Algorithms for Graph Partitioning on the Planted Partition Model

Anne Condon[*]
condon@cs.wisc.edu
Computer Sciences Department
University of Wisconsin
1210 West Dayton St.
Madison, WI 53706

Richard M. Karp[†]
karp@cs.washington.edu
Department of Computer Science
and Engineering
University of Washington
Seattle, WA 98195

March 6, 2001

Abstract

The NP-hard graph bisection problem is to partition the nodes of an undirected graph into two equal-sized groups so as to minimize the number of edges that cross the partition. The more general graph l-partition problem is to partition the nodes of an undirected graph into l equal-sized groups so as to minimize the total number of edges that cross between groups.

We present a simple, linear-time algorithm for the graph l-partition problem and analyze it on a random "planted l-partition" model. In this model, the n nodes of a graph are partitioned into l groups, each of size n/l; two nodes in the same group are connected by an edge with some probability p, and two nodes in different groups are connected by an edge with some probability $r < p$. We show that if $p - r \geq n^{-1/2+\epsilon}$ for some constant ϵ, then the algorithm finds the optimal partition with probability $1 - \exp(-n^{\Theta(\epsilon)})$.

1 Introduction

The graph l-partition problem is to partition the n nodes of an undirected graph into l equal-sized groups so as to minimize the *cut size*, namely the total number of edges that cross between groups. There is an extensive literature on algorithms for this problem because of its many applications, which include VLSI circuit placement, parallel task scheduling, and sparse matrix factorization.

Unfortunately even the special case of this problem when $l = 2$, which is the well-known graph bisection problem, is NP-hard [10]. In light of this, much of the literature on algorithms for graph bisection (as for many other NP-hard problems) reports on average-case performance of algorithms. Such work assumes a probability distribution over the input graphs of a given size (number of nodes and/or edges). While the bulk of this work provides only empirical data, several papers, including this paper, bound (as a function of the parameters of the random graph model) the probability that

[*] Condon's research supported by NSF grants HRD-627241 and CCR-9257241.

[†] Karp's research supported by NSF grant DB1-9601046. Current address: Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720.

a given algorithm finds a minimum bisection. Such analyses are useful in that they can provide insight on when and why certain algorithmic approaches are likely to be effective.

A popular random graph model is the $G(n, m)$ model in which a graph is selected randomly and uniformly from the set of all graphs with n nodes and m edges. A closely related model is the $G(n, p)$ model in which each pair of nodes is connected by an edge independently with probability p. No polynomial time algorithm is known that provably finds the minimum bisection with high probability on either of these models for general p. The lack of such an analysis may stem from the fact that, if $m/n \to \infty$, then for almost all graphs with n nodes and m edges the cut sizes of the best and worst bisections differ by only a low order term (see Bui et al. [3]).

Instead, some researchers have worked with random graph models in which the cut size of the best bisection is much smaller than the average cut size. The earliest results, due to Bui et al. [2, 3] concerned the $G(n, m, b)$ model, in which a graph is chosen randomly and uniformly from the set of graphs that have n nodes, m edges, and minimum cut size b. Bui et al. describe an algorithm, based on network flow techniques, that with probability $1 - o(1)$ finds an optimal bisection on this model with the additional constraints that the graph is regular, say with degree d and $b = o(n^{1-1/\lfloor (d+1)/2 \rfloor})$. Since every graph with dn edges has average cut size $dn/2$, the minimum bisection for the Bui et al. graphs is asymptotically smaller than the average bisection. Dyer and Frieze [6] analyze an algorithm for (not necessarily regular) graphs with $\Omega(n^2)$ edges and $b \leq (1 - \epsilon)m/2$ for a fixed $\epsilon > 0$. The Dyer-Frieze algorithm is based on comparison of vertex degrees; it finds the minimum bisection in polynomial expected time. Boppana [5] presents a graph bisection algorithm based on eigenvector methods. He shows that if m is $\Omega(n \log n)$ and $b \leq (m - 5\sqrt{mn \log n})/2$, then his algorithm finds the minimum bisection with probability $1 - O(1/n)$. Thus, Boppana's analysis applies to a larger class of graphs than the analysis of either Bui et al. or Dyer and Frieze. However, the running time of Boppana's algorithm is high since the algorithm uses the ellipsoid method for finding the maximum of a concave function.

Jerrum and Sorkin [12] analyzed a constant-temperature simulated annealing algorithm for graph bisection on a slightly different random graph model. Simulated annealing is a heuristic, originally proposed by Kirkpatrick, Gelatt, and Vecchi [16], that can be applied to a wide range of combinatorial problems. Roughly, in the case of the graph bisection problem this algorithm attempts to find a good bisection from an initial bisection by repeatedly employing the following procedure (see Johnson et al. [13]). A pair of nodes, one on each side of the current bisection, is chosen. These are swapped with some probability that depends on two parameters, namely the change in cut size that results if the nodes are swapped and a temperature parameter which decreases over time (as the temperature parameter is decreased, so does the probability of a "bad" swap). Jerrum and Sorkin point out that, although the temperature parameter is believed to improve the effectiveness of simulated annealing algorithms, there is no rigorous theoretical analysis that supports this belief. Indeed, Jerrum and Sorkin's results show that, on their random graph model, with high probability, constant temperature simulated annealing (also known as the Metropolis algorithm) succeeds in finding the minimum bisection in polynomial time.

In the graph model studied by Jerrum and Sorkin, known as the planted bisection model, a random graph with an even number of nodes n is constructed as follows. $n/2$ of the nodes of the graph are assigned one color and the remaining nodes are assigned a different color. The probability of an edge between like-colored nodes is p and the probability of an edge between differently-colored nodes is $r < p$. (The planted bisection model is roughly equivalent to the $G(n, m, b)$ model with $b = rn^2/4$ and $m = (p + r)n^2/4$.) If $p - r > n^{-1/2+\epsilon}$, for any fixed $\epsilon > 0$, then with probability

$1 - \exp(-n^{\Omega(\epsilon)})$ the planted bisection is the unique bisection with minimum cut size (see [3, 12]). Boppana's analysis of his eigenvector algorithm applies also to the planted bisection model, with $p - r = \Omega(\log n/n)$. Jerrum and Sorkin show that there is a choice of the temperature parameter for which, in $\Theta(n^2)$ iterations of the node swap procedure, the Metropolis algorithm finds the minimum bisection with probability $1 - \exp(-n^{\Omega(\epsilon)})$ if $p - r \geq n^{-1/6+\epsilon}$.

The analysis of Jerrum and Sorkin centers on the evolution of the *maximum imbalance* of a color in a bisection (L, R), where the imbalance of a color is the difference between the number of nodes of that color in L and the number in R, all divided by 2. (We note that where we use "maximum imbalance" in this paper, Jerrum and Sorkin use "imbalance.") Intuitively, the pair-swapping process of the Metropolis algorithm has a tendency to improve the maximum color imbalance. The analysis shows that this process leads fairly quickly to a large imbalance of $\Theta(n)$. However, most of the running time of the Metropolis algorithm is then spent when the current bisection is very close to the minimum bisection. For example, when just one node of each color is on the "wrong" side of the partition, it takes $\Theta(n^2)$ time just to select that pair of nodes for a potential swap.

Jerrum and Sorkin point out that this wasteful use of swaps can be avoided by slightly modifying the choice of node-pairs that are candidates for swaps. Another way of circumventing this problem is as follows. First, run the Metropolis algorithm on a randomly chosen set of half of the nodes until $\Theta(n)$ imbalance is reached (with high probability). Now, the partition of nodes, while not completely correct, is still statistically accurate in the sense that an overwhelming fraction of one side is of one color and an overwhelming fraction of the other side is of opposite color. This statistically accurate partition can be used to produce an exact partition of the remaining half of the nodes as follows. For each node in the remaining half, the expected number of edges that it has to the side of the partition containing nodes that are predominantly of the same color is much higher than to the other side of the partition. Moreover, by the concentration of measure phenomenon, the actual number is also sharply concentrated around this value. Hence, the statistically accurate partition provides a highly reliable guide to produce an exactly monochromatic partition of the remaining nodes. In turn, this exact partition can be used as a highly reliable guide to monochromatically partition the first half of the nodes (i.e. partition them so that all nodes in one class have the same color).

In this paper, we use a different algorithm to produce a statistically accurate partition, based on successive augmentation. Starting with an empty partition, our algorithm repeatedly picks a pair of unexamined nodes and places them, one on each side of the partition in a greedy fashion so as to minimize the number of edges added to the cut. Once half of the nodes are added in this way, the partition produced has high imbalance and is statistically correct.

The same idea extends to a multi-partition model. In this planted l-partition model, each node is assigned one of l colors, with n/l nodes of each color, and the probability of an edge between nodes is just as for the planted bisection model. On this model, the algorithm first selects a sequence of pairs of nodes and creates a two-way partition following the greedy procedure outlined in the last paragraph. Each greedy step tends to increase the difference between the imbalances of *any* two colors. Thus, the resulting two-way partition is again a statistical guide because the imbalance between any two colors is $\Theta(n)$. The expected value of the number of edges from a remaining vertex of a color C is a value o_C which is well separated from the corresponding value $o_{C'}$ for a different color C'. Moreover, by the concentration of measure phenomenon, the actual value is sharply concentrated around this expected value. This provides a highly reliable method to monochromatically l-way partition the remaining vertices: Order the number of edges of the remaining nodes to the left side as $l_1 \leq l_2 \leq \ldots$, and form groups by cutting off at the largest differences. This can now be used, in turn, to monochromatically partition the first half of the nodes.

In Section 3, we present our linear-time algorithm for the l-partition problem, and in Section 4 we show that this algorithm correctly constructs a monochromatic partition of a graph with a "planted" l-partition, with high probability. Specifically, we show that, for the planted l-partition model with $p - r \geq n^{-1/2+\epsilon}$, our algorithm outputs the minimum partition with probability $1 - \exp(-n^{\Omega(\epsilon)})$. We note that our analysis holds for the widest possible range of $p - r$, unlike the analysis of Jerrum and Sorkin.

In Section 5.3 we prove a stronger property of our algorithm: $\Theta(n)$ iterations produce a partition in which the difference between *any* pair of imbalances is $\Theta(n)$. Using this insight, we obtain a somewhat simpler algorithm for the l-partition problem in Section 5.

2 Related Work

Perhaps the best-known algorithm for the graph bisection problem is the Kernighan-Lin (K-L) heuristic [15] and its modification by Fiduccia and Mattheyses [8]. Several other algorithms for graph bisection and graph l-partitioning have been proposed, including genetic algorithms [4].

Johnson et al. [13] experimentally compared the performance of the K-L and simulated annealing algorithms on several random graph models. Overall, simulated annealing was found to be superior to K-L on the $G(n, p)$ graphs tested (the parameters were chosen so that in effect $p = O(1/n)$), while K-L was found to be superior on random geometric graphs. The simulated annealing algorithm implemented by Johnson et al. is somewhat different from that analyzed by Jerrum and Sorkin, in that, rather than repeatedly selecting a pair of nodes for swapping, the algorithm selects a *single* node. The cost (and hence the probability) of swapping a selected node is a function not only of the resulting change in cut size, but also includes a penalty if the difference in size between the left and right sides of the partition increases. Interestingly, Johnson et al. mention that algorithms based on successive augmentation, in which an initially empty structure is successively augmented until it becomes a solution, often outclass algorithms based on simulated annealing, but they do not describe any experimental results on successive augmentation for graph bisection in their paper. The early phases of our linear time algorithm can be considered to be successive augmentation.

Also closely related is the work of Juels [14], which analyzes a simple hill-climbing algorithm on the planted bisection model. Starting from a random initial bisection, this algorithm repeatedly selects at random a pair of nodes, one from each side of the partition, and swaps them if and only if the cut size decreases as a result. Juels shows that within $\Theta(n^2)$ iterations, this algorithm succeeds in finding the minimum bisection with probability $\Omega(1)$ if $p - r = \Omega(1)$. (Here and in what follows, the notation $\Omega(1)$ means some constant > 0 that is independent of the graph.)

Other algorithms for graph bisection are related to the algorithm of this paper in that they construct a "core" and build the rest of the solution around this core. The algorithm of Dyer and Frieze [6] mentioned above has a "core" consisting of the neighbors of the maximum-degree vertex of the graph; the remaining vertices are partitioned based on the number of neighbors they have in the core. Another example is Kucera's algorithm for graph partitioning [17]. Our algorithm differs from these in that the core is imperfectly partitioned.

Regarding approximation algorithms for the graph bisection problems on general graphs, no polynomial-time algorithm is known that is guaranteed to output a bisection with cut size that is bounded by a constant times the minimum cut size. For dense graphs, i.e. graphs in which the minimum node degree is $\Omega(n)$, two polynomial time approximation schemes (PTAS) for the graph bisection problem were recently proposed [1, 9]: given a graph and a constant $\epsilon > 0$, these algorithms output a bisection with cut size at most $(1+\epsilon)$ times the minimum cut size. (The running time of these algorithms is exponential in $1/\epsilon$.)

3 Algorithm

In this section we present our linear-time algorithm for the graph l-partition problem. The algorithm consists of four phases. Briefly, the purpose of the first two phases is to build up a partition (L_2, R_2) with $|L_2| = |R_2| = \Theta(n)$ in which some color has an imbalance of $\Theta(n)$. By the *imbalance* of a color in a partition (L, R), we mean the number of nodes of that color in L less the number of nodes of that color in R, all divided by 2. We note that the imbalance may be negative. In the third phase, partition (L_2, R_2) is used to partition the remaining unexamined nodes into two non-empty groups L and R such that no node in L is the same color as a node in R. In the fourth phase, all nodes examined in phases 1 and 2 are added to the "correct" side of the partition (L, R). The problem can now be solved recursively on L and R. We next describe the algorithm in detail.

Algorithm 1

Input: n-node graph G and integer $l > 1$ (the partitioning factor) (assume n is sufficiently large so that $\lceil n^{1-\epsilon/2}\rceil + \lceil n/4\rceil < n/2$):

Phase 1: L_1 and R_1 are initially empty. In each of $n_1 = \lceil n^{1-\epsilon/2}\rceil$ steps, choose a pair of nodes $(1,2)$ randomly and uniformly from the unexamined nodes. Let $l_1(i)$ and $r_1(i)$ be the number of edges from node $i, i \in \{1,2\}$ to nodes in L_1 and R_1 respectively, and let $X = l_1(1) - r_1(1) - l_1(2) + r_1(2)$. If $X > 0$, place nodes 1 and 2 in L_1 and R_1 respectively and if $X < 0$, place nodes 2 and 1 in L_1 and R_1 respectively. If $X = 0$ then place the nodes equiprobably into either L_1 or R_1.

Phase 2: L_2 and R_2 are initially empty. Let $n_2 = \lceil n/4\rceil$. Choose n_2 new pairs of nodes randomly and uniformly from the unexamined nodes. As in phase 1, greedily assign one node from each pair $(1,2)$ to each of L_2 and R_2, depending on the sign of $X = l_1(1) - r_1(1) - l_1(2) + r_1(2)$. Note that all pairs may be assigned concurrently to (L_2, R_2).

Phase 3: For each remaining unexamined node v, let $l_2(v)$ denote the number of edges from node v to nodes in L_2. Let $o_0 < o_1 < \ldots < o_j$ be the ordered set of values $l_2(v)$ and let $o_a - o_{a-1}$ be the maximum difference between consecutive numbers in this ordered list. If $l_2(v) \geq o_a$, put node v in L and if $l_2(v) < o_a$ put node v in R.

Phase 4: In parallel for each node v examined in phases 1 and 2, assign v greedily to L if the fraction of nodes in L that have edges to v is greater than the fraction of nodes in R that have edges to v, and assign v to R otherwise.

Recursion: Let k_L and k_R be such that $|L| = k_L n/l$ and $|R| = k_R n/l$. If $k_L = 1$ then include L as one of the classes in the output l-partition. Otherwise, if k_L is an integer greater than 1 then apply the algorithm recursively to the graph induced by the nodes of L, with partitioning factor k_L and include the k_L resulting classes among the l-partitions. Similarly, if $k_R = 1$ then include R as one of the classes in the output l-partition. Otherwise, if k_R is an integer greater than 1 then apply the algorithm recursively to the graph induced by the nodes of R, with partitioning factor k_R and include the k_R resulting classes among the l-partitions. Finally, if either k_L or k_R is not an integer, declare the algorithm to have failed.

4 Analysis

In this analysis, we assume that $p - r = \Delta = n^{-1/2+\epsilon}$. The analysis proceeds by showing that the following facts are true with probability $1 - \exp(-n^{\Omega(\epsilon)})$, referred to as "high probability" throughout. At the end of phase 1, some color in (L_1, R_1) has an imbalance $\geq n^{1-\epsilon}$. At the end of phase 2, some color in (L_2, R_2) has imbalance $\Theta(n)$. At the end of phase 3, no node in L is the same color as a node in R and both L and R are of size $\Theta(n)$. Finally, at the end of phase 4, by which time all nodes are assigned either to L or to R, no node in L is the same color as a node in R. The underlying probability space for which these facts apply is the product of the random planted bisection graph model with the random choices made by the algorithm.

Sections 4.1 through 4.3 analyze successive phases of the algorithm. The following version of Azuma's Inequality, which is in a form due to McDiarmid [18] (see also [12]) is used throughout.

Theorem 1 (Azuma's Inequality) *Let $Z_1, \ldots, Z_n$ be independent random variables, with Z_k taking values in a set A_k for each k. Suppose that the (measurable) function $f : \prod A_k \to R$ satisfies $|f(x) - f(x')| \leq c_k$ whenever the vectors x and x' differ only in the kth coordinate. Let Y be the random variable $f(Z_1, \ldots Z_n)$. Then for any $t > 0$,*

$$\mathrm{Prob}[|Y - EY| \geq t] \leq 2 \exp\left(-2t^2 / \sum_{k=1}^{n} c_k^2\right).$$

4.1 Phase 1

In Theorem 5 we show that at the end of phase 1, some color in (L_1, R_1) has imbalance at least $n^{1-\epsilon}$ with high probability. The proof of the theorem analyzes how the maximum imbalance in (L_1, R_1) grows over time. The following claim is key to this analysis. It shows (part (ii)) that at every iteration of the greedy partition-building algorithm of phase 1, maximum imbalance is at least as likely to increase as to decrease. Moreover (part (iii)), the higher the maximum imbalance, the more likely it is to increase.

Claim 2 *Let $x (= x(t))$ be the maximum imbalance in partition (L_1, R_1) at time step t of phase 1. Then at step $t + 1$ of phase 1, for any execution of the algorithm up to step t,*

(i) $\mathrm{Prob}[x \textit{ increases}] = \Omega(1)$,

(ii) $\mathrm{Prob}[x \textit{ increases}] - \mathrm{Prob}[x \textit{ decreases}] \geq 0$, *and*

(iii) if $x = \Omega(n^{1/2-\epsilon/2})$ then

$$\mathrm{Prob}[x \textit{ increases}] - \mathrm{Prob}[x \textit{ decreases}] = \Omega(\min\{x\Delta/\sqrt{t}, 1\}).$$

Proof : Since at least one imbalance is at least 0, we have that $x \geq 0$. All three assertions are trivially satisfied when $x = 0$. Hence, consider the case $x > 0$. If there are at least two colors with imbalance x at the end of step t, then x cannot decrease at step $t+1$ and, since the number l of colors is constant, can increase with probability $\Omega(1)$ (namely if nodes in the chosen pair have distinct colors, both of which have imbalance x; in this case, one of the nodes is placed in L_1 and the imbalance increases as a result). Hence, in the rest of the proof we assume that there is exactly one color with imbalance x.

Let $[x, x']$ denote the event that the colors of nodes 1 and 2 chosen at step $t+1$ have imbalance x and x', respectively, where here x is as given in the statement of the Claim and x' is any possible value for the imbalance. In the event $[x, x']$, if $x \neq x'$ then x increases at step $t+1$ if and only if node 1 is placed in L_1. (We note that if $x = x' + 1/2$ then, if node 1 is placed in R_1, the maximum imbalance does not in fact decrease, but rather remains unchanged.) Let

$$X = l_1(1) - r_1(1) - l_1(2) + r_1(2)$$

where $l_1(1), r_1(1), l_1(2)$, and $r_1(2)$ are as defined in phase 1 of the algorithm at a step in which the colors of the pair of chosen nodes have imbalances x and x'. We have that

$$\text{Prob}[x \text{ increases}|\text{ event } [x, x']] = \text{Prob}[X > 0] + \frac{1}{2}\text{Prob}[X = 0]$$

and

$$\text{Prob}[x \text{ does not increase}|\text{ event } [x, x']] = \text{Prob}[X < 0] + \frac{1}{2}\text{Prob}[X = 0].$$

Therefore, to prove part (ii) of the claim it is sufficient to show that $\text{Prob}[X > 0] \geq \text{Prob}[X < 0]$. Averaging over events $[x, x']$ with x as in the statement of the lemma and all $x' \neq x$ then gives the result. Part (i) of the claim follows from this and the additional fact that at each step of phase 1, the probability of each event $[x, x']$ such that there are colors with imbalances x and x' is $\Omega(1)$. This fact is true because phase 1 ends in $o(n)$ steps and so at every step of phase 1, there are $\Omega(n)$ unexamined nodes of each color.

To analyze $\text{Prob}[X \geq 0]$ we use the fact that each term $l_1(i)$ and $r_1(i)$ is binomially distributed. Let $B(n, p)$ denote the number of successes in n independent Bernoulli trials, each with probability p of success. Let $b + x$ and $b' + x'$ be the number of nodes in L_1 which have the same color as nodes 1 and 2, respectively, at the end of step t. At step $t+1$ of the algorithm, $t \geq 0$,

$$l_1(1) - r_1(1) = B(b + x, p) + B(t - b - x, r) - B(b - x, p) - B(t - b + x, r),$$

which has expected value $2x\Delta$. Similarly,

$$l_1(2) - r_1(2) = B(b' + x', p) + B(t - b' - x', r) - B(b' - x', p) - B(t - b' + x', r).$$

Hence,

$$X = B(b+b'+x-x', p) + B(2t-b-b'-x+x', r) - B(b+b'-x+x', p) - B(2t-b-b'+x-x', r).$$

Since $x - x' > 0$, X dominates the random variable X' defined by

$$X' = B(\lceil b + b' \rceil, p) + B(2t - \lceil b - b' \rceil, r) - B(\lceil b + b' \rceil, p) - B(2t - \lceil b - b' \rceil, r).$$

To see why X' dominates X, note that the term $B(\lfloor x - x' \rfloor, p)$ in X is replaced by $B(\lfloor x - x' \rfloor, r)$ in X' and similarly, $-B(\lfloor x - x' \rfloor, r)$ in X is replaced by $B(\lfloor x - x' \rfloor, p)$ in X'.

Since the random variable X' is symmetric with expected value 0, we have

$$\text{Prob}[X > 0] \geq \text{Prob}[X' > 0] = \text{Prob}[X' < 0] \geq \text{Prob}[X < 0].$$

This completes the proof of parts (i) and (ii) of the claim.

To prove part (iii), define X as before except that X now pertains to a step of the algorithm in which, if 1 and 2 are the two nodes chosen at step $t+1$, then the color of node 1 has imbalance x and the color of node 2 has imbalance at most 0. Call the event in which the nodes have these imbalances $[x, \leq 0]$. At every step of phase 1, the probability that event $[x, \leq 0]$ occurs is $\Theta(1)$ because some color must have imbalance at most 0. Hence, to prove part (iii) of the claim, it is sufficient to show that

$$\text{Prob}[X > 0] = 1/2 + \Omega(\min\{\frac{x\Delta}{\sqrt{t}}, 1\}).$$

First note that since the color of node 2 has imbalance at most 0, the expected value of $l_1(2) - r_1(2)$ is at most 0. Hence EX, the expected value of X, is at least $2x\Delta$. To bound $\text{Prob}[X > 0]$ we use Esseen's Inequality, which provides an approximation for the tail probability of a sum of independent random variables in terms of the normal distribution.

Theorem 3 (Esseen's Inequality) *(Petrov [19, Theorem 3, p.111]) Let $X_1, \ldots, X_n$ be independent random variables such that $EX_j = 0$ and $E|X_j|^3 < \infty$, $j = 1, \ldots, n$. Let*

$$\sigma_j^2 = EX_j^2, \quad B = \sum_{j=1}^{n} \sigma_j^2, \quad F(x) = \text{Prob}[B^{-1/2} \sum_{j=1}^{n} X_j < x], \text{ and}$$

$$L = B^{-3/2} \sum_{j=1}^{n} E|X_j|^3.$$

Then

$$\sup_x |F(x) - \Phi(x)| \leq AL$$

where A is an absolute constant and $\Phi(x)$ denotes the normal $(0,1)$ distribution function.

Note that $X = \sum_{j=1}^{4t} X_j + EX$, where each random variable X_j is one of the following: (i) $Y - p$ where Y is 1 with probability p and 0 with probability $1 - p$, (ii) $Y - r$ where Y is 1 with probability r and 0 with probability $1 - r$, (iii) $Y + p$ where Y is -1 with probability p and 0 with probability $1 - p$, or (iv) $Y + r$ where Y is -1 with probability r and 0 with probability $1 - r$. Therefore $EX_j = 0$ and $E|X_j|^3 = O(1)$ and so the random variables X_j satisfy the conditions of Esseen's Inequality. Using the notation in Theorem 3 with $n = 4t$, we have that

$$\begin{aligned} \text{Prob}[X > 0] &= \text{Prob}[B^{-1/2} \sum_{j=1}^{4t} (-X_j) < B^{-1/2} EX] \\ &\geq \Phi(B^{-1/2} EX) - AL \\ &\geq \Phi(\frac{2x\Delta}{\sqrt{B}}) - AL \\ &= 1/2 + \Omega(\min\{x\Delta/\sqrt{B}, 1\}) - O(L). \end{aligned}$$

To see why the last equality holds, we note that $\Phi(\frac{2x\Delta}{\sqrt{B}})$ can be expressed as the sum of the following two areas under the curve defining the normal density function ϕ: (a) the area up to the zero axis and (b) the area between the zero axis and the axis at $\frac{2x\Delta}{\sqrt{B}}$. The contribution of (a) is simply $\Phi(0) = 1/2$. To estimate the contribution of (b), let α be any positive constant less than 1. Then $\phi(z) \geq \phi(\alpha) = \Theta(1)$, for all $z, 0 \leq z \leq \alpha$. Thus, if $\frac{2x\Delta}{\sqrt{B}} < \alpha$, then the area (b) is at least the length between the axes, namely $\frac{2x\Delta}{\sqrt{B}}$, times the value of the curve at its lowest point, namely $\phi(\frac{2x\Delta}{\sqrt{B}})$. But since $\frac{2x\Delta}{\sqrt{B}} < \alpha$, we have that $\phi(\frac{2x\Delta}{\sqrt{B}}) > \phi(\alpha) = \Theta(1)$ and so in this case the area (b) is $\Omega(\frac{2x\Delta}{\sqrt{B}})$. Also, if $\frac{2x\Delta}{\sqrt{B}} \geq \alpha$, then the area (b) is at least the area under ϕ between the zero axis and the axis at α, which in turn is at least $\alpha\phi(\alpha)$. Since α is a constant, $\alpha\phi(\alpha) = \Theta(1)$. Taking the min of the two cases $\frac{2x\Delta}{\sqrt{B}} < \alpha$ and $\frac{2x\Delta}{\sqrt{B}} \geq \alpha$, we conclude that the contribution of (b) to $\Phi(\frac{2x\Delta}{\sqrt{B}})$ is $\Omega(\min\{x\Delta/\sqrt{B}, 1\})$.

We claim that if $x = \Omega(n^{1/2-\epsilon/2})$, then $L = o(\min\{x\Delta/\sqrt{B}, 1\})$. To see this, note that each term $E|X_j|^3$ in L is either $p(1-p)^3 + (1-p)p^3 \leq p(1-p)$ or $r(1-r)^3 + (1-r)r^3 \leq r(1-r)$. Suppose that l_p of the terms $E|X_j|^3$ are $p(1-p)^3 + (1-p)p^3$ and that l_r of the terms are $r(1-r)^3 + (1-r)r^3$ (where $l_p + l_r = 4t$). Then

$$L \leq B^{-3/2}(l_p p(1-p) + l_r r(1-r)).$$

Using the same notation, we have that $B = l_p p(1-p) + l_r r(1-r)$. Therefore, $L \leq B^{-1/2}$. Also, since $x = \Omega(n^{1/2-\epsilon/2})$, we have that $x\Delta = \Omega(n^{1/2-\epsilon/2}n^{-1/2+\epsilon}) = \Omega(n^{\epsilon/2})$. Therefore, $L = o(x\Delta/B)^{1/2})$.

We next show that $L = o(1)$, and therefore $L = o(\min\{x\Delta/B)^{1/2}, 1\})$ as claimed. Since $p - r = \Delta$, either $p(1-p) \geq \Delta/2$ or $r(1-r) \geq \Delta/2$. At least x of the terms in B are of the form $p(1-p)$ and also at least x of the terms are of the form $r(1-r)$; that is, both l_p and l_r are at least x. Hence, $B = \Omega(x\Delta) = \Omega(n^{\epsilon/2})$ and so $L = O(n^{-\epsilon/4})$.

We have now shown that if $x = \Omega(n^{1/2-\epsilon/2})$ then

$$\text{Prob}[X > 0] = 1/2 + \Omega(\min\{\frac{x\Delta}{\sqrt{B}}, 1\}).$$

Part (iii) of the claim follows from the observation that $\sqrt{B} = O(\sqrt{t})$. □

The analysis of the evolution of imbalance x is somewhat complicated by the fact that steps in the random process $x(t)$ are not independent and the transition probabilities vary depending on the history of the algorithm. It is convenient to relate the behavior of x to a (simpler) random walk with identical independent increments. The next lemma does this. Throughout, when we refer to the probability that $x(t+1)$ takes some value, we mean that probability given the history of the algorithm up to step t.

Lemma 4 *Let $\epsilon : \mathbf{N} \rightarrow \mathbf{R}$ be a function (which may take both positive and negative values) such that*

$$\text{Prob}[x \textit{ decreases}] - \text{Prob}[x \textit{ increases}] \leq \epsilon(x(t)).$$

There exist positive constants c and d such that for all nonnegative integers a, b with $a \leq b$, the following holds. Let $Y(t), t = 0, 1, \ldots$ be a random walk with the following properties:

$$\begin{array}{ll} \text{Prob}[Y(t+1) = 1] = 1 & \textit{if } Y(t) = 0, \\ \text{Prob}[Y(t+1) = Y(t) - 1] = 1/2 + c\max_{j\in[a,b]}\{\epsilon(j)\} & \textit{if } Y(t) > 0, \textit{ and} \\ \text{Prob}[Y(t+1) = Y(t) + 1] = 1/2 - c\max_{j\in[a,b]}\{\epsilon(j)\} & \textit{if } Y(t) > 0. \end{array}$$

Then for any nonnegative integer i, $a \leq i \leq b$,

$$\text{Prob}[\textit{starting at } i/2, \ x \textit{ leaves } [a/2, b/2] \textit{ at the right end within } k \textit{ steps (and never leaves the left end)}]$$

$$\geq$$

$$\text{Prob}[\textit{starting at } i, \ Y \textit{ leaves } [a, b] \textit{ at the right end within } dk \textit{ steps (and never leaves the left end)}]] - \exp(-\Omega(k)).$$

Proof We relate Y to x in two stages. First, construct a random process Z from x, with x initially equal to $i/2$ at some point k_0 of phase 1 of the algorithm, by "removing the loop probabilities" from x. More precisely, the process Z is defined from x as follows. Set $Z(0) = i/2$. Run phase 1 from time k_0 until the first time that x changes, say at time $k_1 > k_0$. Note that $x(k_1) \in \{i - 1/2, i + 1/2\}$. Set $Z(1) = x(k_1)$. More generally, if $k_1 < k_2 < \ldots$ are the times after k_0 that x changes, then $Z(m) = x(k_m)$. Clearly, if x reaches $b/2$ at time k_m then Z does so at time m.

Now, let $d > 0$ be a constant and let M be the event that the number of times that x changes within the first k steps is at least dk. From Claim 2, at each step of the algorithm, the probability that x changes is $\Omega(1)$ and so the number of times that x changes dominates the number of successes in k independent Bernoulli trials with probability $\Omega(1)$ of success in each trial. Applying Azuma's inequality 1 , it follows that for sufficiently small $d > 0$, $\text{Prob}[M] = 1 - \exp(-\Omega(k))$. Therefore,

$$\begin{aligned} & \text{Prob[starting at } i/2, \ x \text{ leaves } [a/2, b/2] \text{ at the right end within } k \text{ steps]} \\ \geq \ & \text{Prob[starting at } i/2, \ x \text{ leaves } [a/2, b/2] \text{ at the right end within } k \text{ steps} \mid M]\text{Prob}[M] \\ \geq \ & \text{Prob[starting at } i/2, \ Z \text{ leaves } [a/2, b/2] \text{ at the right end within } dk \text{ steps]Prob}[M] \\ = \ & \text{Prob[starting at } i/2, \ Z \text{ leaves } [a/2, b/2] \text{ at the right end within } dk \text{ steps]}(1 - \exp(-\Omega(k))). \end{aligned}$$

Now let $c > 0$ be a constant such that for all $Z(m)$ in the range $[a/2, b/2]$, $\text{Prob}[Z(m+1) = Z(m) - 1/2] \leq 1/2 + c \max_{j \in [a,b]}\{\epsilon(j)\}$. The existence of such a constant c (which is independent of a, b) follows from the fact that the loop probabilities of x are $1 - \Omega(1)$ (Claim 2). Let Y be as in the statement of Lemma 4, with this constant c. Intuitively, since each step of Y is biased at least as much to the left as each step of Z, it is the case that

$$\text{Prob[starting at } i/2, \ Z \text{ leaves } [a/2, b/2] \text{ at the right end within } dk \text{ steps]}$$

$$\geq$$

$$\text{Prob[starting at } i, \ Y \text{ leaves } [a, b] \text{ at the right end within } dk \text{ steps]}.$$

This can be proved by showing that $2Z$ dominates Y. Our argument is essentially taken from Theorem 6.2 of Jerrum and Sorkin [12]). Base Z and Y on a common sample space, so that, given initially that $2Z(0) = Y(0)$, it is the case that $2Z(m) \leq Y(m)$ for all $m, 0 \leq m \leq dk$. The needed sample space consists of a sequence $\alpha(0), \alpha(1), \ldots, \alpha(dk)$ of independent real numbers, each chosen uniformly from the range $[0, 1]$. Fix m, and let $p_Z^+ = \text{Prob}[Z(m+1) = Z(m) + 1/2]$ and $p_z^- = \text{Prob}[Z(m+1) = Z(m) - 1/2]$. Define p_Y^+ and p_Y^- in the same way, with 1 replacing 1/2. Then the next state of the process Z is given by

$$Z(m+1) = \begin{cases} Z(m) - 1/2 & \text{if } \alpha(m) \leq p_Z^-, \\ Z(m) + 1/2 & \text{otherwise.} \end{cases}$$

The next state for the process Y is defined similarly, with 1 replacing $1/2$ and Y replacing Z. It can be shown by induction on m that $2Z(m) \geq Y(m)$ for all $m, 0 \leq m \leq dk$.

Finally, combining the relationship between x, Z, and Y yields the lemma. □

Theorem 5 *In partition (L_1, R_1) at the end of phase 1, some color has imbalance at least $n^{1-\epsilon}$ with high probability.*

Proof : We partition phase 1 into subphases, based on the value of the maximum imbalance x. The first subphase starts at time 0 and continues until $x \geq n^{1/2-\epsilon/2}/2$. By Claim 2, at every step of this subphase, the probability that x increases is at least the probability that x decreases. If Y is the random walk of Lemma 4 with $\epsilon() = 0$ and n_1 is the number of steps of phase 1, then

$$\text{Prob[starting at 0, } x \text{ reaches } n^{1/2-\epsilon/2}/2 \text{ within } n_1/2 \text{ steps]}$$

$$\geq$$

$$\text{Prob[starting at 0, } Y \text{ reaches } n^{1/2-\epsilon/2} \text{ within } dn_1 \text{ steps]} - \exp(-n^{\Omega(n_1)}),$$

where $d > 0$ is a constant. From Feller [7, XIV.3], the expected time for the unbiased random walk Y to reach $n^{1/2-\epsilon/2}$, starting at 0, is $n^{1-\epsilon}$. By Markov's inequality, with probability at least $1/2$ Y reaches $n^{1/2-\epsilon/2}$ within time $O(n^{1-\epsilon})$; moreover this holds regardless of where the walk starts within the interval. In $dn_1 = d\lceil n^{1-\epsilon/2}\rceil$ steps, the number of periods of length $O(n^{1-\epsilon})$ is $\Omega(n^{\epsilon/2})$. Therefore, the probability that Y reaches $n^{1/2-\epsilon/2}$ within dn_1 steps is $1 - \exp(-\Omega(n^{\epsilon/2}))$.

The tth subphase starts when the $(t-1)$st subphase ends. If $i/2 > 0$ is the value of x at the start of a subphase, then that subphase ends when $x = i$ or when $x = \lfloor i/2 \rfloor/2$ (or when Phase 1 ends).

Let Y be a random walk with no loop probabilities in which the difference between the probability of an increase and a decrease is $\delta = \Omega(\min\{i\Delta/\sqrt{t}, 1\}) = \Omega(\min\{i\Delta/\sqrt{n_1}, 1\})$. By Lemma 4 and Claim 2,

$$\text{Prob[starting at } i/2, \ x \text{ leaves } [\lfloor i/2 \rfloor/2, i] \text{ at the right end within } n^{1-3\epsilon/4} \text{ steps]}$$

$$\geq$$

$$\text{Prob[starting at } i, \ Y \text{ leaves } [\lfloor i/2 \rfloor, 2i] \text{ at the right end within } dn^{1-3\epsilon/4} \text{ steps]} - \exp(-\Omega(n^{1-3\epsilon/4})),$$

for some sufficiently small constant d (independent of i).

We first bound Prob[starting at i, Y leaves $[\lfloor i/2 \rfloor, 2i]$ at the right end]. Let s be the ratio of the probability of a decrease over the probability of an increase, that is, $s = \frac{1/2-\delta}{1/2+\delta} = 1 - \Theta(\delta)$. Using Feller [7, XIV.2.4], the probability that, starting at i, Y reaches $2i$ before $\lfloor i/2 \rfloor$ is at least

$$1 - \frac{s^{\lceil i/2 \rceil} - s^{2i - \lfloor i/2 \rfloor}}{1 - s^{2i - \lfloor i/2 \rfloor}} \geq 1 - \frac{s^{\lceil i/2 \rceil}}{1 - s^{2i - \lfloor i/2 \rfloor}} \geq 1 - \frac{s^{\lceil i/2 \rceil}}{1 - s^{3\lceil i/2 \rceil}}.$$

If $i = \Omega(n^{1/2-\epsilon/2})$ then

$$s^{\lceil i/2 \rceil} = (1 - \Omega(\min\{i\Delta/\sqrt{n_1}, 1\}))^{\lceil i/2 \rceil} = \exp(-\Omega(n^{1-\epsilon}\Delta/\sqrt{n_1})) = \exp(-\Omega(n^{\epsilon/4})).$$

Hence as long as $i = \Omega(n^{1/2-\epsilon/2})$, the probability that, starting at i, Y reaches $2i$ before $\lfloor i/2 \rfloor$ is $1 - \exp(-\Omega(n^{\epsilon/4}))$.

We next show that, starting at $i = \Omega(n^{1/2-\epsilon/2})$, Y leaves the interval $[\lfloor i/2 \rfloor, 2i]$ within $dn^{1-3\epsilon/4}$ steps with high probability, where $d > 0$ is a constant. From Feller [7, XIV.3], the expected time for this event is $O(i/\delta) = O(\max\{i, \sqrt{n_1}/\Delta\} = O(n^{1-\epsilon})$. Moreover, this bound holds regardless of the starting position within the interval $[\lfloor i/2 \rfloor, 2i]$. Again applying Markov's inequality, with probability at least 1/2, Y leaves $[\lfloor i/2 \rfloor, 2i]$ within time $O(n^{1-\epsilon})$ and so with high probability Y leaves $[\lfloor i/2 \rfloor, 2i]$ within time $dn^{1-3\epsilon/4}$.

Therefore, for $i = \Omega(n^{1/2-\epsilon/2})$, with high probability, starting at $i/2$, x leaves $[\lfloor i/2 \rfloor/2, i]$ at the right end within $n^{1-3\epsilon/4}$ steps. It follows that with high probability, within $n_1 = \lceil n^{1-\epsilon/2} \rceil$ steps, sufficiently many subphases of phase 1 are completed, all ending by leaving the corresponding interval at the right end so that the imbalance is at least $2n^{1-\epsilon}$. A similar analysis shows that once the imbalance is $2n^{1-\epsilon}$, then with high probability it remains at least $n^{1-\epsilon}$ for the rest of phase 1. This completes the proof of the theorem. □

4.2 Phase 2

Let C be a color of greatest imbalance in (L_1, R_1). Let y be the imbalance of color C in (L_2, R_2) at the end of phase 2. Each pair of nodes $(1,2)$ examined in phase 2 independently either contributes 1/2, 0 or $-1/2$ to y. We need the following results.

Lemma 6 *For any color C and all $k = n - \Theta(n)$, for any $\delta > 0$, the number of nodes of color C that have* not *been examined after k steps of phases 1 and 2 of the algorithm is $(n-2k)/l \pm O(n^{1/2+\delta})$ with probability $1 - \exp(-\Omega(n^{2\delta}))$.*

Lemma 6 follows from a straightforward application of the Azuma's inequality (Theorem 1).

Claim 7 *Suppose that the maximum imbalance x at the end of the first phase is at least $n^{1-\epsilon}$. Let $(1,2)$ be a pair of nodes examined in phase 2. Then*

$$\text{Prob}[(1,2) \textit{ contributes positively to } y] - \text{Prob}[(1,2) \textit{ contributes negatively to } y] = \Omega(1).$$

Proof Note that $(1,2)$ contributes positively to y if and only if the pair $(1,2)$ is such that exactly one node in the pair has color C and this node is placed in L. Also, $(1,2)$ contributes negatively to y if and only if exactly one node in the pair has color C and this node is placed in R.

A similar argument to that of Claim 2 part (ii) shows that if exactly one node in the pair $(1,2)$ has color C, then for any fixed value of the other node,

$$\text{Prob}[(1,2) \text{ contributes positively to } y] - \text{Prob}[(1,2) \text{ contributes negatively to } y] \geq 0.$$

Also, a similar argument to that of Claim 2 part (iii) shows that if exactly one node in the pair $(1,2)$ has color C, and the other has a color with imbalance at most 0, then

$$\text{Prob}[(1,2) \text{ contributes positively to } y] - \text{Prob}[(1,2) \text{ contributes negatively to } y]$$

$$= \Omega(\min\{\frac{x\Delta}{\sqrt{n_1}}, 1\}) = \Omega(1),$$

where the last equality follows from the fact that $n_1 = \lceil n^{1-\epsilon/2} \rceil$ and $x\Delta \geq n^{1-\epsilon} n^{-1/2+\epsilon}$.

Finally, from Lemma 6 we have that the probability that pair $(1,2)$ is such that one node has color C and the other has a color with imbalance at most 0 is $\Omega(1)$. The claim follows. □

Theorem 8 *At the end of phase 2, with high probability the imbalance of some color is $\Theta(n)$.*

Proof Let C be a color of greatest imbalance x in (L_1, R_1). Let y be the imbalance of color C in (L_2, R_2) at the end of phase 2, given that $x \geq n^{1-\epsilon}$. Let Z_i be the contribution of the ith pair of nodes to y (we assume that (L_1, R_1) is fixed for the definition of all Z_i). By Claim 7, y dominates $\sum_{i=1}^{n_2} Z_i$, where the Z_i are independent random variables taking values in the set $\{-1/2, 0, 1/2\}$, with $\text{Prob}[Z_i = 1/2] - \text{Prob}[Z_i = -1/2] = \Omega(1)$.

The random variables Z_i satisfy the conditions of Azuma's inequality (Theorem 1) with $c_k = 1$, $1 \leq i \leq n_2$. Also $Ey = \Omega(n_2) = \Omega(n)$. Therefore the probability that y is at least half of its expected value is at least $1 - \exp(-\Omega(n))$.

Hence, the probability that $y = \Omega(n)$, given that $x \geq n^{1-\epsilon}$, is $1 - \exp(-\Omega(n))$. By Theorem 5, $x \geq n^{1-\epsilon}$ with high probability. Hence, $y = \Omega(n)$ with high probability. □

4.3 Phases 3 and 4

For each node v that is examined in phase 3, let $l_2(v)$ be the number of edges from v to a node in L_2. For each color C, let $EL_2(C)$ be the expected number of edges of an unexamined node of color C to nodes in the set L_2. First, we will show that with high probability the values $l_2(v)$ are distributed as follows: for all nodes v of color C, the values $l_2(v)$ are clustered in a short interval centered at $EL_2(C)$. More precisely, in Claim 9 it is shown that with high probability, $|l_2(v) - EL_2(C)| \leq n^{1/2+\epsilon/2}$. Second, the interval spanned by the values $EL_2(C)$ is relatively large, namely of length $\Omega(n^{1/2+\epsilon})$. This is shown in Claim 11. Simple algebra then shows (Theorem 12) that two adjacent "clusters" must be far apart, implying that the quantity $o_a - o_{a-1}$ used as the partitioning criterion in phase 3 is large.

Claim 9 *Let v be a node of color C. Then with high probability*

$$|l_2(v) - EL_2(C)| \leq n^{1/2+\epsilon/2}.$$

Proof The random variable $l_2(v)$ is the sum of n_2 independent random variables that satisfy the conditions of Azuma's inequality (Theorem 1) with $c_k = 1$, $1 \leq i \leq n_2$. The result follows by a simple application of this inequality. □

The following claim is useful in the proof of Claim 11.

Claim 10 *For any color C, the number of nodes of color C that are examined in phase 2 is $2n_2/l \pm O(n^{1/2+\epsilon/2})$ with high probability.*

Proof Let X_k be the number of unexamined nodes of color C after $k \leq n/2$ nodes have been examined. From Lemma 6, with high probability, $X_{2n_1} = (n - 2n_1)/l \pm O(n^{1/2+\epsilon/2})$ and $X_{2(n_1+n_2)} = (n - 2(n_1 + n_2))/l \pm O(n^{1/2+\epsilon/2})$. $X_{2(n_1+n_2)}$ is the number of nodes of color C that are not examined in phases 1 or 2. Hence, the number of nodes of color C that *are* examined in phase 2 is $X_{2n_1} - X_{2(n_1+n_2)} = 2n_2/l \pm O(n^{1/2+\epsilon/2})$, as required. □

Claim 11 *Let C_{max} and C_{min} be the colors with the largest and smallest number of nodes, respectively, in L_2. With high probability,*

$$EL_2(C_{max}) - EL_2(C_{min}) = \Omega(n^{1/2+\epsilon}).$$

Proof Let c_{max} and c_{min} be the number of nodes of colors C_{max} and C_{min}, respectively, in L_2. From Theorem 8, with high probability, some color has imbalance $\Theta(n)$ in the partition (L_2, R_2). This, together with Claim 10, implies that with high probability some color has $n_2/l + \Theta(n)$ nodes in L_2, in which case $c_{max} = n_2/l + \Theta(n)$. Therefore for some constant c,

$$EL_2(C_{max}) \geq p(n_2/l + cn) + r((l-1)n_2/l - cn).$$

Also, since not all colors can have more than the average number n_2/l of nodes in L_2, it must be that $c_{min} \leq n_2/l$. Therefore,

$$EL_2(C_{min}) \leq pn_2/l + r(l-1)n_2/l.$$

Taking the difference of these two inequalities, we have that

$$EL_2(C_{max}) - EL_2(C_{min}) \geq (p-r)cn = \Omega(n^{1/2+\epsilon}).$$

□

Theorem 12 *At the end of phase 3, with high probability no node in L is the same color as a node in R and moreover, both L and R are non-empty.*

Proof Let $o_0 < o_1 < \ldots < o_j$ be the ordered set of values $l_2(v)$ over the nodes v examined in phase 3 and let $o_a - o_{a-1}$ be the maximum difference between consecutive numbers in this ordered list. From Claim 9, the values $l_2(v)$ are clustered in l intervals of length $n^{1/2+\epsilon/2}$ around the mean values $EL_2(A)$ with high probability. From Claim 11, the difference between the smallest and largest means is $\Omega(n^{1/2+\epsilon})$ with high probability. Hence with high probability there must be a distance of $\Omega(n^{1/2+\epsilon}/l)$ between some two consecutive clusters.

It follows easily that with high probability, $o_a - o_{a-1} = \Omega(n^{1/2+\epsilon}/l)$, and that for any two nodes v_1 and v_2 of the same color, either both $l_2(v_1)$ and $l_2(v_2)$ are greater than or equal to o_a or both are less than or equal to o_{a-1}. Thus, no node in L is the same color as a node in R.

The fact that L and R are both nonempty follows immediately from the fact that for some pair of nodes v_1 and v_2, $l_2(v_1) = o_a$ and $l_2(v_2) = o_{a-1}$; therefore v_1 and v_2 are placed on opposite sides of the partition (L, R). □

The analysis of phase 4 is very similar to that of phase 3. Although it may appear that independence is lost due to the fact that edges which were "used" in phase 3 (between nodes in (L_2, R_2) and nodes in L) are now "re-used" in phase 4, correctness follows from the following observation. Let $N(3)$ be the set of nodes examined in phase 3. Let (A, B) be some partition of these nodes such that like-colored nodes lie on the same side of the partition and neither A nor B is empty. Note that, given $N(3)$, there are only a finite number of such partitions (A, B) since the number of colors is finite. Therefore, it is sufficient to show that for each of these finitely many possible (A, B), if $(L, R) = (A, B)$ is used to partition the nodes in phase 4 then in phase 4, with high probability all nodes are placed in the correct side of the partition. This statement is independent of the outcomes of the individual steps of phases 1, 2, and 3 and so the analysis can proceed as in phase 3.

We have now shown that, with high probability, all phases 1 through 4 have the properties stated in the first paragraph of Section 4. Moreover, the total number of possible distinct recursive calls is constant (it is bounded by $2^l - 2$, namely the number of ways to choose a subset of the l colors, other than the empty set or the set of all l colors). Therefore, high probability correctness of the whole algorithm, including recursive calls, follows.

5 A Non-Recursive Algorithm

5.1 Motivation

In simulations of phase 1 of Algorithm 1, the maximum imbalance tended to increase over time, as we expected. Moreover, as remarked in the introduction, based on our experiments we hypothesize the following. With two colors, the partition evolves towards one in which, if there are k nodes on each side of the partition, then the imbalances are $\approx k/4$ and $-k/4$. With three colors the partition evolves towards one in which the imbalances are $\approx 2k/9, 0$, and $-2k/9$. More generally, with l colors, the partition evolves towards one with maximum imbalance $\approx (l-1)k/l^2$ and a gap of $2k/l^2$ between successive imbalances.

The following table presents some evidence that this indeed is the case, based on experiments on our algorithm. The values listed in the third row of the table are the average sample imbalances in our experiments, divided by k. The numbers presented are averaged over 20 runs of phase 1 of our algorithm with $n = 256,000$, $p = 1/2$, $\Delta = n^{-1/2+.2} = .0829$, and $k = 100,000$. The values listed in the second row of the table are the numbers towards which we believe the expected imbalances evolve in the limit. In each case, the variance is that for the maximum imbalance.

No. colors	2	3	4
Hypothesis	.250,-.250	.222,.000,-.222	.1875, .0625, -.0625, -.1875
Average sample imbalances	.248,-.248	.215,.000,-.215	.1753, .0619, -.0606, -.1766
Sample Variance:	0.000004	0.000011	0.000041

A heuristic explanation of this hypothesis is as follows. First, in the case of two colors, consider the evolution of phase 1 once the maximum imbalance is large. Let C be the color with maximum imbalance. Roughly, in 1/2 of the steps, exactly one of the chosen pair of nodes has color C and this is likely to be put in L. In 1/4 of the steps, the chosen pair of nodes are both of color C. In the remaining 1/4 of the steps, neither of the chosen nodes are of color C and a node that is not of color C is placed in L. Since in 3/4 of the steps, the node placed in L is of color C, roughly 75% of the nodes in L should be of color C and by symmetry, roughly 25% of the nodes in R should be of color C. If there are k nodes on each side of the partition, then the imbalance of color C should be approximately $k(3/4 - 1/4)/2 = k/4$. By symmetry, the imbalance of the other color is roughly $-k/4$. This heuristic explanation can be generalized to three or more colors, assuming that over time the gap between each pair of successive imbalances grows. With this assumption, for example, in the case of three colors, the color of maximum imbalance is expected to be placed in L in approximately 5/9 of the steps of phase 1, one of the other colors is placed in L in approximately 3/9 of the steps, and the remaining color is placed in L in approximately 1/9 of the steps. This implies that the imbalances should be approximately $2k/9, 0$, and $-2k/9$.

In light of these observations, we should expect a gap of $\Theta(n)$ between any pair of imbalances at the end of phase 2. In this event, it should be possible to separate the nodes from phase 2 into l distinct color classes in phase 3, rather than simply grouping the nodes into two groups as is done in Algorithm 1. In this way, recursion can be avoided.

In the next section, we present an algorithm that partitions all l color classes directly from the partition of phase 1, and in Section 5.3 we prove that it succeeds in finding the minimum partition with high probability.

5.2 Algorithm

Algorithm 2:

Phase 1: Construct L_1 and R_1 as in Algorithm 1.

Phase 2: Construct L_2 and R_2 as in Algorithm 1.

Phase 3: In this phase, the remaining unexamined nodes are partitioned into l, rather than 2, groups as follows. For each remaining unexamined node v, let $l_2(v)$ denote the number of edges from node v to nodes in L_2. Let $o_0 < o_1 < \ldots < o_j$ be the ordered set of values $l_2(v)$. Let the $l-1$ largest differences between pairs of consecutive numbers in this ordered list be

$$o_{a_1} - o_{a_1 - 1}, o_{a_2} - o_{a_2 - 1}, \ldots, o_{a_{l-1}} - o_{a_{l-1}-1}.$$

If $l_2(v) < o_{a_1}$ then put v in S_1. For $2 \leq i \leq l-1$, if $o_{a_{i-1}} \leq l_2(v) < o_{a_i}$ then put node v in S_i. Finally, if $o_{a_{l-1}} < l_2(v)$ then put v in S_l.

Phase 4: In parallel for each node v examined in phases 1 and 2, assign v greedily to S_i if the number of nodes in S_i adjacent to v is at least the number of nodes in S_j adjacent to v for all $j \neq i$ (breaking ties arbitrarily).

5.3 Analysis of Algorithm 2

We claim that the following facts are true of Algorithm 2 with high probability. At the end of phase 1, the difference between the imbalances of any two distinct colors is at least $n^{1-\epsilon}/l$. At the end of phase 2, the difference between the imbalances of any two distinct colors is $\Theta(n)$. At the end of phase 3, within each set S_i all nodes have the same color. Finally, each node v examined in phase 4 is assigned to the set S_i with nodes of the same color as v.

We next analyze phase 1 of the Algorithm 2, by extending the ideas in analysis of phase 1 of Algorithm 1. Phases 2, 3, and 4 of Algorithm 2 can be analyzed by similar extensions of the corresponding phases of Algorithm 1; we comment on these at the end of this section.

5.3.1 Phase 1

Theorem 13 *At the end of phase 1, with high probability, the difference between the imbalances of any two distinct colors is at least $n^{1-\epsilon}/l$.*

Let $x_1 \geq x_2 \geq \ldots \geq x_l$ be the ordered sequence of imbalances of the colors in partition (L_1, R_1), as a function of the number of steps of phase 1. (Note that the rank of a particular color in this list may change over time; for example, x_2 may be the rank of different colors at different times.) From the analysis of Algorithm 1, we already know that with high probability, in $n^{1-\epsilon/2}$ steps, $x_1 - x_l \geq 2n^{1-\epsilon}$. Therefore, for some d, $x_d - x_{d+1} \geq 2n^{1-\epsilon}/l$. We say that *a good gap arises* between d and e at some step of phase 1 if after that step, for the first time $x_d - x_e \geq 2n^{1-\epsilon}/l$. We say that phase 1 is *well behaved* if, once a good gap arises between a pair of contiguous imbalances, a gap of at least $n^{1-\epsilon}/l$ remains in all further steps of phase 1. We say that phase 1 is *normal* if for every color class C and for all k, the number of nodes of color C that have not been examined after k steps of Phase 1 lies between $\frac{n-2k}{l} - n^{1/2+\frac{\epsilon}{4}}$ and $\frac{n-2k}{l} + n^{1/2+\frac{\epsilon}{4}}$.

By Lemma 6, Phase 1 is normal with high probability. The following lemmas state that with high probability phase 1 is well behaved and, given that phase 1 is well behaved and normal, a good gap arises between each pair of contiguous imbalances during phase 1. Theorem 13 follows directly from Lemmas 14 and 15.

Lemma 14 *Phase 1 is well behaved with high probability.*

Lemma 15 *Suppose that phase 1 is well behaved and normal. Suppose also that at some step of phase 1, a good gap has not arisen between x_d and x_e, where $d < e$, but that (i) either $d = 1$ or a good gap has arisen between between $d-1$ and d, and (ii) either $e = l$ or a good gap has arisen between e and $e+1$. Then, within $n^{1-\epsilon/2}/(2l)$ more steps, for some $k, d \leq k < e$, a good gap will arise between x_k and x_{k+1}, with high probability.*

We first consider Lemma 15. To prove this, we need the following claim. The notation "$f(t) \geq -O(g(t))$" used in the claim means that for some constant c, for sufficiently large t, $f(t) \geq -cg(t)$.

Claim 16 *Suppose that Phase 1 is normal. Suppose that at step t, d, e are such that $1 \leq d < e \leq l$, either $d = 1$ or $x_{d-1} - x_d = \Omega(n^{1-\epsilon})$, and either $e = l$ or $x_e - x_{e+1} = \Omega(n^{1-\epsilon})$. Then at step $t+1$ of phase 1 (for any execution of the algorithm up to step t),*
(i) $\mathrm{Prob}[x_d - x_e$ *increases*$] = \Omega(1)$,
(ii) $\mathrm{Prob}[x_d - x_e$ *increases*$] - \mathrm{Prob}[x_d - x_e$ *decreases*$] \geq -O(n^{-1/2+\epsilon/4})$, *and*
(iii) if $x_d - x_e = \Omega(n^{1/2-\epsilon/2})$ then

$$\mathrm{Prob}[x_d - x_e \textit{ increases}] - \mathrm{Prob}[x_d - x_e \textit{ decreases}] = \Omega(\min\{(x_d - x_e)\Delta/\sqrt{t}, 1\}).$$

Proof For notational convenience, we just consider here the case that all of the x_i's are distinct. As in Claim 2, let $[x, x']$ denote the event that the colors of the nodes 1 and 2 chosen at step $t+1$ of phase 1 have imbalances x and x', respectively.

The probability of event $[x_d, x_e]$ is $\Omega(1)$. Moreover, the proof of Claim 2 shows that in the event $[x_d, x_e]$, the probability that $x_d - x_e$ increases is at least the probability that $x_d - x_e$ decreases. From this, part (i) of Claim 16 follows. Also, part (ii) is true in the event $[x_d, x_e]$.

To complete the proof of part (ii), it is sufficient (by symmetry on x and x') to consider events $[x, x']$ where x is not in $\{x_d, x_e\}$ and x' is in $\{x_d, x_e\}$. In the event that $x_d > x > x_e$, the same argument used in Claim 2, part (ii) shows that

$$\mathrm{Prob}[x_d - x_e \text{ increases}] \geq \mathrm{Prob}[x_d - x_e \text{ decreases}].$$

We next show that (ii) holds in the event that $x > x_d$. (This event is only possible if $d > 1$.) The proof that (ii) holds in the event that $x < x_d$ is symmetrical. In what follows, assume that $x > x_d$ and that $x' \in \{x_d, x_e\}$. Let E denote the event that $x' = x_d$ and let E' denote the event that $x' = x_e$.

Note that $\mathrm{Prob}[x_d - x_e$ increases$] - \mathrm{Prob}[x_d - x_e$ decreases$]$ given that $x > x_d$ and $x' \in \{x_d, x_e\}$ equals the following quantity, henceforth denoted by (*):

$$\mathrm{Prob}[E]\,(\mathrm{Prob}[x_d - x_e \text{ increases} \mid E] - \mathrm{Prob}[x_d - x_e \text{ decreases} \mid E])$$
$$+$$
$$\mathrm{Prob}[E']\,(\mathrm{Prob}[x_d - x_e \text{ increases} \mid E'] - \mathrm{Prob}[x_d - x_e \text{ decreases} \mid E'])\,.$$

It is sufficient to show that (*) $\geq -O(n^{-1/2+\epsilon/4})$.

To prove this, we will use three inequalities, to be proved later:

$$\text{Prob}[x_d - x_e \text{ increases} \mid E'] = 1 - \exp(-\Omega(n^{\epsilon/2})) \tag{1}$$

$$\text{Prob}[x_d - x_e \text{ decreases} \mid E] = 1 - \exp(-\Omega(n^{\epsilon/2})) \tag{2}$$

and

$$|\text{Prob}[E] - \text{Prob}[E']| = O(n^{-1/2+\epsilon/4}). \tag{3}$$

The fact that (*) $\geq -O(n^{-1/2+\epsilon/4})$ follows by combining inequalities (1), (2), and (3):

$$\begin{aligned} (*) &\geq (\text{Prob}[E'] - \text{Prob}[E])(-\exp(-\Omega(n^{\epsilon/2})) + 1 - \exp(-\Omega(n^{\epsilon/2}))) \\ &= (\text{Prob}[E'] - \text{Prob}[E])(1 - \exp(-\Omega(n^{\epsilon/2}))) \geq -O(n^{-1/2+\epsilon/4}). \end{aligned}$$

We now prove Equation (1). Note that $x_d - x_e$ increases in event E' if and only if node 1 is placed in L_1 and node 2 is placed in R_1. Let the number of nodes in L_1 that have the same color as node 1 be $b + x$. Let the number of nodes in L_1 that have color with imbalance x_e be $b_e + x_e$. Let

$$X = B(b+b_e+x-x_e, p) + B(2t-b-b_e-x+x_e, r) - B(b+b_e-x+x_e, p) - B(2t-b-b_e+x-x_e, r).$$

(Recall the similar expression in the proof of Claim 2.) Since we assume that $x_{d-1} - x_d = \Omega(n^{1-\epsilon})$, and also we have that $x \geq x_{d-1} > x_e$, it follows that $x - x_e = \Omega(n^{1-\epsilon})$. Therefore, $EX = \Omega(n^{1-\epsilon}\Delta) = \Omega(n^{1/2})$. Since X is the sum of $4t = O(n^{1-\epsilon/2})$ independent random variables, we have from Azuma's inequality (Theorem 1) that

$$\text{Prob}[X \leq 0] \leq 2\exp(-\Omega(n)/\Theta(n^{1-\epsilon/2})) = \exp(-\Omega(n^{\epsilon/2})).$$

Therefore,

$$\text{Prob}[x_d - x_e \text{ increases} \mid E'] = \text{Prob}[X > 0] + (1/2)\text{Prob}[X = 0] \geq 1 - \exp(-\Omega(n^{\epsilon/2})).$$

The proof of Equation (2) follows symmetrically. Equation (3) follows from the hypothesis that Phase 1 is normal. This completes the proof of part (ii) of the claim.

Finally, consider part (iii). Using the arguments of part (iii) of Claim 2, we can show that if $x_d - x_e = \Omega(n^{1/2-\epsilon/2})$ then

$$\text{Prob}[x_d - x_e \text{ increases}] - \text{Prob}[x_d - x_e \text{ decreases}] = \Omega(\min\{(x_d - x_e)\Delta/\sqrt{t}, 1\}) - O(n^{-1/2+\epsilon/4}).$$

(The term $O(n^{-1/2+\epsilon/4})$ arises due to contributions of events $[x, x']$ in which $x \in \{x_d, x_e\}$ and $x' \in [x_{d+1} \ldots x_{e-1}]$. From part (ii) of the Claim, these contributions are lower bounded by $-O(n^{-1/2+\epsilon/4})$.) To complete the proof, we note that $n^{-1/2+\epsilon/4} = o((x_d - x_e)\Delta/\sqrt{t})$, since $t \leq n^{1-\epsilon/2}$, $x_d - x_e = \Omega(n^{1/2-\epsilon/2})$, and $\Delta = n^{-1/2+\epsilon}$. □

We now prove Lemma 15.

Proof of Lemma 15:

Consider the case where $d > 1$ and $e < l$ (the other cases are simpler). The task of analyzing the evolution of $x_d - x_e$ is complicated by the assumption that all of phase 1 is well behaved. We consider a new process for building up a pair of sets (L_1, R_1). In this new process, pairs of nodes are chosen from the set of unexamined nodes and added to (L_1, R_1) as in phase 1 of Algorithm 2, with the following difference: once a gap arises between two contiguous imbalances x_{d-1} and x_d, *the imbalance is artificially kept to be at least* $n^{1-\epsilon}/l$ as follows. If as a result of some step, $x_d - x_{d-1}$ would dip below $n^{1-\epsilon}/l$, then the two nodes chosen at that step are discarded. Instead, new nodes are added to L_1 and R_1 so as to ensure that *all* differences $x_i - x_{i+1}$ increase, while maintaining the equality $|L_1| = |R_1|$. This can be achieved by adding to L_1: $l-1$ nodes of color with imbalance x_1, $l-2$ nodes of color with imbalance x_2, and so on, and also adding to R_1 $\sum_{i=1}^{l}(l-i)$ nodes with imbalance x_l. The newly added nodes are not taken from the pool of n nodes, but there is an edge between each new node and each unexamined node with probability p if the nodes have the same color and with probability r otherwise.

Claim 16 can be shown to be true for the new process. Roughly, at each step of the new process, if the artificial mechanism is not employed, the probability that $x_i - x_j$ increases is as for the original process. If the artificial mechanism is employed, then the probability that any difference $x_i - x_j$ increases is 1.

In what follows, we use Prob$'$ to refer to the probability of an event in the new process, and Prob to refer to the probability of an event in the original process (i.e. phase 1 of algorithm 2). Let *success* denote the event that $x_d - x_e$ reaches $2n^{1-\epsilon}/l$ within $n^{1-\epsilon/2}/(2l)$ steps.

Using Claim 16, we now show that in the new process, Prob$'$[*success*] is high. The proof of this is similar to the proof of Theorem 5, with $x = x_d - x_e$. The only difference is in the analysis of the first subphase, which starts at time 0 and continues until $x \geq n^{1/2-\epsilon/2}/2$. Now, the random process $x(t)$ no longer behaves as an unbiased random walk; instead, the walk potentially has a slight negative drift. That is, from Claim 16 (ii) and Lemma 4, if Y is the random walk of Lemma 4 with $\epsilon() = O(n^{-1/2+\epsilon/4})$, then

$$\text{Prob[starting at 0, } x \text{ reaches } n^{1/2-\epsilon/2}/2 \text{ within } n^{1-\epsilon/2}/(2l) \text{ steps]}$$

$$\geq$$

$$\text{Prob[starting at 0, } Y \text{ reaches } n^{1/2-\epsilon/2} \text{ within } dn^{1-\epsilon/2}/(2l) \text{ steps]} - \exp(-n^{\Theta(\epsilon)}),$$

where $d > 0$ is a constant.

From Feller [7, XIV.2.4]) (see also [12]), the expected time for Y to reach $n^{1/2-\epsilon/2}$ is

$$-\frac{b}{g-u} + \frac{g}{(g-u)^2}\left[\left(\frac{g}{u}\right)^b - 1\right]$$

where $b = n^{1/2-\epsilon/2}$, g is the probability that Y decreases, and u is the probability that Y increases. Note that $g/u = 1 + \Theta(n^{-1/2+\epsilon/4})$. Since $(g/u - 1)b < 1$ for sufficiently large n, we have that

$$(g/u)^b = 1 + \Theta((g/u-1)b) = 1 + \Theta(n^{-1/2+\epsilon/4}n^{1/2-\epsilon/2}) = 1 + \Theta(n^{-\epsilon/4}).$$

Therefore, the expected time for the walk Y to reach $n^{1/2-\epsilon/2}$ is

$$O(\frac{1}{(n^{-1/2+\epsilon/4})^2}n^{-\epsilon/4}) = O(n^{1-3\epsilon/4}).$$

We can now complete the analysis of the first subphase in a manner similar to that of Theorem 5. Namely, by Markov's inequality, with probability at least 1/2, Y reaches $n^{1/2-\epsilon/2}$ within time

$O(n^{1-3\epsilon/4})$; moreover this holds regardless of where the walk starts within the interval. Within $dn^{1-\epsilon/2}/(2l)$ steps, the number of periods of length $O(n^{1-3\epsilon/4})$ is $\Omega(n^{\epsilon/4})$ (since l is a constant). Therefore, the probability that Y reaches $n^{1/2-\epsilon/2}$ within $n^{1-\epsilon/2}/(2l)$ steps is $1-\exp(-\Omega(n^{\epsilon/4}))$.

By analyzing the remaining subphases as in the proof of Theorem 5, it follows that in the new process, $x = x_i - x_j$ reaches $2n^{1-\epsilon}/l$ within $n^{1-\epsilon/2}/(2l)$ steps with high probability. That is, $\text{Prob}'[\mathit{success}] = 1-\exp(-n^{\Theta(\epsilon)})$.

We can now complete the proof. We say that the new process is *uneventful* if the artificial mechanism is never used during this process. It is clearly true that

$$\text{Prob}'[\mathit{success} \mid \text{ uneventful}] = \text{Prob}[\mathit{success} \mid \text{ well behaved}]$$

and that

$$\text{Prob}'[\text{uneventful}] = \text{Prob}[\text{well behaved}].$$

Also, note that

$$\text{Prob}'[\mathit{success}] \leq \text{Prob}'[\mathit{success} \mid \text{ uneventful}] + \text{Prob}[\text{not well behaved}].$$

By Lemma 14, Prob[not well behaved] $= \exp(-n^{\Theta(\epsilon)})$. Therefore,

$$\begin{aligned} & \text{Prob}[x_i - x_j \text{ reaches } 2n^{1-\epsilon}/l \text{ in } n^{1-\epsilon/2}/(2l) \text{ steps} \mid \text{ well behaved}] \\ = & \text{ Prob}[\mathit{success} \mid \text{well behaved}] \\ = & \text{ Prob}'[\mathit{success} \mid \text{ uneventful}] \\ \geq & \text{ Prob}'[\mathit{success}] - \exp(-n^{\Theta(\epsilon)}) \\ = & \text{ } 1-\exp(-n^{\Theta(\epsilon)}). \end{aligned}$$

□

Finally, we note that the proof of Lemma 14 is quite similar to that of Lemma 15. Specifically, suppose that $x_i - x_{i+1} \geq n^{1-\epsilon}/l$ at some step of phase 1. Then it can be shown that, at that step, Prob[$x_i - x_{i+1}$ increases] $-$ Prob[$x_i - x_{i+1}$ decreases] $= \Omega(1)$. Roughly, this is because (i) the probability that a pair of nodes with imbalances x_i and x_{i+1} are chosen at this step is $\Omega(1)$ and in this event Prob[$x_i - x_{i+1}$ increases] $-$ Prob[$x_i - x_{i+1}$ decreases] $= \Omega(1)$; (ii) in the event that a pair of nodes in which exactly one node of the pair has imbalance x_i or x_{i+1} is chosen at this step, Prob[$x_i - x_{i+1}$increases] $-$ Prob[$x_i - x_{i+1}$ decreases] $\geq -O(n^{-1/2+\epsilon/4})$, by an argument similar to that given in the proof of part (ii) of Claim 16; and (iii) in the event that neither node in the chosen pair has imbalance x_i or x_{i+1} then $x_i - x_{i-1}$ remains unchanged in that step.

5.3.2 Phases 2 and 3

We claim that at the end of phase 2, the difference between the imbalances of any two distinct colors is $\Theta(n)$ with high probability. This follows if for each i, at the end of phase 2, $x_i - x_{i+1} = \Theta(n)$ with high probability. For any fixed i, note that each pair of nodes (1,2) examined in phase 2 independently contributes 1/2, 0, or -1/2 to $x_i - x_{i+1}$. The analysis of phase 2 rests on the fact that for any pair of nodes (1,2) examined in phase 2,

$$\text{Prob}[(1,2) \text{ contributes positively to } x_i - x_{i+1}] - \text{Prob}[(1,2) \text{ contributes negatively to } x_i - x_{i+1}] = \Omega(1).$$

In summary, the proof of this fact can be done by considering the following events. If the pair of nodes (1,2) is such that one node has imbalance x_i and the other x_{i+1}, then the above fact holds in this event; moreover the probability of this event is $\Omega(1)$. If exactly one node of the pair has imbalance in the set $\{x_i, x_{i+1}\}$, then an analysis similar to that of Claim 16, part (ii) shows that

$$\text{Prob}[(1,2) \text{ contributes positively to } x_i - x_{i+1}] - \text{Prob}[(1,2) \text{ contributes negatively to } x_i - x_{i+1}]$$
$$\geq -O(n^{-1/2+\epsilon/4}).$$

Finally, if no node of the pair has imbalance in the set $\{x_i, x_{i+1}\}$ or both have the same imbalance, no change to $x_i - x_{i+1}$ results.

We claim that at the end of phase 3, with high probability, within each set S_i all nodes have the same color. As in the analysis of phase 3 of algorithm 1, the values $l_2(v)$ are clustered in short intervals (of length $2n^{1/2+\epsilon/2}$) centered at the values $EL_2(C)$, where $EL_2(C)$ is the expected number of edges of an unexamined node of color C to nodes in the set L_2. Moreover, an extension to Claim 11 shows that the difference between any two values $EL_2(C)$ for distinct colors C is $\Omega(n^{1/2+\epsilon}/l)$ with high probability. From these facts, simple algebra shows that, with high probability, phase 3 puts clusters of nodes centered around one of the values $EL_2(C)$ in the same set, and that such nodes are all of the same color.

6 Future Work

In the partitioning algorithms analyzed in this paper, a pair of nodes is considered at each step of phases 1 and 2. A variant of the algorithm is to consider only one node per step, rather than a pair of nodes. This node is placed on the side of the partition to which it has the greatest edge density. We observed that this variant performs better experimentally than our two-node algorithm. It would be interesting to prove that the one-node variant can be used to find an optimal partition with high probability on the planted partition model.

It would also be interesting to extend our results to the case where the number of color classes is unknown and where the color classes are of unequal size. Such cases arise in certain clustering applications.

The following related problem may also be relevant to data clustering applications. Consider a set of data samples, each of which has some attributes from a given set. Let M be a boolean matrix with entry $[i, j]$ having value 1 if and only if sample i has attribute j. The simplest version of the problem is to bisect both the samples (rows of the matrix) and the attributes (columns of the matrix) into two equal-sized groups, say $R1, R2$ and $C1, C2$, respectively, so as to minimize the number of 1-entries in the submatrices $R1 \times C2$ and $R2 \times C1$. If the matrix M is generated so that it has "planted" structure, with the probability of entries in $R1 \times C1$ and $R2 \times C2$ being p and the probability of entries in $R1 \times C2$ and $R2 \times C1$ being $r < p$, can a variant of the algorithm in this paper locate this planted structure?

7 Acknowledgements

We thank Juan Alemany for conducting experimental tests of our algorithm and for his valuable feedback throughout this work. We also thank the anonymous referees for their very thoughtful and constructive comments on the paper, which have greatly improved the exposition.

References

[1] S. Arora, D. Karger and M. Karpinski. "Polynomial time approximation schemes for dense instances of NP-hard problems," in *Proceedings of the 27th Annual ACM Symposium on Theory of Computing*, 1995, 284–293.

[2] T. Bui. "On bisecting random graphs," Report Number MIT/LCS/TR-287, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1983.

[3] T. Bui, S. Chaudhuri, T. Leighton, and M. Sipser. "Graph bisection algorithms with good average case behavior," *Combinatorica*, 7:2 (1987), 171–191.

[4] T. N. Bui and B. R. Moon. "Genetic algorithm and graph partitioning," *IEEE Transactions on Computers*, 45:7 (1996), 841–855.

[5] R. B. Boppana. "Eigenvalues and graph bisection: an average-case analysis," in *Proceedings of the 28th Annual IEEE Symposium on Foundations of Computer Science*, 1987, 280–285.

[6] M. E. Dyer and A. M. Frieze. "The solution of some random NP-hard problems in polynomial expected time," J. Algorithms 10:4 (1989), 451-89.

[7] W. Feller. *An introduction to probability theory and its applications*, Volume 1, Third Edition, John Wiley and Sons, New York, 1968.

[8] C. M. Fiduccia and R. M. Mattheyses. "A linear-time heuristic for improving network partitions," in *Proceedings of the ACM IEEE Nineteenth Design Automation Conference*, 1982, 174–181.

[9] A. Frieze and R. Kannan. "The regularity lemma and approximation schemes for dense problems," in *Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science*, 1996, 12–20.

[10] M. R. Garey, D. S. Johnson, and L. Stockmeyer. "Some simplified NP-complete graph problems," *Theoretical Computer Science*, 1 (1976), 237–267.

[11] G.R. Grimmett and D.R. Stirzaker. *Probability and Random Processes (second edition)*, Oxford University Press, 1992.

[12] M. Jerrum and G. B. Sorkin. "The Metropolis algorithm for graph bisection," Discrete Applied Mathematics, 82:1-3 (1998), 155-75.

[13] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon. "Optimization by simulated annealing: an experimental evaluation; part 1, graph partitioning," *Operations Research*, 37:6 (November-December 1989), 865–892.

[14] A. Juels. *Topics in Black Box Optimization*, Ph.D. Thesis, EECS Department, U. California at Berkeley, 1996.

[15] B. W. Kernighan and S. Lin. "An efficient heuristic procedure for partitioning graphs," Bell. Syst. Tech. J. 49, 291–307.

[16] S. Kirkpatrick, C. D. Gelatt, and M. Vecchi. "Optimization by simulated annealing," *Science,* 220:4598 (1983), 671–680.

[17] L. Kucera. "Expected complexity of graph partitioning problems," *Discrete Applied Mathematics* 57 (1995), 193-212.

[18] C. McDiarmid. "On the method of bounded differences," in *London Society Lecture Note Series*, Volume 141, Cambridge University Press, 1989, 148–188.

[19] V. V. Petrov. *Sums of independent random variables*, Springer-Verlag, New York, 1975.

[20] G. H. Weiss. "Aspects and Applications of the Random Walk," *Random Materials and Processes*, Series Eds. H. Stanley and E. Guyon, North Holland, 1994.

[21] D. Williams. *Probability with Martingales,* Cambridge Univ. Press, 1991.