Metric-Based Network Exploration and Multiscale Scatterplot

Yves Chiricota*
Université du Québec à Chicoutimi, Canada

Fabien Jourdan, Guy Melançon†
LIRMM UMR CNRS 5506, Montpellier, France

ABSTRACT

We describe an exploratory technique based on the direct interaction with a 2D modified scatterplot computed from two different metrics calculated over the elements of a network. The scatterplot is transformed into an image by applying standard image processing techniques resulting into blurring effects. Segmentation of the image allows to easily select *patches* on the image as a way to extract sub-networks. We were inspired by the work of Wattenberg and Fisher [21] showing that the blurring process builds into a multiscale perceptual scheme, making this type of interaction intuitive to the user. We explain how the exploration of the network can be guided by the visual analysis of the blurred scatterplot and by its possible interpretations.

CR Categories: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques I.3.3 [Computer Graphics]: Picture / Image Generation—Viewing algorithms I.4.3 [Image Processing]: Enhancement—Smoothing

Keywords: Graph navigation, exploration, scatterplot, multiscale perceptual organization, clustering, filtering, blurring

1 INTRODUCTION

Part of the research activity in Information Visualization is devoted to exploratory techniques [4, 12]. Indeed, when designing a tool it is important to distinguish whether the user is facing familiar data and is actually using it for a specific task (annotating it or consulting it, for instance) or if she/he is exploring the data trying to find patterns or correlate the view with some other properties.

The results we describe in this paper contribute to exploratory techniques for graph visualization. Graphs appear as natural entities when modeling social networks, access graphs in software engineering, web (sub)graphs, for example. When faced with a complex network, due either to its size or to its intrinsic structure, a common approach is to exploit structural properties to gain insight on the graph. The work by Page *et al.* [17] and that of Kleinberg [13] are good examples of this methodology applied to the study of web graphs.

A common design is to map structural metrics to graphical cues (see [12, Section 4] for a list of references). It is often claimed that graphically mapping metric values onto the view of the graph produces a valuable effect by underlining the backbone of its structure. Various techniques have been proposed to act on the way structural metrics are mapped on nodes or edges of a graph. Colour maps usually assign hue and intensity to nodes or edges according to their associated metric values.

Metrics can also be used to filter out elements of a lesser importance (with respect to the metric) by mapping its range of values onto a *range slider* ([5]), thus providing an effective way to isolate a subset of interest in the graph. A common and useful application is to specify a threshold by moving the cursor down (or up) and filter out nodes or edges with a value above (or not exceeding) the threshold. This hiding method gains effectiveness when coupled with a colour map as the elements that are filtered out have a lighter hue and/or lesser intensity, are thiner, etc.

The use of multiple range sliders can help the exploration of a dataset by filtering elements based on a combination of criterion. Williamson and Schneiderman [24] have successfully applied this technique when exploring a real estate database, enabling a user to specify a price range and number of bedrooms, for instance. Barry Becker's MineSet [2] is a tool supporting the exploration of multidimensional databases, helping the user to navigate the data through the selection of range values on several dimensions.

It is unclear whether range selectors are as effective when dealing with less intuitive metrics. What if the values correspond to a *theoretical measure* computed over all nodes of the network, such as for example the so-called clustering index used to define small world networks [22, 23] or the pagerank index of web pages [17] ? What if the values are unevenly distributed over the range they cover ? How should a user manipulate the range selectors to correctly monitor the threshold (filter) ? These observations become even more relevant when dealing with two-dimensional metrics. Situations that are hardly predictable may appear where one slider requires finer tuning depending on the values that were selected using the other. Section 2 provides examples and a more detailed discussion on these issues that were one of the starting point of our work.

The technique we put forward in this paper gives the user direct access to the 2D set of values through a *modified* scatterplot view. More precisely, the view the user acts on is obtained from the actual scatterplot after it is blurred, in order to enforce perception and ease selection of significant regions. By dynamically linking regions to the graph layout the user gets immediate feedback when browsing the blurred scatterplot. This appears as a relevant exploration technique since the interpretation of the regions forming the blurred scatterplot is natural to the user. This claim relies on the recent work by Wattenberg and Fisher [21] where they develop a multiscale model of perceptual organization in information graphics. In their paper, Wattenberg and Fisher apply image segmentation to a sequence of blurred images, allowing them to build a hierarchy reflecting the user's internal model. Our experience using the blurred 2D scatterplot for navigating graphs can be seen as a confirmation of Wattenberg and Fisher's hypothesis.

Moreover, giving the user control over the parameter of the blurring process provides helpful support for the identification of regions of interest in the scatterplot. Varying the parameters allows the user to travel across scales. Irregular regions emerging from the blurring process that could otherwise hardly be identified can be selected by a simple mouse click.

2 METRIC-BASED EXPLORATION

The layout of a network, although requiring significant effort, is often not sufficient to help a user when she/he is exploring its underlying data. A common technique is to map network attributes to graphical cues such as hue and/or intensity, node size, edge thickness, etc. We shall be concerned in this paper with metric-based exploration, that is when the exploration itself is guided by observed

*e-mail: ychirico@uqac.ca
†e-mail:{fjourdan, Guy.Melancon}@lirmm.fr

IEEE Symposium on Information Visualization 2004
October 10-12, Austin, Texas, USA
0-7803-8779-1/04/$20.00 ©2004 IEEE

properties of the metric. For instance, the structure of the network might be questionned to explain why a metric does not distribute uniformely over its range. In [11], it is argued that the statistical distribution should be taken into account to faithfully map structural metric values to colours, thickness, etc. Indeed, although the values may vary over an interval, it may well occur that most of the values reached by nodes or links are clustered at one end of the interval, for instance. Taking the statistical distribution of metric values into account avoids mapping a majority of graph elements into a small range of the colour spectrum, making it hard for the human eye to distinguish between them (see [11] for a detailed discussion). The same argument applies when metric values are mapped onto a range slider. Mapping the tick marks underlying the slider according to the statistical distribution prevents the slider from behaving in an unexpected way. More precisely, suppose a metric is computed over a graph and that the range of values distributes as indicated by the curve in Figure 1. That is, the y-axis on the the curve indicates the percentage of elements having a value below a given threshold (x-axis). In the example, we see that approximately 75% of the elements have a value below 1.5, although the range of values cover the interval $[1,4]$. This type of curve occurs when dealing with metrics such as the pagerank [17] or clustering index on nodes [23] or edges of a network [1].

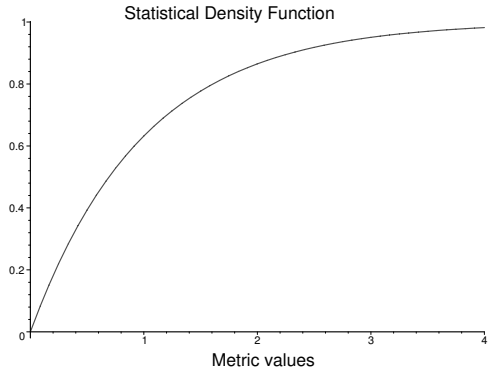

Figure 1: Statistical density function computed from metric values associated with nodes or edges of a network.

Linearly mapping metric values onto a slider would produce undesirable effects since even tiny cursor movements of the slider over the first half of the slider range [0, 1.5] would provoke much larger effects than similar movements over the rest of the range [1.5, 4.0].

It is however incorrect to present this situation as being a problem: it is precisely this property of the metric that places it into the focus of the exploration. Indeed, discovering how the graph elements map onto the metric can reveal part of the network's dynamic. As an example, we can cite our recent work on the navigation of small world networks. The study of a metric computing the *strength of edges* in a network lead us to design a useful clustering technique for visual exploration [1].

2.1 Two-dimensional metrics

The problem we underline, and the precaution we suggest to take to prevent from it, becomes even more intricate when dealing with two-dimensional metrics. These metrics can either be intrinsically 2D, or can be formed by joining two metric values into a 2D point. The latter case is not uncommon and occurs when mixing a structural metric with a contextual one. The network in Figure 2 provides an example. The network itself corresponds to a set of links between the various Java classes of the "Resyn Assistant API"[1].

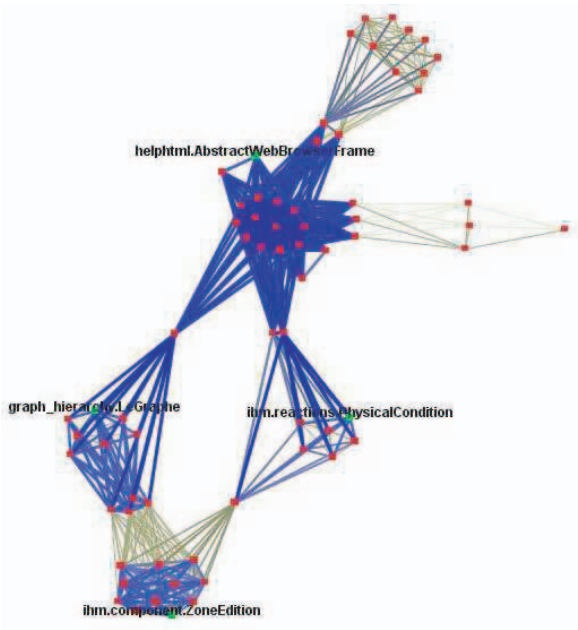

Figure 2: Example network : access graph induced from Java classes in the Resyn Assistant API. Edges have been coloured according to an application metric specific to Resyn.

Edges have been assigned a colour according to an application specific (edge) metric measuring the degree of logical dependencies between classes. The interest of the visualization in this case is to confront the application metric with the cluster structure. On the other hand, the cluster structure is predicted by the strength metric computed for all edges of the network : edges with low values are weak in that they do not contribute to the cohesiveness of their neighbourhood (see [1] for more details). Weak edges thus correspond to gateway routes linking distinct neighbourhoods.

The interest here is to understand how the application specific metric relates to the cluster structure and thus to the strength of edges. The 2D scatterplot in Figure 3 displays how these two metrics relate to each other and distribute over the edges of the network. More precisely, points of the scatterplot correspond to edges of the network where the x and y coordinates of the point are respectively given by the application specific metric and the strength metric associated with the edge. Points are also assigned varying (greyscale) intensity reflecting the frequencies since many edges are associated with the same (x,y) pair of values[2].

2.2 Exploring the network through the scatterplot

The goal is to offer the user a tool to explore the network while being guided by the scatterplot. A solution could be designed based on a combination of two range sliders, one for each dimension, allowing the user to select a rectangular subregion of the scatterplot before examining which part of the network it corresponds to. Note that, in some cases, this can be a satisfactory interaction technique,

[1] "Resyn Assistant" is a software designed for the study of chemical components developed in Montpellier (LIRMM). See `http://www.lirmm.fr/~resyn/presresyn.html`

[2] The image provided here does not support our claim satisfactorily – mainly because of its small size and low resolution. However, the technique we describe in the next section precisely overcomes that type of problem by suggesting to use a different view of the scatterplot.

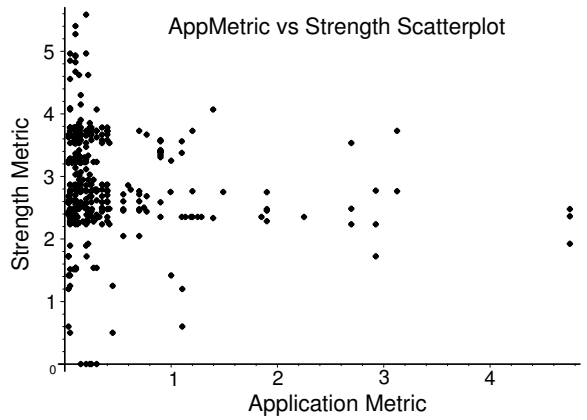

Figure 3: A 2D scatterplot displaying how the strength and application specific metric distribute over the edges of the network in Figure 2.

as assessed by the example from Williamson and Schneiderman [24]. We believe that this is partly due to the fact that users are familiar with the price range in the real estate market and with its relation to other discrete metrics such as the number of bedrooms. In other words users are able to predict how the number of bedrooms impacts on the price of a house.

However, range sliders may appear as being limitative when dealing with less intuitive metrics. The study of social networks or hypermedia structures, for example, often relies on the use of structural metrics such as the clustering index introduced by Watts [22, 23], the metric by Kleinberg allowing the identification of authorities and hubs [13], or such as the pagerank index [17]. It is precisely the correlation between these structural metrics and other contextual metrics such as the size of documents or publication date, etc., that is focused on here.

Note that the limitation we stress does not come from the sliders themselves but rather from the fact that they only enable the selection of rectangular subregions. Brushing the scatterplot using a rectangular brush, or using a rectangular magnifier glass, bear the same inconvenient. Moreover, part of the problem resides in the difficulty to read and interpret the scatterplot.

3 Blurred Scatterplots

We have worked at developing a technique allowing an easier and more flexible selection of points in the scatterplot, by anticipating on the user's interpretation of the plot, while synchronizing the interaction on the view of the network itself.

As one can see, the majority of points in the scatterplot of Figure 3 are located on the left of the diagram and are organized into what seems to look like one or two bigger subgroups. This *unsatisfactory* description of the scatterplot actually points at an important aspect of metric-based exploration. The user needs to be guided not only in her/his discovery of the network, but also when reading the plot in order to accomplish a more accurate and significant selection of subregions (and/or sub-networks).

We follow the work by Wattenberg and Fisher [21] and claim that the user interprets the scatterplot through a multiscale perceptual scheme. That is, at a higher level the scatterplot is perceived as forming a few clouds of points. At a finer level of perception, a cloud may well be seen as being organized into smaller subcomponents of varying shapes and sizes. Wattenberg and Fisher develop their model by constructing a hierarchy or *layers of information* reflecting the different levels of details of each component of an image. The hierarchy is built from the image by applying a standard information graphics technique we now describe. The image itself – think of the scatterplot as being the image – can be considered as a map $f : [0,1] \times [0,1] \rightarrow [0,1]$. Convoluting the image with a Gaussian kernel G_s produces a new image $f_s = G_s * f$, representing the original image after it has been blurred by a factor s. Figure 4 shows an example of this blurring effect when applied to the scatterplot of Figure 3. (The view is zoomed on the subset sitting at the left and mid-height of the plot.)

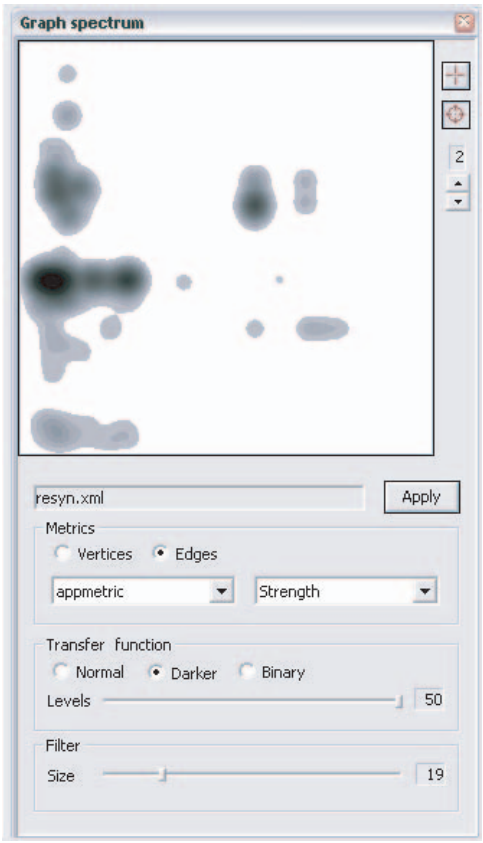

Figure 4: Convoluting the image in Figure 3 with a Gaussian filter produces a blurring effect and helps localize the regions of interest in the scatterplot.

The blurring effect produces exactly what is anticipated and wanted. Points forming dense subregions slowly melt into larger and uniform grey patches which can then be identified by standard image segmentation techniques. The benefits of these operations are tremendous. After setting only two parameters, the user can select a subregion by a simple mouse click, disregarding its complexity or irregular geometry. Incidentally, it might well be the irregularity of a subregion that triggers an interest in it. Again, observe that the grey patches in Figure 4 could hardly be selected using a rectangular device[3].

[3]Note that it would be more accurate to say that the selection of an irregular region is hard to achieve using any device having a regular geometrical shape (be it rectangular – as with the sliders, or round – a brush, etc.).

3.1 Gaussian filter and image segmentation : point locations and intensities

Two ingredients are taken into account when computing the gaussian filter. The proximity of points is not the only parameter to affect on the final "grey patches". Points are also assigned an intensity depending on the frequencies of value pairs (x,y). That is, the more often nodes or edges are mapped onto a pair (x,y) the darker the associated point is. This intensity obviously has an effect when computing the convolution $G_s * f$. Figure 5 can be compared to Figure 4 to evaluate the effect of a change in the value of s: a greater value of s creates wider patches and the shape of the nested strips extends from the already computed subregions. The variations in greyscale indicates how the point are distributed into a subregion, according to their coordinates and frequencies.

In order to make the subregions selectable, we perform a segmentation of the image f_s. A contour is detected where there is a significant change in greyscale. The segments identified in this manner correspond to nested curved strips, each strip gathering points with an intensity varying in a greyscale sub-interval. The subregions then correspond to a set of strips having their intensity above a given threshold. Note that in doing this, we make it possible to select a patch together with all of its interior, since a strip sitting inside the interior of another contains points with a greater intensity (patches are organized around the points with greater intensity in the scatterplot, which themselves sit at the core of denser subregions).

The number of strips corresponds to a parameter that has to be set before performing the segmentation, specifying into how many sub-intervals the greyscale should be divided. Subregions join and separate according to variations of the parameter s. Note that the number of strips also affects the final image. Wattenberg and Fisher have looked at this aspect in great details, explaining why the hierarchy they compute can sometimes result in an acyclic directed graph [21, Section 2.3.3]. Varying the two parameters allows the user to travel across scales.

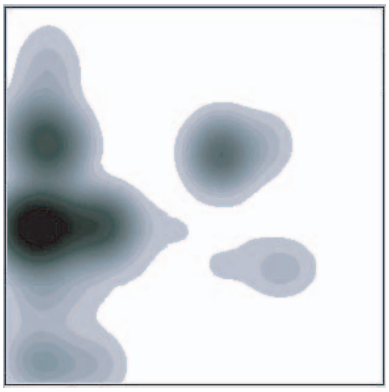

Figure 5: A greater value of s creates wider patches (when compared with Figure 4). The shape of the nested strips extends from the existing subregions.

4 A TWO WAY INTERACTION

We have so far explained the benefits that can be expected from interacting directly on a blurred scatterplot, making it more intuitive and easier to select subregions corresponding to subsets of metric values reached by the network elements. However, the goal is to help the user navigate and explore the network itself, based on what can be inferred from the metric values. In other words, the analysis of the blurred scatterplot naturally raises questions such as: "what are those nodes or edges having a maximum/average value ?" or "where are they located in the network ? do they form a single connected component or do they split into several clusters ?", and so on. The next section provides examples showing how significant structure can be inferred on the network by interacting on the blurred scatterplot.

The questions or tasks suggested here might well occur in the opposite direction. Looking at the layout of the graph, one might wish to test hypothesis on the metric values associated with a given part of the network. That is, the analysis of the network can be conducted by looking at the relative location of nodes or edges, and be guided by any interpretation the two metrics might have individually or when composed together into a scatterplot. In other words, it might be interesting to test whether two similarly looking neighbourhoods are mapped to a same region in the scatterplot, and if not infer reasons that can explain this difference.

The implementation of our technique offers all those avenues. Subregions can be selected (multiple selection) automatically triggering the selection of the associated clusters or neighbourhoods on the graph layout. Conversely, the selection of any cluster in the graph highlights the corresponding patches containing the (x,y) pairs associated with the selected elements. The exploration can thus be realized through a scenario of back and forth adjustments between the laid out network and associated blurred scatterplot.

4.1 Worked examples

The technique has been implemented within an existing graph visualization application. After loading the network into memory, the user can select either node or edge metrics defining the scatterplot which is then blurred and segmented according to default or customized values. We will look at two different kind of examples. First, we have applied our technique to networks with an already known structure, in order to assess the predicted benefits. The last example, consisting of protein-protein interaction networks, illustrates the exploration scenario within which the technique proves to be helpful.

4.1.1 The Resyn Assistant API

The Resyn Assistant API has been at the focus of previous work on clustering techniques developed in the context of software reverse-engineering [6]. Consequently, we used the Resyn network as a benchmark to test whether already known facts about the structure of the API could be recovered using the technique developed here.

In this example, we built a scatterplot computed over two metrics, one structural, the other contextual. The structural metric we used is the *strength of edges* indicating whether a link acts as a weak link between two distinct neighbourhoods, or whether the edge sit at the core of a cluster. We had already shown how this metric can be used to easily compute meaningful clusters (the significance of which had been assessed by the designers of the API themselves, see [6, Section 3.1]). The Resyn specific metric is a weight measuring how much of the class (attributes and methods) is publicly accessible. (This is Java terminology, the metric is not specific to Resyn itself but to software engineering. For more details see [10]).

Figure 9 shows a snapshot of our application when loaded with Resyn together with a zoomed view on the scatterplot. Selecting a darker patch (marked as yellow when selected) automatically extract clusters (left view). The pop-up item list shows the label of all selected nodes.

The selected patch in Figure 6 corresponds to edges simultaneously having an associated average strength value (y coordinate) and a low application specific metric value (x coordinate). Now, an edge with an average strength value necessarily sits in the middle of a moderately connected cluster. On the other hand, the application metric reflects how much a class spreads its services (public

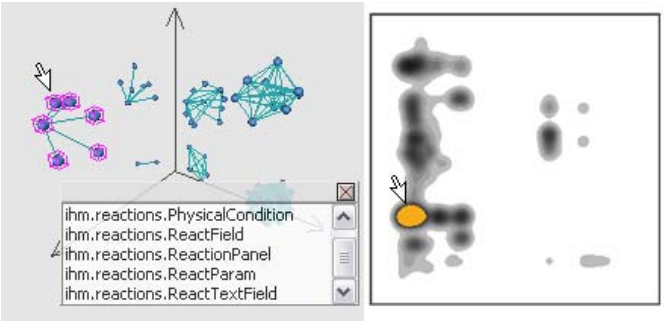

Figure 6: Darker patches in the scatterplot can be selected to recover the package structure of the Resyn Assistant API.

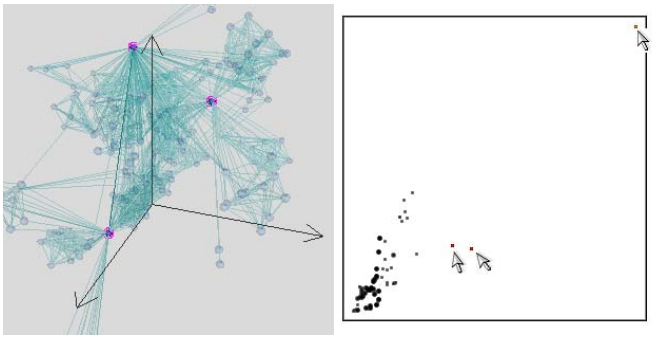

Figure 7: Selecting the three "hub" actors (Cruise, Carrey and Diaz) on the left view highlights their corresponding subregions or patches in the blurred scatterplot (right panel).

or protected attributes and methods) to other members of the API. Hence, by selecting the yellow patch, we seek to find classes that offer services to only a few other API members and that are part of moderately connected clusters. Hence, we expect these classes to live in periphery of more densely connected clusters, which is actually the case, the task being to discover which clusters are involved in this type of possibly unwanted or unpredicted situations.

4.1.2 Internet Movie Database

As another "proof-of-concept" example, we have extracted a sub-network out of the internet movie database (IMDB[4]), by selecting actors and movies stemming from three well known actors (Tom Cruise, Jim Carrey and Cameron Diaz). We have used this time two structural metrics on nodes that lead us to easily identify "hub" actors in the network, from the visual analysis of the lightly blurred scatterplot (low s value; see the right panel of Figure 6). The two metrics used are the *eigenvector metric* derived from the adjacency matrix and the *degree* of nodes. The eigenvector metric can be interpreted as a centrality measure, indicating the proportion of paths going through a node. The left panel of Figure 7 shows the three actors around which this IMDB sub-network is organized (the nodes are surrounded by the pink transparent boxes), which obviously act as local *hubs* (we borrow this terminology from [13]). The mouse pointers on the right panel indicate the subregions of the blurred scatterplot that were highlighted after the hub nodes were selected in the left panel. Observe that the hubs seem to have their own individual patches in the scatterplot, spread towards the right of the diagram. It is worth mentionning that a drill down exploration of the multiscale scatterplot was actually necessary in order to bring us to this situation and achieve a clean separation of the patches. The fact that each hub has its own patch does not come as a surprise since those nodes have a much larger degree (but each having their own eigenvector value).

This observation lead us to examine other small and isolated patches corresponding to high degree nodes. For instance, we were able to locate the node associated with David Letterman, which obviously has a high degree since he *played* (so to say) in his daily TV program with many actors and act as a hub for all these different persons (belonging to distinct sub-network or communities).

The right panel in Figure 8 shows a zoomed view of the scatterplot with an increase on the s parameter, in order to emphasize the shape of the patches. Obviously this 2D confrontation of the eigenvector and degree metrics ease the identification of clusters in the network[5]. The small patches of the scatterplot actually correspond to either movies or TV programs gathering the three actors. For instance, running a "Joint Ventures" search on IMDB we could verify that the actors occuring in the selected cluster (left panel of Figure 7) all played in the 1985 movie *Legend*. Incidentally, we were able to identify movies or TV programs for all clusters for a majority of patches. The confrontation of the two metrics is essential here: the eigenvector metric helps locating cliques while the degree metric distinguishes between cliques of different sizes.

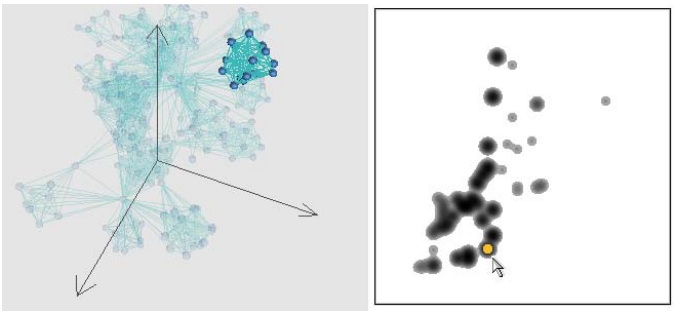

Figure 8: Close-up view of the scatterplot of Figure 6, with an increase in s. This time, a simple click triggered the selection of a whole cluster (connected sub-component).

Finally, we changed the degree metric for the *clustering index* borrowed from the small world theory [23], while keeping the eigenvector metric, which helped us identify the most important actors of this IMDB sub-network. The resulting scatterplot is shown on the right panel of Figure 9. The view itself follows from a sequence of zoom and pan interaction on the scatterplot panel. The actors belonging to the corresponding cluster (selected through a series of shift-mouse click) are Beatty, Nicholson, Huffman, Dunaway, Murphy and Myers. (The three "hub" actors from which the network is built belong to other clusters associated with patches sitting at a distance from the one we consider here.)

4.1.3 Exploring a protein-protein interaction network

Protein-protein networks are currently the focus of intense research in bio-informatics. To put things simply, the challenge for biologists is to understand the structure of those networks, in an attempt to describe the functions of proteins in the organism.

The network we have explored consist of 3278 nodes and 4549 edges where nodes are associated with proteins and links correspond to interactions that have been experimentally observed (not all in a same experiment, of course). The network comes equipped

[4] See the website `http://www.imdb.com`.

[5] As with Resyn Assistant, we had previously studied similar examples [1], so we were able to verify the validity of our conclusions.

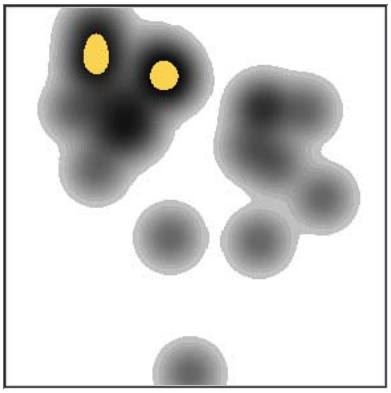

Figure 9: Close-up view of the scatterplot obtained from the eigenvector metric together with the clustering index computed on the IMDB network.

with an interesting contextual information. Each link has an integer attribute measuring the number of published papers explicitly mentioning the two proteins (resulting from a text mining process). We shall call this contextual metric the *co-citation metric*.

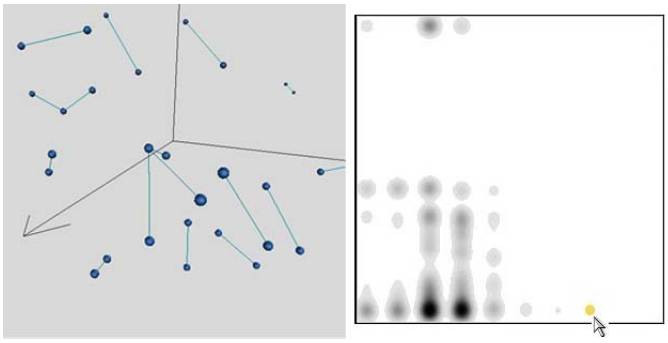

Figure 10: Exploring a protein-protein interaction network based on a scatterplot showing how the co-citation metric relates to the degree of nodes.

When exploring the network, natural questions come to mind. Can we expect the more dense neighbourhood to have been the object of more productive research (in terms of published papers) and consequently to have a higher co-citation measure ? In other words, how is the degree of the proteins incident to an edge related with its co-citation ? The scatterplot in Figure 10 describes how things are organized with respect to degree and co-citation. It apparently turns out that protein pairs having a high co-citation value also have a low degree, as shows the small yellow patch on the lower right part of the scatterplot (patches turn yellow when selected). The subnetwork in the left view corresponds to nodes and links that are being mapped to that subregion of the plot. The view shows a set of separated edges, since interactions involving these proteins have been rarely observed. However, those protein pairs (edges) seem to have been the object of more intense study (further study with biologists is needed to refine and confirm this claim). The interface allows to access the actual name of the selected proteins (which are framed within small pink boxes) displayed in a pop-up item list. (Locating the underlying "co-citation" publications on the PubMed website requires additional work.)

Other observations can be made. A majority of protein pairs seem to have been the object of an average number of 3 or 4 publications (as show the two vertical strips stemming from the darker round patches at the center of the x-axis). However, most of them have a low degree, indicating that interactions involving those proteins were observed only on a few occasions. Also, that the number of higher degree proteins appearing in 1 to 4 publications are comparable, since the pale grey patches at the top of the image all look the same[6].

We pursued our exploration of this network by mixing co-citation with the strength of edges, focusing on stronger edges. One can imagine that stronger edges have a higher co-citation value, because they belong to a tighter neighbourhood. More precisely, the probability that their neighbor proteins (neighbour proteins of the proteins incident to the edge) have been observed to interact is relatively high – this directly follows from the definition of strength, see [1]. Consequently, it seemed reasonable to predict that the two interacting proteins incident to a strong edge have been the object of many papers. However, we were able to locate a strong edge with low co-citation. There is actually one paper addressing issues concerning the two incident proteins DAD1 and SPC34[7]. Moreover, the paper is recent (2002) which partly justifies the low co-citation value of the edge. Interestingly enough, we extended the patches, looking at things from a higher level scale (by increasing s), in order to examine nodes having a close (*co-citation*, *strength*) value pair, and slowly reached a small group of neighbour proteins that were all discussed in the same 2002 paper.

5 Related work

The idea of a direct interaction on a scatterplot goes back to the work of Becker and Cleveland [3]. However, scatterplots are usually a tool for studying the correlation between two variables. Indeed, principal component analysis provides methods and tools for computing and evaluating the tendencies among clouds of points.

5.1 Scatterplot brushing

Becker and Cleveland developed a framework for studying the correlation between several variables through various types of mouse interaction such as brushing. First laying out a set of bivariate scatterplots into a matrix of square boxes (one for each 2D scatterplot, $N(N-1)$ in total if there are N variables), using rectangular brushes they let the user browse part of a scatterplot while highlighting the corresponding points in all the other matrix boxes. By studying the way the brushing acts on the other boxes, the user can then gain insight on the high-dimensional dataset and infer correlation between the dimensions.

In their examples, Becker and Cleveland only show small plots (partly due to computer capabilities in the 80's). The small size of the plot makes it easy to select various subregions with a rectangular brush avoiding phenomenons emerging from unusual statistical behaviors (see Section 2 above). XmdvTool by Ward [20] (see also [16]) investigates brushing and extends the work of Becker and Cleveland, defining exploration strategies on several other types of visualization such as parallel coordinates, glyphs and dimensional stacking display. Ward's XmdvTool certainly is a candidate application to extend our work to more than two dimensions. There might be circumstances where parallel coordinates, for instance, offer a better point of view on the graph's metrics or ease its navigation. More work is needed to determine what types of visualiza-

[6]The study of these diagrams are the focus of current research conducted in collaboration with biologists in Montpellier.

[7]The exact names refer to genes, following the nomenclature used on PubMed. The paper's reference is: Janke C., Ortiz J., Tanaka T. U., Lechner J., Schiebel E. (The Beatson Institute for Cancer Research, CRC Beatson Laboratories, Glasgow G61 1BD, UK.) Four new subunits of the Dam1-Duo1 complex reveal novel functions in sister kinetochore biorientation. *EMBO (European Molecular Biology Organization) Journal*, 2002; 21(1-2) : 181-93.

tions (of N-dimensional spaces) can be explored through a multiscale perceptual scheme.

5.2 Graph splatting

GraphSplatting, introduced by van Liere and de Leeuw [15], transforms a graph into a two-dimensional scalar field rendered as a colour coded map, a height field, or a set of contours. The scalar field is obtained from the layout itself by blurring the image and turning it into a set of connected regions, thus providing density information which can be used to determine the structure of the graph.

Thus, the blurred image here is computed based on the layout, letting the user interact on a scalar field instead of the node and link diagram. The colour map then renders according to the density of neighbourhoods. Metrics can be taken into account by adding a third dimension (height) or by including textures into the view. Note however, that direct interaction with the underlying data of the network is then harder to achieve. Also, in order to make the scalar field meaningful, it is assumed that the graph is laid out using an algorithm reflecting its connectivity or *structure*, which most of the time will be a force-directed method. In our case, the user could well decide that based on her/his exploration, it would be meaningful to layout the graph using a different algorithm. This could be done without loosing the logical correspondence between the view on the blurred scatterplot and the graph (and the already selected elements).

GraphSplatting, as well as our work, shares similar features with Barry Becker's MineSet [2]. Indeed, MineSet is a tool supporting the exploration of multidimensional databases, helping the user navigate the data through the selection of range values on several dimensions. Two dimensions serve to embed the data in 3D space, the other dimensions being rendered as a colour coded Gaussian field. The user can then select part of the data by directly interacting on the displayed surface.

5.3 Density Estimation

The present work suggests that when dealing with plots containing a large numbers of points, interacting with the blurred scatterplots is easier and more intuitive. This enters a chapter widely studied in statistical mathematics called *density estimation* (cf. [19]). Indeed, the computation of the blurred image can be seen as an *ad hoc* technique computing a rough approximation of the density function of the underlying set of points. Incidentally, the computation of standard contour curves would support the analysis of the plot equally (which we determine when computing the levels of gray in the image).

The work by Ester *et al.* [7], also enters this scope. They define a strategy for computing clusters in spatial database based on a density analysis of points in 3D.

The work by Fua, Ward and Rundensteiner [8] suggests how to take the structure of the data under study into account and adapt the brush, turning the selection process into what they named *structure-based brushing*.

In the present work however, the focus is less on the analysis of the metrics than on the ability to explore and interact on its multiscale structure while selecting and navigating parts of the network under study.

6 Conclusion and perspectives

The technique presented here combines direct interaction on a modified scatterplot together with a view of a network. Inspired by the work of Wattenberg and Fisher [21], we consider the scatterplot as an image that we then transform using standard image processing techniques. The blurred images build into a multiscale diagram appearing as an intuitive object that can be easily interpreted and acted on by the user. The original scatterplot is obtained by assigning *x*-*y* coordinates to each of the network's element, the *x* and *y* coordinates being computed from metrics relevant to the network under study. A common situation is to confront a structural metric with a contextual one.

The studied examples assess of the usefulness of the technique. The technique proves as an interesting direction to pursue for studying protein-protein interaction networks. Indeed, those networks come equipped with a large inventory of contextual attributes (numerical and non numerical), apart from the metrics that can be computed from the structure of the network itself. It is precisely in this type of situations that our technique reveals itself as the most useful: the network needs to be *explored* based both on its structure and on contextual information.

Acknowledgements. We wish to thank the anonymous referees for their acute and constructive remarks which helped us improve the final version of the paper.

References

[1] D. Auber, Y. Chiricota, F. Jourdan, and G. Melançon. Multiscale visualization of small-world networks. In Stephen C. North and Tamara Munzner, editors, *IEEE Information Visualization Symposium*, Seattle, USA, 2003. IEEE Computer Press.

[2] Barry G. Becker. Volume rendering for relational data. In *IEEE Symposium on Information Visualization*, 1997.

[3] Richard A. Becker and William S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[4] S.K. Card, J.D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization*. Morgan Kaufmann Publishers, San Francisco, 1999.

[5] D. A. Carr, N. Jog, and H. P. Kumar. Using interaction object graphs to specify and develop graphical widgets. Technical Report CS-TR-3344, University of Maryland, 1994.

[6] Y. Chiricota, F. Jourdan, and G. Melançon. Software components capture using graph clustering. In *11th IEEE International Workshop on Program Comprehension*, Portland, Oregon, 2003. IEEE / ACM.

[7] Martin Ester, Hans-Peter Kriegel, Jrg Sander, and VXiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *2nd International Conference on Knowledge Discovery and Data Mining (KDD 96)*, pages 226–231, Menlo Park, CA, 1996. AAAI Press.

[8] Ying-Huey Fua, Matthew O. Ward, and Elka A. Rundensteiner. Navigating hierarchies with structure-based brushes. In Graham Wills and Daniel Keim, editors, *IEEE Symposium on Information Visualization (InfoVis '99)*, pages 58–64. IEEE CS Press, 1999.

[9] D. Gibson, J. Kleinberg, and P. Raghavan. Inferring web communities from link topology. In *HyperText '98*, pages 225–234, Pittsburgh, PA., 1998.

[10] Olivier Gout, Gilles Ardourel, and Marianne Huchard. Access graph visualization: A step towards better understanding of static access control. In Tom Mens, Andy Schrr, and Gabriele Taentzer, editors, *Electronic Notes in Theoretical Computer Science*, volume 72. Elsevier, 2002.

[11] Ivan Herman, M. Scott Marshall, and Guy Melançon. Density functions for visual attributes and effective partitioning in graph visualization. In Steven F. Roth and Daniel A. Keim, editors, *IEEE Symposium on Information Visualization (InfoVis'2000)*, pages 49–56, Salt Lake City, Utah, U.S., 2000. IEEE Computer Society.

[12] Ivan Herman, M. Scott Marshall, and Guy Melançon. Graph visualisation and navigation in information visualisation: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.

[13] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[14] Jon M. Kleinberg, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew S. Tomkins. The web as a graph: Measurements, models and methods. In *International Conference on Combinatorics and Computing (COCOON'99)*, pages 1–17, 1999.

[15] Robert van Liere and Win de Leeuw. Graphsplatting: Visualizing graphs as continuous fields. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):206–212, 2003.

[16] Allen R. Martin and Matthew O. Ward. High dimensional brushing for interactive exploration of multivariate data. In *IEEE Conference on Visualization '95*, pages 271–278, 1995.

[17] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[18] B. Schneiderman and C. Alhberg. Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In *ACM CHI Conference on Human Factors in Computing Systems*, pages 313–317, 1994.

[19] David W. Scott. *Multivariate Density Estimation : Theory, Practice, and Visualization*. Wiley Series in Probability and Statistics. Wiley-Interscience, 1992.

[20] Matthew O. Ward, Elke A. Rundensteiner, Jing Yang, Punit R. Doshi, and Geraldine Rosario. Xmdvtool: Interactive visual data exploration system for high-dimensional data sets. In *IEEE Symposium on Information Visualization*, pages 52–53, 2002.

[21] Martin Wattenberg and Danyel Fisher. A model of multi-scale perceptual organization in information graphics. In Stephen C. North and Tamara Munzner, editors, *IEEE Symposium on Information Visualization*, Seattle, USA, 2003. IEEE Computer Press.

[22] D.J. Watts and S. H. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393:440–442, 1998.

[23] Duncan J. Watts. *Small Worlds*. Princeton University Press, 1999.

[24] C. Williamson and B. Schneiderman. The dynamic homefinder: Evaluating dynamic queries in a real-estate information exploration system. In *ACM SIGIR '92*, pages 339–346, 1992.