SUBGRAPH METHODS FOR COMPARING COMPLEX NETWORKS

by

Matthew Stephen Hurshman

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
April 2013

DALHOUSIE UNIVERSITY

DEPARTMENT OF MATHEMATICS AND STATISTICS

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled “SUBGRAPH METHODS FOR COMPARING COMPLEX NETWORKS” by Matthew Stephen Hurshman in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Dated: April 3, 2013

External Examiner: ______________________________

Research Supervisor: ______________________________

Examining Committee: ______________________________

____

Departmental Representative: ______________________________

DALHOUSIE UNIVERSITY

DATE: April 3, 2013

AUTHOR: Matthew Stephen Hurshman

TITLE: SUBGRAPH METHODS FOR COMPARING COMPLEX NETWORKS

DEPARTMENT OR SCHOOL: Department of Mathematics and Statistics

DEGREE: Ph.D. CONVOCATION: May YEAR: 2013

Signature of Author

Table of Contents

List of Tables

List of Figures

Abstract

An increasing number of models have been proposed to explain the link structure observed in complex networks. The central problem addressed in this thesis is: how do we select the best model? The model-selection method we implement is based on supervised learning. We train a classifier on six complex network models incorporating various link attachment mechanisms, including preferential attachment, copying and spatial. For the classification we represent graphs as feature vectors, integrating common complex network statistics with raw counts of small connected subgraphs commonly referred to as *graphlets*. The outcome of each experiment strongly indicates that models which incorporate the preferential attachment mechanism fit the network structure of Facebook the best. The experiments also suggest that graphlet structure is better at distinguishing different network models than more traditional complex network statistics.

To further the understanding of our experimental results, we compute the expected number of triangles, 3-paths and 4-cycles which appear in our selected models. This analysis shows that the spatial preferential attachment model generates 3-paths, triangles and 4-cycles in abundance, giving a closer match to the observed network structure of the Facebook networks used in our model selection experiment. The other models generate some of these subgraphs in abundance but not all three at once. In general, we show that our selected models generate vastly different amounts of triangles, 3-paths and 4-cycles, verifying our experimental conclusion that graphlets are distinguishing features of these complex network models.

Key words: complex networks, social networks, machine learning, graph theory

List of Abbreviations and Symbols Used

C_n	the cycle on n vertices, 9
P_n	the path on n vertices, 9
K_n	complete graphs on n vertices, 9
$Aut(G)$	set of all automorphisms for a graph G, 10
$ind(F, G)$	the number of induced subgraphs isomorphic to F contained in G, 11
$inj(F, G)$	the number of injective subgraphs isomorphic to F contained in G, 11
$G \cong H$	G and H are isomorphic, 9
$\mathcal{G}_n$	set of all isomorphism classes on n vertices, 10
$\mathcal{G}_{n,m}$	set of all isomorphism classes on n vertices with m edges, 10

Acknowledgements

I would like to thank Jeannette Janssen for all the direction, support and help she has given me. I would also like to thank the National Research Council of Canada for the financial support which made this thesis possible. I would mostly like to thank my lovely Danielle. Without her love and support, this thesis simply would have not been possible.

Chapter 1

Introduction

1.1 Motivation

It has been readily observed that link-structured information arises naturally in many unrelated contexts in the real world. In many instances a graph, also called a network, is an appropriate model for such phenomena. Since the dawn of the internet age, with computational capacity rapidly advancing, the ability to study these so called *real world* or *complex* networks has increased dramatically. This study points to these networks sharing many common properties such as a power law degree distribution, high level of clustering, small average path lengths, assortativity and disassortativity amongst vertices and community structure. There have been numerous models proposed to replicate the structure of these complex networks. The study of these models has been a steadily advancing area of research [8, 25, 103].

The central question posed in this thesis is: how can you determine which model is the best for a given complex network? In some instances, posing and validating a model can be a straight forward exercise. For example, when Newton first posed his model for gravity, he could easily provide evidence for its accuracy by predicting how objects fall on Earth or the motion of the planets, etc. In this case, what the model should be able to predict is clear. It is not as clear what a complex network model should be able to predict. Many complex networks such as social networks and the web graph are constantly evolving, with vertices and edges being added and deleted all the time. Proposing that the goal of a particular model should be its ability to precisely replicate the given complex network is meaningless. In practice, a particular static instance of a complex network is taken, and models are ranked based on their ability to generate "similar" networks. In Section 1.4 we provide an overview of complex networks including common examples, their properties and some models which have been proposed to replicate them.

Historically, the approach to validating a model for a complex network was to show that the model was capable of replicating the complex network properties which are described in Section 1.4.2. This approach, while leading to many interesting models and techniques, has widely ignored whether or not the models generate graphs which are truly similar to the complex networks they are meant to model. The problem of determining whether or not two graphs are similar is a well-studied problem in its own right [45]. The basic premise of the problem is to develop a similarity measure (sometimes a distance metric) which assigns a score indicating the level of similarity between two graphs. Typical approaches to developing similarity measures have been to compute the maximum common subgraph or minimum common supergraph of the two graphs [47, 113, 122, 46, 65, 44, 43]. A different approach to the graph similarity problem is to use graph kernels. A kernel function is an inner product of two vectors from a feature space. In a graph kernel, each graph is represented as a feature vector which inhabits the feature space, and the similarity score is determined by taking the inner product or graph kernel of the two feature vectors. The feature vectors contain information about the structure of the graph. Graph kernels have been proposed which include all subgraph counts [71], subtree counts [112], cycle counts [73], path counts [38] and graphlets [117].

Techniques for graph similarity often incorporate information about the subgraph structure of the two graphs. It is common to refer to small subgraphs as *graphlets* in the literature [54, 110, 117]. The term *motifs* has also been used but this term refers to induced subgraphs which occur much more frequently than you would expect in a random graph of the same size and density [93]. The goal of this thesis is to explore the role of graphlets in verifying models for complex networks. In placing graphlets at the forefront of the validation procedure, we make a departure from previous approaches which use complex network statistics such as the power law degree distribution and the small world property. Recent work in validating models for complex networks has also started adopting this graphlet perspective. One such paper, which largely inspires our model-selection method in Chapter 3, is the work of Middendorf *et al.* [91]. In this paper, the authors use a supervised learning algorithm to determine which of seven proposed models is most likely to have generated a particular PPI network. Supervised learning is a branch of machine learning in which a classifier learns a set of labelled training data. The training data in this case contains feature vectors whose entries correspond to raw graphlet counts, and the label is the model which the feature vector comes from. Two sets of feature vectors are used: one which incorporates the graphlet counts for all graphlets that can be formed by a walk of length of 8 (148 non-isomorphic graphs), and all graphlets which contain 7 edges (130 non-isomorphic graphs). Once the classifier has been trained, a feature vector computed from a PPI network is evaluated by the classifier to determine which model achieves the highest score. The conclusion of this work is that models which incorporate the copying mechanism are most likely to have generated the PPI. This conclusion corresponds to the biological fact that proteins copy the functions of other proteins. In Chapter 3 we incorporate a similar approach based on supervised learning to determine which model can best replicate network data taken from Facebook.

Additional work in model validation of PPIs using graphlets includes the work of Prẑulj [54, 110]. Also in [33], an unsupervised learning algorithm using graphlet-based feature vectors is used to cluster data taken from a variety of complex networks. The idea is to see whether or not data taken from the same type of complex network, like a social network, gets clustered together.

What is the advantage of using graphlets over complex network statistics in validating models for complex networks? Graphlets are the fundamental building blocks for a graph. Suppose that we know nothing about a particular graph G and we are incrementally given knowledge of the graphlets. We are first given the graphlets of size 1, consisting of a set of n isolated vertices. From this, we deduce the size of the graph is n. Next, we are shown the graphlets of size 2; from which, we can deduce the number of edges. Next, we are given the graphlets of size 3; the number of 3-paths and triangles. From this, we begin to understand the amount of clustering present in the graph. As we are incrementally shown more graphlets of increasing size, our knowledge of G steadily increases culminating in the final reveal of the one graphlet of size n, G itself. The use of complex network statistics does not allow for this possibility of refinement in our understanding of G. Furthermore, we argue that knowledge of only relatively small graphlets provide more information about the graph than complex network statistics. In Chapter 3, we demonstrate this experimentally by showing that our classification algorithm is more accurate at identifying the correct model when it is built using feature vectors incorporating counts for the graphlets of size 3 and 4 than feature vectors which incorporate information about complex network statistics alone. As graphlets serve as building blocks of a graph, we suspect that complex network statistics themselves may be implicitly contained in graphlet counts. This is clearly the case for the clustering coefficient (see Equation 1.4 on page 25) We also show this is the case for the power law coefficient (see Theorem 2.1.2 on page 40.)

Another important advantage is that by using graphlets, one can naturally represent graphs as vector based data, thus opening up a wide variety of machine learning algorithms which are designed for vectors [24]. These algorithms run very efficiently while algorithms which compute graph similarity based on graph structure tend to be NP-Hard and are prohibitively expensive to compute [42]. Unfortunately, as we experience first hand in Chapter 3, algorithms for counting graphlets are also expensive to compute. Fortunately, faster algorithms are currently being developed to speed up graphlet counting [79, 66].

The use of graphlets can also be justified from a theoretical point of view. Originally proposed by S. Ulam and P. Kelly [80], the Reconstruction Conjecture states that if G is an undirected graph on $n \geq 3$ vertices then G is uniquely determined by its vertex-deleted subgraphs. A vertex-deleted subgraph is obtained by deleting a single vertex and its incident edges from the graph. In other words, the graphlets of size $n-1$. While the conjecture is outstanding, it has been proved for several classes of graphs: regular graphs [80], trees [80], disconnected graphs [80], threshold graphs [115] and unit interval graphs [115]. The Reconstruction Conjecture has also been verified for all graphs on at most 11 vertices [90]. Though confirmation of the conjecture has not yet occurred, it has generated much interest in the mathematical community with well over 300 papers being published on the Conjecture. These papers deal with related questions such as: how much information can be determined about G from its set of vertex-deleted subgraphs? A fascinating result by Bollobás [26] shows that almost every graph is uniquely determined by only 3 of its vertex-deleted subgraphs. Note that the result is for a specific set of 3 vertex-deleted subgraphs, not any random set of 3.

A current hot topic which which is related to graphlets is *graph limits*. The recent development of models for complex networks has motivated the study of graph sequences (G_n) with $|V(G_n)| \to \infty$. A natural question arises: what does it mean for such a sequence to converge and what is the limit of a convergent sequence? To define a notion of convergence we must count the number of edge preserving maps, or homomorphisms, from F into G. For two simple graphs F and G, we denote the number of homomorphisms of F into G by $hom(F, G)$ and denote the **homomorphism density** of F into G by

$$t(F, G) = \frac{hom(F, G)}{|V(G)|^{|V(F)|}}.$$

Note that the homomorphism density gives the probability that a random mapping $V(F) \to V(G)$ is a homomorphism. We say that (G_n) converges if for every simple graph F, the sequence $(t(F, G_n))$ converges. Through the works of [35, 36, 37, 34] a complete theory for dense graph sequences (i.e. $|E(G_n)| = \Theta(n^2)$) has emerged. Unfortunately, much of the developed theory falls apart for sparse graphs (i.e. $|E(G_n)| = \Theta(n)$.) The difficulty of establishing an appropriate theory for sparse graphs is due to the struggle to find an appropriate normalization constant for homomorphism densities in sparse graphs. There have been some attempts to develop a theory of graph limits for sparse graphs [29]. As it happens, most real world network models generate sparse graphs. Though the current theory of graph limits may be of limited use in furthering our understanding of complex network models, its prominence in the mathematical community does provide additional validation of the graphlet perspective.

1.2 Outline and Contributions of this Thesis

We begin with a quick overview of the terms and notation used in this thesis in Section 1.3. We follow with a short review of real world networks, common complex network properties and the models used to replicate them.

Given that the graphlet perspective is a relatively new way to think about model-validation, the number of occurrences of specific graphlets in many complex network models has not been analyzed. Notable exceptions include computing the expected number of injective copies of triangles and 3-paths in the preferential attachment model [59, 116] and the expected number of induced subgraphs in the Random Geometric Model [127, 75, 15]. In Chapter 2 we focus on computing the expected number of triangles, 3-paths and 4-cycles in our selected models. Our focus on these particular graphlets is motivated by their importance in the model selection experiment in Chapter 3. In our analysis, we consider the case where the graphs have dn edges for some positive integer d. Our focus is primarily on the rate at which the expected number of these graphlets grow as the size of the graph tends to infinity. The conclusion of our analysis is that each of the models generate vastly different frequencies of small graphlets. Many of the graphlet counts computed in Chapter 2 verify the observations made during the model selection experiment in Chapter 3. In particular, it was observed that the SPA model is the only model we study which generates triangles, 3-paths and 4-cycles in abundance. This observation is through calculation in Chapter 2. For graphs with an expected power law degree distribution, we pose a relationship between the power law coefficient γ and the number of induced triangles and 3-paths in Theorem 2.1.2. An important conclusion of this theorem is that the amount of clustering present in a graph with an expected power law degree distribution cannot be determined from the power law coefficient. We also pose a relationship in Theorem 2.1.1 which relates the power law coefficient to the number of edges in the graph.

In Chapter 3, we perform a model-selection experiment to determine which of our selected models is most likely to generate an online social network. The data used is from Facebook and was obtained from the data sets of [96]. Our conclusion is that models which incorporate the preferential attachment mechanism perform best. This coincides with the conventional wisdom that preferential attachment is an important mechanism in social networks. Our experiment also show that graphlets alone are sufficient in differentiating our selected models. We also show that graphlets are better at distinguishing our selected models than our chosen complex network statistics. The work contained in Chapter 3 has been published in Internet Mathematics [74].

1.3 Mathematical Background

1.3.1 Graph Theory

We take the time to provide a comprehensive explanation of the graph theory terms which will be used throughout this thesis. Note that *graph* and *network* are terms which refer to the same thing. Both terms are common place and we use both in this thesis.

A **graph** G is a set $V(G)$ of **vertices** along with a set $E(G)$ of 2-element subsets of $V(G)$ called **edges**. We can think of an edge in $E(G)$ as a line connecting two vertices $u, v \in V(G)$. The edge e connecting vertices u and v will be denoted $e = uv = vu$. We say that e is **incident** to the vertices u and v. We call a **loop** an edge of the form vv. The size of $V(G)$ and $E(G)$ are denoted $|V(G)|$ and $|E(G)|$ respectively. The size of a graph G is the size of its vertex set so that $|G| = |V(G)|$ typically denoted n. We call a graph **simple** if there is at most one edge between any two vertices. A graph in which multiple edges between two vertices or loops are possible is called a **multi-graph**. Unless otherwise stated, we will deal with simple graphs in this thesis. This previous definition is for what is called an **undirected graph**. We can have a **directed graph** if we think of each edge $uv \in E$ as being directed from u to v.

We say that vertices u and v are **adjacent** in G if $uv \in E(G)$. We denote adjacency by $u \sim v$. The **neighbourhood** of a vertex v denoted by $N(v)$, is the set of all vertices adjacent to v. The **degree** of a vertex v in a graph G, denoted by $deg_G(v)$, is the number of vertices adjacent to v. We will sometimes suppress G in this notation and simply write $deg(v)$ when the graph we are working with is clear. In several cases, we will have a sequence of graphs (G_n). In this case, we will generally write $deg_{G_n}(v) = deg_n(v)$. We call u an **isolated vertex** if $deg(u) = 0$ and call u a **universal vertex** if $deg(u) = n - 1$. In a directed graph, we differentiate between whether or not an incident edge is an in or an out edge. We denote the in-degree of vertex v a directed graph G as $deg_G^-(v)$ and the out-degree as $deg_G^+(v)$.

A **path** in G is a sequence of vertices $v_1v_2\dots v_k$ where v_i is adjacent to v_{i+1} for $1 \leq i \leq k-1$ with each v_i distinct. The length of a path is defined to be the number of vertices in the path.

We say that a graph G is **connected** if for every $u, v \in V(G)$, there is a path starting at u and ending at v in G. Otherwise, we say that G is **disconnected**. If G is directed then we say that G is **strongly connected** if there exists a directed path from u to v and from v to u for all $u, v \in V(G)$.

The **distance** between two vertices u and v in a connected graph, denoted by $d(u, v)$, is the length of the shortest path between u and v. The **diameter** of a graph G is the longest distance between any two vertices in G.

Certain special classes of graphs will arise frequently in this thesis; we take the time to describe these classes. The **complete graph on n vertices** denoted K_n is a graph where each vertex is universal. The graph K_3 is often called a *triangle* in this thesis. The **complete bipartite graph on $n+m$ vertices** $K_{n,m}$ is a graph whose vertex set can be partitioned into two sets of size n and m with no vertex adjacent to a vertex within its own set but each vertex adjacent to every vertex outside its own set.

The **path on n vertices** denoted P_n is a graph which consists of a single path of length n. The **cycle on n vertices** denoted by C_n is a connected graph in which every vertex has degree two. In this thesis we will deal extensively with the size 3 and size 4 connected graphs. We give them special names as indicated in Figure 1.1. Note that some of the graphs in Figure 1.1 have more common names; $g_1 = P_3$, $g_2 = K_3$, $g_4 = P_4$, $g_6 = C_4$ and $g_8 = K_4$. In this thesis we will stick with the more common names while using the names in Figure 1.1 for the less common graphs g_3, g_5 and g_7.

We say that two graphs G and H are **isomorphic**, denoted by $G \cong H$, if there exists a bijection $f : V(G) \to V(H)$ such that $(u, v) \in E(G)$ if and only if

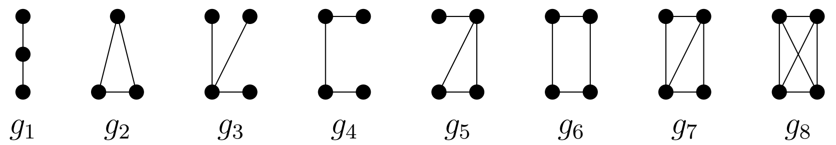


Figure 1.1: Size 3 and Size 4 Connected Graphs

$(f(u), f(v)) \in E(H)$. This means that G and H are identical up to some relabelling of their vertices. Graph isomorphism forms an equivalence relation with equivalence classes called *isomorphism classes*. In this thesis when we refer to a graph G, we are almost always referring to the isomorphism class which contains G. We will denote $\mathcal{G}$ to be the set of all isomorphism classes of graphs. We denote $\mathcal{G}_n$ to be the set of all isomorphism classes of graphs with n vertices and denote $\mathcal{G}_{n,m}$ to be the set of all isomorphism classes of graphs with n vertices and m edges. We also denote $\mathcal{C}_n$ to be the set of all isomorphism classes of connected graphs on n vertices.

A graph automorphism is a graph isomorphism of G to itself. In other words, a relabelling of the vertex set that preserves the graph structure. We denote the set of all automorphisms for a graph G as $Aut(G)$.

There are two different ways in which we can think of a fixed graph F being a subgraph of another graph G. We say that F is an **injective subgraph** of G if $V(F) \subset V(G)$ and $E(F) \subset E(G)$ and the assignment of endpoints to edges in F is the same as in G. We denote the number of injective copies of F in G by $inj(F, G)$. Note that if F is an injective subgraph of G then it is possible that a non-edge between two vertices in F may be an edge in G. We say that F is an induced subgraph of G if there exists an injective function $f : V(F) \to V(G)$ such that $uv \in E(F)$ if and only if $uv \in E(G)$. We denote the number of induced copies of F in G as $ind(F, G)$. It is generally the case in graph theory text books such as [125] that an injective subgraph is referred to simply as a subgraph, while an induced subgraph is referred to explicitly as an induced subgraph. In this thesis we will be primarily interested in the number

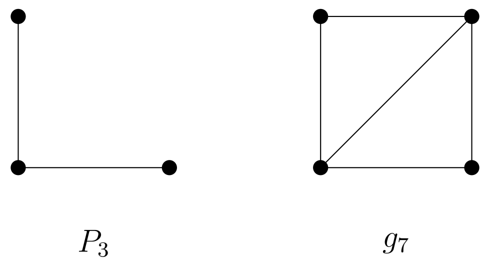


Figure 1.2: $inj(P_3, g_7) = 8, ind(P_3, g_7) = 2$

of induced subgraphs of a certain type contained in a graph. For this reason, we go against the convention and refer to induced subgraphs simply as subgraphs and say injective subgraphs when referring to injective subgraphs.

Example 1.3.1 *Consider the following graphs.*

There is a useful connection between $inj(F, G)$ and $ind(F, G)$ which we will use often in this thesis. The following Theorem is due to Kocay [82].

Theorem 1.3.2 *Let $G_1, G_2, \ldots G_m$ be all the graphs on k vertices and consider any graph G. Then for any $G_i \in \mathcal{G}_k$ we have*

$$inj(G_i, G) = \sum_{j=1}^{m} inj(G_i, G_j) ind(G_j, G).$$

Example 1.3.3 *Consider the two graphs P_3 and g_7 from Figure 1.2. Using Theorem 1.3.2 we can write $inj(P_3, g_7) = inj(P_3, P_3)ind(P_3, g_7) + inj(P_3, K_3)ind(K_3, g_7) = (1)(2) + (3)(2) = 8$.*

It is also observed in [35] that $hom(F, G)$ and $inj(F, G)$ are related. Let Θ be any equivalence relation on $V(F)$ and let F/Θ be the graph formed by identifying vertices in the same equivalence class of Θ. We can write $hom(F, G) = \sum_{\Theta} inj(F/\Theta, G)$. Therefore, since $inj(F, G)$ and $ind(F, G)$ are related, it is clear that $hom(F, G)$ and $ind(F, G)$ are related as well.

1.3.2 Probability Theory

Probability will be used extensively in this thesis. We take the time to outline the terminology and important theorems which will be used frequently.

A **random variable** X is a variable whose value is determined by some random process. For example, if X is the value of a roll of a dice, then X is a random variable whose possible values are $1, 2, 3, 4, 5, 6$. We denote the probability that X takes the value of x by $Pr(X = x) = Pr(x)$.

One important type of random variable is an indicator variable. An **indicator variable** for an event S is a random variable such that

$$X_S = \begin{cases} 1 & if\ S\ occurs \\ 0 & if\ S\ does\ not\ occur \end{cases}$$

For example, if S is the event that the roll of a die is an even number, then $X_S = 1$ if a 0, 2 or 4 is rolled and $X_S = 0$ if a 1, 3 or 5 is rolled.

The **expected value** for a random variable X is defined as

$$E(X) = \sum_i x_i Pr(x_i),$$

where the sum ranges over all possible values x_i for X. It is easy to see that for an indicator variable X_S that $E(X_S) = Pr(X_S = 1)$. The **variance** of a random variable X, which measures the spread of the distribution of X, is defined by

$$V(X) = \sum_i (x_i - E(X))^2,$$

where the sum is over all possible values x_i for X. When computing the variance, often the simple formula $V(X) = E(X^2) - E(X)^2$ is used.

Linearity of Expectation says that if X is a sum of random variables $X_1 +$

$X_2 + \ldots + X_n$ then

$$E(X) = \sum_{i=1}^{n} E(X_i).$$

We often have to consider how the probability of a certain event X is affected by the occurrence of some different event Y. Such a calculation involves **conditional probability**. The probability that X occurs given that we know that $Y = y$ is denoted by $Pr(X|Y = y)$. Using this, we define the **conditional expectation** of X given $Y = y$ as

$$E(X|Y = y) = \sum_{i} x_i Pr(X = x_i|Y = y).$$

Though $E(X|Y = y)$ seems like it should be a number, it is actually a random variable. The following is a useful lemma for determining its expectation.

Lemma 1.3.4 *Consider two random variables X and Y. Then $E(E(X|Y)) = E(X)$.*

We say that an event X occurs **asymptotically almost surely** or **a.a.s.** if the probability that the event occurs tends to 1 as $n \to \infty$. We say that an event X occurs **with extreme probability** or **w.e.p.** if the probability that X occurs is bounded below by $1 - e^{-\Theta(\log^2(n))}$.

1.3.3 Asymptotic Approach and O-notation

In Chapter 2 we compute the expected graphlet counts in various random graph models. In many cases, obtaining exact expressions for these expectations will require meticulous calculation. Most often, we are not interested in the exact expression, but the leading term of the expression which gives the rate at which the expectation grows with the size of the graph n. This allows us to make some simplifications which make the expectation calculations more tractable. The use of order notation will be used frequently in this process, so we take the time to outline the notation here.

- We write $f(n) = O(g(n))$ to mean that there exists an N and constant c such that for all $n \geq N$, $|f(n)| \leq cg(n)$.

- We write $f(n) = \Omega(g(n))$ to mean there exists and N and constant c such that for all $n \geq N$, $|f(n) \geq cg(n)$.

- We write $f(n) = o(g(n))$ to mean that for all $\epsilon > 0$, there exists an N such that for all $n \geq N$, $|f(n)| \leq \epsilon|g(n)|$.

- We write $f(n) = \Theta(g(n))$ to mean there exists an N and two constants c_1, c_2 such that for all $n \geq N$, $c_1 g(n) \leq f(n) \leq c_2 g(n)$.

- We write $f(n) \simeq g(n)$ to mean that $f(n) = (1 + o(1))g(n)$.

To highlight the differences between these notations, let us compare $f(n) = O(n^2)$, $g(n) = \Theta(n^2)$ and $h(n) \simeq n^2$. For $f(n)$, we have that $f(n) \leq cn^2$ for all $n \geq N$ for some constant c. All we know is that $f(n)$ is bounded above by n^2. It could be the case that $f(n) = n$, $f(n) = ln(n)$ or $f(n) = nln(n)$, to list a few possibilities. On the other hand, $g(n) = \Theta(n^2)$ implies that $g(n)$ is sandwiched in between $c_1 n^2 \leq g(n) \leq c_2 n^2$ for all $n \geq N$ for some constants c_1, c_2. This implies that the leading term in $g(n)$ must be of order n^2. Now consider $h(n) = (1 + o(1))n^2 = n^2 + o(n^2)$. In this case, as $n \to \infty$, $h(n)$ tends to n^2. The major difference between $\Theta(n^2)$ and $(1 + o(1))n^2$ is that in the former case, we do not know the coefficient of the n^2 term and in the later, we know the coefficient is 1.

A common application of O-notation is in simplifying the Binomial series.

Example 1.3.5 *For any complex α, we can write the Binomial series $(1 + \frac{1}{n})^\alpha = \sum_{i=1}^{\infty} \binom{\alpha}{i} (\frac{1}{n})^i$ where $n > 1$. Using the O-notation, we can write this as $(1 + \frac{1}{n})^\alpha = 1 + \alpha\frac{1}{n} + O((\frac{1}{n})^2)$.*

In our expected subgraph calculations we will often be confronted with sums of the general form $\sum_{i=a}^{b} f(i)$ in which no simplification exists. Obtaining estimates for such sums can be done by replacing the sum with an integral as prescribed in the following lemma.

Lemma 1.3.6 *Let f be a monotonic integrable function on $a \leq x \leq n+1$. Then we can write $\sum_{i=a}^{n} f(i) = \int_a^{n+1} f(x)dx + \Delta$ where $\Delta = O(f(n) - f(a))$. Furthermore, if f is decreasing then $\sum_{i=a}^{n} f(i) = \int_a^{n+1} f(x)dx + O(f(a))$ and if f is increasing then $\sum_{i=a}^{n} f(i) = \int_a^{n+1} f(x)dx + O(f(n))$.*

Proof Let f be a monotonic function. The fact that $\sum_{i=a}^{n} f(i) = \int_a^{n+1} f(x)dx + \Delta$ follows from a left end point approximation of $\int_a^{n+1} f(x)dx$ using unit intervals. To determine Δ we let $\delta_i = \max_{i \leq x \leq i+1} |f(x) - f(i)|$ which is the maximum error of the approximation on the interval $[i, i+1]$. For the overall error we have $\Delta \leq \sum_i \delta_i$. Since f is monotonic we have $\delta_i = |f(i+1) - f(i)|$. If f is increasing then $\delta_i = f(i+1) - f(i)$ so $\Delta \leq \sum_{i=a}^{n} \delta_i = f(n+1) - f(a) = O(f(n))$. Similarly, if f is decreasing then $\delta_i = f(i) - f(i+1)$ so $\Delta \leq \sum_i \delta_i = f(a) - f(n+1) = O(f(a))$. □

Lemma 1.3.6 will be used extensively in Chapter 2. When using it, we will refer to the lemma and generally suppress some of the details of the calculation in an attempt to make proofs readable. A common application of Lemma 1.3.6 is to simplify the sum $\sum_{i=a}^{n} \frac{1}{i^p}$. As any good calculus student knows, the solution of this sum depends on the value of p. In regards to Lemma 1.3.6, when $p = 1$, the integral we compute is different than the one we compute for $p \neq 1$. In Examples 1.3.7 and 1.3.8 we provide the details of the applications of Lemma 1.3.6 to this sum for both cases.

Example 1.3.7 *Consider $\sum_{i=a}^{n} \frac{1}{i}$. Since $f(i) = \frac{1}{i}$ is a decreasing function, using Lemma 1.3.6, we can write*

$$\begin{aligned}\sum_{i=a}^{n}\frac{1}{i} &= \int_a^{n+1}\frac{1}{x}dx + O(\frac{1}{a})\\ &= ln(n+1) - ln(a) + O(\frac{1}{a}).\end{aligned}$$

The $ln(n+1)$ term in this case can be written as $ln(n(1+\frac{1}{n})) = ln(n) + ln(1+\frac{1}{n}) = ln(n) + O(\frac{1}{n})$. The simplification of $ln(1+\frac{1}{n})$ to $O(\frac{1}{n})$ follows from the Taylor expansion of $ln(1+x) = \sum_{i=1}^{\infty}\frac{(-1)^{i+1}}{i}x^i$ for $|x| < 1$. Overall, we obtain $\sum_{i=a}^{n}\frac{1}{i} = ln(n) - ln(a) + O(\frac{1}{a})$.

Example 1.3.8 *Now consider the sum $\sum_{i=a}^{n}\frac{1}{i^p}$ for $p \neq 1$. The function $f(i) = \frac{1}{i^p}$ is a decreasing function so using Lemma 1.3.6 we get*

$$\begin{aligned}\sum_{i=a}^{n}\frac{1}{i^p} &= \int_a^{n+1}\frac{1}{x^p}dx + O(\frac{1}{a^p})\\ &= \frac{1}{1-p}(n+1)^{1-p} + O(\frac{1}{a^p})\\ &= \frac{1}{1-p}n^{1-p} + O(a^{-p})\end{aligned}$$

In the final step, from the Binomial series we have

$$\begin{aligned}(1+n)^{1-p} &= n^{1-p}(1+\frac{1}{n})^{1-p}\\ &= n^{1-p}(1+O(\frac{1}{n}))\\ &= n^{1-p} + O(n^{-p}).\end{aligned}$$

Now if $p < 1$, then n^{1-p} is the leading term so we write $\sum_{i=a}^{n}\frac{1}{i^p} = \frac{1}{1-p}n^{1-p} + O(a^{-p})$. However, if $p > 1$ then $n^{1-p} \to 0$ and $O(a^{-p})$ is the dominant term. In this *case we write* $\sum_{i=a}^{n} \frac{1}{i^p} = O(a^{-p})$.

One particular equation that will be used frequently is $ind(P_3, G) + 3ind(K_3, G) \simeq \sum_{i=1}^{n} \frac{deg(v_i)^2}{2}$. We provide a lemma containing this result so it can be understood in the remainder of the thesis.

Lemma 1.3.9 *Let G_n be a graph of size n such that the number of edges in G_n tends to infinity with n. Then we can write*

$$ind(P_3, G_n) + 3ind(K_3, G_n) \simeq \sum_{i=1}^{n} \frac{deg_n(v_i)^2}{2}.$$

Proof We begin with the observation that $ind(P_3, G_n) + 3ind(K_3, G_n) = \sum_{i=1}^{n} \binom{deg_n(v_i)}{2}$. This follows since each pair of edges incident to a vertex v_i either results in an induced 3-path or a triangle. Note that counting in this way counts each triangle 3 times. Performing some simplifications gives

$$\begin{aligned}
ind(P_3, G_n) + 3ind(K_3, G_n) &= \sum_{i=1}^{n} \binom{deg_n(v_i)}{2} \\
&= \sum_{i=1}^{n} \frac{deg_n(v_i)(deg_n(v_i) - 1)}{2} \\
&= \frac{1}{2} \sum_{i=1}^{n} deg_n(v_i)^2 - deg_n(v_i) \\
&= (1 + o(1)) \sum_{i=1}^{n} \frac{deg_n(v_i)^2}{2} \\
&\simeq \sum_{i=1}^{n} \frac{deg_n(v_i)^2}{2}.
\end{aligned}$$

Note that the step $\frac{1}{2} \sum_{i=1}^{n} deg_n(v_i)^2 - deg_n(v_i) = (1 + o(1)) \sum_{i=1}^{n} \frac{deg_n(v_i)^2}{2}$ only holds if $\sum_{i=1}^{n} deg_n(v_i)$ tends to infinity. Since $\sum_{i=1}^{n} deg_n(v_i)$ is equivalent to two times the number of edges, this is satisfied. □

1.4 Networks in the Real World

The use of networks to model phenomena in the real world is far reaching, touching many distinct disciplines such as the social sciences, chemistry, biology, computer science and engineering to name a few. In this section we provide an overview of the body of work on complex networks. In this short review, we focus primarily on material which is used in this thesis. For more information, the reader can check the following surveys [8, 25, 103]. There are also several text books that have been written on the subject, such as [123, 39, 56, 107, 31].

1.4.1 Examples of Complex Networks

Different types of complex networks can be coarsely divided into 4 different classes: social, biological, technological and information. The main focus of this thesis is online social networks but the model-selection experiment of Chapter 3 could be used to validate a model for any type of complex network.

Social Networks

In a social network, individuals or groups of individuals are modelled by vertices, and an edge between two vertices represents a relationship, commonly friendship, between the individuals or groups of individuals. Relationships may or may not be mutual, so that a social network may be undirected or directed. For example, in Facebook friendships are mutual, but in Twitter they are not.

The study of social networks commonly called *social network analysis* is one of the pioneering fields of complex network analysis. Besides a general interest in understanding human interactions, social scientists study social networks to develop a better understanding of how information or a disease may spread through a social network.

With the advent of online social networking services, papers are rapidly appearing in an attempt to further our understanding of these large complex networks. Some online social networks which have been studied include Facebook, CyWorld, MySpace, Orkut, Youtube, Flickr, Yahoo! 360, LiveJournal and Twitter [6, 49, 85, 89, 23, 51, 96, 16, 78, 88]. In this thesis, we study the data sets of Porter *et al* from [96] containing Facebook networks of 100 different American universities and colleges.

A classic example of a social network is the collaboration network. In this network, two individuals are connected by an edge if they have collaborated together on a project. For example, a collaboration network can be formed amongst academics who have co-authored papers together. Collaboration networks amongst academics from physics, bio-medical research, high-energy physics and computer science were studied by Newman in [99, 100, 101] and collaboration networks amongst mathematicians and neuroscientists were studied by Barabási *et al* in [19]. Another example of a collaboration network is the movie actor collaboration network which was studied in [118, 104, 17].

Information Networks

Unlike social networks whose edges model a social relationship between individuals, edges in an information networks convey that there is some information shared amongst the two individuals. Perhaps the two most studied information networks are the World Wide Web and citation networks.

In the World Wide Web (WWW), whose network is commonly called the web graph, web pages are represented by vertices and there is a directed edge from one web page to another if the former contains a hyperlink to the latter. The WWW is immensely large with a study by Hirate *et al.* in 2005 [72] putting the size at 53.7 billion web pages. Due to this immense size, only samples of the Web graph obtained by crawling web sites can be feasibly analyzed. Earlier studies of the structure of the web graph include Albert *et al.* [9, 10], Kleinberg *et al.* [81] and Broder *et al.* [40].

In a citation network, papers are represented by vertices, and there is a directed edge from one paper to another if the former cites the latter in its bibliography. Citation networks which have been analyzed include publications in journals cataloged by the Institute of Scientific Information [114], publications in the journal Physical Review D volumes 11-50 [114] and high energy physics citations in ArXiv [89].

Technological Networks

Technological networks are typically man-made networks in which some resource is distributed amongst certain junctures. One interesting example is the power grid. In this network generators, transformers and substations are modelled by vertices, and edges represent power lines. The power grid network for the western US was studied by Watts and Strogatz in [118]. A similar network is the airline network where airports are represented by vertices and edges indicate that flights go between the airports. This network is studied by Amaral *et al.* [12].

The Internet forms a technological network at the router level where routers are modelled as vertices and routers are connected if they send information to one another. At the inter-domain or autonomous level we model domains (a collection of routers and computers) as vertices and domains are connected by an edge if they send information to one another. Faloutsos *et al.* studied the internet at the router and inter-domain levels in [61]. Other papers which have analyzed the internet include [21, 106, 48].

Biological Networks

Our last type of network models biological systems at the cellular level. A large number of different types of biological networks have been studied. One important type of biological networks are protein protein interaction (PPI) networks. In these networks, proteins are modelled as vertices, with edges between proteins if they interact with one another. The process of determining the interactions amongst proteins requires meticulous lab work by biologists. Often interactions are missed or interactions which are not present are mistakenly "discovered". However, there are several high confidence methods such as yeast 2-hybrid systems which work well. Additional information on PPI networks can be found in [109]. The study of PPI networks is an extensive field. Some papers published on PPI networks include [18, 50, 105, 111, 110, 54].

Metabolic pathways are another biological network which has garnered a lot of attention over the years. In this network, metabolic substrates are represented by vertices with directed edges between substrates if a chemical reaction exists that can transform one substrate into the other. The study of these networks is important in enhancing our understanding of cell function and has applications to genetic engineering. Papers which have studied the properties of metabolic pathways include [11, 63, 20, 64, 55, 119].

1.4.2 Properties of Complex Networks

Through the study of complex networks described in Section 1.4, various common properties of these networks have come to light.

Power Law Degree Distribution and Assortativity

Arguably the most important observation of complex networks is that their degree distributions have *heavy tails*. It has been widely reported that these heavy tails follow a power law in many complex networks [9, 87, 40, 61, 96, 114, 55, 119, 18, 11]. Let $N_{k,n}$ indicate the proportion of vertices in G of size n which has degree k. A power law degree distribution for G is a degree distribution which obeys $\frac{N_{k,n}}{n} = Ck^{-\gamma}$ for some constant C and $\gamma > 1$. We refer to γ as the *power law coefficient*. It is uncommon that a complex network follows a power law precisely for all values of k. Note that a power law degree distribution is not defined for $k = 0$. Also, due to the finiteness of samples of complex networks, there are typically large deviations in the tail of the distribution which cannot be accounted for in a power law distribution of the degrees. This makes providing a precise definition for of a power law for finite graphs challenging. Typical methods for verifying that empirical data follows a power law is to plot the proportion of vertices of degree k on a log-log plot. If the data does follows a power law then the plot should be linear as $\log(\frac{N_{k,n}}{n}) = -\gamma \log(k) + \log(C)$. To determine whether or not a linear relationship exists, a linear regression is performed. It has been noted that such an approach is not statistically valid and a more accurate method has been proposed [53].

Many of the models for complex networks have been designed specifically so that they replicate the power law behaviour of the degree distribution of complex networks. In analyzing the degree distribution of these models, it is often shown that the degree distribution of the models asymptotically satisfies a power law degree distribution. We will say that a sequence of graphs (G_n), or simplify G_n, satisfies a power law degree distribution a.s.s. if as $n \to \infty$, $\frac{N_{k,n}}{n} = (1 + o(1))Ck^{-\gamma}$. In many cases, the power law only holds up to some maximum degree k_{max}.

Note that for directed graphs, the notions of a power law in-degree or out-degree distribution are identical to those discussed above.

With a power law degree distribution, there are relatively few vertices incident to a large number of edges, while most vertices are incident to a small number of edges. The presence of such a degree distribution is easily understood in the web graph or social networks since high degree vertices or highly influential individuals or webpages have a tendency to attract more friends or links. On the other hand, individuals or webpages with a low amount of influence struggle to attract more friends or links. This indicates an undemocratic nature of a power law degree distribution.

Another complex network property which depends on the degree is the *assortativity coefficient* introduced by Newman in [102]. The assortativity coefficient $r \in [-1, 1]$ is a measurement of how the degree of a vertex affects the degree of the vertices it links to. An assortativity coefficient which is close to 1 indicates there is a strong tendency for vertices of the same degree to link to one another. We call the network *assortative* in this case. If r is close to -1 then there is a strong tendency for vertices to link to vertices with very different degrees. We call the network *disassortative* in this case. If $r = 0$ then vertices have no tendency to link to any specific vertices. In complex networks, edges are not formed completely at random but are dependent on some properties of the vertices so that complex networks have non zero assortativity coefficients. In [102], the authors' study concludes that social networks tend to be assortative while technological and biological networks tend to be disassortative.

The assortativity coefficient is determined by the following equation

$$r = \frac{\sum_i e_{ii} - \sum_i a_i b_i}{1 - \sum_i a_i b_i}, \tag{1.1}$$

where e_{ij} is the proportion of edges from a vertex of degree i to a vertex of degree j and $a_i = \sum_j e_{ij}$ and $b_j = \sum_i e_{ij}$.

A recent paper by Litvak and Hofstad [97] demonstrates that for large disassortative graphs with a power law degree distribution, the assortativity coefficient decreases with the size of the graph. As a result, for large graphs, the assortativity graph is close to zero and under estimates the level of disassortativity present in the graph. In Chapter 3 we use the assortativity coefficient as a valid network statistic in our experiment. We feel this is justified as we use a social network, which do tend to be assortative [102].

Small World Property

The most well known property of complex networks is certainly the *small world property* which was famously demonstrated by the “six degrees of separation” experiment of Milgram in [92]. In Milgram’s experiment, it was observed that on average, any two random American’s are separated by at most 6 people. Though there has been some scrutiny of this experiment, the small world property is a well observed phenomenon in many complex networks. Roughly stated, the property holds if the distance between vertices grows much slower than the size of the network.

There are several ways in which this concept can be expressed mathematically. The most popular seems to be through the use of the average path length l of the network. Another possibility is to use the diameter of the graph but this is problematic if the graph is disconnected. The *average path length* is defined in [118] to be

$$l = \sum_{u,v \in S} \frac{d(u,v)}{|S|}, \tag{1.2}$$

where S is the largest connected component of G. In the case that G is directed, we take S to be the largest strongly connected component and treat the pairs (u,v) and (v,u) separately in the sum in Equation 1.2. We say that a graph G is a *small world graph* if $l = O(ln(n))$ so that the average path length does not grow faster than the logarithm of the graph size. This definition is often used to define the small world property and was introduced by Watts and Strogatz in [118] in their study of graphs with small average path lengths and high clustering. Studies have shown that many complex networks such as the internet at the router and inter-domain level [21], WWW [9, 40], metabolic pathways networks [63], PPI networks [18] and Facebook [96] are small world graphs..

Clustering

An apparent organizational pattern in many real world networks is that they tend to cluster into small highly connected groups. In the context of social networks, clusters represent groups of individuals who all know each other (they play for the same sports team, take university classes together, etc.). The tendency of a network to cluster is quantified by the *clustering coefficient*. There are two separate but related definitions used for the clustering coefficient. Since both are used regularly we will define both.

The first one was introduced by Watts and Strogatz in [118]. We begin by defining the local clustering coefficient of a vertex v as

$$C_v(G) = \frac{\text{number of edges between neighbours of v}}{\binom{deg_G(v)}{2}}.$$

If the degree of v is 0 or 1 we set $C_v(G) = 0$. The local clustering coefficient measures for each vertex v, the fraction of incident edges which close to form a triangle containing v. The first clustering coefficient is defined by taking the average of all the local clustering coefficients

$$C_1(G) = \sum_{v=1}^{n} \frac{C_v(G)}{n}. \tag{1.3}$$

The other definition of the clustering coefficient is used quite frequently in social network analysis and was introduced by Faust and Wasserman in [62]. The clustering coefficient in this case is simply defined as

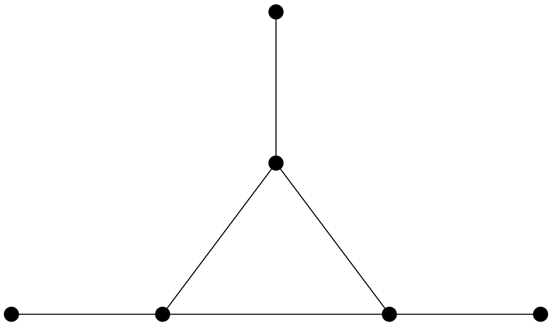
$$C_2(G) = \frac{3ind(K_3, G)}{inj(P_3, G)} = \frac{3ind(K_3, G)}{ind(P_3, G) + 3ind(K_3, G)} \tag{1.4}$$

In [116], Bollobás and Riordan show that the local clustering coefficients can be used to obtain $C_2(G)$ as well

$$C_2(G) = \frac{\sum_{v=1}^{n} \binom{d_G(v)}{2} C_v(G)}{\sum_{v=1}^{n} \binom{d_G(v)}{2}}.$$

For both definitions, $C_1(G), C_2(G) \in [0,1]$, where 0 is achieved for triangle free graphs, 1 is achieved for the complete graph and p is achieved for the random graph. For a general graph G, the coefficients are not the same (see Example 1.4.1). In [8] it is shown that complex networks experience significantly more clustering than a random graph of the same size and density, so that for complex networks either definition is appropriate. The most appropriate definition for this thesis is $C_2(G)$ because of its relation to the number of induced 3-paths and triangles in G. For our experiment in Chapter 3, we do not use the clustering coefficient as a feature. Since we use the number of 3-paths and triangles as features, the addition of the clustering coefficient provides no additional information.

Example 1.4.1 *The the following graphs* $C_1(G) = \frac{1}{6}$, *whereas* $C_2(G) = \frac{1}{3}$.



G

1.5 Graph Models

Many random graph models have been proposed for complex networks over the years. These models are designed so that they generate graphs with the same real world properties described in Section 1.4.2. A model's ability to replicate these properties has historically been enough to justify the use of the model. In this section, we review the models that will be studied in this thesis. These models are generally designed to replicate an apparent mechanism which guides edge formation in complex networks. The three mechanisms we consider are *preferential attachment, copying* *and spatial*. The preferential attachment mechanism increases the likelihood that high degree vertices in the network will continue to accumulate more neighbours, while low degree vertices will not. Through the copying mechanism, a vertex accumulates its neighbours by copying the neighbours of an existing vertex. In the spatial mechanism, vertices lie in a metric space and edges are formed between vertices which are close to one another. We consider these three mechanisms because each could plausibly guide edge creation in a social network. The preferential attachment mechanism is appropriate in a social network because popular individuals have more influence in the network and are more likely to accumulate more friends. The copying mechanism is appropriate because it is common to meet new people through another individual thus copying their friends. Individuals who live close to one another are also more likely to be friends while those who live far away from one another are less likely, thus justifying the spatial mechanism. The geometry of a spatial model doesn't necessarily need to refer to geographical location. The spatial representation could model common interests, with individuals having more common interests being closer in the space than those who have less. In Chapter 3 we conduct a model-selection experiment to determine which of these mechanisms is the most likely to have generated data from Facebook.

The formation of a graph under each of the models follows the same paradigm in which vertices are added to the graph one vertex at a time.

Procedure 1.5.1 (General Random Graph Generation Algorithm)

1. *Begin with an initial graph G_0.*

2. *Form a new graph G_n by adding a new vertex v_n to G_{n-1}.*

3. *Assign edges from v_n to vertices in $V(G_{n-1})$ by a specified mechanism.*

Each model forms a sequence of graphs $(G_0, G_1, \ldots, G_n)$. The only difference between each model is the edge formation mechanism in Step 3.

1.5.1 Erdős-Rényi Random Graph

The Erdős-Rényi Random Graph, or simply the random graph, was first introduced by Erdős and Rényi in [60]. The model has two parameters: the number of vertices n and a probability p. The graph is formed by taking the n vertices and adding an edge independently between each pair of vertices with probability p. Equivalently, the Random Graph can be formed using Procedure 1.5.1 by allowing each vertex to appear one at a time and adding an edge independently to each existing vertex with probability p. We will denote the random graph with parameters n and p by $ER(n, p)$. The properties of ER have been extensively studied. In particular, it is well established that the random graph is not a good model for complex networks. The independence of the edge formation leads to a Binomial degree distribution, not a power law. The Random Graph displays little clustering in contrast to the high level of clustering in complex networks. The Random Graph however does have the small world property [28].

1.5.2 Preferential Attachment Models

The preferential attachment (PA) model was originally introduced by Barabási and Albert in [17]. The motivation for the introduction of this model was to provide a more appropriate real world graph model which produced a power law degree distribution. The authors of [17] provide a heuristic argument that the power law coefficient for the PA graph is 3. A precise definition of the PA model was given by Bollobás *et al.* in [30] along with a rigorous proof that the power law coefficient is 3.

The original PA model has 2 parameters: the number of vertices n, and the number of edges added in each step d. For the PA model, we can either start with a small connected graph or a single vertex with d loops. In Chapter 2, for computing expected subgraph counts in the PA model, we will start with a single vertex with

d loops. In Chapter 3, for our model-selection experiment, we start with a small random graph. Most papers consider an initial vertex with d loops and we describe the PA model for this case.

In the PA model, we iteratively form a graph G_n by adding a vertex v_n to G_{n-1} and forming d edges from v_n to d i.i.d chosen vertices in G_{n-1}. The endpoint v_i of each edge is chosen according to the distribution

$$Pr(i = s) = \frac{deg_{n-1}(v_s)}{2d(n-1)} \quad if \quad 1 \leq s \leq n-1. \tag{1.5}$$

Note that it is possible that v_i is chosen more than once in the same time step. In this case, there are multiple edges from v_n to v_i. New vertices select there neighbours preferentially by picking vertices with a probability proportional to their degree. We denote the PA model with parameters n and d as $PA(n, d)$. In general, the PA model does not generate simple graphs as multiple edges are possible. However, we consider the PA model to generate simple graphs by removing multiple edges after generating the graph.

Many papers [59, 41, 94, 95, 58, 57] consider variations of the original PA models. The motivation behind these generalization is to create a preferential attachment model with a tunable power law degree distribution coefficient. We consider a generalized version of the original PA model from [59] where vertices are given an initial attractiveness parameter $\alpha > 0$ so that $Pr(i = s) \propto deg_{n-1}(v_s) + \alpha$. The generalized PA model denoted by $PA(n, d, \alpha)$ is formed in precisely the same way as in [30] except the probability that an endpoint v_i is chosen as an endpoint of an edge from v_n is

$$Pr(i = s) = \frac{deg_{n-1}(v_s) + \alpha}{(2d + \alpha)(n-1)} \quad if \quad 1 \leq s \leq n-1. \tag{1.6}$$

The generalized PA model allows us to tune the power law coefficient. The following result in [30] shows the $PA(n, d)$ has a power law degree distribution for degree

$k = 0, 1, \ldots, n^{\frac{1}{15}}$ with coefficient 3.

Theorem 1.5.2 *Consider the original preferential attachment model $PA(n, d)$ and let $N_{k,n}$ be the number of vertices of degree k at time n. Let*

$$\alpha_{k,d} = \frac{2d(d+1)}{(k+d)(k+d+1)(k+d+2)}.$$

Then for any fixed $\epsilon > 0$ a.a.s.

$$(1-\epsilon)\alpha_{k,d} \leq \frac{N_{k,n}}{n} \leq (1+\epsilon)\alpha_{k,d}$$

for $0 \leq k \leq n^{\frac{1}{15}}$.

In [94], it is shown that $PA(n, d, \alpha)$ has a power law degree distribution with power law coefficient $3 + \alpha$.

1.5.3 Copy and Duplication Models

There have been many different Copy models, sometimes called Duplication models, which have been suggested to model the growth of the web graph and biological networks over the years [81, 86, 84, 17, 52, 105, 32]. The first copy models appeared as models for the web graph [81, 86]. These models differ in several ways, with the underlaying graph being directed in some [81, 86, 84] and undirected in others [105, 52, 17, 32]. Another major difference is that in some models, a fixed number of edges are added in each step [81, 86] and in others a random number of edges are added in each step [105, 52, 17, 84, 32].

We are going to study a version of the copy model studied in [17] and [32] which gives a more general model of those studied in [105, 52]. This model is an undirected model. We also study the directed version of the model we describe. It will be necessary to include an additional step in the directed case as we will describe later.

Start with an initial connected graph G_{n_0} on n_0 vertices. In each time step $n > n_0$, a new vertex v_n is added to G_{n-1}. Edge formation occurs in two steps. In the *copying step*, a *copy vertex* w is selected uniformly at random from G_{n-1} and each neighbour (out-neighbour) of w is attached by an edge (directed edge) from v_n with probability p. In the directed case we add an edge from v_n to w with probability q. Note this step is necessary otherwise the in-degree of new vertices will always be zero. Vertices with in-degree 0 can never obtain new copy edges. In the second step, edges are added from v_n to d u.a.r. selected vertices in G_{n-1}. We will denote the undirected copy graph with parameters n, p, d, G_{n_0} as $Copy(n, p, d, G_{n_0})$ and the directed copy model with parameters n, p, q, d, G_{n_0} as $DCopy(n, p, q, d, G_{n_0})$. We call the special case where $d = 0$ the *pure copy model* and denote the undirected and directed versions by $Copy(n, p, G_{n_0})$ and $DCopy(n, p, q, G_{n_0})$ respectively. The pure directed copy model is studied in [84] and it is shown that $DCopy(n, 1, 1, G_{n_0})$ has a power law degree distribution with coefficient 2, while it is shown in [17] that the undirected pure copy model does not have a power law degree distribution. However, it is shown that if $d > 0$, the $Copy(n, p, d, G_{n_0})$ does generate a power law degree distribution. The following result is proved in [22] and will be used in this thesis so we include it below.

Theorem 1.5.3 *Let $G_n = Copy(n, p, d, G_{n_0})$ where $d > 0$. Let $N_{k,n}$ be the number of vertices of degree k in G_n. Then for some constant c we can write,*

$$\frac{E(N_{k,n})}{n} = (1 + O(\frac{1}{k}))ck^{-\gamma},$$

where γ is the largest solution to the equation $1 = p\gamma - p + p^{\gamma-1}$.

In Chapter 2, for the undirected copy model, we also consider the case where $p = 0$ and $d > 0$. This graph is called the *uniform attachment graph* and is denoted by $UA(n, d)$.

1.5.4 Random Geometric Graphs

In the random geometric graph (RGG) model, n vertices $X_1, X_2, \ldots, X_n$ are placed in a metric space (S, d) according to some probability density function. An edge is placed between two vertices X_i, X_j if $d(X_i, X_j) < r$. It is most common and will be the case in this thesis, that the vertices are placed uniformly at random in the space. An extensive survey of the random geometric graph can be found in the book [108].

We consider a slight variation of the RGG in which vertices within the specified threshold r are joined by an edge with probability $0 < p \leq 1$. This version is sometimes called the *percolated random geometric graph*. We will denote the RGG in the metric space (S, d), with n vertices, threshold $r \in (0, 1)$ and probability $p \in (0, 1]$ as $Geo(S, d, n, r, p)$. In this thesis we will mostly consider the metric space $([0, 1]^t, d_\infty)$ where d_∞ is the infinity norm induced by the torus metric. The use of the torus metric results in each position in $[0, 1]^t$ being identical. The use of the torus metric simplifies the calculations in Chapter 2. The use of the torus metric is also justifiable from a modelling perspective if there is no inherit advantage in one location over any other. A similar RGG identical to this one but using the Euclidean metric is studied in [13, 14, 15, 98].

The RGG has been used as a model for PPI networks [110, 54], wireless ad-hoc networks [127] and virus spreading [75]. The model has a Poisson degree distribution [108] which might be viewed as a disadvantage in modelling complex networks. Furthermore, small values for the threshold r makes it impossible for vertices which are far apart in the space to be connected. As a result, RGGs tend to have larger average path lengths as compared to our other selected models. However, the work of Pržulj in [110, 54] argues that RGG's are good models for PPI networks because of their ability to replicate the graphlet structure of these networks. In Chapter 3 we determine whether or not RGG's are also good at replicating the graphlet structure of Facebook networks.

We review the metric spaces we will use for the RGG in this thesis. We use $([0,1]^t, d)$ for $t \geq 1$ with several different distance functions d. The **Euclidean distance** between two points $x, y \in [0,1]^t$ where $x = (x_i)_{i=1}^t$ and $(y_i)_{i=1}^t$ is defined as $d_{euc}(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_t - y_t)^2}$. We focus primarily on a variation of the Euclidean metric which removes the boundary in $[0,1]^t$. We define the **torus metric** in $[0,1]^t$ as $d_{tor}(x,y) = min_{u \in \{0,1,-1\}}\{d_{euc}(x, y+u)\}$. When using the torus metric in $[0,1]^2$ we can visualize points as laying on a torus. When using the torus metric in $[0,1]$ we can visualize points as laying on the circumference of a circle. One additional metric we will use is the product metric.

Definition 1.5.4 *Let $(X_1, d_1) \ldots (X_n, d_n)$ be a sequence of metric spaces. The* p-product metric *d_p for $1 \leq p < \infty$ on $X_1 \times \ldots \times X_n$ is defined as*

$$d_p((x_1, x_2, \ldots, x_n), (y_1, y_2, \ldots, y_n)) = (d_1(x_1, y_1)^p + d_2(x_2, y_2)^p + \ldots + d_n(x_n, y_n)^p)^{\frac{1}{p}}.$$

We call these metric spaces the L^p spaces. The special case where $p \to \infty$ gives the infinity norm *defined as*

$$d_\infty((x_1, x_2, \ldots, x_n), (y_1, y_2, \ldots, y_n)) = \max_i\{d_i(x_i, y_i)\}.$$

Note that d_p is the generalization of the Euclidean metric. In Section 2.6.5 we consider the metric space $([0,1]^t, d_\infty)$ where each (X_i, d_i) is $([0,1], d_{tor})$ in Definition 1.5.4.

1.5.5 Spatial Preferred Attachment Model

The Spatial Preferred Attachment model introduced in [7] was originally proposed as a model for the web graph. In the model, vertices are placed independently and uniformly at random into $[0,1]^k$. The distance between two vertices is determined using the torus metric.

The model generates a sequence of directed graphs $(G_t)_{t=1}^n$. At time t, a new node v_t is added uniformly at random into the space. Each previous node v_i has an **influence region** at time t which has area

$$A(v_i,t) = \frac{A_1 deg_t^-(v_i) + A_2}{t},$$

where A_1 and A_2 are constants. The new node v_t forms a directed edge from v_t to v_i with probability p for each v_i such that $v_t \in A(v_i, t-1)$. In words, an edge forms from v_t to v_i with probability p only if v_t is placed in v_i's influence region at time t. The dependence of $A(v_i,t)$ on $deg_t^{-1}(v_i)$ implies that the SPA model implicitly incorporates the preferential attachment mechanism by allowing vertices with higher in-degree to accumulate more neighbours. We will denote a SPA graph of size n generated with parameter n, k, p, A_1, A_2 as $SPA(n,k,p,A_1,A_2)$. The allowable ranges for the parameters are $0 \le pA_1 \le 1$ and $A_2 \ge 0$. We will typically consider the 2D case so we will suppress the t in the notation when it is clear we are in 2D.

The authors show in [7] that with high probability the SPA model generates an in-degree distribution which has a power law coefficient of $1 + \frac{1}{pA_1}$.

Theorem 1.5.5 ([7]) *Consider $SPA(n,t,p,A_1,A_2)$. Then for any $k \ge 0$,*

$$E(N_{k,n}) = (1+o(1))c_k n$$

where $c_0 = \frac{1}{1+pA_2}$ and for $k > 0$, $c_k = (1+o(1))ck^{-(1+\frac{1}{pA_1})}$. Let $k_f = k_f(n) = (\frac{n}{\log^8(n)})^{\frac{pA_1}{4pA_1+2}}$. Then for $k = 0, \ldots, k_f$, w.e.p.,

$$N_{k,n} = (1+o(1))c_k n.$$

From Theorem 1.5.5 we have that $\frac{N_{k,n}}{n}$ follows a power law with coefficient $1 + \frac{1}{pA_1}$ with concentration for all values of $k = 0, 1, \ldots, k_f$.

The presence of vertices with large influence regions leads to the possibility of long links, implying that average path lengths in the SPA model should be smaller than average path lenfgths in RGG's. However, the spatial nature of the network should lead to a clustered structure, an important property of complex networks. This observation is verified by calculation in Chapter 2 and through experiment in Chapter 3.

Chapter 2

Graphlet Counts in Complex Network Models

We investigate the distribution of graphlets amongst our chosen complex network models. Specifically, we focus on the case where the model generates a graph with a linear number of edges (dn for some constant $d \in \mathbb{Z}^+$) and compute the expected number of triangles, 3-paths, and 4-cycles in each of our models. We choose to focus on the case where the models generate a linear number of edges primarily because the edges in the models grow linearly with the size of the graph for much of their parameter ranges. Also, the conventional wisdom in complex network analysis is that the number of edges grows linearly with the size of the network, or equivalently, that the average degree remains constant. There are some that disagree with this conventional wisdom. In [89], Leskovec *et al.* argues that the number of edges grows super linearly or according to a *densification power law* as the size of the graph increases. In Theorem 2.1.1 in Section 2.1 we show for graphs with an expected degree distribution with $\gamma > 2$, the number of edges grows linearly with n and when $\gamma \leq 2$ the number of edges grows super linearly. Analysis of complex networks has shown that power law coefficients in both ranges are possible [8, 25]. Furthermore, the result of Theorem 2.1.1 is for a graph whose expected degree distribution follows the power law precisely, a requirement that is never met in complex networks. In conclusion, it seems that perhaps both camps might be correct, but in this thesis we focus on the linear number of edges cases.

The conclusion from our analysis is that even though some of the models share various similar properties such as a power law degree distribution and the small world property, they generate very different frequencies of graphlets. We begin with an overview and discussion of our results from this chapter in Section 2.1. In the following sections, we provide the details for the proofs whose results are summarized in Figure 2.1.

2.1 Comparing Subgraph Counts Amongst the Models

Some of our models such as the preferential and uniform attachment models always generate graphs with a linear number of edges while others only generate a linear number of edges for a specific range of their parameters. For the pure copy model $Copy(n, p, 0, G_{n_0})$ we require that $p = \frac{1}{2}$. For the more general copy model $Copy(n, p, d, G_{n_0})$ we require that $0 < p < \frac{1}{2}$. For the directed copy model $DCopy(n, p, q, G_{n_0})$ we need $q > 0$ and $0 < p < 1$. For the random geometric graph $Geo([0, 1]^t, d_\infty, n, r, p)$ we require that $r = \Theta((\frac{1}{n})^{\frac{1}{t}})$. For the spatial preferred attachment model we require that $0 < pA_1 < 1$. The asymptotic growth for the number of triangles, 3-paths and 4-cycles for these ranges are shown in Figure 2.1. Additionally we include the coefficient of the power law in this range where applicable.

A general comment on the information contained in Figure 2.1 is that the models generate vastly different concentrations of triangles, 3-paths and 4-cycles. These results verify the observation in Chapter 3 that graphlets differentiate the models very well. Also, as expected, we see that the complex network models generate graphs which experience a much higher degree of clustering than the Random Graph. In the Random Graph, as $n \to \infty$, a constant number of triangles form and a linear number of 3-paths form. As there are so few triangles as compared to 3-paths, there is an extremely low amount of clustering present in the Random Graph.

Model	K_3	P_3	C_4	Power Law
Max	$\Theta(n^{\frac{3}{2}})$	$\Theta(n^2)$	$\Theta(n^2)$	N/A
Min	0	0	0	N/A
ER	$\Theta(1)$	$\Theta(n)$	$\Theta(1)$	x
PA, $\alpha = 0$	$\Theta(ln(n)^3)$	$\Theta(nln(n))$	$\Theta(ln(n)^4)$	$\gamma = 3$
PA, $\alpha > 0$	$\Theta(ln(n))$	$\Theta(n)$	?	$\gamma \in (3, \infty)$
UA	$\Theta(ln(n))$	$\Theta(n)$	$\Theta(ln(n))$	x
Pure Copy, $p = \frac{1}{2}$	$\Theta(n^{\frac{3}{4}})$	$\Theta(n^{\frac{5}{4}})$	$\Theta(n^{\frac{5}{4}})$	x
Copy, $0 < p < \sqrt{2} - 1$	$\Theta(n^{3p^2})$	$\Theta(n)$	$\Theta(n)$	$\gamma \in (3, \infty)$
Copy, $p = \sqrt{2} - 1$	$\Theta(n^{3p^2})$	$\Theta(nln(n))$	$\Theta(nln(n))$	$\gamma = 3$
Copy, $\sqrt{2} - 1 < p < \frac{1}{2}$	$o(n)$	$\Theta(n^{4-\gamma})$	$\Theta(n^{4-\gamma})$	$\gamma \in (2, 3)$
DCopy, $q > 0, 0 < p < \frac{1}{2}$	$\Theta(n)$	$\Theta(n)$	?	?
DCopy, $q > 0, p = \frac{1}{2}$	$\Theta(n)$	$\Theta(nln(n))$	?	?
DCopy, $q > 0, \frac{1}{2} < p < 1$	$\Theta(n)$	$\Theta(n^{2p})$	?	?
GEO-tD	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$	x
SPA2D, $0 < pA_1 < \frac{1}{2}$	$\Theta(n)$	$\Theta(n)$	$\Omega(n)$	$\gamma \in (3, \infty)$
SPA2D, $pA_1 = \frac{1}{2}$	$\Theta(n)$	$\Theta(nln(n))$	$\Omega(n)$	$\gamma = 3$
SPA2D, $\frac{1}{2} < pA_1 < \frac{2}{3}$	$\Theta(n)$	$\Theta(n^{2pA_1})$	$\Omega(n)$	$\gamma \in (2.5, 3)$
SPA2D, $\frac{2}{3} \leq pA_1 < 1$	$\Theta(n)$	$\Theta(n^{2pA_1})$	$\Omega(n^{3pA_1-1})$	$\gamma \in (2, 2.5)$

Figure 2.1: A comparison of the orders of magnitude of the expected number of triangles, 3-paths and 4-cycles for a linear number of edges.

There are several interesting observations which follow from comparing the information in Figure 2.1. One such observation is the effect the power law coefficient has on the number of edges and 3-paths in a graph. The results we state in Theorems 2.1.1 and 2.1.2 are for graphs which have an expected power law degree distribution. That is, if $N_{k,n}$ is the number of vertices of degree k in a graph of size n, then $E(N_{k,n}) = Ck^{-\gamma}n$. Dealing with an expected power law degree distribution allows us to smooth over some of the difficulties in defining an actual power law degree distribution as discussed in Section 1.4.2.

Theorem 2.1.1 *Let $N_{k,n}$ be the number of vertices in G_n of degree k. Suppose that G_n has an expected power law degree distribution $E(N_{k,n}) = ck^{-\gamma}n$ and let e_n be the* *number of edges in G_n. Then*

$$E(e_n) = \begin{cases} \frac{c}{4-2\gamma}n^{3-\gamma} + O(n) & \text{if } 1 < \gamma < 2 \\ \frac{c}{2}nln(n) + O(n) & \text{if } \gamma = 2 \\ O(n) & \text{if } \gamma > 2 \end{cases}$$

Proof We have $e_n = \frac{1}{2}\sum_{i=1}^{n} deg_n(v_i) = \frac{1}{2}\sum_{k=1}^{n-1} kN_{k,n}$. Now taking expectation we can write $E(e_n) = \frac{1}{2}\sum_{k=1}^{n-1} kE(N_{k,n}) = \frac{cn}{2}\sum_{k=1}^{n-1} k^{1-\gamma}$. Using Lemma 1.3.6 to approximate this sum, we have three regions for our solution: $1 < \gamma < 2$, $\gamma = 2$, and $\gamma > 2$.

Case 1: $1 < \gamma < 2$ Using Lemma 1.3.6 we have,

$$\begin{aligned} E(e_n) &= \frac{cn}{2}\sum_{k=1}^{n-1} k^{1-\gamma} \\ &= \frac{cn}{2}\Big(\int_1^n k^{1-\gamma}dk + O(1)\Big) \\ &= \frac{cn}{2}[\frac{n^{2-\gamma}}{2-\gamma} + O(1)] \\ &= \frac{c}{4-2\gamma}n^{3-\gamma} + O(n). \end{aligned}$$

Case 2: $\gamma = 2$ Using Lemma 1.3.6 we have,

$$\begin{aligned} E(e_n) &= \frac{cn}{2}\sum_{k=1}^{n-1} k^{-1} \\ &= \frac{cn}{2}\Big(\int_1^n k^{-1}dk + O(1)\Big) \\ &= \frac{cn}{2}[ln(n) + O(1)] \\ &= \frac{c}{2}nln(n) + O(n) \end{aligned}$$

Case 3: $\underline{\gamma > 2}$ Using Lemma 1.3.6 we have,

$$\begin{aligned} E(e_n) &= \frac{cn}{2}\sum_{k=1}^{n-1} k^{1-\gamma} \\ &= \frac{cn}{2}\Big(\int_1^n k^{1-\gamma}dk + O(1)\Big) \\ &= \frac{cn}{2}[\frac{n^{2-\gamma}}{2-\gamma} + O(1)] \\ &= \frac{c}{4-2\gamma}n^{3-\gamma} + O(n) \\ &= O(n). \end{aligned}$$

Note that if $\gamma > 2$ then $n > n^{3-\gamma}$. □

In Theorem 2.1.1, the graph G is undirected. An equivalent theorem for directed graphs with an expected power law in or out degree distribution can be obtained by removing the factor of $\frac{1}{2}$ from $\sum_{i=1}^{n} deg_n(v_i)$. In either case, if the power law coefficient is greater than two, we have a linear number of edges.

Using a similar proof as the one in Theorem 2.1.1, we can show that the power law coefficient also dictates the number of induced triangles and 3-paths.

Theorem 2.1.2 *Let G_n be a graph on n vertices with an expected degree distribution satisfying $E(N_{k,n}) = ck^{-\gamma}n$ and let $X_n = ind(P_3, G_n)$ and $Y_n = ind(K_3, G_n)$.*

$$E(X_n) + 3E(Y_n) = \begin{cases} \frac{c}{6-2\gamma}n^{4-\gamma} + O(n) & if \;\; 2 < \gamma < 3 \\ \frac{c}{2}nln(n) + O(n) & if \;\; \gamma = 3 \\ O(n) & if \;\; \gamma > 3 \end{cases}$$

Proof Recall the relation $X_n + 3Y_n = \sum_{i=1}^{n} \binom{deg(v_i)}{2}$. Simplifying this we can obtain $X_n + 3Y_n = \frac{1}{2}\sum_{k=1}^{n-1} k^2 N_{k,n} - e_n$. Applying expectation we obtain $E(X_n) + 3E(Y_n) = \frac{cn}{2}\sum_{k=1}^{n-1} k^{2-\gamma} - E(e_n)$. We know $E(e_n)$ from Theorem 2.1.1. What needs to be computed is $\sum_k k^{2-\gamma}$. Using Lemma 1.3.6 to approximate this sum, we have three regions for our solution: $2 < \gamma < 3$, $\gamma = 3$, and $\gamma > 3$.

Case 1: $2 < \gamma < 3$ Using Lemma 1.3.6 we can approximate $\frac{cn}{2}\sum_{k=1}^{n-1} k^{2-\gamma}$ as

$$\begin{aligned} \frac{cn}{2}\sum_{k=1}^{n-1} k^{2-\gamma} &= \frac{cn}{2}\Big(\int_1^n k^{2-\gamma}dk + O(1)\Big) \\ &= \frac{cn}{2}[\frac{n^{3-\gamma}}{3-\gamma} + O(1)] \\ &= \frac{c}{6-2\gamma}n^{4-\gamma} + O(n). \end{aligned}$$

From Theorem 2.1.1 we have that $E(e_n) = O(n)$. Combining this with the above gives $E(X_n) + 3E(Y_n) = \frac{c}{6-2\gamma}n^{4-\gamma} + O(n)$.

Case 2: $\gamma = 3$ Using Lemma 1.3.6 we can approximate $\frac{cn}{2}\sum_{k=1}^{n-1} k^{-1}$ as

$$\begin{aligned} \frac{cn}{2}\sum_{k=1}^{n-1} k^{-1} &= \frac{cn}{2}\Big(\int_1^n k^{-1}dk + O(1)\Big) \\ &= \frac{cn}{2}[ln(n) + O(1)] \\ &= \frac{c}{2}nln(n) + O(n) \end{aligned}$$

From Theorem 2.1.1 we have that $E(e_n) = O(n)$ when $\gamma = 3$. Combining this with the above gives $E(X_n) + 3E(Y_n) = \frac{c}{2}nln(n) + O(n)$.

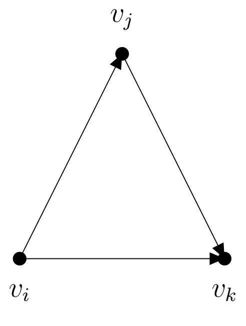
Case 3: $\gamma > 3$ Using Lemma 1.3.6 we can approximate $\frac{cn}{2}\sum_{k=1}^{n-1} k^{2-\gamma}$ as

$$\frac{cn}{2}\sum_{k=1}^{n-1} k^{2-\gamma} = \frac{cn}{2}\Big(\int_1^n k^{2-\gamma}dk + O(1)\Big)$$

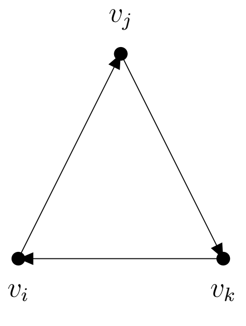
$$\begin{aligned} &= \frac{cn}{2}[\frac{n^{3-\gamma}}{3-\gamma} + O(1)] \\ &= \frac{c}{6-2\gamma}n^{4-\gamma} + O(n) \\ &= O(n). \end{aligned}$$

Note that if $\gamma > 3$ then $n > n^{4-\gamma}$. From Theorem 2.1.1 we have that $E(e_n) = O(n)$ when $\gamma > 3$. Combining this with the above gives $E(X_n) + 3E(Y_n) = O(n)$. □

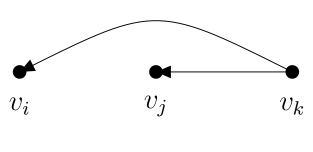
There is also a directed version of Theorem 2.1.2. We will state the directed result in terms of an expected in-degree distribution which is a power law. In this case we write $\sum_{i=1}^{n} \binom{deg^{-}(v_i)}{2}$ as the sum of directed 3-paths and directed triangles. In this case, not every directed 3-path or triangle would be counted, but only those with a vertex with in-degree two. If we only allow at most one edge between vertices and do not allow bi-directional edges then there is only one directed 3-path and one directed triangle which have a vertex of in-degree 2. Below we show all the simple directed triangles and 3-paths.



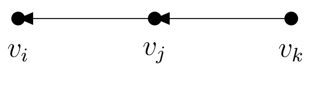
K_3^1



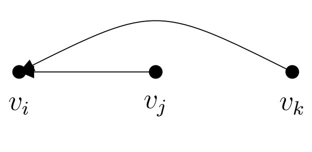
K_3^2



P_3^1



P_3^2



P_3^3

Theorem 2.1.3 *Let $N^-_{k,n}$ be the number of vertices with in-degree k in G_n. Suppose G_n is a graph on n vertices whose in-degree distribution satisfies $E(N^-_{k,n}) = ck^{-\gamma}n$ and let $X_n = ind(P^3_3, G_n)$ and $Y_n = ind(K^1_3, G_n)$.*

$$E(X_n) + E(Y_n) = \begin{cases} \frac{c}{6-2\gamma}n^{4-\gamma} + O(n) & \text{if } 2 < \gamma < 3 \\ \frac{c}{2}nln(n) + O(n) & \text{if } \gamma = 3 \\ O(n) & \text{if } \gamma > 3 \end{cases}$$

Proof The proof proceeds in an identical manner as the proof of Theorem 2.1.2 except that we write $X_n + Y_n = \sum_{i=1}^{n} \binom{deg^-_n(v_i)}{2}$ instead of $X_n + 3Y_n = \sum_{i=1}^{n} \binom{deg_n(v_i)}{2}$. □

In Figure 2.1 on 38 we have three models which generate graphs with a power law coefficient equal to 3: the original PA model, the undirected Copy model with $p = \sqrt{2} - 1 \sim 0.4142$ and $d > 0$, and the SPA model with $pA_1 = \frac{1}{2}$. Note that the SPA model has an in-degree distribution which is a power law. From Theorem 2.1.2, all three models asymptotically have the same number of 3-paths up to the power law constant c. What is interesting is that each of these models displays a different level of clustering. The original PA model displays the least clustering as it generates $\Theta(ln(n)^3)$ triangles. The Copy model generates $\Theta(n^x)$ triangles where $x \sim 0.51$, which is far more than the PA model. The SPA model generates the most with $\Theta(n)$ triangles. The SPA model is the most clustered in this case. Its interesting that graphs with a similar degree distribution can generate vastly differing clustered structure. The number of 4-cycles in each of these models are also vastly different. The PA model has the least with $\Theta(ln(n)^4)$ followed by the SPA model with $\Theta(n)$ and the Copy model has the most with at least $\Omega(nln(n))$. The copying mechanism is what is responsible for the large number of 4-cycles in the Copy model. As is described in Theorem 2.5.18 on 103, a new 4-cycle is formed at time $n + 1$ for every

3-path which is copied at time $n+1$, which is why the number of 4-cycles is bounded below by the number of 3-paths in this case.

The undirected Copy model with $\sqrt{2}-1 < p < \frac{1}{2}$ and $d > 0$ and the SPA model with $\frac{1}{2} < pA_1 < 1$ also generate graphs with a power law in-degree distribution with $\gamma \in (2,3)$. In Figure 2.1 and as a consequence of Theorem 2.1.2, both models have the $\Theta(n^{4-\gamma})$ 3-paths. Note that in the Copy model the power law coefficient runs from 2 to 3 as p goes from $\frac{1}{2}$ to $\sqrt{2}-1$ and in the SPA model the power law coefficient runs from 2 to 3 as pA_1 runs from $\frac{1}{2}$ to 1. Again, the SPA model is the more clustered of the two models generating $\Theta(n)$ triangles while the Copy model generates $o(n)$ triangles. Unlike the $\gamma = 3$ case, the Copy model and SPA model generate the same number of 4-cycles when $\gamma \in (2,3)$. From Figure 2.1, both generate $\Omega(n^x)$ 4-cycles with $x \in (1,2)$.

The PA model with $\alpha > 0$, the Copy model with $0 < p < \sqrt{2}-1$, and the SPA model with $0 < pA_1 < \frac{1}{2}$ all generate power law degree distributions with $\gamma \in (3,\infty)$. From Figure 2.1 and as a consequence of Theorem 2.1.2, all three models generate $O(n)$ 3-paths. The PA model is the least clustered generating $O(ln(n))$ triangles. The copy model has significantly more with $\Theta(n^x)$ triangles where $x \in (0, .51)$. The SPA model is the most clustered with $\Theta(n)$ triangles. The Copy model and the SPA both generate $O(n)$ 4-cycles in this range.

The most important take away from our discussion above, is that the power law coefficient determines the number of 3-paths but generally gives no indication of the number of triangles present in the graph. The SPA model is by far the most clustered of the 3 models we study which generate power law degree distributions. The spatial nature of the SPA model is without a doubt the reason for the clustered structure of the model. Note that for $0 < pA_1 < \frac{1}{2}$, from Figure 2.1, we see that the SPA model and the RGG both generate $O(n)$ 3-paths, triangles and 4-cycles. This indicates that for values of $pA_1 < \frac{1}{2}$, the spatial mechanism of the SPA model is perhaps more dominant than the preferential attachment mechanism while for $p \geq \frac{1}{2}$ the preferential attachment mechanism is the more dominant of the two.

It is also interesting to compare the PA model and the UA model. In both models, a new vertex v_n is added at time n and d edges are added from v_n to already present vertices. In the PA model, the endpoints of these edges are chosen preferentially so that vertices of higher degree are more likely to be picked. In the UA model, the endpoints of these edges are chosen uniformly at random. Comparing the number of triangles generated in these models in Figure 2.1, we see that the original PA model generates $\Theta(ln(n)^3)$ triangles while the UA model generates $\Theta(ln(n))$ triangles. We should not be surprised that preferential attachment leads to a more clustered structure as compared to uniform attachment. It is interesting to note that once we distort the preferential mechanism even slightly in $PA(n, d, \alpha)$, the number of triangles asymptotically grows as $\Theta(ln(n))$ which is the same order of growth as the uniform attachment model. You might have expected that the number of triangles in $PA(n, d, \alpha)$ would have reduced gradually to $\Theta(ln(n))$ as α increased as increasing values of α have a greater distortion on the preferential attachment mechanism in $PA(n, d, \alpha)$. It turns out not to be the case. The generalized PA model however does maintain a power law degree distribution while the UA model has a Poisson degree distribution [30].

2.2 Maximum and Minimum Subgraph Counts

To supplement the comparison of the subgraph counts for our selected models we include the maximum and minimum number of copies of these subgraphs. Specifically, we are interested in finding the maximum and minimum number of triangles, 3-paths and 4-cycles which can appear in n- vertex graphs with dn edges for some positive constant $d \in \mathbb{Z}^+$.

2.2.1 Maximizing the Number of Triangles and 3-Paths Given a Linear Number of Edges

Recall the relation $ind(P_3, G) + 3ind(K_3, G) = \sum\limits_{v \in V(G)} \binom{deg(v)}{2}$. We can write,

$$\begin{aligned} ind(P_3, G) + 3ind(K_3, G) &= \sum_{v \in V(G)} \binom{deg(v)}{2} \\ &= \sum_{v \in V(G)} \frac{deg(v)(deg(v)-1)}{2} \\ &\leq \sum_{v \in V(G)} \frac{deg(v)^2}{2}. \end{aligned}$$

Using this, we can give an upper bound for the maximum number of triangles and 3-paths by considering the maximum value of $\sum_{v \in V(G)} deg(v)^2$. The problem of maximizing $\sum_{v \in V(G)} deg(v)^2$ given G with n vertices and m edges was done by Ahlswede and Katan in [5].

Theorem 2.2.1 ([5]) *Let G be a graph with n vertices and m edges. The graph G which maximizes $\sum\limits_{v \in V(G)} \binom{deg(v)}{2}$ is the quasi-star or the quasi-complete graph. Furthermore, if $0 \leq m < \frac{1}{2}\binom{n}{2} - \frac{n}{2}$ then G is the quasi-star and if $\frac{1}{2}\binom{n}{2} + \frac{n}{2} < m \leq \binom{n}{2}$ then G is the quasi-complete graph.*

Let k and j be the unique integers such that $m = \binom{k}{2} + j$ for $0 \leq j \leq k-1$. The **quasi-complete graph** on n vertices and m edges denoted by $QC(n, m)$ is formed by creating a k-clique and one vertex of degree j whose endpoints all lie within the k-clique. In other words, given n vertices and m edges, the quasi-complete graph is formed by making the largest possible k-clique possible and taking the remaining edges all incident to one vertex outside the clique while making the endpoint inside the clique. It is not possible to have more than $k-1$ edges not in the clique, otherwise a clique of size $k+1$ can be formed. The remaining $n-k-1$ vertices are isolated.

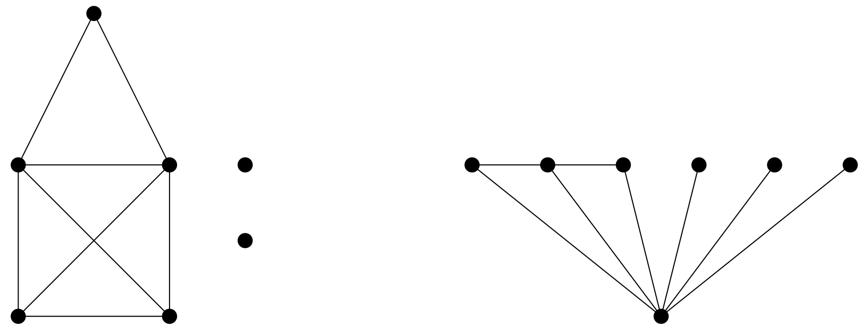


Figure 2.2: $QC(7,8)$ and $QS(7,8)$

Let c and j be the unique integers so that $m = c(n-1) - \binom{c}{2} + j$ where $0 \leq j < n-c-1$. The **quasi-star graph** on n vertices and m edges denoted by $QS(n,m)$ is formed by creating c universal vertices and adding the remaining j edges incident to any one non-universal vertex. In other words, given n vertices and m edges, the quasi star graph is formed by creating the maximum number of universal vertices. As stated, the number of edges in the quasi-star graph is $m = c(n-1) - \binom{c}{2} + j$ where $0 \leq j < n-c-1$. Counting the edges, we get $c(n-1)$ edges incident to a universal vertex, but we count each edge between universal vertices twice so we subtract $\binom{c}{2}$. The number of edges left over can't exceed $n-c-2$ otherwise another universal vertex would be created. It is interesting to note that the $QS(n,m)$ is isomorphic to the complement of $QC(n, \binom{n}{2} - m)$. We show examples of $QC(7,8)$ and $QS(7,8)$ in Figure 2.2.

Since we are only interested in graphs with dn edges, from Theorem 2.2.1, the quasi-star is the graph which maximizes $\sum_{v \in V(G)} deg(v)$. It is shown in [126] that the maximum number of triangles in a graph with n vertices and m edges occurs in $QC(n,m)$. Using this result we can determine the maximum number of triangles in a graph with dn edges.

Theorem 2.2.2 *Let G be a graph with n vertices and $m = nd$ edges for some integer d. Then the maximum number of triangles in such a graph is $\frac{(2d)^{\frac{3}{2}}}{6} n^{\frac{3}{2}} + O(n)$.*

Proof The result of [126] implies that the maximum number of triangles occurs in $QC(n, nd)$. Recall that in $QC(n,m)$ we can write $m = \binom{k}{2} + j$ for $0 \leq j \leq k-1$. It is easy to count the number of triangles in $QC(n,m)$ is $\binom{k}{3} + \binom{j}{2}$: each set of 3 vertices in the k-clique give a triangle and each pair of the j extra edges gives a triangle. To determine the number of triangles in our case we must compute the value of k. To this end we set $nd = \binom{k}{2} + j$. We can express $k = \sqrt{2(dn-j)} + O(1)$. To show this we found upper and lower bounds for k. For the lower bound $\frac{k^2}{2} \geq \binom{k}{2} = dn - j$ implying that $k \geq \sqrt{2(dn-j)}$. For the upper bound $\frac{(k-1)^2}{2} \leq \binom{k}{2} = dn - j$ implying that $k \leq \sqrt{2(dn-j)} + 1$.

Using the fact that $j = O(k)$ we have,

$$\begin{aligned}
ind(K_3, QC(n,m)) &= \binom{k}{3} + \binom{j}{2} \\
&= \frac{k^3}{6} + O(k^2) \\
&= \frac{(\sqrt{2(nd-j)} + O(1))^3}{6} + O((\sqrt{2(nd-j)} + 1)^2) \\
&= 2^{\frac{3}{2}} \frac{((nd-j)^{\frac{3}{2}} + O(n))}{6} + O(n) \\
&= \frac{(2d)^{\frac{3}{2}}}{6} n^{\frac{3}{2}} + O(n).
\end{aligned}$$

□

For the maximum number of 3-paths we use the upper bound of $\frac{1}{2}\sum_{v \in V(G)} deg(v)^2$. We can write $ind(P_3, G) + 3ind(K_3, G) = \frac{1}{2}\sum_{v \in V(G)} deg(v)^2$. Since we are only interested in the case where G has a linear number of edges, we know from [5] that $\sum_{v \in V(G)} deg(v)^2$ is maximized when $G = QS(n,m)$. Therefore, an upper bound can be obtained by summing over the degrees squared in $QS(n,m)$. To obtain a lower bound, we simply count the number of 3-paths in $QS(n,m)$.

Theorem 2.2.3 *Let G be a graph with n vertices and $m = nd$ edges for some positive integer d such that $n > \binom{d}{2} + d + 1$. Then the maximum number of 3-paths in such a graph is $\frac{d}{2}n^2 + O(n)$.*

Proof To get the result we will establish a lower and an upper bound for the maximum number of 3-paths in a graph. Let G be a graph with n vertices and dn edges which achieves the maximum number of 3-paths. For the upper bound we observe that

$$\begin{aligned} ind(P_3, G) &\leq \frac{1}{2} \sum_{v \in V(G)} deg(v)^2 - 3ind(K_3, G_n) \\ &\leq \frac{1}{2} \sum_{v \in V(G)} deg(v)^2 \\ &\leq \frac{1}{2} \sum_{v \in V(QS(n,nd))} deg(v)^2. \end{aligned}$$

Recall that in $QS(n, m)$ we can write $m = c(n-1) - \binom{c}{2} + j$ for $0 \leq j < n - c - 1$. There are c vertices of degree $n - 1$, j vertices of degree $c + 1$, $n - c - j - 1$ vertices of degree c and one vertex of degree $j + c$.

One vertex of degree j and $n - c - 1$ vertices of degree c. Therefore we have that $\frac{1}{2} \sum_{v \in QS(n,m)} deg(v)^2 = \frac{c(n-1)^2}{2} + \frac{(n-c-j-1)c^2}{2} + j\frac{(c+1)^2}{2} + \frac{(j+c)^2}{2}$. To determine the appropriate parameters in the case $m = dn$ we set $dn = c(n-1) - \binom{c}{2} + j$. Observe that if n is large enough than we will always have $c = d$ and $j = d + \binom{d}{2}$. Suppose that $c = d$. Than we have $n - 1 + n - 2 + \ldots + n - (d - 1) = dn - \binom{d}{2}$ edges incident to our d isolated vertices. Therefore, we have $j = \binom{d}{2}$ remaining edges. Therefore, we must have $\binom{d}{2} < n - d - 1$ which implies that $n > \binom{d}{2} + d + 1$ which is always possible as d is fixed. Therefore in $QS(n, dn)$ with $n > \binom{d}{2} + d + 1$, we have $c = j$ and $j = \binom{d}{2} + d$.

$$\begin{aligned}\frac{1}{2}\sum_{v\in QS(n,nd)} deg(v)^2 &= \frac{d(n-1)^2}{2}+\frac{(n-2d-\binom{d}{2}-1)d^2}{2}+(\binom{d}{2}+d)\frac{(d+1)^2}{2} \\ &+ \frac{2d+\binom{d}{2}}{2} \\ &= \frac{d}{2}n^2+O(n).\end{aligned}$$

For the lower bound we count the number of 3-paths in $QS(n,nd)$. Observe that if G is the graph with nd edges which has the maximum number of 3-paths then it follows that $ind(P_3, QS(n,nd)) \leq ind(P_3, G)$. We first observe that if $j = 0$, then each universal vertex along with any pair of the $n-c$ non-universal vertices induces a 3-path giving $c\binom{n-c}{2}$ 3-paths. If $j > 0$, then each additional edge added destroys exactly c 3-paths. Any pair of the j extra edges forms a 3-path so that $\binom{j}{2}$ 3-paths are created. Overall we have that $ind(P_3, QS(n,m)) = c\binom{n-c}{2} - jc + \binom{j}{2}$. When $m = dn$ and $c = d$ and $j = \binom{d}{2} + d$ we have

$$\begin{aligned}ind(P_3, QS(n,nd)) &= d\binom{n-d}{2}-(\binom{d}{2}+d)d+\binom{\binom{d}{2}+d}{2} \\ &= d\frac{(n-d)(n-d+1)}{2}+O(1) \\ &= \frac{d}{2}n^2+O(n).\end{aligned}$$

Since the upper and lower bounds have the same order, we conclude that the maximum number of 3-paths in a graph with dn edges with $n > \binom{d}{2} + d + 1$ is $\frac{d}{2}n^2 + O(n)$. □

2.2.2 Minimum Number of Triangles and 3-Paths Given a Linear Number of Edges

To determine the minimum number of triangles in a graph with n vertices and $m = dn$ vertices we consider Turán's Theorem [121].

Theorem 2.2.4 (Turán's Theorem) *Let G be a graph without a copy of K_{r+1}. Then G can have at most $(1 - \frac{1}{r})\frac{n^2}{2}$ edges.*

The graph which is K_{r+1}-free with the maximum number of edges is called the *Turán graph* $T(n, r)$. In the case of K_3, the Turán graph is $K_{\frac{n}{2},\frac{n}{2}}$ if n is even and is $K_{\lfloor\frac{n}{2}\rfloor,\lfloor\frac{n}{2}\rfloor+1}$ if n is odd. If n is even, then $T(n, 3)$ has $\frac{n^2}{4}$ edges and if n is odd, then $T(n, 3)$ has $\frac{(n-1)(n+1)}{4}$. In the next result we argue that the minimum number of triangles in a graph with dn such that $n \geq 4d$ is 0.

Theorem 2.2.5 *If $d \in \mathbb{Z}^+$ and $n \geq 4d$ then the minimum number of triangles in a graph $G \in \mathcal{G}_{n,dn}$ is 0.*

Proof To create a triangle free graph we follow the procedure of Turán and split the vertex set into two sets of roughly equal size and proceed to create a bipartite graph. Suppose that n is even. Then by Turán's Theorem, as long as $dn \leq \frac{n^2}{4}$, we are able to place edges in a bipartite graph with both partition sets containing $\frac{n}{2}$ vertices, without forming a triangle. Thus if $n \geq 4d$ this is possible. Now suppose n is odd. Then by Turán's Theorem, as long as $dn \leq \frac{(n-1)(n+1)}{2}$, we are able to place edges in a bipartite graph with partition sets of size $\frac{n-1}{2}$ and $\frac{n+1}{2}$ without forming a triangle. In this case, we must have that $n \geq 4d - \frac{1}{n} \geq 4d$.

□

In this next theorem we argue that the minimum number of 3-paths in a graph with dn edges tends to 0 as $n \to \infty$.

Theorem 2.2.6 *For $d \in \mathbb{Z}^+$, for sufficiently large n, the minimum number of 3-paths in a graph $G \in \mathcal{G}_{n,dn}$ is 0.*

Proof Consider the following construction which is similar in nature to the quasi-complete graph. We take our dn edges and form the largest possible k-clique. With the remaining $j = nd - \binom{k}{2}$ edges we add each edge as an isolated K_2 component. This may not be possible for every n and dn but we argue that for large enough n it is possible. We have $nd = \binom{k}{2} + j$ for $0 \leq j < k$ where $k = \sqrt{2(nd - j)} + O(1)$. Therefore we have $j = O(k) = O(\sqrt{n})$ left over edges to distribute amongst $n - k = n - O(\sqrt{n}) = O(n)$ vertices. □

2.2.3 Maximum and Minimum Number of 4-Cycles Given a Linear Number of Edges

Theorem 2.2.7 *For $d \in \mathbb{Z}^+$ and $n \geq 4d$, the maximum number of 4-cycles in a graph $G \in \mathcal{G}_{n,dn}$ is $(1 + o(1))\frac{d^2}{4}n^2$.*

Proof For the upper bound, we observe that each pair of independent edges contributes to at most one 4-cycle. Each 4-cycle contains exactly two pairs of independent edges. Therefore two times the number of 4-cycles is less than or equal to the number of pairs of independent edges, which is less than or equal to the number of pairs of edges. We can write $ind(C_4, G) \leq \frac{1}{2}\binom{dn}{2} = \frac{(dn)^2}{4} - \frac{dn}{4}$.

For the lower bound, we count the number of 4-cycles in the complete bipartite graph $K_{t,t}$. This graph has t^2 edges. Therefore the size of each bipartition set is $t = \lfloor\sqrt{dn}\rfloor$. In this graph, each pair of independent edges forms exactly one 4 cycle. There are exactly $\binom{t^2}{2} - t\binom{t}{2} = \frac{t^4}{2} + O(t^3)$ pairs of independent edges. If we count the number of 4-cycles by pairs of independent edges than each 4-cycle gets counted twice. Therefore $ind(C_4, K_{t,t}) = \frac{t^4}{4} + O(t^3)$. For a linear number of edges we have $ind(C_4, G) \geq \frac{(\lfloor\sqrt{dn}\rfloor)^4}{4} + O((\sqrt{dn})^3) = \frac{d^2}{4}n^2 + O(n^{\frac{3}{2}})$.

Combining the upper and lower bounds we have a maximum number of 4-cycles of $(1+o(1))\frac{d^2}{4}n^2$. □

For the minimum number of 4-cycles we note that each 4-cycle contains 4 induced 3-paths. From Theorem 2.2.6 we can conclude that asymptotically the minimum number of 4-cycles in a graph with dn edges is 0.

2.3 Erdős-Rényi Model

Though we do not consider the Random Graph to be a suitable model for complex networks, it is important as a point of comparison, to give the subgraph counts for this model. A formula for counting the expected induced subgraphs in $ER(n,p)$ was given by Bollobás in [27].

Theorem 2.3.1 ([27]) *Let $G_n = ER(n,p)$ and let H be a graph with $k \leq n$ vertices and m edges. If $X_n = ind(H, G_n)$ then,*

$$E(X_n) = \frac{|H|!}{|Aut(H)|}p^m(1-p)^{\binom{k}{2}-m}\binom{n}{k}. \tag{2.1}$$

With Equation 2.1 we can compute the expected triangle, 3-path and 4-cycle counts in $ER(n,p)$ with dn edges. We know that the expected number of edges in $ER(n,p)$ is $p\binom{n}{2}$. To determine the value of p which gives a linear number of edges we solve $nd = p\binom{n}{2}$ to get $p = \frac{2d}{n-1}$.

Theorem 2.3.2 *Let $G_n = ER(n, \frac{2d}{n-1})$.*

H	$E(ind(H,G_n))$
K_3	$\frac{4}{3}d^3 + O(\frac{1}{n})$
P_3	$2d^2n + O(1)$
C_4	$2d^4 + O(\frac{1}{n})$

Proof From Equation 2.1 since $|Aut(K_3)| = 6$, we have $E(ind(K_3, G_n)) = p^3\binom{n}{3}$. Evaluating at $p = \frac{2d}{n-1}$ we get,

$$\begin{aligned} E(ind(K_3, G_n)) &= (\frac{2d}{n-1})^3(\frac{n^3}{6} + O(n^2)) \\ &= \frac{4}{3}d^3 + O(\frac{1}{n}) \end{aligned}$$

From Equation 2.1 since $|Aut(P_3)| = 2$, we have $E(ind(P_3, G_n)) = 3p^2(1-p)\binom{n}{3}$. Evaluating at $p = \frac{2d}{n-1}$ we get,

$$\begin{aligned} E(ind(P_3, G_n)) &= 3(\frac{2d}{n-1})^2(1 - \frac{2d}{n-1})(\frac{n^3}{6} + O(n^2)) \\ &= 2d^2n + O(1) \end{aligned}$$

From Equation 2.1 since $|Aut(C_4)| = 8$, we have $E(ind(C_4, G_n)) = 3p^4(1-p)^2\binom{n}{4}$. Evaluating at $p = \frac{2d}{n-1}$ we get,

$$\begin{aligned} E(ind(C_4, G_n)) &= 3(\frac{2d}{n-1})^4(1 - \frac{2d}{n-1})^2(\frac{n^4}{24} + O(n^3)) \\ &= 2d^4 + O(\frac{1}{n}) \end{aligned}$$

□

2.4 Preferential Attachment Model

The expected number of injective subgraphs in the original PA model was computed by Bollobás and Riordan in [116] and in the generalized PA model by Eggemann and Noble in [59]. In both these papers, a general method for counting the number of injective copies of a subgraph is given. The goal of these papers was to compute the injective subgraph counts of K_3 and P_3 so that the clustering coefficient (see Equation 1.3) of $PA(n, d, \alpha)$ could be computed. We begin by summarizing these results. We also provide a simple argument to extend the result for the number of injective 3-paths in $PA(n, d, \alpha)$ to the number of induced 3-paths in $PA(n, d, \alpha)$. We also introduce a modified version of the original PA model and count the number of triangles and 4-cycles in this model. We will argue that our simplified model behaves in a similar way as the original PA model by showing that asymptotically the growth of the number of triangles and 3-paths coincide.

We now state the results of [116] and [59] for the expected number of injective copies of K_3 and P_3. We begin with the K_3 count. We state the result as the number of induced K_3's instead of the number of injective K_3's as both quantities are the same.

Theorem 2.4.1 ([116],[59]) *Consider the preferential attachment model $G_n = PA(n, d, \alpha)$ and let $X_n = ind(K_3, G_n)$. Then,*

$$E(X_n) = \begin{cases} (1 + o(1))\frac{d(d-1)(d+1)}{48} ln(n)^3 & if\, \alpha = 0 \\ \left(d(d-1)\frac{(1+\alpha)^2}{\alpha^2} + d(d-1)^2\frac{(1+\alpha)^3}{\alpha^2(2+\alpha)}\right) ln(n) + O(1) & if\, \alpha > 0 \end{cases}$$

Note that if $d = 1$, then no triangles are possible in the PA model. Also, one might expect since as $\alpha \to 0$ the generalized PA model tends to the original PA model the same would be so for the expected triangle counts in Theorem 2.4.1. This is not the case and the authors of [59] admit that they see no clear reason why this is not the case.

Frequency results are given for the number of injective 3-paths. We give results for the original PA model and the generalized PA model separately.

Theorem 2.4.2 ([116]) *Consider $G_n = PA(n,d)$ and let $X_n = inj(P_3, G_n)$. Then,*

$$E(X_n) = (1 + o(1))\frac{d(d+1)}{2}nln(n).$$

Furthermore, for any fixed $\epsilon > 0$, the following holds a.a.s.

$$(1-\epsilon)\frac{d(d+1)}{2}nln(n) \le X_n \le (1+\epsilon)\frac{d(d+1)}{2}nln(n)$$

Theorem 2.4.3 ([59]) *Consider $G_n = PA(n,d,\alpha)$ for $\alpha > 0$ and let $X_n = inj(P_3, G_n)$. Then,*

$$E(X_n) = \Big(\frac{2+5\alpha}{2\alpha}d^2 + \frac{2-\alpha}{2\alpha}d\Big)n + O(n^{\frac{2}{2+\alpha}})$$

Furthermore, for any $\epsilon > 0$ and $\gamma > 0$, there exists an n^* such that for all $n \ge n^*$

$$Pr(|X_n - E(X_n)| \ge n^{\frac{4+\alpha}{4+2\alpha}+\epsilon}) \le \frac{1}{n^\gamma}.$$

We now turn our attention to counting the number of induced triangles, 3-paths and 4-cycles in the preferential attachment model. We've already obtained the number of induced triangles in Theorem 2.4.1 by using the results of [116] and [59] for injective triangles. Using Lemma 1.3.2 to write $ind(P_3, G) = inj(P_3, G) + 3inj(K_3, G)$ we can use Theorems 2.4.2 and 2.4.3 to compute the expected 3-path count.

Theorem 2.4.4 *Let $G_n = PA(n,d,\alpha)$ and let $X_n = ind(P_3, G_n)$. Then,*

$$E(X_n) = \begin{cases} (1+o(1))\frac{d(d+1)}{2}nln(n) & if\ \alpha = 0 \\ \Big(\frac{2+5\alpha}{2\alpha}d^2 + \frac{2-\alpha}{2\alpha}d\Big)n + O(n^{\frac{2}{2+\alpha}}) & if\ \ \alpha > 0 \end{cases}$$

Proof $\underline{\alpha = 0}$: Using Lemma 1.3.2, Theorems 2.4.1 and 2.4.2 and linearity of expectation we can write,

$$\begin{aligned}
E(X_n) &= E(inj(P_3, G_n)) - 3E(inj(K_3, G_n)) \\
&= (1 + o(1))\frac{d(d+1)}{2} nln(n) - 3(1 + o(1))\frac{d(d-1)(d+1)}{48} ln(n)^3 \\
&= (1 + o(1))\frac{d(d+1)}{2} nln(n)
\end{aligned}$$

$\underline{\alpha > 0}$: Again, using Lemma 1.3.2, Theorems 2.4.1 and 2.4.3 and the linearity of expectation we can write,

$$\begin{aligned}
E(X_n) &= E(inj(P_3, G_n)) - 3E(inj(K_3, G_n)) \\
&= \Big(\frac{2 + 5\alpha}{2\alpha} d^2 + \frac{2 - \alpha}{2\alpha} d\Big) n + O(n^{\frac{2}{2+\alpha}}) \\
&- 3(\Big(d(d-1)\frac{(1+\alpha)^2}{\alpha^2} + d(d-1)^2\frac{(1+\alpha)^3}{\alpha^2(2+\alpha)}\Big) ln(n) + O(1)) \\
&= \Big(\frac{2 + 5\alpha}{2\alpha} d^2 + \frac{2 - \alpha}{2\alpha} d\Big) n + O(n^{\frac{2}{2+\alpha}})
\end{aligned}$$

□

The only subgraph left to be counted is the 4-cycle. The number of injective l-cycles for $l \geq 3$ in the original PA model was counted in [116].

Theorem 2.4.5 *Let $l \geq 3$ be fixed and consider $G_n = PA(n, d)$ with $d \geq 2$. Then,*

$$E(inj(C_l, G_n)) = (1 + o(1)) B_{d,l}(ln(n))^l$$

as $n \to \infty$ where $B_{d,l}$ is a positive constant depending on d and l. Furthermore, as $d \to \infty$ we have $C_{d,l} = \Theta(d^l)$.

To compute $ind(C_4, G_n)$ we could attempt a similar approach as the one used to count the number of induced 3-paths. Using Lemma 1.3.2 we can write $inj(C_4, G) = ind(C_4, G) + ind(g_7, G) + 3ind(g_8, G)$. Unfortunately such an approach would require us to compute $E(ind(g_7, G))$ and $E(ind(g_8, G))$ which would be more work than computing $E(ind(C_4, G))$ directly. In general, counting induced subgraphs in the PA model is not an easy task.

We propose a modified PA model in which it is easier to count the number of 4-cycles. Recall for $i < n + 1$ in the PA model, the probability that v_{n+1} selects v_i as an endpoint is proportional to $deg_n(v_i)$. At time n, $deg_n(v_i)$ is a random variable as its value is determined by the number of edges it receives up to time n, which depends on a random process. The purpose of this modified PA model is to make, at time $n + 1$, the probability that v_{n+1} selects v_i as an endpoint to one of its d edges a deterministic value as opposed to a random variable. Specifically, this probability is chosen so that the probability that v_{n+1} selects v_i as an endpoint is proportional to $E(deg_n(v_i))$. The expected degree for a vertex in the PA model was computed in [17], [30].

Theorem 2.4.6 *The expected degree of v_i at time n in $PA(n, d)$ is given by*

$$E(deg_n(v_i)) = d\sqrt{\frac{n}{i}}(1 + O(\frac{1}{i}))$$

.

Unfortunately, as stated in [30], the individual vertex degrees at time n are not concentrated around this expectation. However, it is shown in Theorem 1.5.2 that the number of vertices of degree k at time n is concentrated around its expected value for $k \leq n^{\frac{1}{15}}$.

We define the **modified preferential attachment model** denoted by $\overline{PA}(n, d)$ to be a graph model identical in description to the original preferential attachment model except that at time n, the probability that v_n selects v_i as an endpoint to an edge is

$$Pr(i = s) = \frac{1}{2\sqrt{s}\sqrt{n-1}}. \tag{2.2}$$

Note that if the degree of each vertex v_i in the PA model was equal to $d\frac{n}{i}$ at time n, then the probability of selecting v_i at time n as an endpoint for v_n would be

$$\begin{aligned} Pr(i = s) &= \frac{deg_{n-1}(v_s)}{2d(n-1)} \\ &= \frac{1}{2\sqrt{s}\sqrt{n-1}} \end{aligned}$$

We have defined the modified PA model so that it behaves as the PA model would behave if its individual degrees coincided with their expectation.

Computing the expected number of induced subgraphs of size k can be done using a simple approach in the modified PA model. Compute the probability that an arbitrary set of k vertices induces the subgraph, then sum over all possible sets of size k. Using the linearity of expectation we can find the expected subgraph count. We will use this approach to count the number of triangles and 4-cycles. From Theorem 2.4.1 we already know the number of triangles in the PA model. We recompute this result in the modified PA model showing that, in both cases, the number of triangles grows as $\Theta(ln(n)^3)$ as $n \to \infty$.

Before we start counting subgraphs we first compute some probabilities which we will frequently use. For our asymptotic notation, we note that for vertices $1 \leq i < j < k < l \leq n$, each of the 4 indices i, j, k, l are increasing with n. We begin with the probability that $v_i \sim v_j$. We have

$$\begin{aligned} Pr(v_i \sim v_j) &= 1 - Pr(v_i \nsim v_j) \\ &= 1 - (1 - \frac{1}{2\sqrt{i}\sqrt{j-1}})^d \\ &= 1 - (1 - \frac{d}{2\sqrt{i}\sqrt{j-1}} + O(\frac{1}{ij})) \\ &= \frac{d}{2}\frac{1}{\sqrt{i}\sqrt{j-1}} + O(\frac{1}{ij}) \end{aligned}$$

Simplifying further we could write $\frac{1}{\sqrt{j-1}} = \frac{1}{\sqrt{j(1-\frac{1}{j})}} = \frac{1}{\sqrt{j}}(1-\frac{1}{j})^{-\frac{1}{2}} = \frac{1}{\sqrt{j}} + O(\frac{1}{j})$. Applying this we have $\frac{1}{\sqrt{i}\sqrt{j-1}} = \frac{1}{\sqrt{i}\sqrt{j}} + O(\frac{1}{j\sqrt{i}})$. So,

$$Pr(v_i \sim v_j) = \frac{d}{2}\frac{1}{\sqrt{i}\sqrt{j}} + O(\frac{1}{j\sqrt{i}}) \tag{2.3}$$

Next consider the probability that there is an edge between v_k and v_i and between v_k and v_j. For this to happen, at time k both v_i and v_j have to be selected at least once as one of the d endpoints for edges from v_k. Since multiple edges are allowed, we have to consider the probability that there are $t \geq 1$ edges formed to v_i and $s \geq 1$ edges formed to v_j. We have

$$\begin{aligned} Pr(v_k \sim v_i \cap v_k \sim v_j) &= \sum_{t=1}^{d-1} \binom{d}{t} \Big(\frac{1}{2\sqrt{i}\sqrt{k-1}}\Big)^t \sum_{s=1}^{d-t} \binom{d-t}{s} \Big(\frac{1}{2\sqrt{j}\sqrt{k-1}}\Big)^s \\ &\quad (1 - \frac{1}{2\sqrt{i}\sqrt{k-1}} - \frac{1}{2\sqrt{j}\sqrt{k-1}})^{d-s-t} \\ &= \frac{d(d-1)}{4}\frac{1}{\sqrt{i}\sqrt{j}(k-1)}(1 + O(\frac{1}{\sqrt{i}\sqrt{k}} - \frac{1}{\sqrt{j}\sqrt{k}})) \\ &= \frac{d(d-1)}{4}\frac{1}{\sqrt{i}\sqrt{j}(k-1)} + O(\frac{1}{i\sqrt{j}k^{\frac{3}{2}}}) \end{aligned}$$

We can simplify this further by writing $\frac{1}{k-1} = \frac{1}{k} + O(\frac{1}{k^2})$. Using this we have $\frac{1}{\sqrt{i}\sqrt{j}(k-1)} = \frac{1}{\sqrt{i}\sqrt{j}k} + O(\frac{1}{\sqrt{i}\sqrt{j}k^2})$. Therefore,

$$Pr(v_k \sim v_i \cap v_k \sim v_j) = \frac{d(d-1)}{4}\frac{1}{\sqrt{i}\sqrt{j}k} + O(\frac{1}{\sqrt{i}\sqrt{j}k^2} + \frac{1}{i\sqrt{j}k^{\frac{3}{2}}}) \qquad (2.4)$$

The probabilities computed in Equations 2.3 and 2.4 are sufficient to count the number of triangles. To count the number of 4-cycles we require three more probabilities.

The first is $Pr(v_k \sim v_j \cap v_k \nsim v_i)$. As multiple edges are allowed, we must account for the possibility that there are $t \geq 1$ edges to v_j. We have

$$\begin{aligned} Pr(v_k \sim v_j \cap v_k \nsim v_i) &= \sum_{t=1}^{d}\binom{d}{t}(\frac{1}{2\sqrt{j}\sqrt{k-1}})^t(1 - \frac{1}{2\sqrt{i}\sqrt{k-1}} - \frac{1}{2\sqrt{j}\sqrt{k-1}})^{d-t} \\ &= \frac{d}{2\sqrt{j}\sqrt{k-1}}(1 + O(\frac{1}{2\sqrt{i}\sqrt{k}} + \frac{1}{2\sqrt{j}\sqrt{k}}) \\ &= \frac{d}{2\sqrt{j}\sqrt{k-1}} + O(\frac{1}{k\sqrt{i}\sqrt{j}}), \end{aligned}$$

Using $\frac{1}{\sqrt{k-1}} = \frac{1}{\sqrt{k}} + O(\frac{1}{k})$ we can simplify to get,

$$Pr(v_k \sim v_j \cap v_k \nsim v_i) = \frac{d}{2}\frac{1}{\sqrt{j}\sqrt{k}} + O(\frac{1}{k\sqrt{j}}) \qquad (2.5)$$

The next is the probability that there are edges from v_l to v_i and v_j but not to v_k. Again, since there are multiple edges we account for the possibility that there are $t \geq n$ edges to v_i and $s \geq 1$ edges to v_j chosen in step l. We can write

$$Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k) = \sum_{t=1}^{d-1}\binom{d}{t}(\frac{1}{2\sqrt{j}\sqrt{l-1}})^t\sum_{s=1}^{d-t}\binom{d-t}{s}(\frac{1}{2\sqrt{i}\sqrt{l-1}})^s$$

$$
\begin{aligned}
& (1-\frac{1}{2\sqrt{k}\sqrt{l-1}}-\frac{1}{2\sqrt{j}\sqrt{l-1}}-\frac{1}{2\sqrt{i}\sqrt{l-1}})^{d-s-t} \\
= & \frac{d(d-1)}{4\sqrt{i}\sqrt{j}(l-1)}(1+O(\frac{1}{\sqrt{k}\sqrt{l-1}}-\frac{1}{\sqrt{j}\sqrt{l-1}} \\
& -\frac{1}{\sqrt{i}\sqrt{l-1}})) \\
= & \frac{d(d-1)}{4\sqrt{i}\sqrt{j}(l-1)}+O(\frac{1}{i\sqrt{j}l^{\frac{3}{2}}}).
\end{aligned}
$$

We can simplify this using $\frac{1}{l-1}=\frac{1}{l}+O(\frac{1}{l^2})$. Doing so gives us

$$Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k)=\frac{d(d-1)}{4l\sqrt{i}\sqrt{j}}+O(\frac{1}{l^2\sqrt{i}\sqrt{j}}+\frac{1}{il^{\frac{3}{2}}\sqrt{j}}) \tag{2.6}$$

Finally we consider the probability that there is no edge between v_i and v_j. We write $Pr(v_i \nsim v_j)=(1-\frac{1}{2\sqrt{i}\sqrt{j}})^d$. This is easily simplified to

$$Pr(v_i \nsim v_j)=1-O(\frac{1}{\sqrt{i}\sqrt{j}}) \tag{2.7}$$

We have the probabilities we need and are now ready to compute the expected number of triangles in the modified PA model.

Theorem 2.4.7 *Let $G_n=\overline{PA}(n,d)$ for $d \geq 2$ and let $X_n=ind(K_3,G_n)$. Then,*

$$E(X_n)=\frac{d^2(d-1)}{48}ln(n)^3+O(ln(n)^2).$$

Proof Consider vertices v_i, v_j, v_k with $1 \leq i < j < k \leq n$ and let X_{ijk} be the indicator variable for the event that v_i, v_j, v_k induce a K_3. Since adding edges in different time steps in the modified PA model are independent processes, we have

$$Pr(X_{ijk}=1)=Pr(v_i \sim v_j)Pr(v_k \sim v_j \cap v_k \sim v_i).$$

Using Equations 2.3 and 2.4,

$$\begin{aligned} Pr(X_{ijk}=1) &= \Big(\frac{d}{2}\frac{1}{\sqrt{i}\sqrt{j}} + O(\frac{1}{\sqrt{ij}})\Big)\Big(\frac{d(d-1)}{4}\frac{1}{\sqrt{i}\sqrt{j}k} + O(\frac{1}{\sqrt{i}\sqrt{j}k^2} + \frac{1}{i\sqrt{j}k^{\frac{3}{2}}})\Big) \\ &= \frac{d^2(d-1)}{8}\frac{1}{ijk} + O(\frac{1}{ij^{\frac{3}{2}}k}) \end{aligned}$$

Let $X_n = ind(K_3, G_n)$. By linearity of expectation we have $E(X_n) = \sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=j+1}^{n} E(X_{ij}$ $\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=j+1}^{n}\Big(\frac{d^2(d-1)}{8}\frac{1}{ijk} + O(\frac{1}{ij^{\frac{3}{2}}k})\Big)$.

To solve this triple sum we use the approximation technique described in Lemma 1.3.6 to replace the sums by integrals. This is the first time in this thesis in which we use Lemma 1.3.6 to simplify a multiple sum. For this proof we will provide all the details of computing $E(X_n)$. For future applications of Lemma 1.3.6, we will spare the reader the full details of the calculations.

For the error term, we can deduce $O(\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=j+1}^{n}\frac{1}{ij^{\frac{3}{2}}k}) = O(ln(n)^2)$. We have

$$E(X_n) = \frac{d^2(d-1)}{8}\sum_{i=1}^{n-2}\frac{1}{i}\sum_{j=i+1}^{n-1}\frac{1}{j}\sum_{k=j+1}^{n}\frac{1}{k} + O(ln(n)^2).$$

Using Lemma 1.3.6, we can write the first sum as $\sum_{k=j+1}^{n}\frac{1}{k} = ln(n+1) - ln(j+1) + O(\frac{1}{j})$. Using Taylor expansion $ln(n+1) = ln(n(1+\frac{1}{n})) = ln(n) + ln(1+\frac{1}{n}) = ln(n) + O(\frac{1}{n})$. Therefore, this sum can be written as $\sum_{k=j+1}^{n}\frac{1}{k} = ln(n) - ln(j) + O(\frac{1}{j})$. Plugging this back into $E(X_n)$ we obtain

$$\begin{aligned} E(X_n) &= \frac{d^2(d-1)}{8}\sum_{i=1}^{n-2}\frac{1}{i}\sum_{j=i+1}^{n-1}\frac{1}{j}(ln(n) - ln(j) + O(\frac{1}{j})) + O(ln(n)^2) \\ &= \frac{d^2(d-1)}{8}\sum_{i=1}^{n-2}\frac{1}{i}\sum_{j=i+1}^{n-1}(\frac{ln(n)}{j} - \frac{ln(j)}{j}) + O(ln(n)^2) \end{aligned}$$

Using Lemma 1.3.6, we can write the next sum as $\sum_{j=i+1}^{n-1} \frac{ln(n)}{j} - \frac{ln(j)}{j} = \frac{ln(n)^2}{2} + \frac{ln(1+i)^2}{2} - ln(n)ln(i) + O(\frac{ln(n)}{i+1})$. Note that $\int \frac{ln(x)}{x}dx = \frac{ln(x)^2}{2}$. Again using the Taylor expansion for $ln(1+i)$ we can write $E(X_n)$ as

$$\begin{aligned} E(X_n) &= \frac{d^2(d-1)}{8}\sum_{i=1}^{n-2}\frac{1}{i}(\frac{ln(n)^2}{2} + \frac{ln(i)^2}{2} - ln(n)ln(i) + O(\frac{ln(n)}{i+1})) \\ &+ O(ln(n)^2) \\ &= \frac{d^2(d-1)}{8}\sum_{i=1}^{n-2}(\frac{ln(n)^2}{2i} + \frac{ln(i)^2}{2i} - \frac{ln(n)ln(i)}{i}) + O(ln(n)^2) \end{aligned}$$

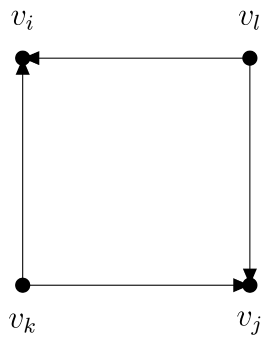
Using Lemma 1.3.6 one final time and the fact that $\int \frac{ln(i)^2}{i}di = \frac{ln(i)^3}{3}$ gives

$$\begin{aligned} E(X_n) &= \frac{d^2(d-1)}{8}(\frac{1}{2}ln(n)^2 ln(n-1) + \frac{1}{6}ln(n-1)^3 - \frac{1}{2}ln(n)ln(n-1)^2) \\ &+ O(ln(n)^2) \\ &= \frac{d^2(d-1)}{48}ln(n)^3 + O(ln(n)^2) \end{aligned}$$

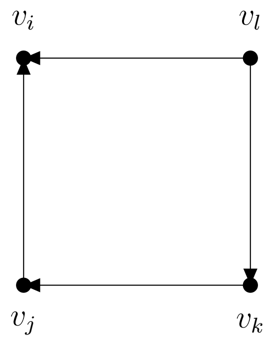
□

Comparing the number of triangles in the PA model and the modified PA model we see that both grow as $\Theta(ln(n)^3)$ as $n \to \infty$. The coefficient of the leading terms differ however by a factor of $\frac{d(d-1)(d+1)}{48} - \frac{d^2(d-1)}{48} = \frac{d(d-1)}{48}$. Calculating the expected number of 3-paths in the modified PA model yields a leading term which grows as $\Theta(nln(n))$ as $n \to \infty$ which matches the growth of the number of 3-paths in the PA model. For the sake of brevity we will not include this calculation. We now calculate the expected number of 4-cycles in the modified PA model.

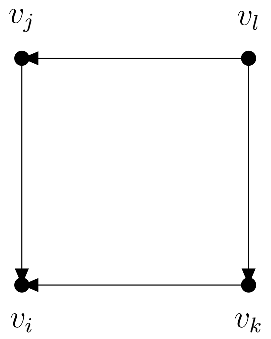
To count the number of 4-cycles in the modified PA model we need to consider the implicit directed nature of the PA model (older vertices select younger vertices as their endpoints). From this perspective, there are 3 unique ways in which a 4-cycle can appear in the modified PA model. Note that the ages of the vertices v_i, v_j, v_k, v_l satisfies $1 \leq i < j < k < l \leq n$.



C_4^1



C_4^2



C_4^3

To compute the number of 4-cycles we count each of these 3 cases separately.

Theorem 2.4.8 *Let $G_n = \overline{PA}(n,d)$ and let $X_n = ind(C_4, G_n)$. Then,*

$$E(X_n) = \frac{d^2(9d-1)(d-1)}{384} ln(n)^4 + O(ln(n)^3).$$

Proof Let $X_n = ind(C_4, G_n)$. Furthermore, let $X_n^i = ind(C_4^i, G_n)$ for $i = 1, 2, 3$. Then $X_n = X_n^1 + X_n^2 + X_n^3$.

Case 1: X_n^1 Let X_{ijkl}^1 be an indicator variable for the event that vertices v_i, v_j, v_k, v_l with $1 \leq i < j < k < l \leq n$ induce a copy of C_4^1. Then $Pr(X_{ijkl}^1 = 1) = Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k) Pr(v_k \sim v_j \cap v_k \sim v_i) Pr(v_j \nsim v_i)$. Using Equations (2.4, 2.6, 2.7),

$$Pr(X_{ijkl}^1 = 1) = \frac{d^2(d-1)^2}{16} \frac{1}{ijkl} + O(\frac{1}{ijk^2l}).$$

Using the linearity of expectation we can write

$$E(X_n^1) = \sum_{i=1}^{n-3}\sum_{j=1+1}^{n-2}\sum_{k=j+1}^{n-1}\sum_{l=k+1}^{n}\Big(\frac{d^2(d-1)^2}{16}\frac{1}{ijkl} + O(\frac{1}{ijk^2l})\Big).$$

Approximating this sum using Lemma 1.3.6 we obtain

$$E(X_n^1) = \frac{d^2(d-1)^2}{384}ln(n)^4 + O(ln(n)^3).$$

The details of the calculation above are identical in nature to the calculation for the number of triangles in the modified PA model.

Case 2: X_n^2 Let X_{ijkl}^2 be an indicator variable for the event that vertices v_i, v_j, v_k, v_l with $1 \leq i < j < k < l \leq n$ induce a copy of C_4^2. Then $Pr(X_{ijkl}^2 = 1) = Pr(v_l \sim v_i \cap v_l \sim v_k \cap v_l \nsim v_j)Pr(v_k \sim v_j \cap v_k \nsim v_i)Pr(v_j \sim v_i)$. Using Equations (2.4, 2.5, 2.3) we can write,

$$Pr(X_{ijkl}^2 = 1) = \frac{d^3(d-1)}{16}\frac{1}{ijkl} + O(\frac{1}{ij^{\frac{3}{2}}kl})$$

Using linearity of expectation we can write

$$E(X_n^2) = \sum_{i=1}^{n-3}\sum_{j=1+1}^{n-2}\sum_{k=j+1}^{n-1}\sum_{l=k+1}^{n}\Big(\frac{d^3(d-1)}{16}\frac{1}{ijkl} + O(\frac{1}{ij^{\frac{3}{2}}kl})\Big).$$

Using Lemma 1.3.6 to approximate this sum we obtain

$$E(X_n^2) = \frac{d^3(d-1)}{96}ln(n)^4 + O(ln(n)^3).$$

Case 3: X_n^3 Let X_{ijkl}^3 be an indicator variable for the event that vertices v_i, v_j, v_k, v_l with $1 \leq i < j < k < l \leq n$ induce a copy of C_4^3. Then $Pr(X_{ijkl}^3 = 1) = Pr(v_l \sim v_j \cap v_l \sim v_k \cap v_l \nsim v_i)Pr(v_k \sim v_i \cap v_k \nsim v_j)Pr(v_i \sim v_j)$. Using Equations (2.6, 2.5, 2.3) we can write

$$Pr(X^3_{ijkl}=1)=\frac{d^3(d-1)}{16}\frac{1}{ijkl}+O(\frac{1}{ij^{\frac{3}{2}}kl})$$

This is identical to $Pr(X^3_{ijkl}=1)$. Therefore we obtain $E(X^3_n)=\frac{d^3(d-1)}{96}ln(n)^4+O(ln(n)^3)$.

Finally by linearity of expectation $E(X_n)=E(X^1_n)+E(X^2_n)+E(X^3_n)$. So,

$$\begin{aligned} E(X_n) &= \frac{d^2(d-1)^2}{384}ln(n)^4+\frac{d^3(d-1)}{96}ln(n)^4+\frac{d^3(d-1)}{96}ln(n)^4+O(ln(n)^3) \\ &= \frac{d^2(9d-1)(d-1)}{384}ln(n)^4+O(ln(n)^3). \end{aligned}$$

□

Comparing our result to the number of injective 4-cycles given in Theorem 2.4.5, we see that both grow asymptotically as $ln(n)^4$.

2.5 Copy Model

Next we compute subgraph counts for our two copy models. Recall the two models we look at are the undirected copy model $Copy(n,p,d,G_{n_0})$ and the directed copy model $DCopy(n,p,q,G_{n_0})$. For the undirected copy model we consider the cases $d=0$ (pure copy model) and $p=0$ (uniform attachment model) as well as the general case. When considering subgraphs in the $DCopy(n,p,q,G_{n_0})$ we treat the graph generated as an undirected graph after its generation. That is to say, after generation we ignore the direction of the edges.

2.5.1 Solving Recurrence Relations

Most of the papers which introduce a copy model include a calculation for the expected number of edges which the model generates [17, 22, 84]. Each of these papers computes this expectation by writing a recurrence relation and solving it by approximating the relation by a differential equation. Such an approach requires assumptions on the recurrence relation that are not met (such as the number of edges changing continuously with n). These recurrence relations can in fact be solved directly. We outline some general solutions and techniques that we will use frequently in the remainder of this section.

The **Gamma function** is defined for every complex $z = a + ib$ with $a > 0$ by the integral $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$. For a positive integer n, we can write the Gamma function simply as $\Gamma(n) = (n-1)!$. For a non integer z, exact computation of $\Gamma(z)$ is very difficult. We can fortunately obtain an approximation through **Stirling's approximation** $\Gamma(z) = \sqrt{\frac{2\pi}{z}}(\frac{z}{e})^z(1 + O(\frac{1}{z}))$. We will often need to use Stirling's approximation to simplify the expression $\frac{\Gamma(n+a)}{\Gamma(n)}$ for $a > 0$ which constantly appears in solutions to our recursive relations.

Lemma 2.5.1 *If $a > 0$ then using Stirling's approximation we can simplify $\frac{\Gamma(n+a)}{\Gamma(n)} = (\frac{n}{e})^a(1 + O(\frac{1}{n}))$ and $\frac{\Gamma(n)}{\Gamma(n+a)} = (\frac{n}{e})^{-a}(1 + O(\frac{1}{n}))$.*

Proof Let $a > 0$ and use Stirling's approximation to simplify $\frac{\Gamma(n+a)}{\Gamma(n)}$.

$$\begin{aligned}
\frac{\Gamma(n+a)}{\Gamma(n)} &= \frac{\sqrt{\frac{2\pi}{n+a}}\frac{(n+a)^{n+a}}{e^{n+a}}(1 + O(\frac{1}{n+a}))}{\sqrt{\frac{2\pi}{n}}\frac{n^n}{e^n}(1 + O(\frac{1}{n}))} \\
&= \frac{1}{e^a}\sqrt{\frac{n}{n+a}}\frac{(n+a)^{n+a}}{n^n}(1 + O(\frac{1}{n})) \\
&= \frac{1}{e^a}(n+a)^a(\frac{n+a}{n})^{n+a-\frac{1}{2}}(1 + O(\frac{1}{n})) \\
&= \frac{1}{e^a}n^a(1 + \frac{a}{n})^{n+2a-\frac{1}{2}}(1 + O(\frac{1}{n}))
\end{aligned}$$

$$= (\frac{n}{e})^a(1+O(\frac{1}{n}))$$

In the second last step we use the Binomial series to write $(1+\frac{a}{n})^{n+2a-\frac{1}{2}} = (1+O(\frac{1}{n}))$ and $(1+O(\frac{1}{n}))(1+O(\frac{1}{n})) = (1+O(\frac{1}{n}))$.

For $\frac{\Gamma(n)}{\Gamma(n+a)}$ we write

$$\begin{aligned}\frac{\Gamma(n)}{\Gamma(n+a)} &= \frac{1}{\frac{\Gamma(n+a)}{\Gamma(n)}}\\ &= \frac{1}{(\frac{n}{e})^a(1+O(\frac{1}{n}))}\\ &= (\frac{n}{e})^{-a}(1+O(\frac{1}{n})).\end{aligned}$$

In the above we used the fact that $\frac{1}{1+O(\frac{1}{n})} = 1+O(\frac{1}{n})$. □

The recurrence relations that appear in this section take two general forms.

Lemma 2.5.2 *Let $a > 0$ and consider the recurrence relation $X_{n+1} = X_n(1+\frac{a}{n})$ starting at index n_0. Then*

$$X_n = X_{n_0}(n_0-1)!\frac{\Gamma(n+a)}{\Gamma(n)\Gamma(n_0+a)}. \tag{2.8}$$

Proof Let $a > 0$ and consider the recursion $X_{n+1} = X_n(1+\frac{a}{n})$. We first verify the solution for the base case X_{n_0} holds. We have

$$\begin{aligned}X_{n_0} &= X_{n_0}(n_0-1)!\frac{\Gamma(n_0+a)}{\Gamma(n_0)\Gamma(n_0+a)}\\ &= \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0)}\\ &= X_{n_0}.\end{aligned}$$

We now verify our solution by substituting $X_n = X_{n_0}(n_0-1)!\frac{\Gamma(n+a)}{\Gamma(n)\Gamma(n_0+a)}$ into the recursion relation to get

$$\begin{aligned}
X_{n+1} &= X_n(1+\frac{a}{n}) \\
&= X_{n_0}(n_0-1)!\frac{\Gamma(n+a)}{\Gamma(n)\Gamma(n_0+a)}(\frac{n+a}{n}) \\
&= X_{n_0}(n_0-1)!\frac{\Gamma(n+1+a)}{\Gamma(n+1)\Gamma(n_0+a)}.
\end{aligned}$$

Here we use the functional equation $\Gamma(z+1) = z\Gamma(z)$ for the Gamma function. □

If a is a positive integer then we can get an exact solution for the recursion. If a is not an integer then we will use Stirling's approximation to get an approximate solution. Using Lemma 2.5.1 we can write the solution in Lemma 2.5.2 as the following.

Lemma 2.5.3 *For a non integer a we can write the solution of the recursive relation in Lemma 2.5.2 as*

$$X_n = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)e^a}n^a + O(n^{a-1}),$$

Proof For the proof we manipulate $X_{n_0}(n_0-1)!\frac{\Gamma(n+a)}{\Gamma(n)\Gamma(n_0+a)}$ using Stirling's approximation. Using Lemma 2.5.1 we can write

$$\begin{aligned}
X_n &= \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)}\frac{\Gamma(n+a)}{\Gamma(n)} \\
&= \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)e^a}n^a(1+O(\frac{1}{n}))
\end{aligned}$$

Therefore we have $X_n = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)e^a}n^a + O(n^{a-1})$. □

The second recursive relation that occurs frequently in this section is a more general form of the relation in Lemma 2.5.2.

Lemma 2.5.4 *Let $a > 0$ and consider the recursive relation $X_{n+1} = X_n(1 + \frac{a}{n}) + b_n$ for some sequence b_n beginning at index n_0. Then*

$$X_n = \frac{\Gamma(n+a)}{\Gamma(n_0+a)\Gamma(n)}\Big[\Big(\sum_{i=n_0}^{n-1} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)!\Big].$$

Proof Let $a > 0$, b_n be some sequence and consider the recursive relation $X_{n+1} = X_n(1 + \frac{a}{n}) + b_n$. We verify the solution by plugging

$$X_n = \frac{\Gamma(n+a)}{\Gamma(n_0+a)\Gamma(n)}\Big[\Big(\sum_{i=n_0}^{n} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)!\Big]$$

into the recursive relation to get

$$\begin{aligned}
& X_n(1+\frac{a}{n}) + b_n \\
= \; & \frac{\Gamma(n+a)}{\Gamma(n_0+a)\Gamma(n)}\Big[\Big(\sum_{i=n_0}^{n-1} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)!\Big](\frac{n+a}{n}) + b_n \\
= \; & \frac{\Gamma(n+1+a)}{\Gamma(n_0+a)\Gamma(n+1)}\Big[\Big(\sum_{i=n_0}^{n-1} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)!\Big] + b_n \\
= \; & \frac{\Gamma(n+1+a)}{\Gamma(n_0+a)\Gamma(n+1)}\Big[\Big(\sum_{i=n_0}^{n-1} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)! \\
+ \; & b_n \frac{\Gamma(n_0+a)\Gamma(n+1)}{\Gamma(n+a+1)}\Big] \\
= \; & \frac{\Gamma(n+1+a)}{\Gamma(n_0+a)\Gamma(n+1)}\Big[\Big(\sum_{i=n_0}^{n} b_i \frac{\Gamma(n_0+a)\Gamma(i+1)}{\Gamma(i+1+a)}\Big) + X_{n_0}(n_0-1)!\Big] \\
= \; & X_{n+1}
\end{aligned}$$

□

We can apply Lemma 2.5.1 to simplify this expression, but it is not possible to fully simplify the expression unless the sequence b_n is known.

Lemma 2.5.5 *Let $a > 0$ and consider the recursive relation $X_{n+1} = X_n(1 + \frac{a}{n}) + b_n$ for some sequence b_n beginning at index n_0. Then*

$$X_n = [n^a \sum_{i=n_0}^{n-1} b_i (\frac{1}{i})^a + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)e^a} n^a + O(n^a \sum_{i=n_0}^{n-1} \frac{b_i}{i^{a+1}})](1 + O(\frac{1}{n})).$$

Proof For $a > 0$ we know that X_n satisfies

$$X_n = \frac{\Gamma(n+a)}{\Gamma(n_0+a)\Gamma(n)} \Big[\sum_{i=n_0}^{n-1} b_i \frac{\Gamma(n_0+a)\Gamma(i+1))}{\Gamma(i+1+a)} + X_{n_0}(n_0-1)! \Big]$$

from Lemma 2.5.4.

Using Lemma 2.5.1 we can simplify this to,

$$\begin{aligned} X_n &= \frac{1}{\Gamma(n_0+a)} (\frac{n}{e})^a [\sum_{i=n_0}^{n-1} \Gamma(n_0+a) b_i (\frac{i}{e})^{-a} (1 + O(\frac{1}{i})) + X_{n_0}(n_0-1)!](1 + O(\frac{1}{n})) \\ &= [n^a \sum_{i=n_0}^{n-1} b_i (\frac{1}{i})^a + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+a)e^a} n^a + O(n^a \sum_{i=n_0}^{n-1} \frac{b_i}{i^{a+1}})](1 + O(\frac{1}{n})) \end{aligned}$$

□

2.5.2 The Undirected Pure Copy Model

For the pure copy model we simply write $Copy(n, p, 0, G_{n_0}) = Copy(n, p, G_{n_0})$. We begin by giving the expected number of edges in $Copy(n, p, G_{n_0})$. This result was given in [17] and states that $E(e_n) \simeq n^{2p}$. We give a result that uses our recursive relation approach and not the DE approach used in [17].

Theorem 2.5.6 ([17]) *Consider the pure copy model $G_n = Copy(n, p, G_{n_0})$ and let e_n be the number of edges in G_n. Then*

$$E(e_n) = \begin{cases} \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)}(\frac{n}{e})^{2p} + O(n^{2p-1}) & \textit{if } \ 0 < p < 1, p \neq \frac{1}{2} \\ \frac{e_{n_0}}{n_0} n & \textit{if } \ p = \frac{1}{2} \\ \frac{e_{n_0}}{n_0(n_0-1)} n(n+1) & \textit{if } \ p = 1 \end{cases}$$

Proof Let $e_n = ind(K_2, G_n)$ and let $N(v_i, n)$ be the neighbourhood of a vertex v_i at time n. For $w \in N(v_i, n)$ let $X(w, v_i, n+1)$ be an indicator variable for the event that the edge from v_i to w is copied by v_{n+1} at time $n+1$. An edge w incident to v_i is copied to v_{n+1} with probability p when v_i is selected as the copy vertex at time $n+1$. Therefore $Pr(X(w, v_i, n+1) = 1) = \frac{p}{n}$. Using conditional expectation we can write

$$\begin{aligned} E(e_{n+1}|G_n) &= e_n + E(\sum_{i=1}^{n} \sum_{w \in N(v_i,n)} X(w, v_i, n+1)) \\ &= e_n + \sum_{i=1}^{n} \sum_{w \in N(v_i,n)} \frac{p}{n} \\ &= e_n + \sum_{i=1}^{n} \frac{p deg_n(v_i)}{n} \\ &= e_n + \frac{2p}{n} e_n \\ &= e_n(1 + \frac{2p}{n}) \end{aligned}$$

Applying expectation again gives the recursive relation $E(e_{n+1}) = E(e_n)(1 + \frac{2p}{n})$. From Lemma 2.5.2 we know the solution of this recursive relation is $e_n = \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)} \frac{\Gamma(n+2p)}{\Gamma(n)}$. When $p = \frac{1}{2}$ we can write this solution as

$$\begin{aligned} E(e_n) &= \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+1)} \frac{\Gamma(n+1)}{\Gamma(n)} \\ &= \frac{e_{n_0}(n_0-1)!}{n_0!} \frac{n!}{(n-1)!} \end{aligned}$$

$$= \frac{e_{n_0}}{n_0} n$$

When $p = 1$ we can write the solution as

$$\begin{aligned} E(e_n) &= \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2)} \frac{\Gamma(n+2)}{\Gamma(n)} \\ &= \frac{e_{n_0}(n_0-1)!}{(n_0+1)!} \frac{(n+1)!}{(n-1)!} \\ &= \frac{e_{n_0}}{n_0(n_0-1)} n(n+1) \end{aligned}$$

If p is not equal to $\frac{1}{2}$ or 1 then $2p$ is not an integer so we use Lemma 2.5.3 to get

$$E(e_n) = \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)} (\frac{n}{e})^{2p} + O(n^{2p-1}).$$

□

Note that to get a linear number of edges in the pure copy model we must set $p = \frac{1}{2}$.

It is easy to extend the approach of Theorem 2.5.6 to compute the expected number of any clique in *Copy*(n, p, G_{n_0}). In the proof of Theorem 2.5.6, an edge can only be copied if it is incident to the copy vertex chosen. Similarly, any clique incident to the copy vertex can be copied to form a new clique where the copy vertex is replaced by the new vertex v_{n+1} if each edge incident to the copy vertex in the clique is copied to v_{n+1}. This is in fact the only way cliques can form in *Copy*(n, p, G_{n_0}). In particular, if the clique number of G_{n_0} is t, then the clique number of *Copy*(n, p, G_{n_0}) will also be t.

Theorem 2.5.7 *Consider the pure copy model* $G_n =$ *Copy*(n, p, G_{n_0}). *Let* K_t *be a clique on* t *vertices and let* $X_n =$ *ind*(K_t, G_n). *Then*

$$E(X_n) = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+tp^{t-1})e^{tp^{t-1}}} n^{tp^{t-1}} + O(n^{tp^{t-1}-1}).$$

Proof A t-clique which is incident to v_i is copied to v_{n+1} if v_i is chosen as the copy vertex, which occurs with probability $\frac{1}{n}$, and each edge in the t-clique incident to v_i is copied to v_{n+1}, which occurs with probability p^{t-1}. We set $X(C, v_i, n+1)$ to be an indicator variable for the event that the t-clique C incident to v_i is copied to v_{n+1}. Then $Pr(X(C, v_i, n+1) = 1) = \frac{p^{t-1}}{n}$. Now let $K_{t,v_i,n}$ be the set containing each t-clique incident to v_i at time n. Using conditional expectation we can write

$$\begin{aligned} E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in K_{t,v_i,n}} X(C, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} \sum_{K_t \in K_{t,v_i,n}} \frac{p^{t-1}}{n} \\ &= X_n + \sum_{i=1}^{n} |K_{t,v_i,n}| \frac{p^{t-1}}{n} \\ &= X_n + \frac{tp^{t-1}}{n} X_n \\ &= X_n(1 + \frac{tp^{t-1}}{n}). \end{aligned}$$

In the above $\sum_{i=1}^{n} |K_{t,v_i,n}| = tX_n$ because each t-clique is incident to t vertices. Applying expectation again we get $E(X_{n+1}) = E(X_n)(1 + \frac{tp^{t-1}}{n})$. The solution to this recursive relation is given in Lemma 2.5.3 as

$$E(X_n) = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+tp^{t-1})e^{tp^{t-1}}} n^{tp^{t-1}} + O(n^{tp^{t-1}-1}).$$

□

Unfortunately counting subgraphs in the pure copy model is not always as straightforward as copying subgraphs incident to the copy vertex. To see how the process gets more complex let's try counting the number of 3-paths in the pure copy model. Let's begin with the case $p = 1$. There are two different ways in which a new 3-path can form at time $n + 1$. Let u be the copy vertex selected for v_{n+1}. In the first way, every 3-path which contains u gets copied with u being replaced by v_{n+1} in the copied 3-path. In the second way, a new 3-path is formed between u, v_{n+1} and each neighbour of u. Note that in this case, the neighbour of u is the vertex of degree two in the new 3-path.

Theorem 2.5.8 *Consider the pure copy model $G_n = Copy(n, 1, G_{n_0})$. Let $e_n = ind(K_2, G_n)$ and $X_n = ind(P_3, G_n)$. Then*

$$E(X_n) = \Big(\frac{2e_{n_0}}{n_0(n_0-1)(n_0+2)} + \frac{X_{n_0}}{n_0(n_0-1)(n_0-2)}\Big)n^3 + O(n^2).$$

Proof Let $P_{v_i,n}$ be the set of all 3-paths which contain v_i at time n. We set $X(C, v_i, n+1)$ to be an indicator variable for the event that a specific 3-path C incident to v_i is copied at time $n + 1$. Since $p = 1$ this event occurs if and only if v_i is selected as the copy vertex at time $n + 1$ so $Pr(X(C, v_i, n+1) = 1) = \frac{1}{n}$. Define $Y(v_i, n+1) = deg_n(v_i)$ if v_i is chosen as the copy vertex at time $n+1$ and 0 otherwise. In other words, $Y(v_i, n+1)$ is the number of 3-paths created at time $n + 1$ in the second way described above. Note that $E(Y(v_i, n+1)) = \frac{deg_n(v_i)}{n}$. So

$$\begin{aligned}
E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in P_{v_i,n}} X(C, v_i, n+1) + \sum_{i=1}^{n} Y(v_i, n+1)) \\
&= X_n + \sum_{i=1}^{n} \frac{|P_{v_i,n}|}{n} + \sum_{i=1}^{n} \frac{deg_n(v_i)}{n} \\
&= X_n + \frac{3X_n}{n} + \frac{2e_n}{n} \\
&= X_n(1 + \frac{3}{n}) + \frac{2e_n}{n}.
\end{aligned}$$

In the above we write $\sum_{i=1}^{n} |P(v_i, n)| = 3X_n$ since each 3-path is incident to 3 vertices. Applying expectation again and using linearity of expectation we can write,

$$\begin{aligned} E(X_{n+1}) &= E(X_n)(1 + \frac{3}{n}) + \frac{2E(e_n)}{n} \\ &= E(X_n)(1 + \frac{3}{n}) + \frac{2e_{n_0}}{n_0(n_0 - 1)}(n + 1). \end{aligned}$$

In the above, we use Theorem 2.5.6 with $p = 1$, to get $E(e_n) = \frac{e_{n_0}}{n_0(n_0-1)}n(n + 1)$. The recursive relation here takes the form of the general case considered in Lemma 2.5.4 with $b_n = \frac{2e_{n_0}}{n_0(n_0-1)}(n + 1)$. We can write the solution as,

$$\begin{aligned} E(X_n) &= \frac{\Gamma(n+3)}{\Gamma(n_0+3)\Gamma(n)}\Big[\Big(\sum_{i=n_0}^{n-1} \frac{\frac{2e_{n_0}}{n_0(n_0-1)}(i+1)\Gamma(n_0+3)\Gamma(i+1)}{\Gamma(i+4)}\Big) + X_{n_0}(n_0 - 1)!\Big] \\ &= \frac{(n+2)!}{\Gamma(n_0+3)(n-1)!}\Big[\Big(\sum_{i=n_0}^{n-1} \frac{\frac{2e_{n_0}\Gamma(n_0+3)}{n_0(n_0-1)}(i+1)!}{(i+3)!}\Big) + X_{n_0}(n_0 - 1)!\Big] \\ &= \frac{n(n+1)(n+2)}{\Gamma(n_0+3)}\Big[\Big(\frac{2e_{n_0}\Gamma(n_0+3)}{n_0(n_0-1)}\sum_{i=n_0}^{n-1} \frac{1}{(i+2)(i+3)}\Big) + X_{n_0}(n_0 - 1)!\Big] \\ &= \frac{2e_{n_0}n(n+1)(n+2)}{n_0(n_0-1)}\frac{n - n_0}{(n+2)(n_0+2)} + \frac{X_{n_0}}{n_0(n_0-1)(n_0-2)}n(n+1)(n+2) \\ &= \Big(\frac{2e_{n_0}}{n_0(n_0-1)(n_0+2)} + \frac{X_{n_0}}{n_0(n_0-1)(n_0-2)}\Big)n^3 + O(n^2). \end{aligned}$$

Above we used $\sum_{i=n_0}^{n-1} \frac{1}{(i+2)(i+3)} = \sum_{i=n_0}^{n-1} \left(\frac{1}{i+2} - \frac{1}{i+3}\right) = \frac{1}{n_0+2} - \frac{1}{n+2} = \frac{n-n_0}{(n+2)(n_0+2)}$.

□

When $p \neq 1$ there is an additional way in which a 3-path can form at time $n + 1$. Suppose u is the copy vertex chosen at time $n+1$. It is possible for a triangle which is incident to u to be copied as a 3-path if one but not both of the edges in the triangle incident to u are copied by v_{n+1}.

Theorem 2.5.9 *Consider the pure copy model $G_n = Copy(n, p, G_{n_0})$ with $0 < p < 1$. Let $e_n = ind(K_2, G_n)$, $T_n = ind(K_3, G_n)$ and $X_n = ind(P_3, G_n)$. Then*

$$E(X_n) = \begin{cases} \left(\frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} + \frac{e_{n_0}(n_0-1)!A_n}{\Gamma(n_0+2p)e^{2p}}\right)n^{p^2+2p} + O(n^{p^2+2p-1}) & \textit{if} \;\; 0 < p < \frac{2}{3} \\ \left(\frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} + \frac{T_{n_0}(n_0-1)!B_n}{\Gamma(n_0+3p^2)e^{3p^2}}\right)n^{p^2+2p} + O(n^{p^2+2p-1}) & \textit{if} \;\; \frac{2}{3} \leq p \leq 1 \end{cases}$$

where $A_n = \sum_{i=n_0}^{n-1} i^{-1-p^2}$ and $B_n = \sum_{i=n_0}^{n-1} i^{2p^2-2p-1}$.

Proof Let $P_{v_i,n,1}$ be the set of 3-paths in which v_i is the middle vertex at time n and let $P_{v_i,n,2}$ be the set of 3-paths in which v_i is an end vertex. Let $T_{v_i,n}$ be the set of triangles which are incident to v_i at time n and let $N(v_i, n)$ be the neighbourhood of v_i at time n. Let $X(C, v_i, n+1, 1)$ be the indicator variable for the event that the 3-path C in which v_i is the middle vertex is copied at time $n+1$. Let $X(C, v_i, n+1, 2)$ be the indicator variable for the event that the 3-path C in which v_i is not the middle vertex is copied at time $n+1$. Let $Y(C, v_i, n+1)$ be the indicator variable for the event that only one edge of the triangle C incident to v_i is copied by v_{n+1} and let $Z(w, v_i, n+1)$ be the indicator variable for the event that the edge $w \in N(v_i, n)$ is copied at time $n+1$. We have that $Pr(X(C, v_i, n+1, 1) = 1) = \frac{p^2}{n}$, $Pr(X(C, v_i, n+1, 2) = 1) = \frac{p}{n}$, $Pr(Y(C, v_i, n+1) = 1) = \frac{2p(1-p)}{n}$ and $Pr(Z(C, v_i, n+1) = 1) = \frac{p}{n}$. By conditional expectation,

$$\begin{aligned} E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in P_{v_i,n,1}} X(C, v_i, n+1, 1) + \sum_{i=1}^{n} \sum_{C \in P_{v_i,n,2}} X(C, v_i, n+1, 2) \\ &\quad + \sum_{i=1}^{n} \sum_{C \in T_{v_i,n}} Y(C, v_i, n+1) + \sum_{i=1}^{n} \sum_{w \in N(v_i,n)} Z(w, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} |P_{v_i,n,1}| \frac{p^2}{n} + \sum_{i=1}^{n} |P_{v_1,n,2}| \frac{p}{n} + \sum_{i=1}^{n} |T_{v_i,n}| \frac{2p(1-p)}{n} \end{aligned}$$

$$\begin{aligned}
&+ \sum_{i=1}^{n} \frac{pdeg_n(v_i)}{n} \\
&= X_n + \frac{p^2}{n} X_n + \frac{2p}{n} X_n + \frac{6p(1-p)}{n} T_n + \frac{2pe_n}{n} \\
&= X_n(1 + \frac{p^2+2p}{n}) + \frac{6p(1-p)}{n} T_n + \frac{2pe_n}{n}.
\end{aligned}$$

Note that $\sum_{i=1}^{n} |P_{v_i,n,1}| = X_n$ because each 3-path has only one middle vertex while $\sum_{i=1}^{n} |P_{v_i,n,2}| = 2X_n$ because each 3-path has two vertices of degree 1. Finally we simplify $\sum_{i=1}^{n} |T_{v_i,n}| = 3T_n$ because each triangle is counted three times in the sum.

Applying expectation again we get

$$E(X_{n+1}) \quad = \quad E(X_n)(1 + \frac{p^2+2p}{n}) + \frac{6p(1-p)}{n} E(T_n) + \frac{2pE(e_n)}{n}.$$

Substituting in $E(e_n) = \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)e^{2p}} n^{2p} + O(n^{2p-1})$ from Theorem 2.5.6 and $E(T_n) = \frac{T_{n_0}(n_0-1)!}{\Gamma(n+3p^2)e^{3p^2}} n^{3p^2} + O(n^{3p^2-1})$ from Theorem 2.5.7 we get the recursive relation

$$\begin{aligned}
E(X_{n+1}) \quad &= \quad E(X_n)(1 + \frac{p^2+2p}{n}) + \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)e^{2p}} n^{2p-1} + \frac{T_{n_0}(n_0-1)!}{\Gamma(n+3p^2)e^{3p^2}} n^{3p^2-1} \\
&+ \quad O(n^{2p-2} + n^{3p^2-2}).
\end{aligned}$$

If $p \leq \frac{2}{3}$ then $2p-1 > 3p^2-1$, so we can write

$$E(X_{n+1}) = E(X_n)(1 + \frac{p^2+2p}{n}) + \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)e^{2p}} n^{2p-1} + O(n^{3p^2-1}).$$

If $p \geq \frac{2}{3}$ then $3p^2-1 > 2p-1$ and

$$E(X_{n+1}) = E(X_n)(1 + \frac{p^2+2p}{n}) + \frac{T_{n_0}(n_0-1)!}{\Gamma(n+3p^2)e^{3p^2}} n^{3p^2-1} + O(n^{2p-1}).$$

Case 1: $p \leq \frac{2}{3}$ From Lemma 2.5.5 we know the can write the solution of this recursive relation as,

$$\begin{aligned}
E(X_n) &= [n^{p^2+2p} \sum_{i=n_0}^{n-1} \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+2p)e^{2p}} i^{-1-p^2} + O(n^{p^2+2p} \sum_{i=n_0}^{n-1} i^{-2-p^2}) \\
&+ \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} n^{p^2+2p}](1+O(\frac{1}{n})) \\
&= [\frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} + \frac{e_{n_0}(n_0-1)!A_n}{\Gamma(n_0+2p)e^{2p}}]n^{p^2+2p} + O(n^{p^2+2p-1})
\end{aligned}$$

where $A_n = \sum_{i=n_0}^{n-1} i^{-1-p^2}$.

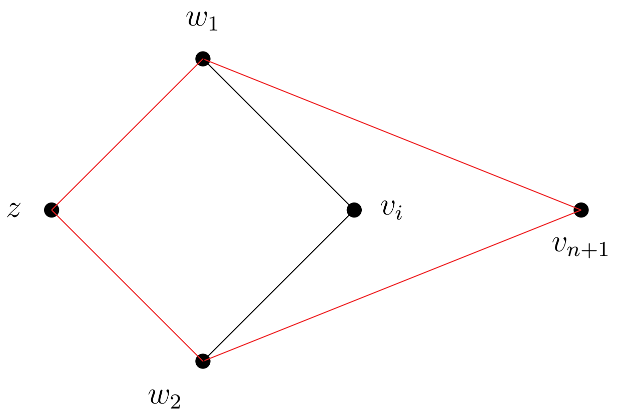
Case 2: $p \geq \frac{2}{3}$ From Lemma 2.5.5 we can write

$$\begin{aligned}
E(X_n) &= [n^{p^2+2p} \sum_{i=n_0}^{n-1} \frac{T_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)e^{3p^2}} i^{2p^2-2p-1} + O(n^{p^2+2p} \sum_{i=n_0}^{n-1} i^{2p^2-2p-2}) \\
&+ \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} n^{p^2+2p}](1+O(\frac{1}{n})) \\
&= [\frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2+2p)e^{p^2+2p}} + \frac{T_{n_0}(n_0-1)!B_n}{\Gamma(n_0+3p^2)e^{3p^2}}]n^{p^2+2p} + O(n^{p^2+2p-1})
\end{aligned}$$

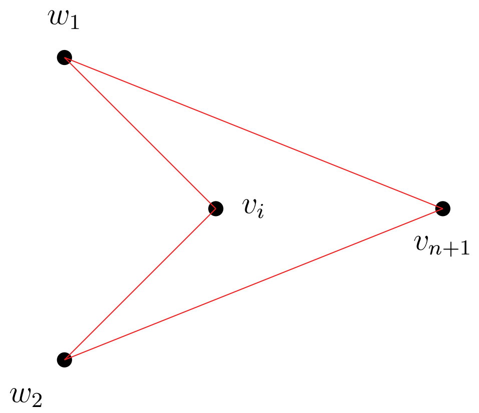
where $B_n = \sum_{i=n_0}^{n-1} i^{2p^2-2p-1}$.

□

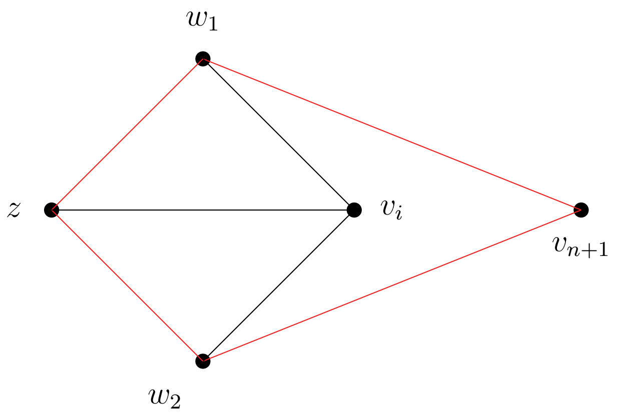
Next we count the number of 4-cycles in the pure copy model. As we should expect, counting subgraphs in the Copy model becomes more difficult as the size of the subgraphs increase as more cases need to be considered. There are 3 different ways in which a 4-cycle can appear at time $n+1$. Let v_i be the copy vertex selected at time $n+1$ and let $w_1, w_2 \in N(v_i, n)$.



The first case is the typical case where v_{n+1} copies a 4-cycle which is incident to v_i. The new 4-cycle here consists of vertices v_{n+1}, w_1, w_2 and z.



In the second case, a new 4-cycle is formed between v_{n+1}, v_i, w_1, w_2 as long as there is no edge between w_1 and w_2. In this case, v_i, w_1, w_2 form an induced 3-path where v_i is the middle vertex. Therefore, for every 3-path with v_i as the middle vertex is copied, a new 4-cycle forms.



In the final case, consider a g_7 which is incident to v_i at time n with v_i as a vertex of degree 3. Then a 4-cycle can form at time $n+1$ between v_{n+1}, w_1, w_2 and z. To count the number of 4-cycles created in this case we would need to develop an expression for the number of g_7's in the pure copy model. Instead we obtain a lower bound for the number of 4-cycles in the pure copy model by only considering the first two cases above.

Theorem 2.5.10 *Consider the pure copy model $G_n = Copy(n, p, G_{n_0})$ with $0 < p < 1$. Let $X_n = ind(C_4, G_n)$. Then,*

$$E(X_n) = \begin{cases} \Omega(n^{p^2+2p}) & \text{if } 0 < p < \frac{2}{3} \\ \Omega(n^{\frac{16}{9}} ln(n)) & \text{if } p = \frac{2}{3} \\ \Omega(n^{4p^2}) & \text{if } \frac{2}{3} < p < 1 \end{cases}$$

Proof Let $Y_n = ind(P_3, G_n)$. Let $X_{v_i,n}$ be the set of C_4's which are incident to v_i at time n. Let $P_{v_i,n}$ be the set of 3-paths in which v_i is the middle vertex at time n. For $C \in X_{v_i,n}$ let $X(C, v_i, n+1)$ be an indicator variable for the event that the 4-cycle C incident to v_i is copied by v_{n+1} at time $n+1$. Note that in this case, v_i is chosen as the copy vertex and the two edges in C which are incident to v_i are copied by v_{n+1}. We have $Pr(X(C, v_i, n+1) = 1) = \frac{p^2}{n}$. For $C \in P_{v_i,n}$, let $Y(C, v_i, n+1)$ be an indicator variable for the event that the 3-path C is copied at time $n+1$ with the role of v_i in C being replaced by v_{n+1} in the new 3-path. Note that in this case, v_i is chosen as the copy vertex and the two edges in C which are incident to v_i are copied by v_{n+1}. So$Pr(Y(C, v_i, n+1)) = \frac{p^2}{n}$. By conditional expectation,

$$\begin{aligned} E(X_n|G_n) &\geq X_n + E(\sum_{i=1}^{n} \sum_{C \in X_{v_i,n}} X(C, v_i, n+1)) + E(\sum_{i=1}^{n} \sum_{C \in P_{v_i,n}} Y(C, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} |X_{v_i,n}| \frac{p^2}{n} + \sum_{i=1}^{n} |Y_{v_i,n}| \frac{p^2}{n} \\ &= X_n + \frac{4p^2}{n} X_n + \frac{p^2}{n} Y_n. \end{aligned}$$

Note that each 4-cycle is incident to 4 vertices so we have $\sum_{i=1}^{n} |X_{v_i,n}| = 4X_n$. Applying expectation again gives

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n}) E(X_n) + \frac{p^2}{n} E(Y_n).$$

From Theorem 2.5.9 we have $E(Y_n) = \Theta(n^{p^2+2p})$ giving the recursive relation,

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n}) E(X_n) + \Theta(n^{p^2+2p-1}).$$

We can write the solution to this recursion using Lemma 2.5.5 as

$$E(X_n) \;=\; \Omega(n^{4p^2} \sum_{i=n_0}^{n-1} i^{-3p^2+2p-1} + n^{4p^2} + n^{4p^2} \sum_{i=n_0}^{n-1} i^{-3p^2+2p-2})$$

To simplify this expression we need to approximate $\sum_{i=n_0}^{n-1} i^{-3p^2+2p-1}$ using Lemma 1.3.6. Note that when $p = \frac{2}{3}$ we have $-3p^2 + 2p - 1 = -1$ so this case needs to be dealt with separately. If $p \neq \frac{2}{3}$ then we have

$$\begin{aligned} E(X_n) &= \Omega(n^{4p^2}\sum_{i=n_0}^{n-1} i^{-3p^2+2p-1} + n^{4p^2} + O(n^{4p^2}\sum_{i=n_0}^{n-1} i^{-3p^2+2p-2})) \\ &= \Omega(n^{4p^2}[n^{-3p^2+2p} - n_0^{-3p^2+2p}] + n^{4p^2}) \\ &= \Omega(n^{p^2+2p} + n^{4p^2}) \end{aligned}$$

If $p < \frac{2}{3}$, then n^{p^2+2p} is the dominant term and when $\frac{2}{3} < p < 1$, n^{4p^2} is the dominant term. When $p = \frac{2}{3}$ we have

$$\begin{aligned} E(X_n) &= \Omega(n^{\frac{16}{9}}\sum_{i=n_0}^{n-1} i^{-1} + n^{\frac{16}{9}} + O(n^{\frac{16}{9}}\sum_{i=n_0}^{n-1} i^{-2})) \\ &= \Omega(n^{\frac{16}{9}} ln(n)) \end{aligned}$$

□

2.5.3 Copy Model with $p = 0$ and $d > 0$ (Uniform Attachment Model)

In this section we consider the undirected copy model $Copy(n, 0, d, G_{n_0})$ in which the copying probability is set to zero. The model in this case is similar to the PA model only that at time n, the d end points of the edges from v_n are chosen u.a.r. as opposed to proportional to the degree of the destination vertex. The procedure for counting subgraphs in this model follows the same method we used in the modified PA model. For our calculations we will begin with G_{n_0} being the empty graph so at time 1, v_1 is added and creates d loops to itself. We will call this graph model the uniform attachment model denoted by $UA(n, d)$. The process generates multi-graphs. After generating the graph, we delete all loops and multi-edges so that a simple graph remains. We give the expected number of triangles, 3-paths and 4-cycles in the uniform attachment model.

Theorem 2.5.11 *Let $G_n = UA(n, d)$ with $d > 1$ and let $X_n = ind(K_3, G_n)$. Then*

$$E(X_n) = d^2(d-1)ln(n) + O(1).$$

Proof Let $X_n = ind(K_3, G_n)$ and for $1 \leq i < j < k \leq n$ let X_{ijk} be an indicator variable for the event that v_i, v_j, v_k induce a triangle. We can write $Pr(X_{ijk} = 1) = Pr(v_i \sim v_j \cap v_i \sim v_k \cap v_j \sim v_k) = Pr(v_i \sim v_j)Pr(v_i \sim v_k \cap v_j \sim v_k)$. We have,

$$Pr(v_i \sim v_j) \quad = \quad 1 - (1 - \frac{1}{j})^d$$

This gives

$$Pr(v_i \sim v_j) = \frac{d}{j} + O(\frac{1}{j^2}). \tag{2.9}$$

Also,

$$Pr(v_i \sim v_k \cap v_j \sim v_k) \quad = \quad \sum_{t=1}^{d-1} \binom{d}{t} (\frac{1}{k})^t \sum_{s=1}^{d-t} \binom{d-t}{s} (\frac{1}{k})^s (1 - \frac{2}{k})^{d-s-t}.$$

This can be simplified to give

$$Pr(v_i \sim v_k \cap v_j \sim v_k) = \frac{d(d-1)}{k^2} + O(\frac{1}{k^3}). \tag{2.10}$$

Therefore

$$Pr(X_{ijk} = 1) \quad = \quad (\frac{d}{j} + O(\frac{1}{j^2}))(\frac{d(d-1)}{k^2} + O(\frac{1}{k^3}))$$

$$= \frac{d^2(d-1)}{jk^2} + O(\frac{1}{jk^3}).$$

By linearity of expectation,

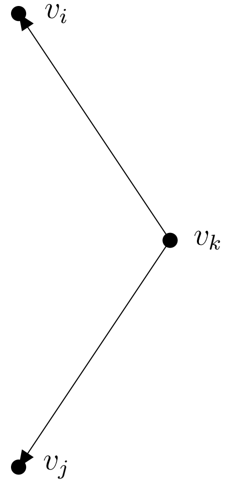
$$\begin{aligned} E(X_n) &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} E(X_{ijk} = 1) \\ &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \frac{d^2(d-1)}{jk^2} + O(\frac{1}{jk^3}). \end{aligned}$$

Using Lemma (1.3.6) we can approximate this sum to obtain,

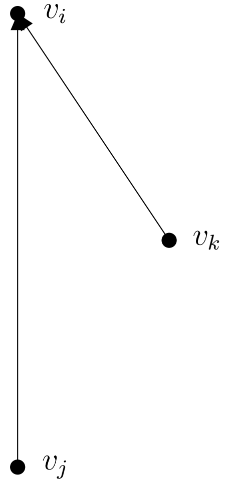
$$E(X_n) = d^2(d-1)ln(n) + O(1).$$

□

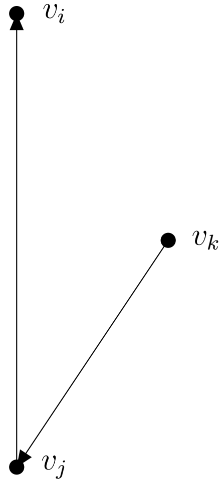
Now we give the calculation for the number of 3-paths in the UA model. For the result, we must consider the inherent directed nature of the UA model (younger vertices form edges to older vertices) to see that there are 3 unique ways in which a 3-path can form.



P_3^1



P_3^2



P_3^3

Theorem 2.5.12 *Let $G_n = UA(n,d)$ and $X_n = ind(P_3, G_n)$. Then,*

$$E(X_n) = \frac{d(5d-1)}{2}n + O(ln(n)^2)$$

Proof Let $X_n = ind(P_3, G_n)$ and let $X_n^i = ind(P_3^i, G_n)$ for $i = 1, 2, 3$. By linearity of expectation we can write $E(X_n) = E(X_n^1) + E(X_n^2) + E(X_n^3)$.

To compute these expectations we will need the following 4 probabilities: $Pr(v_i \sim v_j), Pr(v_i \nsim v_j), Pr(v_k \sim v_i \cap v_k \sim v_j)$ and $Pr(v_k \sim v_i \cap v_k \nsim v_j)$. We have already shown $Pr(v_i \sim v_j) = \frac{d}{j} + O(\frac{1}{j}^2)$ and $Pr(v_k \sim v_i \cap v_k \sim v_j) = \frac{d(d-1)}{k^2} + O(\frac{1}{k^3})$. Additionally,

$$Pr(v_i \nsim v_j) \;\; = \;\; (1 - \frac{1}{j})^d,$$

which gives

$$Pr(v_i \nsim v_j) = 1 + O(\frac{1}{j}). \qquad (2.11)$$

Also we have,

$$Pr(v_k \sim v_i \cap v_k \nsim v_j) \quad = \quad \sum_{t=1}^{d} \binom{d}{t} (\frac{1}{k})^t (1 - \frac{1}{k})^{d-t}$$

which simplifies to,

$$P(v_k \sim v_i \cap v_k \nsim v_j) = \frac{d}{k} + O(\frac{1}{k^2}). \tag{2.12}$$

Case 1: X_n^1 For $i \leq i < j < k \leq n$ let X_{ijk}^1 be the event that v_i, v_j, v_k induce a copy of P_3^1. We have

$$\begin{aligned} P(X_{ijk}^1 = 1) &= Pr(v_k \sim v_i \cap v_k \sim v_j) Pr(v_i \nsim v_j) \\ &= \Big(\frac{d(d-1)}{k^2} + O(\frac{1}{k^3})\Big)\Big(1 + O(\frac{1}{j})\Big) \\ &= \frac{d(d-1)}{k^2} + O(\frac{1}{k^3}). \end{aligned}$$

Using linearity of expectation and Lemma 1.3.6,

$$\begin{aligned} E(X_n^1) &= d(d-1) \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \frac{1}{k^2} + O(\frac{1}{k^3}) \\ &= \frac{d(d-1)}{2} n + O(ln(n)). \end{aligned}$$

Case 2: X_n^2 For $i \leq i < j < k \leq n$ let X_{ijk}^2 be the event that v_i, v_j, v_k induce a copy of P_3^2. We have

$$\begin{aligned} Pr(X_{ijk}^2 = 1) &= Pr(v_k \sim v_i \cap v_k \nsim v_j)Pr(v_i \sim v_j) \\ &= (\frac{d}{k} + O(\frac{1}{k^2}))(\frac{d}{j} + O(\frac{1}{j^2})) \\ &= \frac{d^2}{jk} + O(\frac{1}{j^2 k}). \end{aligned}$$

Using linearity of expectation and Lemma 1.3.6,

$$\begin{aligned} E(X_n^2) &= d^2 \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \frac{1}{jk} + O(\frac{1}{j^2 k}) \\ &= d^2 n + O(ln(n)^2). \end{aligned}$$

Case 3: X_n^3 For $i \leq i < j < k \leq n$ let X_{ijk}^3 be the event that v_i, v_j, v_k induce a copy of P_3^3. Then,

$$\begin{aligned} Pr(X_{ijk}^3 = 1) &= Pr(v_k \sim v_j \cap v_k \nsim v_i)Pr(v_i \sim v_j) \\ &= (\frac{d}{k} + O(\frac{1}{k^2}))(\frac{d}{j} + O(\frac{1}{j^2})) \\ &= \frac{d^2}{jk} + O(\frac{1}{j^2 k}) \end{aligned}$$

This is the same as $Pr(X_{ijk}^2 = 1)$. Therefore $E(X_n^3) = d^2 n + O(ln(n)^2)$.

Overall,

$$E(X_n) = E(X_n^1) + E(X_n^2) + E(X_n^3)$$

$$
\begin{aligned}
&= \frac{d(d-1)}{2}n + 2d^2n + O(ln(n)^2) \\
&= \frac{d(5d-1)}{2}n + O(ln(n)^2).
\end{aligned}
$$

□

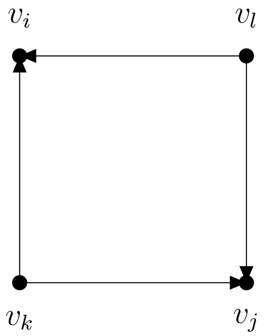
Next we count the number of 4-cycles in the uniform attachment model. Again, the procedure here is identical to the procedure used to count 4-cycles in the modified PA model. There is only one probability that we will need that we have not yet computed. This probability is

$$
\begin{aligned}
Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k) &= \sum_{t=1}^{d-1} \binom{d}{t} (\frac{1}{l})^t \sum_{s=1}^{d-t} \binom{d-t}{s} (\frac{1}{l})^s \\
&\quad (1 - \frac{3}{l})^{d-s-t} \\
&= \frac{d(d-1)}{l^2}(1 - O(\frac{1}{l}))
\end{aligned}
$$

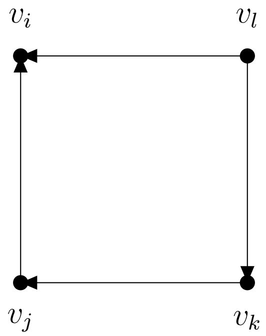
This gives,

$$
Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k) = \frac{d(d-1)}{l^2} + O((\frac{1}{l})^3)
$$

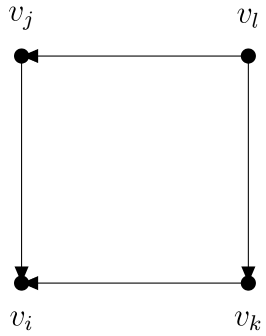
Recall that when we counted the number of 4-cycles in the modified PA model, there were 3 unique types of 4-cycles that could form. We remind the reader of the 3 types in the diagram below.



C_4^1



C_4^2



C_4^3

Theorem 2.5.13 *Let $G_n = UA(n,d)$ with $d > 1$ and let $X_n = ind(C_4, G_n)$. Then*

$$E(X_n) = \frac{d^2(5d-1)(d-1)}{2} ln(n) + O(1).$$

Proof Let $X_n = ind(C_4, G_n)$ and let $X_n^i = ind(C_4^i, G_n)$ for $i = 1,2,3$. By linearity of expectation $E(X_n) = E(X_n^1) + E(X_n^2) + E(X_n^3)$. To compute $E(X_n)$ we compute the expectation of each X_n^i separately.

Case 1: X_n^1 For $1 \leq i < j < k < l \leq n$ let X_{ijkl}^1 be the indicator variable for the event that v_i, v_j, v_k, v_l induce a copy of C_4^1. Using Equations 2.13, 2.10, and 2.11 we can write

$$\begin{aligned} Pr(X_{ijkl}^1 = 1) &= Pr(v_l \sim v_i \cap v_l \sim v_j \cap v_l \nsim v_k) Pr(v_k \sim v_j \cap v_k \sim v_i) Pr(v_i \nsim v_j) \\ &= (\frac{d(d-1)}{l^2} + O(\frac{1}{l^3}))(\frac{d(d-1)}{k^2} + O(\frac{1}{k^3}))(1 + O(\frac{1}{j})) \\ &= \frac{d^2(d-1)^2}{l^2k^2} + O(\frac{1}{l^2k^2j}). \end{aligned}$$

Using linearity of expectation we can write

$$E(X_n^1) = \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} \Big(\frac{d^2(d-1)^2}{l^2k^2} + O(\frac{1}{l^2k^2j}) \Big).$$

Using Lemma 1.3.6 we can write this as,

$$E(X_n^1) = \frac{d^2(d-1)^2}{2} ln(n) + O(1).$$

Case 2: X_n^2 Let X_{ijkl}^2 be the indicator variable for the event that v_i, v_j, v_k, v_l induce a copy of C_4^2 for $1 \leq i < j < k < l \leq n$. Using Equations 2.13, 2.12, and 2.11 we can write,

$$\begin{aligned} Pr(X_{ijkl}^1 = 1) &= Pr(v_l \sim v_i \cap v_l \sim v_k \cap v_l \nsim v_j) Pr(v_k \sim v_j \cap v_k \nsim v_i) Pr(v_i \sim v_j) \\ &= (\frac{d(d-1)}{l^2} + O(\frac{1}{l^3}))(\frac{d}{k} + O(\frac{1}{k^2}))(\frac{d}{j} + O(\frac{1}{j^2})) \\ &= \frac{d^3(d-1)}{jkl^2} + O(\frac{1}{j^2kl^2}). \end{aligned}$$

Using linearity of expectation we can write

$$E(X_n^2) = \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} \Big(\frac{d^3(d-1)}{jkl^2} + O(\frac{1}{j^2kl^2}) \Big).$$

Using Lemma 1.3.6 to approximate the sums by integrals we can write

$$E(X_n^2) = d^3(d-1) ln(n) + O(1).$$

Case 3: X_n^3 Let X_{ijkl}^3 be the indicator variable for the event that v_i, v_j, v_k, v_l induce a copy of C_4^3 for $1 \leq i < j < k < l \leq n$. Using Equations 2.13, 2.12, and 2.11 we can write,

$$Pr(X_{ijkl}^3 = 1) \;=\; Pr(v_l \sim v_j \cap v_l \sim v_k \cap v_l \nsim v_i) Pr(v_k \sim v_i \cap v_k \nsim v_j) Pr(v_i \sim v_j)$$

$$
\begin{aligned}
&= (\frac{d(d-1)}{l^2} + O(\frac{1}{l^3}))(\frac{d}{k} + O(\frac{1}{k^2}))(\frac{d}{j} + O(\frac{1}{j^2})) \\
&= \frac{d^3(d-1)}{jkl^2} + O(\frac{1}{j^2kl^2}).
\end{aligned}
$$

Using linearity of expectation we can write

$$E(X_n^3) = \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} \Big(\frac{d^3(d-1)}{jkl^2} + O(\frac{1}{j^2kl^2}) \Big).$$

This is the same expression we obtained for $E(X_n^2)$ so we have

$$E(X_n^3) = d^3(d-1)ln(n) + O(1).$$

Combining all three cases we obtain

$$E(X_n) = \frac{d^2(5d-1)(d-1)}{2} ln(n) + O(1).$$

□

2.5.4 Copy Model with $p > 0$ and $d > 0$

The undirected copy model $Copy(n, d, p, G_{n_0})$ combines the mechanisms of the pure copy model and the uniform attachment model. The motivation in adding the uniform attachment mechanism is that it results in a power law degree distribution because the copy mechanism alone is not sufficient to produce a power law [22]. In this section we count the expected number of triangles, 3-paths and provide a lower bound for the number of 4-cycles. We begin with the number of edges that was given in [22]. Their result was computed by approximating the recursive relation by a differential equation. We resolve the equation using our method of directly solving the recursive relation.

Theorem 2.5.14 ([22]) *Let $G_n = Copy(n, p, d, G_{n_0})$ and $e_n = ind(K_2, G_n)$. Then*

$$E(e_n) = \begin{cases} \frac{d}{1-2p}n + O(n^{2p}) & if \;\; p < \frac{1}{2} \\ dnln(n) + O(n) & if \;\; p = \frac{1}{2} \\ \Theta(n^{2p}) & if \;\; p > \frac{1}{2} \end{cases}$$

Proof Let $e_n = ind(K_2, G_n)$. For a vertex v_i and $w \in N(v_i, n)$ let $X(w, v_i, n+1)$ be an indicator variable for the event that v_{n+1} picks v_i as its copy vertex and copies w. We have $Pr(X(w, v_i, n+1) = 1) = \frac{p}{n}$. At time $n+1$, $v+n+1$ adds d random edges to vertices in G_n.

Setting up the recursive relation gives,

$$\begin{aligned} E(e_{n+1}|G_n) &= e_n + E(\sum_{i=1}^{n} \sum_{w \in N(v_i,n)} X(w, v_i, n+1) + d) \\ &= e_n + \sum_{i=1}^{n} |N(v_i, n)| \frac{p}{n} + d \\ &= e_n + e_n \frac{2p}{n} + d. \end{aligned}$$

Note that $\sum_{i=1}^{n} |N(v_i, n)| = 2e_n$. Applying expectation again gives the recursive relation

$$E(e_{n+1}) = (1 + \frac{2p}{n})E(e_n) + d.$$

We can obtain a solution for this recursion using Lemma 2.5.5,

$$E(e_n) = [n^{2p} \sum_{i=n_0}^{n-1} di^{-2p} + O(n^{2p} \sum_{i=n_0}^{n-1} i^{-2p-1}) + n^{2p} \frac{e_{n_0}(n_0 - 1)!}{\Gamma(n_0 + 2p)e^{2p}}](1 + O(\frac{1}{n})).$$

To simplify this further we approximate $\sum_{i=n_0}^{n-1} i^{-2p}$. Using Lemma 1.3.6 to approximate this sum we consider 3 possibilities for p: $0 \leq p \leq \frac{1}{2}$, $p = \frac{1}{2}$, and $\frac{1}{2} < p < 1$.

Case 1: $0 < p < \frac{1}{2}$

$$\begin{aligned} E(e_n) &= [dn^{2p} \int_{n_0}^{n} i^{-2p} di + O(n^{2p})](1 + O(\frac{1}{n})) \\ &= \frac{d}{1-2p} n + O(n^{2p}) \end{aligned}$$

Case 2: $p = \frac{1}{2}$

$$\begin{aligned} E(e_n) &= [dn \int_{n_0}^{n} i^{-1} di + O(n)](1 + O(\frac{1}{n})) \\ &= dnln(n) + O(n) \end{aligned}$$

Case 3: $\frac{1}{2} < p < 1$

$$\begin{aligned} E(e_n) &= [dn^{2p} \int_{n_0}^{n} i^{-2p} di + O(n^{2p})](1 + O(\frac{1}{n})) \\ &= \Theta(n^{2p}) \end{aligned}$$

□

The Copy model has a linear number of edges when $0 < p < \frac{1}{2}$. As this is the case we are interested in, we only count the expected number of triangles, 3-paths, and 4-cycles for $0 < p < \frac{1}{2}$.

We begin with the number of triangles. There will be three ways in which a new triangle can form at time $n+1$. Consider the two edges in the new triangle which are incident to v_{n+1}. We can either have both of these edges formed due to the copying mechanism, one edge from copying and the other from one of the d extra edges or both from the d extra edges. We might suspect that since d is fixed and that $n \to \infty$, that the majority of the new triangles are formed via the copying mechanism. We show that this is almost true. In our result we use the power law degree distribution of $\mathit{Copy}(n, p, d, G_{n_0})$ which we stated in Theorem 1.5.3. Recall that the result stated that $\mathit{Copy}(n, p, d, G_{n_0})$ has a power law degree distribution coefficient γ which is the largest solution of $1 = p\gamma - p + p^{\gamma-1}$. Note that for a power law to form we must have $d > 0$ but otherwise γ is independent of the value of d. For a linear number of edges in $\mathit{Copy}(n, p, d, G_{n_0})$ we require that $0 < p < \frac{1}{2}$. In our result, we need to determine the range of power law coefficients in $\mathit{Copy}(n, p, d, G_{n_0})$ for this range of p. We in fact already have this answer from Theorem 2.1.1. Recall that in this theorem, we proved that in a graph with a power law degree distribution with a linear number of edges, the power law coefficient must be greater than 2. Therefore, in $\mathit{Copy}(n, p, d, G_{n_0})$ with $0 < p < \frac{1}{2}$, we have that $\gamma > 2$. It is easy to verify by plugging $p = \frac{1}{2}$ into $1 = p\gamma - p + p^{\gamma-1}$ that this value of p corresponds to $\gamma = 2$. We argue that as p decreases from $\frac{1}{2}$ to 0 that γ increases from 2 to ∞. As $p \to 0$ it follows that $p^{\gamma-1} \to 0$ (since we know $\gamma > 2$). Therefore for $1 = p\gamma - p + p^{\gamma-1}$ to hold we require that $p\gamma \to 1$ as $p \to 0$ which implies that $\gamma \to \infty$. Therefore, when $0 < p < \frac{1}{2}$, we have $\gamma \in (2, \infty)$.

Theorem 2.5.15 *Let $G_n = \mathit{Copy}(n, d, p, G_{n_0})$ where $p < \frac{1}{2}, d > 0$ and let $X_n = \mathit{ind}(K_3, G_n)$. Then*

$$E(X_n) = \begin{cases} o(n) & \text{if } 0.4739 < p < 0.5 \\ \Theta(n^{3p^2}) & \text{if } p \leq 0.4739 \end{cases}$$

Proof Let $X_n = \mathit{ind}(K_3, G_n)$. There are three ways in which a new triangle can form at time $n+1$. The first is that both of the edges in the new triangle incident to v_{n+1} are produced by copying. For this case let $T_{v_i,n}$ be the set of triangles incident to vertex v_i at time n and let $X(C, v_i, n+1)$ be an indicator variable for the event that a triangle $C \in T_{v_i,n}$ is copied to v_{n+1}. Then $P(X(C, v_i, n+1) = 1) = \frac{p^2}{n}$. For the second case, of the two edges incident to v_{n+1} in a new triangle, one is formed by copying and the other is an extra edge. For $w_1 \in N(v_i, n)$ and $w_2 \in N(w_1, n)$ let $Y(v_i, w_1, w_2, n{+}1)$ be the event that v_{n+1} selects v_i as its copy vertex, copies an edge to w_1 and forms a random edge to w_2. We have that $Pr(Y(v_i, w_1, w_2, n+1) = 1) = \frac{dp}{n^2}$. In the final way, we have that both of the edges in the new triangle incident to v_{n+1} are formed through extra edge addition. Let $Z(w_i, w_j, n{+}1)$ be the event that w_i and w_j are both selected as random edges from v_{n+1}. We have that $Pr(Z(w_i, w_j, n+1) = 1) = \frac{d^2}{n^2}$. Using conditional expectation we can write,

$$\begin{aligned}
E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in T_{v_i,n}} X(C, v_i, n+1)) \\
&+ E(\sum_{i=1}^{n} \sum_{w_1 \in N(v_i,n)} \sum_{w_2 \in N(w_1,n)} Y(v_i, w_1, w_2, n)) \\
&+ E(\sum_{i=1}^{n} \sum_{w_j \in N(w_i,n)} Z(w_i, w_j, n+1)) \\
&= X_n + \sum_{i=1}^{n} |T_{v_i,n}| \frac{p^2}{n} + \sum_{i=1}^{n} \sum_{w_1 \in N(v_i,n)} \sum_{w_2 \in N(w_1,n)} \frac{pd}{n^2}
\end{aligned}$$

Applying expectation again we can write

$$\begin{aligned}
E(X_{n+1}) &= E(X_n) + E(\sum_{i=1}^{n} |T_{v_i,n}| \frac{p^2}{n}) \\
&+ E(\sum_{i=1}^{n} \sum_{w_1 \in N(v_i,n)} \sum_{w_2 \in N(w_1,n)} \frac{pd}{n^2}) + E(\sum_{i=1}^{n} \sum_{w_j \in N(w_1,n)} \frac{d^2}{n^2}) \\
&= E(X_n) + \frac{3p^2}{n} E(X_n) + E(\sum_{i=1}^{n} \sum_{w_1 \in N(v_i,n)} deg_n(w_2) \frac{pd}{n^2}) + E(\sum_{i=1}^{n} deg_n(w_i) \frac{d^2}{n^2}) \\
&= E(X_n) + \frac{3p^2}{n} E(X_n) + \frac{pd}{n^2} E(\sum_{i=1}^{n} deg_n(v_i)^2) + \frac{2d^2}{n^2} E(e_n).
\end{aligned}$$

Note that $\sum_{i=1}^{n} \sum_{w_1 \in N(v_i,n)} \sum_{w_2 \in N(w_1,n)} \frac{pd}{n^2} = \frac{pd}{n^2} \sum_{i=1}^{n} deg_n(v_i)^2$.

From Theorem 2.5.14 we have $E(e_n) = \frac{d}{1-2p}n + O(n^{2p})$. The only piece here that remains to be worked out is $E(\sum_{i=1}^{n} deg_n(v_i)^2)$. Let $N_{k,n}$ be the number of vertices of degree k in G_n. Then $E(\sum_{i=1}^{n} deg_n(v_i)^2) = E(\sum_{k=1}^{n-1} k^2 N_{k,n}) = \sum_{k=1}^{n-1} k^2 E(N_{k,n})$. Using the power law degree distribution for G_n from Theorem 1.5.3 we can write

$$\begin{aligned} \sum_{k=1}^{n-1} k^2 E(N_{k,n}) &= \sum_{k=1}^{n-1} k^2 (1 + O(\frac{1}{k}) c k^{-\gamma} n \\ &= cn \sum_{k=1}^{n-1} (k^{2-\gamma} + O(k^{1-\gamma})). \end{aligned}$$

To simplify this sum with integrals using Lemma 1.3.6 we have to consider 3 possible ranges for γ: $2 < \gamma < 3$, $\gamma = 3$ and $\gamma > 3$.

Case 1: $2 < \gamma < 3$ Using Lemma 1.3.6 we have,

$$\begin{aligned} cn \sum_{k=1}^{n-1} k^{2-\gamma} + O(k^{1-\gamma}) &= cn[\frac{n^{3-\gamma}}{3-\gamma} + O(1)] \\ &= \frac{c}{3-\gamma} n^{4-\gamma} + O(n) \end{aligned}$$

Using this we can write the recursion relation as $E(X_{n+1}) = (1 + \frac{3p^2}{n})E(X_n) + \frac{pdc}{3-\gamma} n^{2-\gamma} + O(\frac{1}{n})$. Solving this using Lemma 2.5.5 gives

$$\begin{aligned} E(X_n) &= [n^{3p^2} \sum_{i=n_0}^{n-1} (\frac{pdc}{3-\gamma} i^{2-\gamma-3p^2} + O(i^{-1-3p^2})) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)} n^{3p^2} \\ &+ O(n^{3p^2} \sum_{i=n_0}^{n-1} i^{1-\gamma-3p^2})](1 + O(\frac{1}{n})) \end{aligned}$$

To approximate $\sum_{i=n_0}^{n-1} i^{2-\gamma-3p^2}$ we need to know the range of values for $2-\gamma-3p^2$. Specifically we need to identify the values of p and γ for the following three ranges: $2-\gamma-3p^2 < -1$, $2-\gamma-3p^2 = -1$ and $2-\gamma-3p^2 > -1$. Through numerical calculation we can show that when $p \approx .4739$ we have that $\gamma \approx 2.3262$ and $2-\gamma-3p^2 = -1$. If follows that for $p \in (.4739, .5)$ we have $\gamma \in (2, 2.3262)$ and $2-\gamma-3p^2 \in (-.75, -1)$. Note that when $p = .5$ we have $2-\gamma-3p^2 = -.75$.

Let's begin with the case that $2-\gamma-3p^2 < -1$. In this case we can write the solution as

$$\begin{aligned}
E(X_n) &= [n^{3p^2}\sum_{i=n_0}^{n-1}(\frac{pdc}{3-\gamma}i^{2-\gamma-3p^2} + O(i^{-1-3p^2})) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)}n^{3p^2} \\
&+ O(n^{3p^2}\sum_{i=n_0}^{n-1} i^{1-\gamma-3p^2})](1+O(\frac{1}{n})) \\
&= [\frac{pdcn^{3p^2}}{(3-\gamma)(3-\gamma-3p^2)}n^{3-\gamma-3p^2} + O(n^{3p^2})](1+O(\frac{1}{n})) \\
&= [\frac{pdc}{(3-\gamma)(3-\gamma)}n^{3-\gamma} + O(n^{3p^2})] \\
&= o(n).
\end{aligned}$$

Note that in this case $3-\gamma < 1$ and $3p^2 < 1$ so overall, the number of triangles is sub-linear.

We next consider the case where $2-\gamma-3p^2 = -1$. In this case we can write the solution as

$$E(X_n) = [n^{3p^2}\sum_{i=n_0}^{n-1}(\frac{pdc}{3-\gamma}i^{-1} + O(i^{-1-3p^2})) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)}n^{3p^2}$$

$$\begin{aligned} &+ \quad O(n^{3p^2}\sum_{i=n_0}^{n-1} i^{1-\gamma-3p^2})](1+O(\frac{1}{n})) \\ &= \quad [\frac{pdc}{3-\gamma}n^{3p^2}ln(n)+O(n^{3p^2})](1+O(\frac{1}{n}) \\ &= \quad o(n). \end{aligned}$$

Note that the leading term here $n^{3p^2}ln(n)$ is sub-linear here since $ln(n) = o(n^x)$ for all $x > 0$.

Finally we consider the final case where $2 - \gamma - 3p^2 > -1$. In this case we can write the solution as

$$\begin{aligned} E(X_n) \quad &= \quad [n^{3p^2}\sum_{i=n_0}^{n-1}(\frac{pdc}{3-\gamma}i^{2-\gamma-3p^2}+O(i^{-1-3p^2}))+\frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)}n^{3p^2} \\ &+ \quad O(n^{3p^2}\sum_{i=n_0}^{n-1} i^{1-\gamma-3p^2})](1+O(\frac{1}{n})) \\ &= \quad \Theta(n^{3p^2}). \end{aligned}$$

<u>**Case 2:** $\gamma = 3$</u> Using Lemma 1.3.6 we have,

$$\begin{aligned} cn\sum_k k^{-1}+O(k^{-2}) \quad &= \quad cn[ln(n)+O(1)] \\ &= \quad cnln(n)+O(n). \end{aligned}$$

We can write the recursion relation as $E(X_{n+1}) = (1+\frac{3p^2}{n})E(X_n) + pdc\frac{ln(n)}{n} + O(\frac{1}{n})$. Solving this using Lemma 2.5.5 gives

$$\begin{aligned}
E(X_n) &= [n^{3p^2} \sum_{i=n_0}^{n-1} pdc(ln(i)i^{-1-3p^2} + O(i^{-1-3p^2})) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)} n^{3p^2} \\
&+ O(n^{3p^2} \sum_{i=n_0}^{n-1} ln(i)i^{-2-\gamma-3p^2})](1+O(\frac{1}{n})) \\
&= \Theta(n^{3p^2}).
\end{aligned}$$

<u>**Case 3:** $\gamma > 3$</u> Using Lemma 1.3.6 we have,

$$\begin{aligned}
cn\sum_k k^{2-\gamma} + O(k^{1-\gamma}) &= cn[\frac{n^{3-\gamma}}{3-\gamma} + O(1)] \\
&= \frac{c}{3-\gamma} n^{4-\gamma} + O(n) \\
&= O(n).
\end{aligned}$$

Note that in this case $n > n^{4-\gamma}$ since $\gamma > 3$.

$$\begin{aligned}
E(X_{n+1}) &= (1+\frac{3p^2}{n})E(X_n) + O(\frac{1}{n}) \\
&= (1+\frac{3p^2}{n})E(X_n) + O(\frac{1}{n}).
\end{aligned}$$

We can solve this recursive relation using Lemma 2.5.5 to obtain

$$\begin{aligned}
E(X_n) &= [n^{3p^2} \sum_{i=n_0}^{n-1} O(i^{-1-3p^2}) + O(n^{3p^2} \sum_{i=n_0}^{n-1} i^{-2-3p^2}) \\
&+ n^{3p^2} \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+3p^2)e^{3p^2}}](1+O(\frac{1}{n}))
\end{aligned}$$

$$= \Theta(n^{3p^2}).$$

□

When $p < .4739$, the number of triangles in the Copy model grows at a rate of $\Theta(n^{3p^2})$ which is the same that triangles grow in the pure copy model (Theorem 2.5.7). This is not surprise, since the graph is sparse, it is unlikely that the d extra edges added in each step would contribute many triangles. When $p > .4739$, we see from the proof of Theorem 2.5.15 that the number of triangles grows sub-linear but at a rate faster than $\Theta(n^{3p^2})$. A possible explanation for this is that as $p \to \frac{1}{2}$ the graph is becoming denser. The result indicates as p approaches $\frac{1}{2}$ the number of triangles is approaching $\Theta(n)$.

If the extra edges are not contributing very many extra triangles then they must be contributing many extra induced P_3's. Recall the relation $ind(P_3, G) = inj(P_3, G) - 3ind(K_3, G)$. Since we suspect that there are many more P_3's than triangles in $Copy(n, p, d, G_{n_0})$ will count the number of injective P_3's. The easiest method of computing this is use the expression $inj(P_3, G_n) = \sum_{i=1}^{n} \binom{deg_n(v_i)}{2}$.

Theorem 2.5.16 *Let $G_n = Copy(n, p, d, G_{n_0})$ for $0 < p < \frac{1}{2}$, $d > 0$ and let $X_n = inj(P_3, G_n)$. Then*

$$E(X_n) = \begin{cases} \frac{c}{2}\frac{n^{4-\gamma}}{3-\gamma} + O(n) & \text{if } \sqrt{2}-1 < p < \frac{1}{2} \\ \frac{c}{2} n ln(n) + O(n) & \text{if } p = \sqrt{2}-1 \\ \Theta(n) & \text{if } 0 < p < \sqrt{2}-1 \end{cases}$$

Proof This follows immediately from Theorem 2.1.2 and the relation $inj(P_3, G_n) = ind(P_3, G_n) + 3ind(K_3, G_n)$. The only detail missing is find the value of p which corresponds to the power law coefficient $\gamma = 3$. This is easy to obtain by placing $\gamma = 3$ into $1 = p\gamma - p + p^{\gamma-1}$. Solving this you obtain $p = \sqrt{2} - 1$. The result follows since when $0 < p < \sqrt{2} - 1$ $\gamma > 3$, when $p = \sqrt{2} - 1$ $\gamma = 3$ and when $\sqrt{2} - 1 < p < 1$ $2 < \gamma < 1$.

□

Corollary 2.5.17 *Let $G_n = Copy(n, p, d, G_{n_0})$ with $d > 0$, $0 < p < \frac{1}{2}$ and let $X_n = ind(P_3, G_n)$. Then,*

$$E(X_n) = \begin{cases} \frac{c}{2}\frac{n^{4-\gamma}}{3-\gamma} + O(n) & if \ \ \sqrt{2} - 1 < p < \frac{1}{2} \\ \frac{c}{2}nln(n) + O(n) & if \ \ p = \sqrt{2} - 1 \\ \Theta(n) & if \ \ 0 < p < \sqrt{2} - 1 \end{cases}$$

Proof The result follows immediately from Theorems 2.5.15 and 2.5.16 and the relation $inj(P_3, G) = ind(P_3, G) + 3ind(K_3, G_n)$.

□

We conclude with a lower bound for the number of 4-cycles in the copy model. For our result, we will only consider those 4-cycles that are created through copying, while ignoring those which are created through random edge addition. It is suspected that 4-cycles created in that manner are minimal as compared to those created by copying. Our proof will proceed in the same manner as the proof for the number of 4-cycles in the pure copy model in Theorem 2.5.10.

Theorem 2.5.18 *Let $G_n = Copy(n, p, d, G_{n_0})$ with $0 < p < \frac{1}{2}$ and let $X_n = ind(C_4, G_n)$. Then we have*

$$E(X_n) = \begin{cases} \Omega(n^{4-\gamma}) & if \ \ \sqrt{2} - 1 < p < \frac{1}{2} \\ \Omega(nln(n)) & if \ \ p = \sqrt{2} - 1 \\ \Omega(n) & if \ \ 0 < p < \sqrt{2} - 1 \end{cases}$$

Proof Let $X_n = ind(C_4, G_n)$ and $Y_n = ind(P_3, G_n)$. Let $X_{v_i,n}$ be the set of C_4's which are incident to v_i at time n. Let $P_{v_i,n}$ be the set of 3-paths in which v_i is the middle vertex at time n. For $C \in X_{v_i,n}$ let $X(C, v_i, n+1)$ be an indicator variable for the event that the 4-cycle C incident to v_i is copied by v_{n+1} at time $n+1$. It is easy to see that $Pr(X(C, v_i, n+1) = 1) = \frac{p^2}{n}$. For $C \in P_{v_i,n}$ let $Y(C, v_i, n+1)$ be an indicator variable for the event that the 3-path C is copied at time $n+1$ with the role of v_i in C being replaced by v_{n+1} in the new 3-path. It is easy to see $Pr(Y(C, v_i, n+1) = 1) = \frac{p^2}{n}$. By conditional expectation

$$\begin{aligned} E(X_n|G_n) &\geq X_n + E(\sum_{i=1}^{n} \sum_{C \in X_{v_i,n}} X(C, v_i, n+1)) + E(\sum_{i=1}^{n} \sum_{C \in P_{v_i,n}} Y(C, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} |X_{v_i,n}| \frac{p^2}{n} + \sum_{i=1}^{n} |P_{v_i,n}| \frac{p^2}{n} \\ &= X_n + \frac{4p^2}{n} X_n + \frac{p^2}{n} Y_n. \end{aligned}$$

Note that each 4-cycle is counted 4 times in $\sum_{i=1}^{n} |X_{v_i,n}|$ so we have $\sum_{i=1}^{n} |X_{v_i,n}| = 4X_n$. Also, each 3-path with v_i as the middle vertex is counted once in $\sum_{i=1}^{n} |P_{v_i,n}|$ so $\sum_{i=1}^{n} |P_{v_i,n}| = Y_n$. Applying expectation again gives

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n}) E(X_n) + \frac{p^2}{n} E(Y_n).$$

From Theorem 2.5.17 we have 3 different ranges of p which gives different expressions for $E(Y_n)$. We consider each of these separately.

Case 1: $0 < p < \sqrt{2} - 1$ From Theorem 2.5.17 we have that $E(Y_n) = \Theta(n)$. For our recursive relation this gives,

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n}) E(X_n) + \Theta(1).$$

Using Lemma 2.5.5 we can solve this to obtain,

$$\begin{aligned} E(X_n) &\geq [n^{4p^2} \sum_{i=n_0}^{n-1} \Theta(i^{-4p^2}) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+4p^2)e^{4p^2}} n^{4p^2} \\ &+ O(n^{4p^2} \sum_{i=n_0}^{n-1} i^{-1-4p^2}](1+O(\frac{1}{n})) \\ &= \Omega(n). \end{aligned}$$

Case 2: $p = \sqrt{2}-1$ From Theorem 2.5.17 we have that $E(Y_n) = \Theta(nln(n))$. Plugging this into our recursive relation gives us

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n})E(X_n) + \Theta(ln(n)).$$

Using Lemma 2.5.5 we can solve this to obtain,

$$\begin{aligned} E(X_n) &\geq [n^{4p^2} \sum_{i=n_0}^{n-1} \Theta(ln(i)i^{-4p^2}) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+4p^2)e^{4p^2}} n^{4p^2} \\ &+ O(n^{4p^2} \sum_{i=n_0}^{n-1} i^{-1-4p^2} ln(i)](1+O(\frac{1}{n})) \\ &= \Omega(nln(n)) \end{aligned}$$

Case 3: $\sqrt{2}-1 < p < \frac{1}{2}$ From Theorem 2.5.17 we have that $E(Y_n) = \Theta(n^{4-\gamma})$ where γ is the power law coefficient which is in the range of $2 < \gamma < 3$ for this range of p. Plugging this into our recursive relation gives us,

$$E(X_{n+1}) \geq (1 + \frac{4p^2}{n})E(X_n) + \Theta(n^{3-\gamma}).$$

Using Lemma 2.5.5 we can solve this to obtain,

$$\begin{aligned} E(X_n) &\geq [n^{4p^2}\sum_{i=n_0}^{n-1}\Theta(i^{3-\gamma-4p^2}) + \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+4p^2)e^{4p^2}}n^{4p^2} \\ &+ O(n^{4p^2}\sum_{i=n_0}^{n-1} i^{2-4p^2}](1+O(\frac{1}{n})) \\ &= \Omega(n^{4-\gamma}). \end{aligned}$$

□

2.5.5 The Directed Pure Copy Model

The pure directed copy model is studied in [84] where G_{n_0} is a single isolated vertex. We use their proof to extend their result for the number of edges in the pure directed copy model. We also give the case where $q = 0$ which is not included in their result. The number of edges in the $q = 0$ case is obtained using the same approach as the undirected pure copy model.

Theorem 2.5.19 ([84]) *Consider the directed pure copy model $G_n = DCopy(n, p, q, G_{n_0})$ and let $e_n = ind(K_2, G_n)$. Then,*

$$E(e_n) = \begin{cases} \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+p)e^p}n^p + O(n^{p-1}) & \textit{if } \; q = 0, 0 < p < 1 \\ \frac{q}{1-p}n + O(n^p) & \textit{if } \; q > 0, 0 < p < 1 \\ qnln(n) + O(n) & \textit{if } \; q > 0, p = 1 \end{cases}$$

Proof Let $e_n = ind(K_2, G_n)$ and let $N^+(v_i, n)$ be the out neighbourhood of a vertex v_i at time n. For $w \in N^+(v_i, n)$, let $X(w, v_i, n+1)$ be an indicator variable for the event that the out-edge from v_i to w is copied at time $n + 1$. It is easy to see that $Pr(X(w, v_i, n+1) = 1) = \frac{p}{n}$. Let $Y(n + 1)$ be the indicator variable for the event there is an edge between v_{n+1} and the copy vertex selected at time $n + 1$. It is easy to see that $Pr(Y(n + 1) = 1) = q$. Using conditional expectation we can write

$$\begin{aligned}
E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{w \in N^+(v_i,n)} X(w, v_i, n+1)) + E(Y(n+1)) \\
&= X_n + \sum_{i=1}^{n} \sum_{w \in N^+(v_i,n)} \frac{p}{n} + q \\
&= X_n + \sum_{i=1}^{n} \frac{pdeg_n^+(v_i)}{n} + q \\
&= X_n + \frac{pX_n}{n} + q.
\end{aligned}$$

Note that for a directed graph G, $\sum_{v \in V(G)} deg^+(v_i) = X_n$. Applying expectation again gives the recursive relationship $E(X_{n+1}) = E(X_n)(1 + \frac{p}{n}) + q$. This recursive relation is of the general form in Lemma 2.5.4 with $b_n = q$. When $q = 0$ we get the recursive relation $E(X_{n+1}) = E(X_n)(1 + \frac{p}{n})$. Using Lemma 2.5.2, the solution in this case is $E(X_n) = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p)e^p} n^p + O(n^{p-1})$.

When $q \neq 0$, we get the solution from Lemma 2.5.5,

$$E(X_n) = [n^p \sum_{i=n_0}^{n-1} qi^{-p} + \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+p)e^a} n^p + O(n^p \sum_{i=n_0}^{n-1} i^{-p-1})](1 + O(\frac{1}{n})).$$

In simplifying $\sum_{i=n_0}^{n-1} i^{-p}$ using Lemma 1.3.6 we deal with the cases $0 < p < 1$ and $p = 1$ separately. If $0 < p < 1$ then we have

$$\begin{aligned}
E(X_n) &= [n^p \sum_{i=n_0}^{n-1} qi^{-p} + \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+p)e^a} n^p + O(n^p \sum_{i=n_0}^{n-1} i^{-p-1})](1 + O(\frac{1}{n})) \\
&= \frac{q}{1-p} n + O(n^p).
\end{aligned}$$

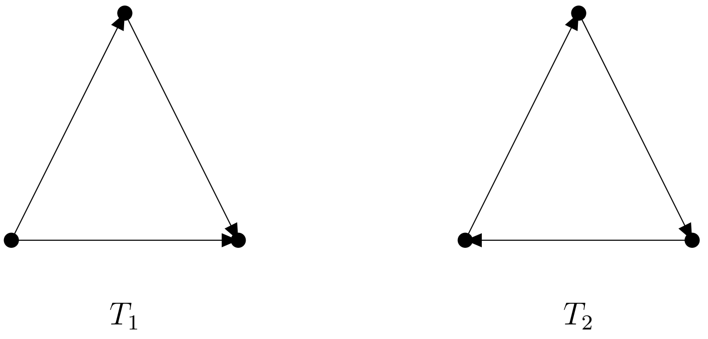
If $p = 1$ then we have,

$$\begin{aligned} E(X_n) &= [n\sum_{i=n_0}^{n-1} qi^{-1} + \frac{e_{n_0}(n_0-1)!}{\Gamma(n_0+1)e^a}n + O(n\sum_{i=n_0}^{n-1} i^{-2})](1+O(\frac{1}{n})) \\ &= qnln(n) + O(n). \end{aligned}$$

□

The general approach in determining the expected number of triangles and 3-paths for the directed pure copy model is similar to the approach used for the undirected pure copy model. However, the directed nature can affect the types of subgraphs that can be copied. Let's first consider counting triangles.

In a directed graph there are 2 different types of triangles which can appear, we will see that only one type can form at time $n+1$ in the directed pure copy model.



In T_1, there is a vertex with out degree 2 and in T_2 the edges form a directed 3-cycle. New directed triangles which appear at time $n+1$ are only of the T_1 type. A new T_1 will form at time $n+1$ if the copy vertex v_i is incident to a T_1 as the vertex with out-degree 2 and each of these edges are copied by v_{n+1}. There is no way for a new T_2 to form at time $n+1$ in the directed pure copy model. Any T_2's in $DCopy(n, p, q, G_{n_0})$ will only be those that were initially present in G_{n_0}.

At time $n+1$, triangles can form in two different ways. The first way happens when v_{n+1} copies a directed triangle incident to the copy vertex as we described above.

Additional directed triangles can form in the directed copy model if an edge forms from v_{n+1} to the copy vertex u. In this case, v_{n+1}, u and each out-neighbour of v_{n+1} induce a directed triangle. In these directed triangles, v_{n+1} has out-degree two so that a new T_1 is formed in this case.

Theorem 2.5.20 *Consider the directed copy model $G_n = DCopy(n, p, q, G_{n_0})$ and let $X_n = ind(T_1, G_n)$. Then*

$$E(X_n) = \begin{cases} \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2)e^{p^2}} n^{p^2} + O(n^{p^2-1}) & \textit{if} \;\; q = 0, 0 < p < 1 \\ \frac{q^2 p}{(1-p)(1-p^2)} n + O(n^p) & \textit{if} \;\; q > 0, 0 < p < 1 \end{cases}$$

Proof Let $X_n = ind(T_1, G_n)$ and let $e_n = ind(K_2, G_n)$. Let $X_{v_i,n}$ denote the set of T_1's incident to v_i at time n in which v_i has out-degree two in the directed triangle. Let $N^+(v_i, n)$ denote the out-neighbourhood of v_i at time n. For all directed triangles C in which v_i has out-degree 2 let $X(C, v_i, n+1)$ be the indicator that the triangle C incident to v_i is copied at time $n+1$. This event occurs if v_i is chosen as the copy vertex at time $n+1$ and each of the out-neighbours in C of v_i are copied. Therefore $Pr(X(C, v_i, n+1) = 1) = \frac{p^2}{n}$. Let $Y(w, v_i, n+1)$ be the indicator variable for the event that v_i is selected as the copy vertex at time $n+1$, an edge forms between v_i and v_{n+1} and v_{n+1} forms an edge to $w \in N^+(v_i, n)$. All three of these events are independent so $Pr(Y(w, v_i, n+1) = 1) = \frac{qp}{n}$. Using conditional expectation we can write

$$\begin{aligned} E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in X_{v_i,n}} X(C, v_i, n+1)) + E(\sum_{i=1}^{n} \sum_{w \in N^+(v_i,n)} Y(w, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} \frac{p^2}{n} |X_{v_i,n}| + \sum_{i=1}^{n} \frac{qp}{n} |N^+(v_i, n)| \\ &= X_n + \frac{p^2}{n} X_n + \frac{qp}{n} e_n. \end{aligned}$$

Note that $\sum_{i=1}^{n} |X_{v_i,n}| = X_n$ as v_i is the vertex of out-degree 2 in exactly one T_1. Applying expectation once more gives the recursive relation

$$E(X_{n+1}) = (1 + \frac{p^2}{n})E(X_n) + \frac{qp}{n}E(e_n).$$

This is the general form we have in Lemma 2.5.4. When $q = 0$ the recursive relation simplifies to $E(X_{n+1}) = (1 + \frac{p^2}{n})E(X_n)$ whose solution from Lemma 2.5.3 is

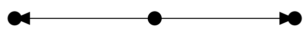
$$E(X_n) = \frac{X_{n_0}(n_0 - 1)!}{\Gamma(n_0 + p^2)e^{p^2}} n^{p^2} + O(n^{p^2-1}).$$

If $q \neq 0$ and $0 < p < 1$ then using Lemma 2.5.5 we can write the solution of the recursive relation as

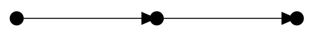
$$\begin{aligned} E(X_n) &= [n^{p^2} \sum_{i=n_0}^{n-1} (\frac{q^2 p}{1-p} + O(i^{p-1}))i^{-p^2} + \frac{X_{n_0}(n_0-1)!}{\Gamma(n+p^2)e^{p^2}} n^{p^2} \\ &+ O(n^{p^2} \sum_{i=n_0}^{n-1} i^{-p^2+p-2})](1 + O(\frac{1}{n})) \\ &= [n^{p^2} \sum_{i=n_0}^{n-1} \frac{q^2 p}{1-p} i^{-p^2} + \frac{X_{n_0}(n_0-1)!}{\Gamma(n+p^2)e^{p^2}} n^{p^2} + O(n^{p^2} \sum_{i=1}^{n-1} i^{-p^2+p-1})](1 + O(\frac{1}{n})) \\ &= \frac{q^2 p}{(1-p)(1-p^2)} n + O(n^p). \end{aligned}$$

□

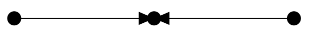
Counting 3-paths in the directed copy model is more complex than counting triangles. There are three different directed 3 paths which can form at time $n + 1$ in $DCopy(n, p, q, G_{n_0})$.



P_3^1



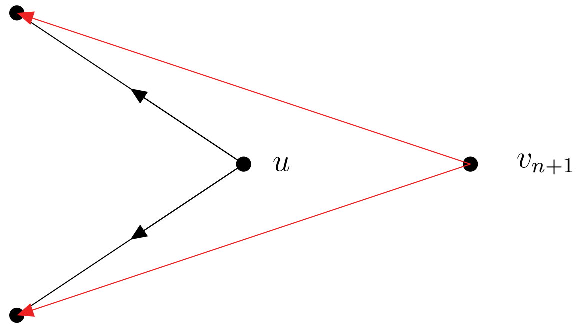
P_3^2



P_3^3

We count these 3 different types of 3-paths separately and combine them afterwards to get the 3-path count for $DCopy(n, p, q, G_{n_0})$.

Let's begin with P_3^1. There is only one way in which a P_3^1 can form at time $n+1$.



In the diagram above, the red edges are added at time $n+1$ and the black edges are present at time n. A new P_3^1 forms at time $n+1$ if v_{n+1} copies a P_3^1 which is incident to the copy vertex u where u is the vertex of out-degree 2 in the P_3^1.

Theorem 2.5.21 *Consider the pure directed copy model $G_n = DCopy(n, p, q, G_{n_0})$ and let $X_n = ind(P_3^1, G_n)$. Then*

$$E(X_n) = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2)e^{p^2}} n^{p^2} + O(n^{p^2-1}).$$

Proof Let $X_n = ind(P_3^1, G_n)$. Let $P_{v_i,n}^1$ be the number of P_3^1's incident to v_i at time n in which v_i is the vertex with out-degree two. For each v_i and each $C \in P_{v_i,n}^1$, let $X(C, v_i, n+1)$ be the event that a $C \in P_{v_i,n}^1$ is copied at time $n+1$. This event occurs if v_i is chosen as the copy vertex at time $n+1$ and each out-neighbour of v_i in C is copied by v_{n+1}. Since all these events are independent $Pr(X(C, v_i, n+1) = 1) = \frac{p^2}{n}$. Using conditional expectation we can write

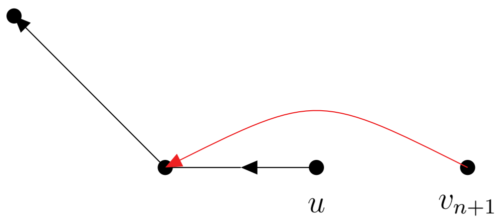
$$\begin{aligned} E(X_{n+1}|G_n) &= = X_n + E(\sum_{i=1}^{n} \sum_{C \in P^1_{v_i,n}} X(C, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} \sum_{C \in P^1_{v_i,n}} \frac{p^2}{n} \\ &= X_n + \sum_{i=1}^{n} |P^1_{v_i,n}| \frac{p^2}{n} \\ &= X_n + X_n \frac{p^2}{n}. \end{aligned}$$

Note that $\sum_{i=1}^{n} |P^1_{v_i,n}| = X_n$. Applying expectation again we get the recursive relation $E(X_{n+1}) = (1 + \frac{p^2}{n})E(X_n)$. Using Lemma 2.5.2 we get the solution

$$E(X_n) = \frac{X_{n_0}(n_0-1)!}{\Gamma(n_0+p^2)e^{p^2}} n^{p^2} + O(n^{p^2-1}).$$

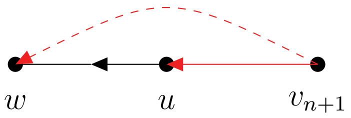
□

There are 3 different ways that a new P_3^2 can form at time $n+1$ which we depict in the diagrams below. Red edges are added at time $n+1$, dashed red edges are edges that could be added at time $n+1$ but need to not be present for the induced P_3^2 appear and black edges are those edges present at time n.



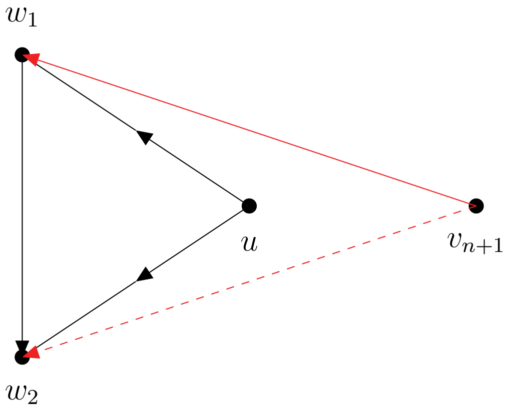
Case 1

In Case 1, v_{n+1} copies a P_3^2 incident to the copy vertex u by replacing u in the new P_3^2 formed at $n+1$. It does not matter whether or not an edge forms from v_{n+1} to u.



Case 2

In Case 2, a new P_3^2 is formed between v_{n+1}, the copy vertex u and each out-neighbour of u if v_{n+1} forms an edge to u but not to w.



Case 3

In Case 3, a new P_3^2 is formed between v_{n+1} and the copy vertex u's out-neighbours w_1 and w_2. In this case we require that an edge does not form between v_{n+1} and w_2, otherwise v_{n+1}, w_1 and w_2 would induce a triangle. Also note that in the diagram above, if the edge formed from v_{n+1} to w_2 and not from v_{n+1} to w_1 then v_{n+1}, w_1 and w_2 would induce a P_3^3 and not a P_3^2.

Theorem 2.5.22 *Consider the pure directed copy model $G_n = DCopy(n, p, q, G_{n_0})$ with $q > 0$ and $0 < p < 1$. Let $X_n = ind(P_3^2, G_n)$. Then*

$$E(X_n) = \frac{q^2}{(1-p^2)(1-p)}n + O(n^p).$$

Proof Let $X_n = ind(P_3^2, G_n)$, $e_n = ind(K_2, G_n)$ and $T_n = ind(T_1, G_n)$. For Case 1, let $P_{v_i,n}^2$ be the set of P_3^2's incident to u at time n in which u is the vertex with out-degree 1 and in-degree 0 in the P_3^2. For a vertex v_i and each $C \in P_{v_i,n}^2$ let $X(C, v_i, n+1)$ be an indicator variable for the event that C is copied at time $n+1$. This event occurs if v_i is selected as the copy vertex at time $n+1$ and v_{n+1} copies the out-edge of u in C. Therefore $Pr(X(C, v_i, n+1) = 1) = \frac{p}{n}$. For Case 2, let $N^+(v_i, n)$ be the set of out-neighbours of v_i at time n. For a vertex v_i and $w \in N^+(v_i, n)$ let $Y(w, v_i, n+1)$ be an indicator variable for the event that v_i is selected as the copy vertex at time $n+1$ and an edge forms from v_{n+1} to u but no edge forms from v_{n+1} to w. Therefore $Pr(Y(w, v_i, n+1) = 1) = \frac{q(1-p)}{n}$. For Case 3, let $T_{v_i,n}$ be the set of T_1's incident to v_i at time n in which v_i is the vertex with out-degree 2. For $C \in T_{v_i,n}$ consisting of vertices v_i,w_1, w_2 we assume wlog that there is a directed edge from w_1 to w_2. For a vertex v_i and $C \in T_{v_i,n}$ let $Z(C, v_i, n+1)$ be an indicator variable for the event that v_i gets selected as the copy vertex at time $n+1$ and v_{n+1} form an edge to w_1 but not to w_2. We have that $Pr(Z(C, v_i, n+1) = 1) = \frac{p(1-p)}{n}$. Using conditional expectation we can write

$$\begin{aligned} E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in P_{v_i,n}^2} X(C, v_i, n+1)) + E(\sum_{i=1}^{n} \sum_{w \in N^+(v_i,n)} Y(w, v_i, n+1)) \\ &+ E(\sum_{i=1}^{n} \sum_{C \in T_{v_i,n}} Z(C, v_i, n+1)) \\ &= X_n + \sum_{i=1}^{n} |P_{v_i,n}^2| \frac{p}{n} \sum_{i=1}^{n} deg^+(v_i, n)\frac{q(1-p)}{n} + \sum_{i=1}^{n} |T_{v_i,n}| \frac{p(1-p)}{n} \end{aligned}$$

$$= \quad X_n + \frac{p}{n}X_n + \frac{q(1-p)}{n}e_n + \frac{p(1-p)}{n}T_n$$

Note that $\sum_{i=1}^{n} |P^2_{v_i,v}| = X_n$ and $\sum_{i=1}^{n} |T_{v_i,n}| = T_n$.

Applying expectation again we obtain,

$$E(X_{n+1}) = (1+\frac{p}{n})E(X_n) + \frac{q(1-p)}{n}E(e_n) + \frac{p(1-p)}{n}E(T_n).$$

If $q > 0$ and $0 < p < 1$ then $E(e_n) = \frac{q}{1-p}n + O(n^p)$ and $E(T_n) = \frac{q^2p}{(1-p)(1-p^2)}n + O(n^p)$. Therefore we can write,

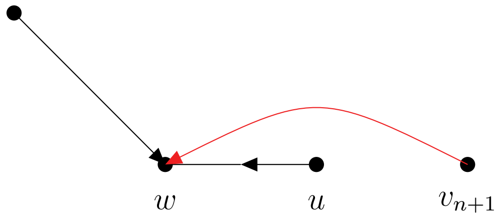
$$\begin{aligned} E(X_{n+1}) &= (1+\frac{p}{n})E(X_n) + q^2 + \frac{q^2p^2}{(1-p^2)} + O(n^{p-1}) \\ &= (1+\frac{p}{n})E(X_n) + \frac{q^2}{1-p^2} + O(n^{p-1}) \end{aligned}$$

In the proof of Theorem 2.5.20 we solved an equation of a similar form. Using Lemma 2.5.5 we get the solution of this recursive relation as

$$E(X_n) = \frac{q^2}{(1-p^2)(1-p)}n + O(n^p).$$

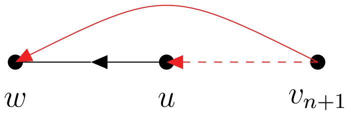
□

We now consider the case of counting the number of P_3^3's in $DCopy(n, p, q, G_{n_0})$. There will be 4 ways in which a P_3^3 can form $DCopy(n, p, q, G_{n_0})$.



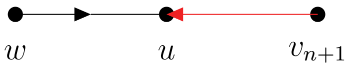
Case 1

In Case 1, a new P_3^3 is formed by v_{n+1} copying a P_3^3 incident to u at time $n+1$ where u is the vertex with out-degree 1 and in-degree 0. Note that there is also a potential additional P_3^3 formed here between v_{n+1}, u and w if there is no edge added between v_{n+1} and u at time $n+1$. We consider this to be a separate case depicted below.



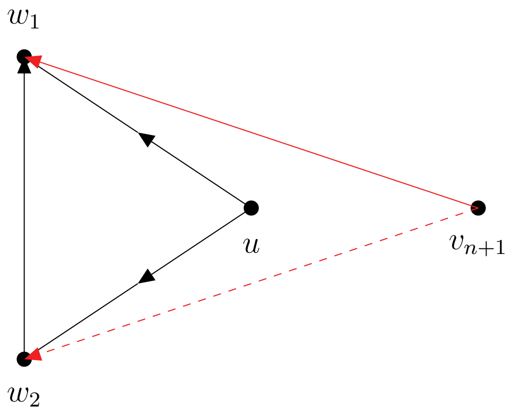
Case 2

In Case 2, a new P_3^3 is formed at time $n+1$ between v_{n+1}, the copy vertex u and each out-neighbour w of u if v_{n+1} copies an edge to w but does not form an edge to u.



Case 3

In Case 3, a new P_3^3 forms between v_{n+1}, the copy vertex u and each in-neighbour w of u in the event that an edge forms from v_{n+1} to u.



Case 4

In Case 4, a new P_3^3 is formed between v_{n+1} and the copy vertex u's out-neighbours w_1 and w_2. In this case we require that an edge does not form between v_{n+1} and w_2, otherwise v_{n+1}, w_1 and w_2 would induce a triangle. Also, note that in the diagram above, if the edge formed from v_{n+1} to w_2 and not from v_{n+1} to w_1 then v_{n+1}, w_1 and w_2 would induce a P_3^2 and not a P_3^3.

Theorem 2.5.23 *Consider the pure directed copy model $G_n = DCopy(n, p, q, G_{n_0})$ with $q > 0$ and $0 < p < 1$. Let $X_n = ind(P_3^3, G_n)$. Then*

$$E(X_n) = \begin{cases} \frac{pq+q+p^2q}{(1-p^2)(1-2p)}n + O(n^{2p}) & \text{if } 0 < p < \frac{1}{2} \\ \frac{pq+q+p^2q}{1-p^2}n\ln(n) + O(n) & \text{if } p = \frac{1}{2} \\ \Theta(n^{2p}) & \text{if } \frac{1}{2} < p < 1 \end{cases}$$

Proof Let $X_n = ind(P_3^3, G_n)$, $e_n = ind(K_2, G_n)$ and $T_n = ind(T_1, G_n)$. To determine $E(X_n)$ we consider the four cases described above. For Case 1, let $P_{v_i,n}^3$ be the set of P_3^3's incident to u at time n in which u is the vertex with out-degree 1 and in-degree 0 in the P_3^3. For a vertex v_i and each $C \in P_{v_i,n}^3$ let $X(C, v_i, n+1)$ be an indicator variable for the event that C is copied at time $n+1$. This event occurs if v_i is selected as the copy vertex at time $n+1$ and v_{n+1} copies the out-edge of u in

C. Therefore $Pr(X(C, v_i, n+1) = 1) = \frac{p}{n}$. For Case 2, let $N^+(v_i, n)$ be the set of out-neighbours of v_i at time n. For a vertex v_i and $w \in N^+(v_i, n)$ let $Y(w, v_i, n+1)$ be an indicator variable for the event that v_i is selected as the copy vertex at time $n+1$, an edge forms from v_{n+1} to w but no edge forms from v_{n+1} to v_i. Therefore $Pr(Y(w, v_i, n+1) = 1) = \frac{p(1-q)}{n}$. For Case 3, let $N^-(v_i, n)$ be the set of in-neighbours of v_i at time n. For a vertex v_i and $w \in N^-(v_i, n)$ let $Z(w, v_i, n+1)$ be an indicator variable for the event that v_i is selected as the copy vertex and an edge forms from v_{n+1} to v_i. We have $Pr(Z(w, v_i, n+1) = 1) = \frac{q}{n}$. For Case 4, let $T_{v_i,n}$ be the set of T_1's in which v_i has out-degree 2. For $C \in T_{v_i,n}$ consisting of vertices v_i, w_1, w_2 we assume wlog that there is a directed edge from w_1 to w_2. For a vertex v_i and $C \in T_{v_i,n}$ let $W(C, v_i, n+1)$ be an indicator variable for the event that v_i gets selected as the copy vertex at time $n+1$ and v_{n+1} forms an edge to w_1 but not to w_2. We have that $Pr(W(C, v_i, n+1) = 1) = \frac{p(1-p)}{n}$. Using conditional expectation we can write,

$$\begin{aligned}
E(X_{n+1}|G_n) &= X_n + E(\sum_{i=1}^{n} \sum_{C \in P^3_{v_i,n}} X(C, v_i, n+1)) + E(\sum_{i=1}^{n} \sum_{w \in N^+(v_i,n)} Y(w, v_i, n+1)) \\
&+ E(\sum_{i=1} \sum_{w \in N^-(v_i,n)} Z(w, v_i, n+1) + E(\sum_{i=1}^{n} \sum_{C \in T_{v_i,n}} W(C, v_i, n+1)) \\
&= X_n + \sum_{i=1}^{n} |P^3_{v_i,n}| \frac{p}{n} + \sum_{i=1}^{n} deg^+(v_i, n) \frac{p(1-q)}{n} + \sum_{i=1}^{n} deg^-(v_i, n) \frac{q}{n} \\
&+ \sum_{i=1}^{n} |T_{v_i,n}| \frac{p(1-p)}{n} \\
&= X_n + \frac{2p}{n} X_n + \frac{p(1-q)}{n} e_n + \frac{q}{n} e_n + \frac{p(1-p)}{n} T_n.
\end{aligned}$$

Note that in the above we have $\sum_{i=1}^{n} |P^3_{v_i,n}| = 2X_n$ as in each P^3 containing 2 vertices with out-degree 1 and in-degree 2.

Applying expectation again gives the recursive relation

$$E(X_{n+1}) = (1+\frac{2p}{n})E(X_n) + \frac{p(1-q)+q}{n}E(e_n) + \frac{p(1-p)}{n}E(T_n).$$

If $q > 0$ and $0 < p < 1$ then $E(e_n) = \frac{q}{1-p}n + O(n^p)$ and $E(T_n) = \frac{q^2 p}{(1-p)(1-p^2)}n + O(n^p)$. Therefore we can write,

$$\begin{aligned} E(X_{n+1}) &= (1+\frac{2p}{n})E(X_n) + \frac{(p(1-q)+q)q}{1-p} + \frac{p(1-p)q^2p}{(1-p)(1-p^2)} + O(n^{p-1}) \\ &= (1+\frac{2p}{n})E(X_n) + \frac{pq+q+p^2q}{1-p^2} + O(n^{p-1}) \end{aligned}$$

If we set $d = \frac{pq+q+p^2q}{1-p^2}$, then we solved the same recursive relation in Theorem 2.5.14. Therefore we obtain

$$E(X_n) = \begin{cases} \frac{pq+q+p^2q}{(1-p^2)(1-2p)}n + O(n^{2p}) & \text{if } \ 0 < p < \frac{1}{2} \\ \frac{pq+q+p^2q}{1-p^2}nln(n) + O(n) & \text{if } \ p = \frac{1}{2} \\ \Theta(n^{2p}) & \text{if } \ \frac{1}{2} < p < 1 \end{cases}$$

□

Combining Theorems 2.5.21, 2.5.22,and 2.5.23 gives the expected number of 3-paths in the pure directed copy model.

Theorem 2.5.24 *Consider the pure directed copy model $G_n = DCopy(n,p,q,G_{n_0})$ with $q > 0$ and $0 < p < 1$. Let $X_n = ind(P_3, G_n)$. Then*

$$E(X_n) = \begin{cases} \frac{qp^3-q^2+2pq^2-1}{(1-p)^2(2p-1)(1+p))}n + O(n^{2p}) & \textit{if } \ 0 < p < \frac{1}{2} \\ \frac{pq+q+p^2q}{1-p^2}nln(n) + O(n) & \textit{if } \ p = \frac{1}{2} \\ \Theta(n^{2p}) & \textit{if } \ \frac{1}{2} < p < 1 \end{cases}$$

2.6 Random Geometric Graphs

In this section we compute the expected subgraph counts in $Geo([0,1]^t, d_\infty, n, r, p)$. Recall that the infinity norm we use is derived from the torus metric. More specifically, d_∞ is derived from the product metric on $([0,1], d_{tor}) \times ([0,1], d_{tor}) \times \ldots \times ([0,1], d_{tor})$. Note that when $t = 1$, we have $d_\infty = d_{tor}$.

Our selection of the metric space $([0,1]^t, d_\infty)$ is due in part because the geometry of the space leads to tractable calculations for subgraph counts. We begin this section with an overview of the work that has already been done in computing expected subgraph counts in RGG's. In particular the $1D$ case of this RGG has received a considerable amount of attention due to its connection to interval graphs. The RGG with $p = 1$ has been well studied and some results concerning expected subgraph counts have been given. We proceed with a discussion of these results.

2.6.1 Overview of Other Works

A comprehensive study of RGG's with $p = 1$ and $r = 1$ is compiled by Penrose in his book [108]. The RGG's with the Euclidean metric that are studied in this book are sometimes called unit disk graphs. In this book, the author gives many results including work on subgraph counts, vertex degrees and connectivity. Specific expected subgraph counts are not given in this book but a general theory about how the expected value behaves is given. In particular, they show that subgraph counts in the RGG's satisfies a central limit theorem. Though [108] provides this powerful general result for expected subgraph counts, it does not provide a comprehensive mechanism for computing expected subgraph counts for a specified subgraph. There have been some papers that have considered specific subgraph counts. In [127] the authors consider ad-hoc networks which they model using $Geo([0,1]^2, d_{tor}, n, r, 1)$. In this work the authors give a precise calculation of the expected number of triangles and

3-paths. In [75] the authors consider a virus spreading in the $Geo([0,1], d_{tor}, n, r, 1)$ and $Geo([0,1]^2, d_{tor}, n, r, 1)$. The authors give precise calculations for the expected number of edges and triangles in these models. In [15] the authors consider the RGG $Geo([0,1]^t, d_\infty, n, r, 1)$ as a model for telecommunication networks. Note that in this paper, d_∞ is derived from the Euclidean metric. In this paper they give a precise count for the expected number of k-cliques for all k in the RGG model. To the best of our knowledge there are no further results on specific subgraph counts in the RGG model in the literature.

2.6.2 General Method for Counting Subgraphs in Random Geometric Graphs

In this section we motivate our procedure for counting subgraphs in RGG's. Here we provide general definitions and theorems that can be used for any RGG.

Suppose we have a graph H of size k and wish to count the expected number of times H appears in $Geo(S, d, n, r, p)$. Our approach follows two steps. We first consider the case that $p = 1$ and compute the probability that $Geo(S, d, k, r, 1) \cong H$. When $p = 1$, two vertices u and v are adjacent if $d(u, v) < r$. In the second step, we consider the probability that H will remain when we "flip on" the value of p. Now edges that were present when $p = 1$ are present with probability p and not present with probability $1 - p$.

All previous work in computing expected subgraph counts in RGGs [75, 15, 127] has only dealt with the $p = 1$ case. Though each of these papers dealt with different geometric spaces, their general approach was the same; compute the probability that a set of k vertices induce H and sum over all possible k-sets. Our approach will be similar but more descriptive than the approaches described in [75, 15, 127].

We define $Pr_H = Pr(Geo(S, d, k, r, 1) \cong H)$. We will see for some graphs H, in some geometric spaces, that $Pr_H = 0$. If $Pr_H > 0$, then we call H a **feasible graph**.

The computation of Pr_H depends on the underlying metric space and in general is a tricky computation. To properly compute Pr_H you must consider all possible orderings of k vertices in the space which lead to an induced H. We will develop general methods for computing Pr_H for the metric space $([0,1]^t, d_\infty)$ in Sections 2.6.3 and 2.6.5.

Once Pr_H is computed, the following lemma can be used to compute

$E(ind(H, Geo(S, d, n, r, 1)))$. This result is observed in [108] but the proof is omitted. The proof is straightforward and we include it here.

Lemma 2.6.1 ([108]) *Consider* $H \in \mathcal{C}_k$. *Let* $G_n = Geo(S, d, n, r, 1)$ *and* $H_n = ind(H, G_n)$. *Then* $E(H_n) = Pr_H \binom{n}{k}$.

Proof For any k-set $S_k \subset V(G_n)$, the probability that S_k induces H in G_n is Pr_H. By the linearity of expectation $E(H_n) = \sum\limits_{S_k \subset V(G_n)} Pr_H = Pr_H \binom{n}{k}$. □

Next we give a lemma for computing the expected number of subgraphs in the general RGG $Geo(S, d, n, r, p)$. The result and the proof will be easier to understand by imagining the construction of $Geo(S, d, n, r, p)$ in the following way. We first form the graph $Geo(S, d, n, r, 1)$. Then, for each edge, we either retain the edge with probability p or remove the edge with probability $1-p$. Now consider $H, G \in \mathcal{C}_k$ with $inj(H, G) \neq 0$. Consider a k-set of $Geo(S, d, n, r, 1)$ which induces G. It is possible that this k-set will induce H in $Geo(S, d, n, r, p)$ if the correct edges are removed to leave an induced copy of H. For example, three vertices all mutually within distance r of one another would form a triangle in $Geo(S, d, n, r, 1)$. These three vertices would induce a 3-path in $Geo(S, d, n, r, p)$ if exactly one of the edges is removed.

Lemma 2.6.2 *Consider* $H \in \mathcal{C}_k$. *Let* $G_n = Geo(S, d, n, r, p)$ *and* $H_n = ind(H, G_n)$. *Then*

$$E(H_n) = \sum_{G \in \mathcal{C}_k} inj(H,G) Pr_G p^{e(H)} (1-p)^{e(G)-e(H)} \binom{n}{k},$$

where $e(G)$ and $e(H)$ are the number of edges in G and H respectively.

Proof Let $\mathcal{T}$ denote the set of all k-sized subsets of $V(G_n)$. For $T \in \mathcal{T}$, let H_T be an indicator variable for the event that T induces H in G_n and $\overline{H_T}$ be the event that T would have induced H if $p = 1$. We can write

$$Pr(H_T = 1) = \sum_{G \in \mathcal{C}_k} Pr(H_T = 1 \cap \overline{G_T} = 1) = \sum_{G \in \mathcal{C}_k} Pr(H_T = 1 | \overline{G_T} = 1) Pr(\overline{G_T} = 1).$$

By definition, $Pr(\overline{G_T} = 1) = Pr_G$. For $Pr(H_T = 1|\overline{G_T} = 1)$, we have for each injective copy of H in G, an induced copy of H can remain if the edges in G which are not in H are removed while the edges in G that are in H remain. Therefore we have that $Pr(H_T = 1|\overline{G_T} = 1) = inj(H,G)p^{e(H)}(1-p)^{e(G)-e(H)}$. Therefore

$$Pr(H_T = 1) = \sum_{G \in \mathcal{C}_k} inj(H,G) Pr_G p^{e(H)} (1-p)^{e(G)-e(H)}.$$

Since H_T is an indicator variable, $Pr(H_T = 1) = E(H_T)$. Overall we have

$$\begin{aligned}
E(H_n) &= \sum_{T \in \mathcal{T}} Pr(H_T = 1) \\
&= \sum_{T \in \mathcal{T}} E(H_T) \\
&= \sum_{T \in \mathcal{T}} \sum_{G \in \mathcal{C}_k} inj(H,G) Pr_G p^{e(H)} (1-p)^{e(G)-e(H)} \\
&= \sum_{G \in \mathcal{C}_k} inj(H,G) Pr_G p^{e(H)} (1-p)^{e(G)-e(H)} \binom{n}{k}.
\end{aligned}$$

□

We see that if we take $p = 1$, the result from Lemma 2.6.2 reduces to give the result from Lemma 2.6.1. In Lemma 2.6.2 the term in the sum which corresponds to H is $Pr_H p^{e(H)} \binom{n}{k}$ as $inj(H, H) = 1$. When $p = 1$, all other terms in the sum disappear and we are left with $Pr_H \binom{n}{k}$.

2.6.3 General Method for Counting Subgraphs in 1D Random Geometric Graphs

In this section we describe a general method for computing Pr_H in 1D geometric graphs. The 1D case is the easiest case because vertices placed in $[0, 1]$ have a natural ordering in the space. Thus considering all possible vertex placements which induce H is more straightforward than in higher dimensions.

Consider vertices $u_1, u_2, \ldots, u_k$ placed in $[0, 1]$. Consider these vertices located in $[0, 1]$ at $x_1 < x_2 < \ldots < x_k$. We use the convention of relabeling the vertices $v_1, v_2, \ldots, v_k$ so that the vertex v_i corresponds to the vertex located at x_i. We define $Pr_H(v_1, v_2, \ldots, v_k) = Pr(Geo([0, 1], d, k, r, 1) \cong H | x_1 < x_2 < \ldots < x_k)$. To determine $Pr_H(v_1, v_2, \ldots, v_k)$ we need to consider all placements of $v_1, v_2, \ldots, v_k$ which satisfy $x_1 < x_2 < \ldots < x_k$ which lead to an induced copy of H. The computation of $Pr_H(v_1, v_2, \ldots, v_k)$ is one fundamental part in the computation of Pr_H. Two further considerations are needed. One of these considerations is simply the $k!$ labelings of the vertices $u_1, u_2, \ldots, u_k$ as $v_1, v_2, \ldots, v_k$. In other words, $k!$ refers to the number of ways to place $u_1, u_2, \ldots, u_k$ can be placed so that they occupy the same position in the space, i.e. the $k!$ orders in which the v_i's can be placed in the space.

The final consideration concerns what we call the *automorphism orderings* of the vertices in the space. It is best to illustrate this idea with an example. Consider three vertices u_1, u_2, u_3. We wish to compute the probability that these 3 vertices induce a 3-path in $[0, 1]$. To do so we consider placing the vertices in the space so that they are located at $x_1 < x_2 < x_3$. The final concern becomes clear when we think; which vertex in the 3-path is located at x_1? In the 3-path, there are 2 vertices of degree 1 and one vertex of degree 2. It matters which order these vertices are placed in the space. For example, suppose we place the degree 2 vertex at x_1 and the degree 1 vertices at x_2 and x_3. For this to induce a 3-path we would need $|x_3 - x_1| < r$, $|x_2 - x_1| \geq r$ and $|x_2 - x_3| \geq r$, which is impossible since $x_1 < x_2 < x_3$. However, if we place the degree 2 vertex at x_2 it is possible for a 3-path to form. We denote the automorphism orbits for a graph G by $A_1, A_2, \ldots, A_m$. We write a_i for a vertex in A_i. We define an **automorphism ordering**, denoted by $a_<$, as an ordering of the vertices by their automorphism orbits. In an automorphism ordering, we do not distinguish between vertices from the same automorphism orbit. The ordering of the vertices in the automorphism ordering corresponds to the ordering in $[0,1]$. We denote $A_<(G)$ as the set of all automorphism orderings in G. In the 3-path there are two automorphism orbits: A_1 containing the vertices of degree 1 and A_2 containing the vertices of degree 2. We have $A_<(P_3) = \{a_1a_1a_2, a_1a_2a_1, a_2a_1a_1\}$. Given $a_< \in A_<(H)$ we define $Pr_{H,a_<}(v_1, v_2, \ldots, v_k) = Pr(Geo([0,1], d, k, r, 1 \cong H)|a_<)$. For the 3-path, for the automorphism orderings $a_1a_1a_2$ and $a_2a_1a_1$, we have $Pr_{P_3,a_<}(v_1, v_2, v_3) = 0$ so that $a_2a_1a_1$ is the only automorphism ordering which contributes a non-zero probability to the calculation of Pr_{P_3}.

We are now ready to give our subgraph counting lemma.

Lemma 2.6.3 *Consider a graph $H \in \mathcal{C}_k$ and let $Geo([0,1], d, n, r, 1)$. Then*

$$Pr_H = k! \sum_{a_< \in A_<(H)} Pr_{H,a_<}(v_1, v_2, \ldots, v_k). \tag{2.13}$$

Proof Consider k vertices $u_1, u_2, \ldots, u_k$. To determine the probability that these k vertices induce H in G_k we need to consider all the arrangements of $u_1, u_2, \ldots, u_k$ in $[0,1]$ which induce H. Let $L(u_1, u_2, \ldots, u_k)$ the set of all labelings of $u_1, u_2, \ldots, u_k$ as $v_1, v_2, \ldots, v_k$ located at $x_1 < x_2 < \ldots < x_k$. Summing over all possible vertex labelings and automorphism orderings we have,

$$\begin{aligned} Pr_H &= \sum_{L(u_1,u_2,\ldots,u_k)} \sum_{a_< \in A_<(H)} Pr_{H,a_<}(v_1, v_2, \ldots, v_k) \\ &= k! \sum_{a_< \in A_<(H)} Pr_{H,a_<}(v_1, v_2, \ldots, v_k) \\ &= \sum_{a_< \in A_<(H)} k! Pr_{H,a_<}(v_1, v_2, \ldots, v_k). \end{aligned}$$

□

2.6.4 Counting Subgraphs in $Geo([0,1], d_{tor}, n, r, p)$

In this section we use Lemma 2.6.2 and Lemma 2.6.3 to compute the expected subgraph counts for size 3 and size 4 connected graphs in $Geo([0,1], d_{tor}, n, r, p)$. For brevity, for the remainder of this section we write $Geo(n, r, p)$ as we work exclusively with $([0,1], d_{tor})$.

Before we proceed, we would like to point out a feature of the metric space $([0,1], d_{tor})$ which simplifies the computation of Pr_H.

With the torus metric, the interval $[0,1]$ can also be viewed as a circle with circumference 1, with the distances between vertices determined by their distance along the circumference of the circle. For our computations, it will be more convenient to imagine the vertices placed in $[0,1]$. Therefore, to determine the torus distance between two points in $[0,1]$, we will have to consider the influence region of vertices close to 0 or 1 as "wrapping" around the boundary and coming out of the other side of interval. This "wrap around" effect serves to complicate the computation of Pr_H. Fortunately, for small enough values of r, we show it is possible to ignore this effect. First consider the following example to illustrate how the value of r affects $Geo(n, r, 1)$.

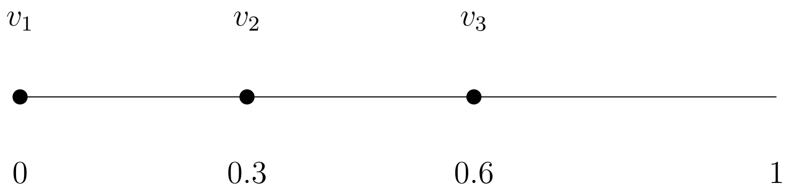


Figure 2.3: Three vertices v_1, v_2, v_3 placed in $Geo(3, r, 1)$.

We see that v_1, v_2, v_3 are located at $x_1 = 0, x_2 = 0.3$ and $x_3 = 0.6$. Therefore $d_{tor}(v_1, v_2) = 0.3$, $d_{tor}(v_2, v_3) = 0.3$ and $d_{tor}(v_1, v_3) = 0.4$. Note that for v_1 and v_3, a distance of 0.4 is determined by wrapping around the boundary of $[0, 1]$. If $r < 0.3$, then no two vertices are within distance r so $Geo(3, r, 1)$ consists of three isolated vertices. If $0.3 \leq r < 0.4$, then $v_1 \sim v_2$, $v_2 \sim v_3$ and $v_1 \nsim v_3$ so $Geo(3, r, 1) \cong P_3$. However if $0.4 \leq r \leq 0.5$, then $v_1 \sim v_2$, $v_2 \sim v_3$ and $v_1 \sim v_3$ so $Geo(3, r, 1) \cong K_3$. Note that if $r \geq \frac{1}{2}$ then $Geo(n, r, 1) \cong K_n$ for every n.

As this example illustrates, depending on the value of r, the subgraph induced by a given set of vertices can change. In particular, for larger values of r the influence region of a vertex can wrap around the boundary which will lead to more complex considerations. To make things simple, we would like to only consider arrangements in which the wrap around effect can be ignored. In other words, we do not wish to consider vertices placed in $(1 - r, 1)$.

In Lemma 2.6.4 we argue that any placement of k vertices in $[0, 1]$ is equivalent to another placement of vertices such that one vertex is located at 0, no vertex lies outside of $[0, 1 - \frac{1}{k}]$ and the distance between each pair of vertices is preserved.

Lemma 2.6.4 *Consider the placement of k vertices $v_1, v_2, \ldots, v_k$ in $[0, 1]$ located at $x_1 < x_2 < \ldots < x_k$. The vertices can be relabeled and have their vertex locations shifted so that all k vertices lie in $[0, 1 - \frac{1}{k}]$, $x_1 = 0$ and torus distance between each pair of vertices is preserved.*

Proof Consider an arbitrary placement of $v_1, v_2, \ldots, v_k$ satisfying $x_1 < x_2 < \ldots < x_k$.

If no vertex lies in $[1-\frac{1}{k},1]$ then map each vertex v_i to $x_i - x_1$. Then $x_1 = 0$, no vertex lies in $[1-\frac{1}{k},1]$ and the torus distance between each pair is preserved.

Suppose now that some vertex does lie in $(1-\frac{1}{k},1]$. Suppose further that for each $i, i+1$, $d_{tor}(x_i, x_{i+1}) = \frac{1}{k}$. In this case, the vertices are maximally spread out in the space and each interval $[\frac{i}{k}, \frac{i+1}{k}), i = 0, 1, \ldots, k-1$ contains exactly one vertex. Map each vertex to the new location $x_i - x_1$. Now the vertices are located at $0, \frac{1}{k}, \ldots, 1-\frac{1}{k}$. Thus $x_1 = 0$, no vertex lies in $(1-\frac{1}{k},1)$ and the torus metric distance between each pair of vertices is preserved.

Again, suppose that some vertex lies in $(1-\frac{1}{k},1]$. If the distance between each pair of vertices is not $\frac{1}{k}$ then it implies that there exists a pair of vertices v_i, v_{i+1} such that $x_{i+1} - x_i > \frac{1}{k}$. Translate the location of each vertex v_j to $x_j - x_i + \frac{k-1}{k}$. Now x_i is located $1-\frac{1}{k}$ and such $x_{i+1} - x_i > \frac{1}{k}$, there is no vertex located in $(1-\frac{1}{k},1]$. At this point we re-labeled the vertices by $v_1, v_2, \ldots, v_k$ so that $x_1 < x_2 < \ldots < x_k$. Now since no vertex is located in $(1-\frac{1}{k},1]$, we translate the vertices one more time by $x_i - x_1$. □

If we only consider $r \leq \frac{1}{k}$, then Lemma 2.6.4 allows us to ignore the "wrap around" effect by only consider placements of $u_1, u_2, \ldots, u_k$ in $[0, 1-\frac{1}{k}]$.

We are now ready to start computing some subgraph counts. We begin with the simple task of counting the number of edges. The calculation of the expected number of edges with $p = 1$ is given in [75]. We restate their result using our method in Lemma 2.6.3 and extend the result to the more general case $0 < p < 1$.

Theorem 2.6.5 *Let $G_n = Geo(n, r, p)$, $e_n = ind(K_2, G_n)$ and $r \leq \frac{1}{2}$. Then $E(e_n) = 2rp\binom{n}{2}$.*

Proof Let $H = K_2$ and let $e_n = ind(H, G_n)$. We first use Lemma 2.6.3 to compute Pr_H. For K_2, there is only one automorphism ordering $a_<$ to consider. Consider

v_1, v_2 located at $x_1 = 0 < x_2$. For v_2 to be adjacent to v_1 is needs to fall into the interval $(0, r)$. Therefore,

$$Pr_{H,a_<}(v_1, v_2) = \int_0^r dx_2 = r.$$

By Lemma 2.6.3, $Pr_H = 2r$. Using Lemma 2.6.2 we have $E(e_n) = 2rp\binom{n}{2}$.

□

We now proceed to use Lemma's 2.6.3 and 2.6.2 to compute the expected subgraph counts of the size 3 and size 4 connected graphs. For the size 4 subgraphs we use the names given in Figure 1.1.

The calculation for the number of triangles in the $p = 1$ case can be found in [75]. We extend this to the general $0 < p < 1$ case and include the calculation for the expected number of P_3's in the following theorem.

Theorem 2.6.6 *Let $G_n = Geo(n, r, p)$ with $0 \leq r \leq \frac{1}{3}$ and let $X_n = ind(K_3, G_n)$ and $Y_n = ind(P_3, G_n)$. Then $E(X_n) = 3r^2p^3\binom{n}{3}$ and $E(Y_n) = (9r^2p^2(1-p) + 3r^2p^2)\binom{n}{3}$.*

Proof $\underline{K_3:}$ In K_3 there is only one automorphism orbit and thus only one automorphism ordering $a_<$ to consider in the computation of Pr_{K_3}. Consider vertices v_1, v_2, v_3 located at $x_1 = 0 < x_2 < x_3$. For all three of these vertices to be adjacent to one another we need $x_1 = 0 < x_2 < x_3 < r$. Therefore,

$$Pr_{K_3,a_<}(v_1, v_2, v_3) = \int_0^r \int_{x_2}^r dx_3 dx_2 = \frac{r^2}{2}.$$

By Lemma 2.6.3,

$$Pr_{K_3} = 3!\frac{r^2}{2} = 3r^2.$$

When $0 < p < 1$, we use Lemma 2.6.2 to obtain

$$E(X_n) = \sum_{G \in \mathcal{C}_3} inj(K_3, G) Pr_G p^{e(K_3)} (1-p)^{e(G)-e(H)} \binom{n}{3}$$

Since K_3 is the only $G \in \mathcal{C}_3$ such that $inj(K_3, G) \neq 0$, we have $E(X_n) = 3r^2p^3\binom{n}{3}$.

$\underline{P_3:}$ Let $Y_n = ind(P_3, G_n)$. In a 3-path there are two automorphism orbits A_1 and A_2. Let A_1 contain the two vertices of degree 1 and let A_2 contain the vertex of degree 2. There are three distinct automorphism orderings of these vertices: $a_{<,1} = a_1a_1a_2, a_{<,2} = a_1a_2a_1$, and $a_{<,3} = a_2a_1a_1$. Of these three orderings, only $a_{<,2}$ results in a non-zero $Pr_{P_3,a_<}(v_1, v_2, v_3)$. For $a_{1,<}$, the two vertices in A_1 are not adjacent to one another so $x_2 > r$. However the vertex in A_2 is adjacent to both the vertices in A_1. However since this vertex is located at x_3 and $x_3 > x_2 > r$ it cannot be adjacent to the vertex located at x_1. Thus $Pr_{P_3,a_{<,1}}(v_1, v_2, v_3) = 0$ for this automorphism ordering. A similar explanation gives $Pr_{P_3,a_{<,3}}(v_1, v_2, v_3) = 0$. Consider the automorphism ordering $a_1a_2a_1$. For the vertices to induce a P_3 we require that $x_1 = 0 < x_2 < r < x_3 < x_2 + r$. Therefore,

$$Pr_{P_3,a_{<,2}}(v_1, v_2, v_3) = \int_0^r \int_r^{x_2+r} dx_3 dx_2 = \frac{r^2}{2}.$$

By Lemma 2.6.3, $Pr_{P_3} = 3!\frac{r^2}{2} = 3r^2$. Since $inj(P_3, P_3) = 1$ and $inj(P_3, K_3) = 3$, by Lemma 2.6.2 we have $E(Y_n) = (9r^2p^2(1-p) + 3r^2p^2)\binom{n}{3}$.

□

It is interesting to note that in $Geo(n, r, 1)$ we expect to see the same number of P_3 and K_3 if $r \leq \frac{1}{3}$.

Theorem 2.6.6 serves as a good example for the reader in how Lemmas 2.6.2 and 2.6.3 can be used to find expected subgraph counts in $Geo(n, r, p)$. We now give the expected subgraph counts for the size 4 connected graphs. We use the g_i notation for these subgraphs which was introduced in Figure 1.1. In an attempt to not overwhelm the reader, we break the result into two Theorems: one which gives the Pr_{g_i} values and another which gives the expected subgraph counts.

Theorem 2.6.7 *Consider $G_n = Geo(4, r, 1)$ where $0 \leq r < \frac{1}{4}$. Then*

g_i	g_3	g_4	g_5	g_6	g_7	g_8
Pr_{g_i}	0	$8r^3$	$8r^3$	0	$4r^3$	$4r^3$

Proof $\underline{g_3 :}$ In g_3 there are two automorphism orbits: A_1 which contains all the degree 1 vertices and A_2 which contains the degree 3 vertex. There are 4 automorphism orderings to consider: $a_{<,1} = a_1a_1a_1a_2, a_{<,2} = a_1a_1a_2a_1, a_{<,3} = a_1a_2a_1a_1$ and $a_{<,4} = a_2a_1a_1a_1$. We argue that for each of these orderings

$Pr_{g_{3},a_{<,i}}(v_1, v_2, v_3, v_4) = 0$. For $a_{<,1}$, the vertex in A_2 is adjacent to each of the vertices in A_1 but no two vertices in A_1 are adjacent to one another. According to this ordering we have $x_1 = 0 < x_2 < x_3 < x_4$. The vertex in A_2 is located at x_4. Since the vertex in A_2 is adjacent to the vertex located at x_1 it follows that $x_4 \in [0, r]$. Since $x_2 < x_3 < x_4 < r$ it follows that the vertices at x_2 and x_3 are also adjacent to the vertex at x_1. This gives a contradiction. A similar argument shows that the other automorphism orderings result in $Pr_{g_{3},a_{<,i}}(v_1, v_2, v_3, v_4) = 0$.

$\underline{g_4 = P_4 :}$ In P_4, there are two automorphism orbits: A_1 which consists of the degree 1 vertices and A_2 which consists of the degree 2 vertices. There are 6 automorphism orderings: $a_{<,1} = a_1a_1a_2a_2, a_{<,2} = a_1a_2a_1a_2, a_{<,3} = a_1a_2a_2a_1, a_{<,4} = a_2a_2a_1a_1, a_{<,5} = a_2a_1a_2a_1$, and $a_{<,6} = a_2a_1a_1a_2$. It is straightforward to check that only $a_{<,3} = a_1a_2a_2a_1$ results in a non-zero $Pr_{P_4,a_<}$. For this ordering of

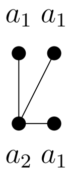
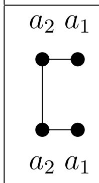
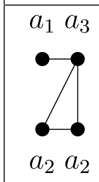
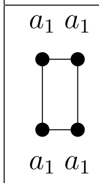
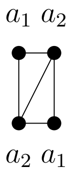
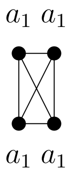
g_i	$A_<(g_i)$
	$a_1a_1a_1a_2, a_1a_1a_2a_1, a_1a_2a_1a_1, a_2a_1a_1a_1$
	$a_1a_1a_2a_2, a_1a_2a_1a_2, a_1a_2a_2a_1, a_2a_2a_1a_1, a_2a_1a_2a_1, a_2a_1a_1a_3$
	$a_1a_2a_2a_3, a_1a_2a_3a_2, a_1a_3a_2a_2, a_2a_1a_2a_3, a_2a_1a_3a_2, a_2a_2a_1a_3, a_2a_2a_3a_1, a_2a_3a_1a_2, a_2a_3a_2a_1, a_3a_1a_2a_2, a_3a_2a_1a_2, a_3a_2a_2a_1$
	$a_1a_1a_1a_1$
	$a_1a_1a_2a_2, a_1a_2a_1a_2, a_1a_2a_2a_1, a_2a_2a_1a_1, a_2a_1a_2a_1, a_2a_1a_1a_2$
	$a_1a_1a_1a_1$

Figure 2.4: Automorphism orderings for $\mathcal{C}_4$

vertices we need placements of vertices at $x_1 = 0 < x_2 < r < x_3 < x_2 + r < x_4 < x_3 + r$ to get an induced copy of P_4. This gives

$$Pr_{P_4,a_{<,3}}(v_1, v_2, v_3, v_4) = \int_0^r \int_r^{x_2+r} \int_{x_2+r}^{x_3+r} dx_4 dx_3 dx_2 = \frac{r^3}{3}.$$

Using Lemma 2.6.3 we have $Pr_{P_4} = 4!\frac{r^3}{3} = 8r^3$.

$\underline{g_5:}$ In g_5 there are 3 automorphism orbits: A_1 which contains the degree 1 vertex, A_2 which contains the degree 2 vertices and A_3 which contains the degree 3 vertex. We have 12 automorphism orderings of the vertices: $a_{<,1} = a_1a_2a_2a_3, a_{2,<} = a_1a_2a_3a_2, a_{3,<} = a_1a_3a_2a_2, a_{<,4} = a_2a_2a_1a_3, a_{<,5} = a_2a_2a_3a_1, a_{<,6} = a_2a_1a_2a_3, a_{<,7} = a_2a_1a_3a_2, a_{<,8} = a_2a_3a_1a_2, a_{<,9} = a_2a_3a_2a_1, a_{<,10} = a_3a_1a_2a_2, a_{<,11} = a_3a_2a_1a_2$, and $a_{<,12} = a_3a_2a_2a_1$. It is straightforward to check the automorphism orderings $a_{<,3} = a_1a_3a_2a_2$ and $a_{<,5} = a_2a_2a_3a_1$ are the only two of the 12 which gives a non-zero value for $Pr_{g_5,a_{<,i}}(v_1, v_2, v_3, v_4)$. Let's consider $a_1a_3a_2a_2$ first. For this ordering to induce a g_5 we require $x_1 = 0 < x_2 < r < x_3 < x_4 < x_2 + r$. This gives

$$Pr_{g_5,a_{<,3}}(v_1, v_2, v_3, v_4) = \int_0^r \int_r^{x_2+r} \int_{x_3}^{x_2+r} dx_4 dx_3 dx_2 = \frac{r^3}{6}.$$

For the ordering $a_2a_2a_3a_1$ to induce a g_5 we require $x_1 = 0 < x_2 < x_3 < r < x_2 + r < x_4 < x_3 + r$. This gives

$$Pr_{g_5,a_{<,5}}(v_1, v_2, v_3, v_4) = \int_0^r \int_{x_2}^r \int_{x_2+r}^{x_3+r} dx_4 dx_3 dx_2 = \frac{r^3}{6}.$$

Using Lemma 2.6.3 we have

$$\begin{aligned} Pr_{g_5} &= \sum_{a_< \in A_<} k! Pr_{g_5, a_<}(v_1, v_2, v_3, v_4) \\ &= 4!\frac{r^3}{6} + 4!\frac{r^3}{6} \\ &= 8r^3. \end{aligned}$$

$g_6 = C_4$: There is only one automorphism orbit for C_4 and thus only one automorphism ordering to check. Consider the order $x_1 = 0 < x_2 < x_3 < x_4$. Each of these vertices must be adjacent to exactly two of the other vertices. Since v_2 and v_3 are the two vertices closest to v_1, it follows that v_1 is adjacent to both of these vertices. It follows that v_4 is also adjacent to v_2 and v_3 (v_4 has two neighbours one of which can not be v_1). Since v_4 is adjacent to v_2 it follows that $x_4 < x_2 + r$. Since $x_3 < x_4$, we must also have $x_3 < x_2 + r$ so v_2 and v_3 are adjacent as well. But then v_2 and v_3 are universal vertices. A contradiction which implies that $Pr_{C_4, a_<}(v_1, v_2, v_3, v_4) = 0$.

g_7 : In g_7 there are two automorphism orbits: A_1 containing the degree 2 vertices and A_2 containing the degree 3 vertices. There are 6 automorphism orderings: $a_{<,1} = a_1a_1a_2a_2, a_{<,2} = a_1a_2a_1a_2, a_{<,3} = a_1a_2a_2a_1, a_{<,4} = a_2a_2a_1a_1, a_{<,5} = a_2a_1a_2a_1$, and $a_{<,6} = a_2a_1a_1a_2$. It is straightforward to check that $a_{<,3}$ is the only arrangement in which $Pr_{g_7, a_{<,i}} \neq 0$. For this ordering to induce a g_7 we require $x_1 = 0 < x_2 < x_3 < r < x_4 < x_2 + r$. This gives

$$Pr_{g_7, a_{<,3}}(v_1, v_2, v_3, v_4) = \int_0^r \int_{x_2}^r \int_r^{x_2 + r} dx_3 dx_2 dx_1 = \frac{r^3}{6}.$$

Using Lemma 2.6.3 we have $Pr_{g_7} = 4!\frac{r^3}{6} = 4r^3$.

$g_8 = K_4$: There is only one automorphism class for K_4. For this ordering to induce a K_4 we require that $x_1 = 0 < x_2 < x_3 < x_4 < r$. This gives

$$Pr_{K_{4,a_<}} = \int_0^r \int_{x_2}^r \int_{x_3}^r dx_4 dx_3 dx_2 = \frac{r^3}{6}.$$

Using Lemma 2.6.3 we have $Pr_{K_4} = 4!\frac{r^3}{6} = 4r^3$.

□

In Figure 2.5, we experimentally verify our results from Theorem 2.6.7 and 2.6.6 by running 5000 simulations of $Geo(1000, 0.01, 1)$ and comparing our expected subgraph calculations with the expected value of the samples.

We obtain the expected subgraph counts in $Geo(n, r, p)$ by taking our result above in Theorem 2.6.7 and plugging them into Lemma 2.6.2. The only detail which is left to work out is to determine for each $G \in \mathcal{C}_k$ and each $g_i \in \mathcal{C}_k$ the number of injective copies of g_i in each G. This is a straightforward exercise and we tabulate the results in Figure 2.6.

Theorem 2.6.8 *Let $G_n = Geo(n, r, p)$ where $0 \leq r < \frac{1}{4}$. Let $X_{i,n} = ind(g_i, G_n)$. Then*

g_i	$E(X_{i,n})$
g_3	$(8r^3p^3(1-p) + 8r^3p^3(1-p)^2 + 16r^3p^3(1-p)^3)\binom{n}{4}$
g_4	$(48r^3p^3(1-p)^3 + 24r^3p^3(1-p)^2 + 16r^3p^3(1-p) + 8r^3p^3)\binom{n}{4}$
g_5	$(8r^3p^4 + 48r^3p^4(1-p)^2 + 16r^3p^4(1-p))\binom{n}{4}$
g_6	$(12r^3p^4(1-p)^2 + 4r^3p^4(1-p))\binom{n}{4}$
g_7	$(4r^3p^5 + 24r^3p^5(1-p))\binom{n}{4}$
g_8	$4r^3p^6\binom{n}{4}$

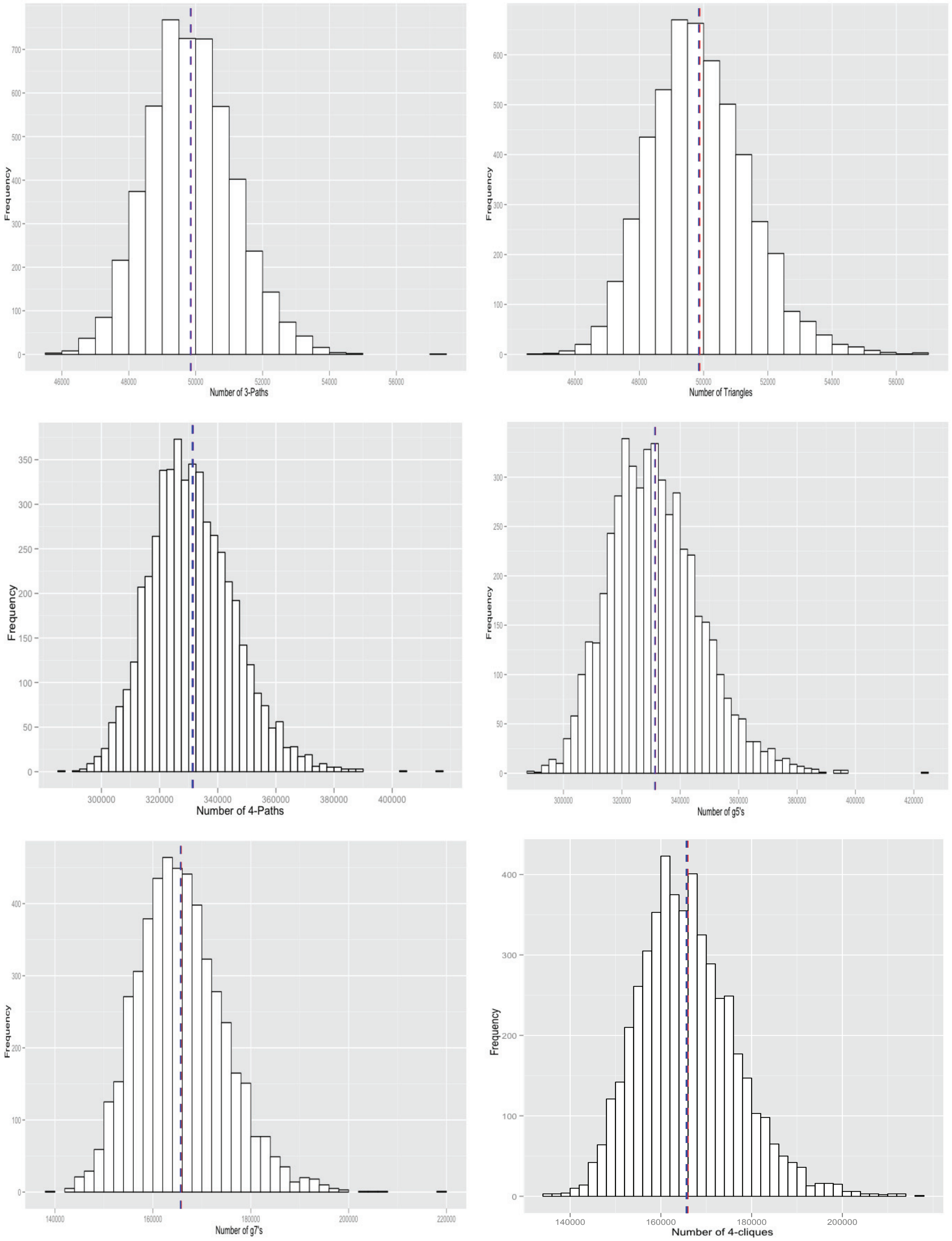


Figure 2.5: Histograms showing subgraph counts for $P_3, K_3, P_4, g_5, g_7, K_4$ from 5000 simulations of $Geo([0,1], d_{tor}, 1000, 0.01, 1)$. The blue line corresponds to the expectations from Theorems 2.6.6 and 2.6.7 and the red line corresponds to the average of the sample.

$inj(g_i, g_j)$	g_3	g_4	g_5	g_6	g_7	g_8
g_3	1	0	1	0	2	4
g_4	0	1	2	4	6	12
g_5	0	0	1	0	4	12
g_6	0	0	0	1	1	3
g_7	0	0	0	0	1	6
g_8	0	0	0	0	0	1

Figure 2.6: Number of injective copies of g_i contained in g_j

Proof $\underline{g_3}$: Using the results from Theorem 2.6.7 and Lemma 2.6.3 we have

$$E(X_{3,n}) = \sum_{G \in \mathcal{C}_k} inj(g_3, G) Pr_{g_3} p^{e(g_3)} (1-p)^{e(G)-e(g_3)} \binom{n}{4}.$$

From Theorem 2.6.7 we have that $Pr_{g_3} = 0$ so the only non-zero terms to consider are those in which $inj(g_3, g_j) \neq 0$. These give,

$$\begin{aligned}
E(X_{3,n}) &= \sum_{G \in \mathcal{C}_k} inj(g_3, G) Pr_G p^{e(g_3)} (1-p)^{e(G)-e(g_3)} \binom{n}{4} \\
&= (inj(g_3, g_5) Pr_{g_5} p^{e(g_3)} (1-p)^{e(g_5)-e(g_3)} + inj(g_3, g_7) Pr_{g_7} p^{e(g_3)} (1-p)^{e(g_7)-e(g_3)} \\
&+ \; inj(g_3, g_8) Pr_{g_8} p^{e(g_3)} (1-p)^{e(g_8)-e(g_3)}) \binom{n}{4} \\
&= ((1)(8r^3)p^3(1-p) + (2)(4r^3)p^3(1-p)^2 + (4)(4r^3)p^3(1-p)^3) \binom{n}{4} \\
&= (8r^3p^3(1-p) + 8r^3p^3(1-p)^2 + 16r^3p^3(1-p)^3) \binom{n}{4}
\end{aligned}$$

The results for the remaining graphs follow in an identical manner.

□

The method used in Theorems 2.6.6 and 2.6.8 can be used to count the expected number of subgraphs in $Geo(n, r, p)$ for any subgraph you may be interested in. It is straightforward to use this method to count the expected number of subgraphs of any complete subgraph of size m. The expected number of complete subgraphs of size m in the $p = 1$ case was previously computed in [15]. We prove their result using our method and extend it to the $0 < p < 1$ case in the following theorem.

Theorem 2.6.9 *Let $G_n = Geo(n, r, p)$ with $r \leq \frac{1}{m}$ and let K_m be a complete graph on $m \leq n$ vertices. Then $E(ind(K_m, G_n)) = mr^{m-1}p^{\binom{m}{2}}\binom{n}{m}$.*

Proof Let $X_n = ind(K_m, G_n)$. In K_m, there is only one automorphism orbit and thus only one automorphism ordering $a_<$. For this ordering to induce a K_m we require that $x_1 = 0 < x_2 < \ldots < x_m < r$. This gives

$$Pr_{K_m,a_<}(x_1, x_2, \ldots, x_m) = \int_0^r \int_{x_2}^r \int_{x_3}^r \cdots \int_{x_{m-1}}^r dx_m dx_{m-1} \ldots dx_2 = \frac{r^{m-1}}{(m-1)!}.$$

This gives $Pr_{K_m} = m!\frac{r^{m-1}}{(m-1)!} = mr^{m-1}$. Using Lemma 2.6.2 we have $E(X_n) = mr^{m-1}p^{\binom{m}{2}}\binom{n}{m}$. □

Counting Subgraphs in $Geo([0, 1], d_{euc}, n, r, p)$

Before continuing our investigation of $Geo([0, 1]^t, d_\infty, n, r, p)$ in higher dimensions it would be interesting to first consider the 1D RGG with the Euclidean metric instead of the torus metric. The main difference between the Euclidean metric and the torus metric is that under the Euclidean metric each point in $[0, 1]$ is distinct. For this reason, when computing $Pr_{H,a_<}(x_1, \ldots, x_k)$, we can no longer fix the location of x_1 at 0 or use Lemma 2.6.4. This makes the computation of Pr_H more difficult because now that x_1 is not at a fixed location, there are more vertex arrangements to consider. However, Lemma 2.6.2 and Lemma 2.6.3 can still be used. The 1D RGG with the Euclidean metric was studied in [13, 14, 15] for the case $p = 1$. The following calculation for the number of edges is originally given in [15] for the $p = 1$ case. We extend to the $0 < p < 1$ case.

Theorem 2.6.10 *Let $G_n = Geo([0,1], d_{euc}, n, r, p)$ and let $e_n = ind(K_2, G_n)$. Then $E(e_n) = (2r - r^2)p\binom{n}{2}$.*

Proof In K_2 there is only one automorphism class and one automorphism ordering $a_<$ to consider. Consider the placement of vertices v_1 and v_2 at $x_1 < x_2$. If x_1 falls into $[0, 1-r]$ the right hand side of the influence region of v_1 lies entirely in $[0,1]$. If $x_1 \in [1-r, 1]$ then the left hand side of v_1's gets truncated by the barrier at 1. We can write

$$\begin{aligned} Pr_{K_2,a_<}(v_1, v_2) &= \int_0^{1-r}\int_{x_1}^{x_1+r} dx_2 dx_1 + \int_{1-r}^{1}\int_{x_1}^{1} dx_2 dx_1 \\ &= r(1-r) + r - \frac{1}{2} + \frac{(1-r)^2}{2} \\ &= r - \frac{r^2}{2}. \end{aligned}$$

By Lemma 2.6.3 $Pr_{K_2} = 2!(r - \frac{r^2}{2}) = 2r - r^2$. By Theorem 2.6.2 we have $E(e_n) = (2r - r^2)p\binom{n}{2}$.

□

We see that if r is small, then the number of edges in the 1D RGG with the torus metric and the Euclidean metric are almost the same since r^2 will be small. This is not surprising because when r is small the border effect created by using the Euclidean metric is minimal.

We conclude our work with the 1D RGG by considering the subgraph counts for K_3 and P_3. The calculation for K_3 for the $p = 1$ case was originally given in [15]. We extend this result for the case $0 < p < 1$.

Theorem 2.6.11 *Let $G_n = Geo([0,1], d_{euc}, n, r, p)$ with $0 \le r \le 1$ and let $X_n = ind(K_3, G_n)$. Then $E(X_n) = r^2(3 - 2r)p^3\binom{n}{3}$.*

Proof In K_3 there is only one automorphism orbit and only one automorphism ordering $a_<$ to consider. Consider the placement of the vertices v_1, v_2, v_3 at $x_1 < x_2 < x_3$. Much like the case for K_2, we consider the case where $x_1 \in [0, 1-r]$ and $x_1 \in [1-r, 1]$ separately. We can write

$$\begin{aligned} Pr_{K_3, a_<}(v_1, v_2, v_3) &= \int_0^{1-r}\int_{x_1}^{x_1+r}\int_{x_2}^{x_1+r} dx_3dx_2dx_1 + \int_{1-r}^{1}\int_{x_1}^{1}\int_{x_2}^{1} dx_3dx_2dx_1 \\ &= \frac{r^2(1-r)}{2} + \frac{r}{2} - \frac{1}{3} + \frac{(1-r)^2}{2} - \frac{(1-r)^3}{6} \\ &= r^2(\frac{1}{2} - \frac{r}{3}). \end{aligned}$$

From Lemma 2.6.3 we have that $Pr_{K_3} = 3!r^2(\frac{1}{2} - \frac{r}{3}) = r^2(3-2r)$. By Lemma 2.6.2 we have that $E(X_n) = r^2(3-2r)p^3\binom{n}{3}$.

□

Theorem 2.6.12 *Consider $G_n = Geo([0,1], d_{euc}, n, r, p)$ and let $X_n = ind(P_3, G_n)$. Then*

$$E(X_n) = \begin{cases} ((3r^2 - 4r^3)p^2 + 3r^2(3-2r)p^2(1-p))\binom{n}{3} & \text{if } r \in [0, \frac{1}{2}) \\ ((r-1)^2(4r-1)p^2 + 3r^2(3-2r)p^2(1-p))\binom{n}{3} & \text{if } r \in [\frac{1}{2}, 1] \end{cases}$$

.

Proof As was the case with the torus metric, only the automorphism ordering $a_< = a_1a_2a_1$ leads to an induced P_3. Consider vertices v_1, v_2, v_3 located at $x_1 < x_2 < x_3$. The cases with $r \leq \frac{1}{2}$ and $r > \frac{1}{2}$ need to be dealt with separately. The reason for this, which will become clear as we progress through the proof, is that there are less possible vertex arrangements if $r > \frac{1}{2}$.

Case 1: $0 < r < \frac{1}{2}$ There are two possible regions in which v_1 can be placed. The first interval is $x_1 \in [0, 1-2r]$. In this case $x_2 \in [x_1, x_1+r]$ and $x_3 \in [x_1+r, x_2+r]$. The second possible location for x_1 is in $[1-2r, 1-r]$. Note that if $x_1 \in [1-r, 1]$ then there would be no where to place x_3 so that $d_{euc}(x_1, x_3) < r$. If $x_1 \in [1-2r, 1-r]$ then we can either have $x_2 \in [x_1, 1-r]$ and $x_3 \in [x_1+r, x_2+r]$ or $x_2 \in [1-r, x_1+r]$ and $x_3 \in [x_1+r, 1]$. We compute

$$\begin{aligned} Pr_{P_{3,a_<}}(v_1, v_2, v_3) &= \int_0^{1-2r} \int_{x_1}^{x_1+r} \int_{x_1+r}^{x_2+r} dx_3 dx_2 dx_1 \\ &+ \int_{1-2r}^{1-r} \int_{x_1}^{1-r} \int_{x_1+r}^{x_2+r} dx_3 dx_2 dx_1 \\ &+ \int_{1-2r}^{1-r} \int_{1-r}^{x_1+r} \int_{x_1+r}^{1} dx_3 dx_2 dx_1 \\ &= \frac{1}{2}r^2 - \frac{2}{3}r^3. \end{aligned}$$

By Lemma 2.6.3, we have that $Pr_{P_3} = 3!(\frac{1}{2}r^2 - \frac{2}{3}r^3) = r^2(3-4r)$. Using Theorem 2.6.2 we get $((3r^2 - 4r^3)p^2 + 3r^2(3-2r)p^2(1-p))\binom{n}{3}$.

Case 2: $\frac{1}{2} \leq r \leq 1$ For Case 2, it is no longer possible to place $x_1 \in [0, 1-2r]$ as $1 - 2r < 0$. We now must place $x_1 \in [0, 1-r]$. Then we can either have $x_2 \in [x_1, 1-r]$ and $x_3 \in [x_1+r, x_2+r]$ or $x_2 \in [1-r, x_1+r]$ and $x_3 \in [x_1+r, 1]$. We compute

$$\begin{aligned} Pr_{P_{3,a_<}}(v_1, v_2, v_3) &= \int_0^{1-r} \int_{x_1}^{1-r} \int_{x_1+r}^{x_2+r} dx_3 dx_2 dx_1 \\ &+ \int_0^{1-r} \int_{1-r}^{x_1+r} \int_{x_1+r}^{1} dx_3 dx_2 dx_1 \\ &= \frac{(4r-1)(r-1)^2}{6}. \end{aligned}$$

By Lemma 2.6.3 we have $Pr_{P_3} = 3!\frac{(4r-1)(r-1)^2}{6} = (4r-1)(r-1)^2$. Using Theorem 2.6.2 we have $E(X_n) = ((r-1)^2(4r-1)p^2 + 3r^2(3-2r)p^2(1-p))\binom{n}{3}$. □

2.6.5 Counting Subgraphs in RGG's in Higher Dimensions

Computing subgraph probabilities is naturally more difficult in higher dimensions because the geometry becomes more complicated. For this reason, we consider a geometric graph on $[0,1]^t$ using the d_∞ metric induced by the torus metric. For this metric, it is possible to use the geometry in one dimension to obtain the subgraph counts in higher dimensions. Before proceeding in our study of this RGG, we summarize the results in the literature for expected subgraph counts in RGG's in higher dimensions. We also extend these results to the $0 < p < 1$ case using Theorem 2.6.2

Subgraphs in $Geo([0,1]^2, d_{tor}, n, r, p)$

In [127] and [75], the authors study the RGG $Geo([0,1]^2, d_{tor}, n, r, 1)$. The number of edges, triangles and 3-paths in $Geo([0,1]^2, d_{tor}, n, r, 1)$ are computed in [127]. The edge and triangle calculations are also performed in [75].

Theorem 2.6.13 *Consider* $G_n = Geo([0,1]^2, d_{tor}, n, r, 1)$. *Let* $e_n = ind(K_2, G_n)$, $X_n = ind(K_3, G_n)$ *and* $Y_n = ind(P_3, G_n)$. *Then* $E(e_n) = \pi r^2\binom{n}{2}$, $E(X_n) = (\pi - \frac{3\sqrt{3}}{4})\pi r^4\binom{n}{3}$ *and* $E(Y_n) = \frac{3\sqrt{3}}{4}\pi r^4\binom{n}{3}$.

We use Theorem 2.6.2 to extend these results to $Geo([0,1]^2, d_{tor}, n, r, p)$.

Theorem 2.6.14 *Consider* $G_n = Geo([0,1]^2, d_{tor}, n, r, p)$. *Let* $e_n = ind(K_2, G_n)$, $X_n = ind(K_3, G_n)$ *and* $Y_n = ind(P_3, G_n)$. *Then* $E(e_n) = \pi r^2 p\binom{n}{2}$, $E(X_n) = (\pi - \frac{3\sqrt{3}}{4})\pi r^4 p^3\binom{n}{3}$ *and* $E(Y_n) = (3(\pi - \frac{3\sqrt{3}}{4})\pi r^4 p^2(1-p) + \frac{3\sqrt{3}}{4}\pi r^4 p^2)\binom{n}{3}$.

Proof Follows from the results from Theorem 2.6.13 and Theorem 2.6.2.

□

Counting Subgraphs in $Geo([0,1]^t, d_\infty, n, r, p)$

In this section we consider the RGG $Geo([0,1]^t, d_\infty, n, r, p)$ where d_∞ is the infinity norm induced from the product metric on $([0,1], d_{tor}) \times ([0,1], d_{tor}) \times \ldots \times ([0,1], d_{tor})$. A similar RGG is studied in [13, 14, 15] where the metric in each dimension is the Euclidean metric, not the torus metric.

We begin by showing that $G_k = Geo([0,1]^t, d_\infty, k, r, 1)$ is equal to the graph intersection of t 1D RGG's. Consider vertices $u_1, u_2, \ldots, u_k$ placed in $[0,1]^t$ with each vertex u_i located at $(x_i^j)_{j=1}^t$. We write $S = [0,1]^t = S_1 \times S_2 \times \ldots \times S_t$. We define the **$i$-th projection of** G_k to be the random geometric graph $G_k^i = Geo(S_i, d_{tor}, k, r, 1)$ on the vertex set $u_1, u_2, \ldots, u_k$ where u_j is located at $x_j = x_j^i$. Note the location of vertices in G_k^i is the projection of the vertex location in G_k to the i-th coordinate. As an illustration, let u_1 be a vertex in $Geo([0,1]^2, d_\infty, k, r, 1)$ which is located at $(\frac{1}{2}, \frac{1}{3})$. Then in G_k^1, u_1 is located at $\frac{1}{2}$ and in G_k^2, u_1 is located at $\frac{1}{3}$.

For two graphs G and H, we define the **intersection of G and H** as the graph $G \cap H$ with $V(G \cap H) = V(G) \cap V(H)$ and $E(G \cap H) = E(G) \cap E(H)$.

Lemma 2.6.15 *Let $G_k = Geo([0,1]^t, d_\infty, k, r, 1)$ and let G_k^i be the i-th projection of G_k^i. Then $G_k = \cap_{i=1}^t G_k^i$.*

Proof Consider k vertices $u_1, u_2, \ldots, u_k$ where u_i is located at $(x_j^i)_{j=1}^t$. We first note that G_k and $\cap_{i=1}^t G_k^i$ are on the same vertex set. To prove the lemma we show that each edge in G_k is also an edge in $\cap_{i=1}^t G_k^i$ and vice-versa.

Suppose $(u_i, u_j) \in E(G_k)$. Then

$$d_\infty(x_i, x_j) = max(d_{tor}(x_i^1, x_j^1), d_{tor}(x_i^2, x_j^2), \ldots, d_{tor}(x_i^t, x_j^t)) < r.$$

This implies that in each coordinate l, $d_{tor}(x_i^l, x_j^l) < r$. Therefore (u_i, u_j) is an edge is each projection G_k^l. Therefore (u_i, u_j) is an edge in $\cap_{i=1}^t G_k^l$.

Now suppose $(u_i, u_j) \in E(\cap_{i=1}^{t} G_k^i)$. Then (u_i, u_j) is in each l projection G_k^l. This implies that in an coordinate l that $d_{tor}(x_i^l, x_j^l) < r$. Therefore $d_\infty(x_i, x_j) = max(d_{tor}(x_i^1, x_j^1), d_{tor}(x_i^2, x_j^2), \ldots, d_{tor}(x_i^t, x_j^t)) < r$ and (u_i, u_j) is also an edge in G_k.

□

The result of Lemma 2.6.15 states that G_k is equivalent to the intersection of t $1D$ RGG's with the torus metric. We use this fact to develop a method of counting subgraphs in $Geo([0,1]^t, d_\infty, n, r, 1)$ which relies on the projections of G_k. The basic idea is to sum over all possible sets of projections $(G_k^1, G_k^2, \ldots, G_k^t)$ such that $\cap_{i=1}^{t} G_k^i \cong H$. Since each G_k^i is formed independently, we can simply multiply the probabilities of each projection forming in $Geo([0,1], d_{tor}, k, r, 1)$ together. There is one additional fact that needs to be accounted for. In 1D, all labelings of the vertices $u_1, u_2, \ldots, u_k$ were allowable but this will not be the case in tD for the intersection of t 1D RGG's. For example, if we have $G_3^1 \cong P_3$ and $G_3^2 \cong P_3$ it is not necessarily the case that $G_3^1 \cap G_3^2 \cong P_3$. Any labeling of G_3^1 is allowed but G_3^2 must be labeled so that edges exist between the same vertices in G_3^2 as G_3^1. We consider this specific case further in Example 2.6.16. In Section 2.6.3 we computed the probability $Pr_H = Pr(Geo([0,1], d_{tor}, k, r, 1) \cong H)$. Recall that our method for doing so was to fix a labeling for $Geo([0,1], d_{tor}, k, r, 1)$ and compute the probability that for this labeling $Geo([0,1], d_{tor}, k, r, 1) \cong H$. We then simply summed over all labelings (or multiplied by $k!$) to account for all labelings of $Geo([0,1], d_{tor}, k, r, 1)$ to get Pr_H. For each projection, it will not always be the case that each vertex labeling is possible. We introduce the following notation which will aid in the developing a general method for counting subgraphs in $Geo([0,1]^t, d_{tor}, k, r, 1)$. Let $Pr_{H,t} = Pr(Geo([0,1]^t, d_\infty, k, r, 1) \cong H)$. For a fixed ordering of vertices in $Geo([0,1], d_{tor}, k, r, 1)$ denoted by ϕ, let $Pr_{H,\phi} = Pr(Geo([0,1], d_{tor}, k, r, 1) \cong H|\phi)$. We can easily return to the probability that a fixed labeling of $Geo([0,1], d_{tor}, k, r, 1)$ is isomorphic to H by dividing the results of Theorems 2.6.6 and 2.6.7 by $k!$.

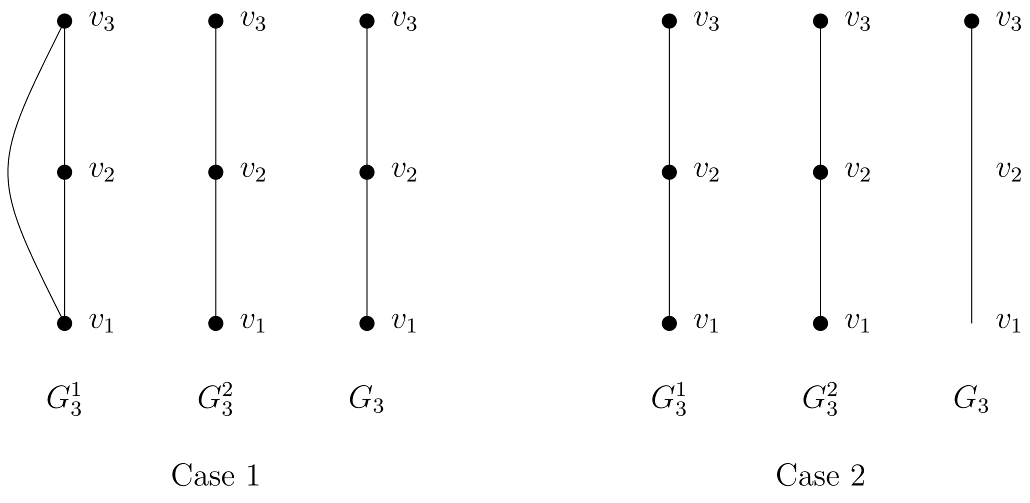


Figure 2.7: 3-path example for G_3

To illustrate our procedure for counting subgraphs in $Geo([0,1], d_{tor}, k, r, 1)$, we compute $Pr_{P_3,2}$ in Example 2.6.16.

Example 2.6.16 *We have three vertices v_1, v_2, v_3 located at $x = (x_1, x_2), y = (y_1, y_2)$ and $z = (z_1, z_2)$. In Figure 2.7 we give 2 of the 3 possible projections whose intersection induce P_3 (the third possibility is $G_3^1 \cong P_3$ and $G_2^2 \cong K_3$ which is just the reverse of Case 1).*

In Case 1, we have a K_3 and a P_3 which intersect to give $G_3 \cong P_3$. The main difficulty in computing $Pr_{P_3,2}$ is determining all the possible vertex labelings of the projections which will lead to an intersection isomorphic to P_3. For a fixed labeling of vertices ϕ, from Theorem 2.6.6 we know that $Pr_{K_3,\phi} = Pr_{P_3,\phi} = \frac{3r^2}{3!} = \frac{r^2}{2}$. It is easy to see that in this case, any ordering of G_3^1 and of G_3^2 leads to an intersection isomorphic to P_3. This brings the probability of G_3^1 and G_3^2 giving an intersection of P_3 in Case 1 to $3!\frac{r^2}{2}3!\frac{r^2}{2} = 9r^4$. We additionally must account for the fact that in Case 1 we could also have $G_3^1 = P_3$ and $G_3^2 = K_3$. Therefore, the total probability of Case 1 is $18r^4$

In Case 2, we have that both projections are P_3's. For G_3^1 we have the freedom to select any labeling of the vertices; for G_3^2 we do not. Consider the labeling in the

Figure 2.7 above. In G_3^2 the only restriction is that the vertex v_2 must be the vertex of degree 2 in P_3. This gives two possible orderings of the vertices in G_3^2 (v_1 and v_3 can be swapped). Overall, the total contribution of Case 2 to the probability that $G_3 \cong P_3$ is $3!\frac{r^2}{2}2\frac{r^2}{2} = 3r^4$.

Overall, $Pr_{P_3,2} = 21r^4$.

This example highlights the process of counting subgraphs in $Geo([0,1]^d, d_\infty, n, r, 1)$. We break down each set of projections such that $\cap_{i=1}^t G_k^i \cong H$ into separate cases where each case consists of the isomorphism classes of the projection which can intersect to give an isomorphic copy of H. For each case, we must account for all possible ordering of the vertices in each projection and all possible orderings of the projections. To compute $Pr_{H,t}$, we simply sum over all these ordering. We use this procedure to count the number of cliques, 3-paths and 4-cycles in $Geo([0,1]^t, d_\infty, n, r, p)$. We begin by counting the number of cliques.

Theorem 2.6.17 *Consider $G_n = Geo([0,1]^t, d_\infty, n, r, p)$ with $r \leq \frac{1}{m}$ and let $X_n = ind(K_m, G_n)$. Then $E(X_n) = (mr^{m-1})^t p^{\binom{m}{2}} \binom{n}{m}$.*

Proof The only possible set of projections $G_m^1, G_m^2, \ldots, G_m^t$ such that $\cap_{i=1}^t G_m^i \cong K_m$ is if each $G_m^i \cong K_m$. For this special case, any vertex labeling of these projections gives $\cap_{i=1}^t G_m^i \cong K_m$. From Theorem 2.6.9 we have that $Pr_{K_m} = mr^{m-1}$ so that $Pr_{K_m,\phi} = \frac{mr^{m-1}}{m!} = \frac{r^{m-1}}{(m-1)!}$. Since each projection is formed independently we have

$$Pr_{K_m,t} = (mr^{m-1})^t$$

Applying Theorem 2.6.2 we obtain $E(X_n) = (mr^{m-1})^t p^{\binom{m}{2}} \binom{n}{m}$. □

Next we compute the number of 3-paths in $Geo([0,1]^t, d_\infty, n, r, 1)$. The proof is merely an extension of the argument in Example 2.6.16 for counting 3-paths in 2D.

Theorem 2.6.18 *Consider $G_n = Geo([0,1]^t, d_\infty, n, r, 1)$ with $r \leq \frac{1}{3}$. Then $Pr_{P_3}(G_n) = 3r^{2t}(4^t - 3^t)$.*

Proof If $\cap_{i=1}^t G_3^i \cong P_3$ then each projection must either be a P_3 or a K_3. Each set of projections consists of $i = 1, 2, \ldots, t$ P_3's and $t - i$ K_3's. Recall from Theorem 2.6.6 that $Pr_{P_3,\phi} = Pr_{K_3,\phi} = \frac{r^2}{2}$. For each K_3, any vertex labeling is allowed. For the first P_3, any labeling of the vertices is allowed but in each of the remaining $i - 1$ P_3's, the vertex of degree two must have the same label as the first projection $G_3^i \cong P_3$. Therefore, each subsequent P_3 can be labeled in 2 ways. The probability for each K_3 as a projection is $3r^2$. The probability for the first P_3 is $3r^2$ but the probability for each subsequent P_3 is $2\frac{r^2}{2} = r^2$. Additionally, we must also account for all the possible locations of the P_3's and K_3's in a set of projections. Since there are i P_3's there are $\binom{t}{i}$ possible ways to place them in a set of projections. Overall, given a set of projections consisting of i P_3's and $t - i$ K_3's, we have a probability that the intersection of such a set induces P_3 is $\binom{t}{i}(3r^2)(r^2)^{i-1}(3r^2)^{t-i} = \binom{t}{i}3^{t+1}r^{2t}3^{-1}$. Summing over all possible number of P_3 projections $i = 1, 2, \ldots, t$ we obtain

$$\begin{aligned} Pr_{P_3,2} &= 3^{t+1}r^{2t}\sum_{i=1}^{t}\binom{t}{i}3^{-i} \\ &= 3^{t+1}r^{2t}((\frac{4}{3})^t - 1) \\ &= 3r^{2t}(4^t - 3^t) \end{aligned}$$

□

Note that when $t = 2$ we get that $P_{P_3,2} = 21r^4$ which corresponds to what we computed in Example 2.6.16.

Using Theorem 2.6.2, we can extend our 3-path count to the general $0 < p \leq 1$ case.

Theorem 2.6.19 *Consider* $G_n = Geo([0,1]^t, d_\infty, n, r, 1)$ *with* $r \leq \frac{1}{3}$. *Let* $X_n = ind(P_3, G_n)$. *Then* $E(X_n) = (3r^{2t}(4^t - 3^t)p^2 + 3p^2(1-p)(3r^2)^t)\binom{n}{3}$.

Proof This immediately from Theorem 2.6.17 with $m = 3$, Theorem 2.6.18 and Lemma 2.6.2.

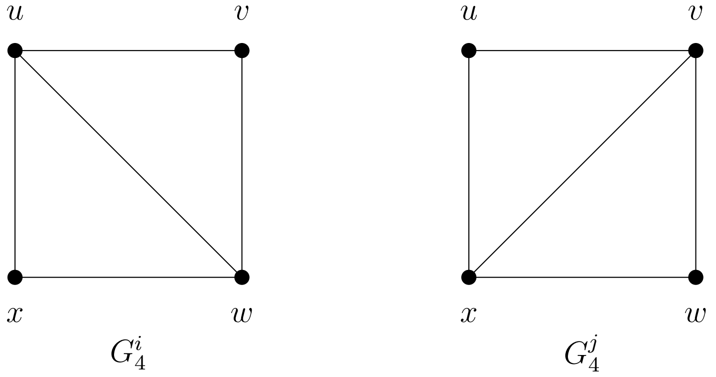
□

For our final result we count the number of 4-cycles in $Geo([0,1]^t, d_\infty, n, r, 1)$ for $t \geq 2$. Recall from the result of Theorem (2.6.7) that the 4-cycle could not form in $Geo([0,1]t, d_{infty}, n, r, 1)$. However, as the next result shows, 4-cycles can appear in higher dimensions.

Theorem 2.6.20 *Let* $G_n = Geo([0,1]^t, d_\infty, n, r, 1)$ *for* $t \geq 2$ *and* $r \leq \frac{1}{4}$ *and let* $X_n = ind(C_4, G_n)$. *Then*

$$E(X_n) = 4^t r^{3t}[(\frac{7}{6})^t - \frac{t}{6} - 1]\binom{n}{4}.$$

Proof For G_4 to be isomorphic to C_4, we require that each projection of G_4 is a g_7. Additionally we require that the vertices in each projection are labeled in a certain way. Consider the labeling of G_4^i and G_4^j below.



G_4^i G_4^j

For this labeling we have $G_4^i \cap G_4^j \cong C_4$. However, if G_4^j had the same labeling as G_4^i then $G_4^i \cap G_4^j \cong g_7$. Suppose that the first projection is a g_7 with the same labeling as G_4^i above. Then any labeling of this projection is permitted. Each remaining projection must either be a g_7 labeled as G_4^i (or with u and w swapped or x and v swapped), labeled as G_4^j (or with u and w swapped or x and v swapped), or a K_4. In the case of a K_4 projection, any labeling is permitted. Recall from Theorem 2.6.7 that $Pr_{g_7} = Pr_{K_4} = 4r^3$. The first g_7 to appear can have any labeling but each subsequent g_7 only has 4 permitted labelings. The probability for these remaining g_7 projections is $4(\frac{4r^3}{4!}) = \frac{2r^3}{3}$. Suppose you have a projection is $i = 2, 3, \ldots, t$ g_7's and $t - i$ K_4's. Accounting for the $\binom{t}{i}$ orderings of the g_7's in the set of projections, the probability that such a set induced a 4-cycle is $\binom{t}{i}(4r^3)(\frac{2r^3}{3})^{i-1}(4r^3)^{t-i} = \binom{t}{i}4^t r^{3t}6^{-i}$. Summing over all possible projections with $i = 2, 3, \ldots, t$ g_7's and $t-i$ K_4's we obtain

$$\begin{aligned} Pr_{C_4,t} &= 4r^3r^{3t}\sum_{i=2}^{t}\binom{t}{i}6^{-i} \\ &= 4^t r^{3t}[(\frac{7}{6})^t - \frac{t}{6} - 1] \end{aligned}$$

By Lemma 2.6.1 $E(X_n) = 4^t r^{3t}[(\frac{7}{6})^t - \frac{t}{6} - 1]\binom{n}{4}$.

□

2.6.6 $Geo([0,1]^t, d_\infty, n, r, p)$ with a Linear Number of Edges

We now return to the case where $G_n = Geo([0,1]^t, d_\infty, n, r, p)$ has a linear number of edges. Let $e_n = ind(K_2, G_n)$. From Theorem 2.6.17 with $m = 2$, we have that $E(e_n) = (2r)^t\binom{n}{2} = (2r)^2 p\frac{n(n-1)}{2}$. Setting $E(e_n) = dn$ for $d \in \mathbb{Z}^+$ and solving for r we obtain,

$$r = (\frac{d}{2^{t-1}pn})^{\frac{1}{t}}(1 + O(\frac{1}{n})).$$

From Theorem 2.6.17 with $m = 3$, we have that the expected number of triangles is $(3r^2)^t p^3 \binom{n}{3}$. The expected number of triangles in G_n with a linear number of edges is,

$$\begin{aligned}
& (3[(\frac{d}{2^{t-1}pn})^{\frac{1}{t}}(1+O(\frac{1}{n}))]^2)^t p^3 \binom{n}{3} \\
= & \frac{3^t d^2 p^3}{4^{t-1}p^2 n^2}\frac{n^3+O(n^2)}{6} \\
= & (\frac{3}{4})^{t-1}\frac{pd^2}{2}n + O(1).
\end{aligned}$$

From Theorem 2.6.18 we have that the expected number of 3-paths is $[3r^{2t}(4^t - 3^t)p^2 + 3p^2(1-p)(3r^2)^t]\binom{n}{3}$. For a linear number of edges it can be shown that the expected number of 3-paths is $[\frac{3d^2(4^t-3^t)}{4^{t-1}} + \frac{3^{t+1}(1-p)d^2}{4^{t-1}}]n + O(1)$.

Finally from Theorem 2.6.20 we have that the expected number of 4-cycles is $4^t r^{3t}[(\frac{7}{6})^{t-1} - 1]\binom{n}{4}$. For a linear number of edges it can be shown that the expected number of 4-cycles is $(\frac{1}{2})^t \frac{d^3}{3}[(\frac{7}{6})^t - \frac{t}{6} - 1]n + O(1)$.

We see that with a linear number of edges in G_n, the number of triangles, 3-paths and 4-cycles all grow linearly with n.

2.7 Spatial Preferred Attachment Model

In this section we count the expected number of triangles, 3-paths, and 4-cycles in the SPA model. We begin with the number of edges that was computed in the introductory paper [7].

Theorem 2.7.1 ([7]) *Let $G_n = SPA(n, m, p, A_1, A_2)$ and let $e_n = ind(K_2, G_n)$. Then*

$$E(e_n) = \begin{cases} (1+o(1))\frac{pA_2}{1-pA_1}n & \mathit{if}\ pA_1 < 1 \\ (1+o(1))nln(n) & \mathit{if}\ pA_1 = 1 \end{cases}$$

Moreover, if $pA_1 < 1$*, then a.a.s. we have that*

$$e_n = (1+o(1))\frac{pA_2}{1-pA_1}n$$

We are only interested in the $0 < pA_1 < 1$ case which gives a linear number of edges in the SPA model. From Theorem 2.7.1, we know that the number of edges is concentrated around its expectation in this case.

Our method for counting the number of triangles, 3-paths and 4-cycles relies on the number of common in-neighbours of vertices v_i and v_j at time n. The number of common in-neighbours of v_i and v_j at time n was explored in [76] and [77]. We review the results of these papers and begin with a result on the in-degree of a vertex v at time t.

Theorem 2.7.2 ([77]) *Let* $\omega = \omega(n)$ *be any function which goes to infinity with* n*. The following statement holds a.a.s. for every vertex* v *for which* $deg_n^-(v) = k = k(n) \geq \omega ln(n)$ *Let* $i = f^{-1}(k)$*, and let* t_k *be*

$$t_k = f^{-1}(\frac{A_2k}{A_1\omega ln(n)}).$$

Then, for all values of t *such that* $t_k \leq t \leq n$*,*

$$deg^-(v,t) = (1+o(1))\frac{A_2}{A_1}\left(\frac{t}{i}\right)^{pA_1} = (1+o(1))k\left(\frac{t}{n}\right)^{pA_1}$$

This theorem states that once a vertex accumulates $\omega ln(n)$ edges, its in-degree is known for the remainder of the process. Recall that the influence region of v_i at time t has area $A(v_i,t) = \frac{A_1deg_n^-(v_i)+A_2}{t}$. So it follows that the influence region of these vertices are known for the remainder of the process. Unfortunately, it is not possible to predict the in-degree of each individual vertex from birth due to randomness which occurs near birth: whether or not a new vertex receives new edges close to its birth greatly influences its future in-degree.

In [77] the authors use the result of Theorem 2.7.2 to compute the number of common neighbours between vertices of at least final degree $\omega ln(n)$. In [76], the authors give a similar result in terms of a modified SPA model. Counting triangles, 3-paths, and 4-cycles in this modified SPA model will be much easier than in the SPA model. We argue that these two models behave in a similar way so that working with the modified SPA model is justified.

Procedure 2.7.3 (Modified SPA Model $SPA^*(n, m, p, A_1, A_2)$) *We define the modified SPA model to be identical to the SPA model except that we define the influence region of v_i at time t for $i \leq t \leq n$ to be $A(v_i, t) = \frac{A_2(\frac{t}{i})^{pA_1} + A_2}{t}$.*

Observe that the modified SPA model is designed so that implicitly, each vertex has in-degree as specified in Theorem 2.7.2 (with the $(1+o(1))$ factor removed). Thus if each vertex in the SPA model had in-degree as specified in Theorem 2.7.2 then the modified SPA model and SPA model would coincide. We provide the following argument to show that the two models are asymptotically the same. Consider the graph $SPA(n, 2, p, A_1, A_2)$ with vertices $v_1, v_2, \ldots, v_n$ which are ranked so that $i < j$ implies that $deg^-(v_i, n) > deg^-(v_j, n)$. From Theorem 1.5.5 we know that the SPA model follows a power law degree distribution with coefficient $1 + \frac{1}{pA_1}$. If $N_{k,n}$ is the number of vertices with in-degree k then $\frac{N_{k,n}}{n} \simeq ck^{-1-\frac{1}{pA_1}}$ for some constant c. Therefore we have that $\sum_{i=1}^{k} ci^{-1-\frac{1}{pA_1}} = 1 - cpA_1k^{-\frac{1}{pA_1}}$ is the proportion of vertices with in-degree less than or equal to k. Therefore, $cpA_1k^{-\frac{1}{pA_1}}$ is the proportion of vertices with in-degree more than k. Also we have that $\frac{j}{n}$ is the proportion of vertices with in-degree greater than k. Now consider the vertex v_j which has degree

$k \geq \omega(n)ln(n)$ and let j_k be the birth time of v_j. Then by Theorem 2.7.2, $k = (1+o(1))\frac{A_2}{A_1}\left(\frac{n}{j_k}\right)^{pA_1}$. By the discussion above

$$\begin{aligned} j &\simeq cpA_1 k^{-\frac{1}{pA_1}} n \\ &= \Theta((\left(\frac{n}{j_k}\right)^{pA_1})^{-\frac{1}{pA_1}} n) \\ &= \Theta(j_k) \end{aligned}$$

Therefore the birth time of v_j is on the order of j. Therefore, vertices whose in-degree are at least $\omega(n)ln(n)$ behave the same way in the SPA model as they do in the modified SPA model.

For the remainder of this section we consider the modified SPA model in $2D$. The mathematics involved with the modified SPA model is more tractable than the mathematics involved with the SPA model and we expect that the results for both models are the similar. The simplicity in using the modified SPA model arises because the influence region of a vertex v at time n is deterministic in the modified SPA model and a random variable in the SPA model. However, from the discussion above, many of the vertices in the SPA model have in-degrees that are concentrated around their expectations so that these vertices have influence regions that behave in a similar way as the modified SPA model.

We define the influence radius of a vertex v_i at time n to be the radius of the influence region $A(v_i, n)$. In $2D$ this gives $\pi r(v_i,t)^2 = \frac{A_2(\frac{t}{i})^{pA_1}+A_2}{n}$. Writing this in terms of $r(v_i,t)$ we obtain,

$$r(v_i,t) = \sqrt{\frac{A_2}{\pi}}\sqrt{\frac{(\frac{t}{i})^{pA_1}+1}{t}}$$

$$
\begin{aligned}
&= \sqrt{\frac{A_2}{\pi}} t^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} \sqrt{1+(\frac{t}{i})^{-pA_1}} \\
&= \sqrt{\frac{A_2}{\pi}} t^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} (1+O(t^{-pA_1} i^{pA_1}) \\
&= \sqrt{\frac{A_2}{\pi}} t^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} + O(t^{-(\frac{pA_1+1}{2})} i^{\frac{pA_1}{2}})
\end{aligned}
$$

Let $c(n, v_i, v_j)$ be the number of common in-neighbours of v_i and v_j at time n. In [77] the authors give a relationship between $cn(v_i, v_j, n)$ and the distance between v_i and v_j in the metric space for the SPA model and in [76] the authors give a similar result for the modified SPA model. We give the result from [76] as we will use the modified SPA model.

Theorem 2.7.4 ([76]) *Consider vertices v_i and v_j $(1 \leq i < j \leq n)$ in $SPA^*(n, 2, p, A_1, A_2)$ with metric d_{tor}. Then we have,*

1. *If $d(v_i, v_j) > r(v_i, j+1) + r(v_j, j+1)$, then $cn(v_i, v_j, n) = 0$*

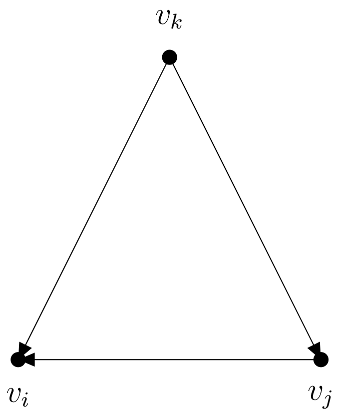
2. *If $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$, then $E(c(v_i, v_j, n)) = (1+o(1))\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}$.*

3. *If $r(v_i, n) - r(v_j, n) < d(v_i, v_j) \leq r(v_i, j+1) + r(v_j, j+1)$, then $E(cn(v_i, v_j, n)) = C(i^{-\frac{(pA_1)^2}{1-pA_1}})(j^{-pA_1})(d^{-\frac{2pA_1}{1-pA_1}})(1+O((\frac{i}{j})^{\frac{pA_1}{2}}))$ where $C = pA_1^{-1} A_2^{\frac{1}{1-pA_1}} \pi^{-(\frac{pA_1}{1-pA_1})}$.*

In [76], the authors state that if $cn(v_i, v_j, n)$ is large enough than it is concentrated around its expectation in Theorem 2.7.4. Given our asymptotic analysis, this is a reasonable assumption for us to make.

In Case 1, v_i and v_j are born far enough apart that their influence regions have an empty intersection at time n. In the modified SPA model, the influence region $A(v_i, n)$ constantly shrinks over time so it is not possible that these influence regions will intersect at some future time so the number of common neighbours will be zero. In Case 2, v_j is born so close to v_i that v_j's entire influence region remains enclosed in v_i's influence region for the entire process. Thus, each in-neighbour of v_j is also an in-neighbour of v_i provided these in-neighbours form an edge to v_i which occurs with probability p. In Case 3, v_j is initially contained in the influence region of v_i, but by the end of the process their influence regions are disjoint.

Let us consider the problem of counting the number of triangles in the modified SPA model. Given the directed nature of the modified SPA model, it is only possible for directed edges to form from younger vertices to older vertices. Therefore only one type of directed triangle can form which is T_1 which we introduce when we counted triangles for the directed copy model.



T_1

We see that v_k is a common in-neighbour of v_i and v_j. This suggests that counting the number of triangles can be achieved by counting common neighbours through the use of Theorem 2.7.4. To count the number of triangles, we consider each pair of vertices v_i and v_j with $i < j$. For each v_k which is a common in-neighbour of v_i and v_j, v_i, v_j, v_k will induce a triangle if there is a directed edge from v_j to v_i. From Theorem 2.7.4 we must consider Case 2 and Case 3 (v_i and v_j have no common neighbours in Case 1). In both of these cases v_j is born inside v_i's influence region so the probability that there is an edge from v_j to v_i is p.

Theorem 2.7.5 *Let $G_n = SPA^*(n, 2, p, A_1, A_1)$ with $pA_1 < 1$ and let $X_n = ind(K_3, G_n)$. Then*

$$E(X_n) = \Theta(n).$$

Proof Let $X_n = ind(K_3, G_n)$ and let X_{ijk} be the indicator variable for the event for $1 \leq i < j < k \leq n$ that v_i, v_j, v_k induce a K_3 in G_n. We compute X_n by considering three disjoint exhaustive events. Let X^1_{ijk} be an indicator variable for the event that v_i, v_j, v_k induce a triangle and $d(v_i, v_j) > r(v_i, j+1) + r(v_j, j+1)$; X^2_{ijk} be an indicator variable for the event that v_i, v_j, v_k induce a triangle and $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ and let X^3_{ijk} be the event that v_i, v_j, v_k induce a triangle and $r(v_i, n) - r(v_j, n) < d(v_i, v_j) \leq r(v_i, j+1) + r(v_j, j+1)$. Observe that $X_{ijk} = X^1_{ijk} + X^2_{ijk} + X^3_{ijk}$. We can compute $X_n = \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} X_{ijk} = X^1_n + X^2_n + X^3_n$ where $X^l_n = \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} X^l_{ijk}$ for $l = 1, 2, 3$.

Case 1 X^1_n: To compute X^1_n we need to compute X^1_{ijk} for each triple $1 \leq i < j < k \leq n$. Observe that $X^1_{ijk} = 0$ for each i, j, k as $d(v_i, v_j) > r(v_i, j+1) + r(v_j, j+1)$ implies that v_i and v_j has no common neighbours from Theorem 2.7.4 and thus v_i, v_j, v_k can not induce a triangle.

Case 2 X^2_n: Define the indicator variable Y_{ijk} for the event that v_k is a common neighbour of v_i and v_j given $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$. Let Z_{ij} be the indicator variable for the event that $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ and $v_i \sim v_j$. Note that we can write $X_{ijk} = Y_{ijk} Z_{ij}$. Overall we can write,

$$\begin{aligned} X^2_n &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} X^1_{ijk} \\ &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} Y_{ijk} Z_{ij} \\ &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} Z_{ij} \sum_{k=j+1}^{n} Y_{ijk} \end{aligned}$$

Note that $\sum_{k=j+1}^{n} Y_{ijk} = cn(v_i, v_j, n)$. From Theorem 2.7.4 we have $cn(v_i, v_i, n) = (1+o(1))\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}$. Plugging this into the above and applying expectation gives,

$$\begin{aligned} E(X_n^2) &= \sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} E(Z_{ij})(1+o(1))\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1} \\ &= (1+o(1))\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} E(Z_{ij})\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}. \end{aligned}$$

Since Z_{ij} is an indicator variable we have that $E(Z_{ij}) = Pr(Z_{ij} = 1)$. To compute $Pr(Z_{ij} = 1)$ we need to use conditional probability since Z_{ij} is an indicator variable for two events. We can write

$$\begin{aligned} Pr(Z_{ij} = 1) &= Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n) \cap v_i \sim v_j) \\ &= Pr(v_i \sim v_j | d(v_i, v_j) \\ &\leq r(v_i, n) - r(v_j, n))Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)) \\ &= pPr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)). \end{aligned}$$

Note that if $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$, then $v_i \sim v_j$ occurs with probability p. For $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$, we need v_j to be within distance $r(v_i, n) - r(v_j, n)$ of v_i. This happens if v_j falls into a circle of radius $r(v_i, n) - r(v_j, n)$ centered at v_i. Such a circle has area $\pi[r(v_i, n) - r(v_j, n)]^2$. Earlier we computed that $r(v_i, n) = \sqrt{\frac{A_2}{\pi}} n^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} + O(n^{-(\frac{pA_1+1}{2})} i^{\frac{pA_1}{2}})$. Note that the error term here comes from the error term associated with the Taylor expansion from the square root. Finally we can compute

$$\begin{aligned}
Pr(Z_{ij}=1) &= p\pi[\sqrt{\frac{A_2}{\pi}}n^{\frac{pA-1}{2}}i^{-\frac{pA_1}{2}}+O(n^{-(\frac{pA_1+1}{2})}i^{\frac{pA_1}{2}}) \\
&- p\sqrt{\frac{A_2}{\pi}}n^{\frac{pA-1}{2}}j^{-\frac{pA_1}{2}}+O(n^{-(\frac{pA_1+1}{2})}j^{\frac{pA_1}{2}})]^2 \\
&= pA_2n^{pA_1-1}[i^{-\frac{pA_1}{2}}-j^{-\frac{pA_1}{2}}+O(n^{-pA_1}j^{\frac{pA_1}{2}})]^2 \\
&= pA_2n^{pA_1-1}i^{-pA_1}[1-(\frac{i}{j})^{\frac{pA_1}{2}}+O(n^{-pA_1}(\frac{j}{i})^{\frac{pA_1}{2}})]^2 \\
&= pA_2n^{pA_1-1}i^{-pA_1}(1-2(\frac{i}{j})^{\frac{pA_1}{2}}+O(n^{-pA_1}(\frac{j}{i})^{\frac{pA_1}{2}})) \\
&= pA_2n^{pA_1-1}i^{-pA_1}-2pA_2n^{pA_1-1}(\frac{1}{ij})^{\frac{pA_1}{2}}+O(n^{-1}i^{-\frac{3pA_1}{2}}j^{\frac{pA_1}{2}})
\end{aligned}$$

We can compute

$$\begin{aligned}
E(X_n^2) &= (1+o(1))\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\frac{p^2A_2}{A_1}(\frac{n}{j})^{pA_1}[A_2n^{pA_1-1}i^{-pA_1}-2A_2n^{pA_1-1}(\frac{1}{ij})^{\frac{pA_1}{2}} \\
&+ O(n^{-1}i^{\frac{-3pA_1}{2}}j^{\frac{pA_1}{2}})] \\
&\simeq \frac{p^2A_2^2}{A_1}n^{2pA_1-1}[\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}(\frac{1}{ij})^{pA_1}-2i^{-\frac{pA_1}{2}}j^{-3\frac{pA_1}{2}}] \\
&+ O(n^{pA_1-1}\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}i^{-\frac{3pA_1}{2}}j^{-\frac{pA_1}{2}})
\end{aligned}$$

We have three different double sums to work out. We can use Lemma 1.3.6 to get an approximate answer by using integrals instead of sums. Working out the error term above gives an error term smaller in magnitude than the error associated with approximating the other two sums so we will drop that term in the next calculations. For the sum $\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}(\frac{1}{ij})^{pA_1}$ we can work out,

$$\begin{aligned}
\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}(\frac{1}{ij})^{pA_1} &= \sum_{i=1}^{n-2} i^{-pA_1}\sum_{j=i+1}^{n-1} j^{-pA_1} \\
&= \sum_{i=1}^{n-2} i^{-pA_1}(\frac{n^{1-pA_1}}{1-pA_1}+O(i^{-pA_1})) \\
&= \frac{n^{1-pA_1}}{1-pA_1}\sum_{i=1}^{n-2}(\frac{1}{i})^{pA_1}+O(\sum_{i=1}^{n-2} i^{-2pA_1}) \\
&= \frac{n^{1-pA_1}}{1-pA_1}(\frac{n^{1-pA_1}}{1-pA_1}+O(1)) \\
&= \frac{n^{2-2pA_1}}{(1-pA_1)^2}+O(n^{1-pA_1})
\end{aligned}$$

Now we consider the sum $\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-\frac{pA_1}{2}} j^{-3\frac{pA_1}{2}}$. In working out $\sum_{j=i+1}^{n-1} j^{-\frac{3pA_1}{2}}$ we need to consider 3 different ranges: $0 < pA_1 < \frac{2}{3}, pA_1 = \frac{2}{3}$ and $\frac{2}{3} < pA_1 < 1$.

If $0 < pA_1 < \frac{2}{3}$ then we have,

$$\begin{aligned}
\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-\frac{pA_1}{2}} j^{-3\frac{pA_1}{2}} &= \sum_{i=1}^{n-2} i^{-\frac{pA_1}{2}}\sum_{j=i+1}^{n-1} j^{-\frac{3pA_1}{2}} \\
&= \frac{n^{2-2pA_1}}{(1-\frac{3pA_1}{2})(1-\frac{pA_1}{2})}+O(n^{1-\frac{3pA_1}{2}})
\end{aligned}$$

If $pA_1 = \frac{2}{3}$ then we have,

$$\begin{aligned}
\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-\frac{1}{3}} j^{-1} &= \sum_{i=1}^{n-2} i^{-\frac{1}{3}}\sum_{j=i+1}^{n-1} j^{-1} \\
&= \sum_{i=1} n-2 i^{-\frac{1}{3}}[ln(n)-ln(i)+O(\frac{1}{i})]
\end{aligned}$$

$$\begin{aligned} &= \sum_{i=1}^{n-2} ln(n)i^{-\frac{1}{3}} - \sum_{i=1}^{n-2} ln(i)i^{-\frac{1}{3}} + O(\sum_{i=1}^{n-2} i^{-\frac{4}{3}}) \\ &= \frac{3ln(n)n^{\frac{2}{3}}}{2} + O(ln(n)) - (\frac{3ln(n)n^{\frac{2}{3}}}{2} + \frac{9}{4}n^{\frac{2}{3}} + O(n^{-\frac{2}{3}})) + O(1) \\ &= \frac{9}{4}n^{\frac{2}{3}} + O(ln(n)) \end{aligned}$$

If $\frac{2}{3} < pA_1 < 1$ then we have,

$$\begin{aligned} \sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-\frac{pA_1}{2}} j^{-3\frac{pA_1}{2}} &= \sum_{i=1}^{n-2} i^{-\frac{pA_1}{2}} \sum_{j=i+1}^{n-1} j^{-\frac{3pA_1}{2}} \\ &= \sum_{i=1}^{n-2} i^{-\frac{pA_1}{2}} [\frac{n^{1-\frac{3pA_1}{2}}}{1-\frac{3pA_1}{2}} - \frac{i^{1-\frac{3pA_1}{2}}}{1-\frac{3pA_1}{2}} + O(i^{\frac{-3pA_1}{2}})] \\ &= \frac{2}{2-3pA_1} \sum_{i=1}^{n-2} n^{1-\frac{3pA_1}{2}} i^{-\frac{pA_1}{2}} - i_O^{1-2pA_1}(i^{-\frac{3pA_1}{2}}) \\ &= \frac{2}{2-3pA_1}[\frac{n^{2-2pA_1}}{1-\frac{pA_1}{2}} - \frac{n^{2-2pA_1}}{2-2pA_1} + O(n^{1-\frac{3pA_1}{2}}) \\ &= \frac{n^{2-2pA_1}}{(2-pA_1)(1-pA_1)} + O(n^{1-\frac{3pA_1}{2}}). \end{aligned}$$

Combining all of these gives,

$$E(X_n^2) = \begin{cases} \frac{p^2A_2^2(8pA_1-4-5p^2A_1^2)}{(1-pA_1)^2(3pA_1-2)(pA_1-2)}n + O(n^{1-pA_1}) & \text{if } 0 < pA_1 < \frac{2}{3} \\ \frac{9p^2A_2^2}{2A_1}n + O(\sqrt{n}) & \text{if } pA_1 = \frac{2}{3} \\ \frac{p^2A_2^2}{(1-pA_1)(2-pA_1)}n + O(n^{1-pA_1}) & \text{if } \frac{2}{3} < pA_1 < 1 \end{cases}$$

Case 3: X_n^3 For the third case, the expected number of common in-neighbours for v_i and v_j depends on the distance between v_i and v_j. We set $d_1 = r(v_i,n) - r(v_j,n)$ and $d_2 = r(v_i,j+1) + r(v_j,j+1)$. Then from Theorem 2.7.4, if $d_1 < d(v_i,v_j) < d_2$ then $E(cn(v_i,v_j,n)|d(v_i,v_j)) = C(i^{-\frac{(pA_1)^2}{1-pA_1}})(j^{-pA_1})(d^{-\frac{2pA_1}{1-pA_1}})(1 + O((\frac{i}{j})^{\frac{pA_1}{2}}))$ where $C = pA_1^{-1}A_2^{\frac{1}{1-pA_1}}\pi^{-(\frac{pA_1}{1-pA_1})}$.

To determine X_n^3 we need to work out an unconditional expectation for $cn(v_i, v_j, n)$ for the case that $d_1 < d(v_i, v_j) < d_2$. We can do so by,

$$\begin{aligned} E(cn(v_i, v_j, n)) &= E(E(cn(v_i, v_j, n)|d(v_i, v_j))) \\ &= \int_{r_1}^{r_2} E(cn(v_i, v_j, n)|d(v_i, v_j))f(x)dx \end{aligned}$$

where $f(x) = 2\pi x$ is the probability density function for $x = d(v_i, v_j)$. Combining the above with the result of Theorem 2.7.4 we get

$$\begin{aligned} E(cn(v_i, v_j, n)) &= \int_{r_1}^{r_2} 2\pi C(i^{-\frac{(pA_1)^2}{1-pA_1}})(j^{-pA_1})(x^{1-\frac{2pA_1}{1-pA_1}})(1 + O((\frac{i}{j})^{\frac{pA_1}{2}}))dx \\ &= 2\pi C(i^{-\frac{(pA_1)^2}{1-pA_1}})(j^{-pA_1})(1 + O((\frac{i}{j})^{\frac{pA_1}{2}})) \int_{r_1}^{r_2} (x^{1-\frac{2pA_1}{1-pA_1}})dx \end{aligned}$$

In working out $E(X_n^3)$ we will achieve the result $\Theta(n)$. Note that it is not possible to get a tighter result as expanding out the $(1 + O(\frac{i}{j}))$ will result in an error term which is also of order n. Note that this is the case since both i and j are going to infinity we have $\frac{i}{j} \to 1$ so that the error term tends to $(1 + O(1))$ in the limit. Therefore it is not possible to work out the constant for n. Going forward we focus only on obtaining a $\Theta(n)$ result for $E(X_n^3)$.

Let us consider $\int_{r_1}^{r_2} (x^{1-\frac{2pA_1}{1-pA_1}})dx$ in the above. If $pA_1 = \frac{1}{2}$ then $1 - \frac{2pA_1}{1-pA_1} = -1$ we need to treat this as a separate case. If $pA_1 \neq \frac{1}{2}$ then

$$\int_{r_1}^{r_2} (x^{1-\frac{2pA_1}{1-pA_1}})dx = \frac{1 - pA_1}{2 - 4pA_1} x^{\frac{2-4pA_1}{1-pA_1}} |_{r_1}^{r_2}.$$

We have that,

$$\begin{aligned} r_1 &= r(v_i, n) - r(v_j, n) \\ &= \sqrt{\frac{A_2}{\pi}} n^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} (1 + O((\frac{i}{j})^{\frac{pA_1}{2}}) \end{aligned}$$

and

$$\begin{aligned} r_2 &= r(v_i, j+1) + r(v_j, j+1) \\ &= \sqrt{\frac{A_2}{\pi}} j^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}} (1 + O((\frac{i}{j})^{\frac{pA_1}{2}})) \end{aligned}$$

Plugging this back into $E(cn(v_i, v_j, n))$ gives

$$\begin{aligned} E(cn(v_i, v_j, n)) &= \Theta((i^{-\frac{(pA_1)^2}{1-pA_1}})(j^{-pA_1})[(j^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}})^{\frac{2-4pA_1}{1-pA_1}} - (n^{\frac{pA_1-1}{2}} i^{-\frac{pA_1}{2}})^{\frac{2-4pA_1}{1-pA_1}}]) \\ &= \Theta(i^{-pA_1} j^{-pA_1} [j^{2pA_1-1} - n^{2pA_1-1}]). \end{aligned}$$

If $pA_1 < \frac{1}{2}$ then $j^{2pA_1-1} > n^{2pA_1-1}$. In this case we have,

$$\begin{aligned} E(X_n^3) &= \Theta(\sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} i^{-pA_1} j^{-pA_1} [j^{2pA_1-1} - n^{2pA_1-1}]) \\ &= \Theta(\sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} i^{-pA_1} j^{pA_1-1}) \\ &= \Theta(n) \end{aligned}$$

If $pA_1 > \frac{1}{2}$ then $n^{2pA_1-1} > j^{2pA_1-1}$. In this case we have,

$$\begin{aligned} E(X_n^3) &= \Theta(\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-pA_1}j^{-pA_1}[j^{2pA_1-1} - n^{2pA_1-1}]) \\ &= \Theta(\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-pA_1}n^{pA_1-1}) \\ &= \Theta(n) \end{aligned}$$

The only case which remains is $pA_1 = \frac{1}{2}$. For this case we have,

$$\begin{aligned} \int_{r_1}^{r_2} x^{-1}dx &= ln(\frac{r_2}{r_1}) \\ &= ln(\frac{\sqrt{\frac{A_2}{\pi}}j^{\frac{-1}{4}}i^{-\frac{1}{4}}(1+O((\frac{i}{j})^{\frac{1}{4}}))}{\sqrt{\frac{A_2}{\pi}}n^{\frac{-1}{4}}i^{-\frac{1}{4}}(1+O((\frac{i}{j})^{\frac{1}{4}})}) \\ &= ln((\frac{n}{j})^{\frac{1}{4}}(1+O((\frac{i}{j})^{\frac{1}{4}})) \\ &= \frac{1}{4}ln(\frac{n}{j}) + O(1) \end{aligned}$$

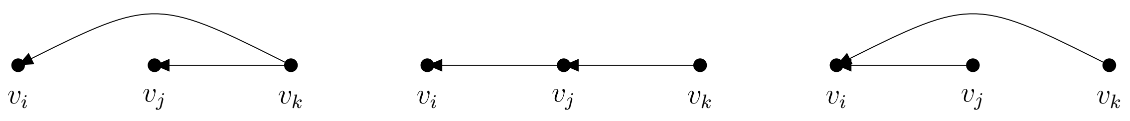
This gives $E(c(n, v_i, v_j, n)) = \Theta(i^{-\frac{1}{2}}j^{-\frac{1}{2}}(1+O((\frac{i}{j})^{\frac{1}{4}}))(ln(\frac{n}{j})+O(1)) = \Theta(i^{-\frac{1}{2}}j^{-\frac{1}{2}}ln(\frac{n}{j}))$

$$\begin{aligned} E(X_n^3) &= \Theta(\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1} i^{-\frac{1}{2}}j^{-\frac{1}{2}}ln(\frac{n}{j})) \\ &= \Theta(n) \end{aligned}$$

The above follows by the use of Lemma 1.3.6 to approximate the sums using integrals. Thus we have for $0 < pA_1 < 1$ we have $E(X_n^3) = \Theta(n)$. Overall we have $E(X_n) = \Theta(n)$.

$\square$

Next we count the number of 3-paths in the modified SPA model. There are three possible ways in which a 3-path can form in the modified SPA model (and the SPA model).



P_3^1 P_3^2 P_3^3

To count the number of 3-paths in the modified SPA model we count the expected number of P_3^1's, P_3^2's and P_3^3's. For P_3^1 note that v_k is a common in-neighbour of v_i and v_j. In fact, counting the number of P_3^1's follows an almost identical calculation as counting the number of triangles in the modified SPA model and gives $E(P_3^1, G_n) = \Theta(n)$. To compute the number of induced copies of P_3^2 and P_3^3, it will be necessary to first compute the number of injective copies and then use Lemma 1.3.2 to determine the number of induced copies.

Theorem 2.7.6 *Let $G_n = SPA^*(n, 2, p, A_1, A_1)$ with $0 < pA_1 < 1$ and let $X_n = ind(P_3, G_n)$. Then,*

$$E(X_n) = \begin{cases} \Theta(n) & \text{if } 0 < p < \frac{1}{2} \\ 2nln(n) + \Theta(n) & \text{if } p = \frac{1}{2} \\ \frac{p-1}{2p^2(2p^2-3p+1)} n^{2p} + \Theta(n) & \text{if } \frac{1}{2} < p < 1 \end{cases}$$

Proof Let $X_n = ind(P_3, G_n)$, $X_n^1 = ind(P_3^1, G_n)$, $X_n^2 = ind(P_3^2, G_n)$ and $X_n^3 = ind(P_3^3, G_n)$. It is easy to see that $X_n = X_n^1 + X_n^2 + X_n^3$.

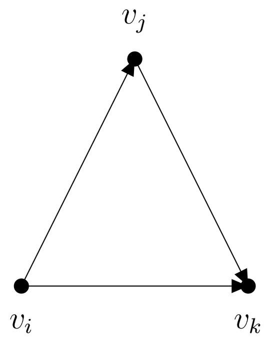
Case 1: X_n^1 The computation of X_n^1 is identical to the computation of $E(ind(K_3, G_n))$ except that a factor of p corresponding to the probability of the event $v_j \sim v_i$ is swapped with a factor of $1-p$ corresponding to the event $v_j \nsim v_i$. Switching this factor does not change any other details of the calculation from Theorem 2.7.5. Since the rest of the calculation proceeds in an identical manner we conclude that $E(X_n^1) = \Theta(n)$.

Case 2: X_n^2 Recall Lemma 1.3.2 which states: let $F_1, F_2, \ldots F_m$ be all the graphs on k vertices and consider any graph G. Then for any F_i we have

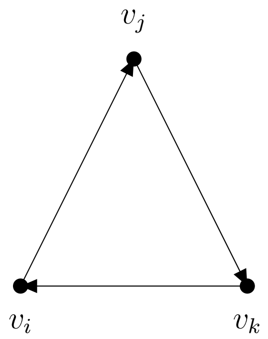
$$inj(F_i, G) = \sum_{j=1}^{m} M_{ij} ind(F_j, G)$$

where $M_{ij} = inj(F_i, F_j)$.

Using this we can write $inj(P_3^2, G_n) = ind(P_3^2, G_n) + ind(T_3^1, G_n) + 3ind(T_3^2, G_n)$ where



T_3^1



T_3^2

From Theorem 2.7.5 we have $E(ind(T_3^1, G_n)) = \Theta(n)$. We also have that $E(ind(T_3^2, G_n)) = 0$ since it is not possible for a directed edge to form from a younger vertex to an older one (the directed edge from v_k to v_i is not possible. Therefore we can conclude that $E(ind(P_3^2, G_n)) = E(inj(P_3^2, G_n)) + \Theta(n)$.

Counting the number of injective copies of P_3^2 is a simpler task than counting the number of induced copies.

Let X_{ijk} be the event that vertices v_i, v_j, v_k for $1 \leq i < j < k \leq n$ form an injective copy of P_3^2. This is equivalent to the event that there is a directed edge from v_j to v_i and from v_k and v_j. In the modified SPA model, edge formations in different time steps are independent processes. Therefore we have that $E(X_{ijk} = 1) = Pr(X_{ijk} = 1) = Pr(v_j \sim v_i)Pr(v_k \sim v_j)$. There is an edge from v_j to v_i with probability p if v_j lands in v_i's area of influence. We have,

$$\begin{aligned} Pr(v_j \sim v_i) &= p(\frac{A_2(\frac{j}{i})^{pA_1} + A_2}{j}) \\ &= pA_2(\frac{(\frac{j}{i})^{pA_1} + 1}{j}) \end{aligned}$$

Similarly we have $Pr(v_k \sim v_j) = pA_2(\frac{(\frac{k}{j})^{pA_1}+1}{k})$. Therefore we have,

$$\begin{aligned} E(inj(P_3^2, G_n)) &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} E(X_{ijk}) \\ &= \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} (pA_2)^2[i^{-pA_1}j^{-1}k^{pA_1-1} + i^{-pA_1}j^{pA_1-1}k^{-1} \\ &+ k^{pA_1-1}j^{-1-pA_1} + j^{-1}k^{-1}] \end{aligned}$$

Approximating each of these sums by integrals using Lemma 1.3.6 we can show that $E(inj(P_3^2, G_n)) = \Theta(n)$. Since $E(ind(P_3, G)) = E(inj(P_3, G_n)) + \Theta(n)$, we can conclude that $E(X_n^2) = \Theta(n)$.

Case 3: X_n^3 The computation of $E(X_n^3)$ proceeds in an identical manner to the computation of $E(X_n^2)$. From Lemma 1.3.2 we have that $inj(P_3^3, G_n) = ind(P_3^3, G_n)+$

$ind(K_3^1, G_n) + ind(K_3^2, G_n)$. Rearranging this gives $E(X_n^3) = E(inj(P_3^3, G_n)) + \Theta(n)$. Let X_{ijk} be the event that vertices v_i, v_j, v_k for an injective copy of P_3^3. This is equivalent to the event that there is a directed edge from v_j to v_i and from v_k and v_i. Therefore we have that $E(X_{ijk} = 1) = Pr(X_{ijk} = 1) = Pr(v_j \sim v_i)Pr(v_k \sim v_i)$. This gives,

$$\begin{aligned} Pr(X_{ijk} = 1) &= pA_2[\frac{(\frac{j}{i})^{pA_1} + 1}{j}]pA_2[\frac{(\frac{k}{i})^{pA_1} + 1}{k}] \\ &= (pA_2)^2(i^{-2pA_1}j^{pA_1-1}k^{pA_1-1} + i^{-pA_1}j^{pA_1-1}k^{-1} + i^{-pA_1}j^{-1}k^{pA_1-1} \\ &+ j^{-1}k^{-1}) \end{aligned}$$

Continuing we have,

$$\begin{aligned} E(inj(P_3^3, G_n)) &= (pA_2)^2 \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{i=1}^{n} (i^{-2pA_1}j^{pA_1-1}k^{pA_1-1} + i^{-pA_1}j^{pA_1-1}k^{-1} \\ &+ i^{-pA_1}j^{-1}k^{pA_1-1} + j^{-1}k^{-1}) \end{aligned}$$

The last 3 of these 4 sums can be worked out to give an $\Theta(n)$ term. To work out the first term $\sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{i=1}^{n} i^{-2pA_1}j^{pA_1-1}k^{pA_1-1}$ we need to consider 3 possibilities for pA_1: $0 < pA_1 < \frac{1}{2}, pA_1 = \frac{1}{2}$ and $\frac{1}{2} < pA_1 < 1$.

If $pA_1 \neq \frac{1}{2}$ then approximating the sum using Lemma 1.3.6 gives

$$\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=j+1}^{n} i^{-2pA_1}j^{pA_1-1}k^{pA_1-1} \quad = \quad \frac{p-1}{2p^2(2p^2-3p+1)}n^{2p} + \frac{2p^2}{2p^2-3p+1}n + O(n^p)$$

If $0 < pA_1 < \frac{1}{2}$ then the n term is dominant and

$$\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{i=1}^{n} i^{-2pA_1}j^{pA_1-1}k^{pA_1-1} = \frac{2p^2}{2p^2-3p+1}n + O(n^{2p}).$$

If $\frac{1}{2} < pA_1 < 1$ then the n^{2p} term is dominant and

$$\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{i=1}^{n} i^{-2pA_1}j^{pA_1-1}k^{pA_1-1} = \frac{p-1}{2p^2(2p^2-3p+1)}n^{2p} + O(n).$$

If $pA_1 = \frac{1}{2}$, then by approximating the sums with integrals by using Lemma 1.3.6

$$\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{i=1}^{n} i^{-1}j^{-\frac{1}{2}}k^{-\frac{1}{2}} = 2nln(n) + O(n).$$

Combining all of this gives

$$E(inj(P_3^3, G_n)) = \begin{cases} \Theta(n) & \text{if } 0 < p < \frac{1}{2} \\ 2nln(n) + \Theta(n) & \text{if } p = \frac{1}{2} \\ \frac{p-1}{2p^2(2p^2-3p+1)}n^{2p} + \Theta(n) & \text{if } \frac{1}{2} < p < 1 \end{cases}$$

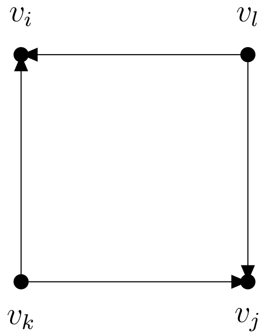
It follows that $E(X_n^3) = E(inj(P_3^3, G_n)) + \Theta(n)$ that $E(X_n^3)$ has the same expression as $E(inj(P_3^3, G_n))$ above.

Finally, computing $E(X_n) = E(X_n^1) + E(X_n^2) + E(X_n^3)$ gives the result of the Theorem.

□

To conclude our analysis of the SPA model, we include a lower bound for the number of 4-cycles. To give a complete count for the number of 4 cycles, we need to consider each of the 3 possible ways a 4-cycle can appear in a directed graph as outlined in the proof of Theorem 2.4.8. Given the geometry and constantly shrinking influence regions of the modified SPA model, counting 4 cycles in each of these cases is difficult. Luckily, the techniques we have used in this section to count triangles and 3-paths can be used to give a lower bound for one of these 4-cycles and hence the number of 4-cycles in total.

The directed 4-cycle we will count to give our lower bound is C_4^1 from the proof of Theorem 2.4.8.



C_4^1

In the C_4^1 pictured above we have that v_k and v_l are common in-neighbours of v_i and v_j. Now to count the number of C_4^1's we can use a similar common in-neighbour argument using Theorem 2.7.4 that we used to count the number of triangles and 3-paths in the modified SPA model. For our lower bound calculation we only consider case ii from Theorem 2.7.4 where $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ and $E(cn(v_i, v_j, n)) = (1 + o(1))\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}$. It is noted in [76], the if the number of common of in-neighbours is large enough, then the number of common in-neighbours is concentrated around its expectation . For our asymptotic analysis, we can assume that this is the case so in the following proof we will simply write $cn(v_i, v_j, n) = \frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}$. Our method for determining a lower bound for the number of C_4^1's will be to sum over all 4-sets of vertices v_i, v_j, v_k, v_l with $1 \leq i < j < k < l \leq n$ with $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ no edge between v_i and v_j and each pair of common neighbours v_k and v_l of v_i and v_j.

Theorem 2.7.7 *Let $G_n = SPA^*(n, 2, p, A_1, A_2)$ with $0 < pA_1 < 1$ and let $X_n = ind(C_4, G_n)$. Then,*

$$E(X_n) = \begin{cases} \Omega(n) & \text{if } 0 < pA_1 \geq \frac{2}{3} \\ \Omega(n^{3p-1}) & \text{if } \frac{2}{3} < pA_1 < 1 \end{cases}$$

Proof Since a lower bound for the number of C_4^1's is also a lower bound for the number of C_4's, we count $X_n = ind(C_4^1, G_n)$. Let X_{ijkl} be an indicator variable for the events that v_i, v_j, v_k, v_l with $1 \leq i < j < k < l \leq n$ induce a C_4^1. Let Y_{kl} be an indicator variable for the event that v_k and v_l are common in-neighbours of v_i and v_j. Let $Z_{i,j}$ be an indicator variable for the event that $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ and $v_i \nsim v_j$. Note that $X_{ijkl} \geq Y_{kl} Z_{ij}$ as there are additional possibilities that v_i, v_j, v_k, v_l can induce a C_4^1 (see case iii) of Theorem 2.7.4.

$$\begin{aligned} X_n &= \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} X_{ijkl} \\ &\geq \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} Y_{kl} Z_{ij} \\ &= \sum_{i=1}^{n-3} \sum_{j=i+1}^{n-2} Z_{ij} \sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} Y_{kl} \end{aligned}$$

Note that $\sum_{k=j+1}^{n-1} \sum_{l=k+1}^{n} Y_{kl} = \binom{cn(v_i, v_j, n)}{2}$, the number of pairs of common in-neighbours of v_i and v_j. When $Z_{ij} = 1$ we have $d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)$ so from

Theorem 2.7.4 that

$$\begin{aligned}
\sum_{k=j+1}^{n-1}\sum_{l=k+1}^{n} Y_{kl} &= \binom{cn(v_i, v_j, n)}{2} \\
&= \binom{(1+o(1))\frac{pA_2}{A_1}(\frac{n}{j})^{pA_1}}{2} \\
&= (1+o(1))\frac{1}{2}(\frac{pA_2}{A_1})^2(\frac{n}{j})^{2pA_1}
\end{aligned}$$

Plugging this into the above and taking expectation gives

$$\begin{aligned}
E(X_n) &\geq \sum_{i=1}^{n-3}\sum_{j=i+1}^{n-2} E(Z_{ij})(1+o(1))\frac{1}{2}(\frac{pA_2}{A_1})^2(\frac{n}{j})^{2pA_1} \\
&= (1+o(1))\frac{1}{2}(\frac{pA_2}{A_1})^2\sum_{i=1}^{n-3}\sum_{j=i+1}^{n-2} E(Z_{ij})(\frac{n}{j})^{2pA_1}.
\end{aligned}$$

Since Z_{ij} is an indicator variable we have that $E(Z_{ij} = 1) = Pr(Z_{ij} = 1)$. We can write

$$\begin{aligned}
Pr(Z_{ij} = 1) &= Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n) \cap v_i \nsim v_j) \\
&= Pr(v_i \nsim v_j | d(v_i, v_j) \leq r(v_i, n) - r(v_j, n))Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)) \\
&= (1-p)Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)).
\end{aligned}$$

Recall from the proof of Theorem 2.7.5 that $Pr(d(v_i, v_j) \leq r(v_i, n) - r(v_j, n)) = A_2 n^{pA_1-1} i^{-pA_1} - 2A_2 n^{pA_1-1}(\frac{1}{ij})^{\frac{pA_1}{2}} + O(n^{-1} i^{-\frac{3pA_1}{2}} j^{\frac{pA_1}{2}})$. Plugging all this back into our expression we obtain

$$\begin{aligned} E(X_n) &\geq \sum_{i=1}^{n-3}\sum_{j=i+1}^{n-2}(1-p)(\frac{pA_2}{A_1})^2(A_2n^{pA_1-1}i^{-pA_1}-2A_2n^{pA_1-1}(\frac{1}{ij})^{\frac{pA_1}{2}}+O(n^{-1}i^{-\frac{3pA_1}{2}}j^{\frac{pA_1}{2}})) \\ &\quad (\frac{n}{j})^{2pA_1} \\ &= \Omega(n^{3pA_1-1}\sum_{i=1}^{n-3}\sum_{j=i+1}^{n-2} i^{-pA_1}j^{-2pA_1}-i^{-\frac{pA_1}{2}}j^{-\frac{5pA_1}{2}}) \\ &= \Omega(n^{3pA_1-1}\sum_{i=1}^{n-3}\sum_{j=i+1}^{n-2} i^{-pA_1}j^{-2pA_1}) \end{aligned}$$

We approximate the sum above using Lemma 1.3.6. In doing so, we have two ranges for the solution. When $p \neq \frac{1}{2}$ we have,

$$E(X_n) = \Omega(n + n^p + n^{3p-1})$$

When $pA_1 \leq \frac{2}{3}$ then the dominate term is n and when $pA_1 > \frac{2}{3}$ then n^{3pA_1-1} is the dominate term. When $pA_1 = \frac{1}{2}$ then we can write,

$$E(X_n) = \Omega(n + ln(n) + \sqrt{n}),$$

and n is the dominate term. Overall we have

$$E(X_n) = \begin{cases} \Omega(n) & \text{if } 0 < pA_1 \geq \frac{2}{3} \\ \Omega(n^{3p-1}) & \text{if } \frac{2}{3} < pA_1 < 1 \end{cases}$$

Chapter 3

Model Validation for Social Networks using Alternating Decision Trees

In this chapter, we perform a model-selection experiment with the purpose of determining which of our selected models is the most appropriate for data from Facebook. Our approach is largely adopted from the work of Middendorf *et al.* [91] for validating models for protein-protein interaction networks via supervised classification algorithms from machine learning. In our work, we modify the approach of Middendorf *et al.* and extend it to online social networks. Our work is different in 4 ways: (*i*) we use a different classification algorithm, (*ii*) we extend their approach to much larger and denser graphs, (iii) we complement the use of graphlets by including features based on global properties of the networks and (iv) we use a different set of models. In a classification algorithm, a set of vector based *training data* from different classes is learned by the classifier. A new instance of vector based data, called *test data*, is evaluated by the classifier, and the class that the test data most likely belongs to is selected. The goal of our experiment is to find which of our six selected models is the most capable of replicating a real online social network. Finding the network model most appropriate for online social networks provides insight into the growth mechanism that is the most dominant in determining the structure of online social networks. We consider models based on preferential attachment, copying and embedding the nodes in a geometric space. A critical question we wish to answer is: which of these mechanisms are the main drivers in the formation of the graphlet structure in online social networks? It is easy to see how all three of these mechanisms have some role in the formation of a social network.

Preferential Attachment: Popular individuals are more likely to exert a greater influence in the network and thus attract more friends.

Copying: It is very common that social relationships are established by being introduced to new individuals by someone who knows those individuals.

Geometric: Individuals who live closer are more likely to meet each other and become friends than individuals who live further away. The geometric space need not model physical distance between two people; it could also model a *topic space* where individuals with similar interests are closer to one another in the space than individuals with diverging interests.

In Section 3.2, we argue that our experiment clearly demonstrates that preferential attachment is the most important of the three in determining graphlet structure in online social networks.

Additionally in Section 3.2, we test the accuracy of our classifier on test data taken from our selected models. To test our hypothesis that graphlet counts characterize the structure of a network, we developed three versions of our model-selection method: one based only on non-graphlet features, one based only on graphlet counts, and a third based on all features together. We found that the classifier built on the graphlet count features alone were just as accurate as the full feature set on test data taken from our training models. We conclude from this that graphlet counts alone are sufficient in characterizing our selected models. This conclusion is also supported by our results in Chapter 2.

In Section 3.1 we provide a complete description of our experimental procedure including the models, the testing data, the features selected and the classification algorithm used. In Section 3.2 we give a detailed analysis of our results.

3.1 Experimental Procedure and Implementation

Our model selection method follows three steps. First, we obtain the training data by generating 1000 graphs according to each of our six selected models: the Preferential Attachment Model (PA), the Copy Model (COPY), the Random Geometric Model (GEO2D and GEO3D), and the Spatial Preferred Attachment Model (SPA2D and SPA3D). We briefly remind the reader of the details of the models in Section 3.1.1 below. The parameters of the models are randomly sampled from a range such that the graphs generated are similar in size and density to the test data. Specifically, we generate the models so that they have the same number of vertices and are within 5% of the number of edges as the test graph. To sample the parameters, we rearrange the expected number of edge calculations we performed in Chapter 2. The restriction of the sample range of the parameters is necessitated by the fact that the graphlet counts depend heavily on the size and density of a graph, even for graphs generated by the same model. For this reason, it is necessary to generate a new training set for each test graph. This greatly increases the amount of time it takes to test different Facebook graphs because the experiment needs to be replicated from scratch for each new test graph. It is currently unclear whether there exists an adequate normalization method, which would make it possible to compare graphlet counts for graphs of different densities.

Next, we use the training data to build a multi-class alternating decision tree (ADT). The details of the construction of the ADT are given in Section 3.1.3 below. We represent the graphs using features that capture both the local structure of the graph, through the graphlets, and the global structure. A description of the features is given in Section 3.1.2 below.

Finally, we compute the feature vectors corresponding to on-line social network data; in this case snapshots of Facebook. Evaluating this feature vector through the classifier, gives a score for each model corresponding to how well the model fits the test data. Our experimental procedure is repeated for four different Facebook networks taking from the following American universities : Princeton, American University, MIT and Brown. We obtained this data from the data sets in [96]. We discuss the results of these four experiments in Section 3.2.

3.1.1 Models

We have implemented six different graph models. Our choice of models was motivated by a desire to test a wide range of models commonly proposed for social networks, based on a number of different attachment mechanisms. Special attention was given to spatial models, a class of models that is gaining support because of their ability to model node attributes through spatial representation. Wherever more than one variation of a model has been proposed in the literature, we have opted for the more well known version.

All model-generation algorithms are written in Python using the graph-tool module [2]. Our training set includes only undirected graphs without multiple edges. Some of the models allow for multiple edges; if this occurs, we remove the multiple copies. For all models under consideration, this is known to affect only a tiny amount of edges. The SPA model and copy model are formulated to generate directed graphs; we ignore the direction of the edges after generation.

We have already described the models we use in this experiment in Section 1.5. We now give the exact details for each model as it was used in our experiment.

Preferential Attachment Model (PA).

We use the generalized PA model $PA(n, d, \alpha)$ from [59]. In our theoretical analysis in Chapter 2, this model started with an initial graph G_0 consisting of a single vertex with d loops. In this experiment we begin with $G_0 = ER(100, p)$, where p is chosen so that G_0 has the same edge density as the Facebook graph being tested. To generate a graph with similar density as the test graph used, we sample d from a range which guarantees that the PA graph generated has a number of edges within 5% of the test graph. Since $PA(n, d, \alpha)$ always generates graphs with dn edges, this results in a narrow bound form which we sample d. The parameter α is sampled uniformly from the range $[0, \lfloor \frac{d}{2} \rfloor]$. This range for α was selected so as to not obscure the preferential attachment mechanism.

Copy Model (COPY).

We use a directed copy model with extra edge addition studied in [4, 32]. This copy model is the directed version of $Copy(n, p, d, G_{n_0})$ introduced in Section 1.5.3. The copy model has two parameters: the copying probability p and the number of extra edges d. We begin with an initial graph G_0 which is a directed version of the Erdős-Rényi Random Graph generated on 100 vertices so that G_0 has the same edge density as the selected test graph. The copying probability is sampled uniformly from $[0, 1]$ and d is selected so that the expected average degree of each vertex is equal to the average degree of the test graph. If this results in a graph with more than 5% of the number of edges as the test graph, then it is discarded and a new graph is generated. The method for selecting the parameters for the Copy model differs from the procedure used for the other models. This is necessary as the expected number of edges in the Copy model does not appear to be concentrated around its expectation, resulting in an unreliable use of the expected edge calculation in determining the parameters for this model.

Random Geometric Model (GEO2D/GEO3D).

We use the two RGG's $Geo([0, 1]^2, d_{tor}, n, r, p)$ and $Geo([0, 1]^3, d_{tor}, n, r, p)$ as potential models for our Facebook graphs. From our previous description, this model has two parameters $p \in [0, 1]$ and $r \in (0, 1)$. The parameter p is sampled uniformly at random from $[0, 1]$ and r is computed so that the number of edges is the graphs generated is within 5% of the number of edges in the test graph. We compute r by rearranging the expected number of edges calculation and substituting in the known values for p, n and the desired number of edges. We found that to obtain the desired density, it was necessary to take a small value for the threshold r. Doing so results in a large graph distance between vertices that are far apart in the metric space, and thus resulting in a large average path length in the GEO model. This places the GEO model at an immediate disadvantage, as it is well known that social networks, as well as the other models, have a small average path length. To remedy this handicap, we select a small number of pairs of vertices after the graph has been generated and place an edge between each pair. In our experiments we select $\lfloor ln(n) \rfloor$ pairs of vertices where n is the number of vertices in the test graph. This addition of a small number of random edges significantly lowers the average path length without significantly affecting the other features such as graphlet counts and the degree distribution.

Spatial Preferential Attachment Model (SPA2D/SPA3D).

For the SPA model we consider two different versions that are placed in the same metric spaces as in the GEO model. Recall that the SPA model has three parameters $p, A_1 \in (0, 1]$ and $A_2 \geq 0$. The parameters p and A_1 are selected uniformly at random from their respective ranges and A_2 is computed by rearranging the expected edge calculation for the SPA model.

3.1.2 Features

We represent our graphs by 17 features in a vector representation. These features include information about the global properties of the graphs, specifically the degree distribution, the assortativity coefficient and the average path length between vertices. In addition, we capture the local structure through the raw graphlet counts for the connected subgraphs of size 3 and size 4. Below is a description of each of these features.

Degree Distribution Percentiles. The degree distribution is a favourite property studied for most "real world" networks. A distribution with a power law tail is a distinguishing property of many such networks, including the friendship network of Facebook [96]. The most logical feature to use here would be the coefficient of the power law degree distribution. Unfortunately, not all of our selected models generate graphs with a power law degree distribution (*e.g.* random geometric models). Also, even if a model does generate a power law degree distribution, it can be difficult to determine its power law coefficient. Instead, to measure the spread of the degree distribution, we consider the percentiles of the distribution formed by breaking it evenly into 8 different pieces. This corresponds to taking the 12.5th, 25th, 37.5th, 50th, 62.5th, 75th and 87.5th percentiles. We call these features $deg_1, deg_2, deg_3, deg_4, deg_5, deg_6$, and deg_7 respectively. These percentiles are determined through a simple code using the graph-tool module.

Assortativity Coefficient.

Recall that the assortativity coefficient $r \in [-1, 1]$ is a measurement of how well vertices of similar degree link to one another in the network. The assortativity coefficient is computed using the graph-tool module.

Average Path Length. The small world property, implying a small average distance between nodes, is another distinguishing aspect of social networks. It is shown in [23] that online social networks have a small average path length. Here we compute the average path length between nodes by selecting 100 random pairs of nodes and calculating the length of the shortest path between each pair using a breadth-first-search that is implemented in graph-tool.

Graphlets.

To characterize local structure, we include as features all the counts of connected subgraphs of size 3 (two non-isomorphic graphs) and 4 (six non-isomorphic graphs.)

These graphlets are depicted in Figure 1.1. Unfortunately, no algorithm is known which computes the full counts for these subgraphs efficiently (though there are algorithms to count triangles quickly [83, 120]). As a compromise, we use the sampling algorithm of Wernicke [124] to sample the number of these graphlets. The advantage of Wernicke's algorithm is that it can be used to give an unbiased sample of a specified portion of the subgraphs.

As input, Wernicke's algorithm takes in a labeled graph and an integer k, the size of the connected subgraphs to be counted. The algorithm generates a tree of depth k by looping through each vertex and performing a depth first search on the k-neighbourhoods of each vertex. Recall that the k-neighbourhood of a vertex is the set of all vertices within graph distance k of the vertex. When the algorithm terminates, a tree of depth k has been formed where the leaves of the tree correspond to all the size k connected subgraphs of G.

Building the entire tree is extremely time consuming for the size of graphs we are considering. Wernicke's algorithm allows for the unbiased sampling of the size k subgraphs by probabilistically skipping steps in the algorithm. Experiments performed by Wernicke in [124] show that the sampling algorithm samples the correct proportion. To supplement Wernicke's analysis we investigate the effectiveness of the sampling algorithm by testing it on a graph with 3000 vertices and 70270 edges (see Table 3.1.2). You can see that there is good agreement between the real counts and the estimates from Wernicke's algorithm. In light of this almost perfect agreement between sampling and exhaustive counts, we deem that a sampling rate of 0.01% would be sufficient for our purposes.

For our experiments, we sample 1% of the size 3 graphlets and 0.01% of the size 4 graphlets. The counting of the graphlets is the most time consuming step of our model-selection procedure. For the size and density of graphs we are considering, it was not feasible to include subgraphs of size greater than 4. In [91], the authors

%	g1	g2	g3	g4	g5	g6	g7	g8
100	2323538	320097	18389736	65090655	22256380	3115254	4317267	434608
10	232335	32075	1837970	6508583	2227640	310958	431961	43176
1	23142	3243	184115	650031	222899	30905	43062	4378
0.1	2368	343	18341	65156	22381	3143	4281	453
0.01	224	33	1804	6524	2163	315	422	49

Table 3.1: Performance of Wernicke's Algorithm on a graph with 3000 vertices and 70270 edges.

consider subgraphs up to size 7, but this is only possible because the graphs are much smaller and sparser than those considered here. Inclusion of graphlets of larger size will only be possible for graphs of the size and density considered here if new methods are developed to compute or estimate graphlet counts which show a dramatic increase in efficiency. However, our results show that the graphlets of size 3 and 4 are highly effective in separating the models. Based on our results, we do not expect that the inclusion of higher-order graphlets would lead to a significant improvement in the accuracy of our model-selection method.

3.1.3 Classification Algorithm

To classify our data, we use the multi-class alternating decision tree (ADT) algorithm LADTree of Holmes *et al.* [67]. ADTs are a class of boosted decision trees that were introduced by Freund and Mason in [68]. Boosting [69] is a well established classification technique which combines so called "weak classifiers" to form a single powerful classifier. In successive steps, called *boosting steps*, weighted combinations of the weak classifiers are applied to the training data and the weights are adjusted in each step to improve the classification.

The first ADTs were built using the AdaBoost boosting algorithm [69]. The ADT used here, LADTree, is built on the lesser known LogitBoost boosting algorithm of Friedman, Hastie and Tibshirani [70]. Friedman *et al.* show in their work that both boosting algorithms fit an additive logistic regression model. They argue that LogitBoost is the more appropriate algorithm because it fits the regression model using the more typical maximum likelihood minimization criteria, whereas AdaBoost uses an exponential minimization criteria.

In Figure 3.1, we show a partial LADTree that was constructed during our experiment. An ADT has two types of nodes, *decision nodes* (rectangles in Figure 3.1) and *prediction nodes* (ellipses in Figure 3.1). Decision nodes contain a Boolean predicate that corresponds to a threshold on one of the features in the feature vectors for the training data. The prediction nodes contain real-valued scores, one for each of the classes in the training set. In our case, we have six different classes or models so each prediction node contains six scores.

The LADTree begins with a prediction node that has a score of zero for each of the models. In each boosting iteration, a decision node is added to the tree along with two prediction nodes as its children in the tree. The new decision node can be added as a child to any existing prediction node in the tree. The placement of the decision node and its Boolean predicate is the one that gives the best separation of the training data. The exact criterion for this is provided by the LogitBoost algorithm [70].

Once the LADTree has been formed, new instances, typically called the test data, can be classified by the tree. For us, the test data is the feature vector for the Facebook graph we wish to classify. The feature vector for the Facebook graph will determine its flow through the tree. The test instance travels through all possible paths it can reach in the tree, resulting in a classification score, which is the sum of all prediction nodes, reached along the way. This results in six different scores, one for each of the six different models F_j $j = 1, 2, 3, 4, 5, 6$. A positive score indicates a good fit and a negative score indicates a bad fit. The model that obtains the highest score is deemed to be the model that best describes the test data. The absolute values of

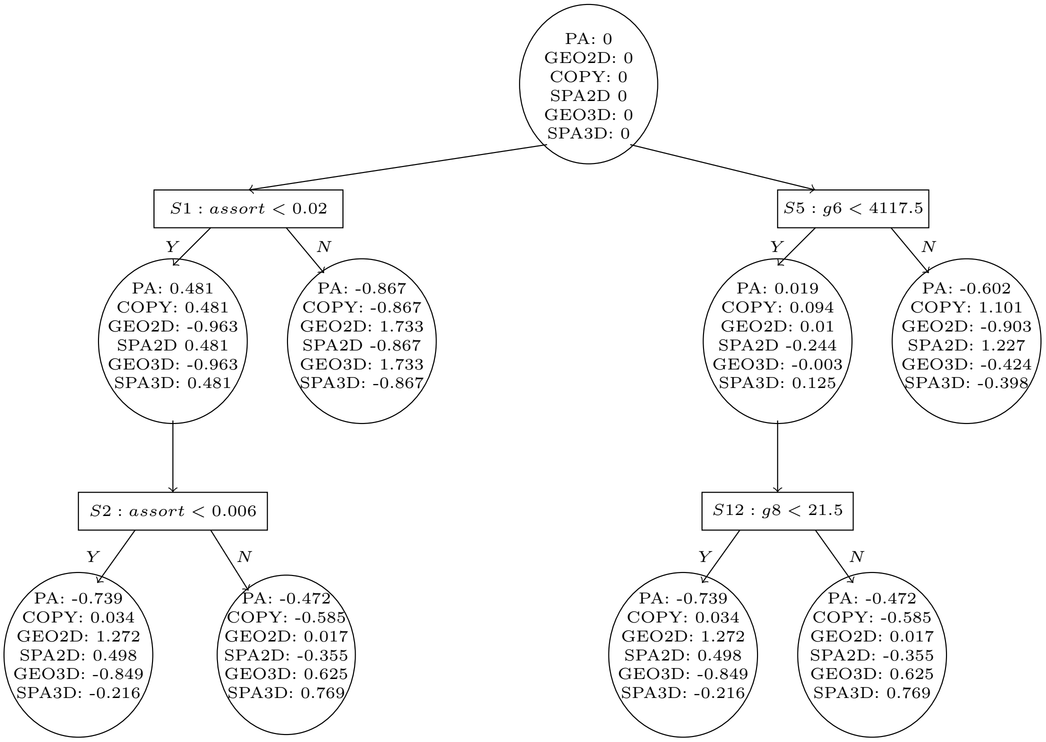


Figure 3.1: Partial LADTree using the full feature vector with 200 boosting iterations.

the scores provide the level of confidence in the prediction. Thus, a large positive F_j indicates that model j is a very good model for the test instance and a large negative F_j indicates that model j is a very bad model for the test instance. The scores F_j can be readily interpreted as class probabilities p_j by the equation

$$p_j = \frac{e^{F_j}}{\sum_{j=1}^{6} e^{F_j}}$$

which results by inverting the additive logistic model that is fitted by the LADTree algorithm [67].

The advantage of using ADTs is that they require no specific assumption about the geometry of the input space for the features. Thus we are free to incorporate any range of features such as degree distribution percentiles, average path length and subgraph counts without considering any potential dependence amongst them. The importance of each feature is based on how well it separates the 6 different models. We use the Weka software package for Java [1] to build all the LADTrees used in our experiments.

3.2 Discussion of Experiment Results

We tested our approach on four different social network graphs taken from Mason Porter's Facebook100 data set [3]. Each graph in the data set corresponds to users at different universities. For our test data we take: Princeton University which has 6596 vertices and 293329 edges, American University which has 6386 vertices and 217661 edges, MIT which has 6440 vertices and 251252 edges, and Brown University which has 8600 vertices and 384525 edges. Note that if we were to express the number of edges in these graphs as dn then the approximate values for d are: Princeton - $d = 44$, American - $d = 34$, MIT - $d = 39$ and Brown - $d = 45$.

For each of these test graphs, our experiment procedure is as follows. First, we generate a training set of 6000 graphs which are the of same size as the test graph, and have edge density which differs by at most 5% from that of the test graph. In order to test the effect of different features and a different number of boosting iterations, we build 9 LADTree classifiers. The classifiers are built using 3 different types of feature vectors; the *full feature vector* that incorporates all 17 features described in Section 3.1.2, the *graph feature vector* that uses only the graphlet features and the *non-graph feature vector* that uses only the non-graphlet based features. For each of the feature vectors, we build a classifier using 50, 100 and 200 boosting iterations, giving 9 classifiers in total for each experiment. Finally, we use the classifiers to classify the Facebook graph. The model that obtains the best score is considered to be the best fir for the test data.

3.2.1 Testing the Classifier

Before performing our experiments on the actual Facebook data, it is important to test the classifier to find out how we should interpret the results. To this end, we generate an additional 100 graphs from each of the models, and apply the classifier to this known data set. Since we know exactly which model these synthetic graphs belong to, we can test whether or not our classifier can predict the correct model. Moreover, this should establish an important baseline for the maximum and minimum possible scores achievable by each model.

We also test the robustness of the classifier. To do this, we take the 600 synthetic graphs and change a percentage of the edges by removing an edge from the graph and replacing it with a new edge chosen uniformly at random. The goal is to see how fast the classification accuracy deteriorates as a greater number of edges are changed. Overall, we have 6 test data sets of 600 graphs each, with 0%, 5%, 10%, 15% , 20% and 25% of the edges randomly changed. We generate the initial 600 graphs with the same density as the Princeton network and classify them using the LADTree classifiers we have generated for the Princeton data. To determine the importance of the graphlet features, we consider the classifiers built using both the full feature vector and the graph feature vector.

Models	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
PA	**8.96 ± 1.18**	-3.91 ± 2.39	-4.16 ± 1.18	0.17 ± 1.69	-2.38 ± 1.25	1.32 ± 0.82
COPY	-2.3 ± 0.34	**7.02 ± 0.24**	-2.19 ± 0.27	-0.19 ± 0.23	-3.2 ± 0.28	0.85 ± 0.27
GEO2D	-6.78 ± 1.59	-7.82 ± 3.55	**9.13 ± 2.89**	2.65 ±2.42	3.57 ± 1.47	-0.76 ± 1.55
SPA2D	-5.51 ± 2.5	-11 ± 3.86	2.89 ± 2.27	**10.16 ± 3.05**	-2.36 ± 2.04	5.81 ± 1.76
GEO3D	-6.14 ± 1.31	-8.42 ± 3.18	3.58 ± 1.61	-0.73 ± 1.05	**9.04 ± 2.94**	2.67 ± 2.32
SPA3D	-4.09 ± 2.48	-9.97 ± 4.54	0.03 ± 2.2	5.22 ± 2.06	-0.26 ± 2.79	**9.07 ± 2.84**

Table 3.2: Average value with standard deviation for full feature vector with 50 Boosting iterations

Consider the scores generated by the classifier for the unchanged synthetic graphs, shown in Tables 3.2, 3.3, 3.4, and 3.5. As expected, the graphs are overwhelmingly

Models	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
PA	**11.92 ± 0.89**	-4.61 ± 1.25	-4.61 ± 1.93	2.39 ± 1.65	-5.11 ± 1.46	0.03 ± 1.03
COPY	-5.5 ± 1.67	**11.73 ± 0.80**	-0.34 ± 1.11	1.37 ± 1.02	-8.25 ± 1.93	0.99 ± 1.4
GEO2D	-10.83 ± 1.96	-10.64 ± 4.54	**12.59 ± 3.19**	4.07 ± 2.42	6.02 ± 1.91	-1.2 ± 1.84
SPA2D	-8.08 ± 3.57	-13.61 ± 4.57	3.04 ± 2.88	**13.79 ± 3.72**	-2.38 ± 3.45	7.25 ± 1.99
GEO3D	-10.72 ± 2.21	-12.55 ± 5.11	5.81 ± 2.30	1.56 ± 2.07	**13.25 ± 3.79**	2.66 ± 2.4
SPA3D	-6.62 ± 3.79	-13.04 ± 5.68	-0.09 ± 2.97	6.94 ± 2.17	-0.45 ± 4.13	**13.26 ± 4.05**

Table 3.3: Average value with standard deviation for full feature vector with 100 Boosting iterations

Models	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
PA	**16.69 ± 1.5**	-5.9 ± 1.55	-6.62 ± 2.75	2.42 ± 2.21	-8.90 ± 2.15	2.31 ± 1.66
COPY	-8.2 ± 4.9	**18.31 ± 0.8**	-6.31 ± 1.61	1.95 ± 2.03	-9 ± 2.46	3.24 ± 2.05
GEO2D	-16.79 ± 4.09	-18.23 ± 7.03	**19.27 ± 3.0**1	5.48 ± 3.05	9.41 ± 3.44	0.86 ± 3.18
SPA2D	-10.75 ± 5.6	-20.41 ± 6.61	5.46 ± 4.24	**19.53 ± 4.40**	-3.71 ± 5.34	9.88 ± 2.56
GEO3D	-17.57 ± 4.34	-21.78 ± 8.46	8.92 ± 3.44	2.32 ± 2.96	**20.5 ± 4.98**	7.6 ± 4.12
SPA3D	-8.73 ± 5.76	-20.74 ± 7.99	0.82 ± 4.55	9.59 ± 2.7	0.07 ± 5.94	**18.99 ± 4.06**

Table 3.4: Average value with standard deviation for full feature vector with 200 Boosting iterations

assigned to the class corresponding to the model that generated them. The scores range roughly between -10 and 10 for 50 boosting iterations, -15 and 15 for 100 boosting iterations and -25 and 25 for 200 boosting iterations for both the full and graph features. The performance of the classifier is consistent over the different number of boosting iterations.

To determine the importance of the graphlet features, we compare the performance of the classifiers built using the full feature vector with those built using only the graph feature vector. Table 3.5 shows the performance on the synthetic graph when only the graph feature vector is used. Almost all graphs are classified correctly. In comparing Tables 3.5 and 3.3, we can observe that the test graphs receive similar scores regardless of whether the full feature vector or the graph feature vector is used. In some cases, using the graph feature vector produced higher scores for the geometric-based models but not significantly higher. Thus we can conclude that graphlets alone are sufficient to recognize the graph structure of the models under consideration.

Models	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
PA	**10.36 ± 0.51**	0.45 ± 0.44	-5.26 ± 0.72	-0.6 ± 1.02	-6.19 ± 0.78	1.24 ± 0.98
COPY	0.07 ± 0.95	**9.71 ± 0.85**	-5.27 ± 0.36	0.35 ± 1.16	-5.95 ± 0.64	1.1 ± 0.36
GEO2D	-12.18 ± 3.11	-14.9 ± 4.77	**13.86 ± 3.98**	5.97 ± 3.11	6.53 ± 2.49	0.72 ± 1.85
SPA2D	-11.71 ± 2.89	-13.74 ± 4.32	2.35 ± 2.55	**15.41 ± 3.44**	-0.78 ± 3.31	8.47 ± 2.27
GEO3D	-13.21 ± 3.28	-15.36 ± 4.39	7.45 ± 1.86	3.27 ± 2.02	**13.39 ± 2.84**	4.46 ± 2.67
SPA3D	-11.41 ± 3.34	-14.22 ± 4.61	-0.03 ± 2.32	9.06 ± 2.17	1.62 ± 3.56	**14.99 ± 3.03**

Table 3.5: Average value with standard deviation for graph feature vector with 100 Boosting iterations

Finally, we test the robustness of the classifier with respect to perturbations of the graph structure. Tables 3.6 and 3.7 give the classification accuracy for each of the 6 test data sets using the full feature vector and graph feature vector respectively. The classification accuracy on the original unchanged test data is very high for both the full and graph feature vectors. The classification accuracy is slightly but not significantly higher when only the graph feature is used. When 5% of the edges are changed, the classification accuracy for the full feature drops to just below 75%, while for the graph feature vector, the accuracy is just below 80%. In this case, the graph feature vector alone performs better than the full feature vector. For all other percentages of edge changes, the difference between the two is not significant.

The conclusion of this experiment is that the graph features alone provide just as much information as the full feature set. In fact, as is the case when 5% of the edge were changed, including additional non-graph information can decrease the accuracy of the classifier. When 10% of the edges are changed, both feature vectors give classification accuracies around 65%, which is still a fair performance. When 15% of the edges are changed, the accuracy for both feature vectors drops to around 55%. At 20% and 25%, the accuracy dips below 50%. The accuracy at this level is not good, but there clearly still is information present in the link structure, since classifying the graphs completely at random would give the correct classification less than 17% of the time.

Another interesting observation is that the overall classification accuracy does not necessarily increase with the number of boosting iterations. It is the case that increasing the number of boosting iterations improves the classification accuracy on the unchanged data, but this is not necessarily the case for the changed data. For most of the test data sets, the difference is not significant, but when 25% of the edges are changed, the classification accuracy is about 3% better when only 50 boosting iterations are performed as compared to 200 boosting iterations. We suspect that increasing the number of boosting iterations leads to an over fitting of the perturbed data.

Edge Changes	Boosting Iterations		
%	50	100	200
0	94.67	95.67	97.17
5	73.83	71.5	74.33
10	64	63.33	65.17
15	57.33	56.17	56.33
20	51.17	48.67	48.83
25	44.17	43	41.17

Table 3.6: Classification accuracy for full feature

Edge Changes	Boosting Iterations		
%	50	100	200
0	94.83	96.67	97.83
5	78.67	79.83	79.67
10	64	63.5	63.67
15	56.17	55.67	54.8
20	49.33	48	48.17
25	44	40.5	40.67

Table 3.7: Classification accuracy for graph feature

To find out exactly how the graphs are misclassified, we present in Table 3.8, the complete classification results for the classifier trained with the graph feature vector. Here we can see that the 3D models (GEO3D and SPA3D) are very robust against the changing of edges while their 2D (GEO2D and SPA2D) counterparts are not. Precisely, a large part of the misclassification of perturbed graphs is due to classification of GEO2D and SPA2D as GEO3D and SPA3D respectively. Even with the lowest level of perturbation of 5%, roughly half of the 2D models are classified as their 3D counterparts. When 25% of the edges have been changed, only around 5% of their 2D models are classified correctly, with most of the graphs being classified as the 3D counterpart. Meanwhile, the 3D models maintain good classification accuracy even when 25% of the edges are changed.

Another interesting observation is that the COPY model is also somewhat robust against the changing of edges. Even with 5% of the edges switched, all the COPY graphs are classified correctly. The accuracy dips to around 95% when 10% of the edges are changed. Even when 25% of the graph is changed, the classification accuracy stays within 50%–70%. The PA model, on the other hand, is not robust against the changing of edges. The classification accuracy quickly decreases as edge changes start to accumulate. Interestingly, PA graphs are confused only with the copy model, not with any geometric model. Note that PA graphs are never confused with SPA models, even though both models incorporate the preferential attachment principle. A reasonable explanation for this is that the SPA model generates too many triangles and 4-cliques as compared to the PA model, and changing 25% of the edges is not a sufficient enough change to alter this reality. The same reason would explain why the PA model is never confused with the GEO models.

One purpose of testing the robustness of the classifier is to attempt to simulate the behavior of the classifier on noisy data. One conclusion is that even if a little bit of noise is introduced into the data, the 2D models are more likely to get classified as a 3D model. The conclusion is that if unknown data is classified as a 3D model, it is possible that the correct model should be the 2D model. We also can conclude that using the graph feature vector may be more reliable than using the full feature vector.

3.2.2 Classification of the Facebook Networks

After verifying the quality of the classifier, we now apply the classifiers to the data sets for which they were designed. In Table 3.9, we present the classification scores for each of the four data sets, for classifiers built using the full feature vector, graph feature vector, and the non-graph feature vector. The highest score is in bold; when

Change	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D	Models
0%	100	0	0	0	0	0	PA
	0	100	0	0	0	0	COPY
	0	0	92	2	6	0	GEO2D
	0	1	0	97	0	2	SPA2D
	0	0	5	0	95	0	GEO3D
	0	0	0	4	0	96	SPA3D
5%	88	2	0	0	0	10	PA
	0	100	0	0	0	0	COPY
	0	0	49	2	49	0	GEO2D
	0	0	0	47	0	53	SPA2D
	0	0	4	0	96	0	GEO3D
	0	0	0	1	0	99	SPA3D
10%	78	14	0	0	0	8	PA
	0	94	0	6	0	0	COPY
	0	0	11	1	88	0	GEO2D
	0	0	0	3	1	96	SPA2D
	0	0	3	0	97	0	GEO3D
	0	0	0	1	1	98	SPA3D
15%	51	45	0	0	0	4	PA
	0	82	0	18	0	0	COPY
	0	0	7	2	91	0	GEO2D
	0	0	0	2	6	92	SPA2D
	0	0	2	0	98	0	GEO3D
	0	0	0	1	5	94	SPA3D
20%	24	76	0	0	0	0	PA
	0	69	0	31	0	0	COPY
	0	0	8	2	90	0	GEO2D
	0	0	0	2	12	86	SPA2D
	0	0	4	0	96	0	GEO3D
	0	0	0	1	10	89	SPA3D
25%	2	98	0	0	0	0	PA
	0	53	0	46	1	0	COPY
	0	0	6	2	92	0	GEO2D
	0	0	1	4	18	77	SPA2D
	0	0	4	0	96	0	GEO3D
	0	0	0	1	17	82	SPA3D

Table 3.8: Classification of perturbed graphs. Graph feature vector with 100 boosting iterations.

two scores are close both are in bold.

Classifier	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
full Princeton	-0.303	-14.551	4.599	**11.287**	-5.451	4.42
graph Princeton	**6.699**	-2.227	-3.914	3.085	-3.676	0.033
non-graph Princeton	-0.858	-3.622	-7.447	**8.022**	-5.029	**8.941**
full American	-0.414	-12.164	-0.183	8.307	-5.578	**10.025**
graph American	0.779	-10.639	0.381	5.834	-7.693	**11.332**
non-graph American	-4.612	-2.442	-3.627	**6.517**	-3.348	**7.512**
full MIT	2.956	-12.512	2.715	**13.528**	-8.561	1.873
graphs MIT	4.097	-9.49	3.061	**5.304**	-2.91	-0.063
non-graph Brown	-0.197	-3.58	-2.61	**4.549**	-1.606	3.44
full Brown	4.998	-15.163	-0.305	1.733	-6.161	**14.897**
graphs Brown	**6.283**	-0.085	-3.774	1.827	-3.771	-0.479
non-graph MIT	1.956	-7.305	-2.458	2.518	-2.901	**8.192**

Table 3.9: Scores for each data set, for each of the classifiers with 100 boosting iterations

The first clear conclusion from the outcome of the experiments is that all significantly high scores are for models that incorporate the preferential attachment principle: PA, SPA2D and SPA3D. In most cases, both the SPA2D and SPA3D give high positive scores. From our analysis of the classifier on perturbed graphs, we know that misclassification between SPA2D and SPA3D is common. The only reasonable conclusion is that the SPA model in general fits the data well. A different type of analysis would need to be done to determine the dimension. The PA model gives the highest score for two of the data sets, but only with the classifier that uses the graph features. When testing our classifiers on synthetic graphs from our six models, we concluded that graphlet features were at least as efficient as the full feature set in distinguishing the models. For real data, the situation is clearly different. From Table 3.9 we see that there can be a fairly large discrepancy between graph feature and the full feature results. Since the full feature set is based on the most complete set of features, it reasons we should base our final conclusion on the classifier built using all the features. In this light, the SPA model clearly gives the best fit for the

Facebook data.

Classification algorithms are built under the assumption that the test data actually belongs to one of the classes the classifier is trained to distinguish. This assumption is often not met in realistic applications, as is the case here, though it is common practice to evaluate unknown data using a classification algorithm. With this in mind, it is important for us to exercise caution when interpreting our results from the Facebook data. To enhance our understanding of the results, we consider how each feature contributes to the score for each model. Specifically, we analyze the features that appear in the first layer of nodes (of depth 1) in the ADT as they are the most influential in classifying the data. Furthermore, we consider how often each feature is visited when the classifier is applied to the Facebook data. Combined with our knowledge about the different models, and their typical behavior in regards to the selected features, this analysis will provide a more detailed picture of the classification results. Here we give an overall discussion of our conclusions from this analysis. A precise analysis for each of the four experiments can be found in Appendix A. To further aid our analysis, we generate box plots to visualize how well each feature generated in the models matched the features generated in the Facebook graph.

Our first observation is that in every classifier built using the full feature vector, the first node in the ADT corresponds to the assortativity coefficient. Therefore, the assortativity coefficient is the most significant in separating the classes. In Figure 3.2, we show the box-plot for the assortativity coefficient from the Princeton experiment.

We can see that the assortativity coefficient is significantly higher for the GEO model than the rest of the models. This can also be explained theoretically. The GEO model has a binomial degree distribution that implies that many vertices will have similar degrees leading to a higher assortativity coefficient. Note that the assortativity coefficient is not included in the graph feature vector. Since we concluded from our analysis on synthetic graphs that the graph feature and full feature have the

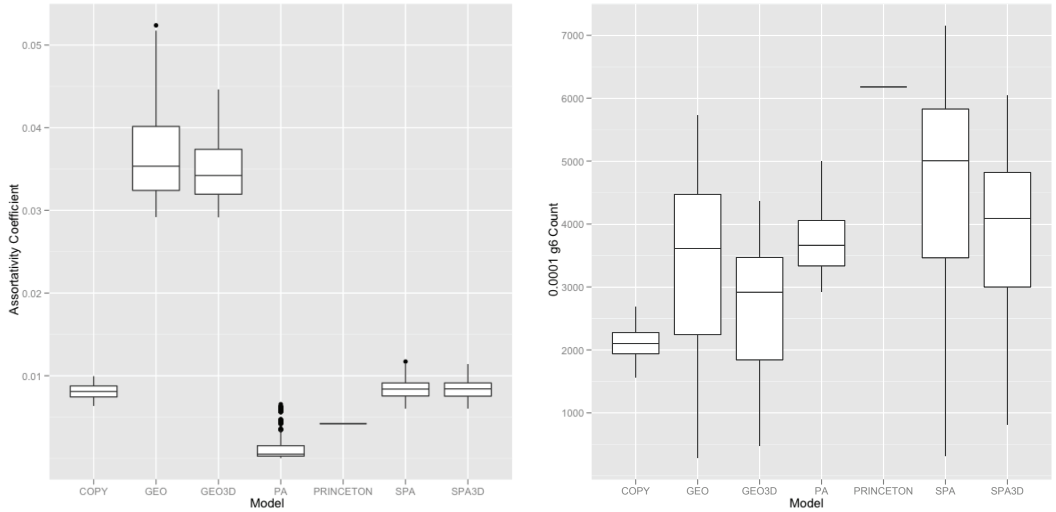


Figure 3.2: Box-plots representing the spread of the assortativity coefficient (left) and the $g6$ graph feature (right) for the Princeton network.

same classification accuracy, we suspect that the assortativity coefficient is implicitly contained in the graphlet counts.

The most important graph feature was g_6, which corresponds to the 4-cycle. The 4-cycle feature tends to be the most important feature overall. It appears frequently in the first layer of nodes in the ADT, and it is usually the feature that is most visited by the Facebook data when it is applied to the classifier. In some cases, the outcome of the classification can be deduced by only considering the 4-cycle. In most cases, the SPA models were the models that were able to generate 4-cycle counts that were the closest to the 4-cycle counts in the Facebook graphs. This can be seen in the boxplot for the Princeton experiment in Figure 3.2. This is a major factor in explaining why the SPA models performed so well in our experiments. Recall that this observation coincides with the expected 4-cycle counts in the model in Chapter 2.

An important difference between the models is that the PA and COPY models tend not to generate highly connected subgraphs whereas the GEO models do tend to generate highly connected subgraphs. Conversely, the PA and COPY models generate many sparse subgraphs whereas the GEO models do not. By highly connected subgraphs we mean those that contains a triangle, namely: g_2, g_5, g_7. Sparse subgraphs are those without a triangle: g_1, g_3, g_4. For some experiments, the ability of the PA model to generate a number of 3-paths and 4-paths in almost perfect agreement with the number present in the Facebook graphs resulted in the PA model receiving the highest score when the graph feature vector was used. Specifically, this was the case for the Princeton and Brown networks. Overall, the SPA model's ability to generate a mixture of dense subgraphs and sparse subgraphs, explains the superior overall performance of the SPA model.

A final interesting observation comes from comparing the experiments for the Princeton and Brown networks. Though the Princeton network has 6596 vertices and the Brown network has 8600 vertices, they have almost the same edge density. The conclusions of the two experiments are similar and for the graph feature vector in particular, they are almost identical. Moreover, the ADTs produced for each of the networks are very similar. They have the exact same first layer of nodes for both the full and graph feature vector. This suggests that training sets with graphs of the same density generate similar ADTs. This indicates that the same classifier could be used to classify test data that has comparable size and density. If the appropriate normalization factor could be found for comparing graphlet counts from graphs of different sizes but similar densities, then the building of the classifier would only have to be done once. The same classifier could then be applied to suitably normalized feature vectors of the data. Since the counting of the graphlets in the training data is the most time consuming step of the experiment, such a normalization would greatly increase efficiency.

Chapter 4

Discussions and Further Work

A significant conclusion of this thesis stems from the theoretical analysis in Chapter 2. In this chapter, we show that each complex network model has a unique profile in terms of the graphlets they generate. This phenomena is also observed through the experimental procedure performed in Chapter 3. The result in Theorem 2.1.2 provides a relationship between the expected number of triangles and 3-paths and the power law coefficient for graphs that have a degree distribution whose expectation follows a power law. Furthermore, we show that even for models that have the same power law degree distribution, the level of clustering is vastly different.

In Chapter 3 we argue that the preferential attachment mechanism is the most important mechanism in the formation of a Facebook network. Moreover, we argue that the SPA model provides the best fit to the Facebook data considered. This occurs for two reasons: the SPA model is able to generate a number of 4-cycles which provides the best fit to what is observed in the Facebook data, and the SPA model provides a reasonable fit to all of the graphlets considered. To see why the 4-cycle is an important graphlet in the classification algorithm we need to consider the box plots shown in Figure 3.2. Looking at the spread of 4-cycles in each of the models, we see that each model has a unique profile. Thus, when the classifier is learned, it must distinguish each of the 6 different models from one another. Thus, in the ADT, the 4-cycle feature gets used in prediction nodes much more frequently than any other feature. It is difficult to determine what the 4-cycle might mean in the context of a social network. One possibility is that the edges that are not present in the 4-cycle represent previously existing friendships that turned sour.

The experimental procedure in Chapter 3 provides insight into improvements that can be made in developing a new model for social networks. One possibility is to build a model that provides a match to the graphlet structure observed in complex networks as opposed to matching the common complex network properties. Another possibility is to provide a slight adjustment to the SPA model. We observed that the SPA model does a good job at providing a fair match to all of the graphlet features used. However, it does tend to fall short of generating enough 3-paths and 4-paths. Recall that the PA model was able to produce the right amount of 3-paths and 4-paths. To bridge the gap for the SPA model we could allow for an additional d edges to be added to each new vertex where the end points are selected preferentially according to degree. Note that this step would ignore the geometry aspect of the model.

In Chapter 2, we tabulated results for the expected number of triangles, 3-paths and 4-cycles in our selected models. For this analysis, we only computed the expected value and did not provide any analysis on how far a typical observation tends to deviate from the expectation. Through the computational analysis of these models in Chapter 3, we suspect that many of these graphlet counts may be concentrated around their expectations. An avenue of future work would be to prove mathematically that this is indeed the case. Additionally, it would be interesting to study different random graph models and compare the expected number of small subgraphs generated in these models to our selected models. Specifically, for a graph with dn edges, we show in Figure 2.1 that the maximum number of triangles is $\Theta(n^{\frac{3}{2}})$, the maximum number of 3-paths is $\Theta(n^2)$ and the maximum number of 4-cycles is $\Theta(n^2)$. We have examples from our models of graphs which generate $\Theta(n^2)$ 3-paths and $\Theta(n^2)$ 4-cycles, but we do not have any graph models which generate $\Theta(n^{\frac{3}{2}})$ triangles. The highest order we obtain for the expected number of triangles is $\Theta(n)$. It would be interesting to explore whether or not any other popular complex network models generate an order closer to the maximum.

In Chapter 3, we verified that classifiers built using graphlets were very accurate at distinguishing graphs from our selected models. A potential avenue of further analysis is to increase the number of models under consideration and evaluate the classifier's performance as the number of training models increases. Another future research topic is to apply the model-selection procedure of Chapter 3 to different types of complex networks with a different selection of models to see which one is the most appropriate. As noted in Chapter 3, a new classifier, and as a result new training data, needs to be generated for each new test graph with different size and density. If we were able to determine an appropriate normalization so that graphlet counts in graphs of different size and density could be compared, then we would only have to generate one classifier. Unfortunately, the results of Chapter 2 do not point to the validity of using any normalizing constant to compare graphlet counts amongst graphs with dn edges. To illustrate this point, let's consider 3 of our models: the original PA, the pure copy model with $p = \frac{1}{2}$ and the SPA model with $0 < pA_1 < \frac{1}{2}$. Let us attempt to normalize the graphlet counts for the size 3 graphlets. Let's suppose that we simply use n as our normalizing constant. For the PA model we get normalized triangle and 3-path counts of $\Theta(\frac{ln(n)^3}{n})$ and $\Theta(ln(n))$ respectively. For the pure copy model we have $\Theta(n^{-\frac{1}{4}})$ and $\Theta(n^{\frac{1}{4}})$ respectively. For the SPA model we have $\Theta(1)$ and $\Theta(1)$ respectively. Now as $n \to \infty$, the concentrations of triangles in the PA and the copy model both go to zero even though the copy model asymptotically generates many more triangles than the PA model. Our selection of n as the normalizing constant has resulted in a loss of information about the different amounts of triangles in the PA and copy models. The purpose of this normalizing constant is to allow for the comparison of graphlets, not hinder it. Attempting a different normalization constant in this case results in a similar loss information as $n \to \infty$.

This unfortunately means that the entire procedure of generating the training data, counting the graphlets and training the classifier needs to be repeated for each experiment, a process that is extremely time consuming. The only way we can improve the efficiency of our procedure is to develop faster algorithms for counting graphlets. Luckily, there does appear to be some head way in recent papers for algorithms that will speed up graphlet counting [79, 66].

Bibliography

[1] Home page for the open source machine learning software weka. http://www.cs.waikato.ac.nz/mi/weka/. Accessed: March 2nd, 2011.

[2] Web page for the graph-tool python module. http://projects.skewed.de/graph-tool/. Accessed: July 3rd, 2010.

[3] Web page for the international network for social network analysis. www.insna.org. Accessed: 3rd, 2011.

[4] M. Adler and M. Mitzenmacher. Towards compressing web graphs. In *Proc. of the IEEE Data Compression Conference (DCC)*, pages 203–212, 2000.

[5] R. Ahlswede and G. Katona. Graphs with maximal number of adjacent pairs of edges. *Acta Mathematica Academiae Scientiarum Hungaricae*, 32:97–120, 1978.

[6] Y. Ahn, S. Han, H. Jeong, H. Kwak, and S. Moon. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 835–844, New York, NY, USA, 2007. ACM.

[7] W. Aiello, A. Bonato, C. Cooper, J. Janssen, and P. Pralat. A spatial web graph model with local influence regions. *Internet Mathematics*, 5(1):175–196, 2008.

[8] R. Albert and A. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47–97, 2002.

[9] R. Albert, A. Barabási, and H. Jeong. Diameter of the world wide web. *Nature*, 401:130–131, 1999.

[10] R. Albert, A. Barabási, and H. Jeong. Scale-free characteristics of random networks: the topology of the world wide web. *Physica A: Statistical Mechanics and its Applications*, 281:69–77, 2000.

[11] R. Albert, A. Barabási, H. Jeong, N. Oltvai, and B. Tombar. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000.

[12] L. Amaral, M. Barthélémy, A. Scala, and H. Stanley. Classes of small-world networks. *Proceedings Natl. Acad. Sci. USA*, 97:11149–11152, 2000.

[13] M. Appel and R. Russo. The maximum vertex degree of a graph on uniform points in $[0, 1]^d$. *Adv. Appl. Prob.*, 29:567–581, 1997.

[14] M. Appel and R. Russo. The minimum vertex degree of a graph on uniform points in $[0,1]^d$. *Adv. Appl. Prob.*, 29:582–594, 1997.

[15] M. Appel and R. Russo. The connectivity of a graph on uniform points in $[0,1]^d$. *Statistics and Probability Letters*, 60:351–357, 2002.

[16] L. Backstrom, P. Boldi, M. Rosa, J. Uganda, and S. Vigna. Four degrees of separation. *arXiv:1111.4570*, 2011.

[17] A. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[18] A. Barabási, H. Jeong, S. Mason, and Z. Oltvai. Lethality and centrality in protein networks. *Nature*, 411:41–42, 2001.

[19] A. Barabási, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, 311(3):590–614, 2002.

[20] A. Barabási, H. Jeong, N. Oltvai, J. Podani, E. Szathmary, and B. Tombor. Comparable system-level organization of archaea and eukaryotes. *Nature Genetics*, 29:54–56, 2002.

[21] A. Barabási, H. Jeong, and S. Yook. Modelling the internet's large-scale topology. *PNAS*, 99:13382–13386, 2002.

[22] G. Bebek, P. Berenbrink, C. Cooper, T. Friedetzky, J. Nadeau, and S. Sahinalp. The degree distribution of the generalized duplication model. *Theoretical Computer Science*, 369(1):239–249, 2006.

[23] B. Bhattacharjee, P. Druschel, K.Gummadi, M. Marcon, and A. Mislove. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 29–42, 2007.

[24] C. Bishop. *Pattern recognition and machine learning.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[25] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang. Complex networks: structure and dynamics. *Physical Reports*, 424, 2006.

[26] B. Bollobás. Almost every graph has reconstruction number three. *J. Graph Theory*, 14(1):1–4, 1990.

[27] B. Bollobás. *Random graphs.* Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2001.

[28] B. Bollobás and W. Fernandez de la Vega. The diameter of random regular graphs. *Combinatorica*, 2(2):125–134, 1982.

[29] B. Bollobás and O. Riordan. Sparse graphs: metrics and random models. *Random Struct. Algorithms*, 39(1):1–38, 2011.

[30] B. Bollobás, O. Riordan, J. Spencer, and G. Tusnády. The degree sequence of a scale-free random graph process. *Random Struct. Algorithms*, 18(3):279–290, 2001.

[31] A. Bonato. *A course on the web graph*. Graduate Studies in Mathematics. American Mathematical Society, 2008.

[32] A. Bonato and J. Janssen. Infinite limits and adjacency properties of a generalized copy model. *Internet Mathematics*, 4:199–223, 2009.

[33] I. Bordino, D. Donato, A. Gionis, and S. Leonardi. Mining large networks with subgraph counting. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 737–742, 2008.

[34] C. Borgs, J. Chayes, L. Lovász, V. Sós, and K. Vesztergombi. Limits of randomly grown graph sequences. *Eur. J. Comb.*, 32(7):985–999, 2011.

[35] C. Borgs, J. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi. Counting graph homomorphisms. In *Topics in Discrete Math*, pages 315–371. Springer, 2006.

[36] C. Borgs, J. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi. Convergent graph sequences i: subgraph frequencies, metric properties, and testing. *Advances in Math*, 219:1801–1851, 2008.

[37] C. Borgs, J. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi. Convergent sequences of dense graph ii. multiway cuts and statistical physics. *Annals of Mathematics*, 176(1):151–219, 2012.

[38] K. Borgwardt and H. Kriegel. Shortest-path kernels on graphs. In *Proceedings of the Fifth IEEE International Conference on Data Mining*, ICDM '05, pages 74–81, Washington, DC, USA, 2005. IEEE Computer Society.

[39] S. Bornholdt and H. Schuster, editors. *Handbook of graphs and networks: from the genome to the internet.* John Wiley Sons, Inc., New York, NY, USA, 2003.

[40] 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(1-6):309–320, 2000.

[41] P. Buckley and D. Osthus. Popularity based random graph models. *Discrete Mathematics*, 282:53–63, 2004.

[42] H. Bunke. Graph matching : theoretical foundations, algorithms, and applications. *Algorithmica*, 2000(2):82–88.

[43] H. Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recogn. Lett.*, 18(9):689–694, 1997.

[44] H. Bunke. Error correcting graph matching: on the influence of the underlying cost function. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(9):917–922, 1999.

[45] H. Bunke. Recent developments in graph matching. In *ICPR*, pages 2117–2124, 2000.

[46] H. Bunke, X. Jiang, and A. Kandel. On the minimum common supergraph of two graphs. *Computing*, 65(1):13–25, 2000.

[47] H. Bunke and K. Shearer. A graph distance metric based on the maximal common subgraph. *Pattern Recogn. Lett.*, 19(3-4):255–259, 1998.

[48] Q. Chen, H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. The origin of power laws in internet topologies revisited. In *IEEE INFOCOM 2002*, pages 608–617, 2002.

[49] X. Cheng, C. Dale, and J. Liu. Statistics and social network of youtube videos. In *16th International Workshop on Quality of Service, IWQoS 2008, University of Twente, Enskede, The Netherlands, 2-4 June 2008*, pages 229–238. IEEE, 2008.

[50] T. Chiba, M. Hattori, R. Ozawa, Y. Sakaki, and M. Yoshida. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc. Natl. Acad. Sci. USA*, pages 4569–4574, 2001.

[51] C. Chuah and S. Raza A. Nazir. Unveiling facebook: a measurement study of social network based applications. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, IMC '08, pages 43–56, 2008.

[52] F. Chung, L. Lu, T. Dewey, and D. Galas. Duplication models for biological networks. *Journal of Computational Biology*, 10:677–687, 2003.

[53] A. Clauset, C. Shalizi, and M. Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703, 2009.

[54] D. Corneil, I. Jurisica, and N. Prẑulj. Modeling interactome: scale-free or geometric. *Bioinformatics*, 20:3508 – 3515, 2004.

[55] S. Dorogovtsev and J. Mendes. Scaling properties of scale-free evolving networks: continuous approach. *Physical Review E*, 63:1–19, 2000.

[56] S. Dorogovtsev and J. Mendes. Evolution of networks. In *Adv. Phys*, pages 1079–1187, 2002.

[57] S. Dorogovtsev, J. Mendes, and A. Samukhin. Structure of growing networks with preferential linking. *Phys. Rev. Lett.*, 85:4633–4636, 2000.

[58] E. Drinea, M. Enachescu, and M. Mitzenmacher. Variations on random graph models for the web. *Technical Report, Harvard University, Department of Computer Science*, 2001.

[59] N. Eggemann and S. Noble. The clustering coefficient of a scale-free random graph. *Discrete Appl. Math.*, 159(10):953–965, 2011.

[60] P. Erdős and A. Rényi. On the evolution of random graphs. *Magyar Tud. Akad. Mat. Kutató Int. Közl*, 5:17–61, 1960.

[61] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. *SIGCOMM Comput. Commun. Rev.*, 29(4):251–262, 1999.

[62] K. Faust and S. Wasserman. *Social network analysis: methods and applications*. Structural Analysis in the Social Sciences. Cambridge University Press, 1994.

[63] D. Fell and A. Wagner. The small world of metabolism. *Nature Biotechnology*, 18:1121–1122, 2000.

[64] D. Fell and A. Wagner. The small world inside large metabolic networks. *Proc. Roy. Soc. London Ser. B.*, 268:1803–1810, 2001.

[65] M. Fernández and G. Valiente. A graph distance metric combining maximum common subgraph and minimum common supergraph. *Pattern Recogn. Lett.*, 22(6-7):753–758, 2001.

[66] F. Fomin, D. Lokshtanov, V. Raman, S. Saurabh, and B. Raghavendra Rao. Faster algorithms for finding and counting subgraphs. *J. Comput. Syst. Sci.*, 78(3):698–706, 2012.

[67] E. Frank, M. Hall, G. Holmes, R. Kirkby, and B. Pfahringer. Multiclass alternating decision trees. In *Proceedings of the 13th European Conference on Machine Learning*, ECML '02, pages 161–172, 2002.

[68] Y. Freund and L. Mason. The alternating decision tree learning algorithm. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, pages 124–133, 1999.

[69] Y. Freunde and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the Second European Conference on Computational Learning Theory*, EuroCOLT '95, pages 23–37, 1995.

[70] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000, 1998.

[71] T. Gärtner, P. Flach, and S. Wrobel. On graph kernels: hardness results and efficient alternatives. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop*, pages 129–143. Springer-Verlag, 2003.

[72] Y. Hirate, S. Kato, and H. Yamana. Web structure in 2005. In *Proceedings of the 4th Workshop on Algorithms and Models for the Web-Graph*, 2006.

[73] T. Horváth, T. Gärtner, and S. Wrobel. Cyclic pattern kernels for predictive graph mining. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 158–167, New York, NY, USA, 2004. ACM.

[74] M. Hurshman, J. Janssen, and N. Kalyaniwalla. Model selection for social networks using graphlets. *Internet Mathematics*, 8(4):338–363, 2012.

[75] A. Jadbabaie and V. Preciado. Spectral analysis of virus spreading in random geometric networks. In *Proc. of the 48th IEEE Conference on disease and control*, pages 4802–4807, 2010.

[76] J. Janssen, P. Pralat, and R. Wilson. Estimating node similarity from co-citation in a spatial graph model. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 1329–1333, New York, NY, USA, 2010. ACM.

[77] J. Janssen, P. Pralat, and R. Wilson. Geometric graph properties of the spatial preferred attachment model. *Advances in Applied Mathematics*, 50:243–267, 2013.

[78] A. Java, T. Finin, X. Song, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, WebKDD/SNA-KDD '07, pages 56–65, 2007.

[79] D. Kane, K. Mehlhorn, T. Sauerwald, and H. Sun. Counting arbitrary subgraphs in data streams. In *ICALP (2)*, pages 598–609, 2012.

[80] P. Kelly. A congruence theorem for trees. *Pacific J. Math*, 7:961–968, 1957.

[81] J. Kleinberg, S. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkin. The web as a graph: measurements, models, and methods. In *Proceedings of the 5th annual international conference on Computing and combinatorics*, COCOON'99, pages 1–17, 1999.

[82] W. Kocay. Some new methods in reconstruction theory. *Lecture Notes in Mathematics*, 952:89–114, 1982.

[83] M. Kolountzakis, G. Miller, R. Peng, and C. Tsourakakis. Efficient triangle counting in large graphs via degree-based vertex partitioning. *Internet Mathematics*, 8(1-2):161–185, 2012.

[84] P. Krapivsky and S. Redner. Network growth by copying. *Physical Review E.*, 71, 2005.

[85] R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 611–617, 2006.

[86] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the web graph. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, FOCS '00, pages 57–65, 2000.

[87] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Trawling the web for emerging cyber-communities. *Computer networks*, 31(11):1481–1493, 1999.

[88] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 591–600, 2010.

[89] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 177–187, 2005.

[90] B. Mckay. Small graphs are reconstructible. *Australas. J. Combin.*, 15:123–126, 1997.

[91] M. Middendorf, C. Wiggins, and E. Ziv. Inferring network mechanism: The drosophila melanogaster protein interaction network. *PNAS*, 102:3192–3197, 2005.

[92] S. Milgram. The small world problem. *Phsyc. Today*, 2, 1967.

[93] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U.Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[94] T. Móri. On random trees. *Studia Sci. Math. Hungar.*, 39:143–155, 2003.

[95] T. Móri. The maximum degree of the barabśialbert random tree. *Comb. Probab. Comput.*, 14(3):339–348, 2005.

[96] P. Mucha, M. Porter, and A. Traud. Social structure of facebook networks. *http://arxiv.org/abs/1102.2166*, 2011.

[97] R. Hofstad N. Litvak. Uncovering disassortativity in large scale-free networks. *http://arxiv.org/abs/1204.0266v3*, 2012.

[98] C. Najim and R.Russo. On the number of subgraphs of a specified form embedded in a random graph. *Methodology and Computing in Applied Probability*, 5:23–33, 2003.

[99] M. Newman. Scientific collaboration networks: i. network construction and fundamental results. *Phys. Rev. E.*, 64, 2001.

[100] M. Newman. Scientific collaboration networks: ii. shortest paths, weighted networks and centrality. *Phys. Rev. E.*, 64, 2001.

[101] M. Newman. The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci. USA*, 98:404–409, 2001.

[102] M. Newman. Mixing patterns in networks. *Phys. Rev.*, 67, 2003.

[103] M. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[104] M. Newman, S. Strogatz, and D. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E.*, 64, 2001.

[105] R. Pastor-Satorras, E. Smith, and R. Solé. Evolving protein interaction networks through gene duplication. *Journal of Theoretical biology*, 222(2):199–210, 2003.

[106] R. Pastor-Satorras and A. Vespignani. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.*, 86:3200–3202, 2001.

[107] R. Pastor-Satorras and A. Vespignani. *Evolution and structure of the internet: a statistical physics approach.* Cambridge University Press, New York, NY, USA, 2004.

[108] M. Penrose. *Random Geometric Graphs.* Oxford Studies in Probability. Oxford University Press, USA, 2003.

[109] N. Prẑulj. Graph theory analysis of protein-protein interaction networks. In *Knowledge Discovery in Proteomics.* CRC Press, 2005.

[110] N. Prẑulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23:177–183, 2007.

[111] N. Prẑulj and G. Higham. Modelling protein-protein interaction networks via a stickiness index. *Journal of the Royal Society Interface*, 3(10):711–716, 2006.

[112] J. Ramon and T. Gärtner. Expressivity versus efficiency of graph kernels. In *Proceedings of the First International Workshop on Mining Graphs, Trees and Sequences*, pages 65–74, 2003.

[113] J. Raymond and R. Willett. Maximum common subgraph isomorphism algorithms for the matching of chemical structures. *Journal of Computer-Aided Molecular Design*, 16:2002, 2002.

[114] S. Redner. How popular is your paper? an empirical study of the citation distribution. *Eur. Phys. Jour. B*, 4:131–134, 1998.

[115] M. Rimscha. Reconstructibility and perfect graphs. *Discrete Mathematics*, 47:283–291, 1983.

[116] B. Bollobás O. Riordan. Mathematical results on scale-free random graphs. In *Handbook of Graphs and Networks*, pages 1–37. Wiley, 2003.

[117] N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt. Efficient graphlet kernels for large graph comparison. *Journal of Machine Learning Research - Proceedings Track*, 5:488–495, 2009.

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

[119] L. Stubbs. Genome comparison techniques. In *Genomic Technologies: Present and Future*. Caister Academic Press, 2002.

[120] C. Tsourakakis. Fast counting of triangles in large real networks without counting: algorithms and laws. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 608–617, Washington, DC, USA, 2008. IEEE Computer Society.

[121] P. Turán. On an extremal problem in graph theory. *Matematikai és Fizikai Lapok*, 48:436–452, 1941.

[122] W. Wallis, P. Shoubridge, M. Kraetz, and D. Ray. Graph distances using graph union. *Pattern Recognition Letters*, 22(6-7):701–704, 2001.

[123] D. Watts. *Small worlds: the dynamics of networks between order and randomness*. Princeton Studies in Complexity. Princeton University Press, 2003.

[124] S. Wernicke. Efficient detection of network graphlets. *IEEE/ACM Transaction on Computational Biology and Bioinformatics*, 3(4):347–359, 2006.

[125] D. West. *Introduction to graph theory*. Prentice Hall, 2001.

[126] D. Wood. On the maximum number of cliques in a graph. *Graphs and Combinatorics*, 23:337–352, 2007.

[127] C. Wu Yu. Computing subgraph probability of random geometric graphs with applications in quantitative analysis of ad hoc networks. *IEEE Journal of Selected Areas in Communications*, 27(7):1056–1065, 2009.

Appendix A

Discussion for Each Facebook Experiment

A.1 Princeton

To interpret our results we consider the information in Tables A.1, A.2, A.3, and A.4 as well as the box plots in Figure A.1. For the full features vector, we can see from Table A.4, that the SPA models have the best performance. We argue that g_6, which corresponds to the 4-cycle, is the most influential feature in this case. Examining Table A.2 indicates that this is the most frequently visited feature. This feature also occurs frequently in the first layer of nodes so that it is a descriptive feature in separating the models. Comparing the number of 4-cycles in the Princeton network against the range of 4-cycle counts for each of the models in Figure A.1, we can see that the count for the Princeton network only falls in the range of SPA2D and is very close to falling into the SPA3D range. This is the main reason that SPA2D receives the highest score. Suppose that the classification was to be done by considering the 4-cycle alone. Then by observing Figure A.1 one would conclude a ranking from highest to lowest of SPA2D, SPA3D, GEO2D, PA, GEO3D, COPY which is precisely the ranking that the classifier gives on the full feature vector.

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl	g1	g2	g3	g4	g5	g6	g7	g8
50 full	1	1	0	0	0	0	1	1	3	0	2	1	2	4	**9**	1	3
100 full	1	1	0	0	0	0	1	1	3	1	3	1	2	5	**12**	3	3
200 full	3	1	0	0	0	0	2	1	3	1	6	1	5	6	**21**	4	5

Table A.1: Features visited for full feature classifier for the Princeton network

Classifier	g1	g2	g3	g4	g5	g6	g7	g8
50 graph	1	0	1	2	1	1	1	0
100 graph	2	1	1	2	1	**5**	1	0
200 graph	2	1	3	4	4	**8**	4	0

Table A.2: Features visited for graph feature classifier for Princeton network

Classifier	deg1	deg2	deg3	deg4	deg5	deg6	deg7	assort	apl
50 non-graph	0	0	0	1	0	1	1	**5**	**6**
100 non-graph	1	1	0	2	0	1	3	**12**	8
200 non-graph	1	1	1	2	2	2	3	**18**	12

Table A.3: Features visited for non-graph feature classifier for Princeton network

Since the 4-cycle was the most influential feature for the full feature vector classifier, we might suspect that it should also be the most influential feature in the graph feature classifier. While it remains an influential feature, we argue that g_4, which corresponds to the 4-path, is a more influential feature in this case. If the 4-cycle had the same importance for the graph feature classifier as it did in the full feature classifier, we would expect that the SPA models would receive the highest score in this classifier. From Table A.4 we see that this is not the case: the PA model receives the best score with the two SPA models coming in second and third. To understand this, we need to consider which feature is most influential in determining the scores for the graph feature vector. For 100 and 200 boosting iterations, Table A.2 indicates that g_6 is the most frequently visited feature. However, we must also consider which features appear in the first layer of nodes in the ADT. In this case, for 50, 100 and 200 boosting iterations, only 3 nodes appear in the first layer of nodes and 2 of these correspond to the 4-path feature. By observing the box plot in Figure A.1, you can see that the 4-path count for the Princeton network corresponds almost exactly to the median count for 4-paths in the PA model. Furthermore, the first node in the ADT corresponds to a 4-path, which assigns a score of 3.611 for the PA model and -0.722 for the other models. This gives an advantage of 4.333 for the PA model from the first node, which is close to the overall difference between the PA model and SPA models in the final classification score. Thus, for the graph features, the 4-path appears to be the most influential feature. In this light, we can see from Figure A.1, that the GEO models have the worst match of the 4-path feature in the Princeton network, which corresponds to the fact that the GEO models receive the lowest score in the graph feature classifier.

Classifier	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
50 full	0.232	-12.365	2.626	**8.359**	-3.958	5.106
100 full	-0.303	-14.551	4.599	**11.287**	-5.451	4.42
200 full	1.681	-21.364	5.148	**15.812**	-10.426	9.152
50 graph	**6.321**	-0.891	-3.471	-0.025	-3.026	1.093
100 graph	**6.699**	-2.227	-3.914	3.085	-3.676	0.033
200 graph	**9.377**	-2.667	-3.255	3.805	-7.904	0.644
50 non-graph	-0.728	-0.524	-3.379	**2.549**	-1.396	**3.48**
100 non-graph	-0.858	-3.622	-7.447	**8.022**	-5.029	**8.941**
200 non-graph	-2.97	-4.107	-12.995	9.87	-6.503	**16.708**

Table A.4: The scores for each model, for each classifier, for the Princeton network

The results for the non-graph feature vector can be easily understood by observing Table A.3. We see that the assortativity and the average path length are the most important features. By observing their box-plots in Figure A.1, you can see that the SPA models match the average path length the best. The assortativity is matched almost equally by the SPA models and the COPY model. Notice that PA model's range for the assortativity coefficient is the only one of the six models that is (almost) below the assortativity of the Princeton network. In this case, the ADT is using this feature to distinguish PA from the rest of the models. Therefore, an increasing number of visits to assortativity feature in the ADT tend to give a negative score to PA and a positive score for the other models. This explains why the score of PA is decreasing as the number of boosting iterations increase. The results for the non-graph feature vector are consistent amongst all the experiments, so we will not explain the results in detail for the other three Facebook experiments.

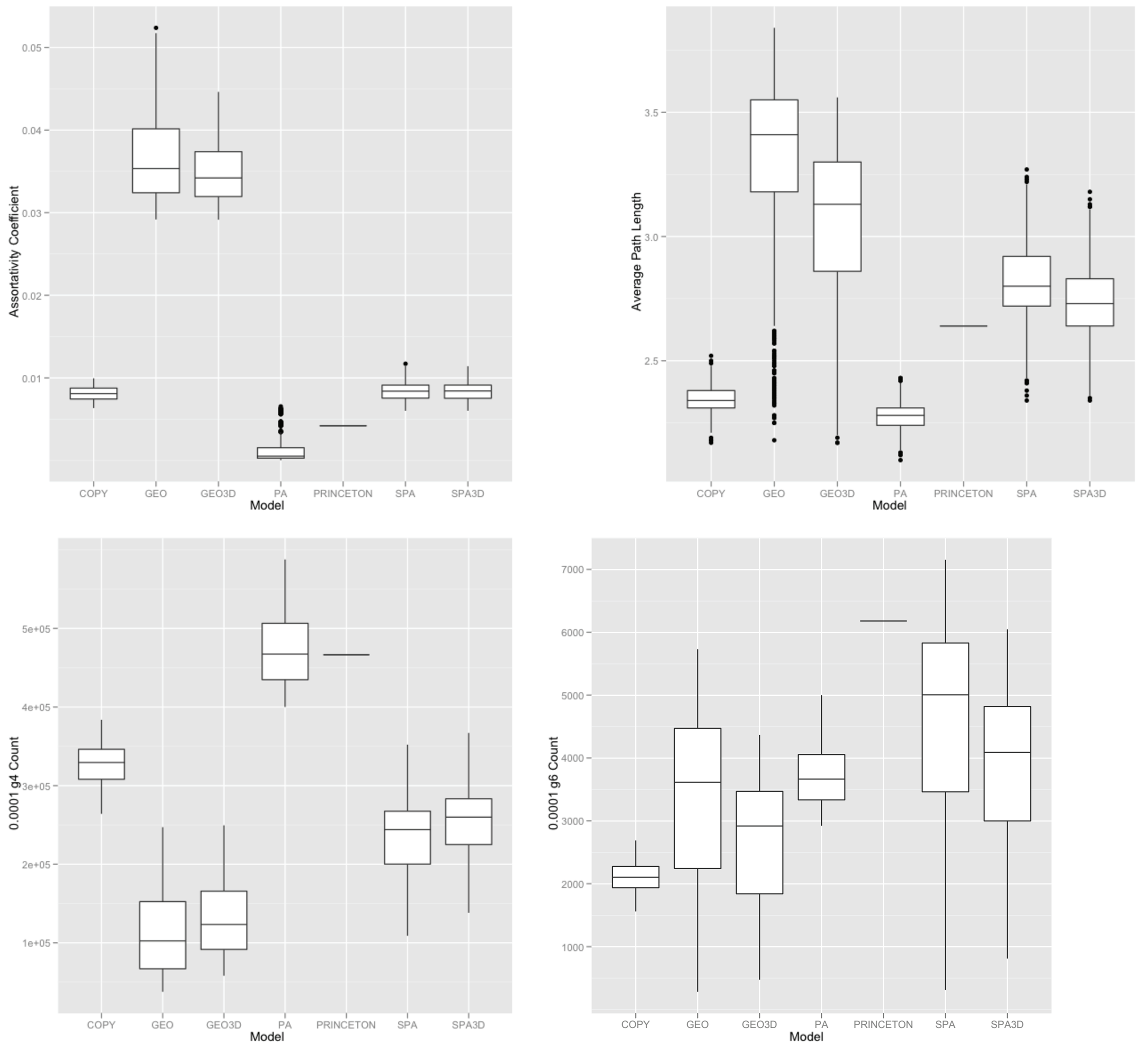


Figure A.1: Box-plots representing the spread of the features for the Princeton network.

A.2 American University

The conclusions for the American University network remain consistent with the conclusion from the Princeton experiment, though there are some differences. It remains the case that the PA and the SPA models have the best performance, but the ranking of the three models differ in some areas. Let's discuss why this occurs.

Classifier	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
50 full	-0.702	-11.768	1.091	**8.505**	-4.532	7.398
100 full	-0.414	-12.164	-0.183	8.307	-5.578	**10.025**
200 full	-0.728	-18.426	-1.191	9.955	-7.246	**17.625**
50 graph	0.172	-7.908	2.271	3.269	-5.6	**7.792**
100 graph	0.779	-10.639	0.381	5.834	-7.693	**11.332**
200 graph	4.516	-16.02	-4.241	7.002	-8.925	**17.656**
50 non-graph	0.172	-7.908	2.271	3.269	-5.6	**7.792**
100 non-graph	-4.612	-2.442	-3.627	**6.517**	-3.348	**7.512**
200 non-graph	-7.279	-2.071	-5.395	8.249	-5.439	**11.936**

Table A.5: The scores for each model, for each classifier, for the American University network

The main difference that occurs for the full feature vector in the American experiment is that SPA3D is the best and SPA2D is second. However, recall from our analysis on perturbed graphs, that misclassification between SPA2D and SPA3D does occur frequently. Only the conclusion that the SPA model is a good fit is valid. As well, the GEO2D model slips from 3rd place to 4th at 200 boosting iterations. Part of the explanation for why SPA3D performs better at this level is that the 4-cycle (g_6) count of the American network falls within the range of SPA3D as well as SPA2D, but in Princeton it only fell within the range of SPA2D. In fact, in the Princeton network, the 4-cycle feature contributed a score of -5.3 to SPA3D while it contributed a score of 1.895 for SPA3D in the American network. This difference almost completely accounts for the discrepancies in the scores between the two experiments. The explanation for why GEO2D slips to 4th place at 200 boosting iterations can be understood by considering the feature g_7. The feature g_7 corresponds to the complete graph minus an edge and is matched well by GEO2D. This feature is visited 4 times in the Princeton classifier at 200 boosting iterations and is not visited at all in the American network classifier at 200 boosting iterations. This should account for the slip in score that puts it in 4th place.

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl	g1	g2	g3	g4	g5	g6	g7	g8
50 full	0	0	1	0	0	1	0	1	2	0	1	1	4	5	**8**	0	4
100 full	0	0	1	1	0	1	0	1	3	0	1	1	5	7	**10**	0	4
200 full	0	0	1	1	0	1	0	1	3	2	5	1	8	9	**16**	0	4

Table A.6: Features visited for full feature classifier for the American network

Classifier	g1	g2	g3	g4	g5	g6	g7	g8
50 graph	1	0	0	3	4	**5**	1	4
100 graph	1	1	2	5	5	**8**	1	4
200 graph	3	1	3	8	6	**12**	2	5

Table A.7: Features visited for graph classifier for American network

Classifier	deg1	deg2	deg3	deg4	deg5	deg6	deg7	assort	apl
50 no-graph	1	0	2	1	0	0	1	**8**	**9**
100 non-graph	1	0	3	1	0	0	1	**12**	**12**
200 non-graph	1	0	3	2	0	2	2	**17**	13

Table A.8: Features visited for non-graph classifier for American network

The ranking is also different for the graph feature classifier. Instead of the ranking being PA, SPA2D, SPA3D as it was in the Princeton network, it is SPA3D, SPA2D, PA. Part of this difference can be explained because the American network visits more decision nodes when it is put through the classifier than the Princeton network visited when it was classified. Specifically the American network visits 40 nodes and the Princeton network visits 26 nodes at the 200 boosting iteration level (see Tables A.2 and A.7). Of these nodes, 26 correspond to either g_6 or dense subgraphs while 14 correspond to sparse subgraphs. Recall that the SPA model provides the best overall fit to all the graphlets. For this reason, and the fact that more features are visited in this network than in the Princeton network, the influence of the g_4 feature which resulted in the top ranking of PA in the Princeton network is weakened. It is also important to note that there are 5 nodes in the first level of nodes in the ADT at 200 boosting iterations. Of these nodes, 2 correspond to g_4 and 2 correspond to g_6 with the remaining node corresponding to g_3. In the Princeton network, 2 out of 3 of the nodes in the first layer corresponded to g_4 and none of the nodes corresponded to g_6. Therefore, for the American network with the graph feature classifier, the g_6 feature is much more influential than the g_4 which explains why the SPA model is ranked above the PA model.

A.3 MIT

The results in this experiment are not completely consistent with the other experiments. In particular, the GEO2D model finishes second for both the full feature vector and the graph feature vector. It is still the case that COPY and GEO3D occupy the last two positions in the rankings. To understand the results we consider which features are the most influential in the classification.

Classifier	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
50 full	1.23	-7.541	3.917	**9.319**	-6.482	-0.441
100 full	2.956	-12.512	2.715	**13.528**	-8.561	1.873
200 full	2.32	-13.795	2.737	**17.881**	-11.679	2.534
50 graph	**5.53**	-5.266	-1.368	3.719	-1.17	-1.441
100 graph	4.097	-9.49	3.061	**5.304**	-2.91	-0.063
200 graph	2.905	-12.759	3.962	**10.004**	-2.512	-1.603
50 non-graph	2.53	-5.236	-2.99	1.933	-2.732	**6.497**
100 non-graph	1.956	-7.305	-2.458	2.518	-2.901	**8.192**
200 non-graph	1.614	-8.182	-4.085	8.28	-8.158	**10.537**

Table A.9: The scores for each model, for each classifier, for the MIT network

For the full feature classifier, the most significant cause of the diverging results for the MIT networks is the influence of the 4-cycle feature. In the first layer of nodes of the ADT at 200 boosting iterations, there are 9 nodes and only one node corresponds to the 4-cycle. Furthermore, by observing the box plot in Figure A.3, you see that the number of 4-cycles in the MIT network does not fall into the range of any of the models, though it comes closest to SPA2D. This is abnormal as in all other experiments the number of 4-cycles's falls into the range of at least the SPA2D model. In Table A.10, we see that the g_6 feature is still the most visited. The good performance of GEO2D could be due to the fact that not many of the sparse subgraphs are visited. Of the 31 graph-based features visited in the full feature vector, only 6 correspond to sparse subgraphs.

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl	g1	g2	g3	g4	g5	g6	g7	g8
50 full	0	1	0	0	0	0	1	1	2	0	0	1	1	1	**6**	3	2
100 full	1	1	0	0	0	0	1	1	2	1	2	1	1	2	**9**	4	3
200 full	1	1	0	0	0	0	2	2	2	1	3	1	2	4	**12**	5	3

Table A.10: Features visited for full feature classifier for the MIT network

Classifier	g1	g2	g3	g4	g5	g6	g7	g8
50 graph	0	1	0	1	2	**5**	0	1
100 graph	1	4	0	1	2	**7**	0	2
200 graph	1	4	0	2	2	**9**	0	5

Table A.11: Features visited for graph classifier for MIT network

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl
50 non-graph	2	0	0	0	0	2	2	**9**	**8**
100 non-graph	2	0	0	0	1	2	3	**12**	**12**
200 non-graph	3	0	1	1	2	4	5	**19**	14

Table A.12: Features visited for non-graph classifier for MIT network

To understand the results for the graph feature classifier we only have to consider the 4-path (g_4) and 4-cycle (g_6). At 50 boosting iterations, the PA model is the best model but at 200 boosting iterations it is ranked 3rd. As explained for the Princeton network, the first node in the ADT corresponds to g_4, which gives an immediate advantage of 4.333 points for the PA model. You can see by observing in Table A.11, that the g_4 feature is not visited very often. Also, not many of the sparse graphs are visited, so the influence of sparse graphs greatly diminishes as boosting iterations accumulate and more decision nodes are added to the tree. The 4-cycle feature appears to be the most influential in determining the scores. Of the five nodes in the first layer of the ADT at 200 boosting iterations, three correspond to g_6, while the other two correspond to g_4 and g_2. The GEO2D model performs so well because of the emphasis on the denser subgraphs in determining the scores. Of the 23 nodes visited by the American network at 200 boosting iterations, only 3 correspond to the sparse subgraphs. Of these nodes, 9 of them correspond to g_2 and g_8 which represent K_3 and K_4. You can see that GEO2D matches these features well by observing the box plots in Figure A.3. For these features, you can see that SPA2D matches them just as well as GEO2D. The American network counts for these features do fall into the range for the 3D versions of these networks, but the fit is not as good as for the 2D versions. This explains why the 2D models are performing better.

A.4 Brown

The results for the Brown network are similar to the results we have already seen for the Princeton and American networks. Interestingly, the results for the graph feature classifier are almost the same as the results for the Princeton network. This is interesting because both networks have almost the same density though the Brown network has around 33% more vertices.

Let's first consider the results of the full feature classifier. As we have been seeing in many of the experiments, models incorporating the preferential attachment mechanism are ranked in the top 3. The reasoning is as explained before. The PA model finishes second because by observing Table A.14, you can see that the sparse graph features are visited in higher proportions than was typical in the other experiments. In particular, the 4-path (g_4) feature, which greatly favours PA, is visited 6 times.

The most interesting result occurs for the graph feature classifier. The ranking of the models is identical (except for an insignificant swap of the GEO models in the final position), to what they were in the Princeton network. This is interesting because both the Princeton and the Brown networks have the same edge density. The explanation for the Princeton results holds here as well. In fact, if we consider the ADT's in both experiments, for both the full and graph feature vectors, the first layer of nodes are exactly the same. This is promising to see because it indicates that training sets with the same edge density might result in the same ADT's, which would greatly cut down on the amount of computational time. All that would be needed is to determine an appropriate normalization for the graph features for two graphs with a different number of vertices.

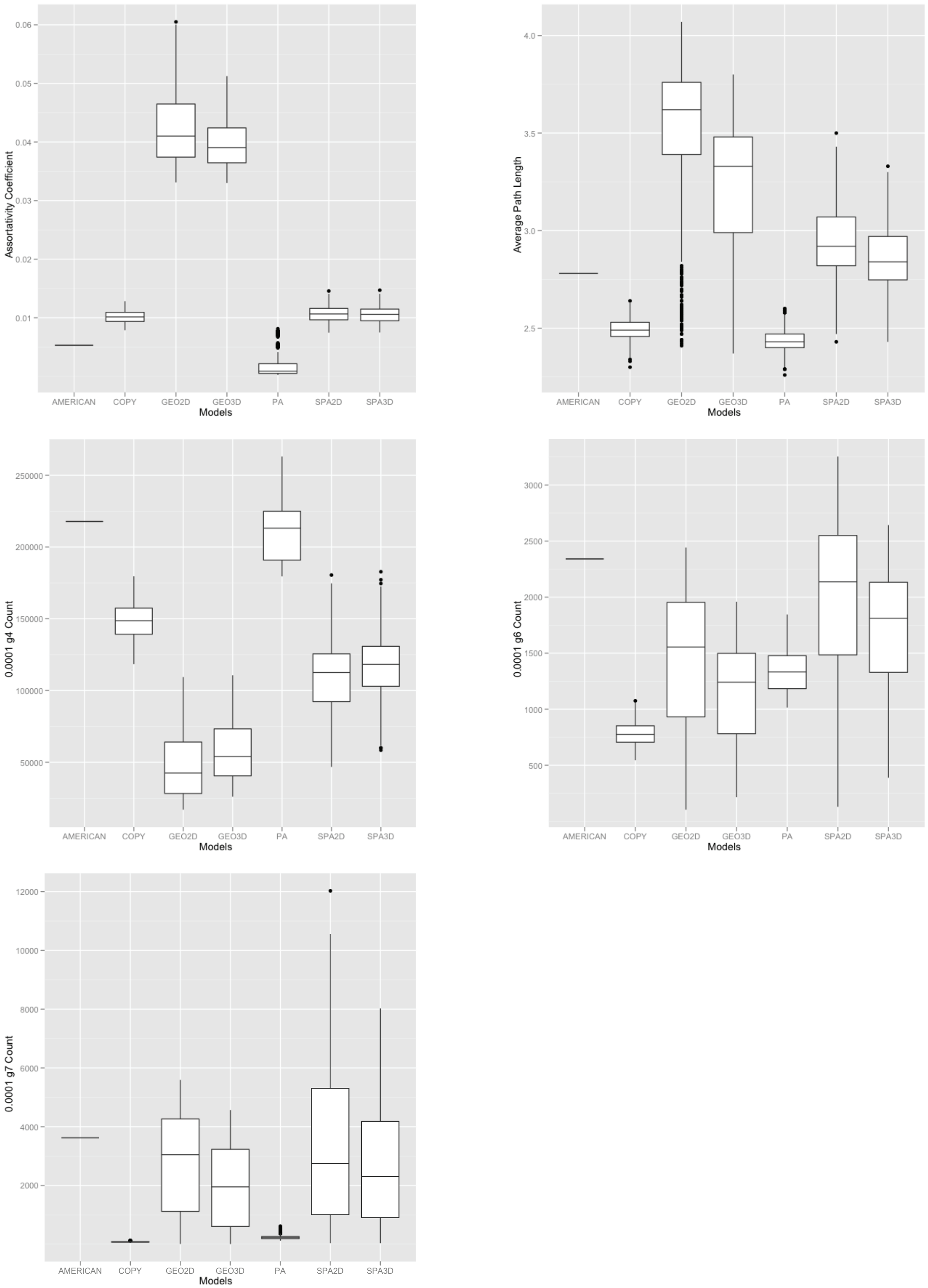


Figure A.2: Box-plots representing the spread of the features for American network.

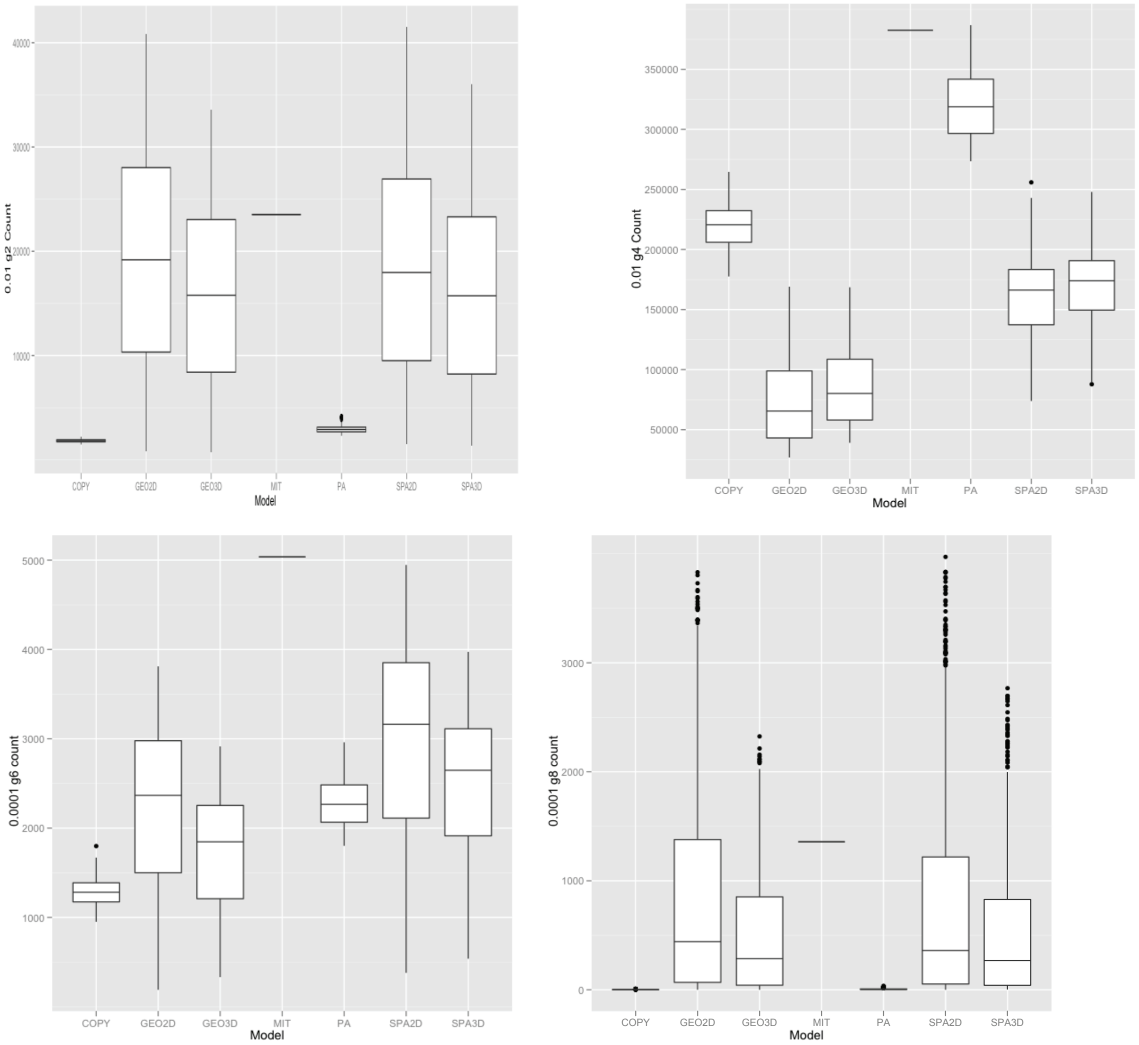


Figure A.3: Box-plots representing the spread of the features for MIT network.

Classifier	PA	COPY	GEO2D	SPA2D	GEO3D	SPA3D
50 full	3.044	-8.622	1.0	0.245	-3.191	**7.522**
100 full	4.998	-15.163	-0.305	1.733	-6.161	**14.897**
200 full	7.085	-24.459	-1.228	4.841	-9.41	**23.171**
50 graph	**5.114**	-1.087	-2.336	1.052	-2.354	-0.388
100 graph	**6.283**	-0.085	-3.774	1.827	-3.771	-0.479
200 graph	**8.672**	-0.772	-6.304	3.668	-6.296	1.033
50 non-graph	0.302	-2.357	-2.063	**2.335**	-0.522	**2.302**
100 non-graph	-0.197	-3.58	-2.61	**4.549**	-1.606	3.44
200 non-graph	1.858	-2.865	-10.064	**10.21**	-5.457	6.313

Table A.13: The scores for each model, for each classifier, for the Brown University network

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl	g1	g2	g3	g4	g5	g6	g7	g8
50 full	0	0	0	0	0	0	1	1	2	1	0	0	3	1	**6**	1	4
100 full	0	0	0	0	0	0	2	3	2	1	3	0	4	3	**11**	2	5
200 full	0	1	0	0	0	0	4	6	4	2	4	0	6	6	**17**	5	7

Table A.14: Features visited for full feature classifier for the Brown network

Classifier	g1	g2	g3	g4	g5	g6	g7	g8
50 graph	0	0	0	2	1	1	0	0
100 graph	0	0	2	2	1	1	0	0
200 graph	0	0	3	4	1	3	0	0

Table A.15: Features visited for graph feature classifier for Brown network

Classifier	d1	d2	d3	d4	d5	d6	d7	assort	apl
50 non-graph	0	1	0	0	1	2	2	**11**	**10**
100 non-graph	0	1	0	0	2	2	3	**15**	11
200 non-graph	2	2	0	2	3	4	4	**23**	14

Table A.16: Features visited for non-graph feature classifier for Brown network

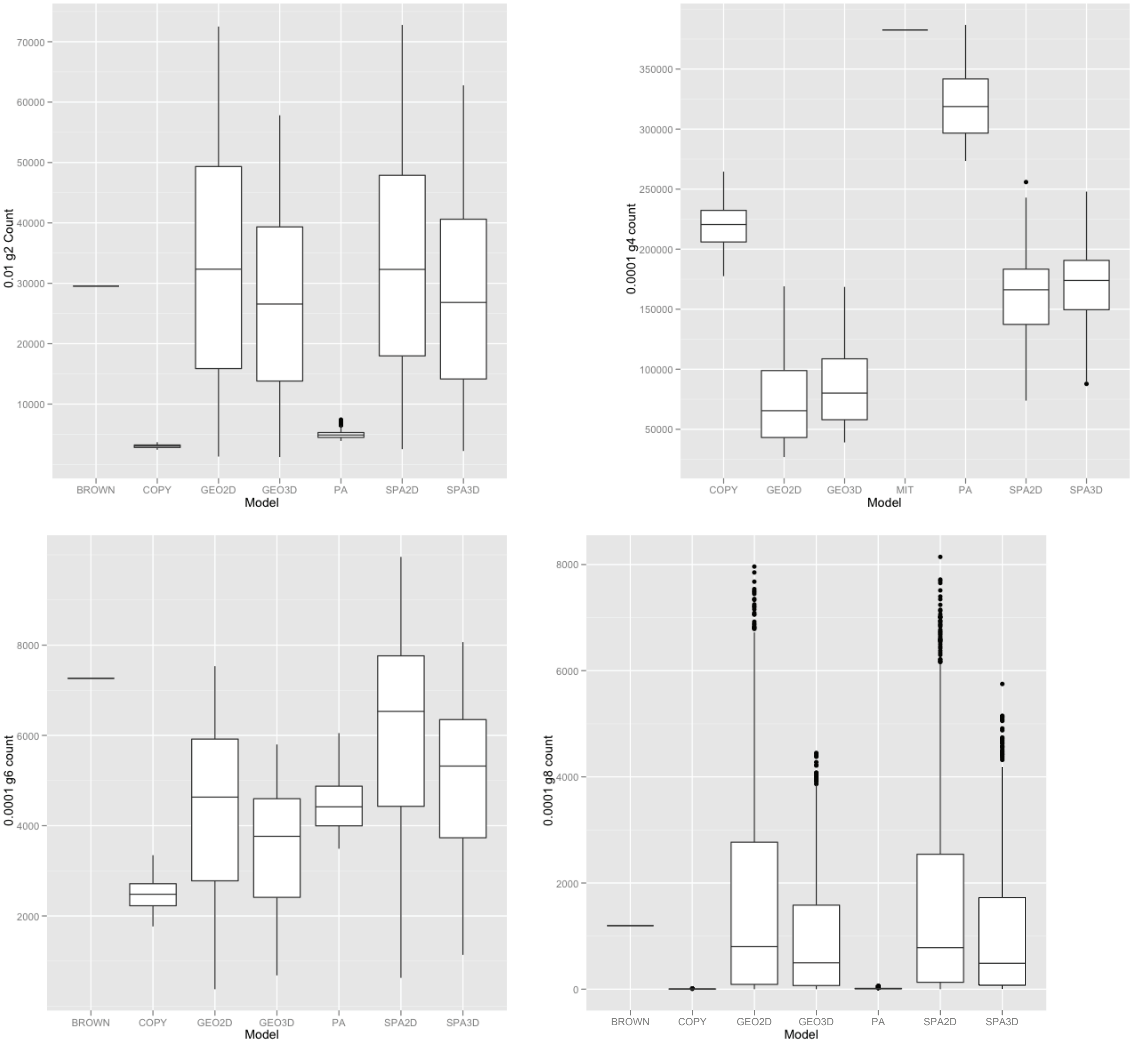


Figure A.4: Box-plots representing the spread of the features for Brown network.

Appendix B

Copyright of Model Selection for Social Networks Using Graphlets

The paper *Model Selection for Social Networks Using Graphlets* by Matt Hurshman, Jeannette Janssen and Nauzer Kalyaniwalla appeared in *Internet Mathematics*, *volume 8*, *number 4* in 2012 which is published by Taylor & Francis. Chapter 3 is based on this paper. Authors of publications in *Internet Mathematics* are allowed the following right as stated in *Taylor & Francis' position on Copyright and Author Rights:*

> 10. The right to include an article in a thesis or dissertation that is not to be published commercially. provided that acknowledgement to prior publication in the relevant Taylor & Francis journal is made explicit.

The full list of author rights can be found at http://www.tandf.co.uk/journals/authorrights.pdf (March 1st, 2013)