DOI: 10.1111/j.1467-8659.2012.03091.x
Eurographics Conference on Visualization (EuroVis) 2012
S. Bruckner, S. Miksch, and H. Pfister
(Guest Editors)

Volume 31 (2012), Number 3

Using Signposts for Navigation in Large Graphs

T. May[1], M. Steiger[1], J. Davey[1] and J. Kohlhammer[1]

[1]Fraunhofer IGD, Darmstadt, Germany

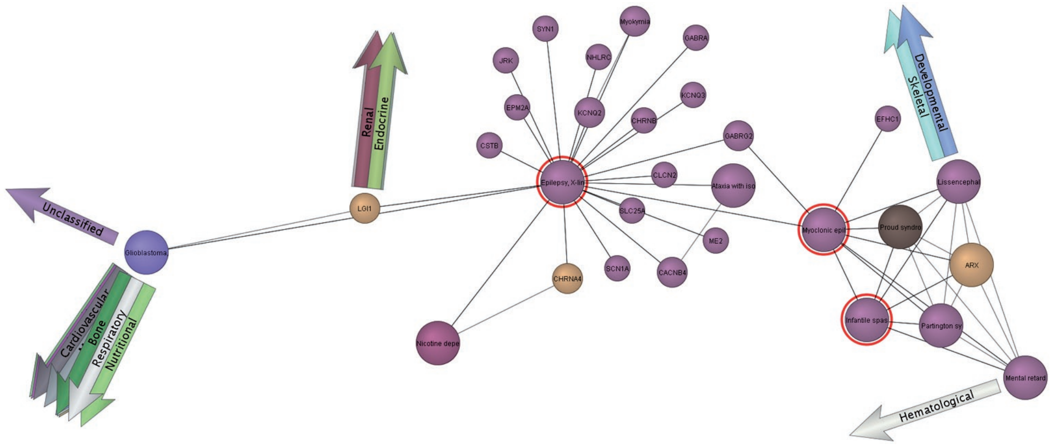

Figure 1: *Navigation in a large graph from the medical domain [GCV*07] using signposts to provide context for the nodes in focus (as used in our user study). The focus is derived from the focal nodes (encircled in red).*

Abstract

In this paper we present a new Focus & Context technique for the exploration of large, abstract graphs. Most Focus & Context techniques present context in a visual way. In contrast, our technique uses a symbolic representation: while the focus is a set of visible nodes, labelled signposts provide cues for the context — off-screen regions of the graph — and indicate the direction of the shortest path linking the visible nodes to these regions. We show how the regions are defined and how they are selected dynamically, depending on the visible nodes. To define the set of visible nodes we use an approach developed by van Ham and Perer that dynamically extracts a subgraph based on an initial focal node and a degree-of-interest function. This approach is extended to support multiple focal nodes. With the symbolic visualization, potentially interesting regions of a graph may be represented with a very small visual footprint. We conclude the paper with an initial user study to evaluate the effectiveness of the signposts for navigation tasks.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1. Introduction

Imagine yourself as a researcher in a young, interdisciplinary field such as visual analytics. Now imagine that you need to search for relevant publications related to the topic of your next paper. Because visual analytics is interdisciplinary, you will need to search for publications from different fields, like information visualization, machine-learning and perhaps an applied science in order to familiarize yourself with your theme. In particular, the connections between these fields are interesting to you, since as a visual analytics researcher, your

© 2012 The Author(s)
Computer Graphics Forum © 2012 The Eurographics Association and Blackwell Publishing Ltd. Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.

delivered by
EUROGRAPHICS DIGITAL LIBRARY
www.eg.org diglib.eg.org

goal is to synthesize results from many fields to find new solutions.

When browsing through the literature you hop from publication to publication via references and keywords. With most of the links you follow, you end up in the same scientific area that you started in. Publications at the boundaries between scientific fields are rare. To make things more difficult, these interdisciplinary publications do not necessarily carry the label *visual analytics*.

The task described here is an exploratory search in the potentially uncharted territory of a graph. The expected results of an exploratory search cannot be named in advance. In many cases, however, the graphs are not completely uncharted. In this paper we propose an approach to graph navigation, which makes use of existing labels (e.g. the names of scientific fields) as cues to assist user orientation in the abstract spaces of graphs.

A common challenge in the visualization of large graphs is to find the balance between the amount of data displayed and the preservation of the user's sense of context. In the past decades, a number of navigation concepts have been established to support the user in graph navigation tasks. In the broadest sense *Overview & Detail*, *Pan & Zoom* and *Focus & Context* techniques provide two or more connected levels of resolution for the display of data. The user may choose between these resolutions either for detailed inspection or for orientation. In most Focus & Context techniques, the context is established primarily by graphical means — for instance with the help of distortion [SB92] or a magic lens [GNBP11]. We use a combination of graphical and textual cues to refer to the context, which actually lies outside of the current field of view.

Our inspiration for the cues came from the real world. Before GPS navigation literally became ubiquitous, signposts helped travellers to find their way through unknown areas. While the visible landscape can be considered the *focus*, signposts provide the *context* with graphical cues, indicating direction, and textual cues, representing potential destinations beyond the focus. In our paper, we apply this concept to abstract graphs. The focus is a node-link diagram of a selected subgraph, while the context consists of named glyphs associated with specific visible nodes. They point at regions which can be reached via these nodes, but lie beyond the boundaries of the visible subgraph.

There are two advantages of the textual representation. Firstly, subgraphs of any size may be represented with a very small visual footprint. Secondly, they can be recalled faster and more accurately than graphical representations [Kos07]. There are also disadvantages to the use of text. Firstly, the labels might be unknown to the user and thus need to be learned. The second drawback is the potentially high initial cost for the creation of meaningful labels. We allow for an interactive definition of regions and labels, but for large graphs their initial creation is a prerequisite. Fortunately, many network data repositories include resources like semantic annotations which can be used for labelling; this especially applies to digital libraries.

Two publications especially influenced our approach. For the focus (see Section 3) we will build upon the ideas of van Ham and Perer [vHP09]. They transferred the concept of degree-of-interest functions to graphs to define the relevant set of nodes, given a focal node of interest. We generalize their approach to multiple focal nodes. We also make use of and extend their expand-on-demand concept for navigation in the graph.

The context (see Section 4) is inspired by Ghani et al. [GRE11] and similar approaches for off-screen visualization. Insets at the boundaries of the focus view refer to neighbouring nodes outside of the visible region. The insets allow panning without zooming, because potential (near) destinations are already present in the focus. While our glyphs are comparable to the insets, they point to *regions* instead of individual nodes. We will show how these regions are selected based on their size and distance to generate the contextual information.

In summary, the contributions of this paper are:

- The extension of van Ham & Perers degree-of-interest function for graphs from single to multiple focus points.
- An adaptation of the procedure to ensure that the subgraph in the focus is connected.
- Providing context with the help of glyphs and textual cues referring to off screen regions (as opposed to single nodes).
- The dynamic selection and prioritisation of the off-screen regions based on a second adapted degree-of-interest function.

The rest of this paper is organized as follows: In Section 2 we give an overview of existing work on different aspects of our approach. The subsequent two sections are the technical core of our paper. In Section 3 we will describe how the set of visible nodes is selected and visualized for the focus. In Section 4 we will describe how contextual information is determined, prioritised and visualized as a reference to off-screen regions of the graph. In section 5 we present an initial user study which assesses the effectiveness of the signposts in providing contextual information for navigation tasks. Finally in Section 6 and 7 we present a discussion of current and future work as well as a conclusion of our paper.

2. Related Work

Our approach is related to previous approaches from different areas including navigation concepts in visualizations, off-screen visualizations and multi-level approaches to graph drawing. We will go through each of the areas and present the similarities and differences between existing approaches and ours.

Navigation Concepts: A recent synopsis of navigation concepts has been given by Cockburn et al. [CKB07]. They categorize *Overview & Detail*, *Zooming & Panning*, *Focus & Context* and other techniques based on their interface mechanism, and give a survey on evaluations of these methods. While Overview & Detail is considered as a spatial separation into two or more distinct visualizations, Zooming & Panning is considered a temporal separation between emphasized and suppressed subsets of the data. Focus & Context techniques avoid spatial or temporal separations by creating a seamless visual embedding of detailed information (the *focus*) within its *context*. We consider our approach a Focus & Context technique, because this definition does not prescribe the level of abstraction used for the context visualization.

Degree-of-Interest-methods (DOI) are actually independent of the concepts mentioned above, but became popular within Focus & Context visualization techniques. Furnas [Fur86] introduced the concept of estimating the set of most interesting data items based on information derived from user interaction. In [Fur06] Furnas clarified that the original intention of the DOI concept was to determine *what* items were to be emphasized in the focus, whereas most techniques concentrated on *how* the visual emphasis should be performed.

An example for the approaches defining *what* is emphasized in the focus is van Ham and Perers approach [vHP09] which adapts Furnas's DOI function to graphs. The focus is defined by a single node, and the neighbourhood of this node provides the context, which can be extended interactively. In our paper, we propose a new distance-based DOI function, which enables the inclusion of multiple nodes in the focus. We adopt Huang's et al. [HEW98] and Crnovrsanin's et al. [CLWM11] approaches, which use the interaction history for the definition of focal nodes. Both approaches use the path of exploration of one user or even the paths recommended by multiple users. Instead, our approach allows the exploration of multiple paths at the same time, while avoiding the splitting of the visible subgraph into disconnected regions. Kreuseler and Schumann [KS99] show how Focus & Context navigation can be combined with the creation of a customized data representation. A hierarchical clustering is dynamically refined based on user interaction. Fading edges for the indication of connections to invisible regions have been used, for example, by Dwyer et al. [DMS*08] and van Ham and Perer [vHP09]. In a recent extension of this concept [vHP12] van Ham and Perer introduce *Graphcues*. Similar to our approach, Graphcues are extensions to visible nodes, representing out links to the interesting but currently invisible parts of the graph. Instead of using static landmarks, the user is guided by the result of interactive queries, defining the interest of single nodes. Our approach uses landmarks — representing regions of the graph, whose interestingness is adapted as the user navigates through the graph.

Other approaches define *how* the focus can be displayed. They can be divided into spatial and non-spatial methods. Radial layouts provide a natural spatial focus - the centre of the view - and have been explored by, for example, Jankun-Kelly and Ma [JKM03], Chou and Yang [CY11], as well as Brandes and Pich [BP11]. Distortion methods, like the graphical fish-eye views [SB92], are also spatial. Munzner [Mun98] presents an approach which uses a 3-dimensional, hyperbolic space to realize a form of distortion. The focus in these methods normally lies in the middle of the visualization, while the context is distorted, but visible around the edges.

Non-spatial methods include Semantic Depth of Field by Kosara et al. [KMH01]. Tominski et al. [TAvHS06] presented a lens-based technique to solve the cluttering problem in the visualization of large graphs. The focus acts as a local filter, which suppresses the display of edges unrelated to the visual area. A similar filter is used to bring the local neighbourhood into the focus. This approach has been extended by Moscovich et al. [MCH*09] with the *Bring & Go* navigation concept. Movement in the graph is supported by a guided panning along the connections to neighbouring nodes, which is also related to techniques for off-screen visualization.

Off-Screen Visualization: Off-screen visualization can be considered a variant of Focus & Context, but deserves further elaboration here. Baudisch and Rosenholtz [BR03] compare two visualizations of off-screen locations (isodistance rings and arrows). While the rings performed better on a geographic map, they have no simple equivalent in the complex topology of an arbitrary graph. In fact, off-screen visualization is most often used in the context of geographic maps like the insets techniques presented by Karnick et al. [KCJ*10] and Ghani et al. [GRE11]. A counterexample is presented by Frisch and Dachselt [FD10] for the navigation in UML diagrams. In abstract graphs the position of visual proxies at the screen boundary preserves the global layout. We adopt their method to aggregate off-screen nodes to clusters based on inheritance hierarchies and generalize it for arbitrary graphs. In all these approaches, the focus is defined by the geometry of the view port; this requires a global layout without cluttering of nodes or edges. Since this is almost impossible to achieve for general graphs, we select the focus based on the graph topology.

Multi-Level Approaches: Our approach basically uses a high-resolution representation for the focus and an abstract representation for the context. Multi-level representations have been used to improve different aspects of graph visualization in order to allow structural zooming within the graph. Balzer and Deussen present a method for the seamless zooming between different levels of abstraction [BD07]. Frishman and Tal [FT04] and Huang et al. [HEL05] present methods to improve the layout. Shen et al. [SMER06] and Di Giacomo

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.

et al. [DGDGL07] show how semantic connections can be used to improve the graph visualization.

In summary, we will show that in addition to the contribution of the general approach described in this paper, we make a contribution to each aspect that we've built on. Our concept for Focus & Context in large graphs generalizes the idea of *context* to invisible off-screen regions, which are referred to with the help of visible signposts. In turn, off-screen visualizations are generalized with regard to the definition of the *view port*, which is not defined by the canvas geometry, but the graph topology. Finally, to our knowledge, off-screen visualizations of abstract graphs have not yet been used in combination with multi-level representations.

3. Visible Nodes in Focus

For the detailed explanation of our approach we will refer to a graph $G(E,V)$, consisting of a set of edges E and a set of nodes V. The techniques presented in this section and the next can only be applied to connected components. Thus we will assume, for simplicity's sake, that the graph G is connected. The *focus* $F \subset G$ is the visible subgraph, which is defined based on recent user interaction. The core of the focus is a small set of *focal nodes* $Z \subset V$ which are considered most interesting for the user. Regions $R_i \subset G$ are arbitrary subgraphs of G. Because it differs from van Ham and Perer's definition, we want to clarify that when using the word *context* we refer to regions beyond the visible subgraph F, and not the immediate, visible neighbourhood of Z.

Following van Ham and Perer's approach [vHP09] we use a degree-of-interest (DOI) function applied to the nodes in the graph. In essence, this approach defines a neighbourhood around a given focal node. Instead of using a single focal node, we extend the DOI function to multiple focal nodes.

Multiple focal nodes can be used to trace the development or refinement of the user's interest. This is especially useful in open search or browsing scenarios where the result cannot be defined a-priori and where the path to the solution may be of interest as well. We will now provide a detailed description of our extension to van Ham and Perer's DOI function. This will be followed by details of the derivation of F from the set of focal nodes Z. The application of the DOI function alone could lead to cases in which F contains disconnected neighbourhoods. This issue will be covered in the last part of this section.

3.1. Extending the DOI Function to Multiple Focus Nodes

The DOI function by van Ham and Perer consists of three components: An *a-priori interest* (API) which never changes, a *user interest* (UI) which changes based on initial user queries and the *distance-based interest* (D) referring to a focal node $z \in V$ (see Equation 1). Our extension affects the last component of the DOI function and the way the subgraph F is derived from this function.

$$DOI(x) = \alpha API(x) + \beta UI(x,y) + \gamma D(x,z) \tag{1}$$

We will now define a DOI function supporting more than one focal node. Let $Z = \{z_1, z_2, \dots\} \subset V$ be the set of focal nodes. Let $d(x,y)$ be a distance function defined for all pairs of nodes in the graph. A node closer to one or more focal nodes should be considered more interesting than a node farther away from the focal nodes. For the distance component Furnas [Fur86] proposed Equation 2.

$$D(x) = -d(x,z) \tag{2}$$

$$D(x) = -d(x,z_1) - d(x,z_2) - d(x,z_3) \cdots \tag{3}$$

Heer and Card [HC04] extended this function to multiple focal nodes in a hierarchy (see Equation 3). However, this extension is not suitable for multiple focal nodes in a general graph, because it emphasizes a convex region defined by the majority of focal nodes. Nodes outside of this region — even other focal nodes — may be excluded from the view. Instead we define an inverse distance vector for every node in the graph (shown in Equation 4). We define our DOI distance component for multiple focal nodes in Equation 5. For $p \geq 1$ the DOI distance component is a norm (the Minkowski-norm of the distance vector). For $p < 1$ the DOI distance is not a norm, since it does not fulfil the triangle inequality.

$$d_Z(x) = \left(\frac{1}{d(x,z_1)+1}, \frac{1}{d(x,z_2)+1}, \dots \right) \tag{4}$$

$$D_Z(x) = \| d_Z(x) \|_p \tag{5}$$

The choice of p defines how the neighbourhoods of the focal nodes are combined: in general, for higher values of p the neighbourhoods are more independent. The maximum norm $\|\cdot\|_\infty$ represents a *nearest-neighbour* approach, where only the distance to the single nearest focal point is taken into account. The norm $\|\cdot\|_1$ averages the inverse distance to all focal points. For our approach we prefer DOI distance components with $0 < p < 1$. The resulting distance component emphasizes the area between two or more focal nodes (see Figure 2). While these DOI components are not norms, they can be useful in cases where the connection between the focal nodes is considered at least as important as the set of focal nodes itself. In addition, these measures are less likely to split the visible subgraph into disconnected parts. However, this cannot be avoided by the DOI function alone, see Subsection 3.3.

3.2. Defining the Set of Visible Nodes

In this subsection we will draw upon approaches from van Ham and Perer [vHP09], Heer and Card [HC04] and Huang

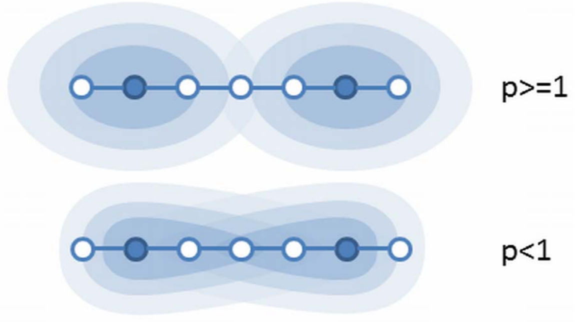

Figure 2: *This schematic view shows the combined isodistances with two focal nodes (blue) for different values of p. The value of p controls to what extent nodes between two or more focal nodes will be preferred over nodes near a single focal node. High values of p will induce independent neighbourhoods; lower values of p will be more likely to cause neighbourhoods to merge.*

et al. [HEW98]. The DOI function does not depend on the specific method chosen to define the set of focal nodes, Z. Existing approaches allow for an interactive definition of new focal nodes by clicking. This causes the visible subgraph F to be expanded to include the neighbourhood of the selected node. A common approach is to use a history queue for the set of focal nodes Z. Old focal nodes are eliminated, if the history queue reaches a predefined maximum size (see [HEW98]).

Upon adding a new focal node, we calculate the distance of this node to all other nodes with the chosen distance function $d(x,y)$. The DOI values are updated accordingly. The derived set of visible nodes van Ham and Perer suggest a method which is basically a modified *Dijkstra* algorithm. The set of visible nodes is grown around the focal node as a seed. Instead of preferring the nodes with the shortest distance, their algorithm prefers nodes with a higher DOI. The growth terminates when a predefined number of nodes has been selected. Still, there is no guarantee that the resulting visible graph is connected. In the following section we show how we deal with this problem.

3.3. Disconnected Neighbourhoods

After the growth process, the neighbourhood of a focal node is defined as the set of visible nodes which can be reached from this node. Multiple focal nodes may share the same neighbourhood. In our approach, we connect the neighbourhoods with bridges, i.e. chains of nodes along a reasonably short path between neighbourhoods. This blends well with the interactive expansion of the focus; every visible node serves as an anchor for detailed inspection. If a user wishes to learn more about the area between her current focal nodes, the bridges are the anchors for this exploration.

The first task is to determine if focal nodes share the same neighbourhood. Every focal node is marked with a unique identifier for its neighbourhood. Every node reachable from a focal node is marked with the same identifier. If two neighbourhoods grow together, their identifiers are unified. Consequently, a pair of neighbourhoods with different identifiers at the end of the process is not connected.

The second task is the definition of the bridges themselves, which is less obvious. We propose a pragmatic, iterative solution which is illustrated in Figure 3. In every iteration, the smallest neighbourhood N_0 is chosen. To calculate a bridge to another neighbourhood, we continue with the growth process described in the previous section as if there were no limit for the number of nodes. Only one neighbourhood N_0 is grown to speed up the process. If a different neighbourhood is reached the shortest bridge is traced back along the links of a spanning-tree. This chain of nodes is added to the set of visible nodes and the neighbourhoods are thus connected. The iteration stops when there is only one neighbourhood left. This single neighbourhood is the focus F, which is the visible subgraph in the visualization.

4. Signposts Referring to Context

In the previous section we described the creation of the *focus* in our Focus & Context technique. The focus $F \subset G$ allows for the detailed inspection of a subgraph and it provides a mechanism for the interactive exploration of adjacent regions. The *context* in our visualization consists of glyphs which resemble signposts (see Figure 4). They are arranged at the boundaries of the focus. Like signposts in the real world they point to specific regions beyond the visible area of the graph. The user can explore the neighbourhoods of nodes in the focus by selecting new focal nodes. The context provides information on the location of the visible subgraph with respect to off-screen regions. The selection, position and direction of the glyphs are updated whenever the user changes the focus.

In this section we will explain how the context is created. We will begin by defining the off-screen regions of a graph that can be referred to with the help of signposts. We will then show how the regions to be referred to are chosen, based on the current focus. Finally, we will describe the positioning and orientation of the signposts in the visualization.

4.1. Labelled Regions

In the real world, a signpost pointing to a place far away will almost never refer to an individual building or street. It will refer to a city, a region or even a country. Of course their labels implicitly represent all minor landmarks within their boundaries. In order to transfer this concept to graphs, we need to aggregate nodes to *labelled regions*. In general, a region is a connected subgraph $R \subset G$. Our method uses multiple regions, which may overlap partially or totally. Over-

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.

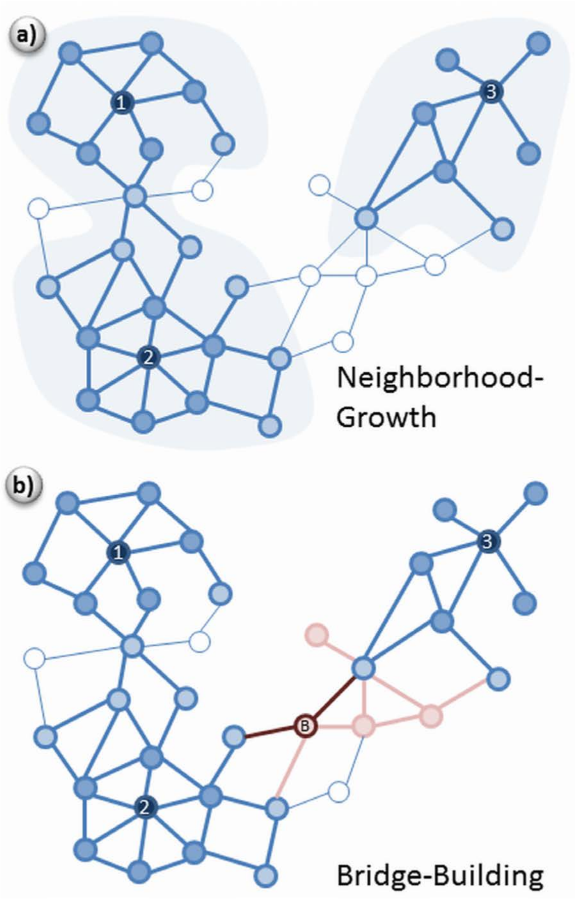

Figure 3: *The selection of nodes for the focus is executed in two steps. In the first step (a), the neighbourhoods around the focal nodes (1,2 and 3) are grown. Neighbourhoods may merge during the process (1 and 2, left). The growth always prefers nodes with a higher DOI. The process stops when a predefined number of nodes is reached. In the second step (b) the connectivity of the visible graph is ensured by creating bridges. The smallest disconnected neighbourhood (3, right) is chosen and grown, until a another neighbourhood is reached. The bridge (B) is created along the shortest path.*

lapping regions are useful to define multiple levels of detail (see Section 4.2). This means that there are no restrictions placed on the set of regions $\mathcal{R} = \{R_1, R_2, \dots\}$. In particular, it does not need to be a partition of the graph.

The labelling of regions is not a contribution of our work, but an important prerequisite. Our method can be used for navigation in graphs enriched with metadata. With the rise of the semantic web, a growing number of repositories provide the resources necessary to define meaningful regions. The *Resource Description Framework* (RDF) used in the *Linked Open-Data Community* [**?**] is an example of this data. *Authors*, *topics* or the *ICD-Codes* for diseases in the dataset we used for the user study (see Section 5 are potentially useful attributes for the definition of regions.

Each region is defined using a single data attribute. This allows us to use the attribute value as a label. All nominal attributes in the data may be used to define candidate regions. If a taxonomy exists for an attribute we can use the hierarchy to create regions of differing levels of detail. Any regions sharing a common super class could be combined and labelled with the name of this class. Even if there is no technical restriction on the definition of regions, not all regions are equally useful as a landmark in the graph. In general, there is no guarantee that the distribution of specific attribute values matches the topology of the graph. To be referred to with a signpost, we require a region to be sufficiently *compact*. A region does not need to be contiguous, but its average distances should be significantly smaller than the average distances of the complete graph. Only then we can expect the region to define a location in the graph topology.

We also considered the use of graph clustering algorithms to generate regions. These regions typically reflect the graph topology, but they do not come with a meaningful label. Applying classification methods to generated regions is a promising option for the identification of labels, but this is beyond the scope of this paper (see Section 7). As an alternative, we allow the user to extend the given set of regions. If the user explores a set of nodes relevant enough to come back to later, she may select them with a brush and add a label for the selected set of nodes (see Figure 4). The new region and its label are stored and the region is referred to with its own signpost if the user moves away.

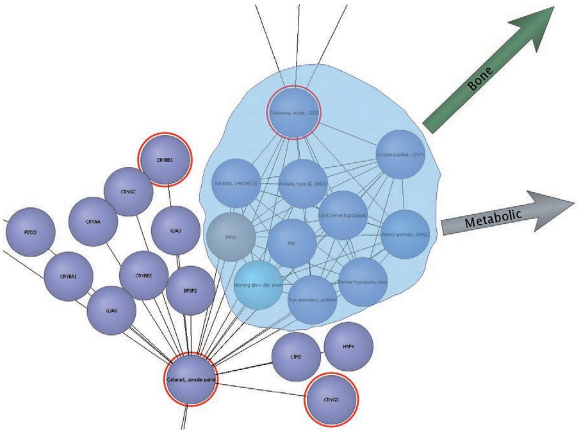

Figure 4: *The signpost directs the user to off-screen regions which can be reached via the adjacent node. Groups of nodes can be selected interactively in order to create user-defined regions.*

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.

4.2. Selecting Regions for the Context

Clutter should be avoided in the display of context. To address this issue, the number of signposts which are displayed at the same time needs to be restricted. Because there is no formal a-priori restriction to the definition of regions, the number of potentially relevant regions can be very high. As for the focus, we apply a degree-of-interest scheme as a filter for the context. Every region is given a DOI value. This value depends on specific properties of the region. In addition, it depends on the set of nodes $F \subset G$ in the focus. Whenever the focus changes, regions may become more or less relevant for reference in the context.

Specifically, the degree-of-interest function depends on the following properties:

- *Intersection with the focus*: if a region R intersects with the focus F then the region is automatically excluded from the selection.
- *Region size* $|R|$: without further information on the user's specific interests, bigger regions are considered more important than smaller regions. There is also a practical reason for this scheme; the whole graph can be covered more easily by referring to a few large regions.
- *Region distance from the focus* $d(R,F)$: regions near the focus are considered more important. It is likely that they are more closely related to the nodes in the focus, i.e. the nodes that the user is exploring actively. Note that this depends on how distance is calculated (see Section 4.3).
- *A-priori interest* $API(R)$ *and user interest* $UI(R)$: these are essentially the same properties as those which were defined for individual nodes (see Section 3). For example, it makes sense to increase the a-priori interest for user-defined regions, because the user explicitly expressed her interest in these regions.
- *Exclusion*: sometimes regions may be similar even when defined with different methods. While this is interesting from an analytical point of view, having multiple signposts referring to the multiple regions which differ only slightly would waste screen space and might confuse the user. Exclusion is applied to any set of similar regions referred to from the same visible node.

Based on the above requirements, the DOI function for the regions is defined in Equation 6. To fulfil the first requirement, regions which intersect the focus are given a DOI value of 0.

$$\widehat{DOI}(R) = \alpha \cdot API(R) + \beta \cdot UI(R) + \gamma \cdot D(d(R,F),|R|) \quad (6)$$

The function $D(.)$ reflects the fact that region size and distance from F are interdependent. Consider a large region in the graph which overlaps a set of smaller regions. From a distance, the large region is generally more interesting than the smaller ones. If, however, the user moves the focus closer to these regions, we assume that she will be interested in more detailed information. To support this level-of-detail scheme, smaller regions should be preferred at close range. We suggest the function class in Equation 7 to satisfy this property. The constant R_{max} is the largest region.

$$\begin{aligned} D((d(R,F),|R|) &= \frac{\omega}{d(R,F)^{\omega}} \\ \omega &= 2 - \frac{log(|R|)}{log(|R_{max}|)} \end{aligned} \quad (7)$$

The only remaining requirement is the exclusion of similar regions. For every region $R \in \mathcal{R}$, a list of excluding regions $X_1, X_2, \cdots$ is defined in advance. With the help of these lists we modify the DOI to obtain the final Equation 8.

$$DOI(R) = \begin{cases} 0 & \text{if } R \cap F \text{ is not empty,} \\ 0 & \text{if } max_i(\widehat{DOI}(X_i)) > \widehat{DOI}(R), \\ \widehat{DOI}(R) & \text{otherwise.} \end{cases} \quad (8)$$

4.3. Definition of Region Distance $d(R,F)$

We define the graph of regions $\mathcal{H}$ as a weighted graph, where nodes are the regions $\mathcal{R}$ of G. The edges of $\mathcal{H}$ represent the adjacency, intersection or nesting of regions. The weight of the edges is given by the pairwise distance between regions. Because the regions are sets, no natural distance measure exists. This problem is primarily dealt with using agglomerative clustering (see, for example, Berkhin [Ber02] for a survey). A number of different measures may be used. In order to support graph navigation it is not necessary to find an optimal solution. An approximate solution which enables a reference to the right direction is sufficient.

Single linkage represents the minimum distance between a pair of nodes in different regions. It is the easiest distance to calculate. Nevertheless, we chose the *average-linkage* distance for the definition of $\mathcal{H}$ for two reasons. Firstly, the minimum distance between two regions underestimates distances on G because it ignores distances *within* regions. Secondly, it can naturally be applied to intersecting and nested regions, while the single-linkage distance would be zero in both cases. The graph $\mathcal{H}$ is a coarse representation of G, which is used to speed up the determination of the distance of far flung regions.

Whenever the focus changes, the relevant regions and their distances need to be recalculated on the fly. Depending on the size of the graph, this recalculation can be costly. Fortunately most of the regions are given in advance and their pairwise distances can be precalculated. When updating the focus we only recalculate the distance to the nearest regions (see next section). The distance to regions which are further away is added by finding the shortest path in $\mathcal{H}$ between a closer and a farther region using the precalculated distances. This means the calculation operates on the graph of nodes G and on the graph of regions $\mathcal{H}$ (see Figure 5).

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.

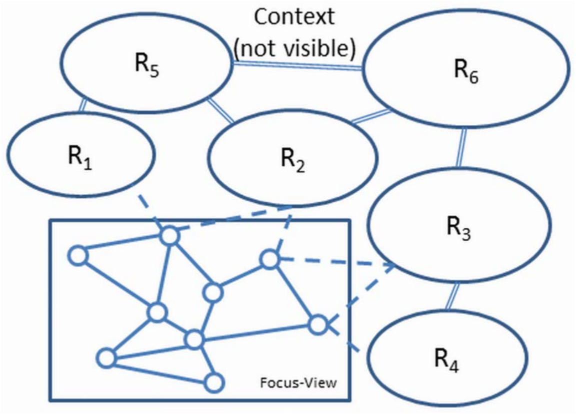

Figure 5: *This figure shows the focus surrounded by a number of invisible regions. The distance to all regions needs to be calculated whenever the focus changes. To speed up the process, we only recalculate the distance to near regions* (R_1,R_2,R_3,R_4)*. To calculate the distance to far regions* (R_5,R_6) *we add the precalculated distance between regions.*

4.4. Placement of Glyphs

The glyphs must convey at least two pieces of information. The first piece is the label of the region referred to. The second piece is the visible node, which can be expanded in order to move the focus closer to this region. Determining the distance between regions and determining the closest visible node are similar problems. They can thus be solved using the same process, which consists of two steps. In the first step, the distance to closer regions is calculated on G. Again we use the modified *Dijkstra spanning-tree* algorithm in a similar way as we did for growing the neighbourhoods in the focus. *All* visible nodes (i.e. all nodes in F) are the seed nodes of the algorithm. Whenever a node on a shorter path is visited we update the visible node it has been reached from and the distance for the regions of this node. This process stops after a fixed number of nodes have been visited. All regions found in the first step are, by definition, *near* regions.

In the second step the distances to far regions is calculated on $\mathcal{H}$. Again, we use the same algorithm. All near regions found in the first step are now the seed nodes for the spanning-tree algorithm. The final distance of a far region is the sum of the distances from the first and the second steps.

For every region, we know its shortest distance to F and the visible nodes it can be reached from on the shortest path. Generally the glyphs are set in order of decreasing interest of their associated regions. If more than one visible node is a candidate for a signpost we select the signpost node with the following strategy. If the glyph was attached to a node the previous interaction cycle, it stays there to minimize flipping. If the signpost was not visible and one candidate node has fewer attached glyphs then this node is chosen, otherwise a random candidate is chosen.

The direction of every glyph is defined by the first edge of the shortest path. This edge connects a visible start node in the focus region with an invisible end node in a near region. In a static layout, this edge direction is defined in advance. In a dynamic layout invisible nodes and edges are usually not considered. To calculate the direction we add all end nodes and all their connections to F to the dynamic layout calculation. This guarantees that our approach works seamlessly with different dynamic layout methods. Note that for dynamic layouts supporting animated transitions, the glyph will follow the transition of its (virtual) edge.

5. User Study

We tested our approach in an initial experiment that compared multi-focal graph exploration with and without signposts. We wanted to test whether the use of signposts supports the user in the navigation and orientation in large graphs.

We recruited 13 participants aged between 21 and 30 for the experiment. All of the participants were graduate or undergraduate students in computer science or computer-science-related courses. Our experiment was a *within-group* test. The experiment consisted of two almost identical tasks, each performed under different conditions. To filter out undesired learning effects we alternated the sequence of the tasks performed by each participant.

In each task, participants were asked to find a pair of nodes, one each from two predefined categories, and a path connecting the two nodes. We selected the categories such that the tasks were easily solvable with the available interface. For the test we used a precalculated Fruchterman-Reingold layout. In both tasks five focal node changes were necessary to reach the target. The number of focal nodes to define the focus view was limited to three. The start conditions for each task were identical. The only difference between the tasks was the availability of signposts, providing additional contextual information to the participants.

Possible user interactions included panning and zooming, browsing the graph by adding a node to the focal node history and the selection of nodes for detailed information. Focal nodes were highlighted with red boundaries. The graph that we used for the experiment was based on data from *The Human Disease Network* by Goh et al [GCV*07]. We extracted the largest connected subgraph, because in its present form, our technique cannot be applied to disconnected graphs. We decided to use this particular data set for two reasons. Firstly, each node was already assigned to one of 23 different categories. Thus we had a predefined set of regions to use for the definition of signposts. Secondly, the meaning of the nodes (either genes or diseases), the links between them and the categories that grouped them was clear

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.

to non-experts without the need for detailed explanation. To ensure an equal level of expertise among participants we made sure that all participants had little or no knowledge of this particular data set or medicine in general.

The nodes in the original graph were either diseases, or genes linked to those diseases. We did not distinguish between diseases and genes in our visualization, since the same categorization was applied to both types. We converted the originally directed graph into an undirected graph and filtered out gene-gene relations to reduce its density. The final graph contained roughly 1500 nodes and 5500 edges.

To test our method, we measured the task completion time, as well as two click-based statistics. The time was measured between the moment the task was given to the participant and the moment the participant he or she had found a path and reported that to us. The click-based measurements were the number of focal node changes and the number of selection clicks (to view detailed information about the nodes). Figure 6 shows the average completion times for both tasks and both conditions. Direct comparison reveals that the testers were faster with the visualization that contained landmarks.

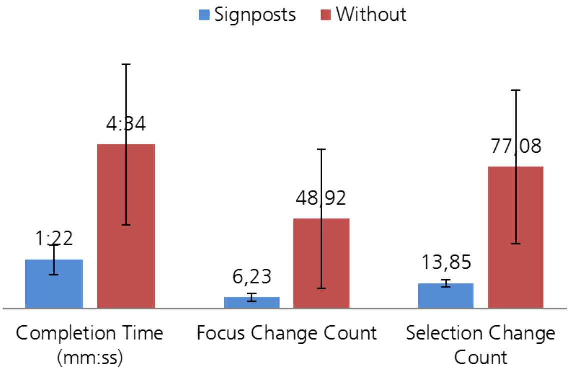

Figure 6: *The results of the experiment: lower mean and lower standard deviation in completion time, focus change count and selection count for the visualization with signposts.*

The difference in the average task completion time, the number of focus changes and selection clicks points to an improvement in graph navigation through the use of signposts.

6. Discussion

We observed the participants in our user study in order to develop an idea of the problems they encountered. Above all, people forgot where they had started and which part of the graph they had already explored. We used a queue with 3 entries for focal nodes. Some participants suggested having a larger number of focal nodes. However, a larger number of focal nodes caused the edges to be cluttered in the static layout. Thus, we concluded that multiple focus points are likely to work better in conjunction with dynamic layout techniques. Other suggestions included a back-tracking hint to the start node.

The idea of having signposts was very well appreciated by the participants. We presented a maximum of three most interesting regions per node to avoid visual cluttering. This sometimes lead to the relevant region for the task not being visible. The ability to interactively browse through all signposts on demand, e.g. as a mouse over, was proposed as a way to combat this problem.

An important limitation of the signpost idea is its dependency on meaningful labels. For large graphs, it is virtually impossible to create them manually. If node attributes do not match the graph topology, purely automated methods might fail to generate effective results. However, the possibility for region definition by the user points to two alternatives for future work. The first one is the interactive refinement of automatically generated regions, the second one is a crowd-sourcing approach to create a shared repository of user-defined regions for large graphs.

7. Conclusion & Outlook

In this paper we presented a new approach to navigation in large graphs. It is an extension of van Ham and Perer's idea of presenting a meaningful context around a single focal node. In our approach we consider a set of focal nodes and deal with the problems that come with this. Additionally, we provide visual cues to other regions of the graph that could be worth exploring. The cues also indicate the direction of the regions in order to ease navigation towards them.

We intend to follow ideas gained in our initial user study to further improve our visualization. Future work will include better support for the history of the exploration and the use of a dynamic layout for the graph. The current implementation uses a single, static layout to ensure that the position of nodes in space does not change over time and thus confuse the user. A dynamic layout applied only to the visible nodes would be significantly faster and need to deal with fewer constraints as the number of visible nodes is a lot smaller. Nodes that become visible through a change of focus would be added to the layout on the fly. However, the initial placement of new nodes and edges in an existing layout is not trivial as it should not impair the existing layout.

The signposts offer the potential to show more information than the region name. Size, length, colour or transparency might be used to represent more information about the region or the path leading to it. Apart from showing information, using these signposts for interaction is another topic for further research.

References

[BD07] Balzer M., Deussen O.: Level-of-detail visualization of clustered graph layouts. In *Proc. of the 6th International Asia-Pacific Symposium on Visualization 2007 (APVIS)* (2007), Hong S.-H., Ma K.-L., (Eds.), IEEE, pp. 133–140. 3

[Ber02] Berkhin P.: Survey of clustering data mining techniques. *Techniques 10*, c (2002), 1–56. 7

[BP11] Brandes U., Pich C.: More flexible radial layout. *Journal of Graph Algorithms and Applications 15*, 1 (2011), 157–173. 3

[BR03] Baudisch P., Rosenholtz R.: Halo: a technique for visualizing off-screen objects. In *Proc. of the 21th International Conference on Human Factors in Computing Systems (CHI)* (New York, NY, USA, 2003), CHI '03, ACM, pp. 481–488. 3

[CKB07] Cockburn A., Karlson A., Bederson B. B.: A review of overview+detail, zooming, and focus+context interfaces. *ACM Comput. Surv. 41* (January 2007), 2:1–2:31. 3

[CLWM11] Crnovrsanin T., Liao I., Wu Y., Ma K.-L.: Visual recommendations for network navigation. In *IEEE Symposium on Visualization (EuroVis 2011)* (2011), Hauser H., Pfister H., van Wijk J. J., (Eds.), vol. 30, Blackwell Publishing Ltd., pp. 1081–1090. 3

[CY11] Chou J.-K., Yang C.-K.: PaperVis: Literature review made easy. *Computer Graphics Forum 30*, 3 (2011), 721–730. 3

[DGDGL07] Di Giacomo E., Didimo W., Grilli L., Liotta G.: Graph visualization techniques for web clustering engines. *IEEE Transactions on Visualization and Computer Graphics 13* (March 2007), 294–304. 4

[DMS*08] Dwyer T., Marriott K., Schreiber F., Stuckey P., Woodward M., Wybrow M.: Exploration of networks using overview+detail with constraint-based cooperative layout. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (Nov. 2008), 1293–1300. 3

[FD10] Frisch M., Dachselt R.: Off-screen visualization techniques for class diagrams. In *Proc. of the 5th International Symposium on Software visualization* (New York, NY, USA, 2010), SOFTVIS '10, ACM, pp. 163–172. 3

[FT04] Frishman Y., Tal A.: Dynamic drawing of clustered graphs. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on* (0-0 2004), pp. 191–198. 3

[Fur86] Furnas G. W.: Generalized fisheye views. *SIGCHI Bull. 17* (April 1986), 16–23. 3, 4

[Fur06] Furnas G. W.: A fisheye follow-up: further reflections on focus + context. In *Proc. of the 24th International Conference on Human Factors in Computing Systems (CHI)* (New York, NY, USA, 2006), CHI '06, ACM, pp. 999–1008. 3

[GCV*07] Goh K.-I., Cusick M. E., Valle D., Childs B., Vidal M., Barabási A.-L.: The human disease network. In *Proc. of the National Academy of Sciences USA* (2007), vol. 104, pp. 8685–8690. 1, 8

[GNBP11] Gasteiger R., Neugebauer M., Beuing O., Preim B.: The FLOWLENS: A focus-and-context visualization approach for exploration of blood flow in cerebral aneurysms. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (dec. 2011), 2183 –2192. 2

[GRE11] Ghani S., Riche N. H., Elmqvist N.: Dynamic insets for context-aware graph navigation. *Computer Graphics Forum 30*, 3 (2011), 861–870. 2, 3

[HC04] Heer J., Card S. K.: DOITrees revisited: scalable, space-constrained visualization of hierarchical data. In *Proceedings of the working Conference on Advanced visual interfaces* (New York, NY, USA, 2004), AVI '04, ACM, pp. 421–424. 4

[HEL05] Huang X., Eades P., Lai W.: A framework of filtering, clustering and dynamic layout graphs for visualization. In *Proceedings of the Twenty-eighth Australasian Conference on Computer Science (ACSC)* (Darlinghurst, Australia, Australia, 2005), ACSC '05, Australian Computer Society, Inc., pp. 87–96. 3

[HEW98] Huang M. L., Eades P., Wang J.: Online animated visualization of huge graphs using a modified spring algorithm. *Journal of Visual Languages & Computing 9*, 6 (1998), 623–645. 3, 5

[JKM03] Jankun-Kelly T. J., Ma K.-L.: MoireGraphs: radial focus+context visualization and interaction for graphs with visual nodes. In *IEEE Symposium on Information Visualization (InfoVis)* (oct. 2003), pp. 59 –66. 3

[KCJ*10] Karnick P., Cline D., Jeschke S., Razdan A., Wonka P.: Route visualization using detail lenses. *IEEE Transactions on Visualization and Computer Graphics 16* (March 2010), 235–247. 3

[KMH01] Kosara R., Miksch S., Hauser H.: Semantic depth of field. In *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)* (Washington, DC, USA, 2001), IEEE Computer Society, pp. 97–. 3

[Kos07] Kosslyn S.: Remembering images. *Memory and Mind: A Festschrift for Gordon H. Bower* (2007), 93–110. 2

[KS99] Kreuseler M., Schumann H.: Information visualization using a new focus+context technique in combination with dynamic clustering of information space. In *Proc. of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation Conference on Information and knowledge management* (New York, NY, USA, 1999), NPIVM '99, ACM, pp. 1–5. 3

[MCH*09] Moscovich T., Chevalier F., Henry N., Pietriga E., Fekete J.-D.: Topology-aware navigation in large networks. In *Proc. of the 27th International Conference on Human Factors in Computing Systems (CHI)* (New York, NY, USA, 2009), CHI '09, ACM, pp. 2319–2328. 3

[Mun98] Munzner T.: Drawing large graphs with H3Viewer and Site Manager. In *Graph Drawing* (1998), pp. 384–393. 3

[SB92] Sarkar M., Brown M. H.: Graphical fisheye views of graphs. In *Proc. of the 27th International Conference on Human Factors in Computing Systems (CHI)* (New York, NY, USA, 1992), CHI '92, ACM, pp. 83–91. 2, 3

[SMER06] Shen Z., Ma K.-L., Eliassi-Rad T.: Visual analysis of large heterogeneous social networks by semantic and structural abstraction. *IEEE Transactions on Visualization and Computer Graphics 12* (November 2006), 1427–1439. 3

[TAvHS06] Tominski C., Abello J., van Ham F., Schumann H.: Fisheye tree views and lenses for graph visualization. In *Proc. of the Conference on Information Visualization (IV)* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 17–24. 3

[vHP09] van Ham F., Perer A.: Search, show context, expand on demand: Supporting large graph exploration with degree-of-interest. *IEEE Transactions on Visualization and Computer Graphics 15* (2009), 953–960. 2, 3, 4

[vHP12] van Ham F., Perer A.: Integrating querying and browsing in partial graph visualizations. *IBM Technical Report 12-01* (2012). 3

© 2012 The Author(s)
© 2012 The Eurographics Association and Blackwell Publishing Ltd.