EDGE SEPARATORS OF PLANAR AND OUTERPLANAR GRAPHS WITH APPLICATIONS

Krzystof Diks, Hristo N. Djidjev,
Ondrej Sykora and Imrich Vrto

SCS-TR-176, MAY 1990

School of Computer Science, Carleton University
Ottawa, Canada, KIS 5B6

EDGE SEPARATORS OF PLANAR AND OUTERPLANAR GRAPHS WITH APPLICATIONS *

Krzystof Diks§ , Hristo N. Djidjev†, Ondrej Sýkora‡, and Imrich Vrťo‡

Abstract. We consider the problem of finding a small set of edges whose removal divides a given graph into roughly equal parts. We show that every planar graph with n vertices and a maximal degree k has an $O(\sqrt{kn})$-edge separator, which bound is asymptotically optimal. This improves the previously best known upper bound of $O(k\sqrt{n})$. We prove that the bisection width of the class of planar graphs and trees of n vertices and degree k is $O(\sqrt{kn})$ and $O(k \log n/\log k)$ respectively, both optimal within a constant factor. All proofs are constructive and yield optimal $O(n)$ algorithms. The separator theorems are applied to embed efficiently planar and outerplanar graphs into binary trees, hypercubes, butterfly graphs, CCC graphs, shuffle-exchange graphs, de Brujn graphs, and d-dimensional meshes. The embeddings are shown to be optimal or almost optimal with respect to the average dilation and the expansion of the embedding.

1. Introduction.

One basic technique in the design of efficient algorithms on graphs is divide-and-conquer. For planar graphs one uses the fact that there exists a small subset of vertices or edges whose removal divides the graph into roughly equal pieces. Formally, we say an n-vertex graph G has an *$f(n)$-vertex separator* if there exists a partition of its vertices into three sets A, B, and C such that $|C| \leq f(n)$, $|A| \leq 2n/3$, $|B| \leq 2n/3$, and no edge joins a vertex from A with a vertex from B. It is known that each tree has an 1-vertex separator and each outerplanar graph has a 2-vertex separator. Lipton and Tarjan [20] showed that planar graphs have $\sqrt{8n}$-vertex separators. Djidjev [6] improved this to $\sqrt{6n}$-vertex separator. Vertex separator of planar graphs were extended to graphs of genus g by Djidjev [8] with $f(n) = \sqrt{6(2g+1)n}$ and an $O(n)$-time algorithm for finding an $O(\sqrt{(g+1)n})$-vertex

---

* A preliminary version of this paper was presented at the MFCS'88 conference

§ Institute of Informatics, Warsaw University, PKiN, 00901 Warsaw, Poland

† Center of Informatics and Computer Technology, Bulgarian Academy of Sciences, Acad G. Bonchev str. bl. 25-A, 1113 Sofia, Bulgaria

‡ Computing Centre, Slovak Academy of Sciences, Dúbravská 9, 842 35 Bratislava, Czechoslovakia

separator was proposed [7]. Independently, Gilbert, Hutchinson, and Tarjan [10] proved a $6\sqrt{(g+1)n}$-vertex separator theorem. Miller [22] showed that every 2-connected planar embedded graph has an $2\sqrt{dn}$-simple cycle vertex separator, where d is the maximum face size. Gazit and Miller [13] constructed a parallel algorithm for finding $O(\sqrt{n})$-vertex separators of planar graphs. Applications of vertex separators include solving sparse systems of linear equations [17,11], NP-complete problems [1], network flow problems [16], and many others [21,26,27].

We say an n-vertex graph G has an *f(n)-edge separator*, if there exists a set of *edges* C and a partition of the vertices into two sets A and B, such that $|C|\leq f(n)$, $|A|\leq 2n/3$, $|B|\leq 2n/3$, and each edge joining a vertex from A with a vertex from B belongs to C. Edge separators have applications to graph layouts for VLSI [18,19,31]. However, only graphs with vertex degrees bounded by a constant were studied in those applications. On the other hand, notice that general planar graphs do not have $o(n)$-edge separator. For example, a minimum edge separator of a star of degree n-1 has size $\lceil n/3 \rceil$. Thus it seems that the minimum size of an edge separator, unlike the case of an vertex separator, essentially depends on the degree of the graph. The best previously known result is the existence of $O(k\sqrt{n})$-edge separators for planar n-vertex graphs of degree k proved by Lipton and Tarjan [20]. In this paper we improve this bound to $O(\sqrt{kn})$ and show that the corresponding edge separator can be found in $O(n)$ time. Next we prove that any n-vertex outerplanar graph of a maximal degree k can be divided into subgraphs of $\leq\lceil n/2 \rceil$ vertices by removing $O(k \log (n/k))$ edges. For the case of trees this bound can be improved to $O(k \log n/\log k)$, which is shown to be optimal within a constant factor. A linear algorithm for finding the separator is described. As a consequence of these results, we obtain estimations on the cutwidth and the bisection width of planar graphs and trees of n vertices and degree k. In Section 3, we consider applications of the edge separator theorems for embedding planar and outerplanar graphs into a binary tree, a hypercube, a butterfly graph, a CCC graph, a FFT graph, a shuffle-exchange graph, a 1-dimensional and a 2-dimensional mesh. All of the embeddings are proved to be optimal or almost optimal with respect to the average dilation and the expansion.

2. Separation of planar graphs

In this section we are going to prove the existence of an $O(\sqrt{kn})$-edge separator of planar graphs. Then we estimate the bisection width of the class of n-vertex planar graphs of degree k. We will make use of the following theorem proved by Miller [22].

Theorem 2.1.[22] Let G be an embedded n-vertex biconnected planar graph, # be an assignment of nonnegative weights to the vertices, edges, and faces of G which sum to no more than 1, and let no face has weight >2/3. Denote by d the maximum face size of G. Then there exists a simple cycle of size $2\sqrt{2\lfloor d/2 \rfloor n}$ whose removal divides the graph into two parts of weight ≤2/3 each. Furthermore such a cycle can be constructed in $O(n)$ time.

Next we prove the main result of this section.

Theorem 2.2. Let G be an n-vertex planar graph of degree k. Then G has an edge separator of size $\leq 2\sqrt{2kn}$. Furthermore such a separator can be constructed in $O(n)$ time.

Proof. Embed G in the plane and construct the dual graph G' of G. Let n, m, f and n', m', f' be the number of vertices, edges, and faces of G and G' respectively. Then $f'=n$, $m'=m$ and $n'=f$. From the Euler's formula $n-m+f=2$. Since $2m \geq 3f$, then $f \leq 2n-4$, whence $n' \leq 2n-4$. Assign a weight $1/n$ to each vertex of G.

Suppose that G is biconnected. Then its dual, G', is also biconnected. Assign a weight to each face of G' equal to the weight of the corresponding vertex of G and a weight 0 to each vertex and each edge of G'. Apply Theorem 2.1 to G' and denote by c' the corresponding separating cycle. Then c' contains no more than $2\sqrt{2\lfloor k/2 \rfloor n'}$ vertices of G'. One can easily notice that the corresponding set C of edges from c' forms a separator of G. Moreover, $|C| \leq 2\sqrt{2\lfloor k/2 \rfloor n'} \leq 2\sqrt{2\lfloor k/2 \rfloor (2n-4)}$ $< 2\sqrt{2kn}$

Assume that G is not biconnected. Let $B_1,\ldots,B_b$ be the biconnected components of G. Consider the graph $H(G)$ containing a vertex v_i for each B_i, $1 \leq i \leq b$, and an edge joining v_i and v_j iff B_i and B_j share a common vertex. Assign a weight

$$wt(v_i) = \sum_{w \in V(B_i)} 1/z(w)$$

to v_i, where $z(w)$ denotes the number of biconnected components of G to which w belongs. Then $\sum_{i=1}^{b} wt(v_i)=n$. Let $F(G)$ be any spanning forest of $H(G)$. There exists a vertex v_j of $F(G)$ whose removal divides $F(G)$ into connected components of weight not exceeding $(2/3)n$. Then removal of B_j from G leaves no connected component of more than $(2/3)n$ vertices. Identify each component of $G-B_j$ with the corresponding articulation point of B_j. Let $K_1,\ldots,K_r$ be the connected

components of $G\text{-}B_j$ corresponding to an articulation point AP. Assign a weight $(|K_1|+...+|K_r|+1)/n$ to AP. If the weight of some vertex AP^* in the modified graph has become greater than 2/3 (this will be possible if AP^* is an articulation point belonging to more than 2 biconnected components), then C can be chosen to be the set of all edges incident to AP^*, whence $|C| \leq k \leq \sqrt{kn} \leq 2\sqrt{2kn}$. Otherwise assign a weight $1/n$ to each vertex of B_j that is not an articulation point and apply to B_j the proof for the case of biconnected graph.
Since embedding the graph in the plane and constructing its dual requires $O(n)$ time, then by Theorem 2.1 the edge separator of G can be found in $O(n)$ time. □

The *bisection width* of a graph G is the minimum number of edges whose removal divides G into two parts whose number of vertices differ at most by one. We will call such a set of edges an *edge bisector* hereafter. The problem of finding the bisection width is known to be NP-complete [12]. Next we will obtain an (optimal up to a constant factor) estimation of the bisection width for the class of planar graphs.

Corollary 2.1. For any n-vertex planar graph G of degree k there exists a set C of $\leq(6\sqrt{2}+4\sqrt{3})\sqrt{kn}$ edges whose removal divides the vertices of G into sets A and B such that $|A| \leq \lceil n/2 \rceil$, $|B| \leq \lceil n/2 \rceil$, and each edge connecting A and B belongs to C.

Proof: Follows directly from Theorem 2.2 by divide-and-conquer (see [20]). □

Corollary 2.2. The bisection width of any n-vertex planar graph of degree k does not exceed $((6\sqrt{2}+4\sqrt{3})\sqrt{kn}$.

Corollary 2.2 is an improvement over the $O(k\sqrt{n})$ result mentioned in [14].

Remark. The constant before $\sqrt{kn}$ in Corollary 2.1 and 2.2 can be further reduced by using the more precise techniques of [6] and [32,33].

The above results can also be used to obtain an optimal estimation of the cutwidth of planar graphs. The cutwidth problem arises in the realization of fault-tolerant arrays of VLSI processors. Formally, a *linear layout* L of a graph G is a one-to-one function from V to $\{1,...,n\}$. The *cut* of L at a point p of the real line is the number of edges connecting vertices on left with vertices on right of p. The *cutwidth of* L is the maximum cut of L over all real points. The *cutwidth of* G is the minimum cutwidth of a linear layout of G.

Corollary 2.3. The cutwidth of any n-vertex planar graph of degree k is $O(\sqrt{kn})$.

Proof: By breaking the graph recursively into smaller parts using Theorem 2.2 one can easily construct a layout with cutwidth $O(\sqrt{kn})$. □

An $O(k\sqrt{n})$ bound on the cutwidth of n-vertex planar graphs of degree k is mentioned in [14]. For obtaining a lower bound on the minimum size of the edge separator and of the cutwidth we will apply the method of Leighton based on the following statement.

Lemma 2.1.[18] Consider an embedding of the complete n-vertex graph K_n into an n-vertex graph G so that the edges of K_n are mapped into paths in G. Let each edge of G is utilized by at most q paths. Then the minimum edge separator of G contains at least $2n^2/(9q)$ edges.

Theorem 2.3. Theorem 2.2 is tight to within a constant factor.

Proof: To prove the theorem we will show that for any $k>0$ such that $k=0 \pmod 4$ there exists an infinite sequence $\{G_i\}$ of planar graphs of degree k such that $n_1<n_2<\ldots$, where n_i is the number of the vertices of G_i, and the minimum edge separator of G_i, is of size at least $\sqrt{kn_i}/9$. The sequence of graphs can be constructed in the following way. Consider a mesh of size $p \times p$. Replace each edge by $k/4$ new edges and on each edge place exactly one new vertex. We get a graph G_p of degree k and $n=n_p=p^2+(p-1)p(k/2)$ vertices. Each two neighbour mesh vertices create with the additional vertices adjacent to both of them a *hammock*.

Indicate the hammocks as follows:

- $H_{i,j+1}$ be the hammock between the mesh vertices (i,j) and $(i,j+1)$ including the vertex $(i,j+1)$ and except the vertex (i,j);
- $F_{i,j}$ be the hammock between the mesh vertices (i,j) and $(i+1,j)$ including the vertex (i,j) and except the vertex $(i+1,j)$;
- $H_{i,1}$ be the mesh vertex $(i,1)$, $1 \le i \le p$;
- $F_{p,i}$ be the mesh vertex (p,i), $1 \le i \le p$.

To apply Lemma 2.1 we will construct a suitable mapping of vertices and edges of K_{np} into vertices and paths of G_p. An edge of K_{np} joining a vertex from H_{i_1,j_1} and a vertex from F_{i_2,j_2} is mapped to a path p in G_p defined as follows:

- if $i_1 \le i_2$ and $j_1 \le j_2$, then p traverses hammocks
  $H_{i_1,j_1}, H_{i_1,j_1+1}, \dots, H_{i_1,j_2}, F_{i_1,j_2}, F_{i_1+1,j_2}, \dots, F_{i_2,j_2}$;

- if $i_1 \le i_2$ and $j_1 > j_2$, then p traverses hammocks
  $F_{i_2,j_2}, H_{i_2,j_2}, H_{i_2,j_2+1}, \dots, H_{i_2,j_1}, F_{i_2,j_1}, F_{i_2-1,j_1}, \dots, F_{i_1,j_1}, H_{i_1,j_1}$;

- if $i_1 > i_2$ and $j_1 \le j_2$, then p traverses hammocks
  $H_{i_1,j_1}, H_{i_1,j_1+1}, \dots, H_{i_1,j_2}, F_{i_1-1,j_2}, F_{i_1-2,j_2}, \dots, F_{i_2,j_2}$;

- if $i_1 > i_2$ and $j_1 > j_2$, then p traverses hammocks
  $F_{i_2,j_2}, H_{i_2,j_2}, H_{i_2,j_2+1}, \dots, H_{i_2,j_1}, F_{i_2,j_1}, F_{i_2+1,j_1}, \dots, F_{i_1-1,j_1}, H_{i_1,j_1}$.

Further, an edge of K_{np} joining a vertex from H_{i_1,j_1} and a vertex from H_{i_2,j_2} is mapped to a path p defined as follows: (Without a loss of generality suppose that $i_1 \le i_2$.)

- if $j_1 \le j_2$ then p traverses hammocks
  $H_{i_{11},j_1}, H_{i_1,j_1+1}, \dots, H_{i_1,j_2}, F_{i_1,j_2}, F_{i_1+1,j_2}, \dots, F_{i_2-1,j_2}, H_{i_2,j_2}$;

- if $j_1 > j_2$ then p traverses hammocks
  $H_{i_1,j_1}, F_{i_1,j_1}, F_{i_1+1,j_1}, \dots, F_{i_2-1,j_1}, H_{i_2,j_1}, H_{i_2,j_1+1}, \dots, H_{i_2,j_2}$.

Finally, an edge of K_{np} joining a vertex from $F_{i_1,j_{11}}$ and a vertex from F_{i_2,j_2} is mapped to a path p defined as follows: (Without a loss of generality suppose that $i_1 \le i_2$.)

- if $j_1 \le j_2$ then p traverses hammocks
  $F_{i_1,j_1}, H_{i_1,j_1}, H_{i_1,j_1+1}, \dots, H_{i_1,j_2}, F_{i_1,j_2}, F_{i_1+1,j_2}, \dots, F_{i_2,j_2}$;

- if $j_1 > j_2$ then p traverses hammocks
  $F_{i_1,j_1}, F_{i_1+1,j_1}, \dots, F_{i_2,j_1}, H_{i_2,j_1}, H_{i_2,j_1-1}, \dots, H_{i_2,j_2}, F_{i_2,j_2}$.

To complete the definition of the mapping, an edge of K_{np} joining two nonadjacent vertices from either $H_{i,j}$ or $F_{i,j}$ is mapped to a path of length 2: from the first endpoint to a mesh vertex (i,j) and then to the second endpoint.

It is evident that q paths traversing a hammock can be distributed in the hammock so that every edge is utilized by at most $4q/k$ paths. From the definition of the paths it follows that the most utilized are the hammocks $F_{\lceil p/2\rceil-1,p}$ and $F_{\lceil p/2\rceil,p}$, if p is odd and $F_{p/2,p}$, if p is even. In both cases the above hammocks are traversed by less than $npk/4$ paths and hence each edge is utilized by $q<np$ paths. Then by Lemma 2.1 one gets for the minimum size of an edge separator a lower bound of $2n^2/(9q)>2n/(9p)>\sqrt{nk}/9$. □

Corollary 4.2. The cutwidth of the graph defined in Theorem 2.3 is $\Omega(\sqrt{kn})$.

Proof: It is easy to see that the cutwidth of any graph is not smaller than its edge separator. □

3. Separation of trees and outerplanar graphs

It is well known that any tree has a 1-vertex separator [20] and any outerplanar graph has a 2-vertex separator [14]. Obviously, this yields a k-edge separator for trees of degree k and a $2k$-edge separator for outerplanar graphs of degree k. To be able to estimate the bisection width of trees and outerplanar graphs we have to obtain bounds on the size of a set of edges dividing the original graph into two subgraphs of not more than $\lceil n/2\rceil$ vertices each. By the above mentioned method [20, Corollary 3] one can get an $O(k \log (n/k))$-edge bisector for both trees and outerplanar graphs. Below we show that for the case of trees it is possible to improve this bound to $O(k \log n/\log k)$, which is proved to be tight upto a constant factor.

Lemma 3.1. Let T be an n-vertex tree of degree k and p be an integer, $1\le p\le n/2$. Then one of the following cases holds:

(i) T can be separated into two subgraphs T_1, T_2 such that $|V(T_1)|=n-p$, $|V(T_2)|=p$ by deleting a single edge;

(ii) T can be separated into three subgraphs T_1, T_2, T_3 such that $|V(T_1)|\le n-p$, $|V(T_2)|\le p$, $|V(T_3)|\le n/l$, whereby T_3 is a tree, by deleting at most l edges, for some integer l, $3\le l\le k$.

Proof. Find a vertex $v\in T$ whose deletion separates T into trees $G_1, G_2, \ldots, G_r$, $k\ge r\ge 2$ such that $n-p\ge n_1\ge n_2\ge \ldots \ge n_r$ and $n_2\ge p$, where n_i denotes the number of the vertices of G_i.

If $r=2$, then the first case holds.
If $r=3$, we have either $n_1<n-p$, or $n_2<p$. Delete the edge joining v with G_3 and either the edge joining v with G_1 (if $n_2<p$), or the edge joining v with G_2 (if $n_1<n-p$).
Let $r\geq 4$. We will construct the graphs T_1 and T_2 iteratively, ensuring that after each iteration it holds $|V(T_1)|\leq n-p$, $|V(T_2)|\leq p$. Let l be the smallest integer such that $l\geq 3$ and $n_l\geq n_{l+1}+\ldots+n_r=n'_l$. Then $n_i<n_{i+1}+\ldots+n_r=n'_i$ for $3\leq i\leq l-1$.

We will prove by induction that for any j, $2\leq j\leq l-1$, the set $\{1,2,\ldots,j\}$ can be divided into two sets A^*, B^* such that

$$N_1^*=\sum_{i\in A^*} n_i\leq n-p,\ N_2^*=\sum_{i\in B^*} n_i\leq p .$$

The case $j=2$ is trivial. Let the claim be true for some $j\leq l-2$. Then either $N_1^*+n_{j+1}\leq n-p$, or $N_2^*+n_{j+1}\leq p$; otherwise

$$n-1=N_1^*+N_2^*+n_{j+1}+n'_{j+1}\geq N_1^*+N_2^*+2n_{j+1}>n.$$

Thus the claim is true for $j+1$ and consequently for all j, $2\leq j\leq l-1$. Notice that this proof yields an $O(l)$ algorithm for constructing the sets A^* and B^*, given the numbers $n_1,\ldots,n_{l-1}$. If we take $j=l-1$, the claim shows that there exists a partition of the set $\{1,\ldots,l-1\}$ into two sets A^*, B^* such that

$$N_1=\sum_{i\in A^*} n_i\leq n-p,\ N_2=\sum_{i\in B^*} n_i\leq p .$$

Then either $N_1^*+n'_l\leq n-p$, or $N_2^*+n'_l\leq p$; otherwise a contradiction as above will arise. Thus we have divided the set $\{1,\ldots,r\}-\{l\}$ into two sets A, B such that

$$N_1=\sum_{i\in A} n_i\leq n-p,\ N_2=\sum_{i\in B} n_i\leq p .$$

and either A or B contains the set $\{l+1,\ldots,r\}$. From the inequalities $n_1\geq\ldots\geq n_r$ and the choice of l,

$$ln_l\leq ln_{l-1}=(l-1)n_{l-1}+n_{l-1}\leq(n_1+\ldots+n_{l-1})+n_l+n'_l=n-1<n.$$

By definition, A contains the number 1 and B contains the number 2. Suppose that $\{l+1,\ldots,r\}\subset B$. Delete all edges joining v with G_i, for $i\in A\cup\{l\}$. Clearly, the number of those edges is at most $l-1$. Let T_1 be the graph induced by $\bigcup_{i\in A} V(G_i)$, T_2 be the graph induced by $\bigcup_{i\in B} V(G_i)\cup\{v\}$ and $T_3=G_l$. Evidently $|V(T_1)|\leq n-p$, $|V(T_2)|\leq p+1$. If $|V(T_2)|=p+1$, then we separate a leaf u of T_2 by deleting one edge and we add u to T_1. A similar proof applies if $\{l+1,\ldots,r\}\subset A$. □

Theorem 3.1. For any n-vertex tree T of degree k and any p, $1 \le p \le n/2$, there exists a partition A, B of the set of the vertices of T such that $|A|=n-p$, $|B|=p$, and any edge connecting a vertex from A with a vertex from B belongs to a set C of $\le k \log n / \log k$ edges.

Proof. For k=2 the theorem is obviously true. The set C can be found applying on T the following recursive procedure. (C is empty at the beginning.)

procedure SEPARATE (T,p)
begin
    if T is a star graph **then**
        $C=C \bigcup$ {any p edges of T}; **return**;
    fi;
    find a set M of edges and a partition T_1, T_2, T_3 of T satisfying
      Lemma 3.1 (possibly T_3 is a null tree);
    $C=C \cup M$;
    if $|M|=1$ (i.e. case (i) of Lemma 3.1 applies) **then return**;
    if $|M|>1$ **then**
        SEPARATE(T_3 , min{ $p-|V(T_2)|$, $n-p-|V(T_1)|$ });
end

We will show that the size of C satisfies the theorem. Let $k \ge 3$ and $c(n,k)$ denote the maximum size of a set C found by the algorithm when applied on any n-vertex tree of degree $\le k$ and for any p, $1 \le p \le n/2$. We will prove by induction on n that

$$c(n,k) \le k \log n / \log k.$$

Obviously, $c(k+1,k) \le k/2 < k \log (k+1)/\log k$. Assume that the claim is true for all j such that $k+1 \le j \le n-1$. Then

$$c(n,k) \le \max\{l+c(|V(T_3)|,k) : 3 \le l \le k\} \le \max\{l+c(\lfloor n/l \rfloor,k) : 3 \le l \le k\}$$
$$\le \max\{l+ k \log (n/l) / \log k) : 3 \le l \le k\}$$
$$\le k \log n/\log k + \max\{\log l \,(l/\log l - k/\log k) : 3 \le l \le k\} = k \log n/\log k. \ \square$$

Corollary 3.1. For any n-vertex tree T of degree k and there exists a partition A, B of the set of the vertices of T such that $|A| \le \lceil n/2 \rceil$, $|B| \le \lceil n/2 \rceil$, and any edge connecting a vertex from A with a vertex from B belongs to a set C of $\le k \log n/\log k$ edges.

Corollary 3.2. The bisection width of any n-vertex tree T of degree k is $\le k \log n/\log k$.

Theorem 3.2. The bisection width of n-vertex k-degree trees is $\Omega(k\log n/\log k)$.

Proof. We will show that for arbitrary $k\geq3$ there exists an infinite sequence $T_1, T_2,\ldots$ of trees of degree $\leq k$ such that each edge bisector of T_i is of size $\geq(1/10)k\log n_i/\log k$ and $n_1<n_2<\ldots$, where $n_i=|V(T_i)|$. Consider the following two cases with respect to k.

Case 1 : k=3. Let h be odd, $n_1=2^{h+1}-1$ and let $T(h)$ be the n_1-vertex complete binary tree. Denote $n=2n_1/3$. Let M be the minimum set of edges whose deletion divides $T(h)$ into three subforests F_0, F_1, F_2 of equal sizes. We will make use of the result of Chung and Rosenberg [5] that $|M/\geq(3/16)(h-19/16)$. Let M_i be the set of edges of M connecting $F_{(i+1) \bmod 3}$ and $F_{(i+2) \bmod 3}$, $i=0,1,2$. Clearly $M_0\cup M_1\cup M_2=M$, $M_i\cap M_j=\varnothing$, $i\neq j$. Let $|M_0|\geq|M_1|\geq|M_2|$.

Consider the forest $F_1\cup F_2$. We claim that the edge bisector of $F_1\cup F_2$ is of size at least $|M_0|$. In the contrary, let M_0^* be an edge bisector of $F_1\cup F_2$ and $|M_0^*|<|M_0|$. Then the set $M_0^*\cup M_1\cup M_2$ of edges also divides $T(h)$ into three subforests of equal sizes and $|M_0^*\cup M_1\cup M_2|<|M|$, which contradicts to the minimality of M.

Construct an n-vertex tree T of degree 3 by adding suitable edges to $F_1\cup F_2$. Evidently, the size of the minimal edge bisector of T is at least $|M_0|$. According to the choice of M_0,

$$|M_0|\geq|M|/3\geq(1/16)(h-19/6)=(1/16)(\log(3n/2+1)-25/6).$$

Case 2 : $k\geq4$ and k is even. We will apply the method of Rosenberg [28]. Let $T(h)$ denote the complete $(k-1)$-ary tree of height h and A and B be disjoint sets of no more than $\lceil n/2\rceil$ vertices of G each. Denote by C the set of edges joining a vertex from A with a vertex from B. Let $h\geq5$ and $n=((k-1)^{h+1}-1)/(k-2)$. Define the *level* of each vertex of $T(h)$ to be its distance from the root and the *level* of each edge to be the maximum of the levels of its two endpoints.

Let e_i denote the number of edges on level i that belong to C. Define the function

$$f(j)=\sum_{i=1}^{h-j} e_{j+i}(k-1)^{-i}\ ,\ j=1,\ \ldots\ ,h-1.$$

Lemma 3.2. For each j, $1 \le j < h/2$, it holds $f(j) \ge (1-(k-1)^{2j-h-1})/2$.

Proof: Assume that there exists j, $1 \le j < h/2$, such that
$f(j) < (1-(k-1)^{2j-h-1})/2$.
Let A_j and B_j denote the vertices of the j-th level that belong to A and B respectively. As k-1 is odd, then either $|A_j| \ge ((k-1)^j+1)/2$, or $|B_j| \ge ((k-1)^j+1)/2$. Without a loss of generality assume that $|A_j| \ge ((k-1)^j+1)/2$. Then we have

$$|A| \ge [((k-1)^j+1)/2].[((k-1)^{h-j+1}-1)/(k-2)] - \sum_{i=1}^{h-j} e_{j+i}\, ((k-1)^{h-j-i+1}-1)/(k-2).$$

The first term corresponds to the contribution to A of the level-j vertices and their subtrees, if no edge of C has level exceeding j, and the second term corresponds to the maximum number of vertices by which the first term could be diminished by considering also the edges of C at levels j+1, ... ,h. Then

$$2(k-2)|A| \ge ((k-1)^{h+1}-1) + (k-1)^{h-j+1} - (k-1)^j - (k-1)^{h-j+1} . \sum_{i=1}^{h-j} e_{j+i}$$

$$= ((k-1)^{h+1}-1) + (k-1)^{h-j+1} - (k-1)^j - (k-1)^{h-j+1} .f(j)$$

$$> ((k-1)^{h+1}-1) + (k-1)^{h-j+1} - (k-1)^j - (k-1)^{h-j+1}(1-(k-1)^{2j-h-1})/2$$

$= ((k-1)^{h+1}-1) = (k-2)n$, this being a contradiction. □

From Lemma 3.2

$$\sum_{j=1}^{\lfloor h/2 \rfloor} f(j) \ge \sum_{j=1}^{\lfloor h/2 \rfloor} (1-(k-1)^{2j-h-1})/2 > h/5.$$

On the other hand

$$\sum_{j=1}^{\lfloor h/2 \rfloor} f(j) \le \sum_{j=1}^{h-1} f(j) = \sum_{j=1}^{h-1} \sum_{i=1}^{h-j} e_{j+i}(k-1)^{-i} = \sum_{j=2}^{h} e_j \sum_{i=1}^{j-1} (k-1)^{-i}$$

$$= 1/(k-1) \sum_{j=2}^{h} e_j \sum_{i=0}^{j-2} (k-1)^{-i} \le 1/(k-1) \sum_{j=1}^{h} e_j \sum_{i=0}^{\infty} (k-1)^{-i}$$

$$\le 1/(k-1)\ |C|\ (k-1)/(k-2) = |C|/(k-2).$$

Thus $|C|>(k-2)h/5$. Since $k^h \geq n$, we obtain

$$|C|> (k-2)(\log n/\log k)/5 \geq (k \log n/\log k)/10.$$

Case 3 : $k \geq 4$ and k is odd. Then $k \geq 5$. Construct the complete $(k-2)$-ary tree of height h. Add to the tree a new vertex and connect it to a vertex at level h-1. We have constructed for $n=((k-1)^{h-1}-1)/(k-2)+1$ an n-vertex tree of degree k. Assume that this tree has an edge bisector of size $<((k-1)\log(n-1)/\log(k-1))/10 -2$. Then it is straightforward to construct an edge bisector of the complete $(n-1)$-vertex $(k-2)$-ary tree of size $\leq((k-1)\log(n-1)/\log(k-1))/10 - 1$. □

We can generalize Theorem 3.2 in order to prove that the worst-case minimum set of edges dividing an n-vertex k-degree graph G into two disjoint parts of $\lfloor pn \rfloor$ and $n-\lfloor pn \rfloor$ vertices is of size $\Omega(k\log n/\log k)$ for any p, $0<p\leq 1$. The proof uses a similar idea, but is technically more complicated.

4. Applications

In this section we will apply the edge separator theorems from the previous sections to obtain results on embedding planar graphs, outerplanar graphs, and trees on binary trees, linear arrays, meshes, hypercubes, butterflies, CCC graphs, perfect shuffle graphs, and de Brujn graphs. (For definition of these graphs one can see i.e. [24].

A large variety of important computational problems can be mathematically formulated as graph embedding problems. For instance, problems concerning representation of some kind of data structure by another kind of data structure [9], adaptation of the interconnection structure of a parallel algorithm differing from the interconnection structure of the used parallel computer [4], simulation of interconnection networks of parallel computers [3], and minimizing wire lengths in VLSI layouts [25].

Formally, an embedding of a graph G_1 into a graph G_2 is a mapping $\varphi : V(G_1) \rightarrow V(G_2)$. There are three widely used measures for estimating the properties of φ : the *worst dilation*, the *average dilation*, and the *expansion of* φ, defined as follows:

$$\mathrm{WDIL}(\varphi) = \max_{(u,v)\in E(G_1)} \{d2(\varphi(u),\varphi(v))\},$$

$$\mathrm{ADIL}(\varphi) = \sum_{(u,v)\in E(G_1)} d_2(\varphi(u),\varphi(v)) /|E(G_1)|,$$

$$\mathrm{EXP}(\varphi) = |V(G_2)|/|V(G_1)|,$$

where $d_2(x,y)$ denotes the distance between vertices x and y in G_2.

For results concerning the worst-case dilation one can consult [29]. In this paper we consider the average dilation and the expansion measures. At first we mention some related results. Lipton and Tarjan proved in [21] that every n-vertex planar graph of degree k can be embedded into a binary tree with average dilation $O(k)$ and expansion 1. De Millo, Eisenstat and Lipton [9] embedded the n-vertex 2-dimensional mesh into the n-vertex one dimensional mesh with optimal average dilation $\Theta(\sqrt{n})$. Scheidwasser [30] and Jordanskij [15] showed that every n-vertex tree of degree k is embeddable into a 1-dimensional mesh (a linear array) with average dilation $\Theta(k \log n/\log k)$. Lower bound techniques for the average dilation of embeddings of graphs into meshes were developed in [34].

4.1 Upper bounds

The divide-and-conquer method [21,15,30] is the standard technique for constructing efficient average dilation embeddings. Firstly, we apply the method on the problem of embedding planar graphs.

Theorem 4.1. Every planar graph can be embedded in a binary tree with average dilation $O(\log k)$ and expansion one.

Proof: Let G be an n-vertex planar graph of degree k. Define an embedding of the vertices of G in the vertices of a binary tree T by using the following recursive procedure. If G has one vertex v, embed v into a tree of one vertex, the image of v. Otherwise, apply Corollary 2.1 to find a partition A,B of $V(G)$ and a set of edges C. Let $|A|=\lfloor n/2\rfloor$ and $|C|\le c\sqrt{kn}$. Let v be a vertex of A. Embed the subgraph G_1 induced by A-$\{v\}$ into a binary tree T_1 by the same method recursively. Embed the subgraph G_2 of G induced by B into a binary tree T_2. Let T consists of a root (the image of v) with two children: the root of T_1 and the root of T_2.

Denote by $h(n)$ the maximum depth of a tree T of n vertices that can be produced by this algorithm. Then if $n>1$ $h(n) \le h(\lfloor n/2 \rfloor)+1$, whence $h(n) \le \log n$.
Let us define $s(G)= \sum_{(u,v) \in E(G)} d_2(\varphi(u),\varphi(v))$. Then

$s(G)=0$, if $n=1$,

$s(G) \le s(G_1)+s(G_2)+2c\sqrt{kn}\log n+k \log n$, if $n>1$.

This follows from the fact that for every edge $(u,w) \in G$ belonging neither to G_1, nor to G_2, it holds

$d_2(\varphi(u),\varphi(v)) \le 2 \log n$, if $(u,v) \in C$;

$d_2(\varphi(u),\varphi(v)) \le \log n$, if this edge is incident to the vertex v.

If $s(n)$ is the maximum value of $s(G)$ for all n-vertex planar graphs of degree k, then

$s(n) \le 4n \log n$, for $k<n \le 2k$,

$s(n) \le s(\lceil n/2 \rceil) + s(\lfloor n/2 \rfloor) + 2c\sqrt{kn}\log n + k \log n$, for $n>2k$.

A direct solution to this recurrence gives

$s(n) = O(n \log k - \sqrt{kn}\log n)$.

Since G is connected and has therefore at least n-1 edges, then

ADIL(φ)=$O(\log k)$. □

Theorem 4.1 improves the best previously known bound of $O(k)$ due to Lipton and Tarjan [21].

From the proof of Theorem 4.1 it is evident that the result holds for embedding an n-vertex planar graph into the *complete* n-vertex binary tree with expansion <2. Since this result will be used below, we give it in a separate statement.

Corollary 4.1. Every planar graph can be embedded into a complete binary tree with average dilation $O(\log k)$ and expansion one.

There is a number of results concerning embeddings with worst dilation $O(1)$ and expansion $O(1)$ of the complete binary tree into other regular structures. Such structures include the hypercube, the butterfly graph, the cube-connected-cycles (CCC) graph, the shuffle-exchange graph, and the de Brujn graph [2,3,24]. All these results can be used in combination with Corollary 4.1 to obtain efficient embeddings of planar graphs into the latter structures. In order to do this we make use of the following obvious fact.

Lemma 4.1. Let φ be an embedding of a graph G_1 into a graph G_2 and let ψ be an embedding of G_2 into a graph G_3. Then there exists an embedding ξ of G_1 into G_3 such that $\mathrm{ADIL}(\xi)\leq\mathrm{ADIL}(\varphi)\mathrm{WDIL}(\psi)$ and $\mathrm{EXP}(\xi)\leq\mathrm{EXP}(\varphi)\mathrm{EXP}(\psi)$.

Corollary 4.2. Every n-vertex planar graph of degree k can be embedded into a hypercube, a butterfly graph, a CCC graph, a shuffle-exchange graph, or a de Brujn graph, with average dilation $O(\log k)$ and expansion $O(1)$.

The method used in the proof of Theorem 4.1 can be applied with minor modifications to other embedding problems. In the proofs of the next theorems we will omit obvious details.

Theorem 4.2. Every n-vertex planar graph of degree k can be embedded in the 1-dimensional mesh with average dilation $O(\sqrt{kn})$ and expansion 1.

Proof: The corresponding recurrence relation gives

$s(n) \leq 3n^2$, for $k<n\leq 2k$,

$s(n)\leq s(\lceil n/2\rceil)+s(\lfloor n/2\rfloor)+c\sqrt{kn}\,n$, for $n>2k$.

The solution satisfies $s(n)=O(\sqrt{kn}\,n)$, which gives $\mathrm{ADIL}(\varphi)=O(\sqrt{kn})$. □

Theorem 4.3. Every n-vertex outerplanar graph of degree k can be embedded in the 1-dimensional mesh with average dilation $O(k\log(2n/k))$ and expansion 1.

Proof: We have the following recursive relation

$s(n) \leq 2n^2$, for $k<n\leq 2k$,

$s(n)\leq\max\{s(i)+s(n-i)\colon n/3\leq i\leq 2n/3\}$, for $n>2k$.

The solution satisfies $s(n)=O(nk\log(2n/k))$, which yields

$\mathrm{ADIL}(\varphi)=O(k\log(2n/k))$. □

Theorem 4.4. Every n-vertex planar graph of degree k can be embedded into a two dimensional mesh with average dilation $O(\sqrt{k}\log(2n/k))$ and expansion <2.

Proof: Assume for the sake of simplicity of notations that $n=4^i$. Choose a mesh of size $2^i \times 2^i$. The corresponding recurrence relation is

$s(n) \leq 6n\sqrt{n}$, for $k<n\leq 2k$,

$s(n)\leq 4s(n/4)+c\sqrt{kn}\,.\,2\sqrt{n}+c\sqrt{kn/2}\,.\,2\sqrt{n}$, for $n>2k$.

The solution of the recurrence satisfies $s(n)=O(n\sqrt{k}\log(2n/k))$, which yields $\mathrm{ADIL}(\varphi)=O(\sqrt{k}\log(2n/k))$. □

Theorem 4.5. Every n-vertex outerplanar graph of degree k can be embedded into a two dimensional mesh with average dilation $O(\sqrt{k})$ and expansion <2.

Proof: Assume that $n=4^i$. Choose a mesh of size $2^i \times 2^i$. The corresponding recurrence relation is

$s(n) \leq 4n\sqrt{n}$, for $k<n\leq 4k$,

$s(n)\leq 4s(n/4)+4k\sqrt{n}$, for $n>4k$.

The solution satisfies $s(n)=O(n\sqrt{k}\log(2n/k))$ yielding $\mathrm{ADIL}(\varphi)=O(\sqrt{k})$. □

Theorem 4.6. Every n-vertex planar graph of degree k can be embedded into a d-dimensional mesh, d≥3, with average dilation $O(dk^{1/d})$ and expansion<2 for sufficiently large n in comparison with d.

Proof: Let $n=2^{id}$. The recurrence relation is

$s(n) \leq 3ndn^{1/d}$, for $2^{d\lceil(\log k)/d\rceil}<n\leq 2^d k$,

$s(n)\leq 2^d s(n/2^d)+(2+\sqrt{2})cd\sqrt{kn}\; n^{1/d}$, for $n>2^d k$.

The solution satisfies $s(n)=O(dnk^{1/d})$ which implies $\mathrm{ADIL}(\varphi)=O(dk^{1/d})$.

If n is not of the form 2^{id} then we embed the graph into a mesh of size r^d, where r is the smallest integer such that $r^d\geq n$. This implies $\mathrm{EXP}(\varphi)\leq(1+1/n^{1/d})^d$, whence $\mathrm{EXP}(\varphi)$ is less than 2 for $n \gg d$. The proof is similar, but technically more complicated. □

Corollary 4.2. Every n-vertex outerplanar graph of degree k can be embedded into a d-dimensional mesh, d≥3, with average dilation $O(dk^{1/d})$ and expansion<2 for sufficiently large n in comparison with d.

4.2. Lower bounds

We start by describing two techniques for deriving lower bounds on the average dilation. The first method is a natural generalization of Theorem 2 from [9]. We state it without proof.

Lemma 4.2. Let φ be an embedding of an n-vertex graph G into the 1-dimensional mesh. Let $f_p(n)$ be the minimum cardinality of a set of edges that divides G into two parts of p and n-p vertices respectively. Then

$$\mathrm{ADIL}(\varphi)\geq\sum_{i=1}^{n/2} f_i(n) \,/\, |E(G)|.$$

The second method is a straightforward extension of Theorem 2 [34], originally proved for embeddings into meshes, applied to star graphs.

Lemma 4.3. Let φ be an embedding of the k-vertex star graph T_1 into a graph H. Define

$$r=\min_{H'}\{\mathrm{radius}(H'): H' \text{ is a } k/2\text{-vertex induced subgraph of } H\}.$$

Then $\mathrm{ADIL}(\varphi)\geq r/4$.

Theorem 4.7. Any embedding of the complete (k-1)-ary tree into a binary tree, a butterfly graph, a CCC graph, a shuffle-exchange graph, or a de Brujn graph requires average dilation $\Omega(\log k)$.

Proof. Consider CCC graphs. For other graphs the proof is similar, because it is essentially based on the fact that all these graphs are of bounded degree.

We proceed by induction on the height h of the complete (k-1)-ary tree T_h. Clearly $|V(T_h)|=((k-1)^{h+1}-1)/(k-2)$, $|E(T_h)|=|V(T_h)|-1$. Let φ be any embedding of T_h in any CCC graph. We will prove that $\mathrm{ADIL}(\varphi)\geq(\log k)/8$.

If $h=1$, then T_h is a k-vertex star. Let H be any CCC graph and H' be an arbitrary $k/2$-vertex induced subgraph of H. Choose a vertex v in H'. As the degree of H is 3, at most $1+3+3.2+\ldots+3.2^{\lfloor(\log k)/2\rfloor} < k/2$ vertices of H' lie within distance $\leq(\log k)/2$ from v. According to Lemma 4.3, this implies $\mathrm{ADIL}(\varphi)\geq(\log k)/8$.

Let $h\geq2$ and let $T_{h-1,i}$, i=1,...,k-1, denote the trees that arise from T_h by deleting the root t of T_h. Denote by $T_{h-1,0}$ the star graph induced by t and the vertices adjacent to t. Then $T_{h-1,0}$ is isomorphic to T_1. According to the inductive assumption

$$|E(T_h)|.\mathrm{ADIL}(\varphi) = \sum_{(u,v)\in E(T_h)} d_2(\varphi(u),\varphi(v))$$

$$= \sum_{(u,v)\in E(T_{h-1,0})} d_2(\varphi(u),\varphi(v)) + \sum_{i=1}^{k-1} \sum_{(u,v)\in E(T_{h-1,i})} d_2(\varphi(u),\varphi(v))$$

$$\geq (k-1)(\log k)/8+(k-1)|E(T_{h-1})|(\log k)/8=|E(T_h)|(\log k)/8. \ \square$$

Corollary 4.2. The worst case average dilation of embedding of a planar or an outerplanar graph into a binary tree, a butterfly graph, a CCC graph, a shuffle-exchange graph, or a de Brujn graph is $\Omega(\log k)$.

Unlike the graphs from the previous statements, the n-vertex hypercube Q_h ($n=2^h$) has degree log n. Thus the proof of Theorem 4.7 does not directly apply to embeddings in a hypercube. Moreover, one can easily notice that for any graph G there exists an embedding of G into Q_h with worst dilation ≤ 2, if h is sufficiently large ($h \geq |V(G)|-1$). Therefore, it is not possible to obtain a tight lower bound on the dilation of an embedding in a hypercube if the expansion of this embedding is not considered. We will examine embeddings into a hypercube with $O(1)$ expansion and will obtain a lower bound that matches the corresponding upper bound of Theorem 4.1.

Theorem 4.8. Any embedding with expansion $O(1)$ of a k-vertex star into a hypercube requires average dilation $\Omega(\log k)$.

Proof. Let $k \geq 3$, $h \geq \lceil \log k \rceil$ and Q_h be the hypercube of 2^h vertices. Let φ be any embedding with expansion O(1) of the k-vertex star S_k into Q_h. Denote by v the image in Q_h of the root of S_k and by $N(i)$ the number of the vertices of S_k that are mapped by φ onto vertices at distance i from v. Then

$$|V(S_K)|\,\mathrm{ADIL}(\varphi) = \sum_{i=0}^{h} iN(i) \geq \min \{ \sum_{i=0}^{h} in_i : \sum_{i=0}^{h} n_i = |V(S_K)|,\ 0 \leq n_i \leq C_{h,i} \}$$

$$\geq 2 \sum_{i=0}^{h'} i\, C_{h,i}\,,$$

where $C_{h,i}$ is the binomial coefficient $C_{h,i} = \frac{h!}{i!(h-i)!}$ and h' is the maximum integer such that $2\sum_{i=0}^{h'} C_{h,i} \le k$.

Since $|V(S_K)|=\Omega(|V(Q_n)|$, then $k=\Omega(2^h)$ and thus $h'=\Omega(h)$. Denote $c=h'/h=\Omega(1)$. Then

$$|V(S_K)|\,\mathrm{ADIL}(\varphi) \ge 2\sum_{i=\lceil ch/2\rceil}^{ch} i\, C_{h,i} \ge 2\lceil ch/2\rceil \sum_{i=\lceil ch/2}^{ch} C_{h,i}$$

$$\ge \lceil ch/2\rceil \sum_{i=0}^{ch} C_{h,i} \ge (ch/2)\sum_{i=0}^{h'} C_{h,i}.$$

Denote $A = 2\sum_{i=0}^{h'} C_{h,i}$. We will prove that $A=\Omega(k)$. If $h'=\lfloor h/2\rfloor$, then $A=k$. Suppose $h'<\lfloor h/2\rfloor$. Then

$$C_{h,h'+1} = C_{h,h'}\,(h-h')/(h'+1) \le C_{h,h'}\,(1-c)/c,$$

whence

$$k \le 2\sum_{i=0}^{h'+1} C_{h,i} = A + 2C_{h,h'+1} \le A + C_{h,h'}\,(1-c)/c \le A/c.$$

Therefore

$$|V(S_K)|\,\mathrm{ADIL}(\varphi) = \Omega(h)\Omega(k) = \Omega(k\log k). \ \square$$

Corollary 4.3. The worst case average dilation of any $O(1)$-expansion embedding of a planar or outerplanar graph of degree k into a hypercube of $O(k)$ vertices is $\Omega(\log k)$.

Theorem 4.9. For any $k>0$ and any $n>24$ there exists an n-vertex planar graph G of degree $\le k$ such that any embedding φ of G into a 1-dimensional mesh requires average dilation $\ge \sqrt{kn}/108$.

Proof: Combining Theorem 2.3 and Lemma 4.2 we get

$$|E(G)|\mathrm{ADIL}(\varphi) > \sum_{p=\lceil n/3\rceil}^{\lfloor n/2\rfloor} \sqrt{kn}/9 > (n/2-n/3-2)\sqrt{kn}/9 \geq (n/12)\sqrt{kn}/9.$$

As $|E(G)|\leq 2n$, then $\mathrm{ADIL}(\varphi)>(1/216)\sqrt{kn}$. □

Theorem 4.10. Any embedding of the complete (k-1)-ary tree into the d-dimensional mesh requires average dilation $\Omega(dk^{1/d})$, for $k>(2e)^d$.

Proof: We prove that any embedding of the k-vertex star into the d-dimensional mesh requires average dilation $\Omega(dk^{1/d})$. The rest of the proof is the same as in Theorem 4.7. Let v be a vertex of the mesh. Consider the set M of all the vertices that are at distance at most 2 from v. We can assume that the mesh is embedded into the d-dimensional space with unit distance between adjacent vertices. Define a unit d-dimensional cube around each vertex u of M such that u is placed in the center of the corresponding cube and cubes are oriented along the axes. The set of all cubes forms a polyhedron P_1. This polyhedron entirely lies in the minimal convex polyhedron P_2 containing all points whose i-th coordinate is equal to $l+d/2$ and other coordinates are zeroes, $i=1,\ldots,d$. Because the volume of P_2 is $2^d(l+d/2)^d/d\,!$, at most $(2l+d)^d/d\,!$ vertices lie within distance $\leq l$ from v. Choose $l=dk^{1/d}/(4e)$. Then at most $(dk^{1/d}/(4e)+d/2)^d/d\,!<k/2$ vertices lie within distance $\leq l$ from v, for $k\geq 2(4e)^d$. Therefore the radius of any k-vertex induced subgraph of the d-dimensional mesh is at least $dk^{1/d}/(4e)$. □

The upper and lower bounds of the average dilation of embedding planar and outerplanar graphs with $O(1)$ expansion obtained in this section are summarized in the following table. The lower bound marked by * is due to [15,30].

5. Conclusion

We investigated edge separation problems for planar graphs, outerplanar graphs, and trees of given degree. The results were used for obtaining estimations on the bisection width and the cutwidth of these graphs as well as for obtaining efficient average cost embeddings. Most of the bounds found are optimal within a constant factor. The results of this paper evoke some open problems:

ADIL	bin.tree, hypercube, CCC, SE, butterfly, de Brujn graph	1-dimensional mesh	2-dimensional mesh	d-dimens. mesh, d>2
planar graphs	$O(\log k)$	$O(\sqrt{kn})$	$O(\sqrt{k \log(2n/k)})$	$O(dk^{1/d})$
	$\Omega(\log k)$	$\Omega(\sqrt{kn})$	$\Omega(\sqrt{k})$	$\Omega(dk^{1/d})$
outerplanar graphs	$O(\log k)$	$O(k \log(2n/k))$	$O(\sqrt{k})$	$O(dk^{1/d})$
	$\Omega(\log k)$	$\Omega(k \log n/\log k)$ *	$\Omega(\sqrt{k})$	$\Omega(dk^{1/d})$

n - number of vertices
k - degree

Table 1. Summary of embedding results

5.1. Improving the constants in the upper and the lower bounds for the size of the separators.

5.2. We conjecture that any n-vertex planar graph G has an edge bisector of size $\leq n$ (independent on the degree of G). We know a graph requiring an n-edge separator.

5.3. Reducing the gaps between the upper and lower bounds for embedding outerplanar graphs into 1-dimensional meshes and planar graphs into 2-dimensional meshes. We conjecture that the average dilation in the latter case does not depend on n.

5.4. Obtaining upper and lower bounds on the *worst* dilation of embeddings of planar and outerplanar graphs.

References

[1] B. Baker, Approximation algorithms for NP-complete problems on planar graphs, *Proc. 24-th Ann. Symp. Found. Comp. Sci.*, IEEE,1983, 265-273.

[2] S.N. Bhatt, F.R.K. Chung, J.W. Hong, F.T. Leighton, A.L. Rosenberg, Optimal simulation by butterfly networks, *Proc. 20-th Ann. Symp. on Theory of Computing*, ACM, 192-204.

[3] S.N. Bhatt, F.R.K. Chung, F.T. Leighton, A.L. Rosenberg, Optimal simulation of tree machines, *Proc. 27-th Ann. Symp. Found. Comp. Sci.*, IEEE, 1986, 274-282.

[4] F. Berman, L. Snyder, On mapping parallel algorithms into parallel architectures, *J. of Parallel and Distributed Computing, vol. 4,* 1987, 439-458.

[5] F.R.K. Chung, A.L. Rosenberg, Minced trees, *Mathematical Systems Theory, vol. 19,* 1986, 1-12.

[6] H.N. Djidjev, On the problem of partitioning planar graphs, *SIAM J. Alg. Discr. Meth., vol. 3, no. 2,* 1982, 229-240.

[7] H.N. Djidjev, A linear algorithms for partitioning graphs of fixed genus, *SERDICA, vol. 11,* 1985, 369-387.

[8] H.N. Djidjev, A separator theorem for graphs of fixed genus, *SERDICA, vol. 11,* 1985, 319-329.

[9] R.A. De Millo, S.C. Eisenstat, R.J. Lipton, Preserving average proximity in arrays, *Comm. ACM, vol. 21,* 1978, 228-230.

[10] J.R. Gilbert, J.P. Hutchinson, R.E. Tarjan, A separator theorem for graphs of bounded genus, *Journal of Algorithms*, no. 5, 1984, 391-398.

[11] J.R. Gilbert, *Graph separator theorems and sparse Gaussian elimination*, Ph. D. Thesis, Stanford University, 1980.

[12] M.R. Garey, D.S. Johnson, L. Stockmeyer, Some simplified NP-complete graph problems, *Theor. Comp. Sci., vol. 1,* 1976, 237-267.

[13] H. Gazit, G.L. Miller, A parallel algorithm for finding a separator for planar graphs, *Proc. 28-th Ann. Symp. Found. Comp. Sci.,* IEEE, 1987, 238-248.

[14] L.S. Heath, Embedding outerplanar graphs in small books, *SIAM J. Alg. Discr. Meth., vol. 8,* no. 2, 1987, 198-218.

[15] M.A. Jordanskij, Minimal numbering of tree vertices, *Probl. Kib., vol. 31,* 1976, 109-133 (in Russian).

[16] D.B. Johnson, S.M. Venkatesan, Partition of planar flow networks, *Proc. 24-th Ann. Symp. Found. Comp. Sci.,* IEEE, 1983, 259-263.

[17] R.J. Lipton, D.J. Rose, R.E. Tarjan, Generalized nested dissection, *SIAM J. Numer. Anal., vol.16*, 1979, 346-358.

[18] F.T. Leighton, A layout strategy which is provably good, *Proc. 23-rd Ann. Symp. Found. Comp. Sci.,* IEEE, 1982, 85-98.

[19] C.E. Leiserson, Area efficient graph layouts (for VLSI), *Proc. 21-st Ann. Symp. Found. Comp. Sci.,* IEEE, 1980, 270-281.

[20] R.J. Lipton, R.E. Tarjan, A separator theorem for planar graphs, *SIAM J. Appl. Math., vol. 36, no. 2,* 1979, 177-189.

[21] R.J. Lipton, R.E. Tarjan, Applications of a planar separator theorem, *SIAM J. Computing, vol. 9, no. 3,* 1980, 615-627.

[22] G.L. Miller, Finding small simple cycle separators for 2-connected planar graphs, *J. of Comp. System Sci., vol. 32,* 1986, 265-279.

[23] B. Monien, The complexity of embedding graphs into binary trees, Proc. FCT'85, *Lecture Notes Comp. Sci., vol. 199,* Springer-Verlag, Berlin, 1985, 170-180.

[24] B.Monien, I.H. Sudborough, Comparing interconnection networks, Proc. MFCS'88, *Lecture Notes Comp. Sci., vol. 320,* Springer-Verlag, Berlin, 1988, 138-153.

[25] M.S. Paterson, W.L. Ruzzo, L. Snyder, Bound on minimax edge length for complete binary trees, *Proc. 13-th Ann. Symp.on Theory of Computing,* ACM, 293-299.

[26] S.S. Ravi, H.B. Hunt, An application of the planar separator theorem to counting problems, *Inf. Proc. Let., vol. 25, no. 6,* 1987, 317-322.

[27] D. Richards, Finding short cycles in planar graphs using separators, *J. of Algorithms, vol. 7,* 1986, 382-394.

[28] A.L. Rosenberg, A hypergraph model for fault tolerant VLSI processor arrays, *IEEE Trans. on Computers, vol. C-34, no. 6,* 1985, 578-584.

[29] A.L. Rosenberg, Graph embeddings 1988: recent breakthroughs, new directions, Proc. AWOC'88, *Lecture Notes Comp. Sci., vol. 319,* Springer-Verlag, Berlin, 1985, 160-169.

[30] M.A. Scheidwasser, On the length and the width of embeddings of graphs in meshes, *Probl. Kib., vol. 29,* 1974, 63-102 (in Russian).

[31] L.G. Valiant, Universality consideration in VLSI circuits, *IEEE Trans. on Computers, vol. C-30, no. 2,* 1981, 135-140.

[32] S.M. Venkatesan, Improved constants for some separator theorems, *J. of Algorithms, vol. 8*, 1987, 572-578.

[33] S.M. Venkatesan, *On separating a planar graph into two halves,* manuscript.

[34] P.M.B. Vitanyi, Locality, communication, and interconnect length in multicomputers, *SIAM J. Computing,* 1988, to appear.

[35] M. Yannakakis, Linear and book embeddings of graphs, Proc. AWOC'86, *Lecture Notes Comp. Sci., vol. 227,* Springer-Verlag, Berlin, 1986.

School of Computer Science, Carleton University
Recent Technical Reports

SCS-TR-147 **On Transparently Modifying Users' Query Distributions**
B.J. Oommen and D.T.H. Ng, November 1988

SCS-TR-148 **An O(N Log N) Algorithm for Computing a Link Center in a Simple Polygon**
H.N. Djidjev, A. Lingas and J.-R. Sack, July 1988, revised full version Sept. 1990

SCS-TR-149 **Smallscript: A User Programmable Framework Based on Smalltalk and Postscript**
Kevin Haaland and Dave Thomas, November 1988

SCS-TR-150 **A General Design Methodology for Dictionary Machines**
Frank Dehne and Nicola Santoro, February 1989

SCS-TR-151 **On Doubly Linked List ReOrganizing Heuristics**
D.T.H. Ng and B. John Oommen, February 1989

SCS-TR-152 **Implementing Data Structures on a Hypercube Multiprocessor, and Applications in Parallel Computational Geometry**
Frank Dehne and Andrew Rau-Chaplin, March 1989

SCS-TR-153 **The Use of Chi-Squared Statistics in Determining Dependence Trees**
R.S. Valiveti and B.J. Oommen, March 1989

SCS-TR-154 **Ideal List Organization for Stationary Environments**
B. John Oommen and David T.H. Ng, March 1989

SCS-TR-155 **Hot-Spot Contention in Binary Hypercube Networks**
Sivarama P. Dandamudi and Derek L. Eager, April 89

SCS-TR-156 **Some Issues in Hierarchical Interconnection Network Design**
Sivarama P. Dandamudi and Derek L. Eager, April 1989

SCS-TR-157 **Discretized Pursuit Linear Reward-Inaction Automata**
B.J. Oommen and Joseph K. Lanctot, April 1989

SCS-TR-158 (revised) **Parallel Fractional Cascading on a Hypercube Multiprocessor**
Frank Dehne, Afonso Ferreira and Andrew Rau-Chaplin, May 1989 (Revised April 1990)

SCS-TR-159 **Epsilon-Optimal Stubborn Learning Mechanisms**
J.P.R. Christensen and B.J. Oommen, June 1989

SCS-TR-160 **Disassembling Two-Dimensional Composite Parts Via Translations**
Doron Nussbaum and Jörg-R. Sack, June 1989

SCS-TR-161 (revised) **Recognizing Sources of Random Strings**
R.S. Valiveti and B.J. Oommen, January 1990
Revised version of SCS-TR-161 "On the Data Analysis of Random Permutations and its Application to Source Recognition", published June 1989

SCS-TR-162 **An Adaptive Learning Solution to the Keyboard Optimization Problem**
B.J. Oommen, R.S. Valiveti and J. Zgierski, October 1989

SCS-TR-163 **Finding a Central Link Segment of a Simple Polygon in O(N Log N) Time**
L.G. Alexandrov, H.N. Djidjev, J.-R. Sack, October 1989

SCS-TR-164 **A Survey of Algorithms for Handling Permutation Groups**
M.D. Atkinson, January 1990

SCS-TR-165 **Key Exchange Using Chebychev Polynomials**
M.D. Atkinson and Vincenzo Acciaro, January 1990

SCS-TR-166 **Efficient Concurrency Control Protocols for B-tree Indexes**
Ekow J. Otoo, January 1990

SCS-TR-167 **A Hierarchical Stochastic Automaton Solution to the Object Partitioning Problem**
B.J. Oommen, January 1990

SCS-TR-168 **Adaptive List Organizing for Non-stationary Query Distributions. Part I: The Move-to-Front Rule**
R.S. Valiveti and B.J. Oommen, January 1990

SCS-TR-169 **Trade-Offs in Non-Reversing Diameter**
Hans L. Bodlaender, Gerard Tel and Nicola Santoro, February 1990

SCS-TR-170 **A Massively Parallel Knowledge-Base Server using a Hypercube Multiprocessor**
Frank Dehne, Afonso Ferreira and Andrew Rau-Chaplin, April 1990

SCS-TR-171 **Parallel Processing of Quad Trees on the Hypercube (and PRAM)**
Frank Dehne, Afonso Ferreira and Andrew Rau-Chaplin, April 1990

SCS-TR-172 **A Note on the Load Balancing Problem for Coarse Grained Hypercube Dictionary Machines**
Frank Dehne and Michel Gastaldo, May 1990

SCS-TR-173 **Self-Organizing Doubly-Linked Lists**
R.S. Valiveti and B.J. Oommen, May 1990

SCS-TR-174 **A Presortedness Metric for Ensembles of Data Sequences**
R.S. Valiveti and B.J. Oommen, May 1990

SCS-TR-175 **Separation of Graphs of Bounded Genus**
Ljudmil G. Aleksandrov and Hristo N. Djidjev, May 1990

SCS-TR-176 **Edge Separators of Planar and Outerplanar Graphs with Applications**
Krzystof Diks, Hristo N. Djidjev, Ondrej Sykora and Imrich Vrto, May 1990

SCS-TR-177 **Representing Partial Orders by Polygons and Circles in the Plane**
Jeffrey B. Sidney and Stuart J. Sidney, July 1990

SCS-TR-178 **Determining Stochastic Dependence for Normally Distributed Vectors Using the Chi-squared Metric**
R.S. Valiveti and B.J. Oommen, July 1990

SCS-TR-179 **Parallel Algorithms for Determining K-width-Connectivity in Binary Images**
Frank Dehne and Susanne E. Hambrusch, September 1990

SCS-TR-180 **A Workbench for Computational Geometry (WOCG)**
P. Epstein, A. Knight, J. May, T. Nguyen, and J.-R. Sack, September 1990

SCS-TR-181 **Adaptive Linear List Reorganization under a Generalized Query System**
R.S. Valiveti, B.J. Oommen and J.R. Zgierski, October 1990

SCS-TR-182 **Breaking Substitution Cyphers using Stochastic Automata**
B.J. Oommen and J.R. Zgierski, October 1990

SCS-TR-183 **A New Algorithm for Testing the Regularity of a Permutation Group**
V. Acciaro and M.D. Atkinson, November 1990

SCS-TR-184 **Generating Binary Trees at Random**
M.D. Atkinson and J.-R. Sack, December 1990

SCS-TR-185 **Uniform Generation of Combinatorial Objects in Parallel**
M.D. Atkinson and J.-R. Sack, January 1991

SCS-TR-186 **Reduced Constants for Simple Cycle Graph Separation**
Hristo N. Djidjev and Shankar M. Venkatesan, February 1991

SCS-TR-187 **Multisearch Techniques for Implementing Data Structures on a Mesh-Connected Computer**
Mikhail J. Atallah, Frank Dehne, Russ Miller, Andrew Rau-Chaplin, and Jyh-Jong Tsay, February 1991