studies of node-link diagrams to comparatively assess the potential utility of the PAD approach. As discussed earlier, node-link visualizations are by far the most commonly used representation in social network analysis, therefore we consider it valid to compare the PAD visualization approach with node-link visualizations.

Graph visualizations are utilized for numerous activities which are comprised of low-level tasks within those activities. Cui's (2006) survey report lists a number of common low level tasks that are commonly accomplished with the use of graph visualizations. These are provided in a summarized form below. Cui states that node-link representations are good enough for the first three of these tasks and matrix layouts are particularly good at revealing different proportions of links from a node that go to different categories.

1. For the whole graph, count the number of nodes.

2. For given node, count the number of its incoming or outgoing links.

3. For a given node, find its adjacent nodes.

4. For a given node, find the nodes that can be reached by a certain number of steps.

5. For the whole graph, find the middleman nodes.

6. For the whole graph, find strongly connected clusters.

7. For the whole graph, find all nodes/links which share some specific attribute.

(Ghoniem, Fekete et al. 2005; Keller, Eckert et al. 2006; Lee 2006) also provide taxonomy of commonly encountered tasks while analyzing graph data. Lee categorizes them in four groups: topology-based, attribute-based, browsing and the overview task. These taxonomies generally align relatively well with Cui's list, with the possible inclusions from Lee:

8. Finding and selecting a node.

9. Finding and selecting a link.

10. Finding the length of the shortest path.

In the following section, we demonstrate the advantages of the PAD approach in the Temporal Interactive Parallel Arc Diagram (TIPAD) application. This application uses character participation in movie scenes as the basis for exploring social interactions over time and facilitates the comparison of the PAD visualization with traditional visualizations of the same network. We will use the extended taxonomy of tasks described above as the basis for comparative evaluation of the approach but will also discuss why one might want to perform some of the listed tasks and why this particular approach to visualization might be appropriate.

Temporal Interactive Parallel Arc Diagram (TIPAD) Described

The primary goal of the TIPAD application is to serve as a means to comparatively evaluate the PAD approach. TIPAD uses a movie script as input and assumes the character dialogue within it is representative of social interaction. Numerous methods have been proposed to analyze movies and usually they focus on audiovisual features of the movie. However, Weng, Chu et al. (Weng, Chu et al. 2007) claim these approaches don't assist in "understanding" movie content and propose analyzing movie content through examining the social relationships between roles. The TIPAD application is designed to visually support analysis that enables "understanding" of the kind proposed by Weng, Chu et al.

In this paper we use *The Matrix* movie script as a simple illustration of the efficacy of using a PAD visualization approach. This particular movie was chosen as it is a relatively well-known movie, has a number of key characters, and presents the development of various relationships during the course of the movie and provides the opportunity to visualize the patterns of interaction.

Similar Work

Similar work was undertaken by (Mutton 2004) in applying his IRC relay chat to visualize animations of social networks derived from the text of the plays of William Shakespeare. TIPAD is differentiated however by the sequential ordering of scenes directly from the script and does not animate a node-link representation.

The work of (Donath, Karahalios et al. 1999; Tat and Carpendale 2006) in designing graphical representations for persistent conversations is also pertinent. Donath, Karahalios et al. focused on the ability to reveal inaction patterns at a glance which are not ordinarily perceivable by simply perusing the conversational archive. Tat and Carpendale also targeted revealing patterns in an individual's online conversations. The TIPAD application is similar to the extent that it immediately reveals patterns of interactions based on a persistent conversation log, a movie script. In addition however, it also reveals inferred relationships, associated dialogue and contextual information in a simple two dimensional display without clutter or occlusion. A goal of this application was the ability to view commonly utilized metrics and characteristics of network visualizations quickly without having to refer to the script.

The Script and the Network

If one were to consider a movie script as a 2-mode bipartite network with one set of vertices composed of all the characters in the movie and the other set of vertices consisting of all the scenes in the movie, co-appearances in scenes might be described by a set of edges composed of an ordered pair from each set of vertices. Describing the movie script as a 2-mode bipartite network appears the most appropriate as it is difficult to discern from a movie script a set of character to character mappings based on dialogue, often what is said in a scene is not directed to an individual but rather broadcast to all.

One should note here that there are existing techniques such as correspondence analysis and bipartite graphs that allow for the analysis and visualization of 2-mode networks (Borgatti, Everett et al. 2002). In addition, there have been attempts to visualize 2-mode social networks using the line-graph of the bipartite adjacency matrix (Alexander 2005). However, these techniques present problems in determining the 'meaning' of values or inhibit the ability to discern ties between nodes, particularly when there is a temporal dimension to the data.

Many movie scripts are documented in a semi-structured format such as the Warner Brothers formatting style (Cole and Haag 1988) making them relatively easy for a computer to parse. *The Matrix* movie script was parsed and encoded in the structure shown in Figure 8. A program was created using the Java programming language to accept files structured in this format, and from this, a network was constructed and visualized using the PAD approach.

```
# by
# Larry and Andy Wachowski
# NUMBERED SHOOTING SCRIPT
# March 29, 1998

Editing: FADE IN:

Scene: 1
Location: ON COMPUTER SCREEN
* so close it has no boundaries.
* A blinding cursor pulses in the electric darkness like a
* heart coursing with phosphorous light, burning beneath
* the derma of black-neon glass.
* A PHONE begins to RING, we hear it as though we were
* making the call. The cursor continues to throb,
* relentlessly patient, until --

Speech: MAN
Mode: V.O.
Yeah?
* Data now slashes across the screen, information flashing
* faster then we can read: "Call trans opt: received.
* 2-19-98 13:24:18 REC:Log>."

Speech: WOMAN
Mode: V.O.
Is everything in place?
* On screen:    "Trace program:  running."
* We listen to the phone conversation as though we were on
* a third line. The man's name is Cypher. The woman,
* Trinity.
```

Figure 8 - Example of *The Matrix* parsed file

Comparative Visualizations

For the purposes of comparison, we now present several common alternative visualizations of the movie *The Matrix.* The bipartite graph (Figure 12) and the adjacency matrix (Figure 10) were created with the Java programming language. Similarly the force-directed (Figures 9 and 13) and radial (Figure 11) layouts were constructed using Java, the Perfuse visualization library (Heer 2004) and the JUNG visualization library (O'Madadhain, Fisher et al.) In the case of the force-directed visualization, the layout parameters were adjusted manually to give the graph the best spatial positioning. This is often the difficulty in using force-directed layouts as a visualization component; the optimal parameters are dependent on the particular network and the requirements of the user.

The Matrix social network could be characterized as small to medium in size, consisting of 220 scene nodes and 28 characters nodes with a total of 316 edges, which is fairly typical in terms of a movie network. We contend that networks in the order of this size are well suited to the PAD approach.
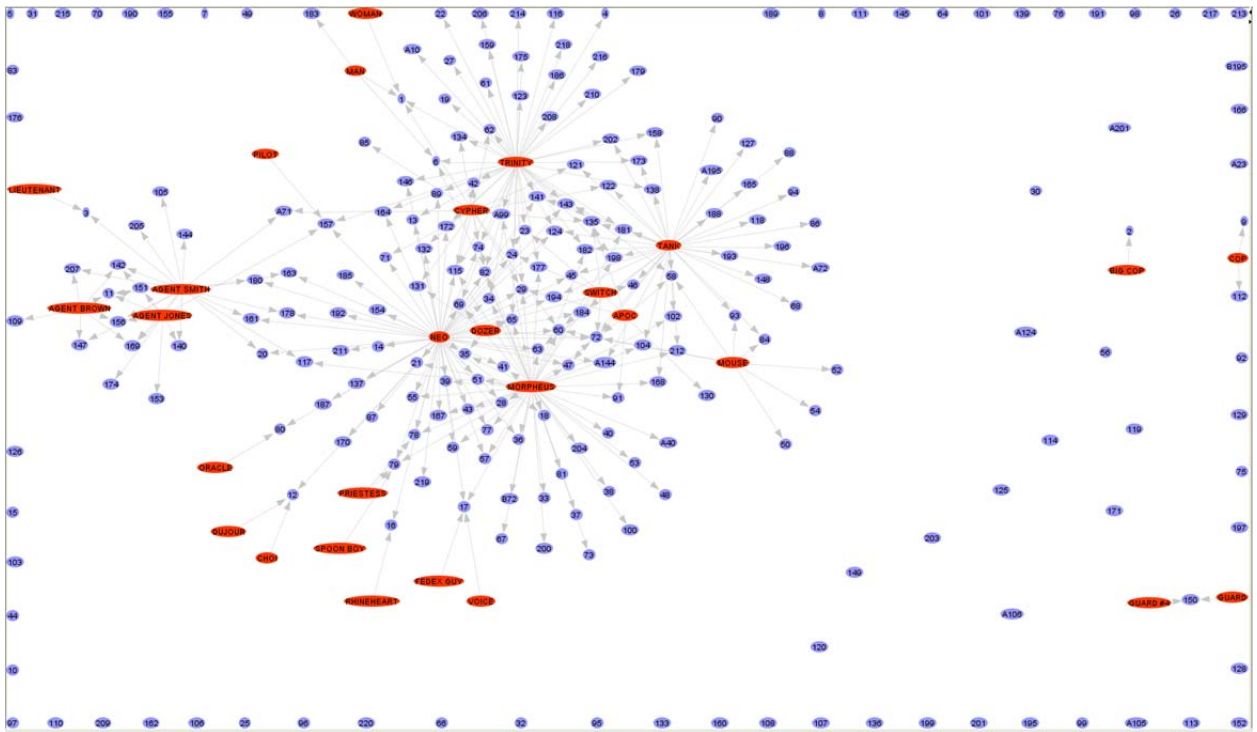

Figure 9 - The Matrix depicted as 2-mode graph using a force-directed layout (aggregated utterances)

When analyzing the alternative representations (Figures 10-13) in respect to the taxonomy described above, it may be possible to discern which characters contribute to the majority of scenes. However, the exact degree to which each contributes is not immediately apparent. The viewer would need to count the edges to confidently place them in rank order. In addition, despite Cui's (2007) assertion that node-line representations facilitate easy counting of the number of nodes, incoming or outgoing links and finding adjacent nodes, these tasks clearly are not "easy."
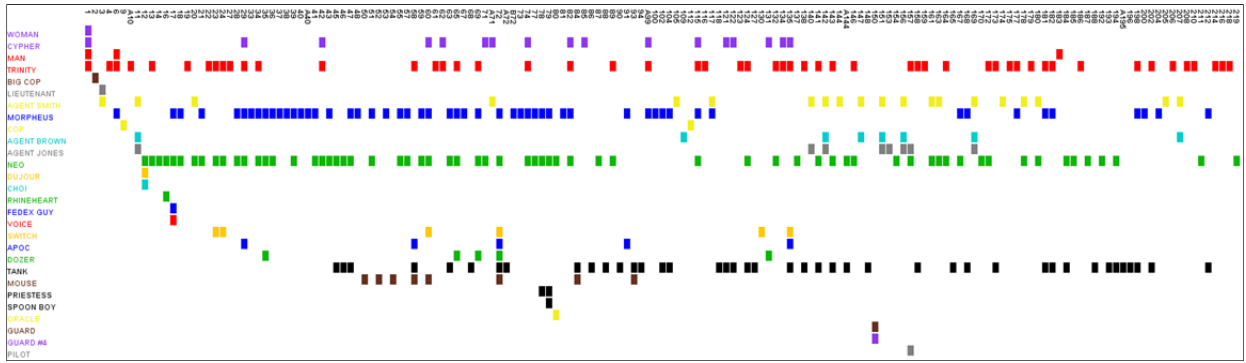
Figure 10 - *The Matrix* depicted as an adjacency matrix (aggregated utterances)
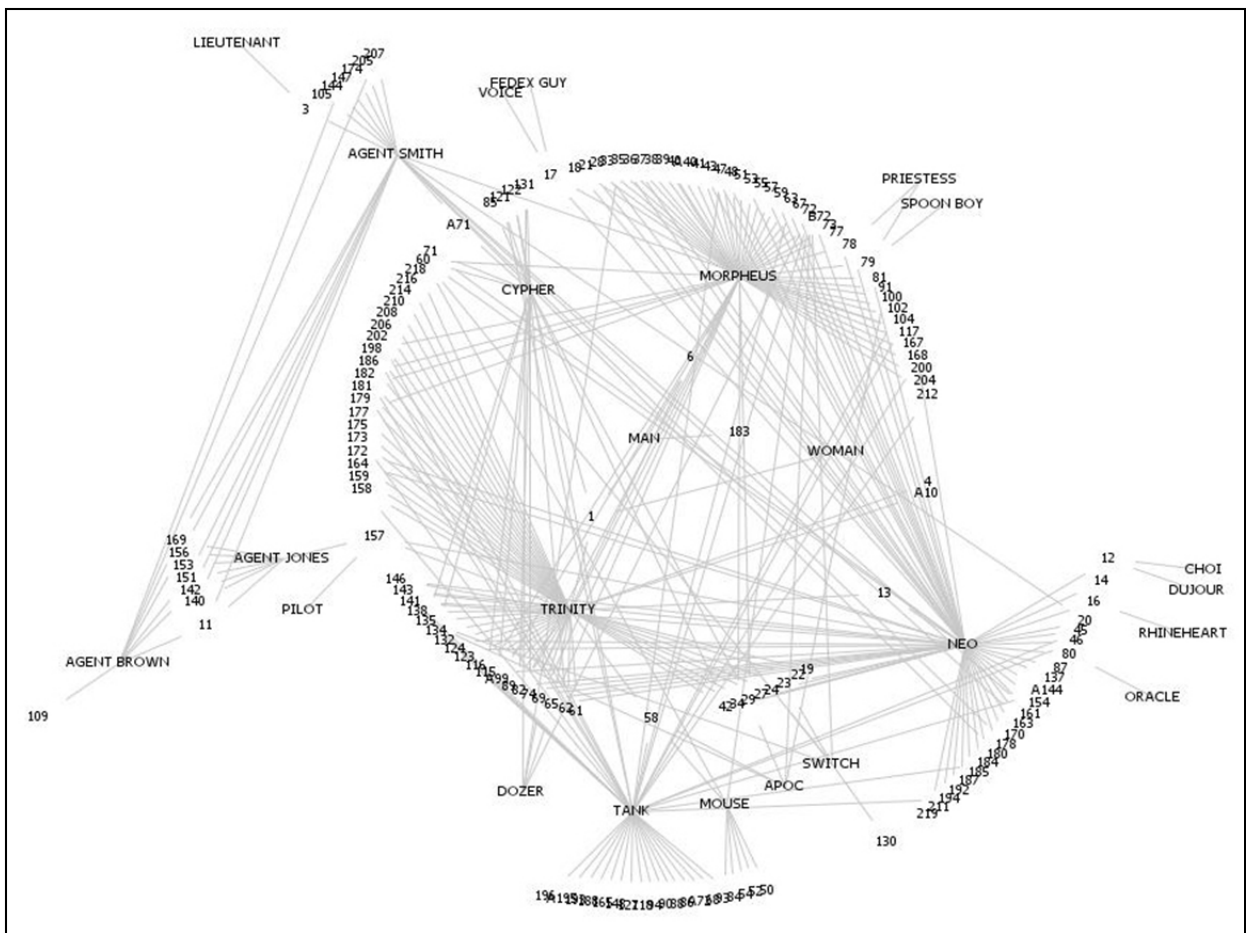

Figure 11 - *The Matrix* depicted using a radial graph layout (aggregated utterances)
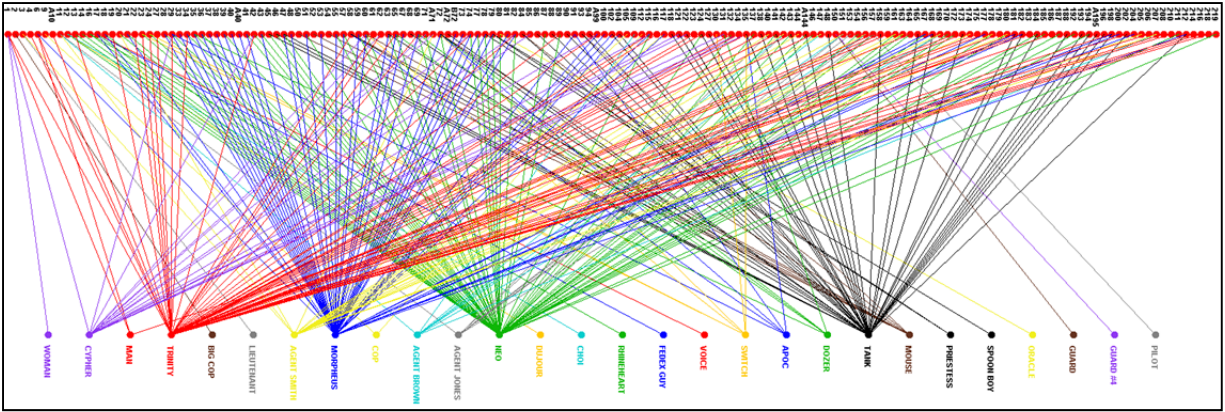
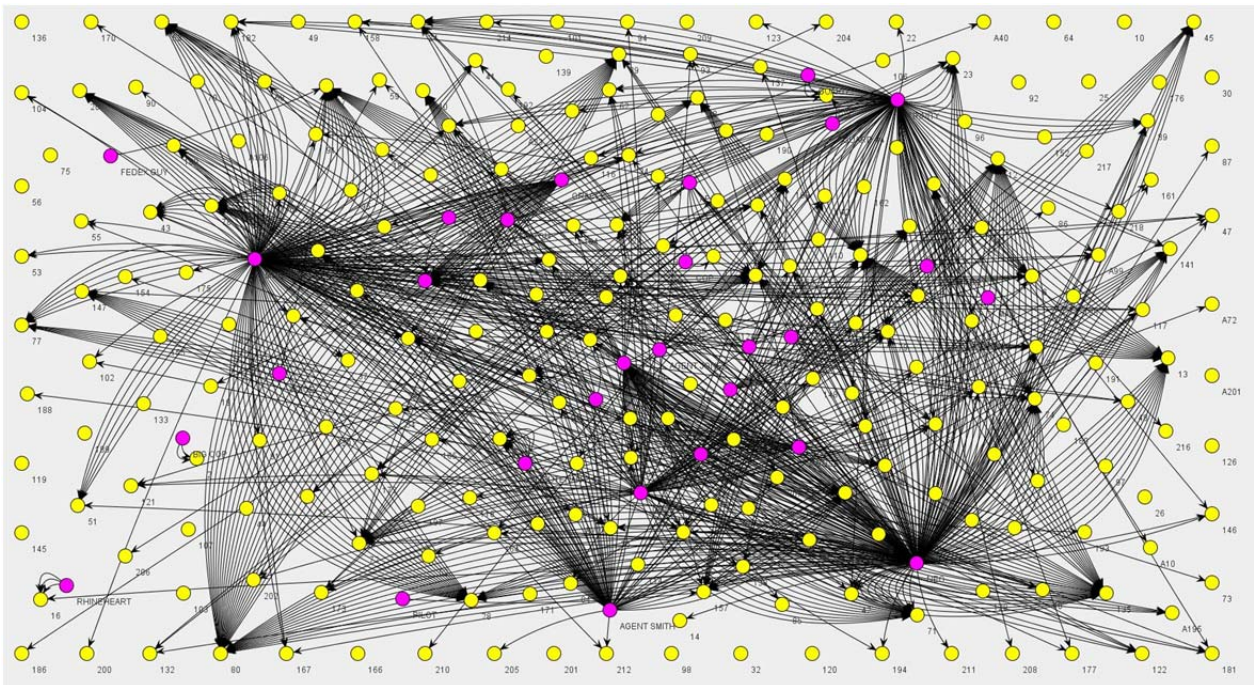Figure 12 - *The Matrix* depicted in a bipartite graph (aggregated utterances)

Figure 13 — *The Matrix* depicted in a 2-mode node-link graph (separate utterances)

Conventional visualizations (above) are static representations and do not offer any ability to discern each character's involvement in movie scenes over time. Additionally, if one wished to view the sequence of dialogue in each scene, within the same visualization, it would be difficult to represent it in an aesthetically appealing and comprehensible way.

Visualizing all utterances of the movie using the node-link approach is shown in Figure 13. Character nodes are rendered in pink and movie scenes in which the utterances occur are colored yellow. This visualization shows the complete set of utterances; however, it still gives no indication of the temporal ordering of them. It is this figure in which the following description of the TIPAD application should be compared. The information density of each is about the same and both attempt to visualize the same dataset with the same level of fidelity.

TIPAD Design and Operation

Authors such as (Brandes, Raab et al. 2001) have noted that in node-link drawing, "the focus is on readability rather than visual communication of substantive content." The design of the TIPAD (shown in Figure 14) was influenced by this and the observation that standard graph layouts often do not make the simple characteristics of the graph immediately apparent, even on small or moderate size networks, as described above. In addition it is difficult using a standard node-link diagram to display interactions over time without the use of animation.

In the TIPAD default view the application attempts to provide an initial overview of the movie as advocated in the (Shneiderman 1996) Visual Information-Seeking Mantra. However this overview does not mean viewing the entire collection, but rather viewing all the characters and their participation in scenes. The TIPAD application organizes the scenes horizontally in sequential order, using panning to see the full complement of scenes. Examples of early visualization environments in which this temporal horizontal ordering is effectively used can be seen in the time-based bar and line graph approach used by (Plaisant, Milash et al. 1996) and (Harris, Allen et al. 1999).

The design of the application acknowledges and makes use of recognized residual effects of English reading habits and the resultant interpretation of graphics (Winn 1994). It does this by following a left to right, top to bottom convention. It temporally ordered with the earliest scenes on the left, and characters from highest degree to the lowest positioned top to bottom.

Those characters that have dialogue in a scene are shown with a horizontal line linking to that scene and each line is adjacent to the next. The difference in the PAD approach to traditional node-link diagrams is in the traditional approach the links are placed at angles dependant on node positioning and it is difficult to estimate the comparative degree of links from one node to another, particularly as the number of edges increase. However, the PAD approach lays these links out adjacent to each other and in sequential order creating a bar with progressively decreasing height over time. The total thickness of the bar resulting from adjacent links reflects the amount of appearances in scenes from a particular point in time. This is similar to the way bar height in a histogram represents a numeric value except that any point in time can be the basis of comparison.

Organizing the links in this way facilitates preattentive visual processing of the information (Healey, Booth et al. 1995; Alexandre and Tavares 2010). Preattentive processing is derived from an area of human cognitive psychology and is based on the idea that there is a limited set of features that enable humans to detect things very rapidly and accurately using the low-level visual system. In this case, the features are color and size, which assist high-speed visual estimation of relative out-degree of character nodes.

One might note that scrolling may be required to see all the characters when viewing a movie with a large cast with the TIPAD. However, by default the characters are positioned in out-degree rank order (in this case, equivalent to the number of scenes in which they contribute to the dialogue). Graphic representation of the degree order gives users an immediate understanding of the ranking by estimated values or at least relative weights, providing a portion of the overview capability described in Lee's taxonomy. In terms of the extended task taxonomy described above, the default view satisfies item 1 and 2, albeit through the use of scrolling and panning. Compare this with a user's ability to do the same in the most commonly-used visualizations (Figures 9 and 13) and one will see that the PAD approach provides this capability much more effectively.

In terms of analyzing the movie (Weng, Chu et al. 2007) claim that degree centrality is directly related to leading roles. In *The Matrix*, one could hypothesize that Neo, Morpheus, Trinity and perhaps Tank are the main characters. Although it is possible in most cases to determine the characters' degree using traditional visualizations, the PAD approach requires less cognitive effort.
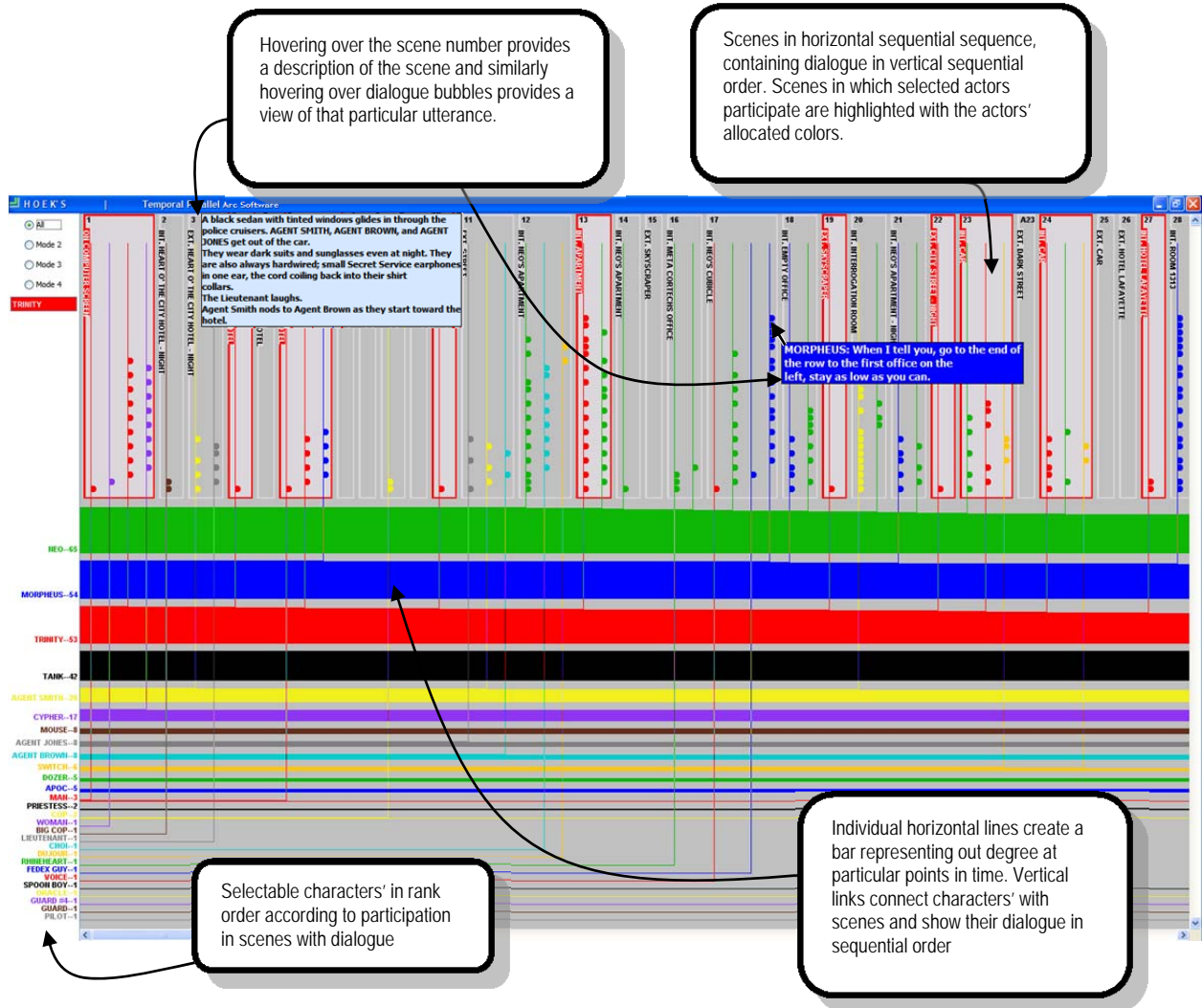

Figure 14 - TIPAD, showing the default view of *The Matrix* in a PAD layout

The design of the TIPAD application recognizes research such as (Lee 2006), which demonstrates that visual exploration of network data benefits from interactive representations. Interactive visualization assists the user by partially transferring the cognitive load to an external representation (Munzner 2000; Goud 2006). Basic interaction in TIPAD is performed with mouse operations including selection of characters, selection of display modes, scrolling, hovering and zooming.

The scene rectangles in the top third of the TIPAD display contain vertical color coded lines indicating contributions by characters in that scene. Hovering over a line will display an information popup with the character's name. Similarly, hovering over a scene number will provide a detailed description of the scene. Each of the vertical character lines has 'bubbles' representing dialogue for that character displayed in sequential order. Hovering over a bubble will show the character's name and the particular utterance. The visualization, therefore, conveys temporal ordering in the horizontal direction based on scene order, as well as temporal ordering in the vertical direction within a scene, based on dialogue order. This ability is generally not provided in the other comparative visualizations and in some cases would be difficult to include within the same visual space.

Selecting a character by clicking on a name in the left-hand pane (or directly in the scene node) and changing mode to 'mode 2' provides a filtered view of the character's participation in the scene or movie. Actors with dialogue in scenes with the selected actor are displayed, providing immediate visual feedback as to actors' co-appearances in scenes (Figure 14). In this example, the character Trinity is selected and provides a somewhat egocentric view of the network, allowing one to see immediately that Neo has the most appearances in scenes with Trinity, followed by Morpheus and Tank.

Figure 19 demonstrates how the PAD visualization naturally shows the amount of interaction over a time period. Rapidly decreasing height of the horizontal bars formed by the link lines is an indication of high participation in scenes and may be quickly identified with the use of scrolling. Figures 20 and 21 identify different patterns of interaction. One can easily determine that Neo's participation in scenes in the early stages is dominated by scenes in which Morpheus also participates. A number of thick bands of lines indicate that, in terms of Neo's total participation in the movie, on three occasions he participates in numerous successive scenes with Morpheus. However, his co-appearances in scenes with Trinity are more evenly distributed.

As the characters (nodes) are presented as a vertical list, finding a node does not require searching the full two dimensional space as is the case in a node-link diagram. This effectively provides the capability to 'find and select a node' (item eight of the extended taxonomy above).

The TIPAD application also highlights the scene nodes to which the selected character links and is equivalent to highlighting all scenes in which the selected character has dialogue. In terms of the task taxonomy this satisfies item three ('for a given node, find adjacent nodes'). In addition, the application filters those nodes to display only those characters that appear in the same scenes; effectively, adjacent nodes that are two links away. This interactive capability can be provided in traditional node-link visualizations. However, the PAD approach provides a filtered representation while maintaining the visual temporal ordering of links.

By selecting 'mode 4' one can view the sequence of co-appearances in scenes with the selected character (see Figure 17). In accordance somewhat with the Shneiderman mantra (Shneiderman 1996), the TIPAD application allows a user to zoom in and examine the patterns of interactions with the selected character (see Figure 18 for a 'zoomed' view). In this mode the user can see how those co-appearances (shown in the character's assigned color) relate in sequence to the rest of the scenes undertaken by the co-appearing character (shown in white).

If the user wishes to focus on the number of characters and patterns of interaction with the selected character, 'mode 3' displays only those links representing co-appearances without the distraction of other link lines (see Figure 16). Mode 3 also provides a crude means of determining communities within the movie. Typically, communities are groups of nodes that have dense connections among themselves but sparse connections with other communities. By progressively viewing characters that have dialogue in the same scenes and logically disregarding those possessing a degree below a particular value, one can make a rough estimate of the members of a community.
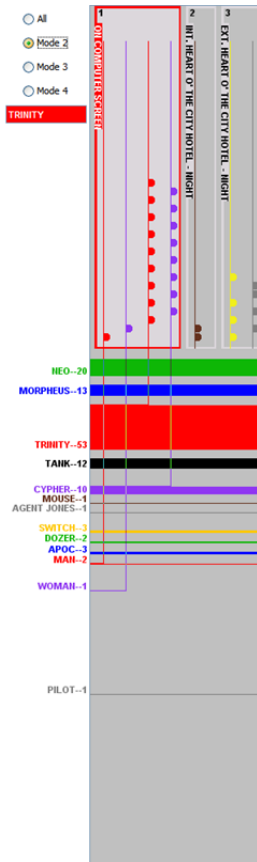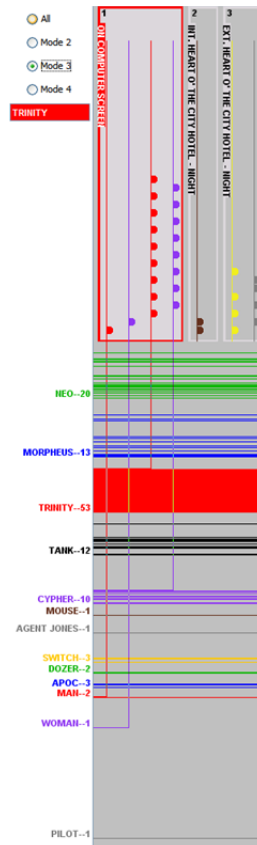

Figure 15 - Basic egocentric view

Figure 16 – Mode 3

Figure 17- Mode 4

Trinity is selected

Complete set of links with only links to scenes with Trinity

Notable periods when Morpheus and Tank do not have dialogue with Trinity.
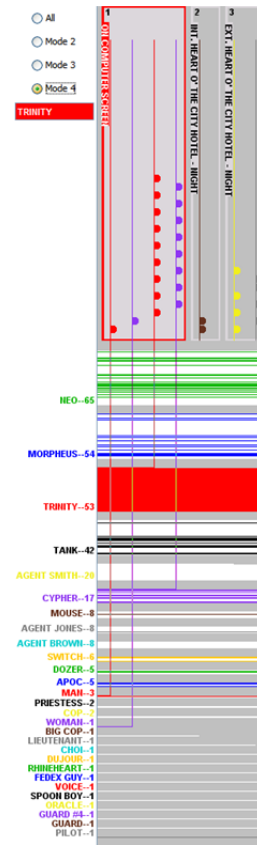
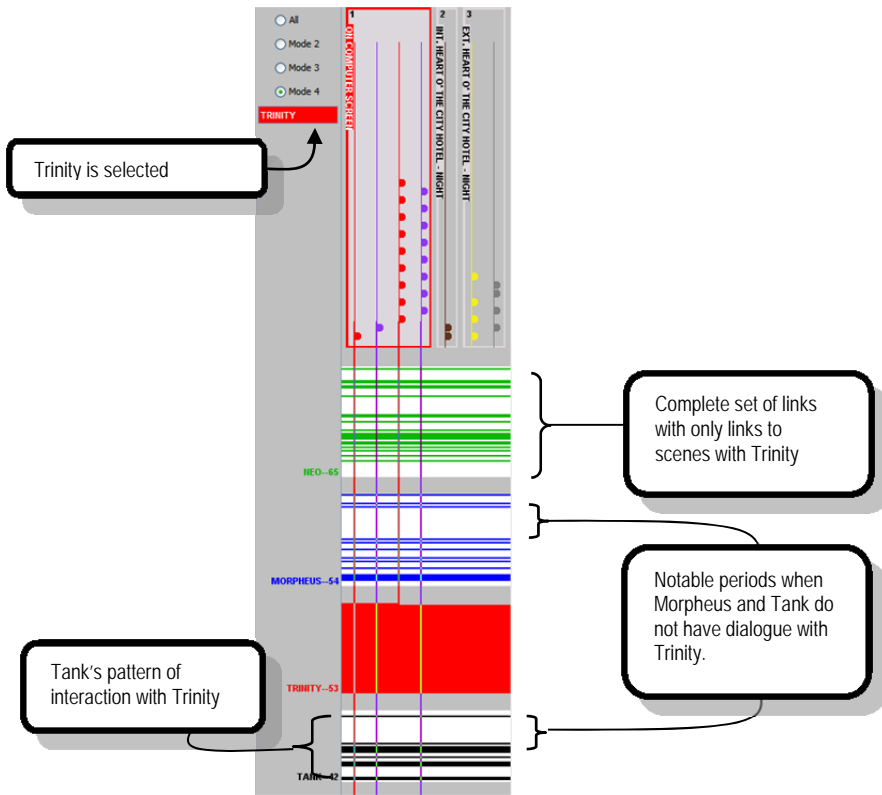Tank's pattern of interaction with Trinity

Figure 18 - Mode 4, zoomed view

Figure 19 - High vs. Low participation in scenes

Figure 20 – Participation with Trinity

Dark bands indicate a number of successive scenes with Morpheus

Trinity has 2 groups (Phases) of interaction with Morpheus
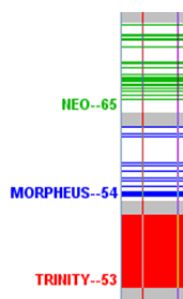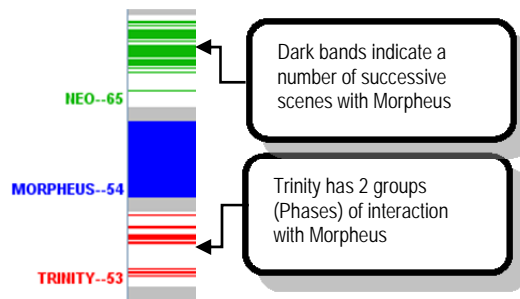
Figure 21 – Participation with Morpheus

It is cumbersome to describe an interactive visualization application using words and static pictures. This short video will help make clear the interactive operation and practical use of the TIPAD and the utility of the PAD approach: http://users.tpg.com.au/pjhoek/TIPADbuffer.html (Adobe Shockwave Player required).

Discussion and Conclusions

Despite the fact that node-link diagrams are widely used across numerous domains, studies indicate that on even moderate-size graphs they lose their utility. This paper demonstrates that, given a simple network and a node-like diagram, it is often difficult to answer the simplest of questions, such as which actor has the most interactions. In addition to this, studies described in this paper have noted that the way the node-link representation is positioned has an effect on the way the graph is perceived.

In the author's opinion, the wide-spread use of node-link diagrams can be explained by their visual appeal, for example, there is something mesmerizing about watching a force-directed node-link visualization wriggle and adjust to find an optimal layout. On this basis, node-link representations may well be the best choice to convey the results of analysis but may not always be the best choice for conducting visual exploratory analysis. The PAD approach enables answering simple questions about the network from the visualization itself. It also allows the analyst to explore the network over time from the point of view of different actors in the network.

This paper subjectively evaluated the PAD approach with the use of a rudimentary implementation of it in the TIPAD application. The PAD approach makes simple network features, such as degree, immediately visually discernable. The studies and research referenced in this paper contributed to the comparative assessment of the PAD visualization approach using common task taxonomy and visually comparing alternative representations of the same network. A summary of this assessment is presented below in Table 1. The evaluation presented in this table was based on the network of the movie *The Matrix* and the alternative representations of the network presented in this paper. In addition, general utility of other approaches were garnered from the papers cited herein.

This assessment is very subjective however and is subject to a great deal of variation dependent on the properties of the network being analyzed. However, in general it is possible to say, for example, that the standard matrix representation is not effective in facilitating path following. No controlled experiments have been conducted to validate any of the assertions presented in this paper and therefore Table 1 serves only to orient the user to which tasks the author perceives the PAD approach being particularly applicable and where it sits in the visualization space.

The PAD approach does have a number of limitations that have yet to be explored. One drawback is its currently-limited value in assisting in the analysis of network topology characteristics or evolution. In addition, although scrolling is utilized to view the full extent of the network, large networks will present a problem. However, as described in this paper, this is similar to the problems faced when trying to visualize large networks with node-link diagrams. One can compare the figures showing the Davis et al. dataset to see the visual complexity presented of the two approaches is similar on small scale networks. On a large scale network it might be possible to use techniques borrowed from other fields, such as aggregating and drill down techniques.

When discussing large networks, the simple advantage of a well implemented PAD approach may well be that it offers a linear rendering of time, compared to force-directed layouts that attempt to

minimize a global energy function and are generally considered to have a running time equivalent to the number of nodes cubed $O(V^3)$.

	# of nodes	# of incoming or outgoing links	Find adjacent nodes	Can be reached in # steps	Find middle-man nodes	Find clusters	Find nodes & links with attribute	Find, select node	Find, select link	Find short-est path	Show dynamic aspects
PAD	✓	estimate relative weight & read value	interaction-showing adjacent to adjacent	ease is dependent graph size and density	✗	✗	possible with color and size additions	✓	✓	✗	✓
Node-link	not as good matrix or PAD	Keller notes it can be hard, can't estimate as easily as PAD	ease is dependent graph size and density	ease is dependent graph size and density	✓	✓	possible with color and size additions	full 2D space search	ease is dependent graph size	✓	possible with animation
Matrix	✓	✓	✓	link tracing is hard	✗	might necessitate matrix reordering	possible with color and size additions	✓	not as easy as PAD	✗	possible with animation
Bi-partite graph	PAD better	not as easy as PAD	ease is dependent graph size	ease is dependent graph size	✗	✗	possible with color and size additions	✓	✓	✗	might be possible with animation

Table 1 – Visualization evaluation

The TIPAD visualization has some inherent advantages that flow from this alternative view that may not be immediately apparent. The use of this tool during evaluation has shown the usefulness of scrolling while maintaining focus on a particular link line to quickly find the scene of interest. In addition, the ability to incorporate and visualize contextual data in the same view is considered to be of value to analysts.

The ability to distinguish patterns of interaction with a particular actor is somewhat unique. 'Patterns of interaction' is a commonly used phrase in dynamic network analysis but very rarely do visualizations provide the ability to display the patterns in an obvious way. More often than not, these 'patterns' are a result of a cognitive process and are constructed in the analyst's head.

The PAD approach presented in this paper represents a viable alternative to the use of node-link diagrams to represent small temporal network graphs. The use of this approach was demonstrated in the TIPAD application to visualize character interactions in a movie and tested for its ability to service the low level tasks commonly required of such a visualization. There is much scope to extend the PAD idea starting with the use of simple filters-based contextual data (such as keywords) or time periods of activity.

The PAD approach is a general concept that can be applied more broadly to the visualization of network graphs in a number of domains. We believe the PAD approach suits data that is of a temporal nature that can ideally be divided into discrete sessions or bursts. Using the PAD approach offers an alternative view on data that will likely reveal new insights and foster new techniques for extracting meaning from such data sets.

Data sets of this nature include online communication interactions including email, discussion forums, and blog comments. The approach might also be applied to meetings, parliamentary sessions, legal cases or even football games based on 'plays.'

Future Work

The assessment of the utility of the PAD approach and the TIPAD application are the author's subjective assessment based on previous studies related to the task taxonomies and causal observation of users using the tool. However, experimentation is required to validate the assertions in this paper and we plan to undertake experimental activities in the future.

The PAD approach based on its use in the TIPAD application seems to offer some promising visualization possibilities particularly on 2-mode bipartite networks with a temporal dimension. Although created to test the PAD idea we consider the TIPAD application to be a very valuable tool in which to view movie scripts. We believe we have only scratched the surface of how this tool can be utilized for movie analysis and scheduling purposes. For example, the 'pattern of interactions' discussed in this paper could be modified so that one could specify the beginning and end points of a visualization. Social and graph theoretic measures could also be incorporated and visualized within the PAD application.

The scrolling and panning method to visualize linear temporal ordering shows promise and we hope to show how this simple approach might be applicable to other domains. To this end, we intend investigating the applicability of the PAD approach to visualizing and analyzing the discussions in online forums. The posting in forums can be categorized into thread themes and generally have a sequential ordering in the timing of postings. Visualizing the temporal communications patterns after applying a keyword filter could be particularly useful for detecting suspicious activity (Gloor, Niepel et al. 2006) and we hope demonstrate this. Viewing the patterns of interactions in this way may be of interest to security, intelligence and forensic analysts.

References

Alexander, M. (2005). *Using the Bipartite Line Graph to Visualize 2-Mode Social Networks*. North American Association for Computational Social and Organizational Science (NAACSOS). Indiana: Notre Dame.

Alexandre, D. S. and J. M. R. S. Tavares (2010). *Introduction of Human Perception in Visualization*. FEUP - Artigo em Revista Internacional.

Battista, G. D., P. Eades et al. (1998). *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall.

Bender-deMoll, S. and D. A. McFarland (2006). "The Art and Science of Dynamic Network Visualization." *Journal of Social Structure* 7 (2). Available: http://www.cmu.edu/joss/content/articles/volume7/deMollMcFarland/ [December 2011].

Borgatti, S. P., M. G. Everett et al. (2002). UCINET 6 for Windows: Software for Social Network Analysis. Harvard Analytic Technologies.

Brandes, U., J. Raab et al. (2001). "Exploratory Network Visualization: Simultaneous Display of Actor Status and Connections." *Journal of Social Structure* 2 (4). Available: http://www.cmu.edu/joss/content/articles/volume2/BrandesRaabWagner.html [December 2011].

Cole, H. R. and J. H. Haag (1988). *The Complete Guide to Standard Script Formats: The Screenplay*. California: CMC Publishing.

Cui, W. (2007). A Survey on Graph Visualization. PhD Qualifying Exam (PQE) Report. Hong Kong, Computer Science Department, Hong Kong University of Science and Technology.

Davis, A., B. Gardner et al. (1941). *Deep South*, Chicago University Press.

Donath, J., K. Karahalios et al. (1999). *Visualizing Conversation*. Thirty-Second Annual Hawaii International Conference on Systems Sciences, Maui, Hawaii, IEEE.

Dwyer, T., B. Lee et al. (2009). "A Comparison of User-Generated and Automatic Graph Layouts." *IEEE Transactions on Visualization and Computer Graphics* 15 (6).

Freeman, L. C. (2000). "Visualizing Social Networks." *Journal of Social Structure* 1 (1). Available: http://www.cmu.edu/joss/content/articles/volume1/Freeman.html [December 2011].

Garey, M. R. and D. S. Johnson (1983). "Crossing Number is NP-Complete." *SIAM Journal on Algebraic and Discrete Methods* 4: 312-216.

Ghoniem, M., J. D. Fekete et al. (2005). "On the Readability of Graphs Using Node-link and Matrix-based Representations: A Controlled Experiment and Statistical Analysis." *Information Visualization* 4 (2): 114-135.

Gloor, P. and Y. Yhao (2004). *TeCFlow: A Temporal Communication Flow Visualizer for Social Networks Analysis*. ACM CSCW Workshop on Social Networks, Chicago.

Gloor, P. A., R. Laubacher et al. (2004). *Temporal Visualization and Analysis of Social Networks.* North American Association for Computational Social and Organizational Science (NAACSOS).

Gloor, P. A., S. Niepel et al. (2006). Identifying Potential Suspects by Temporal Link Analysis, MIT CCS.

Goud, R. (2006). "Visualization of File Relations in Software Systems with Adjacency Matrices." Department of Mathematics and Computer Science. Eindhoven, Technische Universiteit Eindhoven. Master's Thesis.

Harris, C., R. B. Allen et al. (1999). Temporal Visualization for Legal Case Histories, HCIL Technical Report No. 99-18, Computer Science Department, University of Maryland.

Healey, C. G., K. S. Booth et al. (1995). "Visualizing Real-Time Multivariate Data Using Preattentive Processing." *ACM Transactions on Modeling and Computer Simulation* 5 (3): 190-221.

Heer, J. M. (2004). "Prefuse: A Software Framework for Interactive Information Visualization." Department of Electrical Engineering and Computer Sciences. Berkeley, University of California. Master of Science.

Henry, N. (2008). Exploring Social Networks with Matrix-based Representations. Sydney, University of Sydney. PhD.

Huang, W., S.-H. Hong et al. (2006). *How People Read Sociograms: A Questionnaire Study*. Asia-Pacific Symposium on Information Visualization (APVIS 2006), Tokyo, Australian Computer Society.

Huang, W., S.-H. Hong et al. (2007). "Effects of Sociogram Drawing Conventions and Edge Crossings in Social Network Visualization." *Journal of Graph Algorithms and Applications* 11 (2): 397-429.

Kadaba, N. R., P. P. Irani et al. (2007). "Visualizing Causal Semantics Using Animations." *IEEE Transactions on Visualization and Computer Graphics*.

Kang, H., C. Plaisant et al. (2007). "NetLens: Iterative Exploration of Content-actor Network Data." *Information Visualization* 6: 18-31.

Karahalios, K. G. and F. B. Viégas (2006). *Social Visualization: Exploring Text, Audio, and Video Interaction*. Conference on Human Factors in Computing Systems, CHI '06, Quebec, Canada.

Keller, R., C. M. Eckert et al. (2006). "Matrices or Node-link Diagrams: Which Visual Representation is Better for Visualizing Connectivity Models?" *Information Visualization* 5: 62-76.

Lee, B. (2006). "Interactive Visualization for Trees and Graphs." University of Maryland. PhD.

McGrath, C., J. Blythe et al. (1997). "The Effect of Spatial Arrangement of Judgments and Errors in Interpreting Graphs." *Social Networks* 19 (3): 223-242.

Moody, J., D. McFarland et al. (2005). "Dynamic Network Visualization." *American Journal of Sociology* 10 (4).

Moreno, J. L. (1953). Who Shall Survive: Foundations of Sociometry, Group Psychotherapy, and Sociodrama. Washington: Beacon House.

Munzner, T. (2000). "Interactive Visualization of Large Graphs and Networks." Department of Computer Science, Stanford University. PhD.

Mutton, P. (2004). *Inferring and Visualizing Social Netoworks on Internet Relay Chat*. Eighth International Conference on Information Visualisation (IV'04), IEEE.

O'Madadhain, J., D. Fisher et al. (2010)."JUNG (Java Universal Network/Graph Framework)."

Plaisant, C. (2004). *The Challenge of Information Visualization Evaluation*. Working conference on Advanced Visual Interfaces (AVI 2004), Gallipoli, Italy. New York: ACM Press.

Plaisant, C., B. Milash et al. (1996). *Lifelines: Visualizing Personal Histories*. Conference on Human Factors in Computing Systems: Common Ground, SIGCHI, Vancouver, British Columbia.

Shneiderman, B. (1996). "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations." *IEEE Symposium on Visual Languages* 1: 336.

Tat, A. and S. Carpendale (2006). *CrystalChat: Visualizing Personal Chat History*. 39th Hawaii International Conference on System Sciences.

Tversky, B., J. G. Morrison et al. (2002). "Animation: Can It Facilitate?" *International Journal of Human-Computer Studies*.

Viégas, F. B. and J. Donath (2004). "Social Network Visualization: Can We Go Beyond the Graph?" *Workshop on Social Networks, CSCW* 4: 6-10.

Ware, C. (2000). *The Visual Representation of Information Structures*. 8th International Symposium on Graph Drawing.

Weng, C.-Y., W.-T. Chu et al. (2007). *Movie Analysis Based on Roles' Social Network*. IEEE International Conference on Multimedia and Expo, Beijing.

Winn, W. (1994). "Contributions of Perceptual and Cognitive Processes to the Comprehension of Graphics." *Comprehension of Graphics* (3): 27.