Target Netgrams: An Annulus-constrained Stress Model for Radial Graph Visualization

Mingliang Xue, Yunhai Wang, Chang Han, Jian Zhang
Zheng Wang, Kaiyi Zhang, Christophe Hurter, Jian Zhao, and Oliver Deussen

Abstract—We present Target Netgrams as a visualization technique for radial layouts of graphs. Inspired by manually created target sociograms, we propose an annulus-constrained stress model that aims to position nodes onto the annuli between adjacent circles for indicating their radial hierarchy, while maintaining the network structure (clusters and neighborhoods) and improving readability as much as possible. This is achieved by having more space on the annuli than traditional layout techniques. By adapting stress majorization to this model, the layout is computed as a constrained least square optimization problem. Additional constraints (e.g., parent-child preservation, attribute-based clusters and structure-aware radii) are provided for exploring nodes, edges, and levels of interest. We demonstrate the effectiveness of our method through a comprehensive evaluation, a user study, and a case study.

Index Terms—Radial Visualization, Stress Model, Hierarchy Constraint, Graph

1 Introduction

Graphs are a powerful and universal data structure widely used for representing relationships between objects in many important real-world domains. Many graphs have complex topology and various node attributes. While a number of techniques have been proposed for this problem [4], radial graph visualizations [5] are still widely used for studying undirected sub-graphs with a specific central node and its relationship to neighboring nodes defined in terms of *levels*, expressing their minimal graph distances to the central node. For example, 1-level neighbours are neighbors directly connected to the central node. Such graphs are often not strictly hierarchical, since some of their nodes can be connected while being at the same level.

A popular method to visualize such sub-graphs are so-called radial drawings [2] that place the focal node at the canvas center and arrange its neighbours on concentric circles. Therefore, the method automatically preserves radial levels. However, such drawings have two major inherent limitations: i) they are often not able to show cluster structures due to missing or heavily distorted edge information on the circles (see the orange and green nodes in Fig. 1(a)), and ii) placing nodes on the circle's circumference often results in node overlap and multiple edge crossings (see the blue nodes in Fig. 1(a)), even when the placement is carefully refined by a parent-centric layout [6] or a stress-based method [3]. The major reason for these issues is that this method ignores edges between non-root nodes. Instead, the recently proposed flexible radial layout [3] reformulates the discrete radial level constraint into a continuous stress function and linearly combines it with a classical stress model [7] for all nodes, which allows to alleviate the first issue mentioned above. However, the method results in unevenly distributed nodes on the circumferences, creating even more overlaps (see the orange and green nodes in Fig. 1(b)).

The above issues motivated us to revisit the historic precursor of radial graph visualization, Target sociograms [8] (see Fig. 2), which portray social interactions within a group of people. Here, nodes are placed inside annuli corresponding to the radial levels, which provide more space for showing cluster structures than traditional radial visualizations. The relationship between nodes is depicted with lines, readable annotations for each node are provided. While several researchers have been inspired by this work and proposed radial visualization techniques [9], [10], to our knowledge no method exists for the automatic generation of Target sociograms with a readability matching the original illustration. Hogan et al. [11] provide an interactive tool for users to manually generate target sociogram-like visualizations, but no automatic method.

The design of our technique is grounded on the guidelines given by professional illustrators and visualization researchers. By relaxing the constraint responsible for a uniform node distribution along the circles, we propose an annulus-constrained stress model that places nodes on the annuli between adjacent circles for indicating their hierarchy level while maintaining the network structure as much as possible. Fig. 1(c) shows an example, where nodes of the same levels are arranged into the same annuli, while the blue, green and yellow clusters formed by non-root nodes are clearly displayed. In doing so, the space for placing nodes is larger than for traditional radial layouts, ensuring a better

- *Mingliang Xue, Yunhai Wang, Chang Han and Kaiyi Zhang are with the Department of Computer Science, Shandong University, China. E-mail: {xml95007,cloudseawang, hatch.on27, zhang.kaiyi42}@gmail.com*
- *Jian Zhang is with Computer Network Information Center, Chinese Academy of Sciences, Beijing, China. E-mail: zhangjian@sccas.cn*
- *Zheng Wang is with China information consulting & designing institute, CO., LTD (CITC), Beijing, China. E-mail: wangzheng_ai@126.com*
- *Christophe Hurter is with ENAC, Ecole National de l'Aviation Civile, Toulouse, France. E-mail: christophe.hurter@enac.fr*
- *Jian Zhao is with the Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada. E-mail: jianzhao@uwaterloo.ca*
- *Oliver Deussen is with Computer and Information Science, University of Konstanz, Konstanz, Germany. E-mail: oliver.deussen@uni-konstanz.de*
- *Yunhai Wang is the corresponding author*

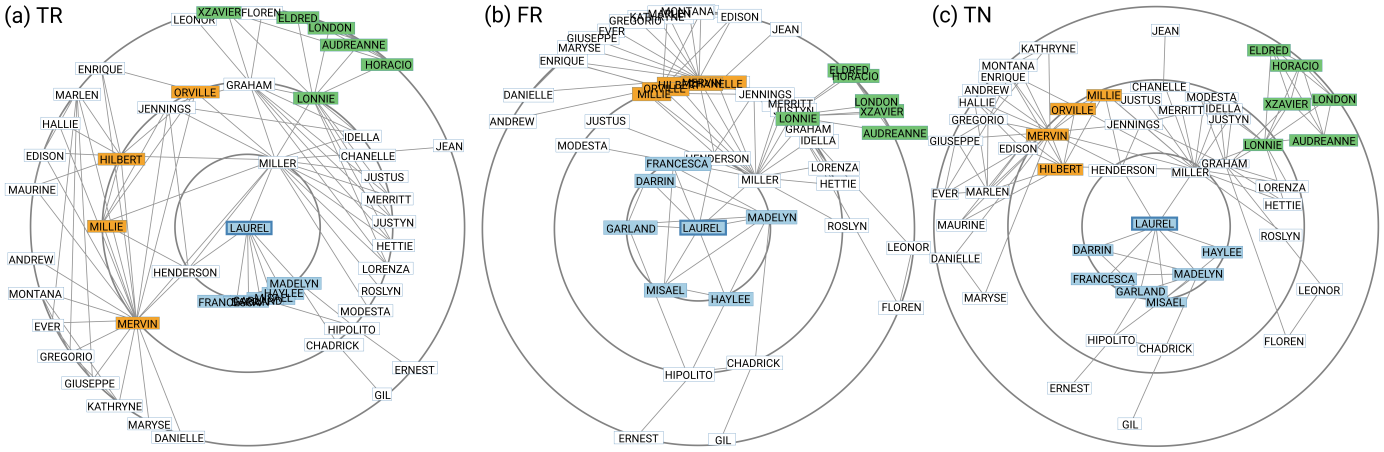


Fig. 1. Visualization of a graph of researchers selected from co-authorships in Network Science [1] using different layout methods: (a) traditional radial layout (TR) [2], (b) more flexible radial layout (FR) [3], and (c) our Target Netgram, (TN). TR fails to show the clusters (green and orange nodes) and results in heavy visual clutter (blue nodes), while FR does preserve these two clusters to some extent but induces heavy node overlap so that the relationship between nodes cannot be clearly discerned. In contrast, our method is able to show cluster structures as well as accurately place nodes into the corresponding annuli.

graph readability. To adapt stress majorization to this model, we integrate these annulus constraints into a state-of-the-art constrained least square optimization [12]. Doing so allows us to solve models efficiently even for larger graphs, while ensuring real-time performance for small and medium-sized graphs up to 1000 nodes.

Furthermore, our model allows users to explore complex network structures with three new constraints: i) *parent-child preservation*: corresponding edges should not be too long to prevent misunderstanding; ii) *attribute-based clusters*: manually specified nodes or elements with similar attributes are grouped together; and iii) *structure-aware radii*: circle radii are adjusted non-uniformly in terms of the number of nodes contained by each radial level or the size of the attribute-based glyphs attached to each node. In addition, users can change the central node. The visualization is then transformed into a new layout with an animated transition that results in only a few edge crossings.

To demonstrate the effectiveness of our method, we conducted three evaluations. First, we compared it to existing techniques for preserving graph structures and readability by using different quantitative measures. Second, with a user study, we investigated if users are able to accurately perceive radial distances from our results and compared it with existing radial layout methods in supporting graph exploration. In addition, we conducted a case study to show the applicability of our method for interactive exploration.

In summary, our main contributions are as follows:

- We propose an annulus-constrained stress model that aims to satisfy radial level constraints while depicting graph clusters, neighborhood structures and reducing node-node/node-edge overlaps as much as possible;
- We provide various constraints and rich interaction methods for exploring nodes, edges, and levels of interest;
- We demonstrate the effectiveness of our method through a quantitative evaluation, a user study, and a case study.

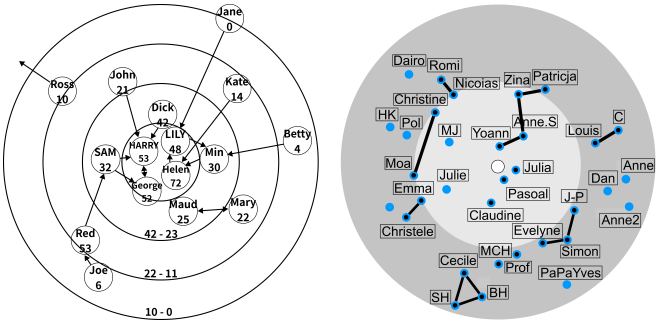


Fig. 2. Two examples of target sociograms, where the neighbors of the focal node are placed in the corresponding annuli according to their levels, while each node contains label information. (a) Our manually reproduced target sociogram introduced by Northway [8]; (b) the sociogram manually created using the interactive tool provided by Hogan et al. [11].

2 Related Work

2.1 Radial Graph Layouts

Radial graph visualizations aim at layouting a graph around a focal area. The most common representation are root-centric radial tree diagrams, where graphs are simplified as a tree rooted in the focal node. After placing the root at the center, classical radial drawings [2], [13] arrange the i-level neighbors at the ith concentric circle around the center and determine the angular width of a node's subtree by the number of its leaf nodes. Yee et al. [14] extend this method to allow nodes of varying sizes and provide smooth animated transitions for switching to a different focus. Jankun-Kelly and Ma [15] introduce radial focus+context layouts with non-uniform annulus widths to assist visual exploration. Our Target Netgrams aim to retain the advantages of all these methods while allowing to reveal graph clusters and neighborhood as much as possible by carefully arranging nodes inside the respective annuli.

The above-mentioned root-centric layout ensures that nodes of the same level are on the same circle. This might obscure interesting aspects, e.g., symmetries in the tree since nodes with a large distance in the graph may be positioned close-by. To address this issue, Pavlo et al. [6] propose

a parent-centric layout that places every subtree around its own sub-root. Huang et al. [16] suggest several angle assignment rules for child nodes to produce a layout with a more uniform node distribution and fewer edge crossings. Such a layout can effectively display self-similar structures of tree branches, but its space utilization is poor. Thus, the root-centric layout is still the most popular method.

One major problem of tree-based representations is that edges, which are not part of the tree are ignored so that the underlying graph structures cannot be accurently represented. Wills [13] suggests refining root-centric radial layouts with an incremental force-based placement of all edges. However, such a placement might not be able to preserve the tree structure. Bachmaier et al. [17] enhance the hierarchical layout algorithm by Sugiyama [18] with curved edge routing to generate radial tree drawings, but it is only able to handle networks with bi-connected components. Brandes et al. [10] introduce a three-phase force-directed model for determining node positions on the circumference of circles while minimizing edge crossing and overlap between nodes and edges. This method is computationally demanding such that only small networks can be explored. Brandes and Pich [3] formulate the radial level constraint as a continuous distance-based constraint and combine it with a stress model [7]. Such an energy model allows to simultaneously preserve graph distances and radial constraints as much as possible so that clusters might be visible. However, it often distributes nodes on the circular lines unevenly, resulting in stronger overlap. In contrast, our annulus-based stress model places the i-level node in the range of the i^{th} annulus, providing more space for revealing structures with less overlap. To improve readability, our model determines the annulus width in terms of node information and uniformly distributes the nodes across the corresponding annulus.

Our model is similar to Dig-Cola [19], which places nodes in a directed graph between two adjacent hierarchical levels for better showing the structure. However, in contrast to them, Target Netgram works for a radial visualizations, considers readability constraints and provides more interactions.

2.2 Radial Visualization Methods

Radial representations play an essential role in visualization as a distinct design metaphor. They originate from the 19th century when pioneers such as William Playfair [20] and Florence Nightingale [21] used pie and rose charts for presenting percentages. In the 1990s, the term radial visualization was introduced by Hoffman et al. [22] for a distinct visualization method. Due to aesthetics and space efficiency, it is commonly used for visualizing four kinds of data: hierarchical structures [23], relationships between entities [24], ranking of search results [25], serial periodic data [26] as well as multi-dimensional data [27]. Our special interest lies in their inherent intuitiveness for visualizing the relationship in graphs.

The most similar radial representation to a Target Netgram, is the above-mentioned *target sociogram* introduced by Northway [8] for showing interpersonal relationships in a group of people. It places nodes inside concentric circles whose radii increase, while the status level of the nodes decreases, with the most important nodes in the center. Later, Northway discussed an interactive version of target sociograms [28] where nodes represented by pegs can be moved and relationships represented by rubber bands can be stretched. To mimic Northway's target sociograms, Brandes et al. [10] proposed an energy-based placement method with carefully designed forces to satisfy aesthetic and readability criteria. However, the resulting visualizations place nodes on the circumference instead of annuli, making it hard to visualize large numbers of nodes. Our method addresses this issue and provides more flexible interactions.

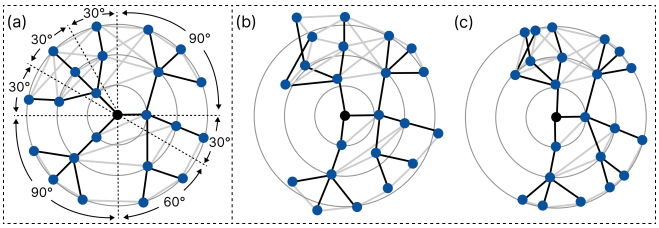


Fig. 3. Classical radial drawing and more flexible radial layout: (a) the result of radial drawing, where the nodes in each layer divide the circle (360 degrees) into sectors, with each node corresponding to one sector; (b) results generated by only using the stress model, which reveals two clusters but cannot place the nodes on the circles; (c) results generated by the more flexible layout, which shows clusters and places the nodes on the circles.

A few empirical evaluation have been conducted to compare radial representations with their Cartesian counterparts and to study their effectiveness. It is shown that the radial representation performs worse in terms of accuracy and completion time for certain tasks [29]–[32]. For example, traditional and orthogonal tree layouts significantly outperform radial tree layouts for some tasks (e.g., finding the least common ancestor) with respect to completion time [33], [34]. For other tasks, radial representations have their strengths. Diehl et al. [35] found that radial visualizations seem to be more effective for focusing on a particular dimension (e.g., persons). Albo et al. [36] compared three radial visualizations for multi-dimensional data and found that Radar charts were the least effective, while the performance of Flower Charts and Circle Charts depend on the task. Likewise, we evaluated two radial layouts for visualizing graphs: a root-centric radial layout [2] and a stress-based radial layout [3] in comparison to our method.

3 Target Netgrams

The primary goal of this work is to automatically produce radial layouts for a better support of graph analysis. Before elaborating our method and interaction techniques, we first briefly describe two most related methods and then the design requirements we followed.

3.1 Most Related Methods and Limitations

Radial Drawing. This method places the focal node at the canvas center and then arranges all nodes onto concentric circles around the focus. For each node, the corresponding circle is determined by its level, first level nodes lie on the smallest inner circle. The angular position of a node on its circle is determined by the sector whose angle is proportional to the number of leaf nodes in the subtree rooted at that node

(see Fig.3 (a)). This method is simple but it does not consider the edges between nodes, resulting in the fact that cluster and neighbor structures of the graph are lost (see Fig. 1(a)).

More Flexible Radial Layout. Alternatively, Brandes and Pich [3] proposed to solve the radial layout by adapting a stress model [7], which minimizes the sum of the squared distance differences between all pairs of graph nodes. It models the radial level constraint as a distance constraint between each node to the focus node (for short constraint stress), and combines it with the original stress model. Hence, it obtains the positions of all nodes $\mathbf{X}$ by minimizing the following objective:

$$\arg\min_{\mathbf{X}} \sum_{i \neq c} \frac{\lambda}{d_{ic}^2}(||\mathbf{x}_i - \mathbf{x}_c|| - d_{ic})^2 + \sum_{j<i} \frac{1-\lambda}{d_{ij}^2}(||\mathbf{x}_i - \mathbf{x}_j|| - d_{ij})^2, \quad (1)$$

where $\mathbf{x}_c$ is the coordinate of the central node c, d_{ic} and d_{ij} are the shortest path distances between two nodes i and c or i and j, respectively. For a pre-defined number of iterations nI, the weight $\lambda(t)$ at the tth iteration is

$$\lambda(t) = \frac{t}{nI}.$$

In other words, the stress model plays a major role at the early iterations but its influence gradually decreases and finally only the constraint stress has an influence. In doing so, it is ensured that nodes are placed on the circles.

Figs.3(b,c) compare the results generated by only using the stress model and the extended model. It shows that this method better reveals cluster structures and maintains hierarchy constraints. However, this method might lose the cluster structures and also suffer from severe visual clutter for medium-sized graphs because of the limited space on the circles (see Fig. 1(b)).

3.2 Design Requirements

Based on previous investigations [37]–[39], one of the typical graph analysis tasks is to investigate the topology around a central node, which depicts the hierarchal structure between the center and its neighbors, graph clusters formed by neighbors, and node attributes. Accordingly, the visualization to be designed should clearly reveal this information as much as possible, while providing rich interactions for an effective exploration. Namely, there are three design requirements:

- **DR1**: Satisfy the radial level constraint to facilitate investigating graphs based on a central node;
- **DR2**: Maintain cluster and neighborhood structures to allow for easy exploration of local graph topology; and
- **DR3**: Distribute the nodes evenly within the concentric circles to reduce clutter in visualization.

Although positioning nodes onto the annuli between adjacent circles rather than on the actual circles does not strictly satisfy DR1, we speculate that it does not influence the perception of the radially placed hierarchical structures. In DR2, the cluster and neighborhood structures are maintained by preserving graph-theoretic distances between all nodes, while DR3 helps for reducing node overlap and edge crossings.

The key challenge in meeting these design requirements are the conflicts and trade-offs between them. The classical radial drawing [14], [15] satisfies DR1 by arranging nodes with varying sizes on circles. However, uniformly placing nodes along the circular lines obscures 2D network structures (see the orange and green clusters in Fig. 1(a)). In contrast, flexible radial layouts are based on stress models and non-uniformly place nodes on the circular lines so as to meet DR1 and DR2. However, their placements are likely to violate DR3 since they result in fragmentation of clusters (see the green cluster in Fig. 1(b)). Moreover, both methods suffer from significant node overlap, see the blue cluster in Fig. 1(a) and the orange cluster in Fig. 1(b). The major reason is that they are constrained by the limited space on the circular lines. To address this issue, we follow the design practices for Target sociograms that position nodes onto the annuli between adjacent circles (see the example in Fig. 1(c)) . This strategy is consistent with the Gestalt principle of *Proximity* [40] saying that nodes within an annulus between any two consecutive concentric circles are considered as a group with the same radial level. In doing so, all requirements can be satisfied.

3.3 Annulus-constrained Stress Model

Based on the three requirements described above, we introduce an objective function consisted of three terms: i) the level constraint meets DR1 by placing nodes with level k on the k^{th} annulus; ii) the stress term satisfies DR2 by minimizing the stress error between all pairs of nodes to ensure a good representation of global structures; and iii) the angular term (DR3) requires that the nodes are uniformly placed within annuli. Although the level constraint and the angular term both can be modeled in polar coordinates, the structure term involves the Euclidean distances among all pairs of nodes positioned in different concentric annuli and thus we still model the whole objective with Euclidean distances.

Accordingly, we formulate our radial layout as an optimization problem to solve for the positions of all nodes $\mathbf{X}$ using the following two-fold objective and a hard constraint:

$$\begin{aligned} \arg\min_{X} & \sum_{k=1}^{l} \sum_{i \in \Omega_k} \omega_{k,i}^a ||\mathbf{x}_{k,i} - \mathbf{x}_c - R_k e_{k,i}||^2 \\ & + \sum\nolimits_{i,j \in \Omega} \omega_{i,j}^s (||\mathbf{x}_i - \mathbf{x}_j|| - d_{ij})^2 \\ & s.t.\ ||R_{k-1}||^2 \leq ||\mathbf{x}_c - \mathbf{x}_{k,i}||^2 \leq ||R_k||^2, i \in \Omega_k, \end{aligned} \quad (2)$$

where $\mathbf{x}_c$ is the coordinate of the central node, $e_{k,i}$ is an ideal orientation initialized by the classical radial layout algorithm [14], $\omega_{k,i}^a$ and $\omega_{i,j}^s$ are the weights for the angular and stress terms in a range of [0,1], l is the level of the input graph, Ω_k is the set of nodes on the k^{th} level with the number n_k, and Ω is the set of all nodes in every set Ω_k, d_{ij} is the target shortest path distance between two nodes i and j, and R_k the radius of the k^{th} circle. By default, the radius of the k^{th} circle is $R_k = k\gamma$ and $\mathbf{x}_c$ is the canvas center. Since the ordering of nodes might change during the iterative process, we re-initialize $e_{k,i}$ based on the obtained $\mathbf{X}$ at each iteration.

The first term in Eq. 2 can also be treated as the stress term between all neighbors and the central node, which

can be solved in a quadratic way. Based on the stress majorization [41], [42], we combine it with the second term (the stress constraint between alters) as the vector form:

$$\arg\min_{X} \sum\nolimits_{i,j\in\Omega\cup\{c\}} \omega_{i,j}||\mathbf{x}_i - \mathbf{x}_j - \mathbf{d}_{ij}||^2 \quad (3)$$
$$s.t.\ ||R_{k-1}||^2 \leq ||\mathbf{x}_c - \mathbf{x}_{k,t}||^2 \leq ||R_k||^2, t \in \Omega_k,$$

where $\omega_{i,j}$ is the weight for the corresponding edge defined as $\mathbf{d}_{ij}^{-2}$, and $\mathbf{d}_{ij}$ is pre-defined for the edges targeted to the central node but is undefined for the other edges. We call this model a *annulus-constrained stress model*.

Differences with the more flexible radial layout (FR). Compared to the extended stress model of FR (see Eq. 1), there are three major differences. First, the annuli-based hard constraint provides more space for node placement but does not influence the perception of hierarchical structure (see Section 4.2). Second, we represent the constraint stress in Eq. 1 as the angular term, which further requires the nodes to be uniformly placed within annuli. Last, all three terms always have the equal influence on the final result, rather than determining the layout with one term.

3.4 Solving the Constrained Optimization

After writing Eq. 3 in matrix form and differentiating it with respect to $\mathbf{X}$, we set the derivative to zero, which yields us:

$$\mathbf{LX} = \mathbf{JD}, \quad (4)$$

where $\mathbf{L}$ is an $n \times n$ weighted Laplacian matrix, $\mathbf{J}$ is an $n \times s$ ($s = n(n-1)/2$) matrix with each column storing the weight of an edge vector, and $\mathbf{D}$ is an $s \times 2$ matrix consisting of all pairwise edge vectors. Following the stress majorization method by Wang et al. [42], Eq. 4 can be solved by alternating between finding the optimal edge vector orientations $\mathbf{D}$ (O-step) and searching for node positions $\mathbf{X}$ (P-step). However, there are two major differences: i) the method has quadratic inequality constraints and thus the P-step cannot be solved by a simple least-squares approach; and ii) the edge vectors targeted to the central node need to be re-initialized at the O-step by first projecting the nodes in Ω_l to the corresponding circle to obtain an appropriate ordering and then determining edge orientations with the classical radial layout algorithm [2] from level l to 1.

Fig. 4 illustrates one iteration of applying the O- and P-step to an initial layout (see Fig. 4(a)). Here, each node P_i at the second level is projected to a position on the second circumference (see Fig. 4(b)), then all projected positions are adjusted by the radial layout algorithm (Fig. 4(c)), and finally a new evenly distributed layout is obtained by using all new edge vectors $e_{2,i}$ (Fig. 4(d)).

The customized P-step is as follows: with fixed $\mathbf{D}$, solving $\mathbf{X}$ is a constrained least-squares problem [43]. However, existing solvers [12], [44] can only handle box constraints or upper bound quadratic constraints, whereas we want to constrain the solution to an annulus. Hence, we adapted the state-of-the-art constrained least squares solver of Mead and Renaut [12] to our problem. The result is shown in Algorithm 1, which consists of two major parts for handling upper and lower bounds.

Lower Bound. For a node $\mathbf{x}_{i,k}$ with a distance to $\mathbf{x}_c$ less than the lower bound R_{k-1}, the stress constraint ($||\mathbf{x}_{i,k} - \mathbf{x}_c|| - R_k)^2$ is not satisfied. Thus, we increase the corresponding weight $\omega_{c,i}$ and $\omega_{i,c}$ to fit the lower bound (see line 11). To guarantee that the local optimum is close to the lower bound, we set ϵ to 0.01.

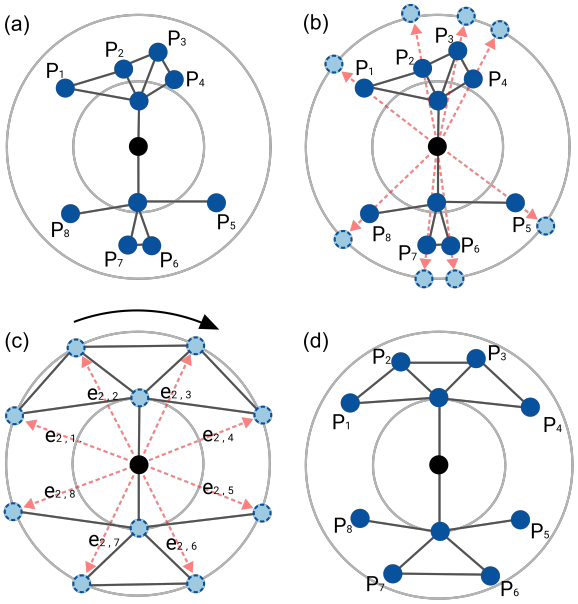


Fig. 4. Process of determining edge vectors from the central node to its neighboring nodes during each iteration of our optimization: (a) result of the previous iteration is used as initial layout; (b) projecting the outside nodes onto the corresponding circle. (c) Edge vectors are determined by the projection order of the nodes and a classical radial layout. Vectors of inner nodes are determined by their outside neighbor nodes according to the radial layout; (d) result after one iteration.

Algorithm 1 P-step using Constrained Least Squares

1: initialization:$\bar{\mathbf{X}} = [\mathbf{x}_c, ..., \mathbf{x}_c]^T$, $\mathbf{C}^{-1} = diag(1/R_i^2)$, $\beta = 1, \epsilon = 0.01$
2: iter = 0
3: $\mathbf{Y} = \mathbf{X} - \bar{\mathbf{X}}$
4: **while** Nodes beyond the boundary exist **do**
5: iter += 1
6: Solve $(\mathbf{L}^T\mathbf{L} + \mathbf{C}^{-1})\mathbf{Y} = \mathbf{L}^T(\mathbf{JD} - \mathbf{L}\bar{\mathbf{X}})$ for $\mathbf{Y}$
7: $\beta = 1/(1 + iter/10)\beta$
8: for k from 1 to l:
9: for i in Ω_k:
10: if $||\boldsymbol{y}_i||^2 > R_k^2$: $(\mathbf{C}^{-1})_{ii} = (\beta\mathbf{C})_{ii}^{-1}$
11: else if $||\boldsymbol{y}_i||^2 < R_{k-1}^2$: $\omega_{i,c} = (1+\epsilon)\omega_{i,c}, \omega_{c,i} = (1+\epsilon)\omega_{c,i}$
12: **end while**
13: $\mathbf{X} = \bar{\mathbf{X}} + \mathbf{Y}$

Upper Bound. By treating the upper bound constraint as a penalty term weighted by R_k, we re-write Eq. 4 as:

$$\min ||\mathbf{LX} - \mathbf{JD}||^2 + ||\mathbf{C}^{-1/2}(\mathbf{X} - \bar{\mathbf{X}})||^2 \quad (5)$$

where $\mathbf{C} = \text{diag}(\beta R_i^2)$, $R_i = R_k$ for $i \in \Omega_k$. $\bar{\mathbf{X}}$ is an array with n duplicated central node positions $\mathbf{x}_c$. The parameter β starts with 1 and decreases with each iteration. Following the original constrained least square approach [12], we update β by $1/(1 + iter/10)\beta$ (see Line 7). If $\mathbf{x}_i$ is beyond the upper bound, $(\mathbf{C}^{-1})_{ii}$ will be increased (see line 10), pushing $\mathbf{x}_i$ towards $\mathbf{x}_c$ at each iteration. By doing so, the upper bound is met after a few iterations.

For easily checking whether $\mathbf{x}_i$ is positioned in the corresponding annulus, we write $\mathbf{Y}$ as $\mathbf{X} - \bar{\mathbf{X}}$ and set the derivative of Eq. 5 with respect to $\mathbf{Y}$ to zero:

$$(\mathbf{L}^T\mathbf{L} + \mathbf{C}^{-1})\mathbf{Y} = \mathbf{L}^T(\mathbf{JD} - \mathbf{L}\bar{\mathbf{X}}), \quad (6)$$

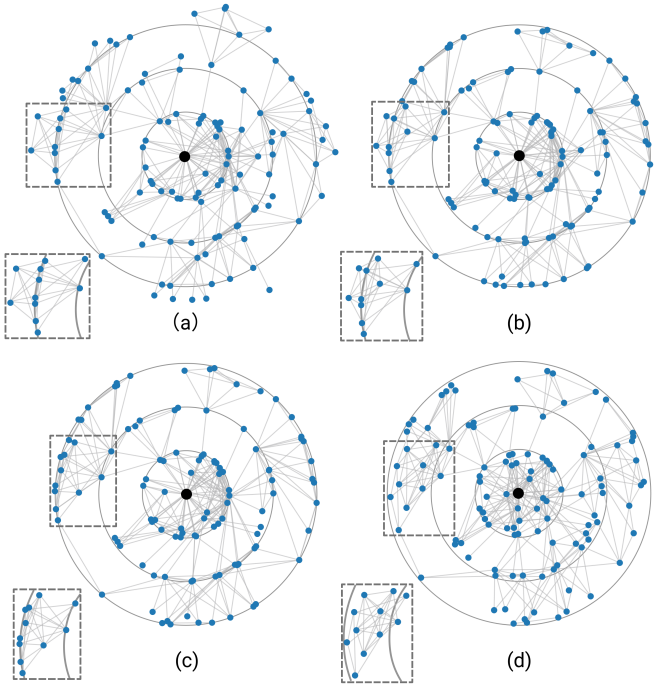


Fig. 5. Illustration of convergence. Result after (a) 1st iteration; (b) 10 iterations; (c) 20 iterations; and (d) 30 iterations.

which is a linear system (see line 6).

The most time-consuming part of our algorithm is iteratively solving Eq. 6. We compared different solvers [45] such as gradient descent-based methods, conjugate gradient, and Cholesky decomposition and found that the stochastic gradient descent solver [46] is the most effective in our experiments. Fig. 5 shows the convergence by showing intermediate optimization results. Here, the result after the 30th iteration almost satisfies all constraints that show the cluster structures and place nodes in the corresponding annuli.

Noted that Algorithm 1 cannot ensure feasible solutions for the constrained optimization in all cases (see Section 4.3). We address this issue by simply moving the node $\mathbf{x}_k$ to the kth circle if $||\mathbf{x}_k - \mathbf{x}_c|| > R_k$ holds; and to the $(k-1)$th circle if $||\mathbf{x}_k - \mathbf{x}_c|| < R_{k-1}$ holds.

3.5 Structural Constraints

During the interactive exploration, users may want to emphasize some structures at some levels. For better supporting such exploration tasks, we implemented three new constraints based on typically wanted graph characteristics: parent-child preservation, attribute-based clusters, and structure-aware radii. In addiction, we provide smooth animations for users to switch the central nodes of interest.

Parent-Child Preservation. This constraint aims at avoiding long parent-child edges, which might let some nodes wrongly be seen as children of neighboring parent nodes (e.g., node P in Fig. 6(a)). The constraint is imposed by finding new edge vectors for such edges in the O-step. Specifically, we first detect these edges from the previous layout $\mathbf{X}^t$ by checking whether the angle between the two nodes of the edge and the central node is smaller than a threshold ($\pi/2$). In this case, we extend the edge vector to intersect with the corresponding

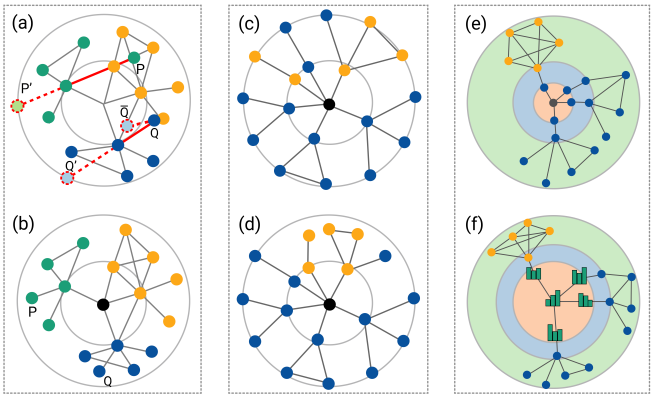


Fig. 6. Illustration of our three constraints: parent-child preservation (a,b), cluster formation (c,d), and structure-aware radii adjustment (e,f).

circle and take the line segment in the other direction as the new edge vector (see the node P′ in Fig. 6(a)). Having most long edges removed this way, we compute the new node position with the P-step (Fig. 6(b)).

Although various other existing constraints, such as node non-overlap and minimizing edge crossings [42], [47], can be applied, they might end up in meaningless positions without considering the hierarchical structure. To address this issue, we check whether the positions adjusted by applying these constraints fit with the hierarchical structures. If not, we use the position adjusted by the above parent-child preservation constraint. Node Q in Figs. 6(a,b) is an example, where the position $\bar{\text{Q}}$ adjusted by the node non-overlap constraint is located at the first circle. Thus, we use the position Q′ to define the new edge vector.

Attribute-based Clusters. In the final layout, users may like to group some nodes together in terms of node attributes. To achieve this goal, we introduce an additional term to Eq. 3:

$$\sum\nolimits_{i,j\in\Theta} \omega_{i,j}||\mathbf{x}_i - \mathbf{x}_j||^2 \tag{7}$$

where Θ is the set of nodes to be grouped together and $\omega_{i,j}$ is 1.0 by default. Figs. 6(c,d) show how this constraint forms a small cluster of orange nodes.

Structure-aware Radii. We provide flexible interaction methods for exploring graph structures. First, the annulus width can be altered by requiring that the area of each annulus is proportional to the contained number of nodes. Accordingly, the circle radius can be defined as:

$$R_1 = R_l\sqrt{n_1/n} \quad , R_k = R_{k-1}\sqrt{\sum_1^k n_i / \sum_1^{k-1} n_i}, \tag{8}$$

where R_l is the radius of the maximal level l.

Users can further magnify the radii of interesting levels for attaching attribute-based statistical (or other) charts to nodes. Fig. 6 shows an example, where the third annulus with the most nodes in (e) is extended to better show the orange cluster and the first annulus in (f) is further enlarged to place the bar charts associated to each node.

Smooth Animation. When a new central node is selected, we provide a smooth animation for switching from the current layout to the next. To reduce excessive edge crossings, we extend the radial layout-based animated transition by Yee

et al. [14] in two aspects. First, we use the same animation speed for all nodes, helping users track individual nodes easily. Second, we separate the animation into stages: one for moving edges with clockwise rotation and another for counter-clockwise rotation. For the effectiveness of such an animation, please see our supplemental video.

3.6 Placement on Annuli vs. Circles

Our stress model positions all nodes onto the 2D annuli between adjacent circles instead of on the circles itself as done by traditional radial drawings [2]. Here we analyze the maximal numbers of non-overlapping nodes that can be placed by these two methods. Suppose each node is shown as a circle with radius r, the maximal number of non-overlapping nodes that can be placed on the kth annulus by a Target Netgram, is:

$$N_k = \frac{\pi R_k^2 - \pi R_{k-1}^2}{\pi r^2}. \tag{9}$$

Likewise, the maximal number of non-overlapping nodes placed on the kth circular line by the traditional radial drawings is:

$$M_k = \frac{2\pi R_k}{2r}. \tag{10}$$

where the occupied arc-length by each node is approximated by its diameter. Substituting the default radius $R_k = k\gamma$ to Eq. 9 and Eq. 10, the ratio of the maximal non-overlapping nodes between these two methods is:

$$\rho_k = \frac{(2k-1)\gamma}{k\pi r} = \frac{(2k-1)}{k\pi}\rho \tag{11}$$

where ρ is γ/r. Since γ is often much larger than r, our method theoretically creates less node/edge-node overlaps for placing a given input graph. Note that this derivation is based on the assumption that all nodes are uniformly distributed, which often is not satisfied due to the stress term in Eq. 2.

4 Evaluation

We implemented our technique in C++ and drew the layout with OpenGL. We tested it on a computer with a 2.60GHz Intel Core i7 processor and 16GB RAM, running on Windows 10 (64 bit). To confirm that our approach satisfies the three design goals, we performed two comparisons by using: (i) numerical measures (see Section 4.1) and (ii) user studies (see Section 4.2) as well as a case study with real-world datasets (see Section 4.3). The comparisons include three state-of-the-art methods: traditional radial layout (TR) [2] and flexible radial layout (FR) [3] and the classical stress model (SM) [7]. Although SM is not designed for radial visualizations, it performs well in preserving structures of general graphs [48]. All these methods are implemented using the same configurations as our Target Netgrams (TN).

4.1 Quantitative Comparison

Here we compare results for radial graph visualizations produced from various methods in terms of structure preservation, graph readability and runtime performance.

TABLE 1
Characteristics of the used datasets including average number of nodes (NN), average node degree (ND), and average running time (second) of different algorithms.

	ca-HepTh	Vispubdata	ca-GrQc	LastFM
NN	272.5	260	472.5	445.5
ND	2.57	3.18	4.50	4.39
TN	1.4	1.2	2.3	2.4
SM	1.3	1.1	1.6	1.6
TR	0.01	0.01	0.01	0.01
FR	5.3	4.1	14.3	14.7

Datasets. To construct our experimental data, we extracted nodes from real-world datasets as the central nodes to form subgraphs. Each subgraph consists of a random range of neighboring nodes of the central node. Because of many edges between all non-root nodes, such graphs are not strictly hierarchical. We used four real-world datasets: ca-Hepth [49], Vispubdata [50], ca-GrQC [49], and LastFM [51]. For each dataset, we randomly selected 100 central nodes with a random neighborhood radius between 3 and 7; that is, 100 subgraphs were formed as the inputs to these techniques. Table 1 shows the characteristics of these subgraphs.

Measures. To assess the structural preservation of each method, we employed three layout quality metrics: average stress, average radial stress, and neighborhood preservation degree [52].

- *Average stress* [41] is a normalized sum of the squared differences between pairwise visual distances in the layout and their target graph distances in data space, where a smaller value means better distance preservation.
- *Average radial stress* is the average stress between all nodes and the central node.
- *Neighborhood preservation degree* is the normalized Jaccard similarity [53] between the input graph and its shape graph, which is a k-nearest neighborhood graph constructed from the placed nodes in the layout.

To assess the graph readability, we measure the quality of each layout with three measures: node non-overlap degree, node-to-edge distances and crosslessness [54]:

- *Node non-overlap degree* is the ratio between the number of non-overlap nodes and the number of all nodes;
- *Node-to-edge distances* computes the average distance from each node to its nearest edge; and
- *Crosslessness* is the ratio between the number of edge pairs without crossing and the number of all edge pairs.

All of them are normalized to [0,1], and large values indicate better readability.

4.1.1 Result Analysis

Screen shots of the results generated by all four layout methods on each dataset and the according scores for each measure can be found in the supplemental material.

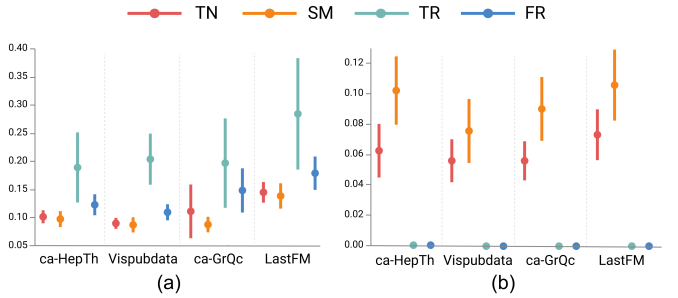


Fig. 7. Comparison of the preservation of distances using the average stress (a) and the average radial stress (b), where a smaller stress value is better.

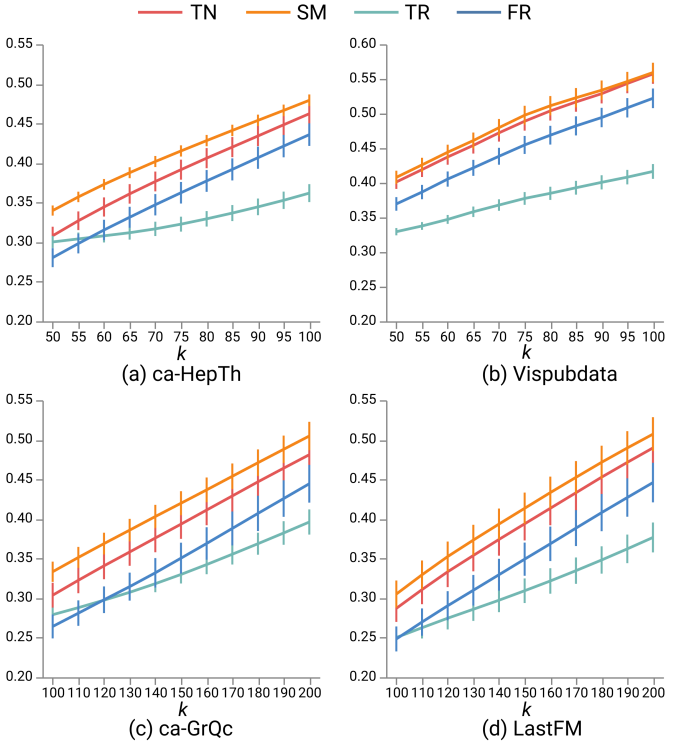


Fig. 8. Comparison of the preservation of neighborhood structures using varying neighborhood sizes k on four datasets. A higher value indicates better neighborhood preservation.

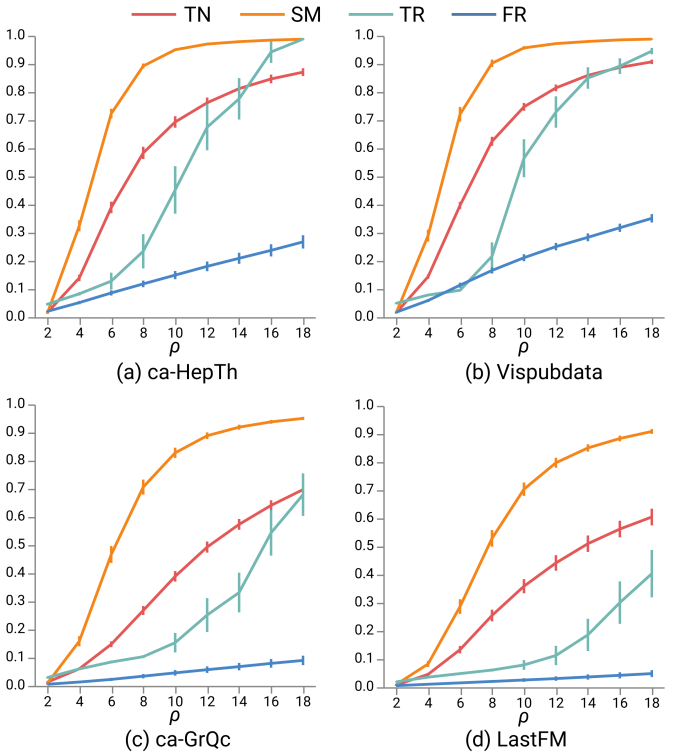


Fig. 9. Comparison of the graph readability using the node non-overlap degree with varying ρ defined as the ratio between unit annulus width and node size on four datasets. A higher value indicates less node overlap.

Results of Structure Preservation. Figs. 7 and 8 summarize the results of our quantitative comparison in structure preservation.

Fig. 7(a) shows the average stress of each method on the four datasets, where TR is the worst and TN is superior to FR but slightly worse than SM. This is plausible, since TR does not consider the graph distances, while the others all are the stress-based methods. Compared to SM, TN and FR limit node placements on annuli and circular lines, respectively. Since the annuli provide more space, TN can better preserve graph distances than FR. In addition, TN results in similar average stress values on the four datasets. This indicates that Target Netgrams perform consistently well across complex data sets.

Fig. 7(b) shows the average radial stress of the methods, where TN and TR perfectly preserve the radial structures for all nodes. In contrast, SM is the worst, since it does not take the radial constraint into account. By placing nodes into the annuli instead of circular lines, our method has a good trade-off in satisfying distance and radial constraints.

Fig. 8 shows the neighborhood preservation of the methods with varying k. To learn how well structures of different scale can be preserved, we set k to a range from 1/4 to 1/2 of the node number in each graph. On all four datasets, TN performs better than FR in maintaining the neighbor structure, but slightly worse than SM, while TR is the worst. The similar performance of TN and SM is plausible here, since both ensure that nodes are placed for preserving graph distances. While the stress term in FR helps for a better neighborhood preservation than TR, its performance is still constrained by the limit space on the circles in contrast to the one provided by the annuli for TN. Regarding the poor performance of TR, it ensures that nodes are uniformly placed within their corresponding circles without considering the graph neighborhood structures.

In summary, Target Netgrams perform better than most of the baselines on these three performance metrics. Although stress and neighborhood preservation degrees of a Target Netgrams are slightly worse than that of SM, it is able to maintain the radial level constraints.

Results for Graph Readability. Fig. 8 shows a comparison of the node overlap for all layout methods on the four datasets with varying ρ, the ratio between unit annulus width γ and node size r defined in Eq. 11. No matter what value of ρ, FR performs the worst. As the increase of ρ, the node size becomes smaller and the node none-overlap degrees of all methods quickly increase.

When ρ is not large, SM performs the best and TN ranks second and both are much better than TR. For reasonable large node sizes (ρ being 10-18), our TN yields good readability, especially for small datasets. Due to the annulus constraint, TN is inferior than SM for large datasets (ca-GrQC and LastFM) but is better than FR and TR. This indicates

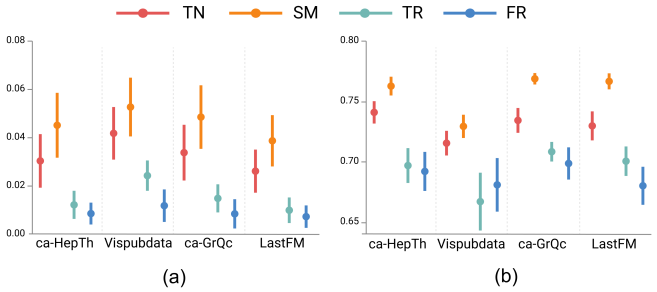


Fig. 10. Comparison of the graph readability using average node-to-edge distance (a) and crosslessness (b). For both measures, a large value indicates better graph readability.

that our TN ensures a uniform distribution of nodes while successfully using the stress model to keep distances between neighbor nodes in each annulus.

With increasing ρ, the non-overlap degrees of all methods increase. When ρ is larger than a specific value, TR performs similarly and even better than TN on the given datasets. This is reasonable, since TR uniformly positions nodes on circles, but the stress model-based methods work in a non-uniform way. However, the node size is often pre-determined and too small sizes make the structures hardly discernible. Thus, TN performs better than TR for a large input graph with a reasonable ρ.

Fig. 10(a,b) show the scores of node-to-edge distances and crosslessness for each method, respectively. We can see that SM performs slightly better than TN, while TR and FR have similar scores but perform worse than TN on both measures. This is reasonable, since the layout space used FR and TR is smaller than TN and the one of TN is smaller than SM.

Compared to FR, TN further attempts to uniformly distribute nodes within annuli by the first term in Eq. 2, resulting in less edge crossing. The tested methods have similar performance on all three measures. In general, TN performs better than TR and FR, but worse than SM, whereas SM cannot preserve the radial level constraint as shown in Fig. 7(b). Similarly to SM and TN, FR is a stress-based method, yet its readability is the worst. We speculate the reason is that it arranges nodes on the circular lines, which has limited space for revealing graph structures.

Runtime. Table 1 shows the average runtime of each method on different datasets. The runtime of our method is reasonable, i.e., less or around 1.5s and 2.5s on average for small and large graphs, respectively. SM performs similarly to our method while being slightly faster on large datasets. In contrast, FR is much slower than TN and SM, due to its used gradient descent solver. TR exhibits a much shorter runtime, because it has only a linear time complexity. However, based on the previous results, its visualizations perform poorly in preserving distances and neighborhood structures and therefore do not maintain readability.

Taking all findings together, we conclude that Target Netgrams maintain a good balance between preserving graph structures and readability. With reasonable runtime, our method performs better than traditional drawings, especially in distance preservation and node non-overlap while being significantly better than flexible radial layouts in multiple aspects.

4.2 User study

We further evaluate Target Netgrams (TN) in terms of human perception by conducting a user study to investigate if TN enables users to effectively locate radial levels and facilitate visual exploration of the sub-graphs with central nodes.

4.2.1 *Experimental Design*

Since SM is not a radial-based method, we only investigated three layout techniques in our user study: TR, FR, and TN.

Experimental Task. To assess the effectiveness of the approaches, we adapted a task proposed by [55] for exploring subgraphs with central nodes: Given a central node, look at all nodes with the radial level k and find the one with the highest degree; k is randomly chosen from 2 to 4. Namely, this task consists of two sub-tasks: i) locating the circle or annulus with the radial level k and ii) identifying the node with the highest degree. In doing so, the capabilities of each method in supporting the exploration of radial level information and local graph structures both have to be examined. To make the exploration easier, participants can click on a node to highlight the connecting edges and indicates its node ID which is randomized in each trial. Participants completed their tasks by entering node IDs.

Hypotheses. We hypothesized that our approach would perform similarly in locating the radial level but generally be more effective than the benchmark methods on the whole task:

- **H1**: TN performs similar to the benchmark conditions (FR and TR) on the sub-task of locating the given radial level.
- **H2**: TN outperforms the benchmark conditions (FR and TR) on the whole task.

Data Generation. We chose the CiteSeer dataset [56] with 3324 nodes and 4731 edges, then randomly selected nodes as the central nodes and constructed a subgraph for each of them. The subgraphs were extracted by limiting the nodes with a maximum graph distance of 5 to the central node. We set this parameter empirically to ensure that participants could smoothly complete the task with a reasonable cognitive load. To test whether our method really works for medium-sized graphs, we selected five subgraphs with more than 512 nodes [55] and generated three layouts for each subgraph by three layout methods (see an example in Fig. 11), resulting in 15 trials for total. We randomly choose a k value for each subgraph and ensure that all three k values (2,3,4) have been tested. The average node number and degree of these five subgraphs are 664.4 and 2.26, respectively.

Participants and Apparatus. We recruited 22 computer science students and researchers (14 males and 8 females) from a local university, with an average age of 21.6. All had a normal or corrected-to-normal vision. All participants had knowledge about basic graph concepts, such as shortest path, node degree, etc. In terms of the apparatus, we used the same setup as introduced in Section 4. A fixed window size (2048×2048 pixels) was used for showing the visualization on a 30 inch display, where the central node of each visualization was displayed in the window center. Participants used a mouse and keyboard to interact with the visualization.

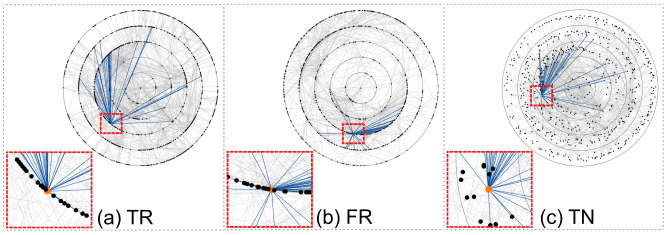


Fig. 11. One example of the sub-graphs employed in the user study with three layout methods, where a node in yellow is selected and the connected edges are highlighted in blue.

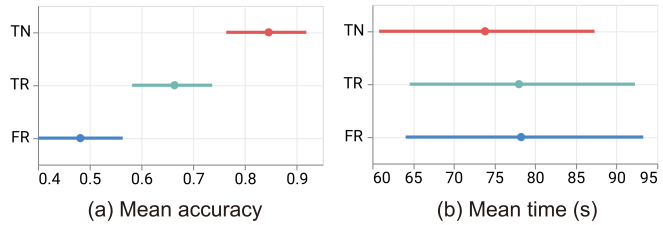


Fig. 12. Experimental results for different techniques: (a) task accuracy (a higher value is better) and (b) task completion time (a lower value is better). Error bars indicate 95% CI.

Procedure. At the beginning of the study, participants were introduced to the graph layout techniques and saw examples in a tutorial. We then described the graph exploration task and demonstrated the usage of the study system. Specifically, we explained that the distance between each node and the central node is reflected by the index of the circle or annulus contained the node. Participants were then given three practice trials, each for one technique, during which they could ask questions.

Then, participants were asked to complete the 15 trials in the actual study. The trials were presented to the participants with random order. The study system recorded the answers of the participants and the time taken for each trial. Participants can take a short break after each trial and the whole study took about 20 minutes on average. After each participant finished all trials, we conducted a short interview with one question: "Which layout style do you prefer?" and "Why do you prefer it?."

Analysis. We used non-parametric tests to examine the hypothesis H2. Namely, we applied Kruskal-Wallis H tests to compare the means of completion time and accuracy across techniques, and Mann-Whitney U tests for pairwise comparisons.

Results. We found that all selected nodes are located on the correct circles or annuli, and thus H1 is confirmed. However, our results partially support H2 that TN outperforms the other layout techniques. We observed a significant effect of the layout technique on the estimation accuracy ($p < 0.001$). Fig. 12(a) shows that TN leads to higher accuracy than the other methods. A post-hoc analysis showed that the accuracy of TN is significantly higher than that of TR ($p = 0.004$) and FR ($p < 0.001$). However, there is no significant difference on completion time($p = 0.941$), although TN takes slightly less time than the others (see Fig. 12(b).

68.2% of participants claimed that the annulus style (TN) works better than the circle style (TR and FR). For instance, one participant mentioned that "Since there are many edges around the nodes with large degrees, I searched in the area with dense edges first. However, for the cases where the nodes are all on the circles, too many adjacent nodes overlap in dense regions, making it difficult for me to identify which node has more edges. By contrast, for the cases where nodes are placed in annuli, there are less overlap even in dense areas and thus I can easily find the answer."

4.3 Case Studies

Here, we investigate the usefulness of the interactions provided by a Target Netgram, for graph exploration and show its effectiveness for dense graphs.

Interactive Exploration. Here, we explore a collaboration network which was extracted from the DBLP dataset [57]. The network allows a user to quickly learn about other researchers in the area of a researcher, and use this as a starting point to understand research in this field. Fig. 13 shows the layouts generated by our method based on such a network, which contains 88 nodes (researchers) and 170 links (co-authorship of researchers). To help the user sorting out the core structure of the network, we only show the edges between two researchers if the number of collaborative publications is greater than 5. As shown in Fig. 13(a), a Target Netgram, not only ensures that the nodes strictly satisfy the radial level constraint, but also keeps the local topology of the network as much as possible. For example, we can observe local connections among the yellow nodes.

However, in Fig. 13(a), the space on the first annulus is limited so that we cannot show any other information about the researchers closest to the central node. Then the user zooms into the first annulus, which enlarges the annulus width for the first layer (Fig. 13(b)). With enough space, a Target Netgram, allows to further displays some node information using statistical charts, so that the user can access details about the network besides its topology while maintaining the network structure.

The user can also adjust the layout based on node attributes through interaction. For example, researchers marked in light blue in Fig. 13(b) worked at or graduated from the same university but the collaboration between them is not close. To reveal their potential partnership, the user can utilize the cluster formation constraint (as described in Section 3.5) to gather them together, as shown in Fig. 13(c). After applying this constraint, *Tobias*, who was too far away from the other researchers with the same attribute, is moved closer to other blue node, forming a new cluster.

During the whole interaction, the local structure of the network is not distorted because of the stress constraint. For example, the local topology of the yellow cluster is always well expressed in all three situations. This again demonstrates the effectiveness of a Target Netgram, in assisting the exploration of the local neighborhood of a selected central node.

Dense Graphs. For all above examples, our solver can find layouts that meet three design requirements. However, it might fail for some dense graphs. Fig. 14(a) shows an example for a graph with 461 nodes and 5698 edges [58]. On the left of Fig. 14(a), the purple nodes are misplaced to be outside of the second circle that violates DR1. To correct

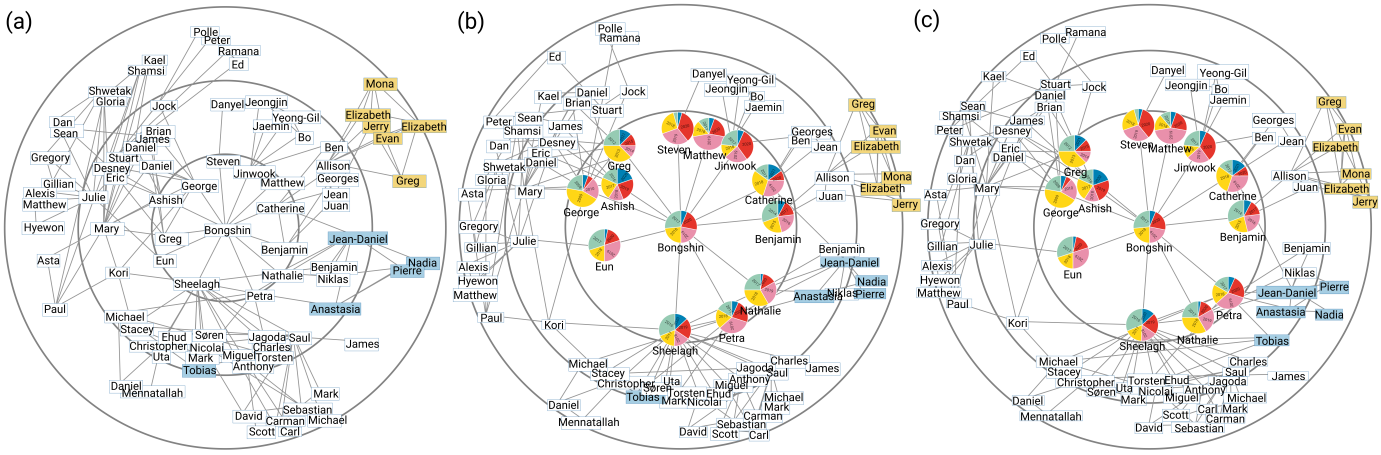


Fig. 13. Interactive exploration of a collaboration network extracted from the DBLP dataset. (a) Initial layout generated by Target Netgram,. (b) A user zooms into the first annulus and adds pie charts on the corresponding nodes. (c) Layout after the user applied the constraint of attribute-based cluster.

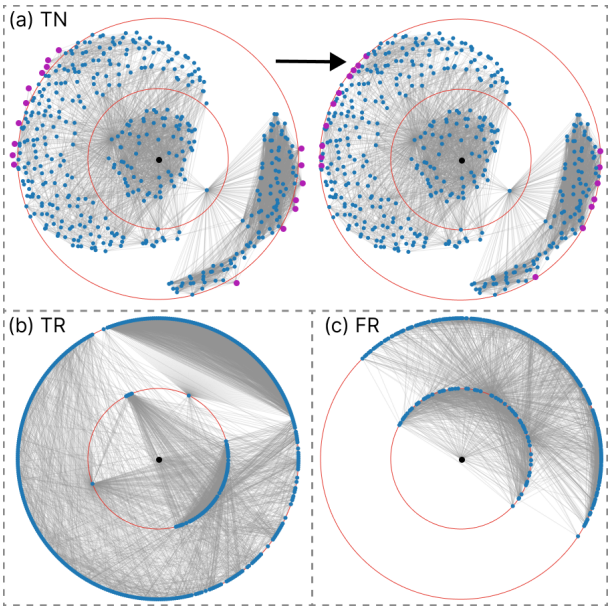


Fig. 14. Visualizing a highly dense graph extracted from the ego-facebook network [58]. (a) Results generated by Target Netgrams, where the purple nodes in the left layout do not meet DR1 but are corrected and placed onto the second circle in the right. (b,c) Results generated by TR and FR, respectively, where the two clusters shown in (a) are obscured.

them, we move them to the second circle as shown in the right of Fig. 14(a). Although such correction might destroy the graph structures to some extent, our Target Netgram, still clearly shows two major clusters. In contrast, TR and FR densely arrange all nodes on the circles without revealing any cluster, see Figs. 14(b,c).

5 Discussion, Conclusion and Future Work

Target Netgrams allow to create radial graph visualization analogous to hand-drawn target sociograms. It is achieved by a novel annulus-constrained stress model, which requires node positions to meet radial level constraints. In doing so, nodes are strictly placed in the corresponding annuli between adjacent circles, while clearly revealing the graph structures.

We further provide three constraints such as parent-child preservation, attribute-based clusters, and structure-aware radii for users to adjust edge angles, node positions and levels of interest. In addition, users can switch to another central node of interest through a smooth animated transition. To demonstrate the effectiveness and usefulness of our method, we performed a user study, quantitative comparisons, and a case study.

So far, Target Netgrams yield good results for graphs with around 1000 nodes. For data with more nodes, all radial layout methods produce dense and crowded visualizations, whereas our method enables users to interactively explore levels and nodes of interest through applying structural constraints and radii adjustment. To further enhance the number of possible nodes, we plan to investigate graph abstraction techniques [59] to reduce data size while maintaining major structures. Due to the non-convex nature of this problem (see Eq. 3), our adapted constrained least square solver does not always find a sub-optimal solution. In the future, we plan to explore more advanced non-convex optimization solvers [60]. On the other hand, our annulus-constrained stress model is a special case of boundary-constrained layouts. In the future, we will extend the current circle constraint to other shapes and apply this model to different data like origin-destination maps.

Our evaluation of Target Netgrams still has limitations. First, the user study investigated only one task that combines radial level and node degree exploration in a graph. Tasks focusing on other aspects, such as identifying common neighbors, can be explored in the future. Second, the user study focused on evaluating the layouts but not the user interactions. While we demonstrate the useful interactivity of our method in a case study, a comprehensive user study is needed to further confirm its effectiveness in exploring graphs with rich information. Third, for practical issues, we examined our method with a subset of relevant existing methods in our user studies. Although our quantitative experiments compared a larger set of baselines, there is still room for enhancing the user study with more datasets and users.

ACKNOWLEDGMENTS

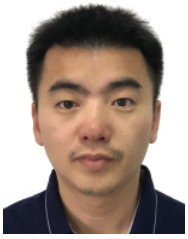
The authors like to thank the anonymous reviewers for their valuable input. This work was supported by the grants of the National Key Research & Development Plan of China (2019YFB1704201), and NSFC (62132017, 62141217).

REFERENCES

[1] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, p. 036104, 2006.
[2] G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*, 1st ed. Prentice Hall PTR, 1998.
[3] U. Brandes and C. Pich, "More flexible radial layout." *Journal of Graph Algorithms and Applications*, vol. 15, no. 1, pp. 157–173, 2011.
[4] P. J. Carrington, J. Scott, and S. Wasserman, *Models and methods in social network analysis*. Cambridge University Press, 2005, vol. 28.
[5] P. Eades, "Drawing free trees," *Bulletin for the Institute for Combinatorics and its Applications*, no. 5, pp. 10–36, 1991.
[6] A. Pavlo, C. Homan, and J. Schull, "A parent-centered radial layout algorithm for interactive graph visualization and animation," *arXiv preprint cs/0606007*, 2006.
[7] T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Information Processing Letters*, vol. 31, no. 1, pp. 7–15, 1989.
[8] M. L. Northway, "A method for depicting social relationships obtained by sociometric testing," *Sociometry*, pp. 144–150, 1940.
[9] L. C. Freeman, "Visualizing social networks," *Journal of social structure*, vol. 1, no. 1, p. 4, 2000.
[10] U. Brandes, P. Kenis, and D. Wagner, "Communicating centrality in policy network drawings," *IEEE Trans. Vis. & Comp. Graphics*, vol. 9, no. 2, pp. 241–253, 2003.
[11] B. Hogan, J. R. Melville, G. L. Phillips II, P. Janulis, N. Contractor, B. S. Mustanski, and M. Birkett, "Evaluating the paper-to-screen translation of participant-aided sociograms with high-risk participants," in *Proc. of the CHI Conference on Human Factors in Computing Systems*, 2016, pp. 5360–5371.
[12] J. L. Mead and R. A. Renaut, "Least squares problems with inequality constraints as quadratic constraints," *Linear Algebra and its Applications*, vol. 432, no. 8, pp. 1936 – 1949, 2010.
[13] G. J. Wills, "Nicheworks: interactive visualization of very large graphs," *Journal of Computational and Graphical Statistics*, vol. 8, no. 2, pp. 190–212, 1999.
[14] K. P. Yee, D. Fisher, R. Dhamija, and M. Hearst, "Animated exploration of graphs with radial layout," in *Proc. of IEEE InfoVis*, 2001, pp. 43–50.
[15] T. Jankun-Kelly and K. L. Ma, "Moiregraphs: Radial focus+context visualization and interaction for graphs with visual nodes," in *Proc. of IEEE InfoVis*, 2003, pp. 59–66.
[16] G. Huang, Y. Li, X. Tan, Y. Tan, and X. Lu, "Planet: A radial layout algorithm for network visualization," *Physica A: Statistical Mechanics and its Applications*, vol. 539, p. 122948, 2020.
[17] C. Bachmaier, F. J. Brandenburg, and M. Forster, "Radial level planarity testing and embedding in linear time," *Journal of Graph Algorithms and Applications*, pp. 393–405, 2004.
[18] K. Sugiyama, S. Tagawa, and M. Toda, "Methods for visual understanding of hierarchical system structures," *IEEE Trans. Syst. Man Cybern.*, vol. 11, no. 2, pp. 109–125, 1981.
[19] T. Dwyer and Y. Koren, "Dig-CoLa: directed graph layout through constrained energy minimization," in *Proc. of IEEE InfoVis*. IEEE, 2005, pp. 65–72.
[20] W. Playfair, *The statistical breviary*. Wallis, 1801.
[21] F. Nightingale, *Notes on matters affecting the health, efficiency, and hospital administration of the British army: founded chiefly on the experience of the late war*. Harrison and Sons, St. Martin's Lane, WC, 1987.
[22] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "Dna visual and analytic data mining," in *Proc. of IEEE InfoVis*, 1997, pp. 437–ff.
[23] J. Stasko and E. Zhang, "Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations," in *Proc. of IEEE InfoVis*, 2000, p. 57.
[24] D. A. Keim, F. Mansmann, J. Schneidewind, and T. Schreck, "Monitoring network traffic with radial traffic analyzer," in *Proc. of IEEE VAST*, 2006, pp. 123–128.
[25] B. M. Institute, "Interactive visualization of multiple query results," in *Proc. of IEEE InfoVis*, 2001.
[26] M. Weber, M. Alexa, and W. Müller, "Visualizing time-series on spirals," in *Proc. of IEEE InfoVis*, 2001, pp. 7–14.
[27] T. Castermans, K. Verbeek, B. Speckmann, M. A. Westenberg, R. Koopman, S. Wang, H. Van Den Berg, and A. Betti, "Solarview: low distortion radial embedding with a focus," *IEEE Trans. Vis. & Comp. Graphics*, vol. 25, no. 10, pp. 2969–2982, 2018.
[28] M. Northway, "A primer of sociometry." *Sociometry*, vol. 15, pp. 400–400, 1952.
[29] W. S. Cleveland and R. McGill, "An experiment in graphical perception," *International Journal of Man-Machine Studies*, vol. 25, no. 5, pp. 491–500, 1986.
[30] J. Goldberg and J. Helfman, "Eye tracking for visualization evaluation: Reading values on linear versus radial graphs," *Proc. of IEEE InfoVis*, vol. 10, no. 3, pp. 182–195, 2011.
[31] M. Waldner, A. Diehl, D. Gračanin, R. Splechtna, C. Delrieux, and K. Matković, "A comparison of radial and linear charts for visualizing daily patterns," *IEEE Trans. Vis. & Comp. Graphics*, vol. 26, no. 1, pp. 1033–1042, 2019.
[32] M. Burch and D. Weiskopf, "On the benefits and drawbacks of radial diagrams," in *Handbook of Human Centric Visualization*, 2014, pp. 429–451.
[33] J. Stasko, R. Catrambone, M. Guzdial, and K. McDonald, "An evaluation of space-filling information visualizations for depicting hierarchical structures," *International Journal of Human-Computer Studies*, vol. 53, no. 5, pp. 663–694, 2000.
[34] M. Burch, N. Konevtsova, J. Heinrich, M. Höferlin, and D. Weiskopf, "Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study," *IEEE Trans. Vis. & Comp. Graphics*, vol. 17, no. 12, pp. 2440–2448, 2011.
[35] S. Diehl, F. Beck, and M. Burch, "Uncovering strengths and weaknesses of radial visualizations—an empirical approach," *IEEE Trans. Vis. & Comp. Graphics*, vol. 16, no. 6, pp. 935–942, 2010.
[36] Y. Albo, J. Lanir, P. Bak, and S. Rafaeli, "Off the radar: Comparative evaluation of radial visualization solutions for composite indicators," *IEEE Trans. Vis. & Comp. Graphics*, vol. 22, no. 1, pp. 569–578, 2015.
[37] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry, "Task taxonomy for graph visualization," in *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, 2006, pp. 1–5.
[38] J. Zhao, M. Glueck, F. Chevalier, Y. Wu, and A. Khan, "Egocentric analysis of dynamic networks with egolines," in *Proc. of the CHI Conference on Human Factors in Computing Systems*, 2016, pp. 5003–5014.
[39] B. L. Perry, B. A. Pescosolido, and S. P. Borgatti, *Egocentric network analysis: Foundations, methods, and models*. Cambridge University Press, 2018, vol. 44.
[40] E. C. Nevis, *Organizational consulting: A Gestalt approach*. Taylor & Francis, 2013.
[41] E. R. Gansner, Y. Koren, and S. North, "Graph drawing by stress majorization," in *International Symposium on Graph Drawing*, 2004, pp. 239–250.
[42] Y. Wang, Y. Wang, Y. Sun, L. Zhu, K. Lu, C. Fu, M. Sedlmair, O. Deussen, and B. Chen, "Revisiting stress majorization as a unified framework for interactive constrained graph visualization," *IEEE Trans. Vis. & Comp. Graphics*, vol. 24, no. 1, pp. 489–499, 2018.
[43] P. B. Stark and R. L. Parker, "Bounded-variable least-squares: an algorithm and applications," *Computational Statistics*, vol. 10, pp. 129–129, 1995.
[44] B. Schofield, "On active set algorithms for solving bound-constrained least squares control allocation problems," in *American Control Conference*, 2008, pp. 2597–2602.
[45] J. W. Demmel, *Applied numerical linear algebra*. SIAM, 1997.
[46] J. X. Zheng, S. Pawar, and D. F. Goodman, "Graph drawing by stochastic gradient descent," *IEEE Trans. Vis. & Comp. Graphics*, vol. 25, no. 9, pp. 2738–2748, 2018.
[47] T. Dwyer, "Scalable, versatile and simple constrained graph layout," *Computer Graphics Forum*, vol. 28, no. 3, pp. 991–998, 2009.
[48] Y. Hu and L. Shi, "Visualizing large graphs," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 7, no. 2, pp. 115–136, 2015.
[49] D. Chakrabarti and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Computing Surveys*, vol. 38, no. 1, pp. 2–es, 2006.
[50] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. Stolper, M. Sedlmair, J. Chen, T. Möller, and J. Stasko, "vispubdata.org: A

metadata collection about IEEE visualization (VIS) publications," *IEEE Trans. Vis. & Comp. Graphics*, vol. 23, no. 9, pp. 2199–2206, 2017.
[51] B. Rozemberczki and R. Sarkar, "Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models," in *Proc. of the ACM International Conference on Information and Knowledge Management*, 2020, pp. 1325–1334.
[52] E. R. Gansner, Y. Hu, and S. North, "A maxent-stress model for graph layout," *IEEE Trans. Vis. & Comp. Graphics*, vol. 19, no. 6, pp. 927–940, 2012.
[53] P. Eades, S.-H. Hong, K. Klein, and A. Nguyen, "Shape-based quality metrics for large graph visualization," in *International Symposium on Graph Drawing*, 2015, pp. 502–514.
[54] H. C. Purchase, "Metrics for graph drawing aesthetics," *Journal of Visual Languages & Computing*, vol. 13, no. 5, pp. 501–516, 2002.
[55] F. Du, N. Cao, Y. R. Lin, P. Xu, and H. Tong, "isphere: Focus+context sphere visualization for interactive large graph exploration," in *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, 2017, pp. 2916–2927.
[56] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: a discriminative approach," in *Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 927–936.
[57] "The dblp team: dblp computer science bibliography. Monthly snapshot release of Match 2021," https://dblp.org/xml/release/dblp-2021-03-01.xml.gz.
[58] J. Leskovec and J. Mcauley, "Learning to discover social circles in ego networks," *Advances in neural information processing systems*, vol. 25, 2012.
[59] J. J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks." in *NIPS*, vol. 2012, 2012, pp. 548–56.
[60] R. G. Strongin and Y. D. Sergeyev, *Global optimization with non-convex constraints: Sequential and parallel algorithms*, 2013, vol. 45.

Mingliang Xue is working toward the fourth-year Ph.D. student in the School of Computer Science and Technology, Shandong University. His research interests include graph visualization and dimensional reduction.



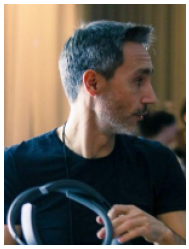
Yunhai Wang is professor in School of Computer Science and Technology at Shandong University. He serves as the associate editor of Computer Graphics Forum. His interests include scientific visualization, information visualization and computer graphics.

Chang Han is an undergraduate student in the School of Information Science and Engineering at Shandong University. His research interests include scientific visualization and information visualization.

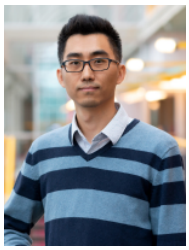
Jian Zhang received his PhD degree in Applied Mathematics from the University of Minnesota in 2005. After a postdoc at Pennsylvania State University, he is now a professor in the Computer Network Information Center, Chinese Academy of Sciences (CAS). His current research interests include scientific computing and scientific visualization.

Zheng Wang is a professor of Chinese Academy of Sciences, and now he serves as Chief Innovation Officer of CITC. His interests topics include AI application in ITC, emerging technologies transfer into industry and Business Intelligence in Internet, etc.

Kaiyi Zhang is working toward the second-year master student in the School of Computer Science and Technology, Shandong University. His research interests include text visualization and graph visualization.



Christophe Hurter received the PhD degree in computer science from the University of Toulouse. He is a researcher at French Civil Aviation University, Toulouse, France. He teaches and conducts research into human-computer interaction, and more specifically information visualization (InfoVis). His work is currently focused on new visual designs and interaction paradigms in order to improve data exploration. He received an Aerospace Valley PhD award in 2011.



Jian Zhao is an assistant professor with the Cheriton School of Computer Science, University of Waterloo. His research interests include Information Visualization, Human-Computer Interaction, Visual Analytics and Data Science. His work contributes to the development of advanced interactive visualizations that promote the interplay of human, machine, and data.

Oliver Deussen graduated at Karlsruhe Institute of Technology and is professor at University of Konstanz (Germany) and visiting professor at the Chinese Academy of Science in Shenzhen. He served as Co-Editor in Chief of Computer Graphics Forum and was President of the Eurographics Association. His areas of interest are modeling and rendering of complex biological systems, non-photorealistic rendering as well as Information Visualization.