Visualizing Fuzzy Overlapping Communities in Networks

Corinna Vehlow, *Student Member, IEEE*, Thomas Reinhardt,
and Daniel Weiskopf, *Member, IEEE Computer Society*

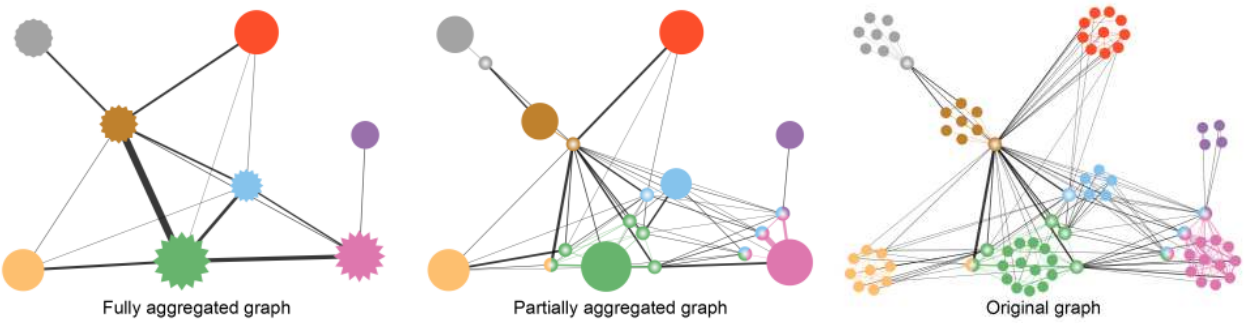


Fig. 1. Fuzzy overlapping communities in a weighted undirected graph of 254 co-appearances of 77 characters in Victor Hugo's novel "Les Misérables" [32] shown at different levels of detail. In the fully aggregated graph, the uncertainty of a community is mapped to the stars' depth of jags. In the partially aggregated graph, all aggregated vertex subsets are completely certain as they are represented by circles instead of stars. In the partially aggregated and the original graphs, the fuzzy community membership for single vertices is mapped to the lightness and saturation of nodes using a radial color gradient.

Abstract—An important feature of networks for many application domains is their community structure. This is because objects within the same community usually have at least one property in common. The investigation of community structure can therefore support the understanding of object attributes from the network topology alone. In real-world systems, objects may belong to several communities at the same time, i.e., communities can overlap. Analyzing fuzzy community memberships is essential to understand to what extent objects contribute to different communities and whether some communities are highly interconnected. We developed a visualization approach that is based on node-link diagrams and supports the investigation of fuzzy communities in weighted undirected graphs at different levels of detail. Starting with the network of communities, the user can continuously drill down to the network of individual nodes and finally analyze the membership distribution of nodes of interest. Our approach uses layout strategies and further visual mappings to graphically encode the fuzzy community memberships. The usefulness of our approach is illustrated by two case studies analyzing networks of different domains: social networking and biological interactions. The case studies showed that our layout and visualization approach helps investigate fuzzy overlapping communities. Fuzzy vertices as well as the different communities to which they belong can be easily identified based on node color and position.

Index Terms—Overlapping community visualization, fuzzy clustering, graph visualization, uncertainty visualization

1 Introduction

Relations among objects are usually modeled as a graph consisting of a set of vertices connected by a set of edges. Graph visualization is a useful tool to understand the network properties. Graphs that represent real systems, e.g., biological or social networks, are not regular, i.e., the distribution of edges is globally and also locally inhomogeneous. Oftentimes, they consist of structural subunits, i.e., highly interconnected sets of vertices, where the density of edges between these groups is low. Such clusters, in the graph-theoretical sense called communities, can be detected by community detection methods.

The analysis of such community structures is of high importance to understand the structural and functional properties. Based on the community structure, we can draw conclusions about the attributes of the network members because objects within the same community usually have some properties in common. Unfortunately, vertices may belong to several communities at the same time. The overlap in networks is obvious in our everyday life, e.g., nodes in social networks participate in a multitude of diverse, overlapping contexts, all encoded in a single network structure [39]. Vertices assigned to more than one community are usually located at the boundary between clusters, thereby representing mediates or bridges between these communities. The identification of such bridges is an important topology-based task for analyzing the connectivity in graphs [35]. In contrast, vertices that have a high edge degree within the community take up a central position in the community and have an important function concerning the stability of the group. To analyze communities in realistic systems, the issue of detecting overlapping communities has become of high interest and various algorithms have been developed [16].

In general, we can differentiate two types of overlapping communities: crisp and fuzzy overlapping communities [20]. With crisp overlapping community detection methods [19, 40, 44], each object fully belongs to one or more communities; with fuzzy overlapping community detection methods [4, 34, 53], vertices may belong to different communities to different extent. Which type of overlapping communities is more suitable, depends on the data.

In this paper, we consider two typical application domains: biological and social networks, in which crisp as well as fuzzy overlapping communities occur. Hence, there is a need for visualization approaches for fuzzy overlapping communities facilitating the analysis of those. Previously available visualization techniques mainly allow us to visualize crisp overlapping or disjoint communities but do

- *Corinna Vehlow is with VISUS, University of Stuttgart, Germany. E-mail: corinna.vehlow@visus.uni-stuttgart.de.*
- *Daniel Weiskopf is with VISUS, University of Stuttgart, Germany. E-mail: daniel.weiskopf@visus.uni-stuttgart.de.*
- *Thomas Reinhardt is with University of Stuttgart, Germany. E-mail: reinhats@studi.informatik.uni-stuttgart.de.*

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

not support analyzing uncertainty of communities described by fuzzy community memberships. We present a visualization approach that enables users to analyze fuzzy overlapping communities in weighted undirected graphs at different levels of detail (see Figure 1). Compared to existing visualization approaches for crisp overlaps, our approach benefits from its advanced visual mappings and hierarchical structure. Our approach is based on node-link diagrams, where the fuzziness of the community memberships is encoded in the nodes' positions as well as their color gradients or shapes. The former is achieved by a layout approach that incorporates the fuzzy memberships. Due to the hierarchical structure of our approach, users can use it to investigate the network at the community level, the vertex level, as well as all intermediate states in between by partially aggregating subsets of vertices with a maximum fuzziness. Therefore, our approach allows users to concentrate on shared nodes of minimal fuzziness only. Finally, the details-on-demand option facilitates the investigation of the strength with that each vertex contributes to each of the communities, which is defined by the belonging coefficients of the respective vertex.

2 Related Work

The most prominent visual representations of relational data are node-link diagrams and adjacency matrices. Node-link diagrams are intuitive and effective for perceiving relations between objects because they exploit Gestalt principles [33] of closure and good continuation. When using node-link diagrams, vertices are mapped to geometric forms such as circles or squares and relations among them are expressed by straight or curved links. A challenge is the computation of an aesthetic layout. Common graph layout algorithms, such as force-directed, orthogonal, or hierarchical layout algorithms, aim at optimizing a set of aesthetic criteria for graph drawings [13]. Force-directed layout algorithms, such as the Fruchterman-Reingold model [18] or the Kamada-Kawai model [31], can reveal clusters. Due to the combination of repulsive spring forces between all nodes and attractive forces between adjacent nodes, vertices that are highly connected are positioned close to each other. Thereby, clusters emerge visually but roughly, although they have not been extracted explicitly. However, this "visual clustering approach" does not extract fuzzy memberships as it does not become apparent to which communities a vertex belongs with what extent. We use a force-directed approach to produce an aesthetic global layout for aggregated graphs representing relations between communities.

Clusters or communities can also be derived algorithmically based on community detection algorithms. Again, layout algorithms can help reveal cluster structures in the graph, e.g., using a divide-and-conquer approach in which each cluster is laid out separately before the clusters are composed to form the graph [8, 52]. Besides spatial proximity, color can be used to convey to which community a vertex belongs [3]. In contrast, hierarchically clustered graphs are usually visualized as recursively nested regions in the plane [14]. Groups of vertices are thereby surrounded by 2D [3] or 3D (semi-transparent) [7] convex hulls. Alternatively, hierarchically clustered graphs can be visualized using multi-level representations that visualize graphs at different abstraction levels by aggregating subgroups of vertices (clusters) and edges [14]. They are usually realized by drawing each level on a plane at a different z-coordinate and with the clustering structure drawn as a tree in the third dimension. In addition, there is much previous work on using hierarchical clustering to control the extent of represented information and to navigate through the graph using graph aggregation methods [2, 15, 49, 51]. The system developed by Archambault et al. [6] even supports the investigation of several hierarchies of the same data based on different clustering criteria. In comparison to the approaches mentioned before, Ham and Van Wijk [49] combine their degree of abstraction function with the fisheye approach to show detailed information for a specific section and fewer details for its surrounding. This degree of abstraction function is based on the topology-based hierarchical clustering they extract in advance based on a force-directed layout using a distance measure. Of course, also non-hierarchically clustered graphs can be visualized on an abstract level, e.g., CFinder [3], which provides users with a graph view of the communities. However, all these methods were developed for non-overlapping communities.

Existing visualization approaches mainly address the visualization of flat disjoint communities and crisp overlapping communities. The results of crisp overlapping community detection algorithms are so far visualized by highlighting shared nodes in a different color [3, 4]. Besides, overlapping sets of vertices or elements in general are commonly represented using Euler-like diagrams [37, 43, 46], overlapping convex hulls [22, 34, 40], or bounding isocontours [12]. While some approaches use only color and transparency of the contours or shapes to convey set membership [12, 34, 40, 43], Simetto et al. [46] additionally make use of texture. Alper et al. [5] use curves of different color connecting all elements of a set instead of surrounding them to represent set memberships. Only few papers included visualizations of fuzzy overlapping communities [4, 50], where the membership distribution, i.e., the belonging coefficients, of a vertex are represented by a pie chart. In the approach by Itoh et al. [24], pie charts are used to visualize the multiple categories to which a node belongs and which are divided into equally sized segments. As pie charts should have a minimum size to allow for the differentiation of membership degrees, the display becomes cluttered easily with increasing size of nodes. However, most of the visualization approaches are only for crisp and not for fuzzy overlapping communities.

Whereas the use of convex hulls does not support the investigation of fuzzy overlapping communities, some of the other approaches can be adapted to that effect. Similar to existing approaches, we use spatial proximity [52] and color [3] to visualize the memberships but indeed, with respect to their fuzziness. In particular, the node-link diagram at the vertex level is laid out based on a divide-and-conquer approach that includes the fuzzy membership into the calculations of the vertex positions. Moreover, our approach supports the user with a graph view of the communities, similar to the work of Adamcsek et al. [3], and is furthermore related to multi-level graph drawings [14], where only the graph view of one particular level is displayed at a time instead of showing a three-dimensional stack of drawings. The multiscale hierarchy does in this case not result from a hierarchical clustering approach but is generated based on the fuzzy memberships.

To analyze fuzzy overlapping communities, the fuzziness of the vertices memberships needs to be visualized. The quantification and visualization of uncertainties within data has been recognized as one of the most important issues in scientific visualization [25]. Here, uncertainties originate during the clustering process because vertices cannot be allocated to one single community due to their topology. This uncertainty, i.e., the fuzziness of the vertices memberships, is described by scalar values, which could be directly visualized using one of the possible approaches: adding glyphs or geometry to the rendered scene, modifying geometry, modifying attributes of the geometry, animation, or by addressing other human senses [41]. A common approach is the modification of geometry using visual attributes, like color, size, position, shape, transparency, and so on, or the plotting of discrete data points as glyphs (e.g., box plots or quartile plots) with specific visual attributes. Concerning color, in particular, the lightness or transparency is commonly used to visualize uncertainty. When using the deformation of geometry, usually the degree of bumpiness is used to represent uncertainty, where smooth shapes imply certainty. We decided to use two different mappings: the lightness of a node to represent the fuzziness of individual vertices and the shape of the node to represent the overall fuzziness of aggregated vertex sets.

While there is extensive research on visualizing uncertainties of flow fields and surface representations [21, 26], only little work has been done on visualizing uncertainties within graphs, i.e., the uncertainties of attributes of the graph. Collins et al. [11] visualized uncertainties of translations using lattice graphs that show multiple linear paths for a translation. They mapped the likelihood of a translation to the fuzziness and hue as well as on the vertical positioning of nodes. Cesario et al. [9] visualized uncertainties of multiple static node attributes using a spatial layout and multiple linked views, e.g., the bullseye. Within the bullseye, nodes are plotted at a specific angle and radius within a circle subsector to visualize the attribute value and

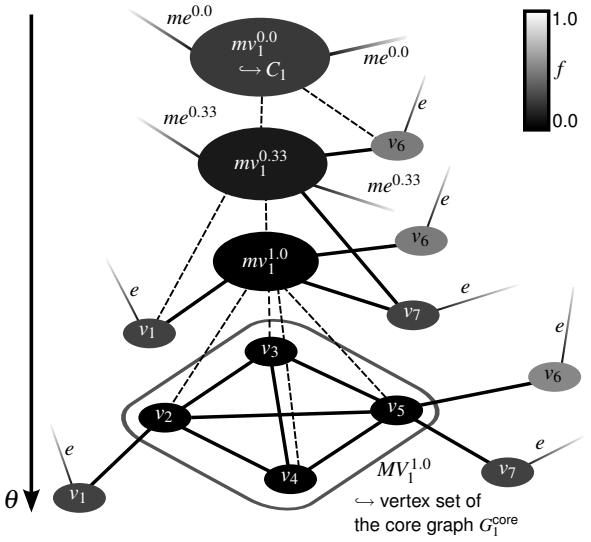


Fig. 2. Meta-graphs G^θ describing different aggregation levels of a graph, where only the section of community C_1 with $|C_1| = 7$ is represented. Vertices v_i and meta-vertices mv_1^θ are saturated with respect to their highest belonging coefficient ${f_i}^{\max}$ and average belonging ${f_1}^{\text{avg}}$.

confidence respectively. Their approach is designed to compare two static graphs and their attributes. Both approaches make use of the position to encode uncertainty but could not be extended to visualize fuzzy (and hence uncertain) overlapping communities.

3 Visualization Technique

Our visualization approach for fuzzy overlapping communities in weighted undirected graphs is based on node-link diagrams. The fuzziness of community memberships is visually encoded by the node positions as well as the attributes shape and color of a node. We decided to combine different mappings because using either the position or color of nodes leaves ambiguities concerning the communities to which a vertex belongs. After introducing our data model, we will present our hierarchical layout approach that considers the fuzzy membership distributions. In the following subsections, we will introduce our layered visualization model and show how the fuzziness of vertices or aggregated sets of vertices as well as the edge weights can be mapped to visual attributes of nodes and links, respectively. Finally, we will explain how users can interact with our visualization approach.

3.1 Data Model

We model an undirected weighted graph $G = (V, E)$ as a set of vertices V and a set of edges $E \subseteq V \times V$, where each edge $e_j \in E$, $1 \leq j \leq m$, $m = |E|$ is assigned a weight $w_{e_j} \in \mathbb{R}$. We define the overlapping community structure of the graph as set of communities $\{C_1, \ldots, C_K\}$, where each $C_k \subset V$ and K denotes the number of clusters. The communities are not necessarily disjoint, i.e., there are at least two communities C_{k_1} and C_{k_2} with $C_{k_1} \cap C_{k_2} \neq \emptyset$. Then, fuzzy overlapping communities, also referred to as covers, can be described by the cover matrix F [38]. In particular, a belonging coefficient, also called fuzzy membership degree, f_{ik} describes how strongly v_i belongs to the k-th cluster C_k. This strength is usually expressed by values $f_{ik} \in \mathbb{R}$ with $0 \leq f_{ik} \leq 1$ such that for each v_i, $\sum_{k=1}^{K} f_{ik} = 1$ [20]. Using this measure, none of the vertices can strongly belong to several communities. Vertices v_i whose membership degrees are equally distributed across communities, e.g., $f_{i1} = 0.5$ and $f_{i2} = 0.5$, represent perfect bridges between these communities [38].

We define the predominant community C_k^i of a vertex v_i to be that of the highest membership degree ${f_i}^{\max}$. The fuzziness ${f_i}^{\text{fuz}}$ of a vertex v_i is then described by $1 - {f_i}^{\max}$. Vertices whose fuzziness is greater than 0.5 are considered as extremely fuzzy vertices because they do belong to their predominant community with less than 50% certainty. If a vertex belongs to several communities equally, e.g., $f_{i1} = 0.5$ and $f_{i2} = 0.5$, the predominant community is selected randomly from them. Based on the predominant community structure, we construct aggregated graphs, also called meta-graphs, $G^\theta = (V, E, V_{\text{meta}}, E_{\text{meta}})$ by collapsing subgraphs $G_k^\theta = (MV_k^\theta, ME_k^\theta)$ with $MV_k^\theta \subseteq C_k$ and $ME_k^\theta \subseteq MV_k^\theta \times MV_k^\theta$ into meta-vertices $mv_k^\theta \in V_{\text{meta}}$ and by transforming inter-cluster-edges e_j of G into meta-edges $me^\theta \in E_{\text{meta}}$. The weight w_{me^θ} of a meta-edge $me^\theta(mv_1^\theta, mv_2^\theta)$ is equal to the sum of weights of all the inter-cluster-edges between vertices of the two subsets MV_1^θ and MV_2^θ. Of course, meta-edges can also connect a meta-vertex mv_k^θ with a single vertex v_i, aggregating the edges between those.

The aggregation level of the graph depends on the threshold $\theta \in \mathbb{R}[0,1]$, as only vertices v_i with ${f_i}^{\max} \geq \theta$ are aggregated into the respective meta-vertex mv_k^θ of their predominant community C_k^i (see schematic illustration in Figure 2). This threshold is used as determining parameter for our degree-of-interest function. An aggregated graph G^θ may therefore consist of vertices v_i and meta-vertices mv_k^θ, where G^0 contains meta-vertices only. G^1 represents a second special case, as it aggregates only non-fuzzy vertices (${f_i}^{\text{fuz}} = 0$). Finally, $G^{>1}$ is equal to the original complete graph G containing vertices only (in the following the superscript $^{>1}$ will be dropped for G and community subsets MV_k). We define the certainty of a meta-vertex mv_k^θ as the average belonging coefficient ${f_k}^{\text{avg}}$ of all its vertices $v_i \in MV_k^\theta$: $\sum \frac{{f_i}^{\max}}{|MV_k^\theta|}$ representing how strongly the members belongs to the community, where ${f_k}^{\text{avg}} = 1$ for a completely certain community. The certain community subsets $MV_k^{1.0}$ with ${f_i}^{\max} = 1.0$ for all $v_i \in C_k$ are referred to as the core communities.

3.2 Graph Layout

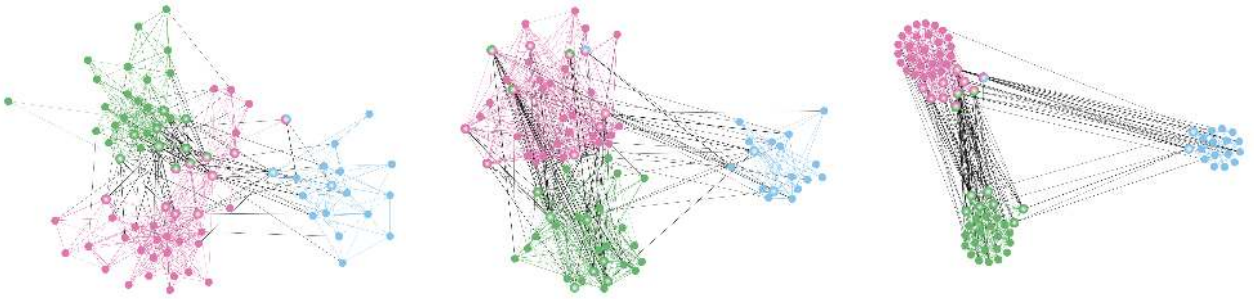
As mentioned in Section 2, force-directed layouts can help reveal communities because vertices that are highly connected are often—but not always—positioned close to each other (see Figure 3(a)). Our visualization approach uses the Gestalt principles of closure (spatial proximity) as well as texture patterns [33] to encode the community memberships using a hierarchical layout approach. In particular, we employ a regular and highly symmetric disk-like layout for the core communities to visually differentiate their (certain) vertices from other (fuzzy) vertices (see Figure 3(c)). For background literature on texture perception, we refer to Julesz) [29, 30]. Mirror symmetry is known to be recognized preattentively [23, 48]. Therefore, our layout is designed to show mirror symmetry; in fact, the regular disk-shaped layout shows mirror symmetry along multiple axes to facilitate preattentive perception and hence effortless and efficient differentiation between certain and fuzzy community members. Regularity is one of the primary texture dimensions [42]. Therefore, we generate layouts with regular patterns for community cores in order to make their texture appearance different from that of fuzzy vertices, adding a visual cue on top of mirror symmetry. Furthermore, the distance of fuzzy vertices to their community gives some indication of their fuzziness.

Our hierarchical approach is related to the divide-and-conquer approach by Wang et al. [52], in which each cluster is laid out separately before the clusters are composed to form the graph (see Figure 3(b)):

Algorithm 1 Calculate the layout of G

```
1: L ← layout(G^0)  {lay out aggregated graph}
2: for k=1 to K do
3:     L_k ← sublayout(G_k)  {lay out community subgraph}
4:     integrate L_k into L
5: end for
```

Similar to the approach by Wang et al., the global layout (line 1 of Algorithm 1) is derived based on a layout algorithm that considers the size of the sublayouts, which is proportional to the number of vertices



(a) (b) (c)

Fig. 3. Comparison of different layout approaches for an example graph: a personal friendship network of a faculty of a UK university, consisting of 81 vertices (individuals) and 817 weighted connections [38]. The network was clustered using the algorithm by Gregory [20] for fuzzy overlapping communities. (a) The graph is laid out with the force-directed Fruchterman-Reingold model [18] regardless the graph's community structure. (b) The layout is derived using a divide-and-conquer approach, where each cluster is laid out separately using the force-directed Fruchterman-Reingold model before the clusters are composed to form the graph. Shared (fuzzy) nodes are thereby included in the subgraph of their predominant cluster. (c) The layout is derived using our extended divide-and-conquer approach that includes the fuzziness of nodes, i.e., the different belonging coefficients, in the calculation. Vertices with certain cluster membership are visually encoded by regular and symmetric disk-shaped layouts.

the subgraphs G_k contain. In contrast to their approach, we incorporate an additional division step to derive the sublayouts (both alternatives are compared in Figures 3(b) and 3(c)):

Algorithm 2 Calculate the sublayout of G_k

1: $MV_k^{1.0} \leftarrow$ all $v_i \in C_k$ with $f_i^{\max} = 1.0$ {extract core-vertices}
2: $G_k^{\text{core}} \leftarrow (MV_k^{1.0}, MV_k^{1.0} \times MV_k^{1.0})$ {extract complete core graph}
3: $L_{\text{core}} \leftarrow \text{layout}(G_k^{\text{core}})$ {lay out core graph}
4: $L_k \leftarrow \text{layout}(G_k, L_{\text{core}})$ {lay out subgraph}
5: **return** L_k

To achieve a regular and symmetric disk-shaped layout for a community core, the core-vertices are positioned based on a layout derived for the complete core-graph G_k^{core} (lines 1–3 of Algorithm 2). In particular, G_k^{core} is laid out using the force-directed Kamada-Kawai model [31] based on spring forces proportional to the graph theoretic distances, as this creates very regular layouts.

In the fourth step of Algorithm 2, all non-core-vertices $v_i \in G_k$ with $f_i^{\max} < 1.0$ are positioned around the core-layout L_{core} using a force-directed approach of attracting and repulsive forces that depend on the belonging coefficients f_{ik} and are applied to non-core-vertices only. The repulsive force between any two vertices v_i is small, where the forces between a vertex v_i and the "pseudo-vertices" representing C_k^i, positioned at the center of L_{core}, depends on $f_i^{\max}$, i.e., the weaker the membership degree of v_i to its predominant community C_k^i, the stronger is the force of repulsion. This produces distances of vertices v_i to the center of the core that are nearly proportional to their fuzziness. Furthermore, the non-core-vertices v_i are pulled toward those communities to which they also belong. This is achieved by attracting forces between a vertex v_i and all its communities C_k represented by "pseudo-vertices", which are again proportional to the respective membership degrees f_{ik}. For example, for a vertex v_i with the belonging coefficients of $f_{i1} = 0.5$, $f_{i2} = 0.4$, and $f_{i3} = 0.1$, the attracting force to community C_2 will be much higher than for C_3 but highest for C_1. Whereas the repulsive forces guarantee a minimum distance between the core-graphs and the fuzzy vertices, the attracting forces keep the vertices within a maximum distance from the core-graph. Vertices will therefore mostly, i.e., if $f_i^{\max} >> f_{ik}$ for each $C_k \neq C_k^i$, be positioned closest to their predominant community and toward one or two other communities to which it significantly belongs. In the final step (line 4 of Algorithm 1), the sublayouts are integrated into the global layout L to form the overall layout (see graphs at $\theta > 1$ in Figure 4).

We provide several alternatives for the global layout: a force-directed layout, either the Fruchterman-Reingold model [18] or the Kamada-Kawai model [31], and a circular layout. For the force-directed approaches, the size of the sublayout is described by 2D rectangular areas. In contrast, for the circular approach it is described by an arc of particular length. For our case studies, we use the Fruchterman-Reingold model as force-directed layout, as it provides layouts that are aesthetically more pleasing, at least for these datasets.

3.3 Layered Visualization Model

Our visualization approaches provides users with graphs at different aggregation levels described by a degree-of-interest function. This function aggregates nodes of a particular degree of fuzziness described by the threshold θ. The layout of the original complete graph (see Figure 4 right) is used as a basis to derive the layout for any aggregated graph G^θ. Meta-nodes (mv_k^θ) are thereby positioned at the barycenter of the vertices $v_i \in MV_k^\theta$ they aggregate. Due to our hierarchical layout approach, meta-nodes are positioned near the center of the core-graph G_k^{core}. Therefore, changing the level of detail by in- or decreasing the threshold θ, produces a sequence of plots of the graphs G^θ at different abstraction levels that preserves the mental map (see Figure 4). This holds for both layout approaches, the force-directed and the circular layout approach.

Figure 4 shows the complete sequence of aggregation states for a small example graph. The graph consists of 12 vertices, 3 communities, and 3 nodes whose community memberships are fuzzy. Starting with threshold $\theta = 0$, the graph contains meta-vertices only and represents the fully aggregated graph. Increasing the threshold to $\theta = 0.7$ separates node e with $f^{\max} = 0.664$ from C_3. A further increase of θ to $\theta \geq 0.91$ separates nodes d and i both with $f^{\max} = 0.908$ from C_1 and C_2, respectively, and results in an aggregated graph whose meta-vertices contain core-vertices only, i.e., $f_k^{\text{avg}} = 1.0$ for all three aggregated subgraphs. Setting the threshold $\theta > 1.0$ finally breaks up the partially aggregated graph to its single vertices v_i.

3.4 Node-Oriented Visual Mapping

Besides the position, our visualization approach uses further visual attributes of nodes to encode the community memberships and to emphasize the degree of fuzziness of shared nodes (see Figure 5). To differentiate between single nodes v_i and meta-nodes mv_k^θ, they are represented using different shapes: circles and stars for v_i and mv_k^θ, respectively. To allocate nodes and meta-nodes to a cluster C_k, each cluster C_k is assigned a color based on a colormaps created with ColorBrewer [1] and nodes (meta-nodes) are colored with respect to their predominant community C_k^i (community C_k). We use two different approaches, the modification of lightness and geometry of the object's shape, to visualize the node's and meta-node's fuzziness, respectively, to differentiate between both types of nodes. Both mappings repre-

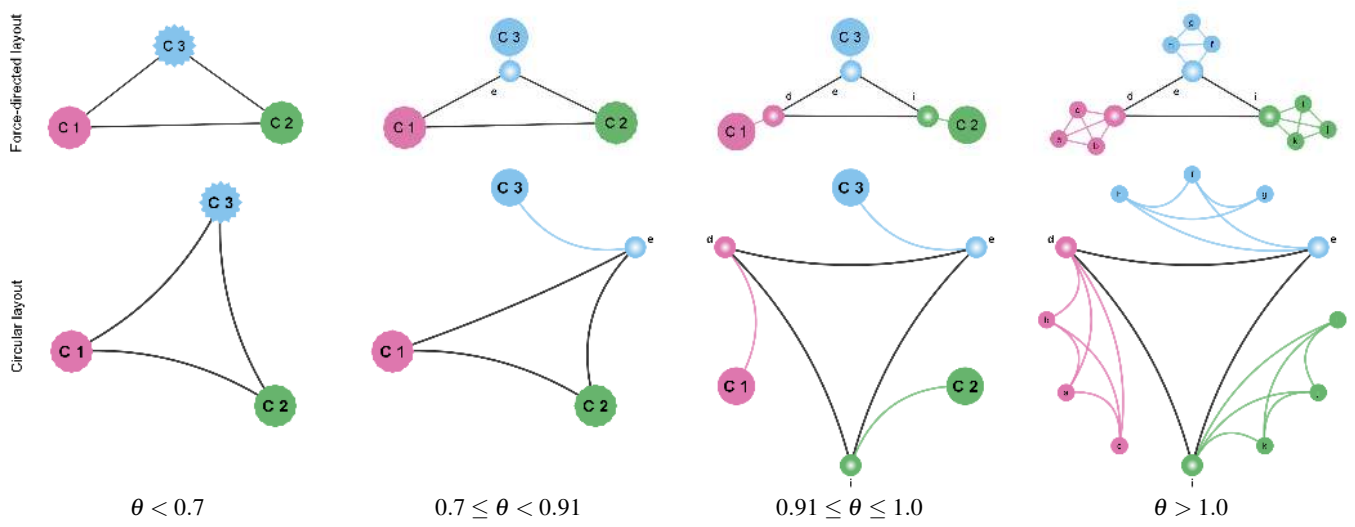
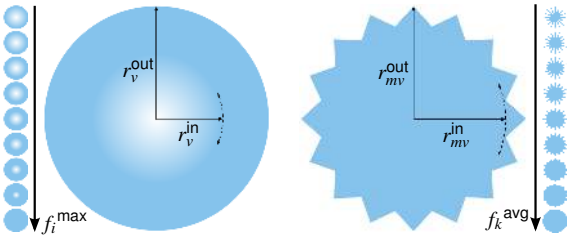


Fig. 4. Sequence of graphs G^θ showing the fuzzy overlapping community structure for an example graph with $|V|$=12 and K=3 at different levels of detail, i.e., using different thresholds θ, starting with the fully aggregated graph (left) and ending with the detailed graph (right). For comparison, the sequences are laid out with a force-directed (top row) and circular (bottom row) global layout L. Increasing θ first of all separates the fuzziest vertex of the graph (e) from its predominant community C_3 followed by the less fuzzy vertices d and i.



(a) Fuzziness of nodes. (b) Fuzziness of meta-nodes.

Fig. 5. Mapping the community membership fuzziness to visual attributes of the nodes representing (a) vertices v_i or (b) meta-vertices mv_k^θ. For vertices, the inner circle of the node defined by $r_v^{\text{in}} \sim 1 - f_i^{\max}$ is rendered with a radial color gradient. Meta-vertices, on the contrary, are represented by stars, where again the inner circle depends on the fuzziness $r_{mv}^{\text{in}} \sim f_k^{\text{avg}}$. The sequences show the visual mappings for $f_i^{\max} : [0.2, 1.0]$ and $f_k^{\text{avg}} : [0.6, 1.0]$, respectively.

sent common approaches to convey uncertainty [41]: the brighter (the more distorted) a shape, the higher the uncertainty.

To visualize the fuzziness of the community membership of a vertex v_i to C_k^i, nodes with $f_i^{\text{fuz}} > 0$ are rendered with a color gradient instead of a constant color (see Figure 5(a)). The gradient progresses radially from the center of the circle to the inner radius r_v^{in}, starting with white in the center and ending with the respective color for C_k^i at the inner circle. The annulus defined by r_v^{in} and r_v^{out} is rendered without gradient in the community color. The inner radius r_v^{in} describes the circle whose area is scaled compared to the outer circle area using the fuzziness: $r_v^{\text{in}} = \sqrt{f_i^{\text{fuz}} r^2}$. To emphasize fuzzy nodes ($f_i^{\text{fuz}} > 0$), these are rendered with a slightly increased radius compared to non-fuzzy nodes ($f_i^{\text{fuz}} = 0$). If a vertex belongs to several communities C_k with similar extent, the circle is divided into the respective number of segments whose size is proportional to the belonging coefficient f_{ik}, where each segment is rendered with the gradient as described before. We use a threshold of 10% for the similarity criterion in the examples of this paper, i.e., two membership degrees are regarded as similar if $f_i^{l+1} \geq 0.9 f_i^l$, where l denotes the index of the descendingly ordered membership degrees f_{ik} of v_i, starting with $f_i^0 = f_i^{\max}$. Similar to the approach by Itoh et al. [24], the segments and hence colors of the circle are arranged such that they are closest to the respective communities. This mapping shows which communities contribute significantly to the fuzziness of a vertex, where small coefficients f_{ik} remain disregarded by normalizing the significant coefficients.

For meta-nodes mv_k^θ, the certainty f_k^{avg} is represented by the fringe degree of the star (see Figure 5(b)). The outer radius r_{mv}^{out} of the star representing mv_k^θ describes the circle whose area is proportional to the number of vertices aggregated in mv_k^θ, i.e., $r_{mv}^{\text{out}} = \sqrt{|MV_k^\theta| r_v^{\text{out}}}$. The inner radius r_{mv}^{in} depends on the average strength: $r_{mv}^{\text{in}} = f_k^{\text{avg}2} r_{mv}^{\text{out}}$. The quadratic mapping of f_k^{avg} allows for an enhanced differentiation of community fuzziness. Using different visual mappings for the fuzziness of vertices and the sharpness of meta-vertices has the advantage that both types of vertices are clearly distinguishable from each other.

Besides the fuzziness of nodes, also the distribution of membership degrees f_{ik} of individual vertices v_i should be visualized because determining to what extent a vertex contributes to its communities is an important task when analyzing fuzzy overlapping communities. We decided to use pie charts and bar charts in the force-directed and circular node-link diagram, respectively. To ensure the readability of the charts, these are rendered not smaller than a user-specified minimum size, i.e., radius or width and length, respectively. To avoid visual overload in the force-directed layout, only selected nodes are rendered as pie charts, because these have a much bigger radius to make the individual segments clearly recognizable. In contrast, in the circular diagram, the bars are attached radially to the nodes instead of replacing them. For both charts, the segments are ordered descendingly by the fuzzy membership degrees f_{ik}.

Whereas in the force-directed layout approach nodes are connected by straight links, in the circular layout approach we use curved links to produce aesthetically pleasing diagrams. The curvature of a link connecting two nodes v_1 and v_2 positioned at θ_1 and θ_2 decreases with increasing angular distance, such that the link is straight for $\Delta\theta = 180°$. To differentiate intra-community edges ($e(v_1, v_2)$ with $C_k^1 = C_k^2$) from inter-community edges ($C_k^1 \neq C_k^2$), the former are rendered in the respective community color for C_k, while the latter are rendered in black. The edge weights (weights w_e of edges e_j and aggregated weights w_{me} of meta-edges me) are mapped to the width of the link.

3.5 Interaction Techniques

The most important interaction technique of our visualization approach is the support to continuously drill from the highest aggregation level (G^0) down to the most detailed level showing the complete graph

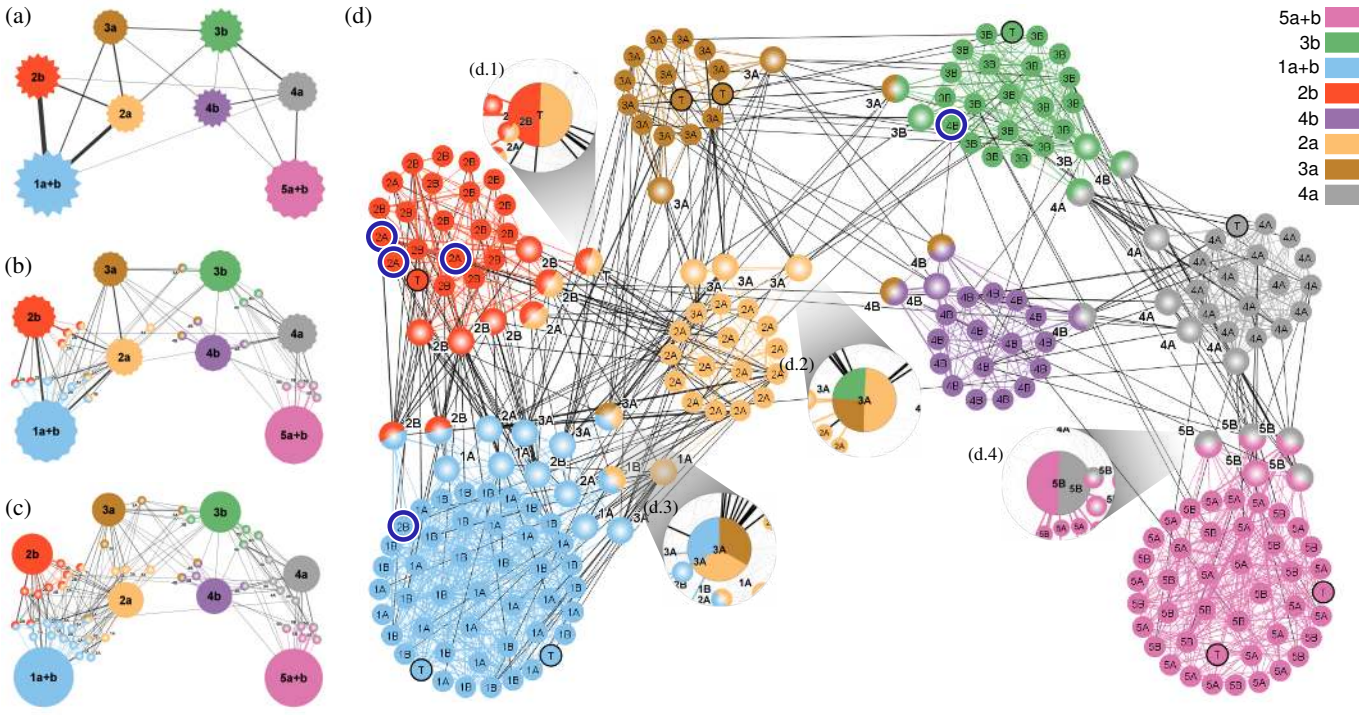


Fig. 6. Social contact network of 232 students and 10 teachers accumulated over one day. The node-link diagram is laid out using our force-directed layout approach. (a) Fully aggregated graph G^0. (b) The partially aggregated graph $G^{0.51}$ separates all extremely fuzzy nodes from their predominant communities. (c) The partially aggregated graph $G^{0.8}$ aggregates only core-vertices into meta-vertices. Using this graph, all nodes whose community membership is fuzzy can be analyzed and investigated. Four individuals have been selected and are therefore represented as pie charts (integrated into (d) as overlays) showing their membership degree distribution. (d) The complete graph G helps investigate core-nodes that fully belong to one community. The children highlighted by the dark blue circles are members of a different class than they were allocated to. The reason for this is that they have much more contact with children of their community than to their own class.

(G). Therefore, our visualization approach adopts the Visual Information Seeking Mantra [45]: it supports the user with an overview of the fuzzy community structure on the community and vertex level, with zoom and pan options, and with details-on-demand functions. The latter include the representation of the membership distributions of individual vertices of interest as well as the available tooltips that appear when hovering over nodes to obtain some exact values describing the properties of vertices and meta-vertices. The zoom and pan options ensure the readability of the fuzzy community memberships on the vertex level, in particular, for large graphs. Besides, selecting a vertex v_i helps analyze its relations because it highlights all edges of v_i and fades all other links to the background.

4 Case Studies

The aim of analyzing fuzzy overlapping communities is to differentiate fuzzy from non-fuzzy vertices. For fuzzy vertices, it is important to identify to which communities a vertex belongs and to what extent it contributes to these communities. To demonstrate the usefulness of our approach concerning these analysis tasks, we performed two case studies in the domains of social networking and biological networks. The networks were clustered with the algorithm by Gregory [20] for fuzzy overlapping communities. The community detection algorithm can cluster weighted or unweighted undirected networks into K communities using a modularity-maximization approach.

4.1 Social Network

In social networks, communities overlap because individuals usually belong to different communities, they have a family, are member of different circles of friends, sports clubs, and the like [17]. Due to limited time and resources, these overlaps are usually fuzzy as individuals cannot be fully involved with all communities to which they belong.

4.1.1 Background

Our first case study is based on a social network modeling the contacts between students and teachers of a primary school [47], a frequently used benchmark dataset for dynamic contact networks. The analysis of contacts between children at school helps understand mixing patterns between classes and grades, which in the end helps quantify transmission possibilities for respiratory infections [47]. Thereby, it is important to identify children that are in strong contact with other classes than their own. In this case study, we want to show how this can be done using our visualization approach to analyze the fuzzy overlapping community structure. The time-resolved face-to-face proximity of children and teachers was recorded using radio frequency identification device badges in 20 second intervals over 2 days. The data comprises contacts between 232 children of grades 1–5 and 10 teachers across 10 classes, two classes per grade (1A, 1B, ... 5A, 5B). The contact data was aggregated into two weighted contact networks, one for each day, where vertices v_i represent individuals and edges e_j represent contacts between them. The weight w_j represents the cumulative time spent by two individuals in face-to-face proximity, over one day. Most contacts are of very short duration, although contacts of very different durations were observed. Therefore, edges between individuals who spent less than 2 minutes during a day were removed. For further details about the study of contact networks in a primary school and the derived data, we refer to [47]. We clustered the networks for both days using the modularity-maximization approach of Gregory's algorithm, which resulted in a set of 8 communities.

4.1.2 Results

For the space constraints of this paper, we will concentrate on the graph of day 1 only. The clustering algorithm detected the different

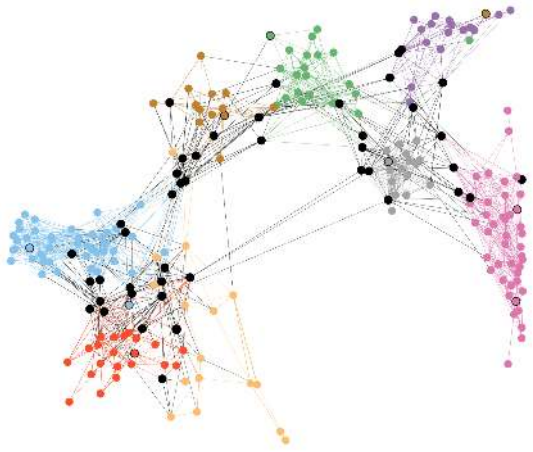


Fig. 7. Shows the same graph G as Figure 6(d) laid out with the force-directed Fruchterman-Reingold model [18]. Using a force-directed layout approach, coloring of nodes with respect to their community (same color scheme as in Figures 6 is used) to which they belong and highlighting shared nodes in a different color (here black), is a common approach to analyze crisp overlapping communities.

classes as individual but highly connected communities (see Figure 6), except for the classes of the 1st and 5th grade for which classes a and b were detected as one cluster (1a+b, 5a+b). Meta-nodes *mv* are labeled with respect to the class(es) they represent. In the following, we use lower case letters for the community labels to differentiate between classes (e.g., 2A) and communities (e.g., 2a). Nodes that represent children are labeled by the class to which they belong and teachers are labeled with the letter "T". Since the labels can be hidden on demand, the teachers are furthermore emphasized by framing the respective nodes with black circles.

We analyze the fuzzy community structure in a top-down approach using our layered visualization model, i.e., by incrementally increasing the threshold θ that determines the degree of aggregation, and using the details-on-demand options. Starting with the fully aggregated graph G^0, we can investigate the fuzziness on the community level (see Figure 6(a)). Due to the fringe degree of the meta-nodes, it becomes clear that all communities are relatively certain, where the least fuzzy communities are 3a, 4a, and 5a+b. The reason for this may be that the children in these (joined) classes are in much more contact among each other than with children of other classes. G^0 furthermore shows that the strength of contact between classes of the same grade or directly consecutive grades is much higher compared to contacts between classes that differ in their grade by more than one year.

Increasing the threshold to $\theta = 0.35$ separates one child with $f^{\text{fuz}} = 0.66$ from the predominant community 1a+b, who actually belongs to class 3A. Using the details-on-demand option by selecting the node (see pie chart (d.3) in Figure 6), we find out that this node almost equally participates in the three communities 2a, 3a, and 1a+b. Increasing θ to 0.51 separates all extremely fuzzy nodes from their predominant communities, which comprise only 21 of 242 individuals ($\approx 9\%$) (see Figure 6(b)). It becomes clear that all extremely fuzzy nodes are in strong contact with other communities too, since many of the outgoing edges are rendered in black. The only exception is a child of class 5B allocated to 5a+b. Using the selection based highlighting of relations and the pie chart (see (d.4) in Figure 6), we find out that although this child is solely in contact with children from 5a+b, she or he is allocated to another community (4a), too. The teacher visible in all graphs $G^{>0.5}$ is positioned inbetween the communities 2a and 2b, which suggests that she or he is in equal contact to both classes. We can confirm this by selecting the node to investigate the membership distribution (see pie chart (d.1) in Figure 6(d)).

Further increasing θ shows that at $\theta = 0.8$ (see Figure 6(c)), all remaining meta-nodes are completely certain, as they are rendered as circles. Hence, $G^{0.8}$ shows that about 44 individuals ($\approx 19\%$) of the graph are fuzzy concerning their community membership. Among these individuals, there are several children whose predominant community is not equal to the class to which they belong. The respective node positions imply that these children are also a member of the class to which they belong. By individually selecting these children, we can investigate the respective pie charts (e.g., pie chart (d.2) in Figure 6) and confirm this observation. Finally, the complete graph G reveals that some children are preferably in contact with children from other classes and thus fully allocated to these communities (see children highlighted in Figure 6(d) by the dark blue circles).

As mentioned in Section 2, force-directed layouts may help reveal communities. Figure 7 shows the complete graph G laid out using a force-directed approach. The clustering derived implictly by the force-directed layout is approximately consistent with the algorithmically derived community structure, which becomes clear by the color mapping on nodes, where the members of community 2a are spread over the diagram. However, using a layout approach to cluster the graph does neither reveal the fuzziness of community memberships nor their distribution, i.e., for many fuzzy vertices it cannot be perceived to which communities they belong.

Analyzing the fuzzy overlapping communities of this social network helps reveal mixing patterns between children of different classes and age groups. This case study showed that contacts occur mostly within the class or at least age group and only few children are in preferred contact with children from other classes or grades.

4.2 Protein-Protein Interaction Network

In our second case study, we concentrate on protein-protein interaction (PPI) networks. The analysis of PPI networks is of high value for investigating human systems and helps understand complex cellular mechanisms and processes [27, 28]. In particular, the global organization of such networks, i.e., their classification into structural subunits, is of great interest, as groups of proteins usually comprise the same specific cell function. This is because cellular functions are not performed by individual proteins but groups of proteins (modules or protein complexes). As in other application domains, communities in PPI networks do overlap, since many proteins can play several distinct roles in different contexts [40], but they cannot interact with different proteins at the same time [10, 27]. Community detection methods can therefore be used to identify communities of proteins based on their interactions, to analyze their overlapping nature and to identify to what extent shared (fuzzy) proteins are involved in particular functions.

4.2.1 Background

In our second case study, we analyze an undirected PPI network based on data of a study by Jonsson and Bates [28]. They derived interactions of proteins that are known to be susceptible to mutations leading to cancer. Based on that, an extensive human PPI network (17,018 vertices and 99,507 edges) was constructed using computational methods. The network is undirected and weighted, where edge weights represent confidence scores for the respective interactions. We analyze the same subnetwork of this PPI network as Jonsson and Bates did in their study, comprising 1,253 weighted interactions between 232 proteins. The network was clustered using the modularity-maximization approach of Gregory's algorithm, which extracted a set of 14 communities.

As described in the original paper [28], proteins that were transcribed from mutated genes are classified as cancer proteins. The network comprises 174 non-cancer and 58 proteins, where cancer proteins were furthermore classified into somatic (S) or germline (G) mutations. The labels of nodes that represent somatic or germline cancer proteins are extended by the identifiers ":CS" and ":CG" respectively, or using ":C" if no information about the cancer type is available. As the labels can be hidden on demand, the cancer proteins are furthermore emphasized by framing the respective nodes with a black or gray circle for somatic and germline mutations, respectively.

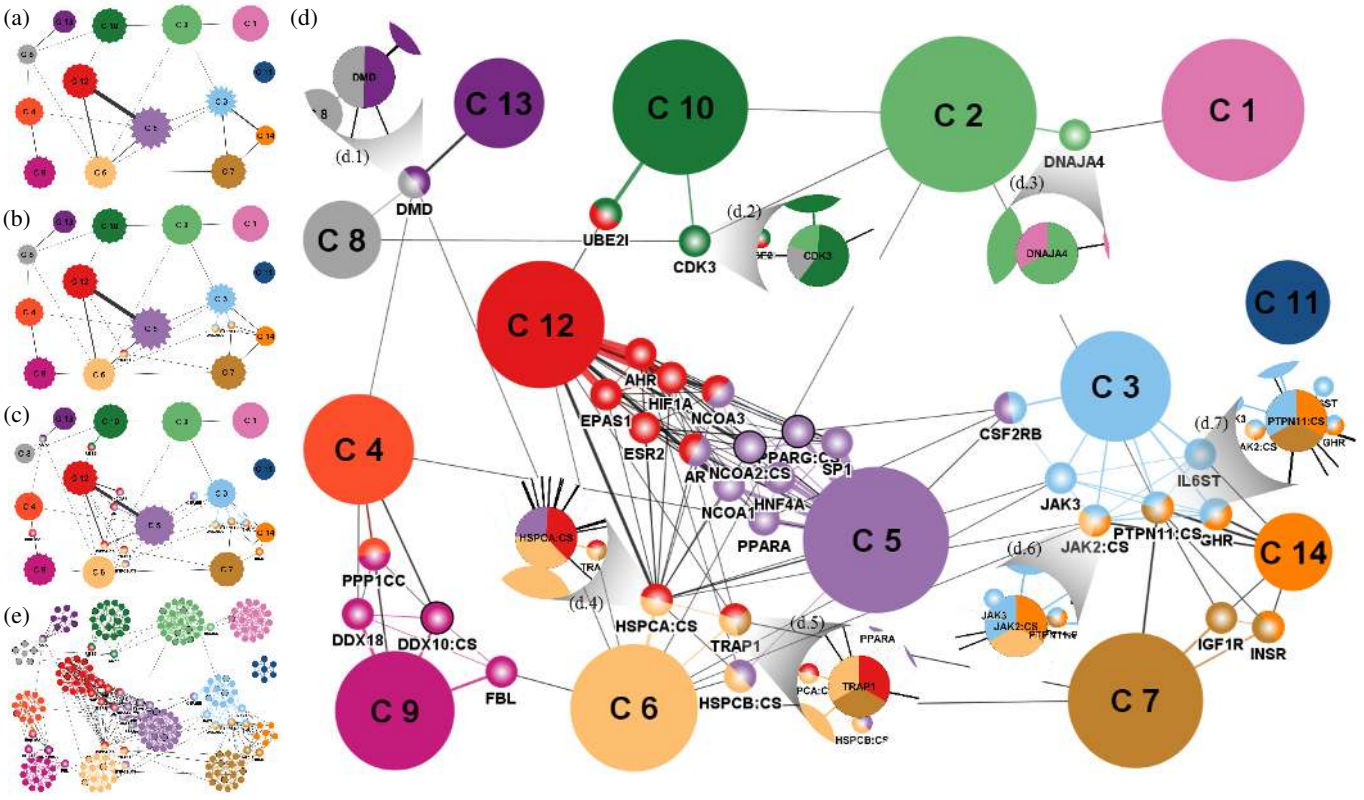


Fig. 8. Protein-protein interaction network comprising 1,253 weighted interactions between 232 proteins of which 174 are non-cancer and 58 are somatic and germline cancer proteins marked by black and gray circles, respectively. (a) Fully aggregated graph G^0. The partially aggregated graphs (b) $G^{0.34}$ and (c) $G^{0.51}$ separate the fuzziest proteins and $G^{0.51}$ furthermore all extremely fuzzy nodes ($f^{\text{fuz}} > 0.5$) from their predominant communities. (d) All remaining meta-nodes of the partially aggregated graph $G^{0.86}$ are completely certain. Some suspicious proteins have been selected in G to reveal their membership distributions using the pie charts, which have been incorporated into (d). (e) G shows that all germline cancer proteins are contained in the cores of communities, hence participating in central hubs rather than peripheral ones.

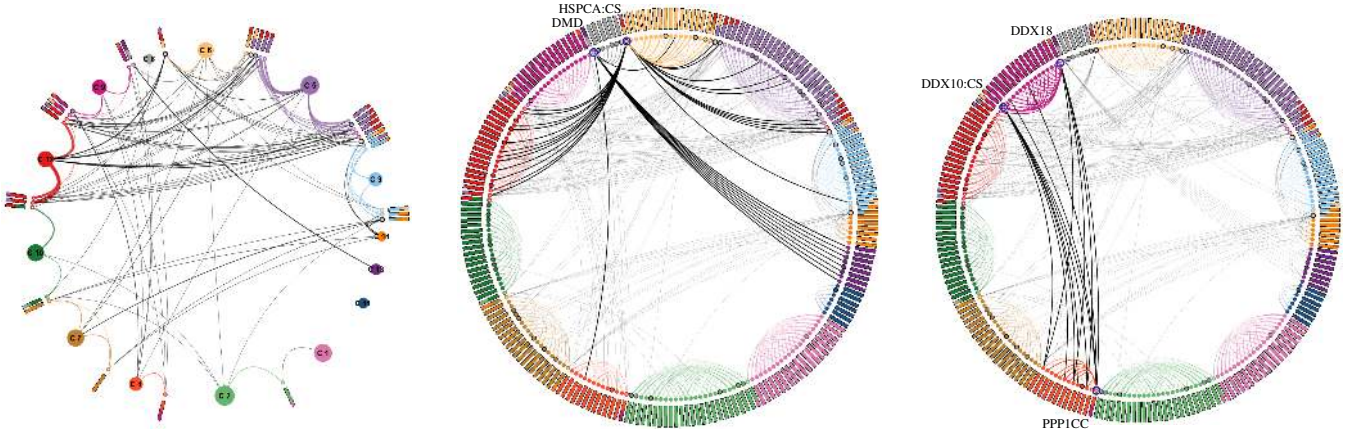
4.2.2 Results

Again, we analyze the fuzzy communities in a top-down approach starting with the highest level of aggregation G^0. Due to the shape of meta-nodes, we figure out that there exist four communities that are completely certain (C_1, C_{11}, C_{13}, and C_{14}) (see Figure 8(a)), i.e., they do not contain any protein that is involved in any other community; besides, hovering over meta-nodes to inspect the exact average belonging coeffcients revealed that even the fuzziest community (C_3) is relatively certain with $f^{\text{avg}} = 0.84$.

Increasing θ to 0.34 separates the two somatic cancer proteins PTPN11 and JAK2 as well as the protein TRAP1 from their predominant communities (see Figure 8(b)). We select these nodes to investigate the membership distributions (see pie charts d.5-d.7 in Figure 8(d)), which shows that all three proteins are equally involved in the three functional units. PTPN11 is allocated to C_3, C_{13}, and C_7, which build a huge protein complex responsible for signal transductions related to the "Jak-ST AT" signaling and adipocytokine signaling but also cell communication at adherens junctions. JAK2 is involved in the functional units C_3, C_{13}, and C_6, where C_6 is mainly responsible for cell growth and death in apoptosis, the process of the programmed cell death, as well as signal transductions related of the immune system. In comparison to JAK2, TRAP1 is furthermore involved in the process of antigen processing of the immune system, which is the function of C_{12}. Increasing θ to 0.51 (see Figure 8(c)) separates all extremely fuzzy nodes from their predominant communities, which comprise 13 of 232 proteins ($\approx 6\%$). Further increasing θ shows that at $\theta = 0.86$ (see Figure 8(d)), all remaining meta-nodes are completely certain because they are rendered as circles. Hence, $G^{0.86}$ shows that 31 proteins ($\approx 14\%$) of the graph are fuzzy concerning their community membership.

The communities C_5, C_6, and C_{12} build a huge functional complex: all three units comprise functions related to the immune system. $G^{0.86}$ reveals that these functional units are held together by several proteins involved in these different functions. In particular the functional units C_5 and C_{12} share many proteins with different extents, which becomes clear by the node positioning (many proteins are positioned in the space between the respective two cores) and color gradients used for rendering (see Figure 9(c)) but also by the huge amount of inter-community links (also see black curves between proteins of C_5 and C_{12} in the circular layout of Figure 9(a)). Again by selecting the respective nodes, we can investigate the membership distributions using the pie charts. The somatic cancer proteins HSPCA, e.g., is involved in all three functional units (see pie chart (d.4) in Figure 8). Figure 9(b) shows that HSPCA has a particularly high degree of interactions in general compared to other cancer but also non-cancer proteins.

Besides functional complexes, also single functional units (communities) share one or more proteins. We selected the respective proteins to investigate their belonging coefficients using the pie chart (see, e.g., d.1–d.3 in Figure 8). The protein DMD, e.g., is involved in the functional units C_8 and C_{13}, which are both responsible for cell motility of the cytoskeleton. In fact, DMD takes a central and important role, as the function is carried out by both communities as a whole, where DMD serves as mediator. The shared role of DMD also becomes clear looking at the circular visualization, which shows that DMD interacts with all proteins of C_8 and all proteins of C_{13}. The proteins positioned between C_4 and C_9 (DDX10, DDX18, PPP1CC) are shared by units of different functions. Where C_4 has the function of another type of



(a) $G^{0.86}$ (b) G with DMD and HSPCA as selected proteins (c) G with DDX10, DDX18, and PPP1CC as selected proteins

Fig. 9. The protein-protein interaction network from Figure 8 shown with our circular layout. (a) The partially aggregated graph $G^{0.86}$ (see also Figure 8(d)). Images (b) and (c) show the complete graph G, each with different proteins selected to highlight all interactions of these proteins and fade all others to the background.

signal transduction (Wnt signaling), C_9 is responsible for the carbohydrate metabolism and insulin signaling. Also the circular visualization (see Figure 9(c)) shows this overlap, as DDX10 and DDX18 have (PPP1CC has) a huge degree of contact with proteins from C_9 (C_4).

The complete graph G (see Figure 8(e)) shows that all germline cancer proteins are contained in the community cores, hence participating in central hubs rather than peripheral ones. Therefore, somatic cancer proteins tend to be at the interfaces of different protein communities. Analyzing the fuzzy overlapping community structure of a PPI network, as presented here, helps investigate which proteins are involved in more than one community and hence functional unit. Using our visualization approach, we could confirm the findings of Jonsson and Bates [28], who presented their results tabularly and visualized single functional units as well as functional complexes using node-link diagrams within which shared proteins are barely recognizable.

5 Conclusion

Communities play a fundamental role in real-life networks, where depending on the system they represent, communities may overlap. Our visualization approach supports the investigation of fuzzy overlapping communities but could be easily applied to crisp communities, too. The only difference compared to fuzzy overlaps refers to the belonging coefficients, which are binary for crisp overlaps, i.e., each object fully belongs to one or more communities. Hence, crisp overlaps could be visualized in a straightforward way using our approach by normalizing all belonging coefficients of a vertex with the number of communities to which it belongs.

Compared to the existing work on visualizing disjoint or crisp overlapping communities in networks, our visualization approach has the advantage that it takes into account the fuzziness of the nodes memberships. Existing layout approaches for (hierarchically) clustered networks do not incorporate any information on shared nodes, regardless whether crisp or fuzzy. Previous divide-and-conquer approaches at least produce layouts that position nodes of the same community closely, at the same time at positions contrasted from other communities. Nevertheless, shared nodes are positioned strictly within one of their communities and may be far away from other communities to which they also belong. In contrast, our force-directed layout approach positions shared nodes close to the barycenter of its communities but closest to their predominant communities. Of course, the position itself is not a non-ambiguous indicator for the membership distribution of a vertex. Therefore, the color gradient of the nodes indicates the degree of fuzziness, where the pie charts can be used to investigate their exact membership distributions.

Our case studies showed that our visualization approach can be used to analyze the fuzziness of a graphs community structure and to investigate the fuzziness of shared nodes. Users can examine to which communities a vertex belongs and to which extent it is involved in these communities. Furthermore, the visualization approach can be used to identify nodes that are outliers with respect to their community memberships and topology, i.e., their relation structure, as shown in the first case study. Besides the layout approach and the other visual mappings used, our visualization approach greatly benefits from different levels of detail supported by our layered visualization model. Interactively changing the degree of aggregation allows us to analyze the fuzzy overlapping community structure at different levels of detail, starting with highest aggregation level, over different stages of partially aggregated graphs and ending with the complete graph that shows all nodes individually. The subsequence of partially aggregated graphs attracts the user's attention to the fuzziest of the shared nodes first, followed by decreasingly fuzzy nodes. The last graph of this subsequence is probably the most important one, as it separates only shared nodes from their predominant community and aggregates all non-shared nodes. Therefore, users can analyze the fuzziness and memberships of all shared nodes, without being distracted by the detailed structure of the communities' cores. Since the layouts of all graphs of the sequence are based on the same layout derived for the original complete graph, the sequence of layouts preserves the mental map.

In the future, we want to extent our visualization toward the visualization of directed graphs instead of just displaying undirected graphs. Furthermore, we want to combine our degree-of-interest function with the focus-and-context approach by applying it to one or more subsets of communities only, while representing the others in an aggregated way. We also plan to improve the color assignment such that adjacent communities have colors of high contrast so that they are easily distinguishable. In general, the colormap used in the examples of this paper can be replaced with any other colormap, e.g., one that is suitable for people with color vision deficiency [36]. Finally, we want to perform a longitudinal study to evaluate our visualization approach in the application domain of social networks. As there is no preliminary work on visualizing fuzzy overlapping communities, it is hard to make direct comparisons.

Acknowledgments

This work was in part supported by DFG SFB 716 D.5. We thank SocioPatterns (`http://www.sociopatterns.org`) for providing the data on social contact networks, Josson and Bates for the interactome data, and Steve Gregory for his community detection algorithm.

REFERENCES

[1] ColorBrewer http://colorbrewer2.org/, May 2011.

[2] J. Abello, S. G. Kobourov, and R. Yusufov. Visualizing large graphs with compound-fisheye views and treemaps. In *Proceedings of Graph Drawing*, pages 431–441, 2004.

[3] B. Adamcsek, G. Palla, I. J. Farkas, I. Derényi, and T. Vicsek. CFinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, 22(8):1021–1023, 2006.

[4] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.

[5] B. Alper, N. Riche, G. Ramos, and M. Czerwinski. Design study of linesets, a novel set visualization technique. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2259–2267, 2011.

[6] D. Archambault, T. Munzner, and D. Auber. Grouseflocks: Steerable exploration of graph hierarchy space. *IEEE Transactions on Visualization and Computer Graphics*, 14(4):900–913, 2008.

[7] M. Balzer and O. Deussen. Level-of-detail visualization of clustered graph layouts. In *Proceedings of APVIS'07*, pages 133–140, 2007.

[8] M. Baur and U. Brandes. Multi-circular layout of micro/macro graphs. In *Proceedings of Graph Drawing*, pages 255–267, 2008.

[9] N. Cesario, A. Pang, and L. Singh. Visualizing node attribute uncertainty in graphs. In *Proceedings of the SPIE (Visualization and Data Analysis 2011)*, 2011.

[10] J. Chen and B. Yuan. Detecting functional modules in the yeast protein–protein interaction network. *Bioinformatics*, 22(18):2283–2290, 2006.

[11] C. Collins, S. Carpendale, and G. Penn. Visualizing uncertainty in lattices to support decision-making. In *Proceedings of Eurographics/IEEE VGTC Symposium on Visualization*, pages 51–58, 2007.

[12] C. Collins, G. Penn, and S. Carpendale. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1009–1016, 2009.

[13] G. di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.

[14] P. Eades and Q.-W. Feng. Multilevel visualization of clustered graphs. In *Proceedings of Graph Drawing*, pages 101–112, 1997.

[15] N. Elmqvist and J.-D. Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2010.

[16] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, 2010.

[17] L. C. Freeman. *The Development of Social Network Analysis: A Study in the Sociology of Science*. Empirical Press, BookSurge, 2004.

[18] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

[19] S. Gregory. A fast algorithm to find overlapping communities in networks. In *Machine Learning and Knowledge Discovery in Databases*, volume 5211, pages 408–423. 2008.

[20] S. Gregory. Fuzzy overlapping communities in networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(02):P01017, 2011.

[21] H. Griethe and H. Schumann. The visualization of uncertain data: methods and problems. In *Proceedings of SimVis '06*, pages 143–156, 2006.

[22] J. Heer and D. Boyd. Vizster: Visualizing online social networks. In *Proceedings of the 2005 IEEE Symposium on Information Visualization*, pages 32–39, 2005.

[23] L. Huang, H. Pashler, and J. Junge. Are there capacity limitations in symmetry perception? *Psychonomic Bulletin & Review*, 11:862–869, 2004.

[24] T. Itoh, C. Muelder, K.-L. Ma, and J. Sese. A hybrid space-filling and force-directed layout method for visualizing multiple-category graphs. In *Proceedings on IEEE Pacific Visualization Symposium*, pages 121–128, 2009.

[25] C. R. Johnson. Top scientific visualization research problems. *IEEE Computer Graphics and Applications*, 24(4):13–17, 2004.

[26] C. R. Johnson and A. R. Sanderson. A next step: visualizing errors and uncertainty. *IEEE Computer Graphics and Applications*, 23(5):6–10, Sept. 2003.

[27] P. Jonsson, T. Cavanna, D. Zicha, and P. Bates. Cluster analysis of networks generated through homology: automatic identification of important protein communities involved in cancer metastasis. *BMC Bioinformatics*, 7(1):2, 2006.

[28] P. F. Jonsson and P. A. Bates. Global topological features of cancer proteins in the human interactome. *Bioinformatics*, 22(18):2291–2297, 2006.

[29] B. Julesz. *Foundations of Cyclopean Perception*. University of Chicago Press, 1971.

[30] B. Julesz. A theory of preattentive texture discrimination based on first-order statistics of textons. *Biological Cybernetics*, 41:131–138, 1981.

[31] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.

[32] D. E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison Wesley, 1993.

[33] K. Koffka. *Principles of Gestalt Psychology*. Harcourt, Brace, 1935.

[34] A. Lázár, D. Ábel, and T. Vicsek. Modularity measure of networks with overlapping communities. *Europhysics Letters*, 90(1):18001, 2010.

[35] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proceedings of 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, pages 1–5, 2006.

[36] G. M. Machado, M. M. Oliveira, and L. A. F. Fernandes. A physiologically-based model for simulation of color vision deficiency. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1291–1298, 2009.

[37] P. Mutton, P. Rodgers, and J. Flower. Drawing graphs in Euler diagrams. In *Diagrammatic Representation and Inference*, volume 2980, pages 66–81. Springer, Berlin, Heidelberg, 2004.

[38] T. Nepusz, A. Petróczi, L. Négyessy, and F. Bazsó. Fuzzy communities and the concept of bridgeness in complex networks. *Physical Review E*, 77:016107, 2008.

[39] M. E. J. Newman and J. Park. Why social networks are different from other types of networks. *Physical Review E*, 68:036122, 2003.

[40] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[41] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997.

[42] A. Rao and G. Lohse. Towards a texture naming system: Identifying relevant dimensions of texture. In *Proceedings of IEEE Conference on Visualization*, pages 220–227, 1993.

[43] N. H. Riche and T. Dwyer. Untangling Euler diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1090–1099, 2010.

[44] H. Shen, X. Cheng, K. Cai, and M.-B. Hu. Detect overlapping and hierarchical community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 388(8):1706–1712, 2009.

[45] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of Symposium on Visual Languages*, pages 336–343, 1996.

[46] P. Simonetto, D. Auber, and D. Archambault. Fully automatic visualisation of overlapping sets. *Computer Graphics Forum*, 28(3):967–974, 2009.

[47] J. Stehlé, N. Voirin, A. Barrat, C. Cattuto, L. Isella, J. Pinton, M. Quaggiotto, W. Van den Broeck, C. Rgis, B. Lina, and P. Vanhems. High-resolution measurements of face-to-face contact patterns in a primary school. *PLoS ONE*, 6(8):e23176, 2011.

[48] C. Tyler. *Human Symmetry Perception and Its Computational Analysis*. Psychology Press, 2002.

[49] F. van Ham and J. van Wijk. Interactive visualization of small world graphs. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 199–206, 2004.

[50] A. Viamontes Esquivel and M. Rosvall. Compression of flow can reveal overlapping-module organization in networks. *Physical Review X*, 1:021025, 2011.

[51] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. van Wijk, J.-D. Fekete, and D. Fellner. Visual analysis of large graphs: State-of-the-art and future research challenges. *Computer Graphics Forum*, 30(6):1719–1749, 2011.

[52] X. Wang and I. Miyamoto. Generating customized layouts. In *Proceedings of Graph Drawing*, volume 1027, pages 504–515, 1996.

[53] S. Zhang, R.-S. Wang, and X.-S. Zhang. Identification of overlapping community structure in complex networks using fuzzy-means clustering. *Physica A: Statistical Mechanics and its Applications*, 374(1):483–490, 2007.