# On the Probable Performance of Heuristics for Bandwidth Minimization

Jonathan S. Turner

WUCS-84-2
February 1984

*ABSTRACT*

Most research in algrorithm design relies on worst-case analysis for performance comparisons. Unfortunately, worst-case analysis does not always provide an adequate measure of an algorithm's effectiveness. This is particularly true in the case of heuristic algorithms for hard combinatorial problems. In such cases, analysis of the probable performance can yield more meaningful results and can provide insight leading to better algorithms. The problem of minimizing the bandwidth of a sparse symmetric matrix by perfoming simultaneous row and column permutations, is an example of a problem for which there are well-known heuristics whose practical success has lacked a convincing analytical explanation. A class of heuristics introduced by Cuthill and McKee in 1969, and referred to here as the ***level algorithms***, are the basis for bandwidth minimization routines that have been widely used for over a decade. At the same time, it is easy to construct examples, showing that the level algorithms can produce solutions that differ from optimal by an arbitrarily large factor. This paper provides an analytical explanation for the practical success of the level algorithms, by showing that for random matrices having optimal bandwidth no larger than $k$, any level algorithm will produce solutions that differ from optimal by a small constant factor. The analysis also suggests another class of algorithms with better performance. One algorithm in this class is shown to produce solutions that are nearly optimal.

---

Submitted to SIAM Journal of Computing 2/84

On the Probable Perfomance of Heuristics
for Bandwidth Minimization

Jonathan S. Turner
Washington University

## 1. Introduction

Let $M$ be a symmetric matrix and let $k$ be the largest integer for which there is a non-zero entry $M[i,i+k]$; $k$ is called the bandwidth of $M$. It is often possible to reduce the bandwidth of a matrix by performing simultaneous row and column permutations. Most common matrix operations can be performed more efficiently if the matrices are in small bandwidth form. The matrices can also be stored more efficiently in this form. The matrix bandwidth minimization problem is usually re-cast as a graph theory problem; for any matrix $M$, the graph corresponding to $M$ has an edge joining vertices $i$ and $j$ if and only if $M[i,j]$ is non-zero.

Let $G=(V,E)$ be a graph with $V=\{1,2, \ldots ,n\}$. A *layout* of $G$ is a permutation on $\{1,2, \ldots ,n\}$. Define the *bandwidth* of $G$ with respect to a layout $\tau$ by $\varphi_\tau(G) = \max_{\{u,v\} \in E} |\tau(u)-\tau(v)|$. The *bandwidth* of $G$ is defined by $\varphi(G) = \min_\tau \varphi_\tau(G)$. The *bandwidth minimization problem* (for graphs) is to determine for a graph $G$ and an integer $k$ if $\varphi(G) \le k$. Papadimitriou [9] first showed that the bandwidth minimization problem is NP-complete. Garey, Graham, Johnson and Knuth [7] later strengthened this result, showing that the problem remains NP-complete when restricted to free binary trees. Several heuristic algorithms for bandwidth minimization were proposed in the late sixties and early seventies. More recently, Saxe [10] has found a dynamic programming algorithm which can determine if a graph has bandwidth $k$ in time $O(n^{k+1})$ for any fixed value of $k$.

Monien and Sudborough [8] showed how to reduce the time bound to $O(n^k)$. One of the most successful heuristic algorithms is one discovered by Cuthill and McKee [5] which is a member of a class of algorithms which are referred to here as the ***level algorithms***. An algorithm is classified as a level algorithm if for all graphs $G=(V,E)$ the layout $\tau$ produced by the algorithm satisfies

$$\forall u,v \in V \quad d(\tau^{-1}(1),u) < d(\tau^{-1}(1),v) \Rightarrow \tau(u) < \tau(v)$$

where $d(x,y)$ denotes the length of the shortest path connecting vertices $x$ and $y$. The level algorithms are reasonably fast and have proved to be quite successful in practice. On the other hand, one can easily construct examples in which the ratio of the bandwidth of the layout produced by a level algorithm to the actual bandwidth of the graph is arbitrarily large. Consequently one must resort to probabilistic analysis to gain insight to their practical success.

Let $G=(V,E)$ be generated by the following random experiment. Let $V=\{1,2, \ldots ,n\}$, $E=\phi$. For each $\{u,v\}$ $1 \le u < v \le n$, add the edge $\{u,v\}$ to $E$ with probability $p$.

The probability distribution defined by this experiment is denoted $\Gamma_n(p)$ and the notation $G \in \Gamma_n(p)$ means that $G$ is a random graph generated by this experiment. In section 2 it is shown that for almost all $G \in \Gamma_n(p)$, $\frac{n}{\varphi(G)} \le 1+\varepsilon$ when $p \ge c \ln n / n$ and $\varepsilon > 0$, $c > 0$ are fixed. (We say that a property holds for ***almost all*** graphs if the probability of the property holding approaches one as the number of vertices gets large.) Consequently, if $\tau$ is any layout at all, $\frac{\varphi_\tau(G)}{\varphi(G)} \le (1+\varepsilon)$ for almost all random graphs $G \in \Gamma_n(p)$. This makes it pointless to compare the probable performance of bandwidth minimization algorithms on random graphs in $\Gamma_n(p)$. Therefore another class of probability distributions is introduced and used for most of the results given here. Let $G=(V,E)$ be generated by the following random experiment. Let $V=\{1,2,\ldots,n\}$. $E=\phi$. For each $\{u,v\}$ $1\le u<v\le n$ such that $|u-v|\le\psi$ include the edge $\{u,v\}$ in $E$ with probability $p$.

The probability distribution defined by this experiment is denoted $\Psi_n(\psi,p)$. Now, let $G\in\Psi_n(\psi,p)$ and randomly re-number the vertices of $G$. The resulting distribution is denoted $\Omega_n(\psi,p)$. Note that if $G\in\Omega_n(\psi,p)$ then $\varphi(G)\le\psi$. Also, if $H$ is a graph with $\varphi(H)\le\psi$, then $H$ can be generated by $\Omega_n(\psi,p)$. The use of $\Omega_n(\psi,p)$ allows us to explore properties that are common to most graphs having bandwidth $\le\psi$, but rare for unrestricted graphs. Heuristics like the level algorithms exploit such properties to produce good layouts for most graphs.

It is shown in section 3 that if $A$ is any level algorithm and $A(G)$ is the bandwidth of the layout produced by $A$ on the graph $G$ then $A(G) < (1+\varepsilon)(3-p)\varphi(G)$ for almost all $G\in\Omega_n(\psi,p)$, where $\varepsilon>0$, $0<p<1$ are fixed, and $\ln n = o(\psi)$. If in addition $\psi<n/2$, then $(1-\varepsilon)(2-p)\varphi(G) < A(G)$. The analysis leads to a new class of algorithms called the ***modified level algorithms***, for which $A(G) < 2(1+\varepsilon)\varphi(G)$ where $A$ is any modified level algorithm and $G\in\Omega_n(\psi,p)$. In section 4, a specific modified level algorithm, $MLA1$ is studied and it is shown that $MLA1(G) < (1+\varepsilon)\varphi(G)$ for almost all $G\in\Omega_n(\psi,p)$ when $\psi<n/4$. Section 5 presents several other modified level algorithms, discusses running times and summarizes empirical studies comparing their performance. Section 6 shows how to improve the running times of the above algorithms through more careful selection of the 'starting vertex'. Finally, section 7 contains several results concerning properties of random graphs. Conditions are given for connectivity of random graphs in $\Psi_n(\psi,p)$ and probable upper bounds are given for the diameter of random graphs in $\Gamma_n(p)$ and $\Psi_n(\psi,p)$.

**2. Bandwidth of Graphs in $\Gamma_n(p)$ and $\Psi_n(\psi,p)$**

Define $\lambda_n(c) = -\frac{\ln n}{\ln c}$. Note that $\lambda_n(c) > 0$ when $0 < c < 1$ and $n > 1$, $c^{\lambda_n(c)} = \frac{1}{n}$ and $\lim_{n \to \infty} \lambda_n(c) = \infty$ for $c$ fixed $0 < c < 1$. We will usually write $\lambda(c)$ for $\lambda_n(c)$. The following results demonstrate that almost all random graphs in the usual model, have bandwidth nearly as large as the number of vertices.

**Theorem 2.1.** Let $0 < p < 1$ be fixed. For almost all $G \in \Gamma_n(p)$, $\varphi(G) > n - 4\lambda(1-p)$.

**Theorem 2.2.** Let $\varepsilon > 0$, $c > 0$ be fixed, $p = c\,\frac{\ln n}{n}$. For almost all $G \in \Gamma_n(p)$, $\varphi(G) \geq n(1-\varepsilon)$.

For $G = (V,E)$, the notation $u - v$ means $\{u,v\} \in E$ and $u \not- v$ means that $\{u,v\} \notin E$. Similarly if $U \subseteq V$ and $W \subseteq V$ then $U - W$ means that some vertex in $U$ is adjacent to some vertex in $W$. The proofs of Theorems 2.1 and 2.2 require the following lemmas.

**Lemma 2.1.** Let $G=(V,E)$ be a graph on $n$ vertices. $\varphi(G) \leq n - 2k \Rightarrow \exists V_1, V_2 \subseteq V$ such that $|V_1| = |V_2| = k$ and $V_1 \not- V_2$.

*proof.* If $\varphi(G) \leq n-2k$ then there is a layout $\tau$ such that $u - v \Rightarrow |\tau(u) - \tau(v)| \leq n-2k$. Let $V_1 = \{\tau^{-1}(1), \ldots, \tau^{-1}(k)\}$ and $V_2 = \{\tau^{-1}(n-k+1), \ldots, \tau^{-1}(n)\}$. If $V_1 - V_2$ then there are vertices $u \in V_1$ and $v \in V_2$ such that $u - v$. But by the definition of $V_1$ and $V_2$, $\tau(u) \leq k$ and $\tau(v) \geq n-k+1$, hence $|\tau(u) - \tau(v)| > n-2k$, which contradicts the definition of $\tau$.

□

**Lemma 2.2.** Let $0 < p < 1$ and $G = (V,E) \in \Gamma_n(p)$. $\mathrm{P}(\varphi(G) \leq n-2k) \leq \left[\frac{en}{k}(1-p)^{k/2}\right]^{2k}$.

*proof*. By Lemma 2.1, $P(\varphi(G) \le n-2k) \le P(\exists V_1, V_2$ such that $|V_1| = |V_2| = k \wedge V_1 \not\sim V_2)$. Since there are $k^2$ 'potential edges' joining $V_1$ and $V_2$, all of which must be absent if $V_1 \not\sim V_2$, this last probability is

$$\le \binom{n}{k}\binom{n-k}{k}(1-p)^{k^2} \le \left[\frac{en}{k}\right]^{2k}(1-p)^{k^2} = \left[\frac{en}{k}(1-p)^{k/2}\right]^{2k} \qquad \square$$

*Proof of Theorem 2.1.* Applying Lemma 2.2 with $k = 2\lambda(1-p)$ gives

$$P(\varphi(G) \le n-4\lambda(1-p)) \le \left[\frac{en}{2\lambda(1-p)}(1-p)^{\lambda(1-p)}\right]^{4\lambda(1-p)} = \left[\frac{e}{2\lambda(1-p)}\right]^{4\lambda(1-p)} \to 0 \quad \square$$

*Proof of Theorem 2.2.* Applying Lemma 2.2 with $k = \varepsilon n/2$ gives

$$P(\varphi(G) \le n(1-\varepsilon)) \le \left[\frac{en}{\varepsilon n/2}(1-p)^{\varepsilon n/4}\right]^{\varepsilon n} \le \left[\frac{2e}{\varepsilon}e^{-\varepsilon np/4}\right]^{\varepsilon n} = \left[\frac{2e}{\varepsilon}n^{-\varepsilon c/4}\right]^{\varepsilon n} \to 0 \square$$

Theorems 2.1 and 2.2 show that even for sparse random graphs $G \in \Gamma_n(p)$, $\frac{n}{\varphi(G)} \to 1$. Consequently, even the most trivial algorithms (for example, the algorithm that always outputs the identity layout) can produce layouts having bandwidth close to $\varphi(G)$ as $n$ gets large. If one is to make meaningful distinctions among algorithms based on their probable performance some other probability distribution is required. The distributions $\Omega_n(\psi,p)$ and $\Psi_n(\psi,p)$ are used here. Obviously, any structural property of a graph occurs with the same probability in both distributions. It is clear that if $G \in \Psi_n(\psi,p)$ then $\varphi(G) \le \psi$. The following theorem gives a probabilistic lower bound on $\varphi(G)$.

**Theorem 2.3.** Let $0<p<1$ be fixed, $\ln n \le \psi < n$. For almost all $G \in \Psi_n(\psi,p)$, $\varphi(G) > \psi - 4\lambda_\psi(1-p)$.

*proof*. Let $G' \subseteq G$ be the subgraph induced by vertices $\{1, 2, \ldots, \psi\}$. Note that $G'$ is a random graph with distribution $\Gamma_\psi(p)$. Applying Theorem 2.1, $\varphi(G') > \psi - 4\lambda_\psi(1-p)$ The theorem follows from the fact that $\varphi(G) \ge \varphi(G')$. $\square$

An immediate consequence of this result is that as $\psi$ gets large, it comes within a factor of $1+\varepsilon$ of $\varphi(G)$, for any fixed $\varepsilon>0$. While Theorem 2.3 is sufficient for the results proved here, it is interesting to consider a tighter relationship between $\psi$ and $\varphi(G)$.

**Conjecture.** Let $0<p<1$ be fixed. There is some constant $c=c(p)>0$ such that if $c \ln n \le \psi \le n - c \ln n$ then for almost all $G \in \Psi_n(\psi,p)$, $\varphi(G)=\psi$.

## 3. Probabilistic Algorithms for Bandwidth Minimization

Before proceeding we need the following definitions. Let $G=(V,E)$ and define $V_i(u)=\{v \mid d(u,v)=i\}$ for all $u \in V$. Also let $V_i=V_i(1)$. Next, define $l_i(u)=\min V_i(u)$ and $r_i(u)=\max V_i(u)$. Let $l_i=l_i(1)$ and $r_i=r_i(1)$. Note that $|V_i| \le r_i - l_i$. Define

$$\mathit{level}(G)=\min_{u \in V} \max_{i \ge 0} |V_i(u)|$$
$$\mathit{LEVEL}(G)=\min_{u \in V} \max_{i \ge 0} |V_i(u) \cup V_{i+1}(u)| - 1$$

Note that if $A$ is any level algorithm at all

$$\mathit{level}(G) \le A(G) \tag{1}$$

and if $A$ makes the best possible choice for $\tau^{-1}(1)$

$$A(G) \le \mathit{LEVEL}(G) \tag{2}$$

In the next few sections, we will consider only algorithms that do always make the best choice. We can satisfy this requirement by trying all possible choices for $\tau^{-1}(1)$, at a cost of a factor of $n$ in the running time. In section 6, we will relax this restriction.

Consider the tree $T$ in Figure 1. It is not difficult to see that $\varphi(T)=2$ and $\mathit{level}(T)=4$. The example is readily extended. For any integer $k>0$ one can construct a tree $T_k$ such that $\varphi(T_k)=2$ and $\mathit{level}(T_k)=k$. This implies that the worst case performance of the level algorithms can be arbitrarily poor. In spite of this, the level algorithms perform quite well on random graphs.

[ Figure 1 here ]

**Theorem 3.1.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\psi<n$, $\ln n = o(\psi)$. For almost all $G\in\Omega_n(\psi,p)$, $LEVEL(G) < (1+\varepsilon)(3-p)\varphi(G)$.

The theorem is proved by deriving probable upper bounds on $|V_i|$ and applying equation (2). These bounds are developed in the following lemmas.

**Lemma 3.1.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)\le\psi<n$. For almost all $G\in\Psi_n(\psi,p)$, there exists a path of length two between every pair of vertices $u,v$ such that $|u-v|\le 2\psi-\alpha$.

*proof.* Let $u,v\in V$ with $|u-v|\le 2\psi-\alpha$. Let $i=2\psi-|u-v|$. The probability that $d(u,v)>2$ is $\le(1-p^2)^i$. Since for each $i$ there are $\le n$ such pairs, the probability that any pair is not joined by a 2-path is

$$\le\sum_{i=\lceil\alpha\rceil}^{2\psi-1} n(1-p^2)^i < n(1-p^2)^{\alpha}\sum_{i=0}^{\infty}(1-p^2)^i = p^{-2}n^{-\varepsilon}\to 0 \qquad \square$$

**Lemma 3.2.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)\le\psi<n$. For almost all $G\in\Psi_n(\psi,p)$, $r_i-3\psi\le r_{i-3}<l_i-(2\psi-\alpha)$, for all $i\ge 3$.

*proof.* The shortest path from 1 to $r_i$ must pass through some $u\in V_{i-3}$. Clearly $r_i-u\le 3\psi$, hence $r_i-3\psi\le u\le r_{i-3}$. To see that $r_{i-3}<l_i-(2\psi-\alpha)$, assume otherwise. Then there is some vertex $v$ on the shortest path from 1 to $r_{i-3}$ such that $l_i-(2\psi-\alpha)\le v<l_i$ and $d(1,v)\le i-3$. By Lemma 3.1 there is a 2-path from $v$ to $l_i$, giving $d(1,l_i)\le i-1$, which is a contradiction. $\square$

**Lemma 3.3.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)\le\psi<n$. For almost all $G\in\Psi_n(\psi,p)$, $r_i-l_i<\psi+\alpha$, for all $i\ge 3$.

*proof.* By Lemma 3.2 $r_i\le r_{i-3}+3\psi$ and $l_i>r_{i-3}+(2\psi-\alpha)$ Hence,

$$r_i - l_i < (r_{i-3}+3\psi)-(r_{i-3}+(2\psi-\alpha))=\psi+\alpha \qquad \square$$

Note that Lemma 3.3 says nothing about the size of $V_1$ and $V_2$. As we shall see, these cases differ from the rest and will be handled in Lemma 3.5. First however, we need a lemma concerning the binomial distribution, $B(n,p)$. By definition if $x \in B(n,p)$ then $P(x=k) = \binom{n}{k} p^k (1-p)^{n-k}$. The following lemma is from Angluin and Valiant [1].

**Lemma 3.4.** If $x \in B(n,p)$ then for any $\varepsilon$, $0<\varepsilon<1$, $P(x \le (1-\varepsilon)np) < e^{-\varepsilon^2 np/2}$ and $P(x \ge (1+\varepsilon)np) < e^{-\varepsilon^2 np/3}$.

**Lemma 3.5.** Let $\varepsilon>0$, $0<p<1$ be fixed, $c = -(1+\varepsilon)/\ln(1-p^2)$, $\alpha = c\ln n \le \psi < n$. For almost all $G \in \Psi_n(\psi,p)$, $(1-\varepsilon)p\psi < |V_1| < (1+\varepsilon)p\psi$ and $|V_2| < (1+\varepsilon)(2-p)\psi$. Also, if $\psi < n/2$ then $(1-\varepsilon)(2-p)\psi - \alpha < |V_2|$.

*proof.* $|V_1|$ is a binomial random variable in $B(\psi,p)$. By Lemma 3.4,

$$P(|V_1| < (1-\varepsilon)p\psi) < e^{-\varepsilon^2 p\psi/2} \to 0$$
$$P(|V_1| > (1+\varepsilon)p\psi) < e^{-\varepsilon^2 p\psi/3} \to 0$$

This establishes the bounds on $|V_1|$. Since $|V_2| \le 2\psi - |V_1|$,

$$|V_2| < 2\psi - (1-\varepsilon)p\psi < (1+\varepsilon)(2-p)\psi$$

When $\psi < n/2$, Lemma 3.1 gives

$$|V_2| \ge (2\psi-\alpha)-|V_1| > (2-(1+\varepsilon)p)\psi-\alpha > (1-\varepsilon)(2-p)\psi-\alpha \qquad \square$$

*Proof of Theorem 3.1.* By Lemmas 3.3 and 3.5, $LEVEL(G) < (1+\varepsilon')(3-p)\psi$ for any fixed $\varepsilon'>0$. By Theorem 2.3, $\psi < (1+\varepsilon')\varphi(G)$. Selecting $\varepsilon'$ so that $(1+\varepsilon')^2 = (1+\varepsilon)$ yields the theorem. $\square$

The analysis that leads to Theorem 3.1 also yields $level(G) > (1-\varepsilon)(2-p)\varphi(G)$ when $\ln n = o(\psi)$, since in this case $|V_2| > (1-\varepsilon)(2-p)\varphi(G)$ with high probability. Consequently the level algorithms are not capable of near optimal performance. However a related class of algorithms, called the modified level algorithms is.

Define

$$V'_2(u)=\begin{cases} V_2(u) & \text{if } V_3(u)=\phi \\ V_2(u)\cap\{v\,|\,v-w\in V_3(u)\} & \text{if } V_3(u)\neq\phi \end{cases}$$
$$V'_1(u)=(V_1(u)\cap V_2(u)) - V'_2(u)$$
$$V'_i(u)=V_i(u) \quad i=0,\ i\geq 3$$

Also, let $V'_i=V'_i(1)$, $l'_i(u)=\min V'_i(u)$, $r'_i(u)=\max V'_i(u)$, $l'_i=l'_i(1)$, $r'_i=r'_i(1)$. Formally, $A$ is a modified level algorithm, if the layout $\tau$ produced for the graph $G=(V,E)$ satisfies

$$\forall u,v\in V \;\; u\in V'_i(\tau^{-1}(1))\wedge v\in V'_{i+1}(\tau^{-1}(1))\Rightarrow\tau(u)<\tau(v)$$

Let

$$level'(G)=\min_{u\in V}\max_{i\geq 0}|V'_i(u)|$$
$$LEVEL'(G)=\min_{u\in V}\max_{i\geq 0}|V'_i(u)\cup V'_{i+1}(u)|-1$$

If $A$ is any modified level algorithm then $level'(G)\leq A(G)$ and if $A$ makes the best possible choice for the starting vertex then $A(G)\leq LEVEL'(G)$. For the modified level algorithms, we can show that for almost all $G\in\Psi_n(\psi,p)$, $|V_i| < (1+\varepsilon)\psi$ for all $i\geq 0$ when $\ln n=o(\psi)$, $\psi\leq(1-\varepsilon)n/2$. From this we obtain the following result.

**Theorem 3.2.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\psi<n$, $\ln n = o(\psi)$. For almost all $G\in\Omega_n(\psi,p)$ $LEVEL'(G) < 2(1+\varepsilon)\varphi(G)$.

The proof of Theorem 3.2 requires the following lemmas.

**Lemma 3.6.** Let $\varepsilon>0$, $0<p<1$ be fixed, $(1+\varepsilon)\lambda(1-p^2)\leq\psi<n$. For almost all $G=(V,E)\in\Psi_n(\psi,p)$, $u\in V\wedge|\{u-\psi,\ldots,u+\psi\}\cap V_i|\geq(1+\varepsilon)\lambda(1-p)\Rightarrow u-V_i$.

*proof.* Note that by Lemma 3.1, for each vertex $u$ there are at most five sets $V_i$ such that $|\{u-\psi,\ldots,u+\psi\}\;\cap V_i|\geq 1$. Hence, the probability that for any $G\in\Psi_n(\psi,p)$, the assertion is not true is $\leq 5n(1-p)^{(1+\varepsilon)\lambda(1-p)}=5n^{-\varepsilon}\to 0$. □

**Lemma 3.7.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)\le\psi<n$. For almost all $G\in\Psi_n(\psi,p)$ there exists a path of length three between every pair of vertices $u,v$ such that $|u-v|\le 3\psi-\alpha$.

[ Figure 2 here ]

*proof*. Let $u,v\in V$ be such that $i=3\psi-|u-v|\ge\alpha$. Let $x_j=u+\psi-j$ for $0\le j\le i$, as illustrated in Figure 2. Clearly any 3-path connecting $u$ and $v$ must pass through one of $x_0,\ldots,x_i$. The probability that no 3-path joins $u$ and $v$ is

$$
\begin{aligned}
&= \mathrm{P}(\text{no 3-path} \wedge u\not- x_0 \wedge \cdots \wedge u\not- x_i)\\
&\quad + \mathrm{P}(\text{no 3-path} \wedge u\not- x_0 \wedge \cdots \wedge u\not- x_{i-1} \wedge u-x_i)\\
&\quad + \mathrm{P}(\text{no 3-path} \wedge u\not- x_0 \wedge \cdots \wedge u\not- x_{i-2} \wedge u-x_{i-1})\\
&\quad + \cdots + \mathrm{P}(\text{no 3-path} \wedge u-x_0)
\end{aligned}
$$

$$
\begin{aligned}
&= (1-p)^{(i+1)}\mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_i)\\
&\quad + (1-p)^{(i+1)}\mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_{i-1} \wedge u-x_i)\\
&\quad + (1-p)^{i}\mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_{i-2} \wedge u-x_{i-1})\\
&\quad + \cdots + (1-p)\mathrm{P}(\text{no 3-path} | u-x_0)
\end{aligned}
$$

$$
\begin{aligned}
&< (1-p)^{(i+1)}[\, \mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_i)\\
&\quad + \mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_{i-1} \wedge u-x_i)\\
&\quad + (1-p)^{-1}\mathrm{P}(\text{no 3-path} | u\not- x_0 \wedge \cdots \wedge u\not- x_{i-2} \wedge u-x_{i-1} \wedge u\not- x_i)\\
&\quad + \cdots + (1-p)^{-i}\mathrm{P}(\text{no 3-path} | u-x_0 \wedge u\not- x_1 \wedge \cdots \wedge u\not- x_i)]
\end{aligned}
$$

$$
= (1-p)^{i+1}\left[1 + (1-p^2) + \frac{(1-p^2)^2}{(1-p)} + \frac{(1-p^2)^3}{(1-p)^2} + \cdots + \frac{(1-p^2)^{i+1}}{(1-p)^i}\right]
$$

$$
= (1-p)^{i+1}\left[1 + (1-p^2)\sum_{j=0}^{i}(1+p)^j\right] = (1-p)^{i+1}\left[1 + (1-p^2)\,\frac{(1+p)^{i+1}-1}{p}\right]
$$

$$= \frac{1}{p}\left[(1-p^2)^{i+2} + (p^2+p-1)(1-p)^{i+1}\right]$$

Since for each value of $i$ there are at most $n$ vertex pairs $u,v$ such that $|u-v|=i$, the probability that any pair $u,v$ with $|u-v| \le 3\psi-\alpha$ is not connected by a 3-path is

$$\le \sum_{i=\alpha}^{3\psi-1} \frac{n}{p}\left[(1-p^2)^{i+2} + (p^2+p-1)(1-p)^{i+1}\right]$$
$$< \frac{n}{p}\left[\left((1-p^2)^{\alpha}\sum_{i=0}^{\infty}(1-p^2)^{i}\right) + \left((p^2+p-1)(1-p)^{\alpha}\sum_{i=0}^{\infty}(1-p)^{i}\right)\right]$$
$$< \frac{n}{p}\left[\frac{1}{p^2}n^{-(1+\varepsilon)} + \frac{p^2+p-1}{p}n^{-(1+\varepsilon)}\right] < \frac{2}{p^3}n^{-\varepsilon} \to 0$$

**Lemma 3.8.** Let $\varepsilon > 0$, $0<p<1$ be fixed, $\alpha = (1+\varepsilon)\lambda(1-p^2)$, $\beta = (1+\varepsilon)\lambda(1-p)$ and $\max(\alpha, 2\beta) \le \psi \le (n-\beta)/2$. For almost all $G \in \Psi_n(\psi,p)$ $|V'_i| \le \psi+\alpha$ for $i \ge 0$.

*proof.* The result follows from Lemma 3.3 for $i \ge 3$ and is immediate for $i=0$. Before proving the theorem for $1 \le i \le 2$ we first need to show that $|V_3 \cap \{\psi+2, \ldots, 2\psi+\beta\}| \ge \beta$. Let $A = \{\psi+2, \ldots, 2\psi+1\}$. By Lemmas 3.3 and 3.7 $A \subseteq V_2 \cup V_3$. Let $x = |A \cap V_3|$. Clearly if $x \ge \beta$ then we're done. Assume then that $x < \beta$ and let $B = \{2\psi+2, \ldots, 2\psi+(\beta-x)+1\}$ and let $y = |B|$. Note that since $\psi \ge 2\beta$ that $\forall u \in B$ $|\{u-\psi, \ldots, u+\psi\} \cap V_2| \ge \beta$. Thus by Lemma 3.6 $B \subseteq V_3$. Since $x+y=\beta$ we have that $|V_3 \cap \{\psi+2, \ldots, 2\psi+\beta\}| \ge \beta$.

Now, by Lemma 3.2, $l_3 \ge 2\psi-\alpha+1$. This implies that $l'_2 \ge \psi-\alpha+1$ and since $r'_2 \le 2\psi+1$, it follows that $|V'_2| \le \psi+\alpha$ as claimed.

Finally, note that if $u \in A$ and $u \ge \psi+\beta$ then by Lemma 3.4 $u - V_3$ and hence $u \notin V'_1$. Thus $|V'_1| \le \psi+\beta < \psi+\alpha$ as claimed. □

*Proof of Theorem 3.2.* If $\psi > (n-\lambda(1-p))/2$ then since $LEVEL'(G) < n$

$$LEVEL'(G) < 2\psi+\lambda(1-p) < (2+\varepsilon')\psi$$

for any fixed $\varepsilon' > 0$. If $\psi \le (n-\lambda(p))/2$ then we can apply Lemmas 3.3 and 3.5 giving

$$LEVEL'(G) < 2\psi + \lambda(1-p^2) < (2+\varepsilon')\psi$$

By Theorem 2.3, $\psi < (1+\varepsilon')\varphi(G)$. Selecting $\varepsilon'$ so that $(2+\varepsilon')(1+\varepsilon') = 2(1+\varepsilon)$ yields the theorem. □

The same analysis shows that $level'(G) < (1+\varepsilon)\varphi(G)$.

## 4. Obtaining Nearly Optimal Layouts

In this section a specific modified level algorithm denoted *MLA*1 is described and analyzed. It is shown that *MLA*1 is capable of producing nearly optimal layouts for random graphs in $\Omega_n(\psi,p)$.

Let $G=(V,E)$ and define for all $u,v \in V$

$$gc_u(v) = V_2(v) \cap V_{i+2}(u) \quad \forall v \in V_i(u)$$
$$gp_u(v) = V_2(v) \cap V_{i-2}(u) \quad \forall v \in V_i(u)$$

Also let $gc(v)=gc_1(v)$, $gp(v)=gp_1(v)$. The algorithm we will analyze is based on the observation that for $G \in \Psi_n(\psi,p)$ if $u,v \in V_i$ and $v-u$ is not too small, then with high probability $|gc(u)| < |gc(v)| \wedge |gp(u)| > |gp(v)|$.

*Modified Level Algorithm 1 (MLA1)*

For each $u \in V$

Let $\tau$ be any layout that satisfies the following conditions for all $x,y \in V$.

(a) $x \in V_i(u) \wedge y \in V_{i+1}(u) \Rightarrow \tau(x) < \tau(y)$

(b) $1 \le i \le 2 \wedge x,y \in V_i(u) \wedge |gc_u(x)| < |gc_u(y)| \Rightarrow \tau(x) < \tau(y)$

(c) $i \ge 3 \wedge x,y \in V_i \wedge |gp_u(x)| > |gp_u(y)| \Rightarrow \tau(x) < \tau(y)$

Output the layout having minimum bandwidth.

Define *MLA*1$(G)$ as the bandwidth of the layout produced by *MLA*1 on graph $G$.

**Theorem 4.1.** Let $\varepsilon > 0$, $0 < p < 1$ be fixed, $\ln n = o(\psi)$, $\psi \le n/4$. For almost all $G \in \Omega_n(\psi,p)$ *MLA*1$(G) < (1+\varepsilon)\varphi(G)$.

The proof of Theorem 4.1 requires the following lemmas.

**Lemma 4.1.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)$, $2\alpha\le\psi<n$. For almost all $G\in\Psi_n(\psi,p)$, $r'_{i-1}-2\alpha<l'_i\le r'_{i-1}+1$ and $r'_{i-1}+\psi-\alpha\le r'_i\le r'_{i-1}+\psi+\alpha$, for all $i\ge 1$. For $i>1$, $r'_i\le r'_{i-1}+\psi$, and for $i\ne 2$, $r'_{i-1}-\alpha<l'_i$.

[ Figure 3 here ]

*proof.* Figure 3 illustrates the assertions being made. For $1\le i\le 2$ the result is implicit in the proof of Lemma 3.8. For $i\ge 3$, Lemma 3.2 gives $l'_i>r'_{i-3}+2\psi-\alpha$. Since $r'_{i-1}\le r'_{i-3}+2\psi$, $l'_i>r'_{i-1}-2\alpha$. By Lemmas 3.3 and 3.7 $\{r'_{i-3}+\psi+1,\ldots, r'_{i-3}+2\psi-\alpha\}\subseteq V'_{i-1}$ and $\{r'_{i-3}+2\psi+1,\ldots, r'_{i-3}+3\psi-\alpha\}\subseteq V'_i$ and $\{r'_{i-3}+2\psi-\alpha+1,\ldots,r'_{i-3}+2\psi\}\subseteq V'_{i-1}\cup V'_i$. This implies $r'_{i-1}+1\in V'_i$, hence $l'_i\le r'_{i-1}+1$. For $i>2$ it is clear that $r'_i\le r'_{i-1}+\psi\le r'_{i-1}+\psi+\alpha$. $r'_i\ge r'_{i-1}+\psi-\alpha$ follows from Lemma 3.7 and $r'_{i-1}\le r'_{i-3}+2\psi$. □

A consequence of Lemma 4.1 is that at least $\psi-3\alpha$ of the vertices in $V'_i$ are found in a region containing only vertices in $V'_i$. These regions are shown as the solid areas in Figure 4. The regions associated with $V'_i$ and $V'_{i+1}$ are separated by a transition region containing at most $2\alpha$ vertices.

[ Figure 4 here ]

**Lemma 4.2.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha=(1+\varepsilon)\lambda(1-p^2)$, $4\alpha\le\psi\le n/4$. For almost all $G\in\Psi_n(\psi,p)$ $1\le i\le 2\wedge u,v\in V'_i\wedge u-v\ge 4\alpha\Rightarrow |gc(u)|>|gc(v)|$.

*proof.* Lemma 3.1 implies that if $u,v\in V'_i$ and $u-v\ge\alpha$ then $gc(v)\subseteq gc(u)$. It remains only to show that there is some vertex $x$ such that $x\in gc(u)-gc(v)$. Let $x=u+2\psi-\lfloor\alpha\rfloor$. If $u-v\ge 4\alpha$, Lemma 4.1 yields,

$$r'_{i+1}\le r'_{i-1}+2\psi<l'_i+2\psi+2\alpha\le v+2\psi+2\alpha<u+2\psi-2\alpha<x$$

Thus $x\notin V'_{i+1}$, and since by Lemma 3.1 there is a 2-path from $u$ to $x$, $x\in gc(u)$. Since $x>v+2\psi$, $x\notin gc(v)$. □

**Lemma 4.3.** Let $\varepsilon > 0$, $0<p<1$ be fixed, $\alpha = (1+\varepsilon)\lambda(1-p^2)$, $4\alpha \le \psi \le n/4$. For almost all $G \in \Psi_n(\psi,p)$ $i \ge 3 \wedge u,v \in V'_i \wedge u-v \ge 4\alpha \Rightarrow |gp(u)| < |gp(v)|$.

*proof.* Lemma 3.1 implies that if $u,v \in V'_i$ and $u-v \ge \alpha$ then $gp(u) \subseteq gp(v)$. It remains to show that there exists some vertex $x$ in $gp(v)-gp(u)$. Let $x = v-2\psi+\lfloor\alpha\rfloor$. If $u-v \ge 4\alpha$, Lemma 4.1 yields

$$l'_{i-1} > r'_{i-2}-2\alpha \ge r'_i-2\psi-2\alpha \ge u-2\psi-2\alpha \ge v-2\psi+2\alpha > x$$

Thus $x \notin V'_{i-1}$ and since by Lemma 3.1 there is a 2-path from $v$ to $x$, $x \in gp(v)$. Since $x < u-2\psi$, $x \notin gp(u)$. □

**Lemma 4.4.** Let $\varepsilon>0$, $0<p<1$ be fixed, $\alpha = (1+\varepsilon)\lambda(1-p^2)$, $\ln n = o(\psi)$, $\psi \le n/4$. For almost all $G \in \Psi_n(\psi,p)$, $|\tau(u)-u| \le 4\alpha$, where $u \in V$ and $\tau$ is the layout produced by *MLA*1 for which $\tau(1)=1$.

*proof.* By Lemmas 4.1 to 4.3, if $u-v > 4\alpha$ then $\tau(v) < \tau(u)$. Consequently, for any $u$ there can be at most $4\alpha$ vertices $v$ such that $u>v$ and $\tau(u)<\tau(v)$. Similarly, there can be at most $4\alpha$ vertices $w$ such that $u < w$ and $\tau(u) > \tau(w)$. Hence, $|\tau(u)-u| \le 4\alpha$. □

*Proof of Theorem 4.1.* By Lemma 4.4, *MLA*1 will compute a layout in which no vertex is more that $4\alpha$ from the 'right position'. This implies that the bandwidth of the layout output by MLA1 is at most $\psi+8\alpha \le (1+\varepsilon')\psi \le (1+\varepsilon')^2\varphi(G)$ for any fixed $\varepsilon'>0$. Choosing $\varepsilon'$ so that $(1+\varepsilon') = (1+\varepsilon)$ yields the theorem. □

## 5. Pragmatics

This section reports on the results of empirical studies of several modified level algorithms, including *MLA*1, described in the previous section. It also contains some implementation details and analyses of the algorithms' running times.

Four modified level algorithms were studied. They are denoted here as *MLA*1 throuugh *MLA*4. An implementation of *MLA*1 is shown in Figure 5. This procedure returns a layout $\tau$, with $u$ as the starting vertex. The strategy for ordering the vertices within levels is the one described in the previous section. The procedure shown calls several others *Make_mod_levels*$(G, u, V'_0, \ldots, V'_{n-1})$ computes $V'_i(u)$ and returns it in the list $V'_i$ for $0 \le i \le n-1$, using breadth-first-search. The procedures *count_gc()* and *count_gp()* count the number of 'grandchildren' and 'grandparents' for vertices in the levels specified by the last two arguments (for example, the call to *count_gc()* in line [4] counts the grandchildren of all vertices in the first two levels). The procedure *sort*$(L, R(x,y))$ sorts the list $L$ so that $x$ precedes $y$ in the sorted list if and only if $x$ is related to $y$ under $R$. For example, *sort*$(V'_i,\ ngc(x) < ngc(y))$ sorts $V'_i$ so that if $ngc(x) < ngc(y)$ then $x$ precedes $y$ in $V'_i$. The running time of $MLA1$ is dominated by the calculation of the $ngc$ and $ngp$ functions. A straightforward implementation of these gives a running time of $O(n\varphi^2)$. The procedure *make_mod_levels()* can be implemented to run in $O(|E|) = O(n\varphi)$ time, and the sorting steps in lines [6] and [7] require at most $O(n\log n)$.

[ Figure 5 here ]

There are other possible strategies for arranging the vertices within each level. Cuthill and McKee [5], who first suggested the level algorithms, arranged the vertices within levels according to the order in which they were visited by a breadth-first search algorithm. This results in an arbitrary ordering of the first level and arranges each vertex in subsequent levels based on the position of its 'leftmost' neighbor. Cheng [2,3] refined this strategy by ordering the vertices in the first level in increasing order of the number of neighbors in the next level. Adapting this algorithm to the modified level strategy gives the algorithm $MLA2$, which is shown in Figure 6. $MLA2$ calls the procedure *count_ch()*, which counts the number of neighbors each vertex has in the 'next' level. As with *count_gc()*

and *count_gp()*, the calculation is done only for those levels specified by the last two arguments (in this case, just the first level). This can be done in $O(\varphi^2)$ time, while the remainder of *MLA*2 can be done in $O(n\varphi)$ time.

[ Figure 6 here ]

The procedure *MLA*3, shown in Figure 7 is a cross between *MLA*1 and *MLA*2. It uses the strategy of *MLA*1 to order the vertices in the first level, then reverts to the strategy of *MLA*2 for all subsequent levels. Computing *ngc* for the vertices in the first level requires $O(\varphi^3)$ time. The remainder of *MLA*3 can be done in $O(n\varphi)$.

[ Figure 7 here ]

*MLA*4 is a refinement of *MLA*3 designed to improve the running time when the bandwidth is fairly large. Instead of using the number of 'grandchildren' to order the vertices in the first level, it uses the number of paths to grandchildren. This can be computed more quickly, since it eliminates the necessity of throwing out duplicates. The total running time of *MLA*4 is $O(n\varphi)$.

*MLA*2 through *MLA*4 are more difficult to analyze than *MLA*1 because decisions made in ordering each level affect the ordering of subsequent levels. Consequently, one might expect that errors made in ordering the early levels could accumulate and cause large errors further on. Experimental results suggest that in fact this does not happen, that the process is self-limiting. However, straightforward analytical techniques for bounding the error give unsatisfactory results.

Figures 8 through 10 summarize the results of a series of experiments that were undertaken to verify the theoretical performance bounds described in the previous sections for *MLA*1, provide tighter bounds for graphs of moderate size and compare *MLA*1 to the other modified level algorithms. For each of the data points shown in Figure 8, ten random graphs in $\Psi_n(n/4, \frac{1}{2})$ were generated and each of the algorithms was run. For each algorithm, these ten results were averaged and the difference between these averages and $n/4$ were plotted. The results show that all the algorithms produce good layouts. All of the results are within 20% of $n/4$ and the best are within 2%.

[ Figure 8 here ]

Figure 9 shows the measured execution times for these runs. (The algorithms were coded in the C programming language and run on a VAX 11/750 under Unix*.) Here, *MLA*2 and *MLA*4 enjoy a substantial advantage. Of course, this speed advantage is directly related to the large value of $\psi$ relative to $n$. For smaller values of $\psi$ the differences would be less.

[ Figure 9 here ]

One last set of results is shown in Figure 10. This shows how the performance of the algorithms deteriorates as $\psi$ becomes large relative to $n$. *MLA*1 deteriorates first, when $\psi \approx n/4$. This is because, the strategy used to order the levels becomes less effective when $V'_4$ becomes much smaller than $\psi$. *MLA*3 is not affected by this phenomenon until $\psi \approx n/3$ since the 'grandchildren' strategy is used only to order the first level. *MLA*2 and *MLA*4 are more robust, maintaining their good performance until $\psi \approx n/2$. At this point all four degenerate from modified level algorithms to level algorithms.

[ Figure 10 here ]

* Unix is a trademark of AT&T.

## 6. Selection of Starting Vertices

Up until this point we have largely ignored the question of how one selects a good starting vertex in the modified level algorithm. Of course, the brute force solution is simply to try all possibilities and pick the best result. This adds a factor of $n$ to the running times quoted in the previous sections, but does ensure the best possible choice. In this section, we consider strategies that permit us to select small sets of candidate starting vertices, that with high probability, contain a good choice.

The most obvious strategy (suggested by Cuthill and McKee) is to concentrate on vertices with small degree. For $G \in \Psi_n(\psi,p)$ it's reasonable to expect the degree of vertex 1 will be smaller than the degree of most other vertices. The following lemma puts a probable upper bound on the number of low degree vertices that need to be tried to obtain near optimal performance. For $G=(V,E)$, define $ld(G)=\{v \in V | d(v) \le d(1)\}$.

**Lemma 6.1.** Let $\varepsilon > 0$, $0<p<1$ be fixed, $12(1+\varepsilon)(1/p)\ln n \le \psi < n$. For almost all $G \in \Psi_n(\psi,p)$, $|ld(G)| < 4\sqrt{(3/p)(1+\varepsilon)\psi \ln n}$.

*proof.* Let $0<\alpha \le \psi p/2$. By Lemma 3.5

$$P(d(1) \ge \psi p + \alpha) < e^{-\alpha^2/3\psi p}$$

For $v \in V$ such that $(2\alpha/p)<v<n-(2\alpha/p)$

$$P(d(v) \le \psi kp + \alpha) < e^{-\alpha^2/2\psi p}$$

Letting $\alpha = \sqrt{3(1+\varepsilon)\psi p \ln n}$ yields

$$P(d(v)<d(1)) < 2e^{-\alpha^2/3\psi p} = 2n^{-(1+\varepsilon)}$$

Since there are $<n$ such vertices $v$,

$$P(\exists v \,|(2\alpha/p)<v<n-(2\alpha/p) \wedge d(v)<d(1)) < 2n^{-\varepsilon} \to 0$$

Consequently there are at most $4\alpha/p$ vertices in $ld(G)$.

Lemma 6.1 gives us a way of ensuring a good starting vertex. The cost is an added factor of $O(\psi \ln n)$ in the running time.

The next theorem suggests another method for identifying a good starting vertex. Let $L_x(G)$ be the layout of $G$ produced by $MLA1$ when $x$ is the starting vertex and let $MLA1_x(G)$ be the bandwidth of $G$ with respect to $L_x(G)$.

**Theorem 6.1.** Let $0<p<1$ be fixed, $\ln n = o(\psi)$, $\psi \le n/16$. For almost all $G \in \Psi_n(\psi,p)$ $(x \in V \wedge \tau = L_x(G) \wedge y = \tau^{-1}(n)) \Rightarrow MLA1_y(G) < (1+\varepsilon)\varphi(G)$.

The procedure suggested by Theorem 6.1 is this. Pick an arbitrary vertex $x$ and run $MLA1$ with $x$ as the starting vertex. Let $y$ be the 'rightmost vertex' in the resulting layout. Now, re-run $MLA1$ with $y$ as the starting vertex. Theorem 6.1 states that the resulting layout is close to optimal. The proof of Theorem 6.1 requires the following lemmas.

**Lemma 6.2.** Let $\varepsilon > 0$, $0<p<1$ be fixed, $\alpha = (1+\varepsilon)\lambda(1-p^2)$, $\ln n = o(\psi)$, $\psi \le n/16$. For almost all $G \in \Psi_n(\psi,p)$ $(x \in V \wedge \tau = L_x(G) \wedge y = \tau^{-1}(n)) \Rightarrow (y < 4\alpha \vee y > n-4\alpha)$.

*proof.* Let $x \in V$, $\tau = L_x(G)$ and $y = \tau^{-1}(n)$. Also let $G_l$ be the subgraph induced by $\{1,2,\ldots,x\}$ and let $G_r$ be the subgraph induced by $\{x,\ldots,n\}$. Note that $G_l \in \Psi_x(\psi,p)$ and $G_r \in \Psi_{n-x+1}(\psi,p)$. Next, let $\tau_l = L_x(G_l)$ and $\tau_r = L_x(G_r)$ and note that for $u,v \in \{1,\ldots,x\}$, $\tau(u) < \tau(v) \Leftrightarrow \tau_l(u) < \tau_l(v)$. Similarly, for $u,v \in \{x,\ldots,n\}$, $\tau(u) < \tau(v) \Leftrightarrow \tau_r(u) < \tau_r(v)$. Now, let $y_l = \tau_l^{-1}(x)$ and $y_r = \tau_r^{-1}(n-x+1)$ and note that $y \in \{y_l, y_r\}$. We consider three cases.

*Case 1.* $n/4 \le x \le 3n/4$. Suppose $y = y_l$. Since the number of vertices in $G_l$ is at least $n/4$ and $\psi \le n/16$, we can apply Lemma 4.4 to $G_l$ and conclude that $y < 4\alpha$. Similarly, if $y = y_r$, we can apply Lemma 4.4 to $G_r$ yielding $y > n-4\alpha$.

*Case 2.* $x < n/4$. By Theorem 7.3, $d(x,y_l) < 2\frac{x}{\psi} + 3 < \frac{n}{2\psi} + 3$. Also, observe that $d(x,n) \ge \frac{n-x}{\psi} > \frac{3n}{4\psi}$. Since $\psi \le n/16$, $d(x,y_l) < \frac{n}{2\psi} + 3 < \frac{n}{2\psi} + \frac{n}{4\psi} < d(x,n)$. This implies that $y = y_r$. Applying Lemma 4.4, we conclude that $y > n - 4\alpha$.

*Case 3.* $x > 3n/4$. Similar to case 2. □

*Proof of Theorem 6.1.* Let $x \in V$, $\tau = L_x(G)$ and $y = \tau^{-1}(n)$. By Lemma 6.2, either $y < 4\alpha$ or $y > n - 4\alpha$. Since the two cases are symmetric, we will only discuss the former. Define $G_r$ and $\tau_r$ as in the proof of Lemma 6.2. By Lemma 3.1, every vertex in $\{1, \ldots, y-1\}$ is connected to y by a 2-path and no vertex in $\{1, \ldots, y-1\}$ is adjacent to any vertex in $V'_3(y)$. Consequently, $\{1, \ldots, y-1\} \subseteq V'_1$. Let $\{u,v\} \in E$, and consider the following three cases.

*Case 1.* $\{u,v\} \subseteq \{y, \ldots, n\}$. By Lemma 4.4, $|\tau_r(u) - \tau_r(v)| \leq \psi + 8\alpha$. Consequently, $|\tau(u) - \tau(v)| \leq \psi + 12\alpha$.

*Case 2.* $\{u,v\} \subseteq \{1, \ldots, y-1\}$. Since $\{u,v\} \subseteq V'_1(y)$ and by Lemma 3.8, $|V'_1(y)| \leq 4\alpha + (\psi + \alpha) = \psi + 5\alpha$, it follows that $|\tau(u) - \tau(v)| \leq \psi + 5\alpha$.

*Case 3.* $u \in \{1, \ldots, y-1\}$, $v \in \{y, \ldots, n\}$. Because $u < y$, $v < y + \psi$. By Lemma 4.4, $|\tau(v) - v| \leq 4\alpha$, giving $\tau(v) < y + \psi + 4\alpha < \psi + 8\alpha$. Since $u \in V_1'$, $\tau(u) \leq \psi + 5\alpha$. Thus, $|\tau(u) - \tau(v)| \leq \psi + 8\alpha$.

In all three cases above, we conclude that $|\tau(u) - \tau(v)| \leq \psi + 12\alpha \leq (1+\varepsilon')\psi \leq (1+\varepsilon')^2 \varphi(G)$ for any fixed $\varepsilon' > 0$. Choosing $\varepsilon'$ so that $(1+\varepsilon')^2 = (1+\varepsilon)$ yields the theorem. □

The method for selecting a starting vertex outlined above can be refined in several directions. One way is to run *MLA*1 several times, each time using the rightmost vertex from the previous run as the starting vertex for the next run. This extends the applicability of the method to larger values of $\psi$. Another refinement is to run *MLA*1 several times as just described, but then take the $4\alpha$ rightmost vertices from the last run and use these as a set of candidate starting vertices. With high probability, either vertex 1 or vertex $n$ is in this set. The results obtained in this way may be somewhat closer to optimal, but the cost is an extra $O(\ln n)$ factor in the running time.

## 7. Properties of Random Graphs

This section is largely independent and examines several properties of random graphs, particularly graphs in $\Psi_n(\psi,p)$. The following theorem is a special case of a result proved by Erdös and Renyi in [6].

**Theorem** 7.1. Let $-1<\varepsilon<1$ be fixed, $p=(1+\varepsilon)\frac{\ln n}{n}$, $G\in\Gamma_n(p)$. If $\varepsilon>0$, $G$ is almost always connected. If $\varepsilon<0$, $G$ is almost always disconnected.

The following is a similar result for random graphs with small bandwidth.

**Theorem** 7.2. Let $-1<\varepsilon<1$ be fixed, $0<p<1$, $\psi=\frac{1}{2}(1+\varepsilon)\lambda(1-p)$, $\psi\to\infty$. If $\varepsilon>0$ then almost all $G\in\Psi_n(2\psi,p)$ are connected. If $\varepsilon<0$ then almost all $G\in\Psi_n(\psi,p)$ are disconnected.

To prove Theorem 7.2 we need to introduce another probability distribution and prove two lemmas. Let $n$ and $\psi$ be positive integers, $\psi<n$, $0<p<1$, and let $G=(V,E)$ be a random variable defined by the following experiment.

- Let $V=\{1,2,\ldots,n\}$. For each pair $u,v$ $1\le u<v\le n$ and $|u-v|\le\psi\vee|u-v|\ge n-\psi$ include the edge $\{u,v\}$ in $E$ with probability $p$.

The probability distribution defined by this experiment is denoted $\Psi_n^c(\psi,p)$.

**Lemma** 7.1. Let $-1<\varepsilon<1$ be fixed, $0<p<1$, $\psi=\frac{1}{2}(1+\varepsilon)\lambda(1-p)$, $1\le\psi\le\frac{n}{2}$, $G\in\Psi_n^c(\psi,p)$. If $\varepsilon>0$ then $G$ almost always contains no isolated vertex. If $\varepsilon<0$ then $G$ almost always contains at least one isolated vertex.

*proof.* First part - $\varepsilon>0$. Let

$$X_v = \begin{cases} 1 & \text{if } v \text{ is isolated} \\ 0 & \text{if } v \text{ is not isolated} \end{cases}$$

$$X = X_1 + X_2 + \cdots + X_n$$

$$\mu = E(X) = \sum_{v=1}^{n} E(X_v) = n(1-p)^{2\psi}$$

$$P(X \geq 1) \leq \mu = n(1-p)^{2\psi} = n^{-\varepsilon} \to 0$$

This completes the proof of the first part.

Second part - $\varepsilon < 0$. Let $X, X_1, \ldots, X_n$ be defined as before.

$$E(X^2) = \sum_{u=1}^{n} \sum_{v=1}^{n} E(X_u X_v)$$

$$= \sum_{u=1}^{n} \sum_{v=1}^{n} P(u \text{ and } v \text{ are both isolated})$$

$$= n(1-p)^{2\psi} + 2\psi n(1-p)^{4\psi-1} + n(n-2\psi-1)(1-p)^{4\psi}$$

By Chebyshev's inequality,

$$P(X=0) \leq \frac{\sigma^2}{\mu^2} = \frac{E(X^2)-\mu^2}{\mu^2} = \frac{1}{\mu} + \frac{2\psi p}{n(1-p)} - \frac{1}{n} < n^{\varepsilon} + (1+\varepsilon)\frac{p \ln n}{n(1-p)\ln(1/(1-p))}$$

The function $\frac{p}{(1-p)\ln(1/(1-p))}$ gets large as $p \to 1$. However, $1 \leq \psi < (1+\varepsilon)\lambda(1-p) \Rightarrow p \leq 1 - n^{-(1+\varepsilon)}$. Hence,

$$P(X=0) < n^{\varepsilon} + (1+\varepsilon)\frac{\ln n}{n^{-\varepsilon}\ln n^{1+\varepsilon}} = 2n^{\varepsilon} \to 0 \qquad \square$$

**Lemma 7.2.** Let $0 < \varepsilon \leq p < 1$ where $\varepsilon$ is fixed, and let $G \in \Gamma_n(p)$. Then $P(D(G) > 2) \leq \binom{n}{2}(1-\varepsilon^2)^{n-2}$.

*proof.* Let $u$ and $v$ be any two vertices in $G$. The number of possible 2-paths between them is $n-2$ and the probability that any one of them is absent is $1-p^2$. Hence the probability that $u$ and $v$ are not connected by a 2-path is $(1-p^2)^{n-2}$. Consequently, the probability that any pair of vertices is not connected by a 2-path is

$$\le \binom{n}{2}(1-p^2)^{n-2} \le \binom{n}{2}(1-\varepsilon^2)^{n-2} \qquad \square$$

*Proof of Theorem 7.2.* First part - $\varepsilon > 0$. $G$ is connected if the first $2\psi$ vertices induce a connected subgraph and all other vertices have at least one edge to a lower numbered vertex. By Lemma 7.2, if $p > \alpha$ for some $\alpha > 0$ then the probability that the first $2\psi$ vertices induce a subgraph of diameter $>2$ is $\le \binom{2\psi}{2}(1-\alpha^2)^{2\psi-2} \to 0$. Hence, if $p$ is bounded below, the first $2\psi$ vertices almost always induce a connected subgraph. If on the other hand $p \to 0$ we must use Theorem 7.1 to establish that the first $2\psi$ vertices induce a connected subgraph. This requires that we show that there exists some $\gamma > 0$ such that $p \ge (1+\gamma)\frac{\ln(2\psi)}{2\psi}$. From the hypothesis of the theorem

$$\frac{p(2\psi)}{\ln(2\psi)} = (1+\varepsilon)\frac{p \ln n}{\ln(1/(1-p))\ln(2\psi)} > (1 + \frac{\varepsilon}{2})$$

for large enough $n$ since $p \to \ln(1/(1-p))$ as $p \to 0$ and $n > 2\psi$. Now, the probability that any of the remaining vertices have no edges to lower numbered vertices is $< n(1-p)^{2\psi} = n^{-\varepsilon} \to 0$. This completes the proof of the first part of Theorem 7.2.

Now let $\varepsilon < 0$ and let $G' \in \Psi_n^\varepsilon(\psi, p)$. Clearly, $P(G$ is connected$) \le P(G'$ is connected) and since by Lemma 7.1, $G'$ is almost always disconnected, it follows that $G$ is almost always disconnected. $\square$

Let $D(G)$ be the diameter of $G$. A simple lower bound for the bandwidth of any connected graph is given by

$$\varphi(G) \ge \omega(G) = \left\lceil \frac{n-1}{D(G)} \right\rceil \qquad (7.1)$$

since the first and last vertices in any optimal layout are connected by a path of length at most $D(G)$ and hence at least one edge in this path has length $\ge \omega(G)$. Chvatal [4] was apparently the first to notice this. A more general lower bound is given by

$$\varphi(G) \geq \omega^*(G) = \max_{G'} \omega(G')$$

where $G'$ ranges over all connected subgraphs of $G$. The graph shown in Figure 11 shows that $\omega^*(G) \neq \varphi(G)$ in general. It is natural to ask if there is any constant $c$ such that for all connected graphs $\varphi(G) \leq c\,\omega^*(G)$. Ronald Graham has pointed out that this is not the case. The argument is given in [11]. In spite of this result however, we can show that for almost all $G \in \Psi_n(\psi,p)$, $D(G) < (1+\varepsilon)\dfrac{n}{\varphi(G)} + 3$.

[ Figure 11 here ]

Let $G = (V,E) \in \Psi_n(\psi,p)$ and let $Q = (v_0, \ldots, v_r)$ be the unique path in $G$ that satisfies

$$\begin{aligned}
&v_0 = 1 \\
&v_i = \max\{u \in V \mid \{u, v_{i-1}\} \in E\} \quad 1 \leq i \leq r \\
&v_i < n - \psi \quad 1 \leq i < r \\
&v_r \geq n - \psi
\end{aligned}$$

Next, let $x_i = v_i - v_{i-1}$ $(1 \leq i \leq r)$ and let $x = \sum_{i=1}^{r} x_i$. In what follows we will derive a probable upper bound on $x$ which will be used to obtain an upper bound on $r$. Its is clear that for $1 \leq i \leq r$

$$\begin{aligned}
&\mathrm{P}(x_i = 0) = p \\
&\mathrm{P}(x_i = 1) = (1-p)p \\
&\qquad \cdots \\
&\mathrm{P}(x_i = j) = (1-p)^j p \quad 1 \leq j \leq \psi
\end{aligned}$$

For $j$ larger than $\psi$, $\mathrm{P}(x_i = j)$ depends on whether or not $v_i \leq \psi$. For simplicity we will assume that $\mathrm{P}(x_i = j) = (1-p)^j p$ for all $k \geq 0$. The error committed by this approximation vanishes as $\psi \to \infty$. This can be used to prove the following.

**Theorem 7.3.** Let $\varepsilon > 0$, $0 < p < 1$ be fixed, $\lambda(1-p^2) \leq \psi < n$. For almost all $G \in \Psi_n(\psi,p)$ $D(G) < (1+\varepsilon)\dfrac{n}{\psi} + 3$.

The proof of Theorem 7.3 requires the following lemmas.

**Lemma 7.3.** Let $0 < p < 1$, $\varepsilon > 0$, $\psi < n$, $\psi \to \infty$. For almost all $G \in \Psi_n(\psi, p)$

$$P(x > (1+\varepsilon)r(1-p)/p) \le \frac{1}{\varepsilon^2 r(1-p)}.$$

*proof*. Using the approximation discussed above,

$$E(x_i) = \sum_{j=1}^{\infty} j(1-p)^j p = p\frac{(1-p)}{p^2} = \frac{1-p}{p}$$

$$E(x_i^2) = \sum_{j=1}^{\infty} j^2(1-p)^j p = p \sum_{j=1}^{\infty} (1-p)^j(1 + 3 + \cdots + 2j-1)$$

$$= p\left[\frac{(1-p)}{p} + \frac{3(1-p)^2}{p} + \frac{5(1-p)^3}{p} + \cdots\right]$$

$$= 2\left[(1-p) + 2(1-p)^2 + 3(1-p)^3 + \cdots\right] - \left[(1-p) + (1-p)^2 + (1-p)^3 + \cdots\right]$$

$$= 2\,\frac{1-p}{p^2} - \frac{1-p}{p}$$

Let $\mu$ be the mean and $\sigma^2$ the variance of $x$.

$$\mu = rE(x_i) = r\,\frac{1-p}{p}$$

$$\sigma^2 = r(E(x_i^2) - E^2(x_i)) = r\,\frac{1-p}{p^2}$$

By Chebyshev's inequality

$$P(x > (1+\varepsilon)\mu) \le \frac{\sigma^2}{\varepsilon^2\mu^2} = \frac{r(1-p)/p^2}{\varepsilon^2 r^2(1-p)^2/p^2} = \frac{1}{\varepsilon^2 r(1-p)} \qquad \square$$

**Lemma 7.4.** Let $0 < p < 1$, $\varepsilon > 0$, $\psi < n$, $\psi \to \infty$. For almost all $G \in \Psi_n(\psi, p)$

$$P\left(r \ge \frac{n}{\psi - (1+\varepsilon)(1-p)/p}\right) \le \frac{1}{\varepsilon^2 r(1-p)}.$$

*proof*. From the definition, $v_r - 1 = r\psi - x$, hence

$$n - 1 \ge v_r - 1 = r(\psi - \frac{x}{r})$$

$$r < \frac{n}{\psi - x/r}$$

The result now follows from Lemma 7.3. $\square$

*Proof of Theorem 7.3.* By Lemma 7.4

$$P\left(r \ge \frac{n}{\psi-(1+\alpha)(1-p)/p}\right) \le \frac{1}{\alpha^2 r(1-p)}$$

holds for any $\alpha > 0$. Letting $\alpha = \psi/\sqrt{n}$,

$$P\left(r \ge \frac{n}{\psi(1-(1-p)/p\sqrt{n})}\right) \le \frac{n}{\psi^2 r(1-p)}$$

Consider two cases. If $\psi > n/2$ then

$$\frac{n}{\psi^2 r(1-p)} < \frac{4}{n(1-p)} \to 0$$

and if $\psi \le n/2$

$$\frac{n}{\psi^2 r(1-p)} \le \frac{n}{\psi^2((n/\psi)-1)(1-p)} \le \frac{2}{\psi(1-p)} \to 0$$

Thus for any fixed $\varepsilon > 0$, $r < (1+\varepsilon)\frac{n}{\psi}$ with high probability for large enough n. By Lemma 3.1 there is a 2-path from each $u \in V$ to some $v_i$ $1 \le i \le r$. This implies $D(G) \le r+3$. □

By Lemma 7.2, if $p \ge \varepsilon > 0$ then for almost all $G \in \Gamma_n(p)$, $D(G)=2$. When $p$ is allowed to approach zero as $n$ gets large the diameter can become larger. By Theorem 7.1, when $p$ is much less than $\frac{\ln n}{n}$ the graph is likely to be disconnected. We now consider the probable diameter of random graphs in $\Gamma_n(p)$ when $p = c\frac{\ln n}{n}$ and $c$ is a constant. We do this by examining the probable size of $V_1, V_2, \ldots$. Let $n_i = |V_i|$. Clearly,

$$\begin{aligned}
&n_0 = 1\\
&n_2 \in B(n-1,p)\\
&n_3 \in B(n-(n_1+1), 1-(1-p)^{n_1})\\
&\cdots\\
&n_{k+1} \in B(n-s_k, 1-(1-p)^{n_k})
\end{aligned}$$

where $s_k = \sum_{j=0}^{k} n_j$. Define $\hat{n}_0=1$, $\hat{n}_{k+1} = (n-\hat{s}_k)(1-(1-p)^{\hat{n}_k})$, where $\hat{s}_k = \sum_{j=0}^{k}\hat{n}_j$. We can use $\hat{n}_k$ as an estimator for $n_k$. Figure 12 gives values of $\hat{n}_k$ for particular values of $n$ and $p$. The sequence grows very rapidly until a large fraction of the vertices in the graph has been 'captured'. Then the remaining vertices are taken in the last step. The figure also gives values of the function $(np)^k$. For $k \le 3$, $(np)^k$ gives an excellent estimate for $\hat{n}_k$.

[ Figure 12 here ]

Let $k^*$ be such that $s_{k^*} = n$. In the following we show that for $k \le k^*-2$, $n_k > (np/8)^k$ with high probability. We can use this to get a probabilistic upper bound on $k^*$ and hence on $D(G)$. The main results are

**Theorem 7.4.** Let $c>8$ be fixed, $p = c\frac{\ln n}{n}$; $\gamma = np/8$. For almost all $G \in \Gamma_n(p)$, $1 \le k \le k^*-2 \Rightarrow n_k > \gamma^k$.

**Theorem 7.5.** Let $c>8$ be fixed, $p = c\frac{\ln n}{n}$; $\gamma = np/8$. For almost all $G \in \Gamma_n(p)$, $D(G) \le 2\left\lceil\left\lceil\frac{\ln(1/p)}{\ln\gamma}\right\rceil + 2\right\rceil$.

The proof of Theorem 7.4 is contained in the following lemmas.

**Lemma 7.5.** Let $c>8$ be fixed, $p = c\frac{\ln n}{n}$; $\gamma = np/8$. For almost all $G \in \Gamma_n(p)$ $1 \le k \le k^*-2 \wedge n_{k-1} < 1/p \wedge s_{k-1} \le n/2 \Rightarrow n_k > \gamma n_{k-1}$.

*proof.* Since $n_k \in B(n - s_{k-1}, 1-(1-p)^{n_{k-1}})$,

$$\bar{n}_k = E(n_k) = (n - s_{k-1})(1-(1-p)^{n_{k-1}}) \ge \frac{n}{2}p\, n_{k-1}(1 - p\, n_{k-1}/2) > \frac{np}{4}n_{k-1} = 2\gamma n_{k-1}$$

By Lemma 3.4

$$P(n_k \le \gamma n_{k-1}) \le P(n_k \le \tfrac{1}{2}\bar{n}_k) < e^{-\bar{n}_k/8} < e^{-\gamma n_{k-1}/4}$$

Let $A_k$ denote the event $n_k \le \gamma n_{k-1}$. The probability that there exists a $k$ satisfying the hypothesis of the lemma, such that $A_k$ holds is

$$\le P(A_1) + P(A_2 \mid \bar{A}_1) + \cdots + P(A_{k^*-2} \mid \bar{A}_1 \cdots \bar{A}_{k^*-3})$$
$$\le e^{-\gamma/4} + e^{-\gamma^2/4} + \cdots + e^{-\gamma^{k^*-2}/4} \to 0$$

**Lemma 7.6.** Let $c > 8$ be fixed, $p = c\frac{\ln n}{n}$. For almost all $G \in \Gamma_n(p)$ $1 \le k \le k^* - 2 \wedge s_{k-1} \le n/2 \Rightarrow n_{k-1} < 1/p$.

*proof.* Assume that $n_{k-1} \ge 1/p$. Then since $n_k \in B(n - s_{k-1}, 1-(1-p)^{n_{k-1}})$,

$$\bar{n}_k = E(n_k) = (n - s_{k-1})(1-(1-p)^{n_{k-1}}) \ge \frac{n}{2}(1-1/e) > n/4$$

By Lemma 3.4

$$P(n_k \le n/8) \le P(n_k \le \tfrac{1}{2}\bar{n}_k) < e^{-\bar{n}_k/8} < e^{-n/32} \to 0$$

Hence, assume $n_k > n/8$. Then the probability that any of the remaining vertices is not adjacent to something in $V_k$ is

$$\le (n - s_k)(1-p)^{n_k} < ne^{-np/8} = n^{1-c/8} \to 0$$

This implies that $k^* \le k+1$ which is a contradiction. □

**Lemma 7.7.** Let $c > 8$ be fixed, $p = c\frac{\ln n}{n}$. For almost all $G \in \Gamma_n(p)$, $1 \le k \le k^* - 2 \Rightarrow s_{k-1} \le n/2$.

*proof.* Assume that $s_{k-1} > n/2$ and let $k'$ be the smallest integer such that $s_{k'} > n/2$. By Lemma 7.6, $n_{k'-1} < 1/p$ and by Lemma 7.5, for all $k \le k'$, $n_k > \gamma n_{k-1}$, where $\gamma = np/8$. Since for large $n$, $\gamma > 2$, we have $s_k > 2s_{k-1}$ for $k \le k'$. Consequently $n_{k'} = s_{k'} - s_{k'-1} > s_{k'}/2 > n/4$. Now, the probability that any of the vertices in $V - (V_0 \cup V_1 \cup \cdots \cup V_{k'})$ is not adjacent to some vertex in $V_{k'}$ is

$$< (n - s_{k'})(1-p)^{n_{k'}} < ne^{-np/4} = n^{1-c/4} \to 0$$

This implies that $k^* \le k' + 1$ which is a contradiction. □

This establishes Theorem 7.4.

*Proof of Theorem 7.5.* Note that $D(G) \le 2k^*$. Let $k'$ be the smallest integer such that $\gamma^{k'} \ge 1/p$. Clearly $k' = \left\lceil \frac{\ln(1/p)}{\ln \gamma} \right\rceil$. If $k' > k^* - 2$, we're done. If $k' \le k^* - 2$ we can apply Theorem 7.4 giving $n_{k'} \geq 1/p$. By the argument used in the proof of Lemma 7.6, this implies $k^* \leq k' + 2$. □

## 8. Conclusions

The work reported here is part of an ongoing research effort aimed at developing better methods for evaluating the performance of heuristic algorithms for hard combinatorial problems. This is an area where the usual analytical tools often fail us, and the available results are unsatisfying. To be useful, a performance evaluation must satisfy two basic criteria. First, it must be able to explain the practical success of popular algorithms and the differences observed between competing algorithms. Second, it should provide insight suggesting new and better algorithms, and supply a basis for making predictions about their success in practice. The ultimate utility of such a method depends on how accurately it predicts the performance of algorithms in real applications.

Worst-case analysis is inadequate for evaluating the performance of heuristics for bandwidth minimization, precisely because it fails to satisfy the criteria given above. As shown in Theorems 2.1 and 2.2, even probabilistic analysis can be of little use if one is naive in choosing the probability distribution. The key to the work reported here is in the choice of distribution. Because $\Psi_n(\psi, p)$ generates only graphs having bandwidth $\leq \psi$, we can explore properties that are common to most such graphs, even though they may be rare among unrestricted graphs. The success of heuristics like the level algorithms is due to the fact that they exploit these properties.

The methods used in this paper at least partially satisify the criteria outlined above. They provide the first satisfactory analytical explanation of the practical success of the level algorithms and they provide insight leading to methods, which at least in theory are better. If the modified level algorithms fare as well in practice as they do on paper, the utility of these methods will have been demonstrated.

**References**


[1] Angluin, D., L. G. Valiant. "Fast Probabilistic Algorithms for Hamiltonian Circuits and Matchings". In *Journal of Computer and System Sciences 18*, 155-193, 1979.

[2] Cheng, K. Y. "Minimizing the Bandwidth of Sparse Symmetric Matrices". In *Computing 11*, 103-110, 1973.

[3] Cheng, K. Y. "Note on Minimizing the Bandwidth of Sparse Symmetric Matrices". In *Computing 11*, 27-30, 1973.

[4] Chvatal, V. "A Remark on a Problem of Harary". In *Czechoslovak Math Journal 20*, 95, 1970.

[5] Cuthill, E., J. McKee. "Reducing the Bandwidth of Sparse Symmetric Matrices". In *ACM National Conference Proceedings 24*, 157-172, 1969.

[6] Erdos, P., A. Renyi. "On Random Graphs I.". In , Publicationes Mathematicae, 290-297, 1959.

[7] Garey, Michael R., R. L. Graham, David S. Johnson, D. E. Knuth. "Complexity Results for Bandwidth Minimization". In *SIAM Journal of Applied Mathematics 34*, 477-495, 5/78.

[8] Monien, B., I. H. Sudborough. "Bandwidth Problems in Graphs". In *Proceedings 18-th Annual Allerton Conference on Communication, Control, and Computing*, 650-659, 1980.

[9] Papadimitriou, Christos H. "The NP-Completeness of the Bandwidth Minimization Problem". In *Computing 16*, 263-270, 1976.

[10] Saxe, James B. "Dynamic Programming Algorithms for Recognizing Small-Bandwidth Graphs in Polynomial Time", Carnegie-Mellon University Technical Report, 1980.

[11] Turner, Jonathan S. "Bandwidth and Probabilistic Complexity", Northwestern University, Ph.D. thesis, 6/82.

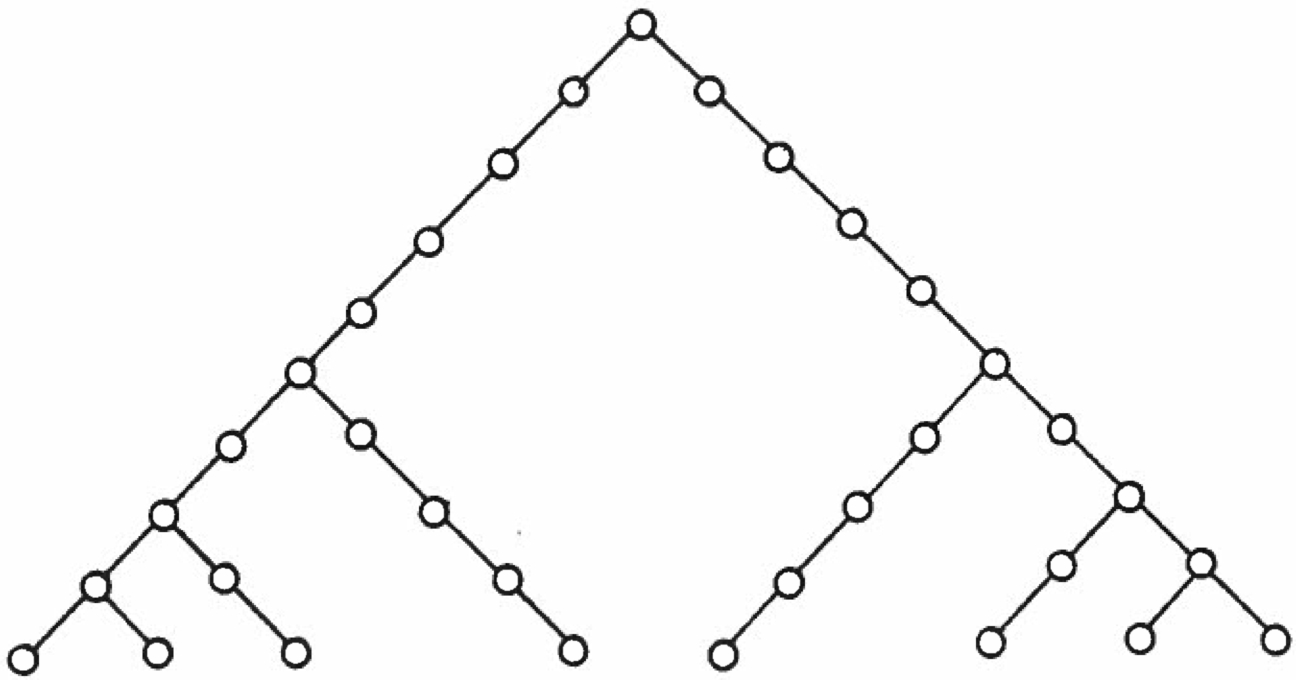


**Figure 1. Tree Demonstrating Poor Worst-Case Performance of Level Algorithms**

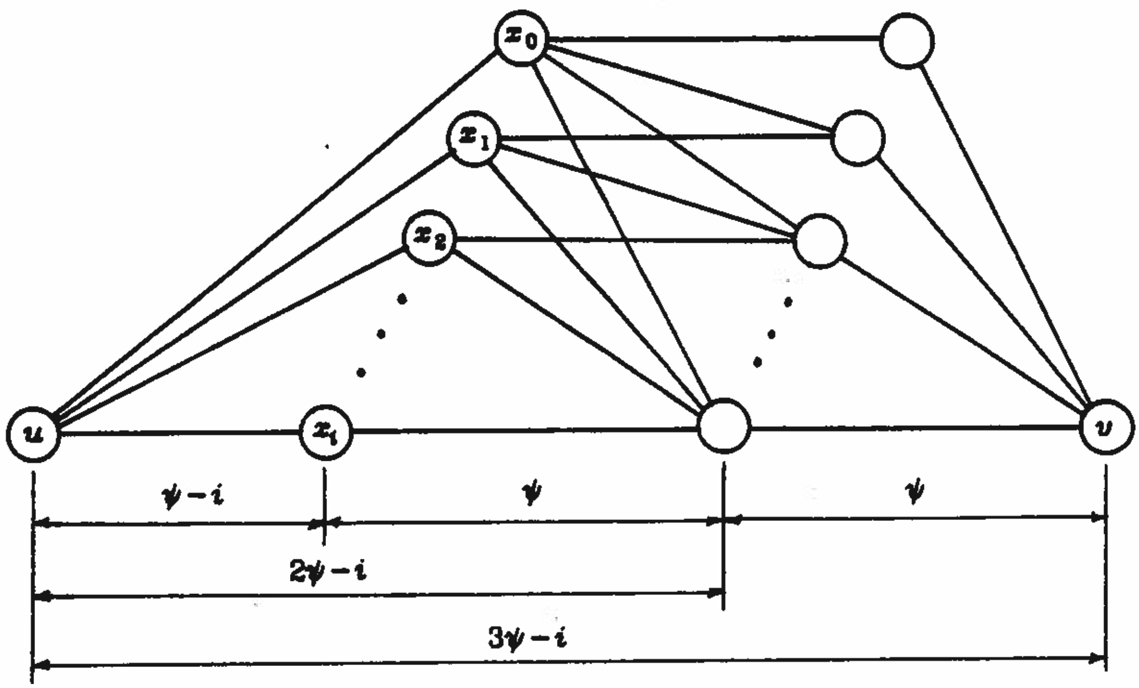



Figure 2. Definition of $x_i$s

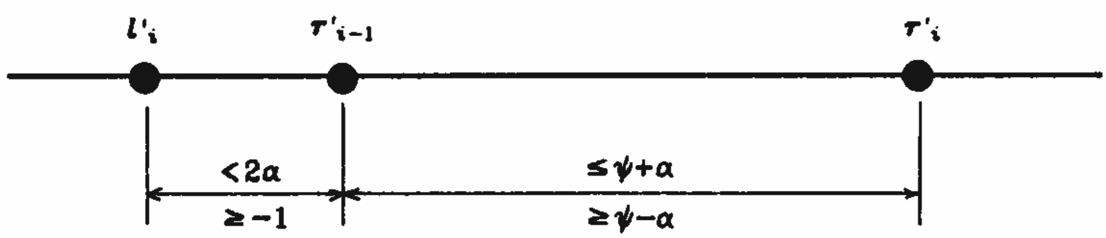



**Figure 3. Illustration for Lemma 4.1**

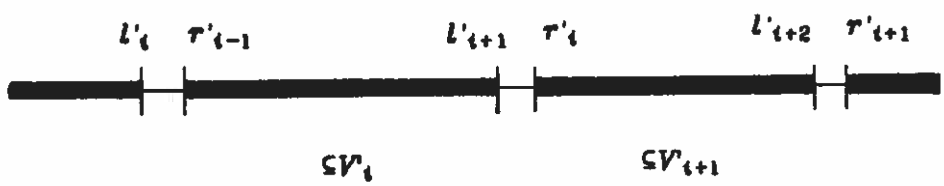



**Figure 4. Separation of Vertices into Regions According to Distance from 1**

```
procedure MLA1(G=(V,E),u,τ)
        n ← |V|;
        make_mod_levels(G,u,V'_0, . . . . , V'_{n-1});
        count_gc(G,u,V'_0, . . . . , V'_{n-1},ngc,1,2);
        count_gp(G,u,V'_0, . . . . , V'_{n-1},ngp,3,n-1);
        for i ← 1 to 2 do sort(V'_i, ngc(x) < ngc(y)) od;
        for i ← 3 to n-1 do sort(V'_i, ngp(x) > ngp(y)) od;
        next ← 1;                { next position in layout }
        for i ← 0 to n-1 do
                for x ∈ V'_i do τ(x) ← next; next ← next+1 od;
        od;
        return;
end;
```

Figure 5. Implementation details for *MLA*1

```
procedure MLA2(G=(V,E),u,τ)
        n ← |V|;
        make_mod_levels(G,u,V'_0, . . . , V'_{n-1});
        count_ch(G,u,V'_0, . . . , V'_{n-1},nch,1,1);
        sort(V'_1,nch(x) < nch(y));
        for x ∈ V do τ(x) ← 0 od;     { 0 denotes undefined }
        τ(u) ← 1;
        next ← 2;                     { next position in layout }
        for x ∈ V'_1 do τ(x) ← next; next ← next+1 od;
        left ← 2;                     { left end of V'_1 in layout }
        right ← next−1;               { right end of V'_1 in layout }
        for i ← 2 to n−1 do
                while left≤right do
                        x ← τ^{-1}(left);
                        for {x,y} ∈ E do
                                if τ(y) = 0 then τ(y)←next; next←next+1 fi
                        od
                        left ← left+1;
                od
                right ← next−1;       { right end of V'_i }
        od;
        return
end
```

Figure 6. Implementation details for *MLA2*

```
procedure MLA3(G=(V,E),u,τ)
        n ← |V|;
        make_mod_levels(G,u,V'_0, . . . , V'_{n-1});
        count_gc(G,u,V'_0, . . . , V'_{n-1},ngc,1,1);
        sort(V'_1,ngc(x) < ngc(y));
        for x ∈ V do τ(x) ← 0 od;     { 0 denotes undefined }
        τ(u) ← 1;
        next ← 2;                { next position in layout }
        for x ∈ V'_1 do τ(x) ← next; next ← next+1 od;
        left ← 2;                { left end of V'_1 in layout }
        right ← next−1;          { right end of V'_1 in layout }
        for i ← 2 to n−1 do
                while left ≤ right do
                        x ← τ^{-1}(left);
                        for {x,y} ∈ E do
                                if τ(y) = 0 then τ(y) ← next; next ← next+1 fi
                        od
                        left ← left+1; { left end of V'_i }
                od;
                right ← next−1;      { right end of V'_i }
        od;
        return
end
```

Figure 7. Implementation details for *MLA*3

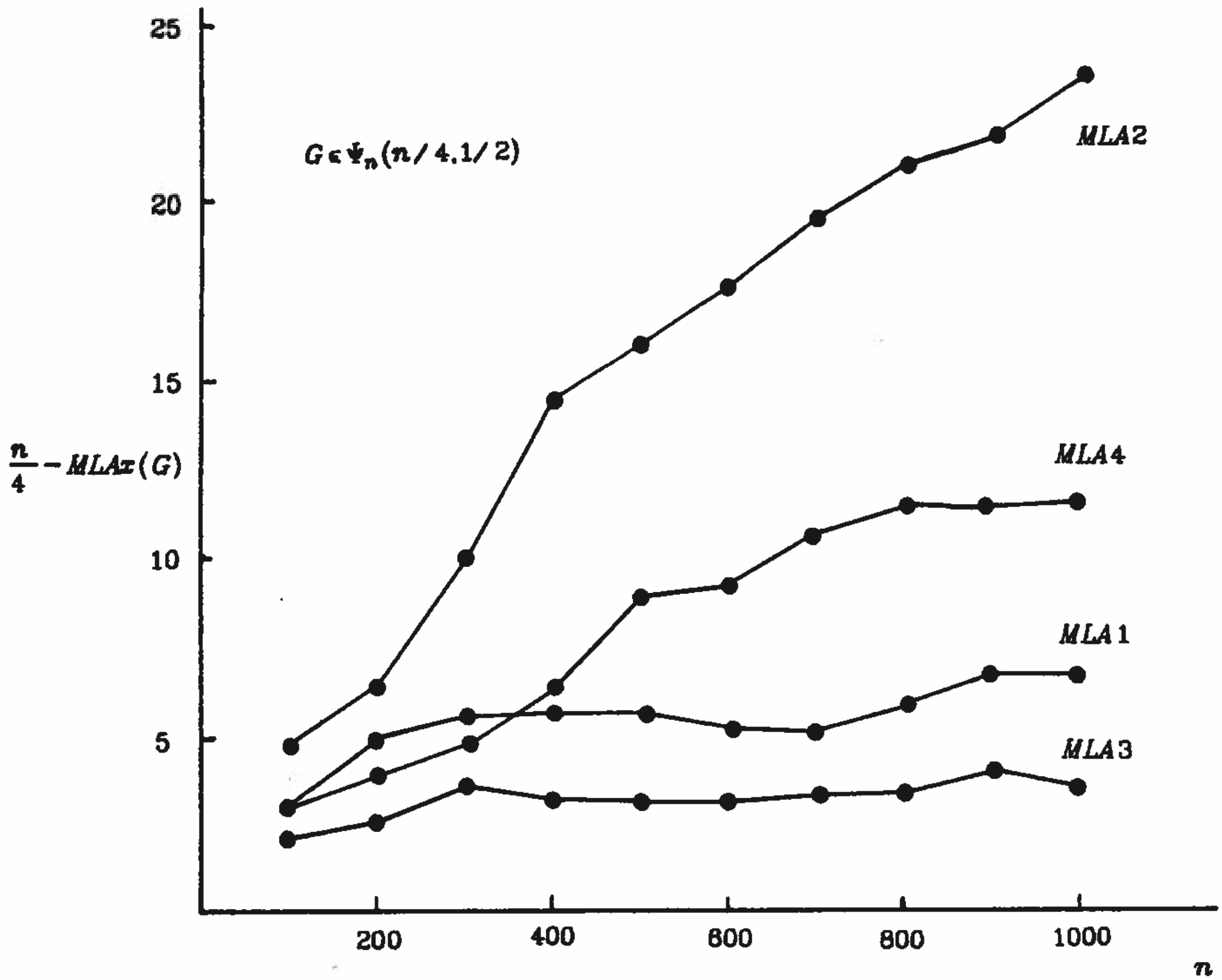



**Figure 8. Performance of Modified Level Algorithms**

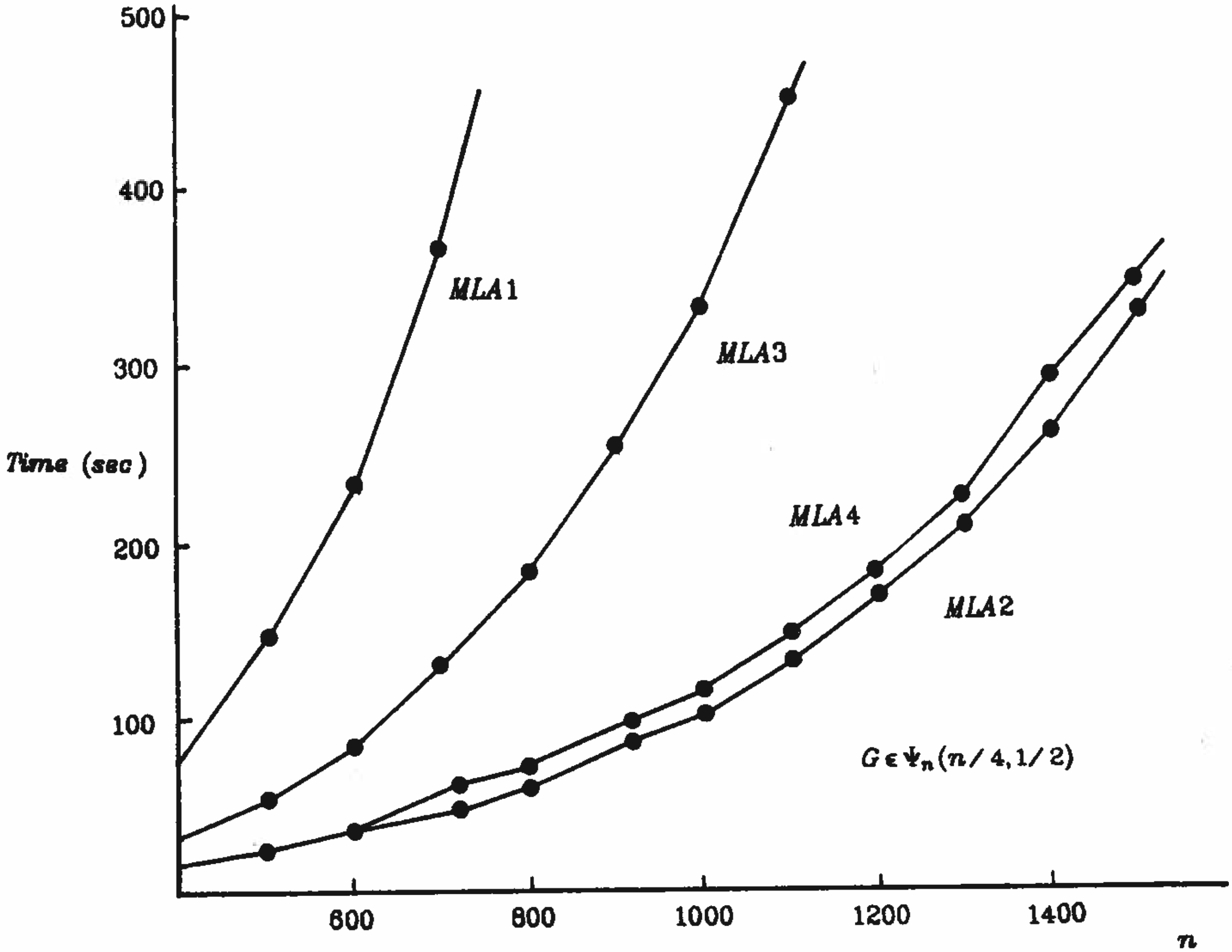



**Figure 9. Running Time of Modified Level Algorithms**

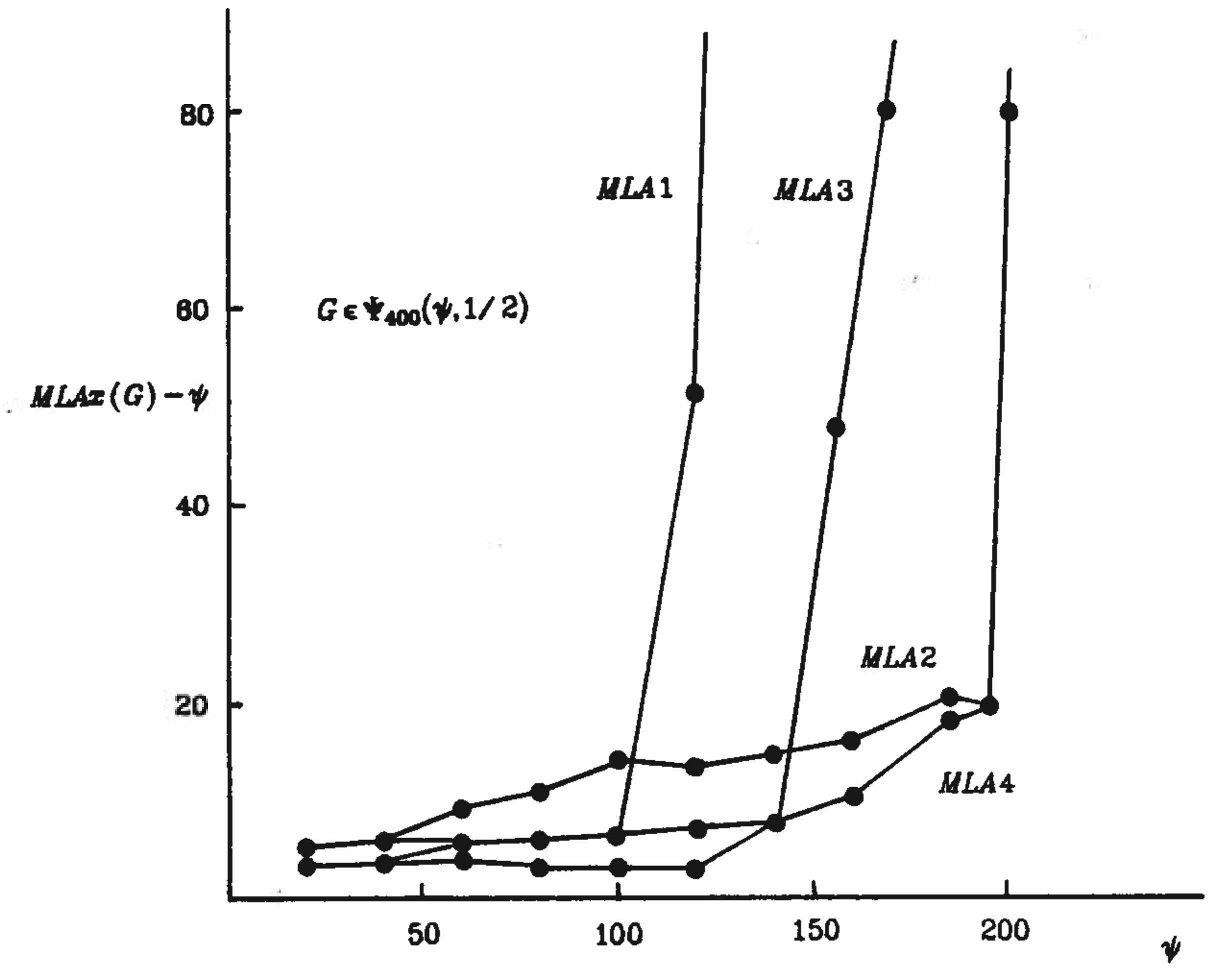
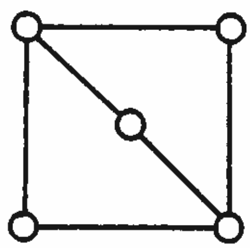



**Figure 10. Deterioration of Modified Level Algorithms as $\psi$ gets Large**



$\varphi(G) = 3 \qquad \omega^*(G) = 2$

**Figure 11. Graph Showing $\omega^*(G) \neq \varphi(G)$**

| $k$ | $\hat{n}_k$ | $(np)^k$ |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 27.6 | 27.6 |
| 2 | 763 | 763 |
| 3 | 20,850 | 21,096 |
| 4 | 428,450 | 582,890 |
| 5 | 549,910 | - |

Figure 12. Comparison of $\hat{n}_k$ with $(np)^k$ for $n = 10^6$, $p = 2\frac{\ln n}{n}$