Why do simple algorithms for triangle enumeration work in the real world?

Jonathan W. Berry [*] Luke A. Fostvedt [†] Daniel J. Nordman [†] Cynthia A. Phillips [*]
C. Seshadhri [*] Alyson G. Wilson [‡]

Abstract

Triangle enumeration is a fundamental graph operation. Despite the lack of provably efficient (linear, or slightly superlinear) worst-case algorithms for this problem, practitioners run simple, efficient heuristics to find all triangles in graphs with millions of vertices. How are these heuristics exploiting the structure of these special graphs to provide major speedups in running time?

We study one of the most prevalent algorithms used by practitioners. A trivial algorithm enumerates all paths of length 2, and checks if each such path is incident to a triangle. A good heuristic is to enumerate only those paths of length 2 where the middle vertex has the lowest degree. It is easily implemented and is empirically known to give remarkable speedups over the trivial algorithm.

We study the behavior of this algorithm over graphs with heavy-tailed degree distributions, a defining feature of real-world graphs. The erased configuration model (ECM) efficiently generates a graph with asymptotically (almost) any desired degree sequence. We show that the expected running time of this algorithm over the distribution of graphs created by the ECM is controlled by the $\ell_{4/3}$-norm of the degree sequence. As a corollary of our main theorem, we prove expected linear-time performance for degree sequences following a power law with exponent $\alpha \geq 7/3$, and non-trivial speedup whenever $\alpha \in (2, 3)$.

1. INTRODUCTION

Finding triangles in graphs is a classic theoretical problem with numerous practical applications. The recent explosion of work on social networks has led to a great interest in fast algorithms to find triangles in graphs. The social sciences and physics communities often study triangles in real networks and use them to reason about underlying social processes [Col88,Por98,WS98,Bur04,Bur07,FWVDC10]. Much of the information about triangles in the last four papers is determined by a complete enumeration of all triangles in a (small) graph. Triangle enumeration is also a fundamental subroutine for other more complex algorithmic tasks [BHLP11,FH97].

From a theoretical perspective, Itai and Rodeh [IR78] gave algorithms for triangle finding in $O(n^\omega)$ time (where n is the number of vertices and ω is the matrix multiplication constant) using fast matrix multiplication. Vassilevska Williams and Williams [WW10] show deep connections between matrix multiplication and (edge-weighted) triangle enumeration. But much of this work is focused on dense graphs. Practitioners usually deal with massive sparse graphs with large variance in degrees, where sub-quadratic time algorithms can be trivially obtained, but are still too slow to run.

Practioners enumerate triangles on massive graphs (with millions of vertices) using fairly simple heuristics, which are often easily parallelizable. This work is motivated by the following question: *can we theoretically explain why simple algorithms for triangle enumeration work in the real world?*

Consider a trivial algorithm. Take an undirected graph with n vertices, m edges, and degree sequence $d_1, d_2, \ldots, d_n$ (so the degree of vertex v is d_v). We refer to paths of length 2 as *wedges*. Call a wedge *closed* if it participates in a triangle and *open* otherwise. Simply enumerate all wedges and output the closed ones. The total running time is $\Theta(\sum_v d_v^2)$ (assume that checking if a wedge is closed can be done in constant time), since every wedge involves a pair of neighbors for the middle vertex. We will henceforth refer to this as *the* trivial algorithm. A simple heuristic is to only enumerate paths where the middle vertex has the lowest degree among the 3 vertices in the path. We call this algorithm MinBucket.

1. Create n empty buckets $B_1, B_2, \ldots, B_n$.
2. For each edge (u, v): if $d_u \leq d_v$, place it in B_u, otherwise place it in B_v. Break ties consistently.
3. For each bucket B_v: iterate over all wedges formed by

[*] Sandia National Laboratories, Albuquerque, NM 87185. {jberry,caphill,csesha}@sandia.gov. This manuscript has been authored by Sandia Corporation under Contract No. DE-AC04-94AL85000 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

[†] Iowa State University, Ames, Iowa 50011. {fostvedt,dnordman}@iastate.edu

[‡] North Carolina State University. {alyson_wilson@ncsu.edu}

edges in B_v and output closed ones.

MinBucket is quite common in practice (sometimes taking the somewhat strange name *nodeIterator++*) and has clean parallel implementations with no load balancing issues [SW05a, Coh09, SV11]. For such simple algorithms, the total work pretty much determines the parallel runtime. For example, it would take n processors with perfect speed up running a $\Theta(n^2)$-work algorithm to compete with a single processor running a $\Theta(n)$-work algorithm.

MinBucket is often the algorithm of choice for triangle enumeration because of its simplicity and because it beats the trivial algorithm by orders of magnitude, as shown in the previous citations. (A quick check shows at least 60 citations to [Coh09], mostly involving papers that deal with massive scale graph algorithms.) The algorithm itself has been discovered and rediscovered in various forms over the past decades. The earliest reference the authors could find was from the mid-80s where Chiba and Nishizeki [CN85] devised a sequential version of the above algorithm. We provide a more detailed history later.

Nonetheless, MinBucket has a poor worst-case behavior. It would perform terribly on a high degree regular bipartite graph. If the input sparse graph (with high variance in degree) simply consisted of many such bipartite graphs of varying sizes, MinBucket would perform no better than its trivial cousin. Then why is it good in practice?

1.1 Results

Since the seminal results of Barabási and Albert [BA99], Faloutsos et al [FFF99], Broder et al [BKM$^+$00], researchers have assumed that massive graphs obtained from the real world have *heavy-tailed degree distributions* (often approximated as a power law). The average degree is thought to be a constant (or very slowly growing), but the variance is quite large. The usual approximation is to think of the number of vertices of degree d as decaying roughly as $1/d^\alpha$ for some small constant α.

This seems to have connections with MinBucket. If edges tend to connect vertices of fairly disparate degrees (quite likely in a graph with large variance in degrees), MinBucket might provably give good running times. This is exactly what we set out to prove, for a natural distribution on heavy-tailed graphs.

Consider any list of positive integers $\mathbf{d} = (d_1, d_2, \dots, d_n)$, which we think of as a "desired" degree sequence. In other words, we wish to construct a graph on n vertices where vertex $v \in [n]$ has degree d_v. The *configuration model* (CM) [BC78, Bol80, MR98, New03] creates a random graph that almost achieves this. Imagine vertex v being incident to d_v "stubs", which can be thought of as half-edges. We take a random perfect matching between the stubs, so pairs of stubs are matched to each other. Each such pair creates an edge, and we end up with a multigraph with the desired degree sequence. Usually, this is converted to a simple graph by removing parallel edges and self-loops [BDML06]. We refer to this graph distribution as the erased configuration model, $ECM(\mathbf{d})$, for input degree sequence $\mathbf{d}$. This model has a fairly long history (which we relegate to a later section) and is a standard method to construct a graph with a desired degree sequence. It is closely connected to models given by Chung and Lu [CL02, CLV03] and Mihail and Papadimitriou [MP02], in the context of eigenvalues of graphs with a given degree sequence. These models simply connect vertices u and v independently with probability $d_u d_v/2m$, similarly to the Erdős-Rényi construction.

Our main theorem gives a bound on the expected running time of MinBucket for $ECM(\mathbf{d})$. We set $m = (\sum_v d_v)/2$. We will henceforth assume that $0 < d_1 \leq d_2 \dots \leq d_n$ and that $d_n < \sqrt{m}/2$. This "truncation" is a standard assumption for analysis of the configuration model [MR98, CL02, MP02, CLV03, New03, BDML06]. We use $\sum_v$ as a shorthand for $\sum_{i=1}^n$, since it is a sum over all vertices. The run time bottleneck for MinBucket is in wedge enumeration, and checking whether a wedge is closed is often assumed to be a constant time operation. Henceforth, when we say "running time," we mean the number of wedges enumerated.

Theorem 1.1. *Consider a degree sequence $\mathbf{d} = (d_1, d_2, \dots, d_n)$, where $m = (\sum_v d_v)/2$ and $d_n < \sqrt{m}/2$. The expected (over $ECM(\mathbf{d})$) number of wedges enumerated by* MinBucket *is $O(n + m^{-2}(\sum_v d_v^{4/3})^3)$.*

(Our main theorem applies to Chung-Lu graphs as well. Details are given in §6.) Before we actually make sense of this theorem, let us look at a corollary of this theorem. It has been repeatedly observed that degree sequences in real graphs have heavy tails, often approximated as a *power law* [BA99]. Power laws say something about the moments of the degree distribution (equivalently, norms of the degree sequence). Since it does not affect our main theorem or corollary, we choose a fairly loose definition of power law. This is a binned version of the usual definition, which states the number of vertices of degree d is proportional to n/d^α. (Even up to constants, this is never precisely true because there are many gaps in real degree sequences.)

Definition 1.2. *A degree sequence $\mathbf{d}$ satisfies a power law of exponent $\alpha > 1$ if the following holds for all $k \leq \log_2 d_n - 1$: for $d = 2^k$, the number of sequence terms in $[d, 2d]$ is $\Theta(n/d^{\alpha-1})$.*

The following shows an application of our theorem for common values of α. This bound is tight as we show in Section 5. (When $\alpha \geq 3$, the trivial algorithm runs in linear time because $\sum_v d_v^2 = O(n)$.)

Corollary 1.3. *Suppose a degree sequence $\mathbf{d}$ (with largest term $< \sqrt{m}/2$) satisfies a power law with exponent $\alpha \in (2, 3)$. Then the expected running time of* MinBucket *of $ECM(\mathbf{d})$ is asymptotically better than the trivial algorithm, and is* linear *when $\alpha > 7/3$.*

1.2 Making sense of Thm. 1.1

First, as a sanity check, let us actually show that Thm. 1.1 beats the trivial bound, $\sum_v d_v^2$. This is a direct application of Hölder's inequality for conjugates $p = 3$ and $q = 3/2$.

$$(\sum_v d_v^{4/3})^3 = (\sum_v d_v^{2/3} \cdot d_v^{2/3})^3 \leq \left(\sum_v d_v^{\frac{2}{3}\cdot 3}\right)^{3\cdot\frac{1}{3}} \left(\sum_v d_v^{\frac{2}{3}\cdot\frac{3}{2}}\right)^{3\cdot\frac{2}{3}} = (2m)^2 (\sum_v d_v^2)$$

Rearranging, we get $m^{-2}(\sum_v d_v^{4/3})^3 = O(\sum_v d_v^2)$, showing that our bound at least holds the promise of being nontrivial.

Consider the uniform distribution on the vertices. Assuming $m > n$, we can write our running time bound as

$n(\mathbf{E}[d_v^{4/3}])^3$, as opposed to the trivial bound of $\sum_v d_v^2 = n\mathbf{E}[d_v^2]$. If the degree "distribution" (think of the random variable given by the degree of a uniform random vertex) has a small 4/3-moment, the running time is small. This can happen even though the second moment is large, and this is where MinBucketbeats the trivial algorithm. In other words, if the tail of the degree sequence is heavy but not too heavy, MinBucket will perform well.

And this is exactly what happens when $\alpha > 2$ for power law degree sequences. When $\alpha > 7/3$, the 4/3-moment becomes constant and the running time is linear. (It is known that for ECM graphs over power law degree sequences with $\alpha > 7/3$, the clustering coefficient (ratio of triangles to wedges) converges to zero [New03].) We show in §5 that the running time bound achieved in the following corollary for power laws with $\alpha > 2$ is tight. When $\alpha \leq 2$, MinBucket gets no asymptotic improvement over the trivial algorithm. For convenience, we will drop the big-Oh notation, and replace it by $\lessdot$. So $A \lessdot B$ means $A = O(B)$.

Proof. (of Cor. 1.3) First, let us understand the trivial bound. Remember than d_n is the maximum degree.

$$\sum_v d_v^2 \lessdot \sum_{k=1}^{\log_2 d_n - 1} (n/2^{k(\alpha-1)})2^{2k} = n \sum_{k=1}^{\log_2 d_n - 1} 2^{k(3-\alpha)} \lessdot n + n d_n^{3-\alpha}$$

We can argue that the expected number of wedges enumerated by the trivial algorithm is $\Omega(\sum_v d_v^2)$ (Claim 3.3). Now for the bound of Thm. 1.1.

$$\begin{aligned} m^{-2}(\sum_v d_v^{4/3})^3 &\lessdot n^{-2}\Big(\sum_{k=1}^{\log_2 d_n - 1} (n/2^{k(\alpha-1)})2^{4k/3}\Big)^3 \\ &= n\Big(\sum_{k=1}^{\log_2 d_n - 1} 2^{k(7/3-\alpha)}\Big)^3 \lessdot n + n d_n^{7-3\alpha} \end{aligned}$$

Regardless of d_n, if $\alpha > 7/3$, then the running time of MinBucket is linear. Whenever $\alpha \in (2,3)$, $7 - 3\alpha < 3 - \alpha$, and MinBucket is asymptotically faster than a trivial enumeration. □

1.3 Significance of Thm. 1.1

Thm. 1.1 connects the running time of a commonly used algorithm to the norms of the degree sequences, a well-studied property of real graphs. So this important property of heavy-tails in real graphs allows for the algorithmic benefit of MinBucket. We have discovered that for a fairly standard graph model inspired by real degree distributions, MinBucket is very efficient.

We think of this theorem as a proof of concept: theoretically showing that a common property of real world inputs allows for the efficient performance of a simple heuristic. Because of our distributional assumptions as well as bounds on α, we agree with the (skeptical) reader that this does not fully explain why MinBucket works in the real world[1]. Nonetheless, we feel that this makes progress towards that, especially for a question that is quite hard to formalize. After all, there is hardly any consensus in the social networks community on what real graphs look like.

But the notion that distinctive properties of real world graphs can be used to prove efficiency of simple algorithms is a useful way of thinking. This is one direction to follow for going beyond worst-case analysis. Our aim here is not to design better algorithms for triangle enumeration, but to give a theoretical argument for why current algorithms do well.

The proof is obtained (as expected) through various probabilistic arguments bounding the sizes of the different buckets. The erased configuration model, while easy to implement and clean to define, creates some niggling problems for analysis of algorithms. The edges are not truly independent of each other, and we have to take care of these weak dependencies.

Why the 4/3-norm? Indeed, that is one of the most surprising features of this result (especially since the bound is tight for power laws of $\alpha > 2$). As we bound the bucket sizes and make sense of the various expressions, the total running time is expressed as a sum of various degree terms. Using appropriate approximations, it tends to rearrange into norms of the degree sequence. Our proof goes over two sections. We give some probabilistic calculations for the degree behavior in §3, which sets the stage for the run-time accounting. In §4, we start bounding bucket sizes and finally get to the 4/3-moment. In §5, we show that bounds achieved in the proof of Cor. 1.3 are tight. This is mostly a matter of using the tools of the previous sections. In §7, we give a tighter theorem (proof in full version) that gives an explicit expression for strong upper bounds on running time and experimentally show these more careful bounds closely approximate the expected runtime of ECM graphs.

2. RELATED WORK

The idea of using some sort of degree binning, orienting edges, or thresholding for finding and enumerating triangles has been used in many results. Chiba and Nishizeki [CN85] give bounds for a sequential version of MinBucket using the degeneracy of a graph. This does not give bounds for MinBucket, although their algorithm is similar in spirit. Alon, Yuster, and Zwick [AYZ97] find triangles in $O(m^{1.41})$ using degree thresholding and matrix multiplication ideas from Itai and Rodeh [IR78]. Chrobak and Eppstien [CE91] use acyclic orientations for linear time triangle enumeration in planar graphs. Vassilevska Williams and Williams [WW10] show that fast algorithms for weighted triangle enumeration leads to remarkable consequences, like faster all-pairs shortest paths. In the work most closely to ours, Latapy [Lat08] discusses various triangle finding algorithms, and also focuses on power-law graphs. He shows the trivial bound of $O(mn^{1/\alpha})$ when the power law exponent is α. Essentially, the maximum degree is $n^{1/\alpha}$ and that directly gives a bound on the number of wedges.

MinBucket has received attention from various experimental studies. Schank and Wagner [SW05b] perform an experimental study of many algorithms, including a sequential version of MinBucket which they show to be quite efficient. Cohen [Coh09] specifically describes MinBucket in the context of Map-Reduce. Suri and Vassilvitskii [SV11] do many experiments on real graphs in Map-Reduce and show major speedups (a few orders of magnitude) for MinBucket over the trivial enumeration. Tsourakakis [Tso08] gives a good survey of various methods used in practice for triangle counting and estimation.

Explicit triangle enumerations have been used for various applications on large graphs. Fudos and Hoffman [FH97] use triangle enumeration for a graph-based approach for solv-

[1] As the astute reader would have noticed, our title is a question, not a statement.

ing systems of geometric constraints. Berry et al [BHLP11] touch every triangle as part of their community detection algorithm for large graphs.

Configuration models for generating random graphs with given degree sequences have a long history. Bender and Canfield [BC78] study this model for counting graphs with a given degree sequence. Wormald [Wor81] looks at the connectivity of these graphs. Molloy and Reed [MR95a, MR98] study various properties like the largest connected component of this graph distribution. Physicists studying complex networks have also paid attention to this model [NSW01]. Britton, Deijfen, and Martin-Löf [BDML06] show that the simple graph generated by the ECM asymptotically matches the desired degree sequence. Aiello, Chung, and Lu [ACL01] give a model for power-law graphs, where edges (u, v) are independently inserted with probability $d_u d_v/2m$. This was studied for more general degree sequences in subsequent work by Chung, Lu, and Vu [CL02, CLV03]. Mihail and Papadimitriou [MP02] independently discuss this model. Most of this work focused on eigenvalues and average distances in these graphs. Newman [New03] gives an excellent survey of these models, their similarities, and applications.

3. DEGREE BEHAVIOR OF $ECM(\mathbf{d})$

We fix a degree sequence $\mathbf{d}$ and focus on the distribution $ECM(\mathbf{d})$. All expectations and probabilities are over this distribution. Because of dependencies in the erased configuration model, we will need to formalize our arguments carefully. We first state a general lemma giving a one-sided tail bound for dependent random variables with special conditional properties. The proof is in the appendix.

LEMMA 3.1. *Let $Y_1, Y_2, \ldots, Y_k$ be independent random variables, and $X_i = f_i(Y_1, Y_2, \ldots, Y_i)$ be 0-1 random variables. Let $\alpha \in [0, 1]$. Suppose $\Pr[X_1] \geq \alpha$ and $\Pr[X_i = 1|Y_1, Y_2, \ldots, Y_{i-1}] \geq \alpha$ for all i. Then, $\Pr[\sum_{i=1}^{k} X_i < \alpha k\delta] < \exp(-\alpha k(1-\delta)^2/2)$ for any $\delta \in (0, 1)$.*

We now prove a tail bound on degrees of vertices; the probability the degree of vertex v deviates by a constant factor of d_v is $\exp(-\Omega(d_v))$. Let $\beta, \beta', \delta, \delta'$ denote sufficiently small constants.

Before we proceed with our tail bounds, we describe a process to construct the random matching of stubs. We are interested in a particular vertex v. Order the stubs such that the d_v v-stubs are in the beginning; the remaining stubs are ordered arbitrarily. We start with the first stub, and match to a uniform random stub (other than itself). We then take the next unmatched stub, according to the order, and match to a uniform random unmatched stub. And so on and so forth. The final connections are clearly dependent, though the choice among unmatched stubs is done independently. This is formalized as follows. Let Y_i be an independent uniform random integer in $[1, 2m-2(i-1)-1]$. This represents the choice at the ith step, since in the ith step, we have exactly $2m - 2(i-1) - 1$ choices. Imagine that we first draw these independent Y_i's. Then we deterministically construct the matching on the basis of these numbers. (So the first stub is connected to the Y_1st stub, the second unmatched stub is connected to the Y_2nd unmatched stub, etc.)

LEMMA 3.2. *Assume $d_n < \sqrt{m}/2$. Let D_v be the random variable denoting the degree of v in the resulting graph. There exist sufficiently small constants $\beta, \beta' \in (0, 1)$, such that $\Pr[D_v < \beta' d_v] < \exp(-\beta d_v)$.*

PROOF. Suppose $d_v > 1$. We again order the stubs so that the d_v v-stubs are in the beginning. Let X_j be the indicator random variable for the jth matching forming a new edge with v. Note that $\sum_{j=1}^{\lfloor d_v/2 \rfloor} X_j \leq D_v$. Observe that X_j is a function of $Y_1, Y_2, \ldots, Y_j$. Consider any $Y_1, Y_2, \ldots, Y_{j-1}$ and suppose the matchings created by these variables link to vertices $v_0 = v, v_1, v_2, \ldots, v_{j-1}$ (distinct) such that there are n_j links to vertex v_j such that $\sum_{i=0}^{j-1} n_i = (j-1)$. Then, for $j = 1, \ldots, \lfloor d_v/2 \rfloor$,

$$\begin{aligned}
& \mathbf{E}[X_j|Y_1, Y_2, \ldots, Y_{j-1}] \\
\geq\ & 1 - \frac{(d_v - j - n_0) + \sum_{1 \leq i \leq j-1; n_i \neq 0}(d_{v_i} - n_i)}{2m - 2(j-1) - 1} \\
\geq\ & 1 - \frac{-2(j-1) - 1 + \sum_{i=0}^{j-1} d_{v_i}}{2m - 2(j-1) - 1} \\
\geq\ & 1 - \frac{\sum_{i=0}^{j-1} d_{v_i}}{2m - 2d_v}.
\end{aligned}$$

Note that $\sum_{i=0}^{j-1} d_{v_i} \leq (\sqrt{m}/2)^2 = m/4$, by the bound on the maximum degree. We also get $2m - 2d_v > m$, so we bound $\mathbf{E}[X_j|Y_1, Y_2, \ldots, Y_{j-1}] \geq 3/4$. By Lem. 3.1 (setting $\delta = 2/3$ and bounding $\alpha k > d_v/4$),

$$\begin{aligned}
\Pr[D_v < d_v/8] &\leq \Pr\left[\sum_{j=1}^{\lfloor d_v/2 \rfloor} X_j < d_v/8\right] \\
&\leq \Pr\left[\sum_{j=1}^{\lfloor d_v/2 \rfloor} X_j < \lfloor d_v/2 \rfloor/2\right] < \exp(-d_v(1/3)^2/8)
\end{aligned}$$

□

This suffices to prove the trivial bound for the trivial algorithm.

CLAIM 3.3. *The expected number of wedges enumerated by the trivial algorithm is $\Omega(\sum_v d_v^2)$.*

PROOF. The expected number of wedges enumerated is $\Omega(\sum_v D_v^2)$, where D_v is the actual degree of v. Using Lem. 3.2, $\mathbf{E}[D_v^2] = \Omega(d_v^2)$. □

‘

We will need the following basic claim about the joint probability of two edges.

CLAIM 3.4. *Let v, w, w' be three distinct vertices. The probability that edges (v, w) and (v, w') are present in the final graph is at most $d_v^2 d_w d_{w'}/m^2$.*

PROOF. Assume $d_v > 1$. Let $C_{v,w}$ be the indicator random variable for edge (v, w) being present (likewise define $C_{v,w'}$). Label the stubs of each vertex as $s_1^v, \ldots, s_{d_v}^v$; $s_1^w, \ldots, s_{d_w}^w$; and $s_1^{w'}, \ldots, s_{d_{w'}}^{w'}$. Let $C_{s_i^v, s_j^w}$ be the indicator random variable for edge being present between stubs s_i^v and s_j^w (likewise define $C_{s_i^v, s_j^{w'}}$). Then the event $\{C_{v,w} C_{v,w'} = 1\}$ that edges (v, w) and (v, w') are present is a subset of the event $\cup_{1 \leq i \neq j \leq d_v} \cup_{k=1}^{d_w} \cup_{\ell=1}^{d_{w'}} \{C_{s_i^v, s_k^w} C_{s_j^v, s_\ell^{w'}} = 1\}$. Hence,

$$\Pr[C_{v,w} C_{v,w'} = 1] \leq \sum_{1 \leq i \neq j \leq d_v} \sum_{k=1}^{d_w} \sum_{\ell=1}^{d_{w'}} \Pr[C_{s_i^v, s_k^w} C_{s_j^v, s_\ell^{w'}} = 1].$$

Fix $1 \le i \ne j \le d_v$, $1 \le k \le d_w$ and $1 \le \ell \le d_{w'}$ and order stubs s_i^v, s_j^v first in the ECM wiring. Then, $\Pr[C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] = \Pr[C_{s_i^v,s_k^w} = 1] \Pr[C_{s_j^v,s_\ell^{w'}} = 1 | C_{s_i^v,s_k^w} = 1]$ where $\Pr[C_{s_i^v,s_k^w} = 1] = [2m-1]^{-1}$ and $\Pr[C_{s_j^v,s_\ell^{w'}} = 1 | C_{s_i^v,s_k^w} = 1] = [2m-3]^{-1}$. Hence,

$$\Pr[C_{v,w} C_{v,w'} = 1] \le d_v(d_v - 1) d_w d_{w'} / m^2$$

using $(2m-1)(2m-3) \ge m^2$ when $m \ge 3$. □

4. GETTING THE 4/3 MOMENT

We will use a series of claims to express the running time of MinBucket in a convenient form. For vertex v, let X_v be the random variable denoting the number of edges in v's bin. The expected running time is at most $\mathbf{E}[\sum_v X_v(X_v - 1)]$. This is because number of wedges in each bin is $\binom{X_v}{2} \le X_v^2 - X_v$.

We further break X_v into the sum $\sum_w Y_{v,w}$, where $Y_{v,w}$ is the indicator for edge (v,w) being in v's bin. As mentioned earlier, $C_{v,w}$ is the indicator for edge (v,w) being present. Note that $Y_{v,w} \le C_{v,w}$, since (v,w) can only be in v's bin if it actually appears as an edge.

We list out some bounds on expectations. Only the second one really uses the binning of MinBucket.

Claim 4.1. *Consider vertices v, w, w' $(w \ne w')$.*

- $\mathbf{E}[Y_{v,w} Y_{v,w'}] \le d_v^2 d_w d_{w'} / m^2$.
- *There exist sufficient small constants $\delta, \delta' \in (0,1)$ such that: if $d_w < \delta d_v$ then* $\mathbf{E}[Y_{v,w} Y_{v,w'}] \le 2\exp(-\delta' d_v) d_v^2 d_w d_{w'} / m^2$.

Proof. We use the trivial bound of $Y_{v,w} Y_{v,w'} \le C_{v,w} C_{v,w'}$. By Claim 3.4, $\mathbf{E}[Y_{v,w} Y_{v,w'}] \le \mathbf{E}[C_{v,w} C_{v,w'}] \le d_v^2 d_w d_{w'} / m^2$.

Now for the interesting bound. The quantity $\mathbf{E}[Y_{v,w} Y_{v,w'}]$ is the probability that both $Y_{v,w}$ and $Y_{v,w'}$ are 1. For this to happen, we definitely require both (v,w) and (v,w') to be present as edges. Call this event $\mathcal{E}$. We also require (at the very least) the degree of v to be at most the degree of w (otherwise the edge (v,w) will not be put in v's bin.) Call this event $\mathcal{F}$. If D_v, D_w denote the degrees of v and w, note that $D_w \le d_w < \delta d_v$, implying event $\mathcal{F}$ is contained in the event $\{D_v < \delta d_v\}$ when $d_w < \delta d_v$. Hence, the event $Y_{v,w} Y_{v,w'} = 1$ is contained in $\mathcal{E} \cap \{D_v < \delta d_v\}$. Assume $d_v > 2, d_w > 0, d_{w'} > 0$ or else $\mathbf{E}[Y_{v,w} Y_{v,w'}] = 0$ trivially when $\delta < 1/2$.

As in the proof of Claim 3.4, let $C_{s_i^v, s_j^w}$ be the indicator random variable for edge being present between stubs s_i^v and s_j^w of vertices v, w (and analogously define $C_{s_i^v, s_j^{w'}}$). Then $\mathcal{E}$ is contained in $\cup_{1 \le i \ne j \le d_v} \cup_{k=1}^{d_w} \cup_{\ell=1}^{d_{w'}} \{C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1\}$ so that

$$\begin{aligned}
& \Pr[Y_{v,w} Y_{v,w'} = 1] \\
\le\ & \Pr[\mathcal{E}, D_v < \delta d_v] \\
\le\ & \sum_{1 \le i \ne j \le d_v} \sum_{k=1}^{d_w} \sum_{\ell=1}^{d_{w'}} \Pr[C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1, D_v < \delta d_v] \\
=\ & \sum_{1 \le i \ne j \le d_v} \sum_{k=1}^{d_w} \sum_{\ell=1}^{d_{w'}} \Pr[C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] \times \\
& \qquad \Pr[D_v < \delta d_v | C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1].
\end{aligned}$$

Given fixed values of i, j, k, ℓ and order stubs s_i^v, s_j^v first in the ECM wiring. Then, $\Pr[C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] \le m^{-2}$ as in the proof of Claim 3.4. Additionally, conditioned on $C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1$, the remaining stubs form an ECM with respect to a new degree sequence formed by replacing $2m, d_v, d_w, d_{w'}$ in the original degree sequence by $2\tilde{m} = 2m - 4, d_v - 2, d_w - 1, d_{w'} - 1$. Let $\tilde{D}_v$ denote the degree of v in the final graph from the new degree sequence. Then, conditioned on $C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1$, $D_v = 2 + \tilde{D}_v$ so that conditional probability is bounded by

$$\begin{aligned}
& \Pr[D_v < \delta d_v | C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] \\
=\ & \Pr[\tilde{D}_v < \delta d_v - 2 | C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] \\
\le\ & \Pr[\tilde{D}_v < \delta (d_v - 2) | C_{s_i^v,s_k^w} C_{s_j^v,s_\ell^{w'}} = 1] \\
\le\ & 2\exp(-\delta' d_v)
\end{aligned}$$

since $\delta < 1$. That is, Lem. 3.2 applies to $\tilde{D}_v$ with respect to the new degree sequence where v has degree $d_v - 2$ and each degree in this new sequence is less than $\sqrt{\tilde{m}}/2$ by assumption. The bound $\Pr[Y_{v,w} Y_{v,w'} = 1] \le 2\exp(-\delta' d_v) d_v^2 d_w d_{w'} / m^2$ then follows. □

Armed with these facts, we can bound the expected number of wedges contained in a single bucket.

Lemma 4.2.

$$\mathbf{E}[X_v(X_v - 1)] = O\left(1 + \left(\frac{d_v}{m} \times \sum_{w : d_w \ge \delta d_v} d_w\right)^2\right)$$

.

Proof. We will write out

$$X_v^2 = (\sum_w Y_{v,w})^2 = \sum_w Y_{v,w}^2 + \sum_w \sum_{w' \ne w} Y_{v,w} Y_{v,w'}$$

where $\sum_w Y_{v,w}^2 = \sum_w Y_{v,w} = X_v$ as each $Y_{v,w}$ is a 0-1 variable. Hence,

$$\begin{aligned}
& \mathbf{E}[X_v(X_v - 1)] \\
=\ & \sum_w \sum_{w' \ne w} \mathbf{E}[Y_{v,w} Y_{v,w'}] \\
\le\ & \sum_{\substack{w: \\ d_w \ge \delta d_v}} \sum_{\substack{w \ne w': \\ d_{w'} \ge \delta d_v}} \mathbf{E}[Y_{v,w} Y_{v,w'}] + \sum_{\substack{w: \\ d_w < \delta d_v}} \sum_{w' \ne w} \mathbf{E}[Y_{v,w} Y_{v,w'}] \\
& + \sum_{\substack{w': \\ d_{w'} < \delta d_v}} \sum_{w \ne w'} \mathbf{E}[Y_{v,w} Y_{v,w'}] \\
=\ & \sum_{\substack{w: \\ d_w \ge \delta d_v}} \sum_{\substack{w \ne w': \\ d_{w'} \ge \delta d_v}} \mathbf{E}[Y_{v,w} Y_{v,w'}] + 2 \sum_{\substack{w: \\ d_w < \delta d_v}} \sum_{w' \ne w} \mathbf{E}[Y_{v,w} Y_{v,w'}] \\
\le\ & \frac{d_v^2}{m^2} \Big(\sum_{w : d_w \ge \delta d_v} d_w \Big)^2 + 2 \sum_{\substack{w: \\ d_w < \delta d_v}} \sum_{w' \ne w} \mathbf{E}[Y_{v,w} Y_{v,w'}]
\end{aligned}$$

by splitting the sums into cases, $d_w \ge \delta d_v$ and $d_w < \delta d_v$, and using the trivial bound of Claim 4.1 for the first quantity.

We satisfy the conditions to use the second part of Claim 4.1.

$$\begin{aligned}\sum_{\substack{w:\\ d_w<\delta d_v}} \sum_{w'\neq w} \mathbf{E}[Y_{v,w}Y_{v,w'}] &\leq 2 \sum_{\substack{w:\\ d_w<\delta d_v}} \sum_{w'\neq w} \exp(-\delta d_v) d_v^2 d_w d_{w'}/m^2 \\ &\leq 8m^{-1}\exp(-\delta d_v)d_v^2,\end{aligned}$$

where $\sum_{i=1}^n d_i = 2m$, The latter is a decreasing function of d_v (for sufficiently large d_v) and is $O(1)$, completing the proof. □

With this bound for $\mathbf{E}[X_v(X_v-1)]$, we are ready to prove Thm. 1.1.

THEOREM 4.3. $\mathbf{E}[\sum_v X_v(X_v-1)] = O(n+m^{-2}(\sum_{i=1}^n d_i^{4/3})^3)$.

PROOF. We use linearity of expectation and sum the bound in Lem. 4.2 as

$$\begin{aligned}&\mathbf{E}[\sum_v X_v(X_v-1)] \\ \lessdot\ & n + m^{-2}\sum_v d_v^2 (\sum_{w:d_w\geq\delta d_v} d_w)^2 \\ =\ & n + m^{-2}\sum_v \sum_{w:d_w\geq \delta d_v} \sum_{w':d_{w'}\geq\delta d_v} d_v^2 d_w d_{w'}\end{aligned}$$

This is the moment where the 4/3 moment will appear. Since $d_w \geq \delta d_v$ and $d_{w'} \geq \delta d_v$, $d_v^{2/3} \leq \delta^{-2/3} d_w^{1/3} d_{w'}^{1/3}$. Therefore, $d_v^2 d_w d_{w'} = d_v^{4/3} d_v^{2/3} d_w d_{w'} \leq \delta^{-2/3}(d_v d_w d_{w'})^{4/3}$. Wrapping it up,

$$\begin{aligned}&m^{-2}\sum_v \sum_{w:d_w\geq\delta d_v} \sum_{w':d_{w'}\geq\delta d_v} d_v^2 d_w d_{w'} \\ \lessdot\ & m^{-2}\sum_v \sum_{w:d_w\geq\delta d_v} \sum_{w':d_{w'}\geq\delta d_v} (d_v d_w d_{w'})^{4/3} \\ \lessdot\ & m^{-2}(\sum_v d_v^{4/3})^3.\end{aligned}$$

□

5. PROVING TIGHTNESS

We show that the bound achieved by Thm. 1.1 is tight for power laws with $\alpha > 2$. This shows that the bounds given in the proof of Cor. 1.3 are tight. The proof, as expected, goes by reversing most of the inequalities given earlier. For convenience, we will assume for the lower bound that $d_n < \sqrt{m}/4$, instead of the $\sqrt{m}/2$ used for the upper bound. This makes for cleaner technical arguments (we could just as well prove it for $\sqrt{m}/2$, at the cost of more pain).

CLAIM 5.1. *Let* $\mathbf{d}$ *be a power law degree sequence with* $\alpha \in (2, 7/3)$ *with* $d_n < \sqrt{m}/4$. *Then the expected number of wedges enumerated by* MINBUCKET *over* $ECM(\mathbf{d})$ *is* $\Omega(nd_n^{7-3\alpha})$.

We need a technical claim give a lower bound for probabilities of edges falling in a bucket.

CLAIM 5.2. *Let* $d_v > 3$. *Consider vertices* v, w, w' ($w \neq w'$) *and let* c *be a sufficiently large constant. If* $\min(d_w, d_{w'}) > cd_v$, *then* $\mathbf{E}[Y_{v,w}Y_{v,w'}] = \Omega(d_v^2 d_w d_{w'}/m^2)$.

PROOF. The random variable $Y_{v,w}Y_{v,w'}$ is 1 if (v,w), (v,w') are edges and the degrees of w and w' are less than that of v. As before, we will start the matching process by matching stubs of v. We partition the stubs into two groups denoted by B_w and $B_{w'}$, and start by matching stubs in B_w. We set $|B_w| = \lfloor d_v/3 \rfloor$. What is the probability that a stub in B_w connects with a w-stub? This is at least $1-(1-d_w/2m)^{\lfloor d_v/3\rfloor} = \Omega(d_v d_w/m)$.

Condition on any matching of the stubs in B_w. What is the probability that a stub in $B_{w'}$ matches with a w'-stub? Since $\min(|B_{w'}|, d_{w'}) \geq 2|B_w|$, this probability is at least $1-(1-d_{w'}/4m)^{\lfloor d_v/3\rfloor} = \Omega(d_v d_{w'}/m)$.

Now condition on any matching of the v-stubs. The number of unmatched stubs connected to w is at least $d_w/2$ (similarly for w'). The remaining stubs connect according to a standard configuration model. For the remaining degree sequence, the total number of stubs is $2\tilde{m} = 2m - 2d_v$. For sufficiently large m, $d_n \leq \sqrt{m}/4 \leq \sqrt{\tilde{m}}/2$. Hence, we can use Lem. 3.2 (and a union bound) to argue that the probability that the final degrees of w and w' are at least d_v is $\Omega(1)$. Multiplying all the bounds together, the probability $Y_{v,w}Y_{v,w'} = 1$ is $\Omega(d_v^2 d_w d_{w'}/m^2)$. □

We prove Claim 5.1.

PROOF. Note that when $\alpha > 2$, then $m = O(n)$. We start with the arguments in the proof of Lem. 4.2. Applying Claim 5.2 for vertex v such that $d_v > 3$,

$$\begin{aligned}\mathbf{E}[X_v(X_v-1)] &= \sum_w \sum_{w'\neq w} \mathbf{E}[Y_{v,w}Y_{v,w'}] \\ &\geq \sum_{\substack{w:\\ d_w\geq cd_v}} \sum_{\substack{w\neq w':\\ d_{w'}\geq cd_v}} \mathbf{E}[Y_{v,w}Y_{v,w'}] \\ &\gg m^{-2}d_v^2 \sum_{\substack{w:\\ d_w\geq cd_v}} \sum_{\substack{w\neq w':\\ d_{w'}\geq cd_v}} d_w d_{w'} \\ &\geq m^{-2}d_v^2(\sum_{\substack{w:\\ d_w\geq cd_v}} d_w)^2 - m^{-2}d_v^2\sum_w d_w^2\end{aligned}$$

The latter part, summed over all v is at most

$$m^{-2}(\sum_v d_v^2)^2 \leq m^{-2}(\max_v d_v \sum_v d_v)^2 \lessdot m$$

Now we focus on the former part. Choose v so that $cd_v \leq d_n/2$, and let 2^r be the largest power of 2 greater than cd_v. (Note that $r \leq \log_2 d_n - 1$.) We bound $\sum_{w:d_w\geq cd_v} d_w \geq \sum_{w:d_w\geq 2^r} d_w \gg \sum_{k=r}^{\log_2 d_n - 1} 2^k n/2^{k(\alpha-1)}$. This is $\sum_{k=r}^{\log_2 d_n -1} n/2^{k(\alpha-2)}$ which is convergent when $\alpha > 2$. Hence, it is at least $\Omega(n2^{-r(\alpha-2)}) = \Omega(nd_v^{-(\alpha-2)})$.

We sum over all (appropriate v).

$$\begin{aligned}&\sum_{v:3<d_v\leq d_n/2c} m^{-2}d_v^2(\sum_{\substack{w:\\ d_w\geq cd_v}} d_w)^2 \\ \gg\ & (n/m)^2 \sum_{v:3<d_v\leq d_n/2c} d_v^2 d_v^{-(2\alpha-4)} \\ =\ & (n/m)^2 \sum_{v:3<d_v\leq d_n/2c} d_v^{6-2\alpha} \gg (n/m)^2 \sum_{k=2}^{\lfloor \log_2 n - \log_2(2c)\rfloor} n2^{k(7-3\alpha)}\end{aligned}$$

When $\alpha < 7/3$, the sum is divergent. Noting that $m = \Theta(n)$, we bound by $\Omega(nd_n^{7-3\alpha})$. Overall, we lower bound the running time MinBucket by $\sum_{v:3<d_v\le d_n/2c}\mathbf{E}[X_v(X_v-1)]$, which is $\Omega(nd_n^{7-3\alpha}-m)$. For $\alpha<7/3$, this is $\Omega(nd_n^{7-3\alpha})$, matching the upper bound in Cor. 1.3. □

6. THE RUNNING TIME OF MinBucket FOR CHUNG-LU GRAPHS

Theorem 6.1. *Consider a Chung-Lu graph distribution with n vertices over a degree distribution $f_1, f_2, \ldots, f_n$. The expected running time of* MinBucket *is given by $O(m + n(\sum_v {d_v}^{4/3})^3)$.*

We remind the reader that the Chung-Lu (CL) model involves inserting edge (i,j) with probability $d_i d_j/2m$ for all unordered pairs (i,j). We need to prove Claim 4.1 for the Chung-Lu model. Thm. 6.1 will then follow directly using the arguments in §4.

We first state Bernstein's inequality.

Theorem 6.2. *[Bernstein's inequality] Let $X_1, X_2, \ldots, X_k$ be zero-mean independent random variables. Suppose $|X_i| \le M$ almost surely. Then for all positive t,*

$$\Pr[\sum_{i=1}^{k} X_i > t] \le \exp\Big(-\frac{t^2/2}{\sum_i \mathbf{E}[X_i^2] + Mt/3}\Big)$$

We now prove some tail bounds about degrees of vertices. The basic form of these statements is the probability that degree of vertex v deviates by a constant factor of d_v is $\exp(-\Omega(d_v))$. We state in terms of conditional events for easier application later. We use β to denote a sufficiently small constant.

Claim 6.3. *Let $d \ge 2$. Suppose v is a vertex such that $d_v \le d$ and e, e' be two pairs. Let $\mathcal{E}$ be the event that e, e' are present, and D_v be the random variable denoting the degree of v. For sufficiently small constant β,*

$$\Pr[D_v > 3d|\mathcal{E}] < \exp(-\beta d)$$

Proof. All edges are inserted independently. So the occurrence of edge $e'' \ne e, e'$ is completely independent of $\mathcal{E}$. Let $\delta(v)$ be the set of all pairs involving v and $\hat{\delta}(v) = \delta(v) \setminus \{e, e'\}$. We express $D_v = \sum_{h\in\delta(v)} C_h$, where C_h is the indicator random variable for edge h being present. Let $\hat{D}_v = \sum_{h\in\hat{\delta_v}} C_h$. Note that $\mathbf{E}[\hat{D}_v] \le \mathbf{E}[D_v] = d_v \le d$. Set $C'_h = C_h - \mathbf{E}[C_h]$, so

$$\Pr[\hat{D}_v - \mathbf{E}[\hat{D}_v] > d] = \Pr[\sum_{h\in\hat{\delta}(v)} (C_h - \mathbf{E}[C_h]) > d] = \Pr[\sum_{h\in\hat{\delta}(v)} C'_h > d]$$

We wish to apply Bernstein's inequality to the C'_h random variables. Observe that $\mathbf{E}[C'_h] = 0$, and $|C'_h| \le 1$. Setting $\mathbf{E}[C_h] = \mu$, note that

$$\mathbf{E}[(C'_h)^2] = \mathbf{E}[(C_h - \mu)^2] = \mathbf{E}[C_h^2] - \mu\mathbf{E}[C_h] + \mu^2 = \mathbf{E}[C_h].$$

So $\sum_{h\in\hat{\delta}(v)} \mathbf{E}[(C'_h)^2] = \sum_{h\in\hat{\delta}(v)} \mathbf{E}[C_h] = \mathbf{E}[\hat{D}_v] \le d$. By Bernstein's inequality (Thm. 6.2),

$$\begin{aligned}\Pr[\hat{D}_v - \mathbf{E}[\hat{D}_v] > d] &= \Pr[\sum_{h\in\hat{\delta}(v)} C'_h > d] \\ &\le \exp\Big(-\frac{d^2/2}{\sum_{h\in\hat{\delta}(v)} \mathbf{E}[(C'_h)^2] + d/3}\Big) \\ &\le \exp\Big(-\frac{d^2/2}{d + d/3}\Big) = \exp(-3d/8)\end{aligned}$$

None of these random variables depend on the event $\mathcal{E}$, so we get that $\Pr[\hat{D}_v - \mathbf{E}[\hat{D}_v] > d \mid \mathcal{E}] \le \exp(-3d/8)$. Suppose $\hat{D}_v \le \mathbf{E}[\hat{D}_v] + d \le 2d$. We always have $D_v \le \hat{D}_v + 2$ and hence $D_v \le 3d$ (using the bound that $d \ge 2$). Hence, $\Pr[D_v > 3d|\mathcal{E}] < \exp(-3d/8)$. We only require $\beta < 3/8$. □

Claim 6.4. *Suppose v is a vertex such that $d_v \ge 4$ and e, e' be two pairs. Let $\mathcal{E}$ be the event that e, e' are present, and D_v be the random variable denoting the degree of v. For sufficiently small constant β,*

$$\Pr[D_v < d_v/3|\mathcal{E}] < \exp(-\beta d_v)$$

Proof. This proof is almost identical to the previous one. Again, we express $D_v = \sum_{h\in\delta(v)} C_h$, where C_h is the indicator random variable for edge h being present. Let $\hat{D}_v = \sum_{h\in\hat{\delta_v}} C_h$. We have $\hat{D}_v \ge D_v - 2$, so $\mathbf{E}[\hat{D}_v] \ge \mathbf{E}[D_v] - d_v/2 = d_v/2$ (using the bound $d_v \ge 4$). Applying a multiplicative Chernoff bound to $\hat{D}_v$,

$$\Pr[\hat{D}_v < 2\mathbf{E}[\hat{D}_v]/3] < \exp(-d_v/36)$$

Since $\hat{D}_v$ is completely independent of $\mathcal{E}$, we can condition on $\mathcal{E}$ to get the same bound. Suppose $D_v < d_v/3$. Since $D_v \ge \hat{D}_v$ and $d_v \le 2\mathbf{E}[\hat{D}_v]$, we get $\hat{D}_v < 2\mathbf{E}[\hat{D}_v]/3$. So the even $D_v < d_v/3|\mathcal{E}$ is contained in $\hat{D}_v < 2\mathbf{E}[\hat{D}_v]/3|\mathcal{E}$, completing the proof. We require $\beta < 1/36$. □

Finally, we need a simple claim about the second moment of sums of independent random variables.

Claim 6.5. *Let $X = \sum_i X_i$ be a sum of independent positive random variables with $X_i = O(1)$ for all i and $\mathbf{E}[X] = O(1)$. Then $\mathbf{E}[X^2] = O(1)$.*

Proof. By linearity of expectation,

$$\begin{aligned}\mathbf{E}\left[X^2\right] = \mathbf{E}\Big[\big(\sum_i X_i\big)^2\Big] &= \sum_i \mathbf{E}\left[X_i^2\right] + 2\sum_{i<j}\mathbf{E}[X_i]\mathbf{E}[X_j] \\ &\le \sum_i O\left(\mathbf{E}[X_i]\right) + \Big(\sum_i \mathbf{E}[X_i]\Big)^2 = O(1).\end{aligned}$$

□

We prove the analogue of Claim 4.1.

Claim 6.6. *Consider vertices v, w, w' $(w \ne w')$.*

- *If $d_v \le 4$, then $\mathbf{E}[X_v^2] = O(1)$.*
- *$\mathbf{E}[Y_{v,w}Y_{v,w'}] \le d_v^2 d_w d_{w'}/4m^2$.*
- *If $d_w \le d_v/10$ and $d_v \ge 4$, then $\mathbf{E}[Y_{v,w}Y_{v,w'}] \le 2\exp(-\beta d_v) d_v^2 d_w d_{w'}/4m^2$.*

Proof. Defining $\hat{X}_v = \sum_w C_{v,w}$, we have $X_v \le \hat{X}_v$. Since these are all positive random variables, $X_v^2 \le \hat{X}_v^2$. Applying Claim 6.5, $\mathbf{E}[\hat{X}_v^2] = O(1)$. That completes the first part.

For the second part, we use the trivial bound of $Y_{v,w}Y_{v,w'} \le C_{v,w}C_{v,w'}$. Taking expectations and using independence, $\mathbf{E}[Y_{v,w}Y_{v,w'}] \le C_{v,w}C_{v,w'} = d_v^2 d_w d_{w'}/4m^2$.

The third case is really the interesting one. The quantity $\mathbf{E}[Y_{v,w}Y_{v,w'}]$ is the probability that both $Y_{v,w}$ and $Y_{v,w'}$ are 1. For this to happen, we definitely required both (v,w) and (v,w') to be present as edges. Call this event $\mathcal{E}$. We also require (at the very least) the degree of v to be at most the degree of w (otherwise the edge (v, w) will not be put in v's bin.) Call this event $\mathcal{F}$. The event $Y_{v,w}Y_{v,w'} = 1$ is contained in $\mathcal{E} \cap \mathcal{F}$. Using conditional probabilities, $\Pr(\mathcal{E} \cap \mathcal{F}) = \Pr(\mathcal{F}|\mathcal{E}) \Pr(\mathcal{E})$. Note that $\Pr(\mathcal{E}) = d_v^2 d_w d_{w'}/4m^2$.

Let D_v, D_w denote the degrees of v and w. Let $\mathcal{F}_v$ denote the event $D_v < d_v/3$ and $\mathcal{F}_w$ denote event $D_w > 3d_v/10$. If neither of these events happens, then $D_w \leq 3d_v/10 < d_v/3 \leq D_v$. So $\mathcal{F}$ cannot happen. Hence, $(\mathcal{F}|\mathcal{E})$ is contained in $(\mathcal{F}_v \cup \mathcal{F}_w|\mathcal{E})$. By the union bound, $\Pr(\mathcal{F}_v \cup \mathcal{F}_w|\mathcal{E}) \leq \Pr(\mathcal{F}_v|\mathcal{E}) + \Pr(\mathcal{F}_w|\mathcal{E})$. Applying Claim 6.4 to the latter and Claim 6.3 to the former, we bound $\Pr(\mathcal{F}|\mathcal{E}) \leq 2\exp(-\beta d_v)$. □

As mentioned earlier, we can now execute the arguments in §4 to prove Thm. 6.1.

7. A MORE CAREFUL ANALYSIS

Because we are interested in practical performance, we performed a more careful analysis to find stronger bounds on the performance of MinBucket, trying to understand the size of the constants in the big-Oh notation. We consider a slightly different ECM distribution. Rather than starting with a given degree sequence, we draw the degree for each vertex independently at random from a reference degree distribution f. Specifically, $f(d)$ is the probability that a node draws degree value d, for d an integer in $[0, \infty)$. After nodes draw degree values, the rest of the ECM construction proceeds as described in §1.1.

Our analysis assumes that when an edge joins two vertices of the same degree, the edge is placed in the bucket for both edges. Thus we slightly overcount the work for MinBucket. Let $X_{v=i,n}$ be the size of the bucket for an arbitrary node i in a graph generated by ECM with n nodes. We wish to bound the expected triangle-searching work $\mathbf{E}[\sum_{i=1}^{n} \binom{X_{i,n}}{2}]$ in an ECM graph, as the number of nodes $n \to \infty$. We denote the rth moment, $r > 0$, of the reference degree distribution f as $\mathbf{E}[D^r] = \sum_{d=0}^{\infty} d^r \cdot f(d)$.

The following theorem gives explicit expressions for the size of each bucket and the total work for MinBucket. The proof is quite involved and uses many measure-theoretic arguments. We provide it in the full version. Given a probability distribution $f(d)$, and a value n, the expression in this theorem computes a strong upper bound on the ratio between the work and n. We focus on linear regimes, where this is the constant in the running time.

Theorem 7.1. *Under the ECM for generating random graphs (cf.§1.1), suppose the reference degree distribution f has a finite mean $\mathbf{E}[D] = \sum_{d=0}^{\infty} d \cdot f(d) \in (0, \infty)$ and $\mathbf{E}[D^{4/3}] \in (0, \infty)$. As $n \to \infty$, $\frac{1}{n}\mathbf{E}\left[\sum_{i=1}^{n} \binom{X_{i,n}}{2}\right]$ (the expected triangle-searching work per bucket) tends to*

$$\frac{1}{2(\mathbf{E}[D])^2} \sum_{d_1=0}^{\infty} d_1 f(d_1) \Big[\sum_{d_2=d_1}^{\infty} d_2 f(d_2) \Big]^2 \in (0, \infty).$$

7.1 Experimental Analysis

We experimentally show the bound of Thm. 7.1 does captures the expected performance of MinBucket on ECM graphs. We generated ECM graphs of various sizes based on a power-law degree distribution with power exponent $\alpha = 2.4 > 7/3$ (which guarantees a finite $\frac{4}{3}$ moment for the degree distribution). Figure 1(a) shows the average value of $\sum_{i=1}^{n} \binom{X_{i,n}}{2}$, total work over all buckets, computed over 10 Monte Carlo trials (i.e., taken as an approximation of $\mathbf{E}[\sum_{i=1}^{n} \binom{X_{i,n}}{2}]$) for ECM graphs of various sizes up to $n = 80$ million. Degrees are truncated at $\sqrt{n}$. The ECM uses power law reference degree distributions for $\alpha = 2.3$, where MinBucket runs in superlinear time, and for $\alpha = 2.4$, where it runs in linear time. Figure 1(a) also shows the theoretical linear bound on the overall expected work $\mathbf{E}[\sum_{i=1}^{n} \binom{X_{i,n}}{2}]$ for a power-law degree distribution with $\alpha = 2.4$. The constant we get from Thm. 7.1 is approximately $C = 0.687935$. As $n \to \infty$, we would anticipate from Thm. 7.1 that the value of $\mathbf{E}[\sum_{i=1}^{n} \binom{X_{i,n}}{2}]$ approximated by Monte Carlo trials should approach nC, also shown in the left part of Fig. 1. The right part of Fig. 1 shows the ratio of work to number of nodes n. For power law distributions with $\alpha = 2$, this ratio is not a constant. But by $\alpha = 2.4$, the factor is leveling off below 1.

Acknowledgments

This work was funded under the Sandia National Laboratories Laboratory Directed Research and Development (LDRD) program. We thank Ali Pinar for suggestions on improving the presentation.

8. REFERENCES

[ACL01] W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10:53–66, 2001.

[AYZ97] N. Alon, R. Yuster, and U. Zwick. Finding and counting given length cycles. *Algorithmica*, 17:354–364, 1997.

[BA99] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, October 1999.

[BC78] E. A. Bender and E.R. Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory A*, 24:296–307, 1978.

[BDML06] T. Britton, M. Deijfen, and A. Martin-Löf. Generating simple random graphs with prescribed degree distribution. *Journal of Statistical Physics*, 124(6), September 2006.

[BHLP11] J. W. Berry, B. Hendrickson, R. A. LaViolette, and C. A. Phillips. Tolerating the Community Detection Resolution Limit with Edge Weighting. *Physical Review E*, 83(5), May 2011.

[BKM+00] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. *Computer Networks*, 33:309–320, 2000.

[Bol80] B. Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal on Combinatorics*, 1:311–316, 1980.

[Bur04] R. S. Burt. Structural holes and good ideas. *American Journal of Sociology*, 110(2):349–399, 2004.

[Bur07] R. S. Burt. Secondhand brokerage: Evidence on the importance of local structure for managers, bankers, and analysts. *Academy of Management Journal*, 50, 2007.

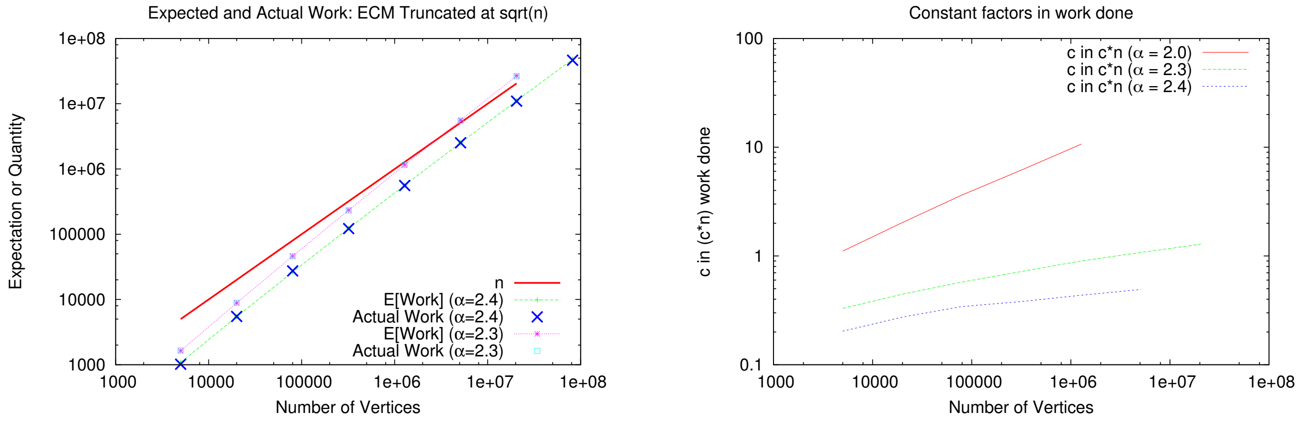

Figure 1: Experimental results with n-node ECM graphs with degrees $\leq n^{1/2}$ drawn from a power-law distribution with exponent $\alpha = 2.3$ or 2.4. The red solid line shows n. The green dashed line shows a theoretical bound $0.687935n$ on the expected number of pairs in buckets for an ECM graph with n vertices and exponent 2.4. The blue crosses shows the average value of pairs observed in 10 generations of ECM graphs. The magenta line and blue boxes show the same for exponent 2.3. (b) Experimental values of the ratio of work to n for power law exponents 2, 2.3, and 2.4.

[CE91] M. Chrobak and D. Eppstein. Planar orientations with low out-degree and compaction of adjacency matrices. *Theoretical Computer Science*, 86:243–266, 1991.

[CL02] F. Chung and L. Lu. The average distances in random graphs with given expected degrees. *PNAS*, 99:15879–15882, 2002.

[CLV03] F. Chung, L. Lu, and V. Vu. Eigenvalues of random power law graphs. *Annals of Combinatorics*, 7:21–33, 2003.

[CN85] N. Chiba and T.Takao Nishizeki. Arboricity and subgraph listing algorithms. *SIAM J. Comput.*, 14:210–223, February 1985.

[Coh09] J. Cohen. Graph twiddling in a MapReduce world. *Computing in Science & Engineering*, 11:29–41, 2009.

[Col88] J. S. Coleman. Social capital in the creation of human capital. *American Journal of Sociology*, 94:S95–S120, 1988.

[FFF99] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *Proceedings of SIGCOMM*, pages 251–262, 1999.

[FH97] I. Fudos and C. M. Hoffmann. A graph-constructive approach to solving systems of geometric constraints. *ACM Transactions on Graphics*, 16(2):179–216, 1997.

[FWVDC10] B. Foucault Welles, A. Van Devender, and N.Noshir Contractor. Is a friend a friend?: Investigating the structure of friendship networks in virtual worlds. In *CHI-EA'10*, pages 4027–4032, 2010.

[IR78] A. Ital and M. Rodeh. Finding a minimum circuit in a graph. *SIAM Journal on Computing*, 7:413–423, 1978.

[Lat08] M. Latapy. Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoretical Computer Science*, 407:458–473, 2008.

[MP02] M. Mihail and C. Papadimitriou. On the eigenvalue power law. In *RANDOM*, pages 254–262, 2002.

[MR95a] M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 6:161–179, 1995.

[MR95b] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[MR98] M. Molloy and B. Reed. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7:295–305, 1998.

[New03] M.E.J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[NSW01] M. E. J. Newman, S. Strogatz, and D. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64:026118, 2001.

[Por98] A. Portes. Social capital: Its origins and applications in modern sociology. *Annual Review of Sociology*, 24(1):1–24, 1998.

[SV11] S. Suri and S. Vassilvitskii. Counting triangles and the curse of the last reducer. In *WWW'11*, pages 607–614, 2011.

[SW05a] T. Schank and D. Wagner. Finding, counting and listing all triangles in large graphs, an experimental study. In *Experimental and*

Efficient Algorithms, pages 606–609. Springer Berlin / Heidelberg, 2005.

[SW05b] T. Schank and D. Wagner. Finding, counting, and listing all triangles in large graphs: an experimental study. *Workshop on Experimental and Efficient Algorithms (WEA)*, 2005.

[Tso08] C. E. Tsourakakis. Fast counting of triangles in large real networks without counting: Algorithms and laws. In *ICDM*, pages 608–617, 2008.

[Wor81] N. C. Wormald. The asymptotic connectivity of labelled regular graphs. *Journal of Combinatorial Theory B*, 31:156–167, 1981.

[WS98] D. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[WW10] V. Vassilevska Williams and R. Williams. Subcubic equivalences between path, matrix and triangle problems. In *Foundations of Computer Science (FOCS)*, pages 645–654, 2010.

APPENDIX

A. PROOF OF TECHNICAL LEMMA

Proof. (of Lem. 3.1) Consider the sequence $X'_1, X'_2, \ldots, X'_k$ of i.i.d. Bernoulli random variables with $\mathbf{E}[X'_i] = \alpha$. We will shortly prove that for any $t > 0$, $\Pr[\sum_{i=1}^k X_i < t] \leq \Pr[\sum_{i=1}^k X'_i < t]$. Given this, we just apply a multiplicative Chernoff bound (Theorem 4.2 of [MR95b]) for $\sum_{i=1}^k X'_i$ with $\mu = \alpha k$. Hence, $\Pr[\sum_{i=1}^k X_i < \alpha k \delta] < \exp(-\alpha(1-\delta)^2/2)$.

For convenience, we show the contrapositive $\Pr[\sum_{i=1}^k X_i \geq t] \geq \Pr[\sum_{i=1}^k X'_i \geq t]$. This is proven by induction on k. First, the base case. Since X_1 and X'_1 are Bernoulli random variables, it suffices to show that $\Pr[X_1 = 1] \geq \Pr[X'_1 = 1] = \alpha$, which holds by assumption.

Now for the induction step. Assume for all $t > 0$ and some index j, $\Pr[\sum_{i=1}^j X_i \geq t] \geq \Pr[\sum_{i=1}^j X'_i \geq t]$. We prove this for $j+1$. Let $\mathcal{E}$ denote the event $\sum_{i=1}^j X_i \geq t$, and $\mathcal{E}'$ be the (disjoint) event $\sum_{i=1}^j X_i \in [t-1, t)$. Let $\mathbb{I}(A)$ denote the indicator function of event A. Because X_i is a 0-1 random variable, we get

$$\begin{aligned}
\Pr\left[\sum_{i=1}^{j+1} X_i \geq t\right] &= \Pr[\mathcal{E}] + \Pr[\mathcal{E}' \wedge (X_{j+1} = 1)] \\
&= \Pr[\mathcal{E}] + \mathbf{E}[\mathbb{I}(\mathcal{E}')\mathbb{I}(X_{j+1} = 1)] \\
& \quad \Pr[\mathcal{E}] + \mathbf{E}\{\mathbf{E}[\mathbb{I}(\mathcal{E}')\mathbb{I}(X_{j+1} = 1) | Y_1, \ldots, Y_j]\}
\end{aligned}$$

Observe that $\sum_{i=1}^j X_i$ only depends on $Y_1, \ldots, Y_j$ so that $\mathbb{I}(\mathcal{E}')$ is a constant in the conditional expectation

$$\begin{aligned}
\mathbf{E}[\mathbb{I}(\mathcal{E}')\mathbb{I}(X_{j+1} = 1) | Y_1, \ldots, Y_j] &= \mathbb{I}(\mathcal{E}')\mathbf{E}[\mathbb{I}(X_{j+1} = 1) | Y_1, \ldots, Y_j] \\
&= \mathbb{I}(\mathcal{E}') \Pr[X_{j+1} = 1 | Y_1, \ldots, Y_j] \\
&\geq \mathbb{I}(\mathcal{E}')\alpha,
\end{aligned}$$

where $\Pr[X_{j+1} = 1 | Y_1, \ldots, Y_j] \geq \alpha$ by the lemma assumption.

Let us denote (for any $s > 0$) $\Pr[\sum_{i=1}^j X_i \geq s]$ by p_s and $\Pr[\sum_{i=1}^j X'_i \geq s]$ by p'_s. The above gives

$$\begin{aligned}
& \Pr\left[\sum_{i=1}^{j+1} X_i \geq t\right] \\
\geq \;& p_t + \alpha \mathbf{E}[\mathbb{I}(\mathcal{E}')] \\
= \;& p_t + (p_{t-1} - p_t)\alpha = p_{t-1}\alpha + p_t(1-\alpha) \\
\geq \;& p'_{t-1}\alpha + p'_t(1-\alpha) \quad \text{(using induction hypothesis and } \alpha \in [0,1]) \\
= \;& p'_t + (p'_{t-1} - p'_t)\alpha \\
= \;& \Pr\left[\sum_{i=1}^{j} X'_i \geq t\right] + \Pr\left[\left(\sum_{i=1}^{j} X'_i \in [t-1, t)\right) \wedge (X'_{j+1} = 1)\right] \\
= \;& \Pr\left[\sum_{i=1}^{j+1} X'_i \geq t\right]
\end{aligned}$$

□