Algorithmica (1994) 11: 379–403

Algorithmica

Edge Crossings in Drawings of Bipartite Graphs

Peter Eades[1] and Nicholas C. Wormald[2]

Abstract. Systems engineers have recently shown interest in algorithms for drawing directed graphs so that they are easy to understand and remember. Each of the commonly used methods has a step which aims to adjust the drawing to decrease the number of arc crossings. We show that the most popular strategy involves an NP-complete problem regarding the minimization of the number of arcs in crossings in a bipartite graph. The performance of the commonly employed "barycenter" heuristic for this problem is analyzed. An alternative method, the "median" heuristic, is proposed and analyzed. The new method is shown to compare favorably with the old in terms of performance guarantees. As a bonus, we show that the median heuristic performs well with regard to the total length of the arcs in the drawing.

Key Words. Graph, Bipartite graph, Directed graph, Edge crossing, Median.

1. Introduction. Directed graphs are used to represent aspects of systems in a wide variety of disciplines, including software and information engineering, management, and desktop publishing. The usefulness of these representations depends on the layout of the graph. Thus there has been considerable interest in algorithms for drawing directed graphs so that they are easy to understand and remember. Tools which use such algorithms are described in [19], [21], [23], and [25]. See [9] and [10] for surveys.

A precise definition of what is a "good" drawing of a directed graph seems to be difficult to obtain, but the following criteria are generally agreed to be necessary:

C1. Flow should be clearly illustrated.
C2. Nodes should be distributed evenly over the page.
C3. There should be as few arc crossings as possible.

A "successive-refinement" technique is commonly used: a drawing is refined according to each criterion in turn, from C1 to C3. There are three steps S1, S2, S3, where Si attempts to fix those aspects of the drawing which relate to criterion Ci, while leaving other aspects of the layout free. Step S3 involves minimizing the number of edge crossings in a drawing of a bipartite graph. This paper is concerned only with the implementation of step S3, but to put our results in context we briefly review the other steps below.

[1] Department of Computer Science, University of Queensland, Queensland, Australia 4072.
[2] Department of Mathematics, University of Melbourne, Parkville, Victoria, Australia 3052.

Received February 8, 1991; revised November 15, 1991. Communicated by F. Thomson Leighton.

Step S1: Remove Cycles. Criterion C1 can be achieved by ensuring that as many arcs as possible are monotonic in a given direction. We assume that the preferred direction is upward, but our remarks apply equally well to any given direction.

Many applications use an acyclic directed graph $G = (V, A)$; in this case, a topological sort can be used to assign a unique label $l(v) \in \{1, 2, \ldots, |V|\}$ to each node v, so that each arc uv has $l(u) < l(v)$. Any drawing of the digraph with the labels in increasing order up the page has flow clearly illustrated as upward.

The difficulty lies when directed graphs containing cycles are required. Not every arc in a directed cycle can point upward, so we must temporarily reverse some arcs so that the digraph becomes acyclic. The temporarily reversed arcs appear "against the flow" in our final drawing, so we must reverse as few arcs as possible. This problem is equivalent to the NP-complete Feedback Arc Set Problem (see Section 2) but effective heuristics have been developed; see, for example, [3] and [8].

Step S2: Layer the Acyclic Digraph. Criterion C2 is achieved by "layering" the acyclic digraph output from step S1, as follows: A *k-layered network* is a digraph $G = (V, A)$ where the node set V is partitioned into *layers* $L_0, L_1, \ldots, L_{k-1}$, with the property that if $uv \in A$, where $u \in L_i$ and $v \in L_j$, then $i < j$. Networks are conventionally drawn so that all nodes in layer L_m lie on the horizontal line $y = m$. This drawing convention also ensures that all arcs point upward, so that the illustration of flow gained in step S1 is not lost.

In some applications the assignment to layers is given by the semantics of the graph; in other applications the drawing method must layer the graph. Essentially, we must choose a y-coordinate for each node.

A topological sort could be used to layer a digraph but each layer would contain just one node, and the nodes would not be spread evenly (unless the page is very tall and very narrow!).

"Longest-path layering" is most commonly used. All sources (nodes of indegree 0) are placed in layer 0; each remaining node v is then placed in layer L_m where the longest path from a source to v has length m. This layering places each node at the lowest position possible. It can be efficiently obtained by a variant of Dijkstra's shortest-path algorithm, and although a drawing that is too wide is possible, for most digraphs it seems to produce a reasonably even spread of nodes.

Layering can also take other parameters into account: see [14] and [20], where layerings which reduce the total length of arcs are used. Further remarks on the complexity of layering can be found in [6].

Step S3: Remove Arc Crossings from the Layered Network. A *long* arc in a layered network is an arc which spans more than two layers; that is, an arc uv with $u \in L_i$ and $v \in L_j$ and $i < j - 1$. A layered network with no long arcs is *proper*. It is difficult to handle crossings which involve long arcs, so we replace each long arc with a path $v_0 = u \to v_1 \to \cdots \to v_{j-i} = v$, adding the dummy nodes $v_1, v_2, \ldots, v_{j-i-1}$. This process is illustrated in Figure 1.

The number of crossings in a drawing of a proper layered network does not depend on the precise position of nodes but only on the ordering of the nodes

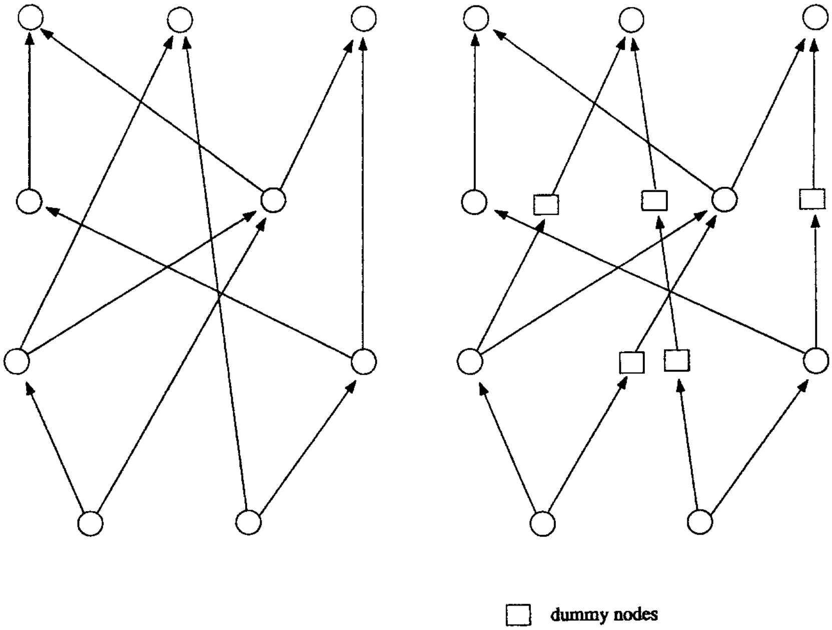


Fig. 1. Adding dummy nodes.

within each layer. Thus the problem of reducing arc crossings is the combinatorial one of choosing an appropriate ordering for each layer, not the geometric one of choosing an x-coordinate for each node. Although this combinatorialization considerably simplifies the problem, it is still difficult: the proof method of [13] implies that the problem of minimizing arc crossings in a layered network is NP-complete, even if there are only two layers.

A kind of "level-by-level sweep" heuristic is usually applied as follows. First, an ordering of L_0 is chosen. Then, for $i = 1, 2, \ldots, k - 1$, the ordering of L_i is held fixed while re-ordering L_{i+1} to reduce crossings between L_{i+1} and L_i. The process can be repeated to reduce crossings further.

This method presupposes a solution to the following problem, which is the major concern of this paper: given a fixed ordering of layer L_i, choose an ordering of L_{i+1} with a minimal number of crossings. In Section 3 this problem is shown to be NP-complete. Section 4 discusses heuristics for the problem. The main results are bounds on the maximum number of arc crossings in the output of two heuristics: the popular "barycenter" heuristic, and the newer "median" heuristic. Section 5 adds a note on the performance of these heuristics with respect to the total length of the arcs.

Steps S1–S3 form an elementary method for drawing directed graphs, but substantial improvements to this technique have been studied. For instance, although the arcs of the proper layered network can be represented as straight lines, the minimization of crossings in step S3 may result in excessively bent long arcs. Sugiyama *et al.* [22] show how such arcs can be "straightened" by choosing an x-coordinate for each node, without altering the order of nodes within each layer.

We should also mention that various versions of the "planarity" problem for directed acyclic graphs has been studied extensively: for instance, algorithms in [2], [16], [23], and [24] can be used for drawing acyclic graphs which are "upward planar," that is, all arcs point upward and no arcs cross. For a brief bibliographic survey of these and related results, see [10]. However, none of these algorithms assist with minimizing crossings in nonplanar acyclic digraphs.

2. Terminology.

This paper concentrates on the following problem:

Given a two-layered network G with layers L_0 and L_1, and a fixed ordering of the nodes in L_0, choose an ordering of the nodes in L_1 so that the number of arc crossings is minimal.

Standard graph-theoretic terminology is from [4]. Some further terminology is needed for a precise discussion.

We can consider a two-layered network to be a bipartite graph $G = (L_0, L_1, A)$ which consists of disjoint sets L_0 and L_1 of nodes and a set $A \subseteq L_0 \times L_1$ of arcs. Note that the direction of the arcs (from L_0 to L_1) has no effect on crossings, so we consider G to be an undirected graph and consider the arcs to be edges. An edge between u and v is denoted by uv. The set $\{u : uv \in A\}$ is denoted by N_v for each $v \in L_1$. For any vertex u denote the degree $|N_u|$ of u by d_u. For each pair u, v of vertices denote $|N_u \cap N_v|$ by ι_{uv}.

We assume that the nodes in L_0 and L_1 are to appear on the horizontal lines $y = 0$ and $y = 1$, respectively. We assume (without loss of generality with respect to edge crossings) that they are straight lines. Thus a drawing of G is specified by giving a unique x-coordinate $x_i(u)$ for each node $u \in L_i$, $i = 0, 1$. Two edges tw and uv cross if and only if $(x_0(t) - x_0(u))(x_1(w) - x_1(v))$ is negative. The number of crossings in a drawing of G specified by x_0 and x_1 is denoted by $cross(G, x_0, x_1)$. Note that $cross(G, x_0, x_1)$ depends on the *order* of the positions of the vertices in L_0 and L_1, not on their precise positions. We say that x_0 and x_1 are *orderings* of L_0 and L_1, respectively. Denote the minimum number of crossings in a drawing of G with L_0 ordered by x_0 by $opt(G, x_0)$.

Using the terms defined above, the central problem of this paper can be stated precisely as follows.

CROSSING PROBLEM (CP). Given a two-layered network $G = (L_0, L_1, A)$ and an ordering x_0 of L_0, find an ordering x_1 of L_1 such that $cross(G, x_0, x_1)$ is as small as possible.

It was shown in [7] that CP is NP-hard; we give the proof in the next Section. This result justifies a heuristic approach; such methods are discussed in Section 4.

3. The Crossing Problem Is Difficult. CP can be stated as a decision problem in the usual complexity theory style as follows:

DECISION CROSSING PROBLEM (DCP)

Instance: Two-layered network $G = (L_0, L_1, A)$, an ordering x_0 of L_0, an integer M.
Question: Is there an ordering x_1 of L_1 so that $cross(G, x_0, x_1) \leq M$?

In this section we present the proof of the following theorem:

THEOREM 1 [7]. *DCP is* NP-*complete.*

It is clear that DCP is in NP. To prove that DCP is NP-complete, we give a problem in directed graphs and a polynomial-time transformation to DCP. If D is a directed graph with node set U and arc set B, then a *feedback arc set* B' for D is a subset of B which contains at least one arc from each directed cycle of D. Note that B' is a feedback arc set if and only if the subgraph $(U, B - B')$ of D is acyclic. The following problem is NP-complete (see p. 192 of [12]).

FEEDBACK ARC SET (FAS)

Instance: Directed graph D, positive integer K.
Question: Does D have a feedback arc set of size at most K?

From an instance $D = (U, B)$, K of FAS we construct a two-layered network $G = (L_0, L_1, A)$ as follows. Let $L_1 = U$. For each arc $a \in B$, we define a "clump" $C(a) = \{c_1(a), c_2(a), \ldots, c_6(a)\}$ of six nodes, and let L_0 be the union over B of all the clumps.

The arcs of G are constructed as follows. For each $u \in U$ and $a \in B$ there are two arcs joining $u \in L_1$ to $C(a)$. If $a = uv$ for some $v \in U$, then u is joined to $c_1(a)$ and $c_5(a)$; if $a = vu$ for some $v \in U$, then u is joined to $c_2(a)$ and $c_6(a)$. If u is not incident with a, then u is joined to $c_3(a)$ and $c_4(a)$. The three possibilities are illustrated in Figure 2.

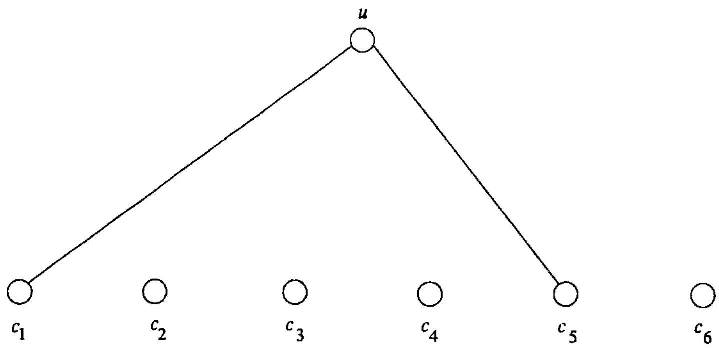
Now let x_0 be any ordering of L_0 which keeps each clump together and in order, that is, so that $x_0(c_i(a)) < x_0(c_j(a))$ for each $1 \leq i < j \leq 6$ and $a \in B$. Denote $|B|$ by β and $|U|$ by υ, and let M be

$$4\binom{\beta}{2}\binom{\upsilon}{2} + \beta\binom{\upsilon - 2}{2} + 4\beta(\upsilon - 2) + \beta + 2K.$$

We show that D has a feedback arc set of size at most K if and only if an ordering x_1 of L_1 can be chosen so that $cross(G, x_0, x_1) \leq M$.

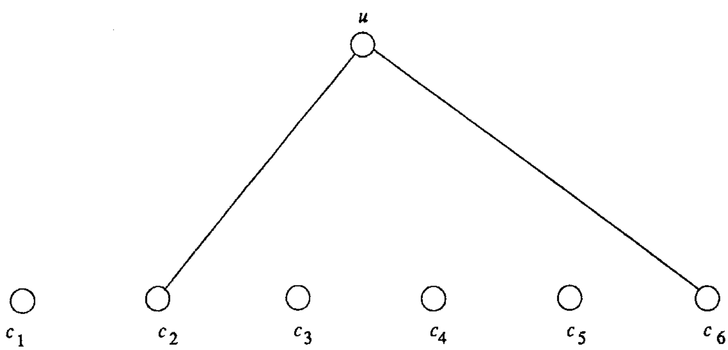
LEMMA 1. *If x_1 is an ordering of L_1 and B' denotes $\{uv \in B: x_1(u) > x_1(v)\}$, then*

$$cross(G, x_0, x_1) = 4\binom{\beta}{2}\binom{\upsilon}{2} + \beta\binom{\upsilon - 2}{2} + 4\beta(\upsilon - 2) + \beta + 2|B'|.$$



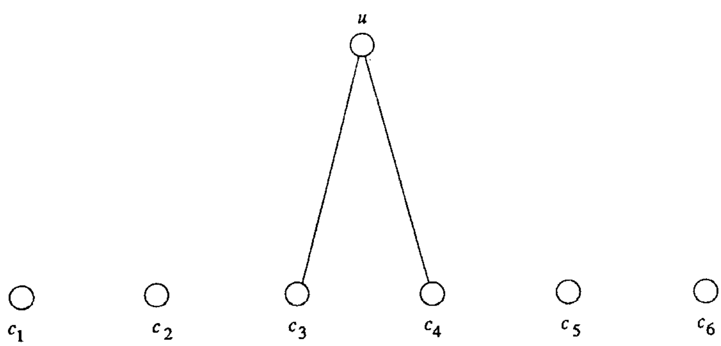
$C(a)$, where $a = uv$

(a)



$C(a)$, where $a = vu$

(b)



$C(a)$, where a is not incident with u

(c)

Fig. 2. Three possibilities.

PROOF. If a and b are distinct arcs, then, since every node u in L_1 has two arcs joining u to $C(a)$ and two to $C(b)$, there are $4\binom{\upsilon}{2}$ crossings between arcs incident with $C(a)$ and arcs incident with $C(b)$. This gives $4\binom{\beta}{2}\binom{\upsilon}{2}$ crossings between arcs from different clumps.

Now consider crossings between arcs from the same clump $C(a)$. There are $\upsilon - 2$ nodes in L_0 which are incident with both $c_3(a)$ and $c_4(a)$, giving $\binom{\upsilon - 2}{2}$ crossings between arcs from these two nodes. If $a = (u, v)$, then each arc out of $c_3(a)$ and $c_4(a)$ crosses one of $(u, c_1(a))$ and $(u, c_5(a))$, and one of $(v, c_2(a))$ and $(v, c_6(a))$, giving another $4(\upsilon - 2)$ crossings. The only remaining crossings are between $(u, c_1(a))$, $(u, c_5(a))$, $(v, c_2(a))$, and $(v, c_6(a))$. This is 1 if $x_1(u) < x_1(v)$, and 3 if $x_1(u) > x_1(v)$. Thus the total number of crossings between arcs from $C(a)$ is $\binom{\upsilon - 2}{2} + 4(\upsilon - 2) + 1$ if $x_1(u) < x_1(v)$, and $\binom{\upsilon - 2}{2} + 4(\upsilon - 2) + 3$ if $x_1(u) > x_1(v)$. Summing over all clumps gives the result stated in the lemma. □

Now suppose that D has a feedback arc set B' of size at most K. Since $D' = (U, B - B')$ is acyclic we can obtain an ordering x_0 of U so that $x_1(u) < x_1(v)$ whenever $(u, v) \in (B - B')$ by a topological sort. Since $|B'| \leq K$ and

$$B' = \{(u, v) \in B\colon x_1(u) > x_1(v)\},$$

the lemma implies that G has at most M crossings.

Conversely, suppose that x_0 is an ordering of L_0 such that G has at most M crossings. It follows from the lemma that if $B' = \{(u, v) \in B\colon x_1(u) > x_1(v)\}$, then $|B'| \leq K$. Furthermore, $D' = (U, B - B')$ must be acyclic and so B' is a feedback arc set.

4. Heuristics. The most common heuristics employed for CP are variations of the *barycentric* method: roughly speaking, the x-coordinate of each vertex $u \in L_1$ is chosen as the *barycenter* (average) of the x-coordinates of its neighbors. That is, we select $x_1(u)$ to be $avg(u)$ for all $u \in L_1$, where

$$avg(u) = \frac{1}{d_u} \sum_{v \in N_u} x_0(v).$$

The number of crossings output by the barycenter heuristic is denoted by $avg(G, x_0)$.

The barycenter heuristic is efficient, and its performance is analyzed below. We show that, in the worst case, it gives a drawing with $K\ opt(G, x_0)$ crossings, where K is $O(\sqrt{|L_0|})$. If G is dense, then this bound is improved.

We also investigate a new method: the *median* heuristic. Roughly speaking, the x-coordinate of each $u \in L_1$ is chosen to be a median of the x-coordinates of the neighbors of u. Precisely, if $N_u = \{v_1, v_2, \ldots, v_j\}$ with $x_0(v_1) \le x_0(v_2) \le \cdots \le x_0(v_j)$, then define $med(u) = x_0(v_{\lfloor j/2 \rfloor})$. If N_u is empty, then we choose $med(u) = 0$. A median heuristic chooses $x_1(u) = med(u)$ for all $u \in L_1$, and separates two vertices with the same median by an infinitesimal amount, with the restriction that if one vertex has odd degree and the other even, then the odd degree vertex is chosen to have the lower x_1 value. In practice, the choice of ordering of vertices with the same median is deterministic. However, the performance bounds given in this paper hold independently of this choice subject to the restriction given.

The number of crossings in the output of a median heuristic is denoted by $med(G, x_0)$. For each u, $med(u)$ can be computed in linear time (see [1]), so the drawing can be computed efficiently.

We show that the median heuristic gives a drawing with no more than three times $opt(G, x_0)$ crossings. Better guarantees are given for some families of graphs; in particular, for dense graphs, and for graphs of low degree.

As a final note, we prove that the median heuristic has another attractive property: it guarantees that the total length of all the edges is close to optimal.

4.1. The Median Heuristic. The main results of this section are guarantees for the performance of the median heuristic.

For typographical convenience we denote $t(t-1)/2$ by $\chi(t)$; if t is a nonnegative integer, then $\chi(t)$ is the number of subsets of cardinality 2 of a set of cardinality t.

THEOREM 2. *Suppose that $G = (L_0, L_1, A)$ is a two-layered network. Denote $|L_0|$ by l_0, $|L_1|$ by l_1, and $|A|$ by m. Then, for any ordering x_0 of L_0,*

$$\frac{med(G, x_0)}{opt(G, x_0)} \le \frac{3m^2 - 16\sigma}{m^2 + 16\sigma - 2\chi(l_1)}, \qquad where \quad \sigma = \chi(l_1)\chi\left(\frac{l_0\chi(m/l_0)}{2\chi(l_1)}\right).$$

The bound given by this theorem appears to be rather complex, but it has the following interesting corollary.

COROLLARY. *Suppose that $\varepsilon > 0$ and $0 < c < 1$. Then there is an integer N_0 such that if G has at least cn^2 edges, and $l_0 = l_1 = n \ge N_0$, then, for any ordering x_0,*

$$\frac{med(G, x_0)}{opt(G, x_0)} \le \frac{3 - c^2}{1 + c^2} + \varepsilon.$$

Note that the corollary implies that, for dense graphs (with c close to 1), the median heuristic is very close to optimal. This, together with Theorem 7 below, explains the experimental results of [15] and [17] for dense graphs.

The key lemma used below to prove Theorem 2 also shows that the median heuristic has a performance bound of 3:

THEOREM 3. *For all two-layered networks G and orderings x_0, $med(G, x_0) \leq 3\, opt(G, x_0)$.*

There are graphs for which the bound of Theorem 3 is almost met; for example, Figure 3 shows a graph with $med(G, x_0)/opt(G, x_0) = 3 - 4/l_0$.

To prove these theorems, the notion of the "crossing array" from [5] and [26] is used. Suppose that u and v are nodes in L_1. The *crossing array* C is an $|L_1| \times |L_1|$ matrix indexed by L_1 such that the uvth entry is the number of crossings that edges

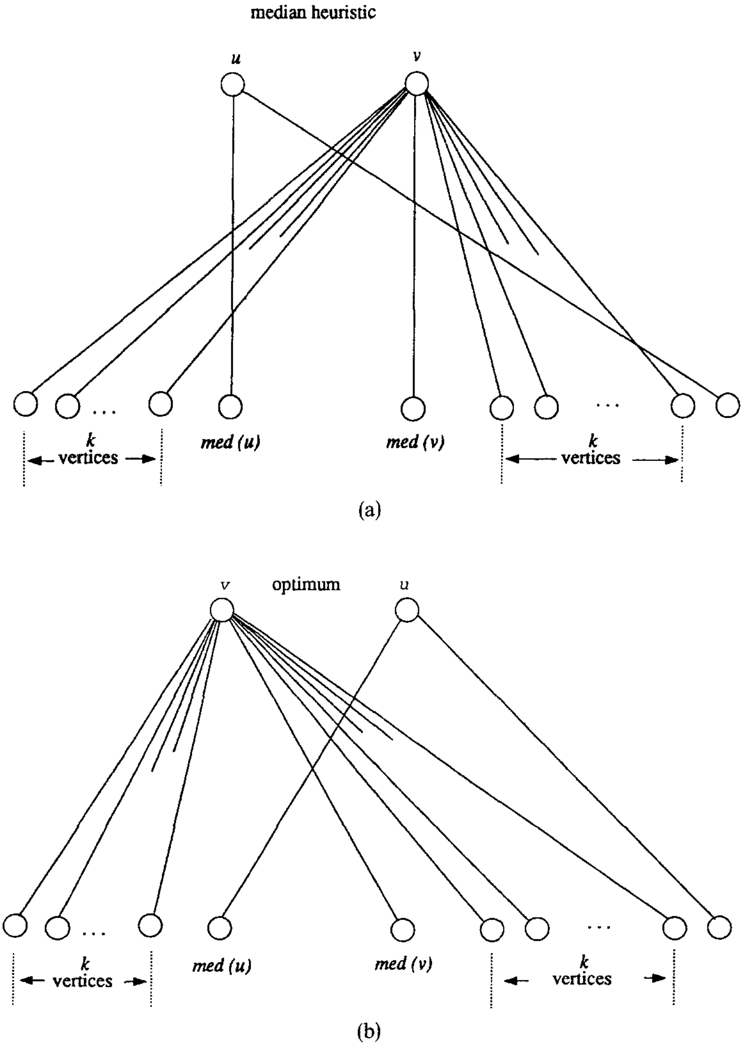


Fig. 3. Worst case for the median heuristic.

incident with u make with edges incident with v when $x_1(u) < x_1(v)$. More formally, $C = (c_{uv})_{u,v \in L_1}$, where, for $u \neq v \in L_1$,

$$c_{uv} = |\{\{ut, vw\} \subseteq A: x_0(t) > x_0(w)\}|$$

and $c_{uu} = 0$.

The next lemma is the crux of the proof of Theorem 2. For this lemma we define $\tau_{uv} = 1$ if $med(u) = med(v)$, and $\tau_{uv} = 0$ otherwise, and recall the definition of ι_{uv} from Section 2.

LEMMA 2. *Suppose that $u, v \in L_1$ and $med(u) \leq med(v)$.*

(a) *If d_u and d_v are both even, then*

$$4c_{uv} \leq 3d_u d_v - 2d_u - 8\chi\left(\frac{\iota_{uv}}{2}\right),$$

$$4c_{vu} \geq d_u d_v + 2d_u + 8\chi\left(\frac{\iota_{uv}}{2}\right) - (1 + 2\iota_{uv})\tau_{uv}.$$

(b) *If d_u is even and d_v is odd, then*

$$4c_{uv} \leq 3d_u d_v - d_u - 8\chi\left(\frac{\iota_{uv}}{2}\right),$$

$$4c_{vu} \geq d_u d_v + d_u + 8\chi\left(\frac{\iota_{uv}}{2}\right) - (1 + 2\iota_{uv})\tau_{uv}.$$

(c) *If d_u is odd and d_v is even, then*

$$4c_{uv} \leq 3d_u d_v - 2d_u - d_v - 2 - 8\chi\left(\frac{\iota_{uv}}{2}\right),$$

$$4c_{vu} \geq d_u d_v + 2d_u + d_v + 2 + 8\chi\left(\frac{\iota_{uv}}{2}\right) - (1 + 2\iota_{uv})\tau_{uv}.$$

(d) *If d_u and d_v are both odd, then*

$$4c_{uv} \leq 3d_u d_v - d_u - d_v - 1 - 8\chi\left(\frac{\iota_{uv}}{2}\right),$$

$$4c_{vu} \geq d_u d_v + d_u + d_v + 1 + 8\chi\left(\frac{\iota_{uv}}{2}\right) - (1 + 2\iota_{uv})\tau_{uv}.$$

PROOF. Divide the edges incident with u and v into four groups α, β, γ, and δ, where

$$\alpha = \{uw : x_0(w) \leq med(u)\},$$

$$\beta = \{vw : x_0(w) \geq med(v)\},$$

$$\gamma = \{vw : x_0(w) < med(v)\},$$

$$\delta = \{uw : x_0(w) > med(u)\}.$$

A network with groups

$$\alpha = \{3, 4, 5\},$$

$$\beta = \{9, 10, 11, 12, 13\},$$

$$\gamma = \{1, 2, 6, 8\},$$

$$\delta = \{7, 12, 13\}$$

is illustrated in Figure 4. Now denote $|\alpha|$ by a, $|\beta|$ by b, $|\gamma|$ by c, $|\delta|$ by d, $|\{uw \in \alpha : vw \in \gamma\}|$ by e, $|\{vw \in \beta : uw \in \delta\}|$ by f, and $|\{vw \in \gamma : uw \in \delta\}|$ by g. If $med(u) = x_1(u) \leq med(v) = x_1(v)$, then edges in α cannot cross edges in β. Furthermore, the number of crossings between edges in α and edges in γ is at most $a(c - g) - \chi(e + 1)$, and similarly for crossings between edges in β and edges in δ. Also, the number of crossings between edges in γ and edges in δ is at most

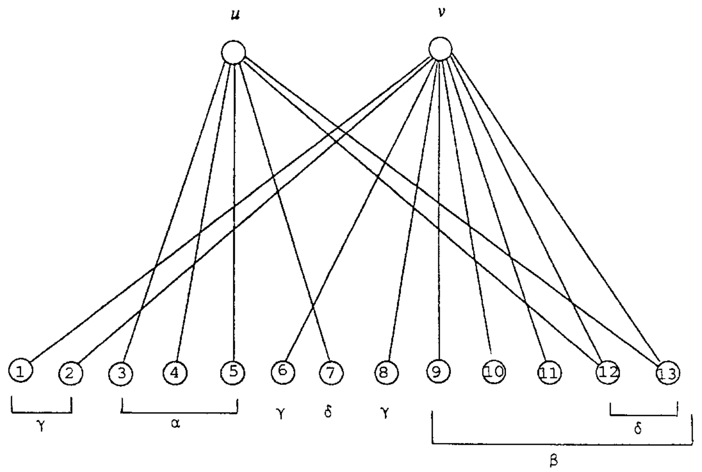


Fig. 4. The groups α, β, γ, and δ.

$cd - \chi(g+1)$. Note that since $\iota_{uv} = e + f + g + \tau_{uv}$,

$$c_{uv} \leq a(c-g) - \chi(e+1) + b(d-g) - \chi(f+1) + cd - \chi(g+1).$$

Now $a \geq e$ and $b \geq f$, and $\chi(t+1) \geq t^2/2$, so

$$\begin{aligned} c_{uv} &\leq ac + bd + cd - eg - fg - \frac{e^2 + f^2 + g^2}{2} \\ &\leq ac + bd + cd - \frac{\iota_{uv}^2 + \iota_{uv} - \tau_{uv} - 2ef}{2} \\ &\leq ac + bd + cd - \frac{(\iota_{uv} - \tau_{uv}^2) + 2\iota_{uv} - 2\tau_{uv}}{4} \end{aligned}$$

and thus

(i) $$c_{uv} \leq ac + bd + cd - 2\chi\left(\frac{\iota_{uv}}{2}\right).$$

Furthermore, if u and v are placed so that $x_1(u) > x_1(v)$, then edges in α must cross edges in β, except for one crossing if $med(u) = med(v)$; hence

$$c_{vu} \geq ab + \chi(e) + \chi(f) + \chi(g) + ef + fg - \tau_{uv}$$

and, analyzing as for (i),

(ii) $$c_{vu} \geq ab + 2\chi\left(\frac{\iota_{uv}}{2}\right) - \frac{(1 + 2\iota_{uv})\tau_{uv}}{4}.$$

Now consider part (a) of Lemma 2, that is, both d_u and d_v are even. This implies that $a = d = d_u/2$ and $b = c + 2 = (d_v + 2)/2$, by the definitions of $med(u)$ and $med(v)$. Hence, from (i),

$$c_{uv} \leq 3ab - 4a - 2\chi\left(\frac{\iota_{uv}}{2}\right) \leq \tfrac{3}{4}d_u(d_v + 2) - 2d_u - 2\chi\left(\frac{\iota_{uv}}{2}\right)$$

and, from (ii),

$$c_{vu} \geq \tfrac{1}{4}d_u(d_v + 2) + 2\chi\left(\frac{\iota_{uv}}{2}\right) - \frac{(1 + 2\iota_{uv})\tau_{uv}}{4}.$$

This establishes part (a) of the lemma. Parts (b)–(d) can be proved in a similar fashion. □

Next we show how the theorems can be deduced from Lemma 2 with the aid of the following two lemmas from [5].

LEMMA 3. $cross(G, x_0, x_1) = \sum_{x_1(u) < x_1(v)} c_{uv}$.

LEMMA 4. $opt(G, x_0) \geq \sum_{u,v} \min(c_{uv}, c_{vu})$, *where the sum is over all unordered pairs* $\{u, v\}$.

These two lemmas imply that if we can establish that there is a uniform bound B such that whenever $x_1(u) < x_1(v)$ we have

$$c_{uv} \leq Bc_{vu},$$

then it follows that

$$cross(G, x_0, x_1) \leq B\ opt(G, x_0)$$

by summing the inequality over all pairs u, v with $x_1(u) < x_1(v)$.

Lemma 2 provides a uniform bound of the type above. For example, Theorem 3 can be obtained easily: whatever the parity of the degrees of u and v, Lemma 2 and the inequalities

$$\imath_{uv} \leq \min\{d_u, d_v\}, \quad 8\chi\left(\frac{\imath_{uv}}{2}\right) \geq -1$$

together imply that $c_{uv} \leq 3c_{vu}$ except when d_u is even. When d_u is even and d_v is odd, we can assume that $\tau_{uv} = 0$, since otherwise the median heuristic forces $x_1(u) > x_1(v)$, and thus this case is also covered by Lemma 2. When d_u and d_v are both even, Lemma 2 implies that

$$4c_{uv} \leq 3d_u d_v - 2d_u + 1 \leq 3d_u d_v - 3$$

as $d_u \geq 2$ and

$$4c_{uv} \geq d_u d_v + 2d_u - 1$$

as $2d_u + 8\chi(\imath_{uv}/2) - 2\imath_{uv} \geq 0$.

To prove Theorem 2, a little more computation is needed. Note that whatever the parity of d_u and d_v, Lemma 2 ensures that $4c_{uv} \leq 3d_u d_v - d_u - 8\chi(\imath_{uv}/2)$ and $4c_{vu} \geq d_u d_v + 8\chi(\imath_{uv}/2) - 1$.

To sum these inequalities we need a lower bound on $\hat{\sum} \chi(\imath_{uv}/2)$, where $\hat{\sum}$ denotes the sum over all unordered pairs of vertices in L_1. For this bound a useful fact about convex functions is employed.

USEFUL FACT. Suppose that $f(t)$ is a convex function. If $t_1 + t_2 + \cdots + t_k = T$, then $\sum_{i=1}^{k} f(t_i) \geq kf(\hat{t})$, where $\hat{t}$ denotes T/k.

Let q denote the number of paths of length 2 which start and finish in L_1, that is, $q = \hat{\sum} \iota_{uv}$. Note also that $q = \sum_{v \in L_0} \chi(d_v)$ and $m = \sum_{v \in L_0} d_v$. Using the Useful Fact, with $T = m$ and $k = l_0$, we obtain

$$\text{(iii)} \qquad q \geq l_0 \chi(\hat{d}),$$

where $\hat{d}$ denotes m/l_0.

Once more using the Useful Fact, this time with $T = q$ and $k = \chi(l_1)$, we obtain a lower bound of the kind that we want:

$$\text{(iv)} \qquad \hat{\sum} \chi\left(\frac{\iota_{uv}}{2}\right) \geq \chi(l_1)\chi\left(\frac{\hat{\iota}}{2}\right),$$

where $\hat{\iota}$ is $q/\chi(l_1)$.

From (iii) and (iv) it follows that $\hat{\sum} \chi(\iota_{uv}/2)$ is at least

$$\sigma = \chi(l_1)\chi\left(\frac{l_0 \chi(m/l_0)}{2\chi(l_1)}\right).$$

Hence

$$\frac{med(G, x_0)}{opt(G, x_0)} \leq \frac{3 \sum d_u d_v - 8\sigma}{\sum d_u d_v + 8\sigma - \chi(l_1)},$$

where both sums are over all pairs of unordered pairs of vertices in L_1.

Finally, note that $\hat{\sum} d_u d_v$ is bounded above by $(\sum_{u \in L_1} d_u)^2/2$, that is, $m^2/2$. This implies Theorem 2.

To prove the corollary, note that if $l_0 = l_1 = n$ and $m = cn^2$, then we obtain a simple expression for σ:

$$\sigma = \frac{c^4 n^4}{16}\left(1 - O\left(\frac{1}{n}\right)\right).$$

4.2. The Barycenter Heuristic. In this section we analyze the performance of the barycenter heuristic.

If the x-coordinates of nodes in L_0 are allowed to vary wildly, then the barycenter heuristic can perform very badly. Figure 5 illustrates a graph with n nodes for which $avg(G, x_0)$ is $n - 4$, but $opt(G, x_0)$ is only 1. However, in most applications (such as [17] and [26]) each x_0 is a one-to-one function from L_0 to $\{1, 2, \ldots, \max(|L_0|, |L_1|)\}$, that is, the x-coordinate of each vertex is a unique integer in the range $1, 2, \ldots, \max(|L_0|, |L_1|)$. We henceforth assume that x_0 satisfies this condition.

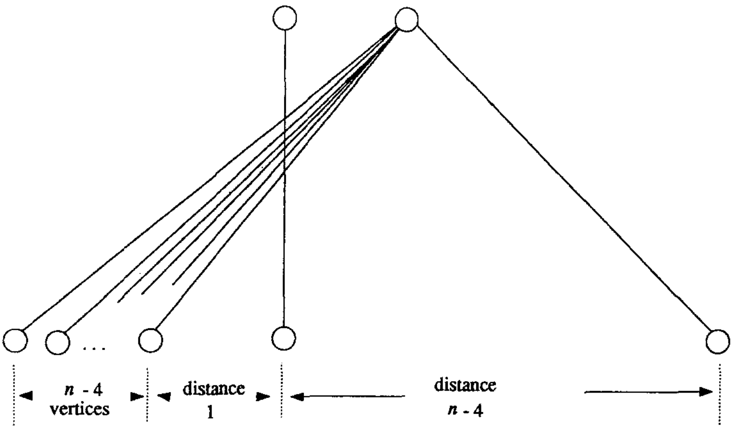


Fig. 5. A graph with n nodes for which $avg(G, x_0)/opt(G, x_0) = n - 4$.

Figure 6 shows a graph with n nodes for which $avg(G, x_0) = (\sqrt{n} - 2)opt(G, x_0)$ so a constant performance bound such as that in Theorem 3 cannot be proven for the barycenter heuristic. However, we can show that the example above is (asymptotically) the worst-possible situation for the barycenter heuristic.

THEOREM 4. *For all G and x_0, $avg(G, x_0)/opt(G, x_0)$ is $O(\sqrt{n})$.*

As with the median heuristic, the performance is better for dense graphs. The following theorem expresses this improvement in terms of the minimum degree of the graph.

THEOREM 5. *If δ is the minimum degree of a graph G, then $avg(G, x_0)/opt(G, x_0)$ is $O(n/\delta)$.*

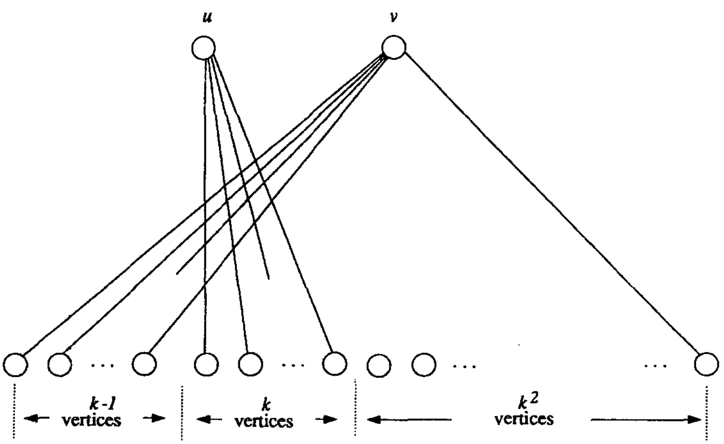


Fig. 6. A graph with n nodes for which $avg(G, x_0)/opt(G, x_0) = \sqrt{n} - 2$.

To prove these theorems, the same basic method as used for the median heuristic is employed: we obtain an upper bound for c_{uv}/c_{vu} for an arbitrary pair u, v of vertices in L_1 with $avg(u) \leq avg(v)$. Then Lemmas 3 and 4 can be employed to deduce that the same bound holds for $avg(G, x_0)/opt(G, x_0)$.

Suppose that u and v are two vertices in L_1, and $avg(u) \leq avg(v)$. Denote the degrees of u and v by j and k, respectively. Let N denote $N_u \cup N_v$. Suppose that x_0 is an ordering of the vertices in L_0.

A set of vertices in $L_0 - N$ which are consecutive with respect to x_0 and lie between elements of N is a *gap*. To prove the theorems, we need a lemma about the case where there are no gaps.

LEMMA 5. *If $c_{vu} \leq ekj$ and N is consecutive, then*

$$avg(u) \geq avg(v) + (\tfrac{1}{8} - 13e)(\max(j, k) - 1).$$

PROOF. First consider the case where $j \leq k$. Without loss of generality assume that the vertices in N have x-coordinates $1, 2, \ldots, t$. Note that $t \leq k + j \leq 2k$, since N is consecutive and $j \leq k$. Choose r so that the number of neighbors of v to the right of $avg(v)$ is rk. Then

$$k(1 - r) \leq avg(v) \leq (1 - r)\left(avg(v) - k\,\frac{1 - r}{2} + \frac{1}{2}\right) + r\left(k\left(2 - \frac{r}{2}\right) + \frac{1}{2}\right),$$

and so

$$avg(v) \leq -\frac{1 - r}{r}\left(\frac{k(1 - r)}{2} - \frac{1}{2}\right) + k\left(2 - \frac{r}{2}\right) + \frac{1}{2},$$

$$1 - r \leq 2 - \frac{2}{r} - \frac{(1 - r)^2}{2r} + \frac{1}{2kr}.$$

Thus $r(r - 4) \leq 1/k - 1$ and so, since $0 < r < 4$, we obtain the simple lower bound $r \geq (1 - 1/k)/4$.

Since $c_{vu} \leq ejk$, at most $8ej$ neighbors of u are to the left of the rightmost $\lceil k/8 \rceil$ of the neighbors of v. So at least $j - 8ej$ neighbors of u are to the right of all but $\lceil k/8 \rceil$ neighbors of v. Since at least $\lceil (k - 1)/4 \rceil$ vertices in N_v are to the right of $avg(v)$, the neighbors of u which are to the right of all but $\lceil k/8 \rceil$ neighbors of v have x-coordinate at least

$$avg(v) + \left\lceil \frac{k - 1}{4} \right\rceil - \left\lceil \frac{k}{8} \right\rceil + 1 \geq avg(v) + \frac{k - 1}{8}.$$

All other neighbors of u have x-coordinate at least one. Hence

$$j\ avg(u) \geq (j - 8ej)\left(avg(v) + \frac{k - 1}{8}\right) + 8ej,$$

that is,

$$avg(u) \geq avg(v) + \frac{k-1}{8} + 8e\left(1 - avg(v) - \frac{k-1}{8}\right) \geq avg(v) + \frac{k-1}{8} - 13ek$$

since

$$avg(v) \leq \frac{t-k+1+t}{2} = \frac{3k}{2} + \frac{1}{2}.$$

Thus

$$avg(u) \geq avg(v) + (\tfrac{1}{8} - 13e)(k-1).$$

Using the same argument with $j \geq k$, replacing $k-1$ by $j-1$, the lemma follows. □

We can now prove the theorems. Suppose that $u, v \in L_1$ and $avg(u) \leq avg(v)$. Firstly note that we can assume that there is only one gap. For suppose that there are two or more gaps; let S be a consecutive set of vertices between two gaps but containing no gap. Say S contains p neighbors of u and q neighbors of v. If $p/j \leq q/k$ we can shift S and its incident edges to the right, without altering c_{uv}, c_{vu}, or the inequality between $avg(u)$ and $avg(v)$. If $p/j > q/k$ we can shift S to the left. Repeating this operation with all the sets of consecutive vertices in N gives at most one gap.

Thus, without loss of generality, L_0 can be partitioned into a right set A, a left set B, and a gap in between, such that $A \cup B = N$, as in Figure 7. Let k_a, k_b denote the degrees of v in A, B, respectively, and $k = k_a + k_b$. Similarly define j_a, j_b,

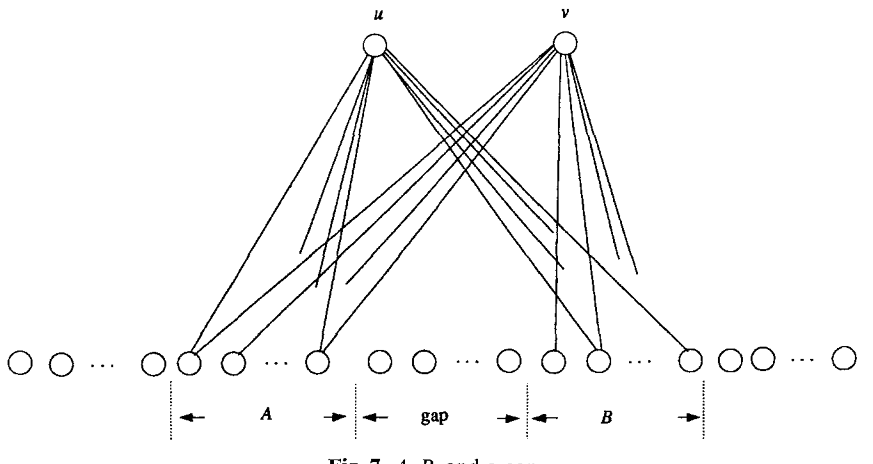


Fig. 7. A, B, and a gap.

and j, as the degrees of u in A, in B, and in total. By the argument immediately above, we can assume that

$$\text{(v)} \qquad \frac{j_a}{j} \leq \frac{k_a}{k}.$$

Furthermore, assume without loss of generality that

$$\text{(vi)} \qquad j_a k_a \geq j_b k_b;$$

if not, the orderings could be reversed. Finally assume that $k_a \geq j_a$ (otherwise we could swap k and j in the following argument).

Suppose that $c_{uv} < mjk$ where $m < \frac{1}{6}$. Then from (v) and (vi) we can deduce that

$$\text{(vii)} \qquad k_a j_b \leq c_{uv} < m(k_a j_a + k_a j_b + k_b j_a + k_b j_b) < m(2k_a j_a + 2k_a j_b).$$

Hence

$$\text{(viii)} \qquad k_a j_b < \frac{2m}{1-2m} k_a j_a < 3mk_a j_a,$$

and so $j_b < 3mj_a$. Thus

$$\text{(ix)} \qquad c_{vu} < 2mk_a j_a\left(1 + \frac{2m}{1-2m}\right) < 3mk_a j_a$$

by (vii) and (viii). Define $avg_a(u)$ to be the mean x-coordinate of the neighbors of u which are in A, and $avg_b(u)$, $avg_a(v)$, and $avg_b(v)$ similarly. By Lemma 5 (with k, j replaced by k_a, j_a) we have

$$\text{(x)} \qquad avg_a(u) \geq avg_a(v) + (k_a - 1)(\tfrac{1}{8} - 39m).$$

However, since $j_b < 3mj_a$ we have

$$\begin{aligned} avg(u) &> (1-3m)avg_a(u) + 3m\ avg_b(u) \\ &\geq (1-3m)avg_a(v) + (1-3m)(k_a-1)(\tfrac{1}{8} - 39m) + 3m\ avg_b(u) \qquad \text{by (x)} \\ &\geq (1-3m)avg(v) + (1-3m)(k_a-1)(\tfrac{1}{8} - 39m) + 3m\ avg_b(u). \end{aligned}$$

Hence since $avg(v) \geq avg(u)$ we have

$$(1-3m)(k_a-1)(\tfrac{1}{8} - 39m) + 3m(avg_b(u) - avg(v)) \leq 0.$$

Since $avg(v) - avg_b(u) \leq n$, this gives

$$(1 - 3m)(k_a - 1)(\tfrac{1}{8} - 39m) \leq 3mn.$$

Now suppose that $39m < 1/72$. Then $1 - 3m > \frac{1}{2}$, and $\frac{1}{8} - 39m > \frac{1}{4}$, and

(xi) $$k_a - 1 \leq 54mn.$$

Now if $k_a = 0$, then we get $j_a = j_b k_b = 0$ (since $k_b j_b \leq k_a j_a$), and if $k_a = 1$, then $j_a \leq 1$, and $j_b k_b = 1$ or 0. In all these situations it is readily verified that $c_{uv} \leq c_{vu}$, whilst, for $k_a \geq 2$, we get from (xi) that $k_a \leq 108mn$. Hence, by (ix), $j_b < 324m^2 n$. Thus either

(xii) $$m^2 \geq \frac{n}{324}$$

or $j_b = 0$. In the latter case, B can be moved right to A, and we can thus assume that B is empty; then by Lemma 5, $\frac{1}{8} - 13m \leq 0$, and so

(xiii) $$m \geq \frac{1}{104}.$$

In all situations we have

(xiv) $$\frac{c_{uv}}{c_{vu}} < \frac{1}{m}$$

which is $O(\sqrt{n})$, by (xii) and (xiii).

Also, if δ is the degree of any vertex in L_1, we can infer that $1/m$ is $O(n/k_a)$ from (xi). As either $j_a/j \geq \frac{1}{2}$ or $j_b/j_a \geq 1$, (v) and (vi) imply $k_a \geq k/2 \geq \delta/2$. So

(xv) $$\frac{c_{uv}}{c_{vu}} = O\left(\frac{n}{\delta}\right).$$

Summing inequalities (xiv) and (xv) gives Theorems 4 and 5.

4.3. Other Heuristics. In this section we prove a proposition which illustrates the limits of the methods described in the previous two sections. The median and the barycenter heuristics both have the following format:

for each node $u \in L_1$
  compute $x_1(u)$ from $\{x_0(w)\colon w \in N_u\}$

For each heuristic, we computed a bound on c_{uv}/c_{vu} for an arbitrary pair of nodes u, $v \in L_1$ and used Lemmas 2 and 3 to deduce a performance bound for the heuristic.

Now we show that no heuristic of this family can be proved to be better than the median heuristic using these methods.

Proposition. *Suppose that $x_1(u)$ is calculated as some function of $X_u = \{x_0(v): uv \in A\}$, say $x_1(u) = f(X_u)$. (If $f(u) = f(v)$, then we separate $x_1(u)$ and $x_1(v)$ by an infinitesimal amount, arbitrarily choosing which one to be first.) Then for any $\varepsilon > 0$ there is a graph G and an ordering x_0 such that $f(X_u) < f(X_v)$ and $c_{uv} > (3 - \varepsilon)c_{vu}$ for some u, $v \in L_1$.*

Proof. Suppose f and ε are such that this is false. Define $N_{u_1}, N_{u_2}, \ldots$ and $N_{v_1}, N_{v_2}, \ldots$ as follows. Given numbers $1 = r_1, r_2, \ldots, r_t$, and k (which we choose appropriately later) define sets of real numbers $P_1, \ldots, P_t$, $Q_1, \ldots, Q_t$, $S_1, \ldots, S_t$ and $T_1, \ldots, T_t$ with

$$P_t < P_{t-1} < \cdots < P_1 = Q_1 < Q_2 \cdots < Q_t < S_t < S_{t-1} \cdots < S_1 \\ = T_1 < T_2 < \cdots < T_t$$

(where $A < B$ means $\max A < \min B$) and with $|P_i| = |T_i| = k$ for every i, and $|Q_i| = |S_i| = r_i k$ for all i. Identify each real number chosen with a corresponding node in L_0 with that x-coordinate.

Set $N_{u_i} = P_i \cup S_i$ and $N_{v_i} = Q_i \cup T_i$ for every i. (In particular $N_{u_1} = N_{v_1}$.) This defines our graph G (see Figure 8). With a judicious choice of $r_2, \ldots, r_t$ we get a contradiction, as follows.

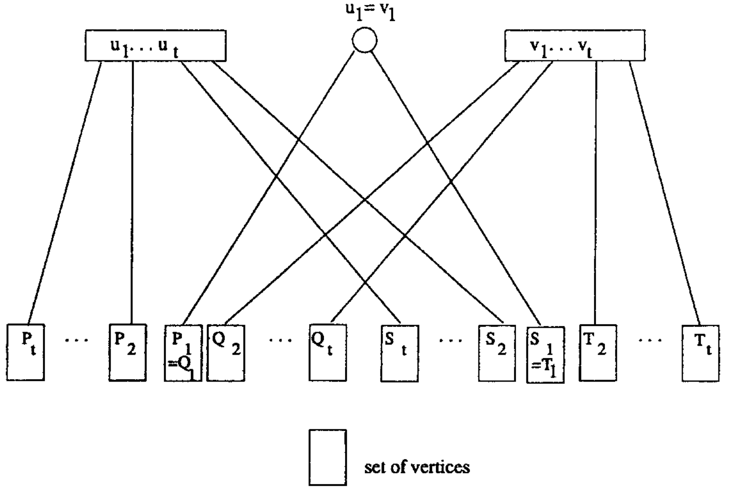


Fig. 8. The groups P_i, Q_i, S_i, and T_i.

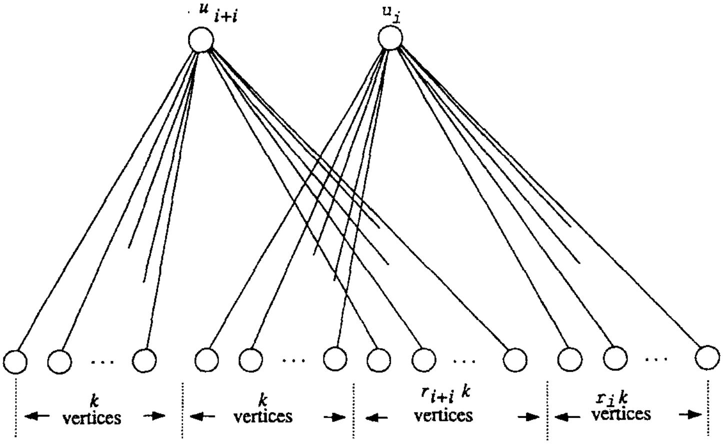


Fig. 9. Crossings for $c_{u_{i+1}u_i}$.

Let $i > 2$. Then $c_{u_{i+1}u_i} = k^2 r_{i+1}$ and $c_{u_i u_{i+1}} = k^2(r_i + r_i r_{i+1} + 1)$. (See Figure 9.) So

$$\frac{c_{u_i u_{i+1}}}{c_{u_{i+1}u_i}} = \frac{r_i + r_i r_{i+1} + 1}{r_{i+1}}.$$

Suppose we have chosen $r_2, \ldots, r_t$ so that the above ratio is, for every i, at least $3 - \varepsilon$. Then, by our assumption, $f(N_{u_i}) > f(N_{u_{i+1}})$. A similar argument shows $f(N_{v_i}) < f(N_{v_{i+1}})$. Hence $f(N_{v_t}) > f(N_{u_t})$.

However, $c_{u_t v_t}/c_{v_t u_t} = r_t^2/(2r_t + 1)$. (See Figure 10.) So if in addition $r_t^2/(2r_t + 1) > 3 - \varepsilon$, we have $f(N_{u_t}) > f(N_{v_t})$, and this is a contradiction.

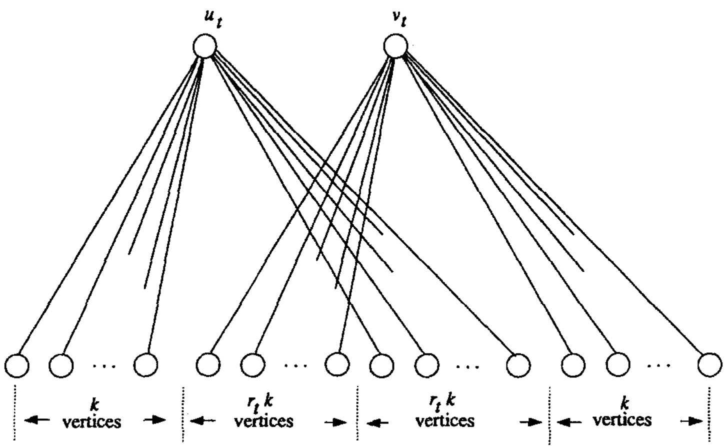


Fig. 10. Crossings for $c_{u_t v_t}$.

It remains to find $r_2, \ldots, r_t$ and k so that $r_i k$ is an integer for every i, and

(xvi) $$\frac{1 + r_i + r_i r_{i+1}}{r_{i+1}} > 3 - \varepsilon$$

for all i, and

(xvii) $$\frac{r_t^2}{2r_t + 1} > 3 - \varepsilon.$$

First we find reals r_i satisfying (xvi) and (xvii) (with $r_1 = 1$); then k can be taken large enough, and the r_i changed slightly, so as to satisfy the integrality condition without affecting (xvi) and (xvii).

To get (xvi) we need $r_{i+1} < 1 + r_i/(3 - \varepsilon - r_i)$. However,

$$\frac{1 + r_i}{3 - \varepsilon - r_i} - r_i = \frac{r_i^2 + r_i(\varepsilon - 2) + 1}{3 - \varepsilon - r_i} \geq \frac{\varepsilon(1 - \varepsilon/4)}{3 - \varepsilon - r_i} > \frac{\varepsilon(1 - \varepsilon/4)}{2}$$

(taking $r_i \geq 1$ and minimizing the quadratic).

Here we assume $r_i < 3 - \varepsilon$, since otherwise (xvi) is always satisfied.

It follows that we can set $r_{i+1} = r_i + \varepsilon(1 - \varepsilon/4)/2$ for each i, and (xvi) will be satisfied. So in a finite number of steps we can make r_t as large as we like (by taking t large). In particular if $r_t > 7$, then (xvii) is satisfied. □

Note, however, that this proposition does not cover some of the methods of [5].

5. Minimizing Length. The following observation suggests that the median heuristic gives a drawing in which the total length of the edges is close to minimal. Consider the function $g(a) = \sum_{v \in N_u} |a - x(v)|$. Note that g is continuous and piecewise linear, and decreasing if

$$|\{v : a > x(v)\}| > |\{v : a < x(v)\}|,$$

and increasing if

$$|\{v : a > x(v)\}| < |\{v : a < x(v)\}|.$$

Thus it is minimized precisely when a is the median of $\{x(v) : uv \in A\}$. However, $g(a)$ is rather close to the sum of the lengths of edges incident with u if it is placed at a. This intuition leads to the following theorem.

THEOREM 6. *The median heuristic gives a drawing whose total length is at most* $(2/\sqrt{3})$ *times the minimal length.*

PROOF. Suppose that $u \in L_1$. The length of an edge uv with one end at the point $(a, 0)$ and the other end at $(x(v), 1)$ is $\sqrt{(a - x(v))^2 + 1}$. Let $f(a)$ denote $\sum_{uv \in A} \sqrt{(a - x(v))^2 + 1}$. Note that $f(a)$ represents the sum of the lengths of all edges uv with $v \in N_u$ when $x(u) = a$. Denote the median of $\{x(v)\colon uv \in A\}$ by m.

For minimal total length, the best place to put the node u is at the point $a = t$ which minimizes $f(a)$. Note that the point t, or at least a close approximation to t, can be computed by the usual analytical methods. However, we cannot guarantee that $x(u) = t$ gives a small number of crossings; on the other hand $x(u) = m$ gives both a small number of crossings and a small total length.

We show that $f(m)/f(t) < 2/\sqrt{3}$ for each $u \in L_1$; the theorem follows. We assume that $t < m$; if $t = m$ we are done, and the case $t > m$ is symmetrical with $t < m$.

We want an upper bound on $f(m)/f(t)$. Suppose that $x(v) > m$ for some $v \in N_u$. Then moving v so that $x(v) = m$ decreases $f(t)$ more than it does $f(m)$. Hence for the situation in which $f(m)/f(t)$ is as big as possible, we can assume $x(v) \le m$ for every $v \in N_u$.

Let N^- be those $v \in N_u$ with $x(v) < m$. Since m is the median of N_u, $|N^-| < |N_u|/2$. However, $v \in (N_u - N^-)$ contributes more to $f(t)$ than to $f(m)$, so we have

$$\frac{f(m)}{f(t)} < \frac{\sum_{v \in N^-} \left(1 + \sqrt{(m - x(v))^2 + 1}\right)}{\sum_{v \in N^-} \left(\sqrt{(t - m)^2 + 1} + \sqrt{(t - x(v))^2 + 1}\right)}$$

by including only $|N^-|$ nodes of $N - N^-$ in the computations.

Hence

$$\frac{f(m)}{f(t)} < M, \qquad \text{where} \quad M = \max_{y<m}\left(\frac{1 + \sqrt{(m - y)^2 + 1}}{\sqrt{(t - m)^2 + 1} + \sqrt{(t - y)^2 + 1}}\right).$$

Suppose this maximum is achieved at $y = w$. The denominator

$$\left(\sqrt{(t - w)^2 + 1} + \sqrt{(t - m)^2 + 1}\right)$$

of M is minimized (over t) when $t = (m + w)/2$. It follows that if $z = m - w$, then $M \le (1 + \sqrt{z^2 + 1})/\sqrt{4 + z^2}$. Let $g(z) = (1 + \sqrt{z^2 + 1})/\sqrt{4 + z^2}$. To maximize $g(z)$, we solve $g'(z) = 0$ and obtain $z = 0$ or $z = \sqrt{8}$. A simple computation reveals that $(f(m)/f(t)) \le M \le 2/\sqrt{3}$. □

The barycenter heuristic does not have such a good bound on the lengths. Consider the example in Figure 11, where k is large and K is $\Omega(k^2)$. Here the total length of the drawing computed by the barycenter heuristic is almost twice the minimal length. However, we conjecture that Figure 11 is a worst case for the barycenter heuristic, that is, the total length of the drawing computed by the barycenter heuristic is at most twice the minimal length.

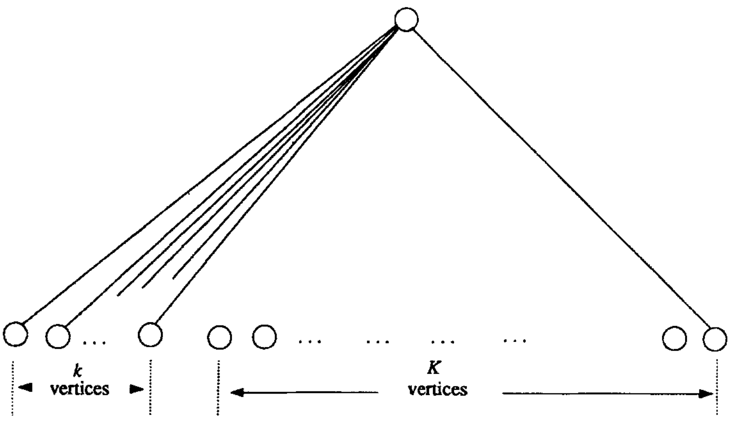


Fig. 11. Poor performance of barycenter heuristic.

6. Final Remarks. In Section 3 it was shown that minimizing the number of crossings in a layered network by a "level-by-level sweep" is intrinsically difficult, and in Section 4.2 we showed that the barycenter heuristic can perform very badly. In contrast, we proved in Section 4.1 that the median heuristic comes within a constant factor of optimal. In Section 5 we proved a bound on the total length of the arcs used in a median drawing, and pointed out that the barycenter heuristic does not achieve this bound.

However, many researchers in combinatorial optimization have pointed out that one should be very careful in interpreting worst-case results such as these (see, for example, [18]). For the problem at hand, several independent researchers (Makinen [17], Catarci (private communication), and Kelly [15]) have tested both the median and barycenter heuristics. These tests suggest that, for "random" bipartite graphs, the performance of the heuristics is very similar, and that the barycenter heuristic is slightly better. The number of "real-world" examples for which the two heuristics have been compared is quite small and it is difficult to make conclusions about their practical performance.

Since the first version of this paper [11] several authors have empirically investigated combinations of the median and barycenter heuristics: see, for example, [14] and [17]. These "hybrid" methods seem to be successful. Some of the bounds in Sections 4.1 and 4.2 of this paper, as well as the proposition in Section 4.3, apply to these methods. However, a full and rigorous analysis would be useful.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[2] G. Di Battista and E. Nardelli, *An Algorithm for Testing Planarity of Hierarchical Graphs*, Lecture Notes in Computer Science, Vol. 246, Springer-Verlag, Berlin, 1987, pp. 277–289.

[3] B. Berger and P. Shor, Approximation Algorithms for the Maximum Acyclic Subgraph Problem, *Proceedings of the First Annual ACM–SIAM Symposium on Discrete Algorithms*, San Francisco, 1990, pp. 236–243.

[4] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, New York, 1976.

[5] P. Eades and D. Kelly, Heuristics for Reducing Crossings in 2-Layered Networks, *Ars Combinatoria* **21A** (1986), 89–98.

[6] P. Eades and X. Lin, Notes on the Layer Assignment Problem for Drawing Directed Graphs, *Australian Computer Science Communications* **13**(1) (1991), 26-1–26-10.

[7] P. Eades, B. D. McKay, and N. Wormald, On an Edge Crossing Problem, *Proceedings of the Ninth Australian Computer Science Conference*, Australian National University, 1986, pp. 327–334.

[8] P. Eades, W. Smyth, and X. Lin, Heuristics for the Feedback Arc Set Problem, Technical Report 1-1989, School of Computing Science, Curtin University of Technology, 1989.

[9] P. Eades and K. Sugiyama, How to Draw a Directed Graph, *Journal of Information Processing* **13**(4) (1990), 424–437.

[10] P. Eades, R. Tamassia, G. Di Battista, and I. Tollis, Algorithms for Drawing Graphs: an Annotated Bibliography, available as /pub/gdbiblio.tex.z from wilma.cs.brown.edu (to appear in *Computational Geometry: Theory and Applications*).

[11] P. Eades and N. Wormald, The Median Heuristic for Drawing 2-Layered Networks, Technical Report 69, Department of Computer Science, University of Queensland, 1986.

[12] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, CA, 1979.

[13] M. R. Garey and D. S. Johnson, Crossing Number is NP-Complete, *SIAM Journal on Algebraic and Discrete Methods* **4**(3) (1983), 312–316.

[14] E. R. Gasner, S. C. North, and K. P. Vo, DAG—A Program that Draws Directed Graphs, *Software Practice and Experience* **18**(11) (1988), 1047–1062.

[15] D. Kelly, A View to Graph Layout Problems, Masters Thesis, Department of Computer Science, University of Queensland, 1987.

[16] D. Kelly, Fundamentals of Planar Ordered Sets, *Discrete Mathematics* **63** (1987), 197–216.

[17] E. Makinen, Experiments of Drawing 2-Level Hierarchical Graphs, Technical Report A-1988-1. Department of Computer Science, University of Tampere, Finland, 1988.

[18] D. J. Rosencrantz, R. E. Stearns, and P. M. Lewis, An Analysis of Several Heuristics for the Traveling Salesman Problem, *SIAM Journal of Computing* **6** (1977), 563–581.

[19] L. A. Rowe, M. Davis, E. Messinger, C. Meyer, C. Spirakis, and A. Tuan, A Browser for Directed Graphs, *Software Practice and Experience* **17**(1) (1987), 61–76.

[20] K. Sugiyama, A Readability Requirement in Drawing Digraphs: Level Assignment and Edge Removal for Reducing the Total Length of Lines, Research Report 45, International Institute for Advanced Study of Social Information Science, Numazu, Japan, 1984.

[21] K. Sugiyama, Drawing and Understanding Systems Structures: An Introduction to the SKETCH System, Working Paper WP-82-97, International Institute for Systems Analysis, Laxenburg, Austria, p. 52, 1982.

[22] K. Sugiyama, S. Tagawa, and M. Toda, Methods for Visual Understanding of Hierarchical System Structures, *IEEE Transactions on Systems, Man and Cybernetics* **11**(2) (1981), 109–125.

[23] R. Tamassia, C. Batini, and G. Di Battista, Automatic Graph Drawing and Readability of Diagrams, *IEEE Transactions on Systems, Man and Cybernetics* **18** (1988), 61–79.

[24] R. Tamassia, Drawing Algorithms for Planar st-Graphs, *Australian Journal of Combinatorics* **2** (1990), 217–236.

[25] H. Trickey, DRAG: A Graph Drawing System, in *Document Manipulation and Typography* (Proceedings of the International Conference on Electronic Publishing, Nice, 1988) (ed. J. C. van Vliet), Cambridge University Press, Cambridge, 1988, pp. 171–182.

[26] J. N. Warfield, Crossing Theory and Hierarchy Mapping, *IEEE Transactions on Systems, Man and Cybernetics* **7** (1977), 502–523.