First publ. in: Proceedings of the 15th International Symposium on Graph Drawing (GD ´07) (LNCS 4875), pp. 255-267

Multi-circular Layout of Micro/Macro Graphs*

Michael Baur[1] and Ulrik Brandes[2]

[1] Department of Computer Science, Universität Karlsruhe (TH), Germany
baur@informatik.uni-karlsruhe.de

[2] Department of Computer & Information Science, University of Konstanz, Germany
Ulrik.Brandes@uni-konstanz.de

Abstract. We propose a layout algorithm for micro/macro graphs, i.e. relational structures with two levels of detail. While the micro-level graph is given, the macro-level graph is induced by a given partition of the micro-level vertices. A typical example is a social network of employees organized into different departments. We do not impose restrictions on the macro-level layout other than sufficient thickness of edges and vertices, so that the micro-level graph can be placed on top of the macro-level graph. For the micro-level graph we define a combinatorial multi-circular embedding and present corresponding layout algorithms based on edge crossing reduction strategies.

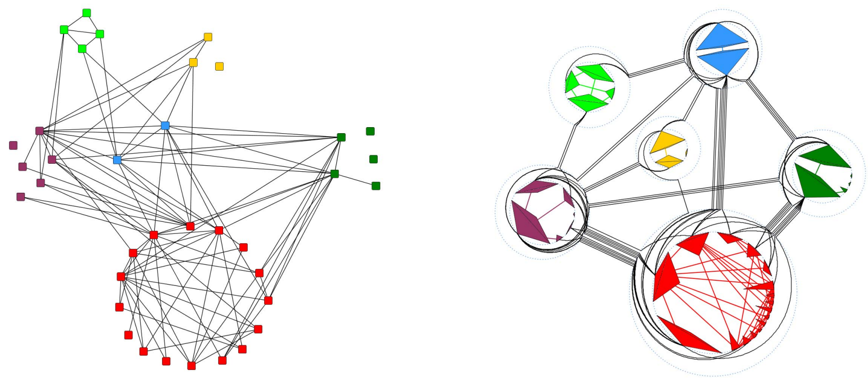
1 Introduction

An important aspect in the visualization of many types of networks is the interplay between fine- and coarse-grained structures. Think, for instance, of low-level interaction giving rise to emergent features at a larger scale, or people implementing organizational relations. Assuming that the structure on the micro level is a graph, a macro-level graph may originate from a group-level network analysis such as clustering or role analysis (e.g., [5]), from an attribute-based partitioning of the vertices, or may just be given in advance.

Depending on the particular application domain and other contexts, different layout methods will be appropriate for the macro graph. Since we only require large nodes and thick edges, we assume it is given. Either the macro-level layout algorithm can handle varying vertex size (e.g., [12,21]) and edge thickness (e.g., [7]), or some post-processing is applied (e.g., [11]).

Given a drawing of the macro-level graph with large nodes and thick edges, each vertex of the micro-level graph is drawn in the area defined by the macro vertex it belongs to, and each micro edge is routed through its corresponding macro edge. We propose a multi-circular layout model for the micro graph. Each micro vertex is placed on a circle inside of the area of its corresponding macro vertex and micro edges whose end vertices belong to the same macro vertex are drawn inside of these circles. All other micro edges are then drawn inside of their corresponding macro edges and at constant but different distances from the border of the macro edge, i.e. in straight-line macro edges they are

* Research partially supported by DFG, grants Wa 654/13-2 and Br 2158/2-3.



(a) geometric grouping and straight-line edges

(b) multi-circular layout

Fig. 1. (a) Example organizational network with geometric grouping and straight-line edges (redrawn from [15]). In our multi-circular layout (b), all details are still present and the macro structure induced by the grouping becomes visible. The height and width of the vertices reflects the number of connections within and between groups.

drawn as parallel lines. These edges must also be routed inside the area of macro vertices to connect to their endpoints, but are not allowed to cross the circles. In principle, an arbitrary layout strategy can be used as long as it complies with these requirements. Figure 1 shows a concrete example of this model. Micro edges connecting vertices in the same macro vertex are drawn as straight lines. Inside of macro vertices, the other edges spiral around the circle of micro vertices until they reach the area of the macro edge. We give a combinatorial description of the above model and then focus on the algorithmically most challenging aspect of these layouts, namely crossing reduction by cyclic ordering of micro vertices and choosing edge winding within macro vertices. Finally, we apply the multi-circular layout to an email communication network to exemplify its use case.

While the drawing convention consists of proven components (geometric grouping is used, e.g., in [15,20], and edge routing to indicate coarse-grained structure is proposed in, e.g., [13,3]), our approach is novel in the way we organize micro vertices to let the macro structure dominate the visual impression without cluttering the micro-level details too much. Note also that the setting is very different from layout algorithms operating on structure-induced clusterings (e.g., [14,1]), since we cannot make any assumptions on the structure of clusters (they may even consist of isolates). Therefore, we neither want to utilize the clustering for better layout, nor do we want to display the segregation into dense subregions or small cuts. Our aim is to represent the interplay between a (micro-level) graph and a (most likely extrinsic) grouping of its vertices.

After defining some basic terminology in Sect. 2, we state required properties for macro-graph layout in Sect. 3. Multi-circular micro-graph layout is discussed in more detail in Sect. 4 and crossing reduction algorithms for it are given in Sect. 5. We conclude with an application in Sect. 6.

2 Preliminaries

Throughout this paper, let $G = (V, E)$ be a simple undirected graph with $n = |V|$ vertices and $m = |E|$ edges. Furthermore, let $E(v) = \{\{u, v\} \in E : u \in V\}$ denote the incident edges of a vertex $v \in V$, let $N(v) = \{u \in V : \{u, v\} \in E\}$ denote its neighbors, and let $sgn : \mathbb{R} \rightarrow \{-1, 0, 1\}$ be the signum function.

Since each micro-vertex is required to belong to exactly one macro-vertex, the macro structure defines a clustering, or partitioning, of the micro-vertices. Contrary to this top-down approach, we can also start from the bottom. A *partition assignment* $\phi : V \rightarrow \{0, \ldots, k-1\}$ for G subdivides the (micro-)vertex set V into k pairwise disjoint subsets $V = V_0 \dot{\cup} \ldots \dot{\cup} V_{k-1}$, where $V_i = \{v \in V : \phi(v) = i\} = \phi^{-1}(i)$. An edge $e = \{u, v\} \in V_i \times \in V_j$ is called an *intra-partition edge* iff $i = j$, otherwise it is called an *inter-partition edge*. The set of intra-partition edges of a partition V_i is denoted by E_i, the set of inter-partition edges of two partitions V_i, V_j by $E_{i,j}$. We use $G = (V, E, \phi)$ to denote a graph $G = (V, E)$ and a related partition assignment ϕ.

A *circular order* $\pi = \{\pi_0, \ldots, \pi_{k-1}\}$ defines for each partition V_i a vertex order π_i as a bijective function $\pi_i : V_i \rightarrow \{0, \ldots, |V_i| - 1\}$ with $u \prec v \Leftrightarrow \pi_i(u) < \pi_i(v)$ for any two vertices $u, v \in V_i$. An order π_i can be interpreted as a counter-clockwise sequence of distinct positions on the circumference of a circle.

3 Macro Layout

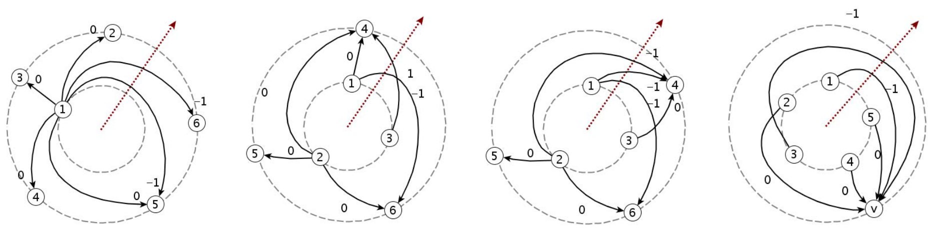
A prototypical macro graph, the *quotient graph*, is defined by a partition assignment. Given a partition assignment $\phi : V \rightarrow \{0, \ldots, k-1\}$, the corresponding quotient graph $Q(G, \phi) = (V_Q, E_Q)$ contains a vertex for each partition of G and two vertices $V_i, V_j \in V_Q$ are connected iff E contains at least one edge between a vertex in V_i and a vertex in V_j.

We do not require a specific layout strategy for the macro graph as long as its elements are rendered with sufficient thickness to draw the underlying micro graph on top of them. To achieve this, post-processing can be applied to any given layout [11] or methods which consider vertex size (e.g., [12,21]) and edge thickness (e.g., [7]) have to be used.

From a macro layout we get *partition orders* $\Pi_i : V_Q \setminus V_i \rightarrow \{0, .., \deg(V_i) - 1\}$ for each partition V_i, defined by the sequence of its incident edges in $Q(G, \phi)$, and a partition order $\Pi = \{\Pi_0, \ldots, \Pi_{k-1}\}$ for G. For each macro vertex this can be seen as a counter-clockwise sequence of distinct docking positions for its incident (macro) edges on its border.

4 Micro Layout

Before we discuss the multi-circular layout model for the micro graph, let us recall the related concepts of (single) circular and radial embeddings. In *(single) circular layouts* all vertices are placed on a single circle and edges are drawn as



(a) some incident edges

(b) node 4 is at position 0

(c) node 4 rotated to position 2

(d) without parting

Fig. 2. Radial layouts. Edges are labeled with their winding value.

straight lines. Therefore, a *(single) circular embedding* ε of a graph $G = (V, E)$ is fully defined by a vertex order π, i.e. $\varepsilon = \pi$ [4]. Two edges $e_1, e_2 \in E$ *cross* in ε iff the end vertices of e_1, e_2 are encountered alternately in a cyclic traversal.

4.1 Radial Layout

In *radial layouts* the partitions are placed on nested concentric circles (*levels*) and edges are drawn as curves between consecutive partitions. Therefore, only graphs $G = (V, E)$ with a *proper* partition assignment $\phi : V \to \{0, \dots, k-1\}$ are allowed, i.e. $|\phi(u) - \phi(v)| = 1$ for all edges $\{u, v\} \in E$. For technical reasons, edges are considered to be directed from lower to higher levels.

Recently, Bachmaier [2] investigated such layouts. They introduced a *ray* from the center to infinity to mark the start and end of the circular vertex orders. Using this ray it is also possible to count how often and in which direction an edge is wound around the common center of the circles. We call this the *winding* $\psi : E \to \mathbb{Z}$ of an edge (*offset* in [2]). $|\psi(e)|$ counts the number of crossings of the edge with the ray and the sign reflects the mathematical direction of rotation. See Figure 2 for some illustrations. Finally, a *radial embedding* ε of a graph $G = (V, E, \phi)$ is defined to consist of a vertex order π and an edge winding ψ, i.e. $\varepsilon = (\pi, \psi)$. Note that the rotation of a partition without permuting the vertices changes the positions and winding values but not the number of crossings.

Crossings between edges in radial embeddings depend on their winding and on the order of the end vertices. There can be more than one crossing between two edges if they have very different winding. We denote the number of crossings between two edges $e_1, e_2 \in E$ in an radial embedding ε by $\chi_\varepsilon(e_1, e_2)$. The (radial) crossing number of an embedding ε and a level graph $G = (V, E, \phi)$ is then naturally defined as $\chi(\varepsilon) = \sum_{\{e_1,e_2\} \in E, e_1 \neq e_2} \chi_\varepsilon(e_1, e_2)$ and $\chi(G) = \min\{\chi(\varepsilon) : \varepsilon$ is a radial embedding of $G\}$ is called the *radial crossing number* of G.

Theorem 1 ([2]). *Let* $\varepsilon = (\pi, \psi)$ *be a radial embedding of a 2-level graph* $G = (V_1 \dot{\cup} V_2, E, \phi)$. *The number of crossings* $\chi_\varepsilon(e_1, e_2)$ *between two edges* $e_1 = (u_1, v_1) \in E$ *and* $e_2 = (u_2, v_2) \in E$ *is*

$$\chi_\varepsilon(e_1,e_2) = \max\left\{0, \left|\psi(e_2) - \psi(e_1) + \frac{b-a}{2}\right| + \frac{|a|+|b|}{2} - 1\right\} ,$$
$$\textit{where } a = sgn(\pi_1(u_2) - \pi_1(u_1)) \textit{ and } b = sgn(\pi_2(v_2) - \pi_2(v_1)) \ .$$

Bachmaier also states that in crossing minimal radial embeddings every pair of edges crosses at most once and incident edges do not cross at all. As a consequence, only embeddings need to be considered where there is a clear *parting* between all edges incident to the same vertex u. The parting is the position of the edge list of u that separates the two subsequences with different winding values. See Figure 2 for layouts with and without proper parting.

4.2 Multi-circular Layouts

Unless otherwise noted, vertices and edges belong to the micro-level in the following. In the micro layout model each vertex is placed on a circle inside of its corresponding macro vertex. Intra-partition edges are drawn within these circles as straight lines. Inter-partition edges are drawn inside their corresponding macro edges and at constant but different distances from the border of the macro edge. To connect to their incident vertices, this edges must also be routed inside of macro vertices. Since they are not allowed to cross the circles, they are drawn as curves around them. We call such a drawing a *(multi-)circular layout*.

Since intra- and inter-partition edges can not cross, all crossings of intra-partition edges are completely defined by the vertex order π_i of each partition V_i. Intuitively speaking, a vertex order defines a circular layout for the intra-partition edges. In the following we thus concentrate on inter-partition edges.

The layout inside each macro vertex V_i can be seen as a 2-level radial layout. The orders can be derived from the vertex order π_i and the partition order Π_i. Similar to radial layouts we introduce a *ray* for each partition and define the beginning of the orders and the edge winding according to these rays. Note that for each edge $e = \{u, v\} \in E$, $u \in V_i$, $v \in V_j$, two winding values are needed, one for the winding around partition V_i denoted by $\psi_i(e) = \psi_u(e)$, and one for the winding around partition V_j denoted by $\psi_j(e) = \psi_v(e)$. If the context implies an implicit direction of the edges we call windings either source or target windings respectively. Since radial layouts can be rotated without changing the embedding, rays of different partitions are independent and can be arbitrary directed. Finally, a *multi-circular embedding* ε is defined by a vertex order π, a partition order Π, and the winding of the edges ψ, i.e. $\varepsilon = (\pi, \Pi, \psi)$.

Observation 2. *For each partition V_i in a multi-circular embedding $\varepsilon = (\pi, \Pi, \psi)$ a 2-level radial embedding $\varepsilon_i = ((\pi_i, \pi'), \psi_i)$ is defined by the vertex order π_i, the partition order Π_i, and the edge winding ψ_i, where $\pi'(v) = \Pi_i(\phi(v)), v \in V \setminus V_i$.*

There is another connection between radial and multi-circular layouts. A 2-level radial layout can easily be transformed in a 2-partition circular layout and vice versa. Given a graph $G = (V_1 \dot\cup V_2, E, \phi)$ and a radial embedding $\varepsilon = (\pi, \psi)$ of G, the 2-partition circular embedding $\varepsilon^* = (\pi^*, \Pi^*, \psi^*)$ defined by $\pi_1^* = \pi_1$, $\pi_2^* = -\pi_2$,

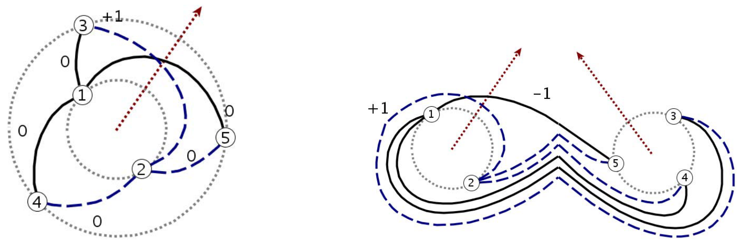


Fig. 3. A 2-level radial layout and its corresponding 2-circular layout

$\Pi_1^* = 0$, $\Pi_2^* = 0$, and $\psi_1^*(e) = \psi(e)$, $\psi_2^*(e) = 0$ realizes exactly the same crossings. See Figure 3 for an example. Intuitively speaking, the topology of the given radial embedding is not changed if we drag the two circles apart and reverse one of the vertex orders. If a 2-partition circular embedding $\varepsilon^* = (\pi^*, \Pi^*, \psi^*)$ is given, a related radial embedding $\varepsilon = (\pi, \psi)$ is defined by $\pi_1 = \pi_1^*$, $\pi_2 = -\pi_2^*$, and $\psi(e) = \psi_1(e) - \psi_2(e)$.

Observation 3. *There is a one-to-one correspondence between a 2-level radial embedding and a 2-circular embedding.*

Crossings in the micro layout are due to either the circular embedding or crossing macro edges. Since crossings of the second type can not be avoided by changing the micro layout, we do not consider them in the micro layout model. Obviously, pairs of edges which are not incident to a common macro vertex can only cause crossings of this type. For pairs of edges which are incident to at least one common macro vertex we can define corresponding 2-level radial layouts using Observations 2 and 3 and compute the number of crossings by modifications of Theorem 1.

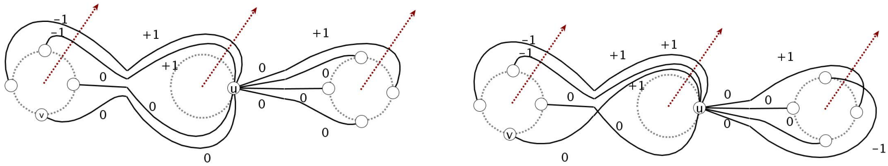
Theorem 4. *Let $\varepsilon = (\pi, \Pi, \psi)$ be a multi-circular embedding of a graph $G = (V, E, \phi)$ and let $e_1 = \{u_1, v_1\}$, $e_2 = \{u_2, v_2\} \in E$ be two inter-partition edges.*

If e_1 and e_2 share exactly one common incident macro vertex, e.g., $V_i = \phi(u_1) = \phi(u_2)$, $\phi(v_1) \neq \phi(v_2)$, then the number of crossings of e_1 and e_2 is

$$\chi_\varepsilon(e_1, e_2) = \max\left\{0, \left|\psi_i(e_2) - \psi_i(e_1) + \frac{b-a}{2}\right| + \frac{|a|+|b|}{2} - 1\right\},$$
$$\text{where } a = sgn(\pi_i(u_2) - \pi_i(u_1)) \text{ and } b = sgn(\Pi(\phi(v_2)) - \Pi(\phi(v_1))) .$$

If e_1 and e_2 belong to the same macro edge, e.g., $V_i = \phi(u_1) = \phi(u_2)$, $V_j = \phi(v_1) = \phi(v_2)$, then the number of crossings of e_1 and e_2 is

$$\chi_\varepsilon(e_1, e_2) = \max\left\{0, \left|\psi'(e_2) - \psi'(e_1) + \frac{b-a}{2}\right| + \frac{|a|+|b|}{2} - 1\right\},$$
$$\text{where } a = sgn(\pi_i(u_2) - \pi_i(u_1)) , b = sgn(\pi_j(v_1) - \pi_j(v_2)) , \text{ and}$$
$$\psi'(e) = \psi_i(e) + \psi_j(e) .$$



(a) parting (b) edge $\{u, v\}$ violates parting

Fig. 4. Not all winding combinations for the incident edges of u result in a good layout

Similar to radial layouts, in a crossing minimal multi-circular embedding incident edges do not cross and there is at most one crossing between every pair of edges. Therefore, only embeddings need to be considered where there is a clear *parting* between all edges incident to the same vertex $u \in V_i$. Since in multi-circular layouts winding in different macro vertices can be defined independently, we split the edge list $E(u)$ of u by target partitions and get edge lists $E(u)_j = \{\{u, v\} \in E(u) : v \in V_j\}$. For each list $E(u)_j$, we get a position ℓ_j that separates the two subsequences with different values of winding ψ_j and defines the parting for this partition. Furthermore, there is also a parting for V_i defined on the edge list $E(u)$. The order of $E(u)$ for this parting depends on the partings ℓ_j in the target partitions V_j. Edges are sorted by the partition order, and for edges to the same partition V_j, ties are broken by the reverse vertex order started not at the ray but at the parting position ℓ_j. Then, the parting for V_i is the position ℓ_i which separates different values of winding ψ_i in the so ordered list. See Figure 4 for a layout with parting and a layout where the edge $\{u, v\}$ violates the parting.

Corollary 1. *Multi-circular crossing minimization is $\mathcal{NP}$-hard.*

Proof. Single circular and radial crossing minimization [2,17] are $\mathcal{NP}$-hard. As we have already seen, these two crossing minimization problems are subproblems of the multi-circular crossing minimization problem, proofing the corollary. □

As a consequence, we do not present exact algorithms for crossing minimization in multi-circular layouts. Instead, we propose extensions of some well known crossing reduction heuristics for horizontal and radial crossing reduction.

5 Layout Algorithms

Since the drawing of inter-partition edges inside a macro vertex can be seen as a radial drawing, a multi-circular layout can be composed of separate radial layouts for each macro vertex (for instance using the techniques of [20,10,2]. Such a decomposition approach, however, is inappropriate since intra-partition edges are not considered at all and inter-partition edges are not handled adequately due to the lack of information about the layout at the other macro vertices. E.g., choosing a path with more crossings in one macro vertex can allow a routing with much less crossings on the other side.

Nevertheless, we initially present in this section adaptations of radial layout techniques because they are quite intuitive, fast, and simple, and can be used for the evaluation of more advanced algorithms.

5.1 Barycenter and Median Layouts

The basic idea of both the barycenter and the median layout heuristic is the following: each vertex is placed in a central location computed from the positions of its neighbors - in either the barycenter or the median position - to reduce edge lengths and hence the number of crossings. For a 2-level radial layout, the *Cartesian Barycenter* heuristic gets the two levels and a fixed order for one of them. All vertices of the fixed level are set to equidistant positions on a circle and the component-wise barycenter for all vertices of the second level is computed. The cyclic order around the center defines the order of the vertices and the edges are routed along the geometrically shortest-path. The *Cartesian Median* heuristic is defined similar. Running time for both heuristics is in $\mathcal{O}(|E| + |V| \log |V|)$.

Both heuristics are easily extended for multi-circular layouts. The layout in each macro vertex V_i is regarded as a separate 2-level radial layout as described in Observation 3 and the partition orders Π_i are used to define the orders of the fixed levels. Because of the shortest-path routing, no two edges cross more than once and incident edges do not cross at all in the final layout. On the other hand are crossings avoided by the used placement and winding strategies only indirectly by edge length reduction.

5.2 Multi-circular Sifting

To overcome the drawbacks of the radial layout algorithms described before, we propose an extension of the sifting heuristic which computes a complete multi-circular layout and considers edge crossings for optimizing both vertex order and edge winding, and thus is expected to generate better layouts.

Sifting was originally introduced as a heuristic for vertex minimization in ordered binary decision diagrams [19] and later adapted for the layered one-sided, the circular, and the radial crossing minimization problems [18,4,2]. The idea is to keep track of the objective function while moving a vertex along a fixed order of all other vertices. The vertex is then placed in its (locally) optimal position. The method is thus an extension of the greedy-switch heuristic [8]. For crossing reduction the objective function is the number of crossings between the edges incident to the vertex under consideration and all other edges. In multi-circular layouts this function depends on both the vertex order and the edge winding. Therefore, we have to find for each position of a vertex the winding values for its incident edges which result in the minimal crossing number.

The efficient computation of crossing numbers in sifting for layered and single circular layouts is based on the locality of crossing changes, i.e. swapping consecutive vertices $u \hookrightarrow v$ only affects crossings between edges incident to u with edges incident to v. In multi-circular layouts this property clearly holds for intra-partition edges since they form (single-)circular layouts. For inter-partition edges the best routing path may require an update of the windings. Such a change can affect crossings with all edges incident to the involved partitions.

Since swapping the positions of two consecutive vertices (and keeping the winding values) only affects incident edges, the resulting change in the number of crossings can be efficiently computed. Therefore, we need an efficient update strategy for edge windings while $u \in V_i$ moves along the circle. We do not consider each possible combination of windings for each position of u. but keep track of the parting of the edges. Note that we have to alter simultaneously the parting for the source partition and all the partings for the target partitions because for an edge, a changed winding in the source partition may allow a better routing with changed winding in the target partition. Intuitively speaking, the parting in the source partition should move around the circle in the same direction as u, but on the opposite side of the circle, while the parting in the target partitions should move in the opposite direction. Otherwise, edge lengths increase and with it the likelihood of crossings. Thus, we start with winding values $\psi_u(e) = 1$ and $\psi_v(e) = 1$ for all $e = \{u, v\} \in E(v)$ and iteratively move parting counters around the circles and mostly decrease this values in the following way:

1. First try to improve the parting at V_i, i.e. the value of ψ_u for the current parting edge is decreased and the parting moved counter-clockwise to the next edge, until this parting can no longer be improved.
2. For edges whose source winding were changed in step one, there may be better target windings which can not be found in step three, because the value of ψ_j has to be increased, i.e. for each affected edge, the value of ψ_j for the edge is increased until no improvement is made.
3. Finally try to improve the parting for each target partition V_j separately, i.e. for each V_j the value of ψ_j for the current parting edge is decreased and the parting moved clockwise to the next edge, until this parting can no longer be improved.

After each update, we ensure that all counters are valid and that winding values are never increased above 1 and below -1.

Based on the above, the locally optimal position of a single vertex can be found by iteratively swapping the vertex with its neighbor and updating the edge winding while keeping track of the change in crossing number. After the vertex has past each position, it is placed where the intermediary crossing counts reached their minimum. Repositioning each vertex once in this way is called a *round of sifting.*

Theorem 5. *The running time of multi-circular sifting is in* $\mathcal{O}(|V| \cdot |E|^2)$.

Proof. Computing the difference in cross count after swapping two vertices requires $\mathcal{O}(|E|^2)$ running time for one round of sifting. For each edge the winding changes only a constant number of times because values are bounded, source winding and target winding are decreased in steps one and three resp., and the target winding is only increased for edges whose source winding decreased before. Counting the crossings of an edge after changing its winding takes time

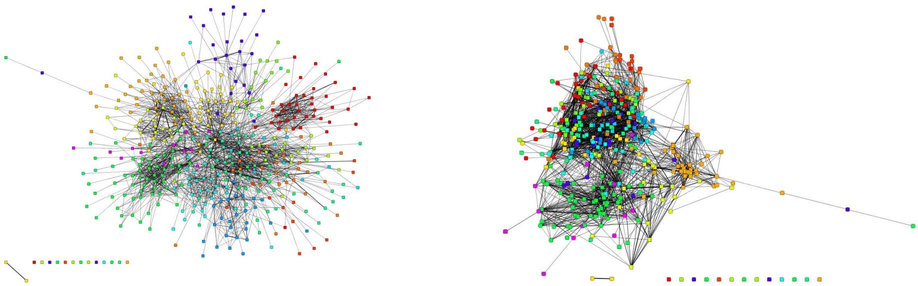


Fig. 5. Drawings of the email network generated by a force-directed method (left) and by multi-dimensional scaling (MDS, right)

$\mathcal{O}(|E|)$. For each vertex $u \in V$ the windings are updated $\mathcal{O}(|V| \cdot \deg(u))$ times, once per position and once per shifted parting. For one round, this results in $\mathcal{O}(|V||E|)$ winding changes taking time $\mathcal{O}(|V| \cdot |E|^2)$. □

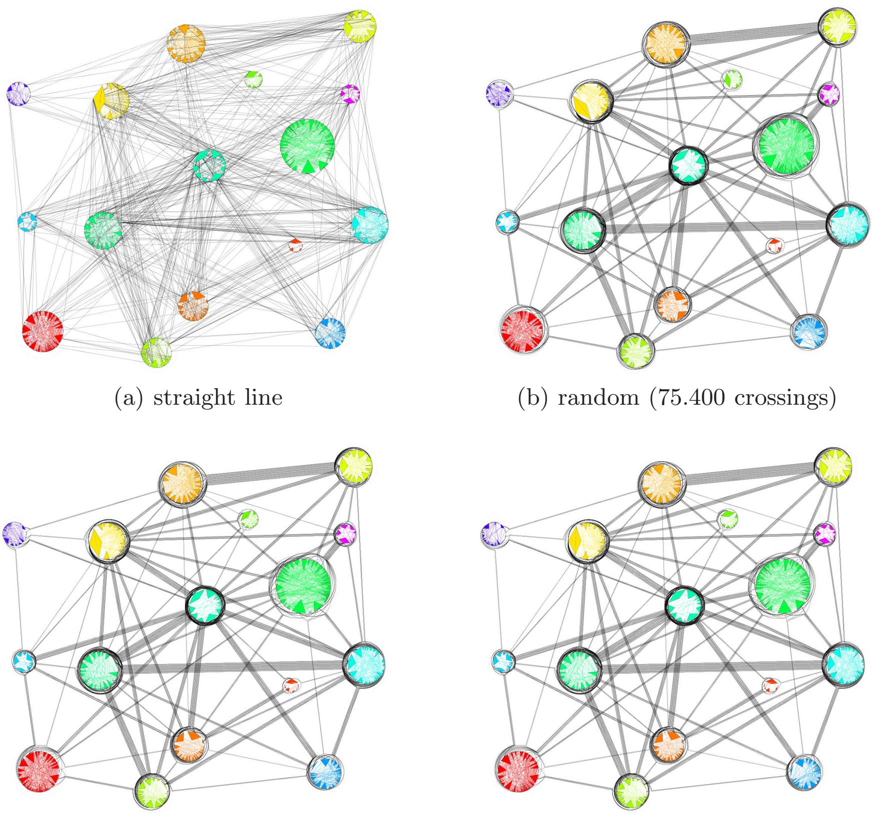
6 Application: Email Communication Network

The strength of a multi-circular layout is the coherent drawing of vertices and edges at the two levels of detail. It reveals structural properties of the macro graph and allows identification of micro level connections at the same time. The showcase for the benefits of our micro/macro layout is a email communication network of a department of the Universität Karlsruhe. The micro graph consists of 442 anonymized department members and 2,201 edges representing at least one email communication in the considered time frame of five weeks. At the macro level, a grouping into 16 institutes is given, resulting in 66 macro edges.

We start by inspecting drawings generated by a general force-directed approach similar to [9] and by multi-dimensional scaling (MDS) [6], see Figure 5. Both methods tend to place adjacent vertices near each other but ignore the additional grouping information. Therefore, it is not surprising that the drawings do not show a geometric clustering and the macro structure can not be identified. Moreover, it is difficult or even impossible to follow edges since they overlap each other.

More tailored for the drawing of graphs with additional vertex grouping are the layout used by Krebs [15], and the force-directed attempts to assign vertex positions by Six and Tollis [20] and Krempel [16]. All three methods place the vertices of each group on circles inside of separated geometric areas. While some efforts are made to find good vertex positions on the circles, edges are simply drawn as straight lines. Figure 6 (a) gives a prototypical example of this layout style. Although these methods feature a substantial progress compared to general layouts and macro vertices are clearly visible, there is no representation of macro edges and so the overall macro structure is still not identifiable.

Finally, we layouted the email network according to the micro/macro drawing convention. Its combinatorial descriptions allows for an enrichment with an



(a) straight line

(b) random (75.400 crossings)

(c) barycenter (68.300 crossings)

(d) sifting (57.400 crossings)

Fig. 6. Multi-circular layouts of the email network

analytical visualization of the vertices. In the Figures 1 and 6 the length of the circular arc a vertex covers is proportional to its share of the total inter-partition edges of this group. The height from its chord to the center of the circle reflects the fraction of present to possible intra-edges.

To investigate the effect of improved vertex orders and appropriate edge windings, we compare two variations of multi-circular layouts: shortest-path edge winding combined with random vertex placement and with barycenter vertex placement, see Figure 6. The macro structure of the graph is apparent at first sight. Since the placement of the vertex circles is the same as in Figure 6 (a), this improvement clearly follows from the grouping of micro edges. A closer look reveals the drawback of random placement: edges between different groups have to cover a long distance around the vertex circles and are hard to follow. Also a lot of edge crossings are generated both inside of the groups and in the area around the vertex placement circles. Assigning vertex positions according to the barycenter heuristic results in a clearly visible improvement and allows the differentiation of some of the micro edges. Using sifting improves the layout even further, resulting from a decrease of the number of crossings from more than 75.000 to 57.400 in the considered email network. The time for computing the layout of this quiet large graph is below half a minute.

7 Conclusion

We proposed a drawing convention for micro/macro graphs where micro-level elements are drawn on top of the elements of the coarse macro graph, so that the contribution of micro-level elements to macro-level structure becomes apparent. Since there is no need to place restrictions on the layout of the macro graph, we assumed it is given and focused on layouts of the micro graph. We presented a multi-circular layout model and investigated layout strategies based on crossing reduction techniques for it.

Backed by the visualizations of the email communication network computed by an initial implementation of our algorithms we claim that the grouping of micro-edges into macro-edges according to the micro/macro drawing convention exhibits benefits over layouts which group the vertices. Furthermore, since vertex orders and edge windings have a large effect on the readability of multi-circular layouts, it is justified to spend a larger effort to improve them.

A major benefit of the multi-circular layout is it combinatorial description since it allows the combination with other visualization techniques to highlight some graph properties or to further improve the visual appearance. A very interesting aspect would be the combination with Holten's [13] edge bundling technique.

References

1. Archambault, D., Munzner, T., Auber, D.: Topolayout: Multi-level graph layout by topological features. IEEE Trans. Visual. and Comp. Graphics 13(2), 305–317 (2007)
2. Bachmaier, C.: A radial adaptation of the sugiyama framework for visualizing hierarchical information. IEEE Trans. Visual. and Comp. Graphics 13(3), 585–594 (2007)
3. Balzer, M., Deussen, O.: Level-of-detail visualization of clustered graph layouts. In: APVIS 2007. Asia-Pacific Symposium on Visualisation 2007 (2007)
4. Baur, M., Brandes, U.: Crossing reduction in circular layouts. In: Hromkovič, J., Nagl, M., Westfechtel, B. (eds.) WG 2004. LNCS, vol. 3353, pp. 332–343. Springer, Heidelberg (2004)
5. Brandes, U., Erlebach, T. (eds.): Network Analysis. LNCS, vol. 3418. Springer, Heidelberg (2005)
6. Cox, T.F., Cox, M.A.A.: Multidimensional Scaling. In: Monographs on Statistics and Applied Probability, 2nd edn., Chapman & Hall/CRC (2001)
7. Duncan, C.A., Efrat, A., Kobourov, S.G., Wenk, C.: Drawing with fat edges. In: Mutzel, P., Jünger, M., Leipert, S. (eds.) GD 2001. LNCS, vol. 2265, pp. 162–177. Springer, Heidelberg (2002)
8. Eades, P., Kelly, D.: Heuristics for reducing crossings in 2-layered networks. Ars Combinatoria 21(A), 89–98 (1986)

9. Fruchterman, T.M.J., Reingold, E.M.: Graph drawing by force-directed placement. Software - Practice and Experience 21(11), 1129–1164 (1991)
10. Gansner, E.R., Koren, Y.: Improved circular layouts. In: Kaufmann, M., Wagner, D. (eds.) GD 2006. LNCS, vol. 4372, pp. 386–398. Springer, Heidelberg (2007)
11. Gansner, E.R., North, S.C.: Improved force-directed layouts. In: Whitesides, S.H. (ed.) GD 1998. LNCS, vol. 1547, pp. 364–373. Springer, Heidelberg (1999)
12. Harel, D., Koren, Y.: Drawing graphs with non-uniform vertices. In: AVI 2002. Proc. Work. Conf. on Advanced Visual Interfaces, pp. 157–166. ACM Press, New York (2002)
13. Holten, D.: Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. IEEE Trans. Visual. and Comp. Graphics 12(5), 741–748 (2006)
14. Kaufmann, M., Wiese, R.: Maintaining the mental map for circular drawings. In: Goodrich, M.T., Kobourov, S.G. (eds.) GD 2002. LNCS, vol. 2528, pp. 12–22. Springer, Heidelberg (2002)
15. Krebs, V.E.: Visualizing human networks. Release 1.0, pp. 1–25 (February 1996)
16. Krempel, L.: Visualisierung komplexer Strukturen. Grundlagen der Darstellung mehrdimensionaler Netzwerke. Campus (2005)
17. Masuda, S., Kashiwabara, T., Nakajima, K., Fujisawa, T.: On the $\mathcal{NP}$-completeness of a computer network layout problem. In: Proc. 20th IEEE Int. Symposium on Circuits and Systems 1987, pp. 292–295 (1987)
18. Matuszewski, C., Schönfeld, R., Molitor, P.: Using sifting for k-layer straightline crossing minimization. In: Kratochvíl, J. (ed.) GD 1999. LNCS, vol. 1731, pp. 217–224. Springer, Heidelberg (1999)
19. Rudell, R.: Dynamic variable ordering for ordered binary decision diagrams. In: Proc. IEEE/ACM Conf. Computer-Aided Design, pp. 42–47. IEEE Society, Los Alamitos (1993)
20. Six, J.M., Tollis, I.G.: A framework for user-grouped circular drawings. In: Liotta, G. (ed.) GD 2003. LNCS, vol. 2912, pp. 135–146. Springer, Heidelberg (2004)
21. Wang, X., Miyamoto, I.: Generating customized layouts. In: Brandenburg, F.J. (ed.) GD 1995. LNCS, vol. 1027, pp. 504–515. Springer, Heidelberg (1996)