Maximizing the Total Resolution of Graphs

Evmorfia N. Argyriou[1], Michael A. Bekos[1] and Antonios Symvonis[1]

School of Applied Mathematical & Physical Sciences,
National Technical University of Athens, Greece
{fargyriou,mikebekos,symvonis}@math.ntua.gr

Abstract. A major factor affecting the readability of a graph drawing is its resolution. In the graph drawing literature, the resolution of a drawing is either measured based on the angles formed by consecutive edges incident to a common node (angular resolution) or by the angles formed at edge crossings (crossing resolution). In this paper, we evaluate both by introducing the notion of "total resolution", that is, the minimum of the angular and crossing resolution. To the best of our knowledge, this is the first time where the problem of maximizing the total resolution of a drawing is studied.
The main contribution of the paper consists of drawings of asymptotically optimal total resolution for complete graphs (circular drawings) and for complete bipartite graphs (2-layered drawings). In addition, we present and experimentally evaluate a force-directed based algorithm that constructs drawings of large total resolution.

Date: *September 10, 2010.*

1 Introduction

Graphs are widely used to depict relations between objects. There exist several criteria that have been used to judge the quality of a graph drawing [3,16]. From a human point of view, it is necessary to obtain drawings that are easy-to-read, i.e., they should nicely convey the structure of the objects and their relationships. From an algorithmic point of view, the quality of a drawing is usually evaluated by some objective function and the main task is to determine a drawing that minimizes or maximizes the specific objective function. Various such functions have been studied by the graph drawing community, among them, the number of crossings among pairs of edges, the number of edge bends, the maximum edge length, the total area occupied by the drawing and so on.

Over the last few decades, much research effort has been devoted to the problem of reducing the number of crossings. This is reasonable, since it is commonly accepted that edge crossings may negatively affect the quality of a drawing. Towards this direction, there also exist eye-tracking experiments that confirm the negative impact of edge crossings on the human understanding of a graph drawing [20,21,22]. However, the computational complexity of the edge crossing minimization problem, which is $\mathcal{NP}$-complete in general [11], implies that the computation of high-quality drawings of dense graph is difficult to achieve.

Apart from the edge crossings, another undesired property that may negatively influence the readability of a drawing is the presence of edges that are too close to each other, especially if these edges are adjacent. Thus, maximizing the angles among incident edges becomes an important aesthetic criterion, since there is some correlation between the involved angles and the visual distinctiveness of the edges.

Motivated by the cognitive experiments by Huang et al. [14,15] that indicate that the negative impact of an edge crossing is eliminated in the case where the crossing angle is greater than 70 degrees, we study a new graph drawing scenario in which both *angular* and *crossing resolution*[1] are taken into account in order to produce a straight-line drawing of a given graph. To the best of our knowledge, this is the first attempt, where both angular and crossing resolution are combined to produce drawings. We prove that the classes of complete and complete bipartite graphs admit drawings that asymptotically maximize the minimum of the angular and crossing resolution (Section 3). We also present a more practical, force-directed based algorithm that constructs drawings of large angular and crossing resolution (Section 4).

1.1 Previous Work

Formann et al. [9] were the first to study the angular resolution of straight-line drawings. They proved that deciding whether a graph of maximum degree d admits a drawing of angular resolution $\frac{2\pi}{d}$ (i.e., the obvious upper bound) is $\mathcal{NP}$-hard. They also proved that several types of graphs of maximum degree d have angular resolution $\Theta(\frac{1}{d})$. Malitz and Papakostas [18] proved that any planar graph of maximum degree d, admits a planar straight-line drawing with angular resolution $\Omega(\frac{1}{7^d})$. Garg and Tamassia [12] showed a continuous tradeoff between the area and the angular resolution of planar straight-line drawings. Gutwenger and Mutzel [13] gave a linear time and space algorithm that constructs a planar polyline grid drawing of a connected planar graph with n vertices and maximum degree d on a $(2n-5)\times(\frac{3}{2}n-\frac{7}{2})$ grid with at most $5n-15$ bends and minimum angle greater than $\frac{2}{d}$. Bodlaender and Tel [2] showed that planar graphs with angular resolution at least $\frac{\pi}{2}$ are rectilinear. Recently, Lin and Yen [17] presented a force-directed method based on edge-edge repulsion that leads to drawings with high angular resolution. In their work, pairs of edges incident to a common node are modeled as charged springs, that repel each other.

A graph is called *right angle crossing* (or *RAC* for short) graph if it admits a polyline drawing in which every pair of crossing edges intersects at right angle. Didimo et al. [6] showed that any straight-line RAC drawing with n nodes has at most $4n-10$ edges. Angelini et al. [1] showed that there are acyclic planar digraphs not admitting straight-line upward RAC drawings and that the corresponding decision problem is $\mathcal{NP}$-hard. They also constructed digraphs whose straight-line upward RAC drawings require exponential area. Di Giacomo et al. [5] presented tradeoffs between the crossing resolution, the maximum number of bends per edges and the area. Dujmovic et al. [7] studied α *Angle Crossing* (or αAC for short) graphs, that generalize the RAC graphs. A graph is called αAC if it admits a polyline drawing in which the smallest angle formed by an edge crossing is at least α. For this class of graphs, they proved upper and lower bounds for the number of edges.

Force-directed methods are commonly used for drawing graphs [8,10]. In such a framework, a graph is treated as a physical system with forces acting on it. Then, a good configuration or drawing can be obtained from an equilibrium state of the

[1] The term *angular resolution* denotes the smallest angle formed by two adjacent edges incident to a common node, whereas the term *crossing resolution* refers to the smallest angle formed by a pair of crossing edges.

system. An overview of force-directed methods and their variations can be found in the graph drawing books [3,16].

2 Preliminaries and Notation

Let $G = (V, E)$ be an undirected graph. Given a drawing $\Gamma(G)$ of G, we denote by $p_u = (x_u, y_u)$ the position of node $u \in V$ on the plane. The unit length vector from p_u to p_v is denoted, by $\overrightarrow{p_u p_v}$, where $u, v \in V$. The degree of node $u \in V$ is denoted by $d(u)$. Let also $d(G) = \max_{u \in V} d(u)$ be the degree of the graph.

Given a pair of points $q_1, q_2 \in \mathbb{R}^2$, with a slight abuse of notation, we denote by $||q_1 - q_2||$ the Euclidean distance between q_1 and q_2. We refer to the line segment defined by q_1 and q_2 as $\overline{q_1 q_2}$.

Let $\overrightarrow{\alpha}$ and $\overrightarrow{\gamma}$ be two vectors. The vector which bisects the angle between $\overrightarrow{\alpha}$ and $\overrightarrow{\gamma}$ is $\frac{\overrightarrow{\alpha}}{||\overrightarrow{\alpha}||} + \frac{\overrightarrow{\gamma}}{||\overrightarrow{\gamma}||}$. We denote by $\texttt{Bsc}(\overrightarrow{\alpha}, \overrightarrow{\gamma})$ the corresponding unit length vector. Given a vector $\overrightarrow{\beta}$, we refer to the unit length vector which is perpendicular to $\overrightarrow{\beta}$ and precedes it in the clockwise direction, as $\texttt{Perp}(\overrightarrow{\beta})$. Some of our proofs use the following elementary geometric properties:

$$\tan(\omega_1 - \omega_2) = \frac{\tan\omega_1 - \tan\omega_2}{1 + \tan\omega_1 \cdot \tan\omega_2} \quad (1) \qquad \tan(\omega/2) = \frac{\sin\omega}{1 + \cos\omega} \quad (2)$$

$$\omega \in (0, \frac{\pi}{2}) \Rightarrow \tan\omega > \omega \quad (3)$$

3 Drawings with Optimal Total Resolution for Complete and Complete Bipartite Graphs

In this section, we define the total resolution of a drawing and we present drawings of asymptotically optimal total resolution for complete graphs (circular drawings) and complete bipartite graphs (2-layered drawings).

Definition 1. *The* total resolution *of a drawing is defined as the minimum of its angular and crossing resolution.*

We first consider the case of complete graphs. Let $K_n = (V, E)$ be a complete graph, where $V = \{u_0, u_1, \ldots, u_{n-1}\}$ and $E = V \times V$. Our aim is to construct a circular drawing of K_n of maximum total resolution. Our approach is constructive and common when dealing with complete graphs. A similar one has been given by Formann et al. [9] for obtaining optimal drawings of complete graphs, in terms of angular resolution. Consider a circle $\mathcal{C}$ of radius $r_c > 0$ centered at $(0, 0)$ and circumscribe a regular n-polygon $\mathcal{Q}$ on $\mathcal{C}$. In our construction, the nodes of K_n coincide with the vertices of $\mathcal{Q}$. W.l.o.g., we further assume that $u_1, u_2, \ldots, u_n$ appear in this order in the counter-clockwise direction around $(0, 0)$, as illustrated in Fig.1a.

Theorem 1. *A complete graph K_n admits a drawing of total resolution $\Theta(\frac{1}{n})$.*

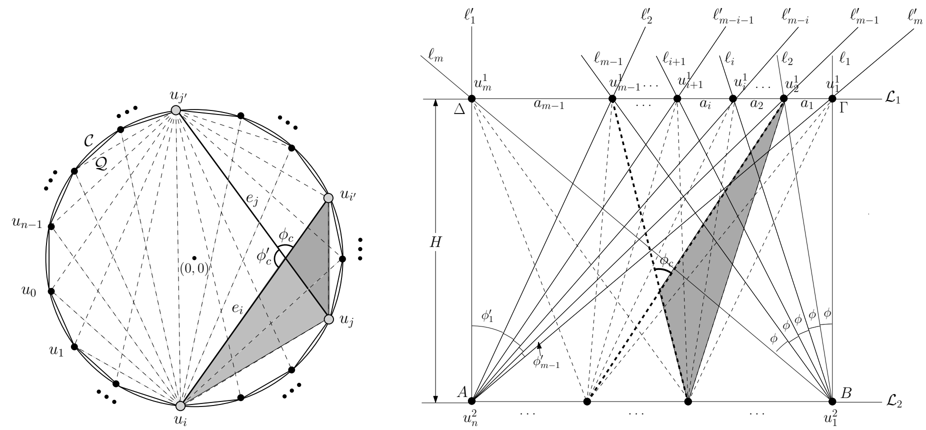
Proof. We prove that the angular resolution of the presented drawing of K_n is $\frac{\pi}{n}$, whereas its crossing resolution is $\frac{2\pi}{n}$. First, observe that the arc of circle $\mathcal{C}$ that connects two consecutive nodes u_i and $u_{(i+1) mod n}$ is equal to $\frac{2\pi}{n}$, for each $i = 0, 1, \ldots, n-1$. Therefore, the angular resolution of the drawing is $\frac{\pi}{n}$, as desired. Let now $e_i = (u_i, u_{i'})$ and $e_j = (u_j, u_{j'})$ be two crossing edges. Without loss of generality, we assume that $i < j < i' < j'$, as in Fig.1a. The crossing of e_i and e_j defines two angles ϕ_c and ϕ'_c such that $\phi_c + \phi'_c = \pi$. In Fig.1a, ϕ_c is exterior to the triangle formed by the crossing of e_i and e_j and the nodes u_j and $u_{i'}$ (refer to the dark-gray triangle of Fig.1a). Therefore: $\phi_c = (j'-i')\frac{\pi}{n} + (j-i)\frac{\pi}{n}$. Similarly, $\phi'_c = (i'-j)\frac{\pi}{n} + (n-(j'-i))\frac{\pi}{n}$. In the case, where $j = (i+1) \mod n$ and $j' = (i'+1) \mod n$ (i.e., the nodes u_i ($u_{i'}$, resp.) and u_j ($u_{j'}$, resp.) are consecutive), the angle ϕ_c receives its minimum value, which is equal to $\frac{2\pi}{n}$. Similarly, we can prove that the minimum value of ϕ'_c is also $\frac{2\pi}{n}$. This establishes that the crossing resolution is $\frac{2\pi}{n}$. □

We now proceed to consider the class of complete bipartite graphs. Since an n-vertex complete bipartite graph is a subgraph of a n-vertex complete graph, the bound of the total resolution of a complete bipartite graph can be implied by the bound of the complete graph. However, if the nodes of the graph must have integer coordinates, i.e., we restrict ourselves on grid drawings, few results are known regarding the area needed of such a drawing. An upper bound of $O(n^3)$ area can be implied by [2]. This motivates us to separately study the class of complete bipartite graph, since we can drastically improve this bound. Note that the tradeoff between resolution and area has been studied by the graph drawing community, in the past. Malitz and Papakostas [18] showed there exist graphs that always require exponential area for straight-line embeddings maintaining good angular resolution. The claim remains true, if circular arc edges are used instead of straight-line [4]. More recently, Angelini et al. [1] constructively showed that there exists graphs whose straight-line upward RAC drawings require exponential area.

Again, we follow a constructive approach. First, we consider a square $\mathcal{R} = AB\Gamma\Delta$ where its top and bottom sides coincide with $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively (see Fig.1b). Let H be the height (and width) of $\mathcal{R}$. According to our approach, the nodes of V_1 (V_2, resp.) reside along side $\Gamma\Delta$ (AB, resp.) of $\mathcal{R}$. In order to specify the exact positions of the nodes $u^1_1, u^1_2, \ldots, u^1_m$ along side $\Gamma\Delta$, we first construct a bundle of m semi-lines, say $\ell_1, \ldots, \ell_m$, each of which emanates from vertex B and crosses side $\Gamma\Delta$ of $\mathcal{R}$, so that the angle formed by $B\Gamma$ and semi-line ℓ_i equals to $\frac{(i-1)\cdot\widehat{\Delta B\Gamma}}{m-1}$, for each $i = 1, \ldots, m$. These semi-lines split angle $\widehat{\Delta B\Gamma}$ into $m-1$ angles, each of which is equal to $\frac{\pi}{4\cdot(m-1)}$, since $\widehat{\Delta B\Gamma} = \pi/4$. Say $\phi = \frac{\pi}{4\cdot(m-1)}$. Then, we place node u^1_i at the intersection of semi-line l_i and $\Gamma\Delta$, for each $i = 1, \ldots, m$ (see Fig.1b). In order to simplify the description of our approach, we denote by a_i the horizontal distance between two consecutive nodes u^1_i and u^1_{i+1}, $i = 1, \ldots, m-1$.

We proceed by defining an additional bundle of m semi-lines, say $\ell'_1, \ldots, \ell'_m$, that emanate from vertex A. More precisely, semi-line l'_i emanates from vertex A and passes through the intersection of l_{m-i} and $\Gamma\Delta$ (i.e., node u^1_{m-i}), for each $i = 1, \ldots, m$ (see Fig.1b). Let ϕ'_i be the angle formed by two consecutive semi-lines l'_i and l'_{i+1}, for each $i = 1, \ldots, m-1$.

So far, we have managed to fix the position of the nodes of V_1 only (along side $\Gamma\Delta$ of $\mathcal{R}$). Symmetrically, we define the position of the nodes of V_2 along side AB of $\mathcal{R}$. This only involves two additional bundles of semi-lines emanating from vertices



(a) A circular drawing of K_n.

(b) A 2-layered drawing of $K_{m,n}$.

Fig. 1: Illustrations of our constructions

Γ and Δ. We now proceed to investigate some geometric properties of the proposed construction.

Lemma 1. *For each $i = 1, 2, \ldots, m-1$, it holds that $a_{i-1} < a_i$.*

Proof. By induction. For the base of the induction, we have to show that $a_1 < a_2$. First observe that $a_1 = H \tan\phi$ and $a_1 + a_2 = H \tan 2\phi$. Therefore:

$$a_2 = H(\tan 2\phi - \tan\phi) \overset{(1)}{=} a_1 \cdot (1 + \tan 2\phi \cdot \tan\phi)$$

However, both $\tan\phi$ and $\tan 2\phi$ are greater than zero, which immediately implies that $a_1 < a_2$. For the induction hypothesis, we assume that $\forall\, k, k < m-1$ it holds that $a_{k-1} < a_k$ and we should prove that $a_k < a_{k+1}$. Obviously, $a_1 + \ldots + a_k = H \tan k\phi$. Based on Equation 1 and similarly to the base of the induction, we have:

- $a_{k+1} = H \tan\phi \cdot (1 + \tan{(k+1)\phi} \cdot \tan k\phi)$
- $a_k \quad = H \tan\phi \cdot (1 + \tan{(k-1)\phi} \cdot \tan k\phi)$

In order to complete the proof, observe that $(k+1)\phi > (k-1)\phi$. □

Lemma 2. *For each $i = 1, 2, \ldots, m-1$, it holds that $\phi'_{i-1} > \phi'_i$.*

Proof. By induction. For the base of the induction, we have to prove that $\phi'_1 > \phi'_2$ or equivalently that $\tan\phi'_1 > \tan\phi'_2$. It holds that $\tan\phi'_1 = a_{m-1}/H$ and $\tan{(\phi'_1 + \phi'_2)} = (a_{m-1} + a_{m-2})/H$. By combining these relationships with Equation 1 we have that $\tan\phi'_2 = \frac{H a_{m-1}}{H^2 + a^2_{m-1} + a_{m-1}a_{m-2}}$. Therefore:

$$\tan\phi'_1 > \tan\phi'_2 \Leftrightarrow H^2(a_{m-1} - a_{m-2}) + a^3_{m-1} + a^2_{m-1}a_{m-2} > 0,$$

which trivially holds due to Lemma 1. For the induction hypothesis, we assume that $\forall\, k, k < m-1$ it holds that $\phi'_{k-1} > \phi'_k$ and we have to show that $\phi'_k > \phi'_{k+1}$. Observe that:

– $\tan \phi'_k = \frac{\tan(\phi_1+\ldots+\phi'_k)-\tan(\phi'_1+\ldots+\phi'_{k-1})}{1+\tan(\phi'_1+\ldots+\phi'_k)\cdot\tan(\phi'_1+\ldots+\phi'_{k-1})} = \frac{Ha_{m-k}}{H^2+(a_{m-1}+\ldots+a_{m-k})(a_{m-1}+\ldots+a_{m-k+1})}$
– $\tan(\phi'_{k+1}) = \cdots = \frac{Ha_{m-(k+1)}}{H^2+(a_{m-1}+\ldots+a_{m-(k+1)})(a_{m-1}+\ldots+a_{m-k})}$

By Lemma 1 we have that $(a_{m-1} + \ldots + a_{m-(k-1)}) > (a_{m-1} + \ldots + a_{m-(k+1)})$ and $H \cdot a_{m-k} > H \cdot a_{m-(k+1)}$. Therefore, $\tan \phi'_k > \tan \phi'_{k+1}$. □

Lemma 3. *Angle ϕ'_{m-1} is the smallest angle among all the angles formed in the drawing.*

Proof. From Lemma 2, it follows that angle ϕ'_{m-1} is the smallest angle among all ϕ'_i, $i = 1, \ldots, m-1$. Additionally, it is not difficult to see that angle ϕ'_{m-1} is larger (but remains the smallest among all ϕ'_i, $i = 1, \ldots, m-1$), if the endpoint of the bundle (i.e., node u_n^2) moves to any internal point (i.e., node u_i^2) of side AB (see Fig.1b). Therefore, ϕ'_{m-1} is the smallest angle among all the angles formed by pairs of consecutive edges incident to any node of V_2. Since $m \geq n$, the same holds for the nodes of V_1. Therefore, ϕ'_{m-1} defines the angular resolution of the drawing.

Consider now two crossing edges (refer to the bold, crossing dashed-edges of Fig.1b). Their crossing defines (a) a pair of angles that are smaller than 90^o and (b) another pair of angles that are larger than 90^o. Obviously, only the acute angles participate in the computation of the crossing resolution (see angle ϕ_c in Fig.1b). However, in a complete bipartite graph the acute angles are always exterior to a triangle having two of its vertices on V_1 and V_2, respectively (refer to the gray-colored triangle of Fig.1b). Therefore, the crossing resolution is always greater than the angular resolution, as desired. □

Lemma 4. *It holds that $\phi'_{m-1} \geq \frac{\phi}{2}$.*

Proof. We equivalently prove that $\tan \phi'_{m-1} > \tan \frac{\phi}{2}$. Using Equation 2, we have that $\tan \frac{\phi}{2} < \frac{a_1}{2H}$. Therefore:

$$\begin{aligned}
\tan \phi'_{m-1} > \tan \frac{\phi}{2} &\iff \frac{\frac{a_1}{H}}{1 + \frac{a_1+\ldots+a_{m-1}}{H} \cdot \frac{a_2+\ldots+a_{m-1}}{H}} > \frac{a_1}{2 \cdot H} \\
&\iff H^2 > (a_1 + \ldots + a_{m-1})(a_2 + \ldots + a_{m-1}) \\
&\iff H \cdot a_1 > 0
\end{aligned}$$

which obviously holds. □

Theorem 2. *A complete bipartite graph $K_{m,n}$ admits a 2-layered drawing of total resolution $\Theta(\frac{1}{\max\{m,n\}})$.*

Proof. Immediately follows from Lemmata 3 and 4. □

Consider now the case where the nodes of the graph must have integer coordinates, i.e., we restrict ourselves on grid drawings. An interesting problem that arises in this case is the estimation of the total area occupied by the produced drawing. We will describe how we can modify the positions of the nodes produced by our algorithm in order to obey the grid constraints. Assume without loss of generality that $\mathcal{L}_1$ and $\mathcal{L}_2$ are two horizontal lines, so that $\mathcal{L}_2$ coincides with y-axis and the drawing produced by our algorithm has $a_1 = 1$. Then, we can express the height of drawing $\Gamma(K_{m,n})$ as a function of ϕ, as follows:

$$a_1 = 1 \Longleftrightarrow \tan\phi \cdot H = 1 \Longleftrightarrow H = 1/\tan\phi$$

Note that this drawing does not obey the grid constraints. To achieve this, we move the horizontal line $\mathcal{L}_1$ to the horizontal grid line immediately above it and each node of both V_1 and V_2 to the rightmost grid-point to its left. In this manner, we obtain a new drawing $\Gamma'(K_{m,n})$, which is grid as desired. By Lemma 1, it follows that there are no two nodes sharing the same grid point, since a_1 is slightly greater than one grid unit. Since neither horizontal line $\mathcal{L}_1$ nor any node of $K_{m,n}$ moves more than one unit of length, the total resolution of $\Gamma'(K_{m,n})$ is not asymptotically affected, and, in addition the height of the drawing is not significantly greater (i.e., asymptotically it remains the same). Based on the above, the area is bounded by $\cot^2\phi$ or equivalently by $1/\tan^2\phi$. By Equation 3, this is further bounded by $1/\phi^2$. By Theorem 2, it holds that $\phi = O(1/\max\{m,n\})$. Therefore, the total area occupied by the drawing is $O(\max\{m^2,n^2\})$. The following theorem summarizes this result.

Theorem 3. *A complete bipartite graph $K_{m,n}$ admits a 2-layered grid drawing of $\Theta(\frac{1}{\max\{m,n\}})$ total resolution and $O(\max\{m^2,n^2\})$ area.*

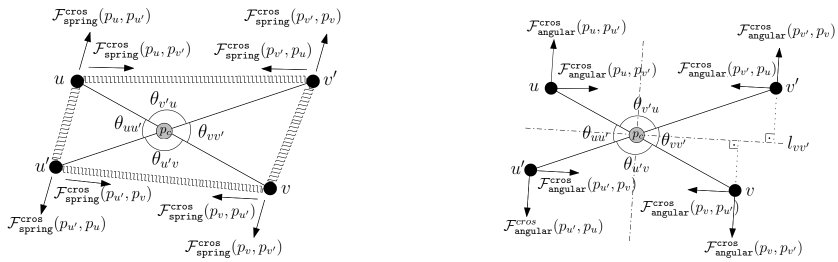
4 A Force Directed Algorithm

We present a force-directed algorithm that given a reasonably nice initial drawing, probably produced by a classical force-directed algorithm, results in a drawing of high total resolution. The algorithm reinforces the classical force-directed algorithm of Eades [8] with some additional forces exerted to the nodes of the graph. More precisely, these additional forces involve springs and some extra attractive or repulsive forces on nodes with degree greater than one and on end-nodes of edges that are involved in an edge crossing. This aims to ensure that the angles between incident edges and the angles formed by pairs of crossing edges will be as large as possible. The classical force-directed algorithm of Eades [8] models the nodes of the graph as electrically charged particles that repel each other, and its edges by springs in order to attract adjacent nodes. In our approach, we use only the attractive forces of the force-directed algorithm of Eades (denoted by $\mathcal{F}_{\texttt{spring}}$), which follow the formula:

$$\mathcal{F}_{\texttt{spring}}(p_u,p_v) = C_{\texttt{spring}} \cdot \log\frac{||p_u - p_v||}{\ell_{\texttt{spring}}} \cdot \overrightarrow{p_u p_v},\ (u,v) \in E$$

where $C_{\texttt{spring}}$ and $\ell_{\texttt{spring}}$ capture the stiffness and the natural length of the springs, respectively. Recall that $\overrightarrow{p_u p_v}$ denotes the unit length vector from p_u to p_v.

We first describe our approach for the case where two edges, say $e = (u,v)$ and $e' = (u',v')$, are involved in a crossing. Let p_c be their intersection point. W.l.o.g., we assume that u and u' are to the left of v and v', respectively, $y_{u'} < y_u$ and $y_v < y_{v'}$, as in Fig.2. Let $\theta_{vv'}$ be the angle formed by the line segments $\overline{p_c p_v}$ and $\overline{p_c p_{v'}}$ in counter-clockwise order around u from $\overline{p_c p_v}$ to $\overline{p_c p_{v'}}$. In order to avoid confusion, we assume that $\theta_{vv'} = \theta_{v'v}$, i.e., we abuse the counter-clockwise measurement of the angles that would result in $\theta_{vv'} = 2\pi - \theta_{v'v}$. Similarly, we define the remaining angles of Fig.2. Obviously, $\theta_{vv'} + \theta_{v'u} = \pi$. Ideally, we would like $\theta_{vv'} = \theta_{v'u} = \frac{\pi}{2}$, i.e., e and e' form



(a) **(b)**

Fig. 2: Forces applied on nodes in order to maximize the crossing resolution. (a) Springs on nodes involved in crossing. (b) Repelling or attractive forces based on the angles.

a right angle crossing. As we will shortly see, the magnitude of the forces that we apply on the nodes u, u', v and v' depends on (a) the angles $\theta_{vv'}$ and $\theta_{v'u}$ and (b) the lengths of the line segments $\overline{p_c p_u}$, $\overline{p_c p_{u'}}$, $\overline{p_c p_v}$ and $\overline{p_c p_{v'}}$.

The physical model that describes our approach is illustrated in Fig.2. Initially, for each pair of crossing edges at point p_c, we place springs connecting consecutive nodes in the counter-clockwise order around p_c, as in Fig.2a. The magnitude of the forces due to these springs should capture our preference for right angles. Consider the spring connecting v and v'. The remaining ones are treated symmetrically. We set the natural length, say $\ell^{vv'}_{\texttt{spring}}$, of the spring connecting the nodes v and v' to be $\sqrt{||p_c - p_v||^2 + ||p_c - p'_v||^2}$. This quantity corresponds to the length of the line segment that connects v and v' in the optimal case where $\theta_{vv'} = \frac{\pi}{2}$. So, in an equilibrium state of this model on a graph consisting only of e and e', e and e' will form a right angle. Concluding, the force on v due to the spring of v' is defined as follows:

$$\mathcal{F}^{\texttt{cros}}_{\texttt{spring}}(p_v, p_{v'}) = C^{\texttt{cros}}_{\texttt{spring}} \cdot \log \frac{||p_v - p_{v'}||}{\ell^{vv'}_{\texttt{spring}}} \cdot \overrightarrow{p_v p_{v'}}$$

The remaining forces of Fig.2a are defined similarly. Note that in the formula above, the constant $C^{\texttt{cros}}_{\texttt{spring}}$ is used to control the stiffness of the springs.

Our preference for right angle crossings can be also captured using the angles $\theta_{vv'}$ and $\theta_{v'u}$ (see Fig.2b). As in the previous case, we restrict our description on the angle formed by the line segments $\overline{p_c p_v}$ and $\overline{p_c p_{v'}}$. Ideally, we would like to exert forces on the nodes v and v' such that: (i) when $\theta_{vv'} \to 0$, the magnitude of the force is very large (in order to repel v and v'), and (ii) when $\theta_{vv'} \to \frac{\pi}{2}$, the magnitude of the force is very small. A function, say $f : \mathbb{R} \to \mathbb{R}$, which captures this property is: $f(\theta) = \frac{|\frac{\pi}{2} - \theta|}{\theta}$. Having specified the magnitude of the forces, we set the direction of the force on v (due to v') to be perpendicular to the line that bisects the angle $\theta_{vv'}$ (refer to the dash-dotted line $l_{vv'}$ of Fig.2b), or equivalently parallel to the unit length vector $\texttt{Perp}(\texttt{Bsc}(\overrightarrow{p_c p_v}, \overrightarrow{p_c p_{v'}}))$. Recall that `Perp` and `Bsc` refer to the perpendicular and bisector vectors, respectively (see Section 2). It is clear that if $\theta_{vv'} < \frac{\pi}{2}$, the forces on v and v' should be repulsive (in order to enlarge the angle between them), otherwise attractive. This can be captured by the *sign* function. We conclude with the following formula which expresses the force on v due to v'.

$$\mathcal{F}^{\texttt{cros}}_{\texttt{angle}}(p_v, p_{v'}) = C^{\texttt{cros}}_{\texttt{angle}} \cdot sign(\theta_{vv'} - \frac{\pi}{2}) \cdot f(\theta_{vv'}) \cdot \texttt{Perp}(\texttt{Bsc}(\overrightarrow{p_c p_v}, \overrightarrow{p_c p_{v'}}))$$

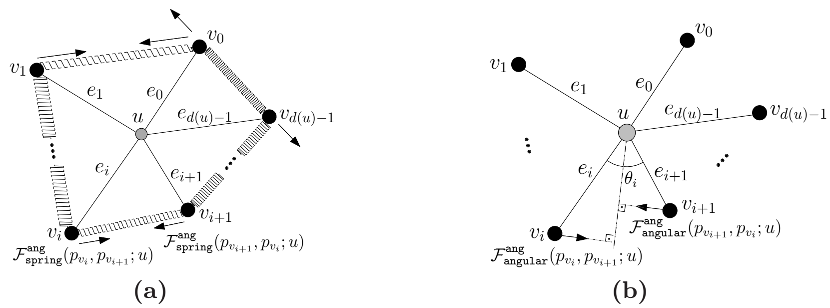


Fig. 3: Forces applied on nodes in order to maximize the angular resolution. (a) Springs on consecutive edges around u. (b) Repelling or attractive forces based on the angles.

where constant $C^{\mathtt{cros}}_{\mathtt{angle}}$ controls the strength of the force. Similarly, we define the remaining forces of Fig.2b.

Consider now a node u that is incident to $d(u)$ edges, say $e_0 = (u, v_0)$, $e_1 = (u, v_1)$, $\ldots$, $e_{d(u)-1} = (u, v_{d(u)-1})$. We assume that $e_0, e_1, \ldots, e_{d(u)-1}$ are consecutive in the counter-clockwise order around u in the drawing of the graph (see Fig.3a). Similarly to the case of two crossing edges, we proceed to connect the endpoints of consecutive edges around u by springs, as in Fig.3a. In this case, the natural length of each spring, should capture our preference for angles equal to $\frac{2\pi}{d(u)}$. In order to achieve this, we proceed as follows: For each $i = 0, 1, \ldots, d(u)-1$, we set the natural length, say $l^i_{\mathtt{spring}}$, of the spring connecting v_i with $v_{(i+1)mod(d(u))}$, to be:

$$\ell^i_{\mathtt{spring}} = \sqrt{||e_i||^2 + ||e_{(i+1)mod(d(u))}||^2 - 2 \cdot ||e_i|| \cdot ||e_{(i+1)mod(d(u))}|| \cdot \cos\left(2\pi/d(u)\right)}$$

where $||e||$ is used to denote the length of the edge $e \in E$ in the drawing of the graph. The quantity $\ell^i_{\mathtt{spring}}$ corresponds to the length of the line segment that connects v_i with $v_{(i+1)mod(d(u))}$ in the optimal case where the angle formed by e_i and $e_{(i+1)mod(d(u))}$ is $\frac{2\pi}{d(u)}$. Therefore, the spring forces between consecutive edges should follow the formula:

$$\mathcal{F}^{\mathtt{angular}}_{\mathtt{spring}}(p_{v_i}, p_{v_{(i+1)mod(d(u))}}; u) = C^{\mathtt{angular}}_{\mathtt{spring}} \cdot \log \frac{||p_{v_i} - p_{v_{(i+1)mod(d(u))}}||}{\ell^i_{\mathtt{spring}}} \cdot \overrightarrow{p_{v_i} p_{v_{(i+1)mod(d(u))}}}$$

where the quantity $C^{\mathtt{angular}}_{\mathtt{spring}}$ is a constant which captures the stiffness of the spring.

Let now θ_i be the angle formed by e_i and $e_{(i+1)mod(d(u))}$, measured in counter-clockwise direction from e_i to $e_{(i+1)mod(d(u))}$, $i = 0, 1, \ldots, d(u) - 1$. Similarly to the case of two crossing edges, we exert forces on v_i and $v_{(i+1)mod(d(u))}$ perpendicular to the bisector of θ_i, as illustrated in Fig.3b. However, in this case we need a magnitude function such that: (i) when $\theta_i \to 0$, the magnitude of the force is very large (in order to repel v_i and $v_{(i+1)mod(d(u))}$), and (ii) when $\theta_i \to \frac{2\pi}{d(u)}$, the magnitude of the force is very small. Such a function, say $g : \mathbb{R} \times V \to \mathbb{R}$, is: $g(\theta; u) = \frac{|\frac{2\pi}{d(u)} - \theta|}{\theta}$. Having fully specified the forces applied on the endpoints of consecutive edges and their directions, we are now ready to provide the exact formulas that the forces follow:

$$\mathcal{F}^{\texttt{angular}}_{\texttt{angle}}(p_{v_i}, p_{v_{(i+1)mod(d(u))}}; u) = C^{\texttt{angular}}_{\texttt{angle}} \cdot sign(\theta_i - \frac{2\pi}{d(u)}) \cdot g(\theta_i; u) \cdot$$
$$\texttt{Perp}(\texttt{Bsc}(\overrightarrow{p_u p_{v_i}}, \overrightarrow{p_u p_{v_{(i+1)mod(d(u))}}}))$$

where $C^{\texttt{angular}}_{\texttt{angle}}$ is a constant to control the strength of the force. In the work of Lin and Yen [17], the above technique which applies large repelling forces perpendicular to the bisector of the angle, when the angle is small, is referred to as edge-edge repulsion. However, in their work, they use different metric to control the magnitude function. Of course, we could also use their metric, but we prefer the one reported above in order to maintain a uniform approach in both crossing and angular cases. In Section 4.1, we provide an experimental comparison of these techniques.

Note that by setting zero values to the constants $C^{\texttt{cros}}_{\texttt{spring}}$, $C^{\texttt{cros}}_{\texttt{angle}}$ or $C^{\texttt{angular}}_{\texttt{spring}}$, $C^{\texttt{angular}}_{\texttt{angle}}$, our algorithm can be configured to maximize the angular, or the crossing resolution only, respectively.

On each iteration, our algorithm computes three types of forces. Computing the attractive forces of the classical force-directed model among pairs of adjacent nodes of the graph requires $O(E)$ time per iteration. The computation of the forces due to the edge crossings needs $O(E^2)$ time, assuming a straight forward algorithm that in $O(E^2)$ time reports all pairwise crossing edges. Finally, the computation of the forces due to the angles between consecutive edges can be done in $O(E + Vd(G)\log d(G))$ time per iteration, where $d(G)$ denotes the degree of the graph, since we first sort the incident edges of each node of the graph in cyclic order. Summarizing the above, each iteration of our algorithm takes $O(E^2 + Vd(G)\log d(G))$ time.

The time complexity can be improved using standard techniques from computational geometry [19]. If K is the number of pairwise-crossing edges, then the K intersections can be reported in $O(K + E\log^2 E/\log\log E)$ time [19, pp.277], which leads to a total complexity $O(K + E\log^2 E/\log\log E + Vd(G)\log d(G))$ per iteration.

4.1 Experimental Results

In this section, we present the results of the experimental evaluation of our algorithm. Apart from our algorithm, we have implemented the force directed algorithms of Eades [8] and Lin and Yen [17]. The implementations are in Java using the yFiles library (http://www.yworks.com). The experiment was performed on a Linux machine with 2.00 GHz CPU and 2GB RAM using the Rome graphs (a collection of around 11.500 graphs) obtained from graphdrawing.org. Fig.4, illustrates a drawing of a Rome graph with 99 nodes and 135 edges produced by our force directed algorithm.

The experiment was performed as follows. First, each Rome graph was laid out using the SmartOrganic layouter of yFiles. This layout was the input layout for all three algorithms, in order to speed up the experiment and overcome problems associated with local minimal traps especially in large graphs. If both the angular and the crossing resolution between two consecutive iterations of each algorithm were not improved more that 0.001 degrees, we assumed that the algorithm has converged and we did not proceed any more. The maximum number of iterations that an algorithm could perform in order to converge was 100.000. We note that the termination condition is quite strict and demands a large number of iterations. Our algorithm is evaluated

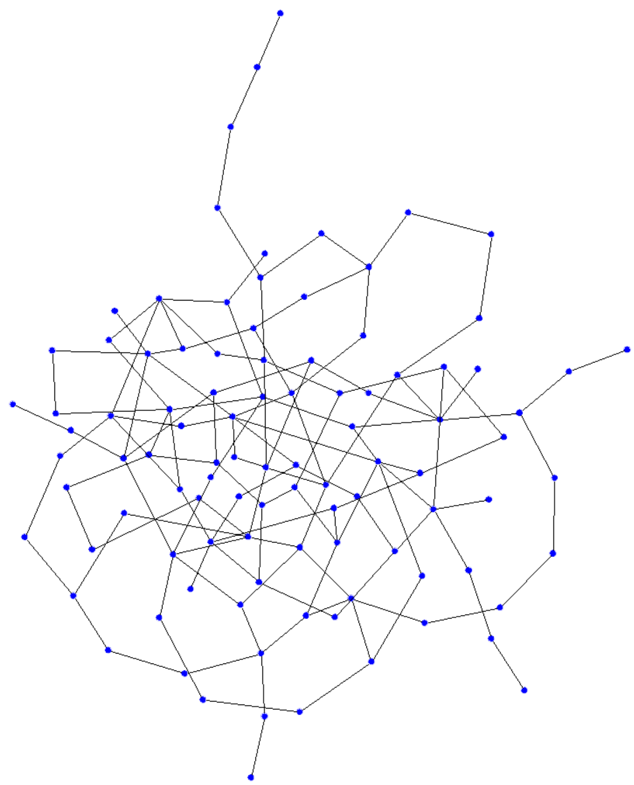
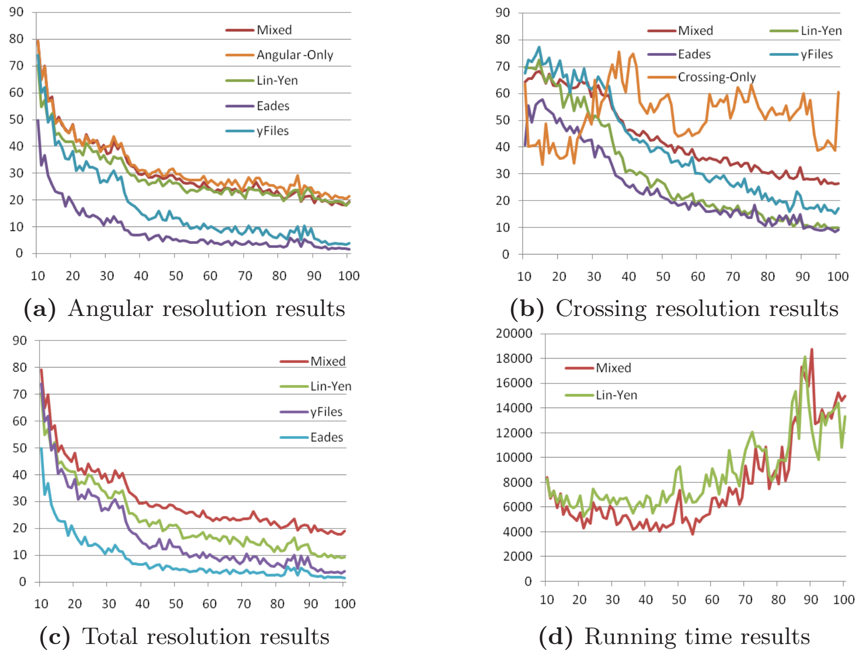


Fig. 4: A drawing of Rome graph grafo10129.99 consisting of 99 nodes and 135 edges with angular resolution 20.15^o and crossing resolution 26.12^o.

as (a) Crossing-Only, (b) Angular-Only and (c) Mixed. The results are illustrated in Fig.5 and should be viewed in color.

The Angular Resolution Maximization Problem: Refer to Fig.5a. Our experimental analysis shows that our Angular-Only algorithm achieves, on average, better angular resolution. The angular resolution of our Mixed algorithm is almost equal, on average, to the one of Lin and Yen. Note that the algorithm of Lin and Yen, in contrast to ours, does not modify the embedding of the initial layout [17] (i.e., it needs a close-to-final starting layout and improve on it). This explains why our Mixed algorithm achieves almost the same performance, in terms of angular resolution, as the one of Lin and Yen. In 59.76% of the graphs our Mixed algorithm yields a better solution compared to Lin-Yen's algorithm with an average improvement of 6.94^o.

The Crossing Resolution Maximization Problem: In Fig.5b the data were filtered to depict only the results of non-planar drawings produced by the algorithms and avoid infinity values in the case of planar ones. It is clear that our Crossing-Only algorithm results in drawings with high crossing resolution. Our Crossing-Only algorithm performs better on large graphs compared to the algorithm of yFiles. The average improvement implied by our Crossing-Only algorithm is 13.63^o w.r.t. the yFiles algorithm.



(a) Angular resolution results **(b)** Crossing resolution results **(c)** Total resolution results **(d)** Running time results

Fig. 5: A visual presentation of our experimental results. The X-axis indicates the number of the nodes of the graph. In Fig.(a)-(c) the Y-axis corresponds to the resolution measured in degrees, whereas in Fig.(d) to the running time measured in milliseconds.

The Total Resolution Maximization Problem: Refer to Fig.5c. This is the most important result of our experimental analysis. It indicates that our Mixed algorithm applied on graphs with more than 50 nodes constructs drawings with total resolution of 20 degrees, on average. Note that this is an achievement on average and under the particular termination condition discussed above, i.e., there is no guarantee, that it would be maintained indefinitely. An example is given in Fig.4.

Finally, Fig.5d summarizes the running time performance of the algorithms. Our algorithm needs, on average, 7340 milliseconds and 1298 iterations to converge, whereas the one of Lin and Yen 8346 and 1952, respectively. Note that the time complexity of Lin-Yen's algorithm is better than ours. However, the termination condition takes into account the crossing improvement and therefore the algorithm of Lin and Yen needs more iterations to converge, which explains this contradiction.

5 Conclusions

In this paper, we introduced and studied the total resolution maximization problem. Of course, our work leaves several open problems. It would be interesting to try to identify other classes of graphs that admit optimal drawings. Even the case of planar graphs is of interest, as by allowing some edges to cross (say at large angles), we may improve the angular resolution and therefore the total resolution.

References

1. P. Angelini, L. Cittadini, G. Di Battista, W. Didimo, F. Frati, M. Kaufmann, and A. Symvonis. On the perspectives opened by right angle crossing drawings. In *Proc. of 17th Int. Symp. on Graph Drawing (GD 2009)*, 2009.
2. I. Bárány and N. Tokushige. The minimum area of convex lattice n-gons. *Combinatorica*, 24(2):171–185, 2004.
3. G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis. Algorithms for drawing graphs: an annotated bibliography. *Comput. Geom.*, 4:235–282, 1994.
4. C. C. Cheng, C. A. Duncanyz, M. T. Goodrichz, and S. G. Kobourovz. Drawing planar graphs with circular arcs. *Discrete and Computational Geometry*, 25:117–126, 1999.
5. E. Di Giacomo, W. Didimo, G. Liotta, and H. Meijer. Area, curve complexity, and crossing resolution of non-planar graph drawings. In *Proc. of 17th Int. Symp. on Graph Drawing (GD 2009)*, 2009.
6. W. Didimo, P. Eades, and G. Liotta. Drawing graphs with right angle crossings. In *Proc. of the Algorithm and Data Structure Symp. (WADS 2009)*, 2009.
7. V. Dujmovic, J. Gudmundsson, P. Morin, and T. Wolle. Notes on large angle crossing graphs. In *Computing: Theory of Computing 2010*. Australian Computer Society, 2010.
8. P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, (42):149–160, 1984.
9. M. Formann, T. Hagerup, J. Haralambides, M. Kaufmann, F. Leighton, A. Symvonis, E. Welzl, and G. Woeginger. Drawing graphs in the plane with high resolution. *SIAM J. Comput.*, 22(5):1035–1052, 1993.
10. T. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software-Practice and Experience*, 21(11):1129–1164, 1991.
11. M. Garey and D. Johnson. Crossing number is np-complete. *SIAM J. Algebraic Discrete Methods*, 4:312–316, 1983.
12. A. Garg and R. Tamassia. Planar drawings and angular resolution: Algorithms and bounds (extended abstract). In *Proc. 2nd Annu. Eur. Sympos. Alg.*, pages 12–23, 1994.
13. C. Gutwenger and P. Mutzel. Planar polyline drawings with good angular resolution. In *Proc. of 6th Int. Symp. on Graph Drawing*, volume 1547 of *LNCS*, pages 167–182, 1998.
14. W. Huang. Using eye tracking to investigate graph layout effects. *Asia-Pacific Symp. on Visualization*, pages 97–100, 2007.
15. W. Huang, S.-H. Hong, and P. Eades. Effects of crossing angles. In *PacificVis*, pages 41–46, 2008.
16. M. Kaufmann and D. Wagner, editors. *Drawing Graphs: Methods and Models*, volume 2025 of *LNCS*. Springer-Verlag, 2001.
17. C.-C. Lin and H.-C. Yen. A new force-directed graph drawing method based on edge-edge repulsion. In *Proc. of the 9th Int. Conf. on Inf. Vis.*, pages 329–334. IEEE, 2005.
18. S. M. Malitz and A. Papakostas. On the angular resolution of planar graphs. In *STOC*, pages 527–538, 1992.
19. F. P. Preparata and M. I. Shamos. *Computational geometry: an introduction*. Springer-Verlag, 1985.
20. H. C. Purchase. Effective information visualisation: a study of graph drawing aesthetics and algorithms. *Interacting with Computers*, 13(2):147–162, 2000.
21. H. C. Purchase, D. A. Carrington, and J.-A. Allder. Empirical evaluation of aesthetics-based graph layout. *Empirical Software Engineering*, 7(3):233–255, 2002.
22. C. Ware, H. Purchase, L. Colpoys, and M. McGill. Cognitive measurements of graph aesthetics. *Information Visualization*, 1(2):103–110, 2002.