MATHEMATICAL FOUNDATIONS AND ALGORITHMS FOR CLIQUE RELAXATIONS IN NETWORKS

A Dissertation

by

JEFFREY LEE PATTILLO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2011

Major Subject: Mathematics

MATHEMATICAL FOUNDATIONS AND ALGORITHMS FOR CLIQUE RELAXATIONS IN NETWORKS

A Dissertation

by

JEFFREY LEE PATTILLO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Sergiy Butenko
	Huafei Yan
Committee Members,	Joseph M. Rojas
	Joel Zinn
Head of Department,	Emil Straube

December 2011

Major Subject: Mathematics

ABSTRACT

Mathematical Foundations and Algorithms for Clique Relaxations in Networks.

(December 2011)

Jeffrey Lee Pattillo, B.S., Wheaton College; M.S., Texas A&M University

Co-Chairs of Advisory Committee: Dr. Sergiy Butenko
Dr. Huafei Yan

This dissertation establishes mathematical foundations for the properties exhibited by generalizations of *cliques*, as well as algorithms to find such objects in a network. Cliques are a model of an ideal group with roots in social network analysis. They have since found applications as a part of grouping mechanisms in computer vision, coding theory, experimental design, genomics, economics, and telecommunications among other fields. Because only groups with ideal properties form a clique, they are often too restrictive for identifying groups in many real-world networks. This motivated the introduction of clique relaxations that preserve some of the various defining properties of cliques in relaxed form. There are six clique relaxations that are the focus of this dissertation: *s-clique*, *s-club*, *s-plex*, *k-core*, *quasi-clique*, and *k-connected* subgraphs. Since cliques have found applications in so many fields, research into these clique relaxations has the potential to steer the course of much future research.

The focus of this dissertation is on bringing organization and rigorous methodology to the formation and application of clique relaxations. We provide the first taxonomy focused on how the various clique relaxations relate on key structural properties demonstrated by groups. We also give a framework for how clique relaxations can be formed. This equips researchers with the ability to choose the appropriate clique relaxation for an application based on its structural properties, or, if an appropriate clique relaxation does not exist, form a new one.

In addition to identifying the structural properties of the various clique relaxations, we identify properties and prove propositions that are important computationally. These assist in creating algorithms to find a clique relaxation quickly as it is immersed in a network. We give the first ever analysis of the *computational complexity* of finding the maximum quasi-clique in a graph. Such analysis identifies for researchers the appropriate set of computational tools to solve the maximum quasi-clique problem. We further create a *polynomial time* algorithm for identifying large 2-cliques within *unit disk graphs*, a special class of graphs often arising in communication networks. We prove the algorithm to have a guaranteed 1/2-approximation ratio and finish with computational results.

Dedicated to my parents

ACKNOWLEDGMENTS

I have been truly blessed to have Dr. Sergiy Butenko as my advisor. He cares genuinely about the success of each of his students and his patient confidence in my abilities is what allowed me to reach my full potential. In my ten years of collegiate study, he has earned my respect more than anyone.

I would like to thank my advisor in the math department, Dr. Catherine Yan, as well as my unofficial advisor, Dr. Natarajan Sivakumar. Dr. Yan has been patient throughout the past few years of study and has been very supportive of my goals for graduate school. Dr. Sivakumar sat down countless times to help me explore options during my time here and it was he who pointed me toward the industrial engineering department and my current area of study. I truly enjoy my area of research and I attribute that to him.

I would like to thank my committee, including Dr. Rojas and Dr. Zinn, for helpful feedback and support during my studies. I would also like to extend thanks to Dr. Kirby Smith for many hours of help as I studied for qualifying exams, without which I would not be here. I would like to thank two undergraduate professors, Dr. Robert Brabenec and Dr. Terry Perciante, for pointing me in the direction of graduate school. Their care for the students and genuine love of what they do is responsible for my pursuit of this degree.

My time in College Station has been as much about personal growth as it has been professional. I would like to thank Ryan Poehl, Chase Friedman, Andy & Marty Hott, Tara & Stephanie, Jeff & Kirsten Selby, Robert Meinke, Jesse Ortega, Andy House, the Cruickshanks clan, Doc Jones, Renee & Anna, Tim & Jeremy, James Hall, and countless others for the friendship and fellowship we shared during my time here. They made College Station a home for me that I will always remember fondly.

I would like to thank my grandparents. They helped to set the stage for my success and have showered me with prayers all along the way. My abilities in math certainly came from "grand-dude," and I got my work ethic from my grandfather, John Boswell. He did not even graduate from high school but will get his name in a dissertation because of how hard he worked for our family.

I would like to thank my brothers. They are my best friends and each has been there beside me during the past few years to encourage me. I feel I could bear my soul to any one of them and words cannot express how much I value that.

More than anyone I would like to thank my parents. This accomplishment belongs as much to them as it does to me. They believe in me more than I believe in myself. They celebrate every small victory and shower me with prayers during trials. There is no way to repay that kind of love and support.

Last I want to thank my Lord and my God, Jesus Christ, for loving me both when I succeed and when I fail. The most valuable thing I leave with from my time in College Station is my walk with Him.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

1. INTRODUCTION

Network is a popular term for a set of nodes with edges representing interaction. The mathematical term most closely aligned with the concept of a network would be a *graph*. While a graph is a rigorously defined mathematical term, the term network is much more functional. People join a network anytime they purchase a cell-phone, sign up for membership at a store or club, or simply connect to the internet. Understanding the concept of a network has become essential to operate in our society. The term network is typically associated with a set of points and lines in space. A graph is defined to be a pair of sets (V, E) where V is a set of vertices, also sometimes called nodes, and E is the set of edges between the vertices. Thus a graph is simply a mathematical abstraction of the visualization most often associated with the term network.

Representing information as a graph allows for interrelated data to be gathered locally but then placed in a larger context. The resulting graph gives a global representation of the data that lends itself to analysis that cannot be done using only partial or local information. Typical analysis of a graph consists of identifying a subgraph with structure that holds meaning for the given data set. Structures that are important in numerous applications are given their own definitions in graph theory. One such structure is a *clique*.

A clique is a set of nodes in a graph for which every pair of nodes is directly linked by an edge. Less formally, a clique is a "tightly knit" set of nodes within a graph [8]. In a social network, for example, a clique is a set of people for which everyone knows everyone else. Cliques demonstrate behavior that we would most often associate with a "group." There is a high volume of connections within a clique, all of which are direct, and accordingly the communication cannot be stopped by removing a few

This dissertation follows the style of Mathematical Programming.

nodes. Because of these properties, cliques are almost always identified in a graph for the purpose of grouping nodes to simplify and better understand the graph.

While cliques demonstrate properties of an ideal group, the structure is not ideal for pinpointing groups in real-world networks [29, 52]. Cliques are not "noise" resistant. A single missing edge disqualifies a node from the clique. Cliques are typically too small to be of interest in real-world networks because they are so overly restrictive. In response to this, various structures have been defined that constitute *clique relaxations*. A clique relaxation is a structure that takes a property by which a clique can be defined, and it is defined to preserve this same property, but in a relaxed form. Low *diameter*, high *density*, and high *connectivity* are just some of the properties by which clique relaxations have been defined. The clique relaxations we study in this dissertation include *s-clique*, *s-club*, *s-plex*, *k-core*, *quasi-clique*, and *k-connected* subgraphs.

The focus of this dissertation is on studying clique relaxations in order to understand their basic properties and develop tools to identify them in a graph. Understanding the properties of a clique relaxation is essential to choosing the best clique relaxation model for an application. How vertices and edges are interpreted, as well as what structure constitutes a "group," depends heavily on the application. With a good understanding of the properties of each clique relaxation, the clique relaxation most resembling a desired structure can be chosen. Although most of the dissertation studies clique relaxations on general graphs, it is helpful to consider some of the specific networks to which they are indeed applied. While by no means a comprehensive list, the following networks help to demonstrate the ubiquity of graphs and the utility of clique relaxations in analyzing them.

1.1 Sample Networks

Within communications, among the most studied data sets are internet and phone call data. In an *internet graph*, vertices represent IP addresses while in a *web graph*,

vertices represent websites. Edges are determined by hyperlinks [17]. In *call graphs*, vertices represent telephone numbers and edges represent a phone call between two nodes during a specified interval of time [1]. Communication data sets are often so massive that data cannot be all stored in one location, much less analyzed in its entirety. The call graph studied in [1] constituted 20 days of telecommunications data and had over 290 million vertices and over 4 billion edges. Grouping can assist in organizing websites by topic or in organizing call information so that it can be stored in pieces, where most nodes are stored in a piece with the other nodes to which it relates. This enables researchers to test search engine design or to estimate the average number of calls made by a customer, using only a portion of the communications graph.

Within business and marketing, *stock market networks* and *social networks* are highly studied. The study of social networks was in fact the original setting for which the structure *clique* was defined [40]. Within a social network, vertices typically represent people, and an edge represents some sort of connection between people. Such a connection may be a relationship, a common interest, or any other sort of sociological link. Groups may help to divide a network into regions by interests so that marketing can be tailored for each region. In stock market graphs, vertices represent stocks, and two stocks are connected if they have high correlation during the time for which data is collected [16]. Stocks do not rise and fall in a vacuum, and groups help to identify a collection of stocks whose behavior seem to be linked. Basing decisions to buy or sell stock upon the behavior of a large collection is far more reliable than simply making a decision based on the behavior of one stock.

Within biology, *protein interaction networks* and *gene co-expression networks* are heavily studied. In a protein interaction network, vertices are proteins, and edges represent all interactions between proteins within a given cell over a given time frame [30]. A group, called a complex, represents a set of proteins that interact within the organism to carry out a specific biological process. Identifying protein

complexes and the specific biological process they carry out helps researchers identify how diseases overcome the body's defenses, with the hope of identifying treatments [54]. In gene co-expression networks, the vertices are genes, and edges represent genes that are co-expressed with high correlation [13]. Scientists can quantify the level to which a gene is expressed in a cell, that is, the level at which each gene is employed to produce proteins and functional RNA. Every cell has control over how it uses genetic information, which is what allows cells to have different structure and function. Groups within gene co-expression networks represent sets of genes that express themselves at high levels at the same time [13]. Clustering in these networks can also help reveal protein complexes by identifying sets of proteins built simultaneously with the purpose of interaction.

1.2 Restricted Graphs

Several subclasses of graphs with extra structure have been defined that mimic key features of real world networks. Such graphs place restrictions on how edges are distributed in the graph, either locally or globally. Classes of graphs that mimic key real-world networks often receive extra study. Tools can be developed to take advantage of the extra structure not necessarily exhibited by general graphs. One such class of graphs are *unit disk graphs*. Study of such graphs is motivated by communications networks [24]. In a unit disk graph, nodes consist of unit disks, and edges are completely determined by the distance between the centers of the disks. Assuming all nodes in the network have equal power, and there is no physical interference, the transmission distance of all nodes in a wireless communication network is well represented by disks of equal size. Links in such a graph may represent anything from communication capability to interference, making unit disk graphs amenable to modeling a variety of broadcasting problems.

By using distance to determine what edges exist, formations such as a node surrounded by six mutually independent nodes cannot exist in a unit disk graph.

The extra structure makes some problems significantly easier to solve than for general graphs. For instance, the maximum clique problem is well-known to be *NP-hard* for general graphs [31], which means the number of steps needed to solve the problem likely grows exponentially with the size of the problem. For unit disk graphs, the maximum clique problem can be solved in *polynomial time* [24], meaning the number of steps needed to solve the problem is bounded by a polynomial, where the size of the problem is the variable. Any problem where connection or interference is distance dependent is well modeled by unit disk graphs, and the extra structure can significantly help analysis.

One feature of a graph that is easily measured is its edge density. The density of a graph is the ratio of how many edges exist to how many edges can possibly exist [26]. Erdös and Rényi used edge density as a parameter to define a subclass of graphs known as *random graphs* [27, 28]. A random graph $G(n, p)$ is a graph with n vertices where the probability that an edge exists between any two vertices, i.e., its density, is $p \in [0, 1]$. By analyzing $G(n, p)$, important features such as the expected size of the largest clique can be established for any network where connections are distributed randomly according to a uniform distribution.

The distribution of edges in many real-world networks, however, is often not uniform. In web graphs, stock market graphs, and many biological networks, edges are distributed among vertices following a *power law degree distribution* [4, 11, 16]. If $X(k)$ denotes the number of vertices in a graph with degree k, a graph exhibits a power law distribution if $X(k)$ is proportional to $k^{-\alpha}$ where $\alpha \geq 1$ is a constant. This means *power law graphs*, also referred to as *scale-free graphs*, may have an enormous number of vertices, but are relatively sparse in number of edges. Further, the edges that do exist exhibit *preferential attachment*, meaning they tend to attach to vertices that already have a large number of neighbors. It would not be appropriate to study power law graphs as a uniform random graph. The probability of an edge existing at a vertex in a random graph is independent of other edges existing at the vertex and

so such graphs emulate power law graphs poorly. Like unit disk graphs, scale-free graphs have extra structure not necessarily exhibited by general graphs and have been studied extensively because they model so many real-world networks.

1.3 Outline

We will return to the networks described above in order to demonstrate the importance of each clique relaxation we study in this dissertation. First, however, we need a clear understanding of the definition and structure exhibited by the various clique relaxations. Key background information will be the focus of Section 2 before we define and explore the structure of each clique relaxation in Section 3. At that point we will return to some of our sample networks and restricted graph classes to demonstrate applications for which the structure of precisely one clique relaxation is well suited for grouping. The sections following will shift focus from the objects themselves to algorithms used to find them within a network. In particular, Section 4 will focus on the MAXIMUM QUASI-CLIQUE problem, where we will establish its complexity and explore an integer programming approach to solving the problem. The MAXIMUM 2-CLIQUE problem on unit disk graphs will be the focus of Section 5, where we will present an algorithm with a guaranteed 1/2-approximation ratio. Section 6 will conclude our study and present potential directions for future research.

2. BACKGROUND

This section gives the background information necessary for this dissertation. Definitions and background needed from graph theory are the focus of Section 2.1. The necessary information from complexity theory will be the focus of Section 2.2.

2.1 Graph Theory Definitions and Notation

For a basic introduction to general graph theory, see [26]. For an introduction to unit disk graphs, see [24]. We start with the most basic definitions for a graph. We then give definitions and notations for subsets, definitions and notations for individual vertices, and basic measurements for a graph. We conclude with some basic background on unit disk graphs.

We consider only finite, simple, undirected, and unweighted graphs denoted by $G = (V, E)$ where V is a set of vertices and E is a set of edges. If G consists of n vertices, we will typically label the vertices $V = \{1, \ldots, n\}$ and indicate the size with $|V| = n$. If G contains m edges, we will express this as $|E| = m$ and refer to a specific edge as $(i, j) \in E$ using the vertices i and j it connects. The number of vertices of a graph is often referred to as its *order*. When the graph under consideration is unambiguous, we may use n to indicate its order instead of $|V|$.

A graph $G' = (V', E')$ is a *subgraph* of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$. For a set $S \subseteq V$, we let $V - S$ or $V \setminus S$ represent the set of vertices in G with S removed. We let $G[S] = (S, E \cap (S \times S))$ denote the *induced subgraph* for a set of vertices $S \subseteq V$, where $S \times S = \{(i, j) \mid i, j \in S\}$ denotes a Cartesian product. It is obtained from G by deleting all vertices in $V - S$ and their incident edges. A graph $G = (V, E)$ is called *complete* if all its vertices are *adjacent*, i.e., if $\forall\, i, j \in V,\ i \neq j$, we have $(i, j) \in E$. By definition a *clique* C is a subset of vertices such that $G[C]$ is complete. A clique with n vertices is denoted by K_n. The *complement* $\bar{G}$ of $G = (V, E)$ is defined by $\bar{G} = (V, \bar{E})$, where $\bar{E}$ is such that $K_{|V|} = (V, E \cup \bar{E})$. A *bipartite graph* consists of

two independent sets P and Q such that all edges cross between vertices in P and Q. A complete bipartite graph with partitions consisting of p and q vertices is denoted $K_{p,q}$ and the complete bipartite graph $K_{1,n}$ is often referred to as a *star*. A *path* on n vertices, denoted by P_n, is an ordered tuple of vertices $(p_1, \ldots, p_n)$ such that consecutive vertices are connected by an edge. A graph where every pair of vertices share a path is called *connected*. A set of vertices S such that $\forall i \in V$, either $i \in S$ or $\exists j \in S$ *s.t.* $(i, j) \in E$, is called a *dominating set*. If S is a dominating set with $|S| = k$, we say the graph is *k-dominated*.

For a vertex $i \in V$, we refer to $N(i) = \{j \in V \mid (i, j) \in E\}$ as the *open neighborhood* of i. We refer to $N[i] = \{i\} \cup N(i)$ as the *closed neighborhood* of i. The value $deg_G(i) = |N(i)|$ is referred to as the *degree* of vertex i. For a pair of vertices $i, j \in V$, we let $d_G(i, j)$ denote the length of the shortest path between i and j in G. This value is referred to as the *distance* between vertices i and j in G. In order to not confuse it with Euclidean distance in the setting of unit disk graphs, we will sometimes refer to it as the *geodesic distance* of the vertices. By convention, the distance between two vertices that are not connected is infinity.

For any graph G, the *diameter* measures the furthest geodesic distance between any two vertices i and j in G and is denoted by $diam(G)$. The *edge density*, or simply, *density* of a graph is the ratio $|E|/\binom{|V|}{2}$, which represents the fraction of edges that exist over how many could possibly exist in G. We denote it with $\rho(G)$. The minimum and the maximum degree of a vertex in G are denoted by $\delta(G)$ and $\Delta(G)$, respectively. The *vertex connectivity* $\kappa(G)$ denotes the minimum number of vertices that must be removed to produce a disconnected graph. By convention a subgraph of one vertex will be considered disconnected.

Unit disk graphs are a subclass of graphs that can be realized as a set of equal radius disks in the Euclidean plane, where edges are completely determined by the distance between the centers of the disks. In the *intersection model*, two disks are connected by an edge if and only if the two disks of equal radius intersect. In the

containment model, two disks are connected by an edge if and only if each disk covers the center of the other. Not every graph can be represented as a unit disk graph. For instance, the graph $K_{1,7}$, which is a vertex surrounded by seven independent neighbors, is not a unit disk graph. For seven disks to intersect one central disk, at least two must intersect each other. The intersection and containment models of unit disk graphs are equivalent, meaning they specify the same subset of graphs from the collection of all graphs [24]. However, we will work exclusively with the containment model in this dissertation because we will be examining the 2-clique problem on unit disk graphs. All disks in a 2-clique on unit disk graphs must pairwise intersect under the containment model. This fact allows us to apply some previous knowledge from mathematics about piercing numbers on a set of pairwise intersecting disks, which will be crucial to our results.

2.2 Complexity Theory

For a comprehensive study of complexity theory, readers are referred to [31].

The theory of *NP-completeness* was designed to help classify *decision problems* based on difficulty. Decision problems answer with either "yes" or "no" to questions about a given object. As an example, consider CLIQUE problem.

Definition 2.2.1 *CLIQUE Problem: given a graph $G = (V, E)$ and a positive integer k, does there exist a clique of size $\geq k$ in G?*

Complexity theory classifies problems by how many steps it takes to solve the problem given the size of the object that is the input to the problem. The framework of *NP-completeness* within complexity theory separates decision problems with solutions that are easy to verify but difficult to compute from those that are easy to compute.

Definition 2.2.2 *A decision problem is in class P or polynomially solvable if an algorithm exists that can answer it correctly, where the number of steps is bounded by a fixed polynomial in the size of the input.*

An algorithm where the number of steps is bounded by a polynomial in the size of the input is frequently referred to as a *polynomial time algorithm.* Completing an algorithm at such a rate is often referred to as returning a solution *in polynomial time.*

Definition 2.2.3 *Suppose Q is a decision problem with input object O, and y is a string with information related to O and length bounded by a fixed polynomial in the size of O. Then the decision problem Q is in class NP if there exists an algorithm A such that:*

1. *Given a yes-instance of Q, there exists a string y that, when given as input to A, returns the answer "yes" to Q in polynomial time, both with respect to the size of y and accordingly O.*

2. *Given a no-instance of Q, no string y input into A will return the answer "yes."*

Essentially, a decision problem is in NP if an algorithm A exists that, if given a yes-instance O and information y, verifies the solution in polynomial time, and if given a no-instance O, will not return a false solution no matter what information y it is fed. Such an algorithm is called a *non-deterministic* polynomial (NP) time algorithm because it verifies a yes-instance in polynomial time when given the appropriate information y, but it cannot construct y in polynomial time.

It is clear that $P \subseteq NP$. This is because the polynomial time algorithm necessary for all problems in P can be used as the non-deterministic algorithm A required for a problem in NP. The algorithm for problems in P does not need to be handed the information y to solve a yes-instance of the problem in polynomial time but can construct it. The question remains open as to whether or not $P = NP$. One

attempt to help solve this problem, which also helps to further classify the difficulty of problems in NP, is the notion of NP-completeness.

The notion of NP-completeness was created to identify the most difficult problems in the class NP. Two problems can be compared for difficulty by a *polynomial time reduction* of one problem to the other.

Definition 2.2.4 *Given two decision problems Q_1 and Q_2, we say Q_1 is polynomial time reducible to Q_2 if there exists a polynomial time algorithm A that, given an input O to Q_1, constructs an input $A(O)$ to Q_2 such that:*

1. *$A(O)$ is polynomial in the size of O.*

2. *O is a yes-instance to Q_1 if and only if $A(O)$ is a yes-instance to Q_2.*

The contrapositive of the if and only if statement means the no-instances also correspond. Essentially a polynomial time reduction demonstrates how to, in polynomial time, transform any instance of Q_1 into an instance of Q_2 so that the solutions precisely correspond. This means that if Q_2 can be solved in polynomial time, so can Q_1, using this transformation. This gives the notion that Q_2 is at least as hard as Q_1.

The crucial first step to creating a class of NP-complete problems, which are the most difficult problems in the class NP, was to identify one problem to which every problem in the class NP could be reduced in polynomial time. This was done by Cook in 1971.

Theorem 2.2.1 *Every decision problem in the class NP is polynomial time reducible to the SATISFIABILITY problem.*

The definition of the SATISFIABILITY problem and the proof can be found in [31]. The result showed the following two classes of problems to be non-empty.

Definition 2.2.5 *A decision problem Q is NP-hard if every problem in NP is polynomial time reducible to Q. An NP-hard decision problem Q is said to be NP-complete if it is also in the class NP.*

Cook made proving problems to be NP-complete much simpler. With one problem Q_1 satisfying the definition of NP-completeness in hand, proving another problem Q_2 to be NP-complete requires the following two basic steps:

1. Show Q_2 is in NP.

2. Find a polynomial time reduction from Q_1 to Q_2.

The polynomial time reduction from Q_1 to Q_2 means Q_2 is at least as hard as Q_1, and since Q_1 is at least as hard as any problem in NP and both problems are in NP, they have equal levels of difficulty. This is the template by which most problems, including quasi-clique in this dissertation, are proven NP-complete.

The list of NP-complete problems has grown considerably due to this method. Each addition provides one more potential problem that might be used in a polynomial time reduction to help classify a future problem as NP-complete. For a list of many NP-complete problems, see [31].

A very crude generalization is that problems in P are "easy" or tractable, and problems that are NP-complete are difficult and intractable. Classifying problems by class is immensely practical. Identifying the basic class to which a problem belongs terminates all need to search for a fundamentally faster solution. Assuming $P \neq NP$, which has not been proven but is the consensus of most experts in complexity, proving a problem to be NP-complete means an exponential time solution is the best that can possibly be found.

3. DEFINITION AND STRUCTURE OF CLIQUE RELAXATIONS

"The whole is more than the sum of its parts."–Aristotle (384-322 BC)

This section formally introduces the clique relaxations that are the focus of this dissertation. We begin in Section 3.1 with the history of cliques and the subsequent creation of clique relaxations. We then review some of the basic terms of graph theory before formally defining our clique relaxations in Section 3.2. In Section 3.3 we give a taxonomy for creating and classifying clique relaxations to help organize this quickly growing field of study. In Section 3.4 and Section 3.5 we explore the computational and structural properties of the most utilized clique relaxations respectively. The goal of our analysis is to assist researchers in choosing an appropriate clique relaxation for grouping by its properties. If no clique relaxation exhibits the properties desired in a given application, the taxonomy in Section 3.3 demonstrates how to instead create a new clique relaxation. We analyze the property of the most utilized weak clique relaxation in Section 3.6 to show potential compromises in structure that might be made for improved computational properties. We finish off with a set of applications that demonstrate the need for each clique relaxation in Section 3.7 before concluding the section.

3.1 History

Initially proposed by Luce and Perry in 1949 [40] as a model of a cohesive subgroup within the context of social network analysis, a *clique* refers to a "tightly knit" set of elements (referred to as "actors"), in which every pair of actors shares some common attribute. The clique model possesses idealized cohesiveness properties, guaranteeing a high level of familiarity, reachability and robustness within a group of actors it describes. In graph-theoretic terms, a clique is a subset of vertices inducing a complete subgraph comprising all possible edges between its vertices. This

allows for perfect familiarity and reachability between members of a clique, since elements are all directly connected to each other. Moreover, removal of any element of a clique results in a slightly smaller clique and does not destroy the perfectly-tied structure of the group, making cliques ideal in terms of the robustness criterion as well. However, requiring all possible links to exist may prove to be rather restrictive for many applications, where interaction between members of the group needs not be direct and could be sufficiently accomplished through a number of intermediaries.

To overcome the clique's overly conservative nature, alternative graph-theoretic models have been introduced in the literature. The *s-clique* model, first introduced by Luce in 1950 [39], relaxes the requirement of direct interaction. Associating the number of intermediary links with the graph-theoretic notion of distance, the s-clique definition requires vertices within the group to be at most s-distant. Since intermediaries may not be part of the s-clique itself, Alba [3] proposed a definition of the so-called *sociometric clique of diameter* s, more commonly known as *s-club*, requiring the intermediary interactions to exist solely through elements belonging to the group. Star-like graphs possess a 2-club structure and suffer from a low familiarity and a high vulnerability to the incident of a hub dysfunction. This drew the attention towards the necessity in some applications to consider clique-like models emphasizing high level of familiarity and robustness. In particular, Barnes [12] adopted the notion of *edge density* to address familiarity within a group. More recently this concept was formalized under the so-called γ-*quasi-clique* model, which ensures a certain minimum ratio γ of the number of existing links to the maximum possible number of links within the group. Seidman [51] argues that edge density is a rather averaging property and may result in a group with highly cohesive regions, where vertices present a large number of direct interactions with their neighbors, coupled with highly sparse regions, where vertices rely more on indirect interactions with the rest of the group. His observation led to defining the k-*core*, a concept restricting the minimum number of direct links an element must have with the rest of the cluster.

While a k-core guarantees a certain minimum number k of neighbors within the group, the number of non-neighbors within the group may still be much higher than k, indicating a low level of familiarity within the group relative to its size. In 1978, Seidman and Foster [52] proposed the notion of s-plex, controlling the number of non-neighbors that elements within the group are allowed to have. In addition to high level of familiarity within the group ensured by its definition for low values of s, s-plex fares well in terms of robustness expressed in terms of *vertex connectivity*, which is the minimum number of vertices that need to be removed in order to disconnect the graph. The vertex connectivity has recently been linked to social cohesion in social network analysis literature [46], where it quickly became a central concept referred to as *structural cohesion*. Thus, the related notion of *k-connected subgraph*, which ensures that the group remains connected unless more than k elements are deleted, can be used as another natural model of a cohesive subgroup or a cluster. Yet another model of a cluster was introduced recently in a study of protein interaction networks [59], where an *s-defective clique*, which differs from a clique by at most s missing edges, was used to predict protein interactions. Some of the more recent cluster models proposed in the literature appear to be "hybrids" enforcing a mix of desired group properties. For instance, the *(λ, γ)-quasiclique* model [18] ensures minimum levels of connections and direct interactions to be met by the group by setting lower bound λ on the fraction of γ-quasi-clique members that each vertex must neighbor.

Note that all concepts mentioned as alternatives to clique in the previous paragraph were defined using a parameter, s; k; γ; or λ. Moreover, for $s = 1$ ($s = 0$ for an s-defective clique); $k = n - 1$; $\gamma = 1$; and $\lambda = 1$, where n is the number of vertices in the group being defined, each of the above definitions describes a clique. Hence, defining each of these concepts for an *arbitrary* value of the corresponding parameter yields a *generalization* of the notion of a clique, since it includes the clique definition as a special case. On the other hand, defining any of the concepts above for a *fixed*

value of the corresponding parameter, i.e., positive integer s or $k > 1$; real γ and $\lambda \in (0, 1)$, provides what we call a *clique relaxation*[1]. Using this term is justified by the fact that fixing the corresponding parameter as just described, each definition is less restrictive than that of a clique (of size at least k for the definitions involving parameter k).

The described clique relaxation concepts, as well as numerous other similar definitions have emerged in an ad-hoc and somewhat spontaneous fashion and were motivated by cluster-detection problems arising in a wide variety of applications. Moreover, some clique relaxation models have been reinvented under different nomenclature. Despite the obvious practical importance of these models, little work has been done towards establishing theoretical and algorithmic foundations for studying the clique relaxations in a systematic fashion. As a result, applied researchers seeking an appropriate model of a cluster in their application of interest may quickly get overwhelmed by the wide range of models available in the literature. This section aims to start filling this gap by proposing a tentative taxonomy classifying the previously defined clique relaxations under a unified framework. More specifically, we build on the elementary graph-theoretic properties of cliques to provide a hierarchically ordered classification of clique relaxation models. We complement the taxonomy by comparing the so-called first-order clique relaxations, defined later on, on the properties they guarantee. This exercise provides solid grounds for a more comprehensive understanding of the relations among the various known clique relaxation models, which could serve as an essential guide for practitioners in selecting a cluster model most suited for a particular application. Moreover, the proposed taxonomy also uncovers potential horizons for developing and analyzing new clique relaxation models.

[1]This is in analogy to how mixed integer programming (MIP) is a generalization of integer programming (IP), however, a linear programming (LP) relaxation of an IP (which is a special case of MIP with all variables being continuous) cannot be called a generalization of IP, even though an optimal solution of the corresponding IP is feasible and sometimes even optimal for the LP relaxation.

3.2 Definitions and Notations

Recall that a *clique* C is defined to be subset of vertices $C \in V$ such that the induced subgraph $G[C]$ is complete. The size of the maximum clique in G is referred to as the *clique number*, denoted by $\omega(G)$. A subset of vertices I is called an *independent set* if the corresponding induced subgraph $G[I]$ has no edges. The *independence number* $\alpha(G)$ is the largest size of an independent set in G. It is easy to see I is an independent set in G if and only if I is a clique in $\bar{G}$.

We now formally define the well known clique relaxation models, which are central for this study and were already mentioned in the previous section. We assume that the constants s and k are positive integers and $\lambda, \gamma \in (0, 1]$ are reals. In all definitions below, we refer to a subset of vertices S in $G = (V, E)$.

Definition 3.2.1 (s-clique) *S is an s-clique if $d_G(v, v') \leq s$, for any $v, v' \in S$.*

Definition 3.2.2 (s-club) *S is an s-club if $diam(G[S]) \leq s$.*

Definition 3.2.3 (s-plex) *S is an s-plex if $\delta(G[S]) \geq |S| - s$.*

Definition 3.2.4 (s-defective clique) *S is an s-defective clique if $G[S]$ has at least $\binom{|S|}{2} - s$ edges.*

Definition 3.2.5 (k-core) *S is a k-core if $\delta(G[S]) \geq k$.*

Definition 3.2.6 (k-connected set) *S is a k-connected set if $\kappa(G[S]) \geq k$.*

Definition 3.2.7 (γ-quasi-clique) *S is a γ-quasi-clique if $\rho(G[S]) \geq \gamma$.*

Definition 3.2.8 ((λ, γ)-quasi-clique) *S is a (λ, γ)-quasi-clique if $\delta(G[S]) \geq \lambda(|S|-1)$ and $\rho(G[S]) \geq \gamma$.*

Definition 3.2.9 (k-robust s-club) *S is a k-robust s-club if $diam(G[S \setminus S']) \leq s$ for any $S' \subset S$ such that $|S'| \leq k$.*

It should be noted that, in general, depending on the choice of k and a graph instance G, a k-core or a k-connected set may not exist in G. This observation, together with the fact that a k-core can be easily computed by recursively removing all vertices of degree less than k from the graph, has led to the introduction of the notion of *graph degeneracy* based on the concept of a k-core. Namely, a graph is called *d-degenerate* if it does not contain a nonempty k-core for $k > d$. The *degeneracy* of G is the smallest d for which G is d-degenerate, which is the same as the largest k for which G has a k-core.

3.3 A Taxonomy of Clique Relaxation Models

One can observe that most of the elementary graph concepts, such as degree, distance, diameter, density, connectivity, and domination, can be used to derive alternative, equivalent definitions of a clique. We will state this observation formally in the following proposition, which is trivial to verify.

Proposition 3.3.1 *A subset of vertices C is a clique in G if and only if one of the following conditions hold:*

a) $d_G(v, v') = 1$, *for any* $v, v' \in C$;

b) $diam(G[C]) = 1$;

c) $D = \{v\}$ *is a dominating set in* $G[C]$, *for any* $v \in C$;

d) $\delta(G[C]) = |C| - 1$;

e) $\rho(G[C]) = 1$;

f) $\kappa(G[C]) = |C| - 1$.

In the remainder of the paper, we refer to the specified conditions as *elementary clique defining properties*. These properties are summarized in Table 3.1, together

with the corresponding graph concepts defining each property. The table is split into two parts, with the first part corresponding to the parameters whose value is set to the lowest possible value in the clique definition (distance, diameter, size of a set guaranteeing domination), and the second part containing the parameters required to have the highest possible value for the set of a *given size* (degree, density, connectivity).

Table 3.1
Alternative clique definitions based on elementary clique-defining properties.

Parameter	**Definition**
Distance	Vertices are distance *one* away from each other
Diameter	Vertices induce a subgraph of diameter *one*
Domination	Every *one* vertex forms a dominating set
Degree	Each vertex is connected to *all* vertices
Density	Vertices induce a subgraph that has *all* possible edges
Connectivity	*All* vertices need to be removed to obtain a disconnected induced subgraph

Aiming to derive a *minimal* set of *simple* rules based on the elementary clique-defining properties that would allow us to obtain the known clique relaxation models in a systematic fashion, we examine the relation of Definitions 3.2.1–3.2.9 to the alternative clique definitions summarized in Table 3.1. It becomes apparent that each of the defined clique relaxation models essentially relaxes at least one of the elementary clique-defining properties according to some simple rules that can be classified into two broad categories. Namely, some relaxations are created by providing an upper bound on the extent to which an elementary clique defining property is allowed to be violated, while others aim to ensure the presence of an elementary clique defining property that characterizes a clique of a *given minimum size*.

3.3.1 Restricting Violation of an Elementary Clique Defining Property

Increasing a parameter that has the lowest possible value in a clique. In some cases, we obtain a clique relaxation model by increasing a parameter that was set to the lowest possible value in an alternative clique definition, as in the first three rows of Table 3.1. Such models are created by naturally replacing *one* in one of the elementary clique-defining properties with *(at most) s*. In particular, instead of requiring the (upper bound on the) diameter of the induced subgraph to be equal to *one*, an s-club relaxes this requirement to allow a diameter *at most* s. Similarly, by replacing *one* with *at most* s in the elementary clique-defining properties based on distance and domination, we obtain definitions of s-clique and s-plex, respectively. In the case of s-plex, we use the fact that S is an s-plex if and only if any subset of s vertices from S forms a dominating set in $G[S]$ [52].

Reducing a parameter that has the highest possible value in a clique of a given size. Note that, while we were able to define s-plex by relaxing an upper bound on the number of vertices ensuring domination, the original definition of s-plex was based on restricting the number of non-neighbors that a vertex can have within the group. This definition naturally corresponds to allowing, for every vertex, s exceptions (including the vertex itself) in the degree-based definition of a clique. Namely, we just replace *all* by *all but* s in the degree-based definition of a clique to obtain the s-plex definition. Similarly, the density-based clique definition yields the s-defective clique model. By applying the same logic to the clique definition based on connectivity, we obtain a new clique relaxation model, which we propose to call an *s-bond*.

Definition 3.3.1 (s-bond) *A subset of vertices is called an s-bond if $\kappa(G[S]) \geq |S| - s$.*

The s-bond model with a small value of $s > 1$ may prove to be a useful alternative to a clique (which can be equivalently defined as a 1-bond) in applications emphasizing

the robustness of a cluster. Moreover, it has some computational advantages, which will be discussed later, over similar models, such as a k-connected set.

3.3.2 Ensuring the Presence of an Elementary Clique Defining Property

In other cases, we replace the overly restrictive requirement of a clique definition to have the highest possible value for a parameter (assuming that the size of a set is given) by, instead, imposing a fixed lower bound on that parameter. In such cases, we replace *all* in one of the elementary clique-defining properties with *(at least) k*. For example, a k-core, does not require each vertex to be connected to *all*, but to *at least* k other vertices. Likewise, we can obtain the definition of a k-connected set by relaxing the connectivity-based definition of a clique in the same fashion. Similarly, we could define an analogous concept corresponding to the density-based definition of a clique. We could call a subset of vertices a *k-edge set* if it induces a subgraph with at least k edges. We are not aware of any studies of this concept in the literature or its potential applications; therefore, we do not investigate it any further in this dissertation. Note that, unlike the other two types of relaxations, the clique relaxation models based on setting a fixed lower bound on a parameter can potentially result in degeneracy (i.e., a structure of this type may be empty if the value of k is set too high for a given graph). Hence, we will also refer to such models as *degeneracy-invoking*. The two methods by which to define a clique relaxation are part of our proposed classification of clique relaxation models, pictured in Fig. 3.1.

3.3.3 Absolute and Relative Relaxations

As suggested by the example of the γ-quasi-clique, size-relative or, simply, *relative* clique relaxations is another category of models that needs to be considered. Thus, it makes sense to refer to the above three categories that use the absolute parameter values (s or k) as *absolute*. We can generate the relative clique relaxation models

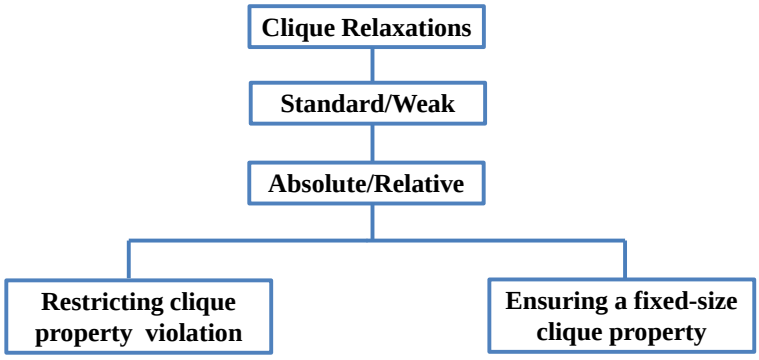

Fig. 3.1. The proposed classification of clique relaxation models.

from the absolute models by replacing s and k by $\alpha|S|$ and $\gamma|S|$ ($\gamma\binom{|S|}{2}$ in case of density), respectively, where $0 \leq \alpha, \gamma \leq 1$. While the γ-quasi-clique is, perhaps, the most well known in this category, other relative size-dependent clique generalizations can be defined similarly. For instance, the relative version of s-club would guarantee the induced subgraph $G[S]$ to have a diameter at most $\alpha|S|$. Similarly, one could ensure that at least all but $\alpha|S|$ vertices need to be removed to disconnect the induced subgraph.

3.3.4 Standard and Weak Relaxations

In the definitions of most of the clique relaxation models discussed above (s-clique being the only exception), we required the relaxed clique defining properties to be satisfied within the *induced subgraph*. However, as the example of s-clique suggests, in some cases we could require the same property to be satisfied within the *original graph* instead of the induced subgraph. In particular, this can be done in the situations involving the elementary clique defining properties based on distance and connectivity, both of which can be defined through paths. In the case of connectivity, this can be done using Menger's theorem asserting that a graph is k-connected if and only if there are at least k vertex-independent paths (i.e., paths with no common

internal vertex) between any two of its vertices. Thus, by requiring the conditions on pairwise distances and connectivity to hold in the whole graph rather than the induced subgraph corresponding to a subset of vertices defining a cluster, we allow the paths in the corresponding definitions to pass through vertices outside of the cluster. As a result, we obtain a relaxation with weaker cohesiveness properties. We will refer to such relaxations as *weak*, while the relaxations that require the relaxed clique defining property to be satisfied in the induced subgraph will be called *standard*. For example, s-club is a standard relaxation, while s-clique is its weak counterpart and could be alternatively called *weak s-club*. Similarly, we could define a *weak k-connected set* as a subset of vertices such that there are at least k vertex-independent paths between any two of its vertices in the original graph.

3.3.5 Order of a Clique Relaxation

So far, we have considered examples of clique relaxations, obtained by relaxing only one aspect of the clique definition. Calling the clique itself a zero-order clique relaxation, the aforementioned clique-like objects are referred to as first-order clique relaxations. Higher order clique relaxations can be obtained by relaxing more than one aspect of the clique definition. The second-order relaxations would correspond to relaxing two elementary clique-defining properties at the same time. For instance, the (λ, γ)-quasiclique based on both degree and density, is a second-order relaxation. Similarly, higher-order relaxations can be defined by relaxing more than two elementary clique-defining properties at a time. While any pair of properties can be enforced simultaneously in order to define a second-order model, in some cases requiring an extra property may be redundant. For example, as we will discuss in Section 3.5, an s-plex usually has a low diameter and a high connectivity to start with, hence it makes little sense to combine it with diameter or connectivity-based relaxations.

Robust higher order relaxations. While a higher-order relaxation can be created by enforcing several relaxed clique-defining properties simultaneously, one of the properties, connectivity, can also be *embedded* into a definition of a clique relaxation. As an example, a k-robust s-club S can be viewed as a second-order clique relaxation structure defined by embedding k-connectivity into the definition of an s-club [56]. Unlike its *simple* second-order counterpart, which would be defined as a subset of vertices S such that $\kappa(G[S]) \geq k$ and $diam(G[S]) \leq s$ and could be called *k-connected s-club*, the k-robust s-club requires that not only does the s-club S induce a k-connected subgraph, but also that removal of up to k vertices still preserves the s-club property. The property of *k-robustness*, or, alternatively, *k-heredity* is embedded within the structure defined by other properties involved in the definition of a robust higher order relaxation, which makes it essentially different from the simple higher order relaxations that combine multiple properties in a straightforward fashion.

3.4 Optimization Considerations

In most application scenarios dealing with clique relaxation models, one is interested in computing *large* clusters of a certain type. While typically multiple large clusters, not necessarily largest possible, are of practical interest, the maximum size of a clique relaxation of a given kind quantifies the *global cohesiveness* of the analyzed network in terms of the considered clique relaxation model of a cohesive subgroup. Besides, it provides the tight upper bound on the size of clusters of the considered type that exist in the network, and hence facilitates computing such clusters. Thus, in the remainder of this section, we are interested in issues associated with the corresponding optimization problems. In particular, we are interested in considerations that may facilitate the process of selecting computational techniques that would be appropriate for solving the corresponding optimization problems for different types of clique relaxation models.

First, let us formally define the general optimization problem for a clique relaxation model. Let RELAXED CLIQUE refer to a subset of vertices that satisfies the definition of an arbitrary clique relaxation concept. The following definitions are general and can be adopted for a particular clique relaxation model by replacing the term RELAXED CLIQUE with the name of this model (i.e., s-club, s-plex, etc.).

Definition 3.4.1 *A subset of vertices S is called a maximal* RELAXED CLIQUE *if it is a* RELAXED CLIQUE *and is not a proper subset of a larger* RELAXED CLIQUE*.*

Definition 3.4.2 *A subset of vertices S is called a maximum* RELAXED CLIQUE *if there is no larger* RELAXED CLIQUE *in the graph. The maximum* RELAXED CLIQUE *problem asks to compute a maximum* RELAXED CLIQUE *in the graph, and the size of a maximum* RELAXED CLIQUE *is called the* RELAXED CLIQUE *number.*

We will use the following notations for specific RELAXED CLIQUE numbers: $\omega_s(G)$ is the s-clique number; $\bar{\omega}_s(G)$ is the s-club number; $\tilde{\omega}_s(G)$ is the s-plex number; $\omega'_k(G)$ is the k-core number; and $\check{\omega}_k(G)$ is the k-connectedness number.

Most of the discussion in this section is centered around the concept of *heredity*, which could be thought of as a dynamic property, since it describes the characteristics of a graph undergoing change, i.e., vertex addition or removal. Heredity is defined with respect to a graph property Π and is formally introduced next.

Definition 3.4.3 (Heredity) *A graph property* Π *is said to be* hereditary on induced subgraphs*, if for any graph G with property* Π *the deletion of any subset of vertices does not produce a graph violating* Π*.*

The presence of heredity on induced subgraphs implies certain properties that may help streamlining the study of the corresponding optimization problems. In particular, it turns establishing the computational intractability of the problem into a simple exercise of checking several basic facts about the property Π. Namely, a property Π is called *nontrivial* if it is true for a single-vertex graph and is not satisfied by every

graph, and Π is called *interesting* if there are arbitrarily large graphs satisfying Π. The following general complexity result is due to Yannakakis [58].

Theorem 3.4.1 (Yannakakis, 1978) *The problem of finding the largest-order induced subgraph not violating property* Π *that is nontrivial, interesting and hereditary on induced subgraphs is NP-hard.*

In addition, heredity on induced subgraphs is the foundational property for some of the most successful combinatorial algorithms for the maximum clique problem [20, 48], which can be easily generalized to solve any other maximum RELAXED CLIQUE problem based on relaxed clique defining properties that are hereditary on induced subgraphs. By analyzing the taxonomy introduced in Section 3.3, we can conclude that the only models that fall within this category are the standard, absolute clique relaxation models obtained by restricting violation of a clique-defining property and based on reducing a parameter that has the highest possible value in a clique of a given size. These are the models described in the second paragraph of subsection 3.3.1, namely, s-plex, s-defective clique, and s-bond. Hence, the corresponding optimization problems are NP-hard and can be solved by adopting the combinatorial algorithms proposed in [20, 48]. The presence of the heredity property also suggests that these problems are good candidates for solving by methods based on polyhedral combinatorics, as was already demonstrated for two of these models, s-plex and s-defective clique, in [9] and [53], respectively. Moreover, computing maximal RELAXED CLIQUE is trivial in this case, as maximality is guaranteed whenever the current solution cannot be expanded by adding any single vertex from outside.

Even though the properties defining other first-order clique relaxation models do not posses heredity, they have closely related characterizations that can also be utilized in designing solution methods. We propose to define these dynamic properties of *weak heredity, quasi-heredity*, and *k-heredity* as follows.

Definition 3.4.4 (Weak heredity) *A graph property* Π *is said to be* weakly hereditary*, if for any graph* $G = (V, E)$ *with property* Π*, all subsets of* V *demonstrate the property* Π *in* G*.*

Definition 3.4.5 (Quasi-heredity) *A graph property* Π *is said to be* quasi-hereditary*, if for any graph* $G = (V, E)$ *with property* Π *and for any size* $0 \leq r < |V|$*, there exists some subset* $R \subset S$ *with* $|R| = r$*, such that* $G[S \setminus R]$ *demonstrates property* Π*.*

Definition 3.4.6 (k-Heredity) *A graph property* Π *is said to be* k-hereditary on induced subgraphs*, if for any graph* G *with property* Π *the deletion of any subset of vertices with up to* k *vertices does not produce a graph violating* Π*.*

Note that weak heredity considers whether a certain property is still applicable for all subsets in the original graph, as opposed to heredity on the induced subgraph. On the other hand, quasi-heredity essentially requires the existence of a sequence of vertices such that their removal in this sequence preserves, at every step of the vertex removal process, the property in the remaining subgraph. However, property Π may not exist for every subset R of vertices removed from S. Also, observe that heredity implies both weak heredity and quasi-heredity, whereas the latter two do not have any definitive relation.

The weak heredity property holds for s-cliques and weak k-connected sets, both of which are weak clique relaxation models. The weak heredity property provides significant computational advantages due to the fact that the corresponding clique relaxation structures can be reduced to cliques in auxiliary graphs. In the case of s-clique, the auxiliary graph is given by *power graph*. Given a graph $G = (V, E)$, its t-th power graph $G^t = (V, E^t)$ has the same set of vertices V and the set of edges E^t that connects pairs of vertices that are distance at most t from each other in G. Obviously, S is an s-clique in G if and only if S is a clique in G^s. Similarly, for the weak k-connected set, we can define an auxiliary graph $G(k) = (V, E(k))$, where $(v_1, v_2) \in E(k)$ if and only if there are at least k vertex-independent paths between

v_1 and v_2 in G. Then, again, S is a weak k-connected set in G if and only if S is a clique in $G(k)$. Thus, the numerous algorithms developed for the maximum clique problem, can be directly applied to auxiliary graphs in order to solve the optimization problems dealing with weak clique relaxations.

The definition of quasi-heredity was motivated by observation that this property holds for the γ-quasiclique model, since the iterative removal of the lowest degree vertex will preserve at least the same density in the induced subgraphs at every step. The presence of this property suggests that developing heuristics based on greedy sequencing of vertices may prove effective in practice. Finally, the k-heredity property is what we enforce in robust higher-order clique relaxations discussed in the previous section. Not surprisingly, the first robust second-order relaxation studied involves an s-club, which does not posses any type of heredity considered. It is also known that computing a maximal s-club is NP-hard [44].

3.5 Cohesiveness Properties of Standard First-Order Clique Relaxation Models

The hierarchical classification proposed in Section 3.3 allows the definition of a wide variety of relaxations with different levels of proximity to the clique structure. However, care must be vested while investing in higher order characterizations. This in fact requires a detailed understanding of the properties of lower-order relaxations. For instance, it may not be worth restricting an additional property for some first-order relaxation if its structure automatically guarantees good bounds on the desired property. This observation motivates the current section, in which we provide a comprehensive study of the various structural properties guaranteed by well-known first-order relaxations. Not only can this help discern useful higher-order characterizations, but it also assists practitioners in appreciating what each of the currently developed models offer in terms of group structure.

This section concentrates on well-known first-order clique relaxations defined within the induced subgraph, notably s-club, s-plex, k-core, γ-quasi-clique and k-

connected subgraph. We treat these models as the *canonical* models for the corresponding parameters used to formulate the elementary clique-defining properties. For instance, s-club is the canonical clique relaxation model for diameter. All of the canonical models, except for the quasi-clique, are absolute clique relaxation models. We selected the quasi-clique over s-defective clique to represent a density-based relaxation in this study due to two reasons. First, the concept of density is traditionally discussed as a relative measure by definition; and second, γ-quasi-clique is by far more widely represented in the literature. Table 3.2 associates every canonical relaxation with its defining property on the diagonal. Note that the distance property for standard clique relaxations is equivalent to the same property for the diameter, since we limit the analysis to induced subgraphs. Furthermore, *dominating set* refers to the smallest size for which any subset of vertices is guaranteed to dominate the entire set. In an attempt to fully understand the behavior of these characterizations, this section aims at exploring the best bounds that could be ensured for each of the non-defining relaxation properties.

Table 3.2
Clique relaxation defining properties.

$S \subset V$	Diameter	Dominating Set	Degree	Density	Connectivity
Clique	"one"	"one"	"all"	"one"	"all"
s-club	"at most s"				
s-plex		"at least s "			
k-core			"at least k"		
γ-quasiclique				"at least γ"	
k-connected					"at least k"

3.5.1 Bounds on Diameter

By definition, s-clubs ensure that any two vertices in the group are no more than distance s apart in the induced subgraph $G[S]$. This subsection presents bounding results on the diameter property for the remaining canonical clique relaxations. Note that we use distance and diameter interchangeably, since we only consider induced subgraphs. Any clique relaxation consisting of pieces from two or more components $G^\dagger$ and $G^\ddagger$ in G clearly results in an unbounded distance between any two vertices v_1, v_2, such that $v_1 \in G^\dagger$ and $v_2 \in G^\dagger$, implying an unbounded diameter. To rule out such situations, we assume that clique relaxations are connected. We next present diameter bounds for each of the s-plex, k-core, γ-quasiclique, and the k-connected clique relaxations.

s-Plex model. We first note that *every connected s-plex is also an s-club.* The familiarity requirements are stringent enough that they result in low diameter. To see this, consider the shortest path between the two most distant vertices v_1 and v_2 in an s-plex. If this path contained two neighbors of v_1, a shorter path could have been obtained by connecting v_1 directly to its second neighbor in the sequence. By contradiction, the shortest path contains at most *one* neighbor of v_1. Now, since v_1 has at most $s-1$ non-neighbors in an s-plex, the path between v_1 and v_2 is at worst of length s, consisting of one neighbor of v_1 and $s-1$ non-neighbors of v_1, including v_2. As a result, the largest distance in the worst case is s yielding a diameter s. This proves our claim that every connected s-plex is an s-club. As a result, any connected s-plex ensures a diameter, and therefore a distance, of no more than s within the subgraph. This bound is sharp because a path on $s+1$ vertices is an s-plex of diameter s. Moreover, if an s-plex has size n, and $s < \frac{n}{2}$, the s-plex will only have diameter two, since every pair of vertices will have a common neighbor.

k-Core model. We cannot claim a fixed upper bound on the distance between vertices in a k-core, even if it is connected. We back this claim by considering the following construction: consider a clique K_k of size k and two vertices $a, b \notin K_k$ such that they are each connected to all elements within the clique. The resulting graph denoted by G is obviously a k-core since each vertex is connected to at least k other elements in G. Also consider a $(k-1)$-core G' created in a similar fashion, labeling the two outer vertices a' and b'. Between any two copies of G, one could place an arbitrarily large number of subgraphs G', connecting a' of the first copy of G' with b of the first copy of G, a' with b' for all intermediate copies of G', and b' of the last copy of G' with a of the second copy of G (refer to Fig. 3.2). We denote this construction by G_1. Note that G_1 is a k-core allowing for arbitrarily large distances between any two vertices. While no absolute bound on the distance may be claimed for a k-core model, a tight bound may be generated as a function of the k-core size $\omega'_k(G)$.

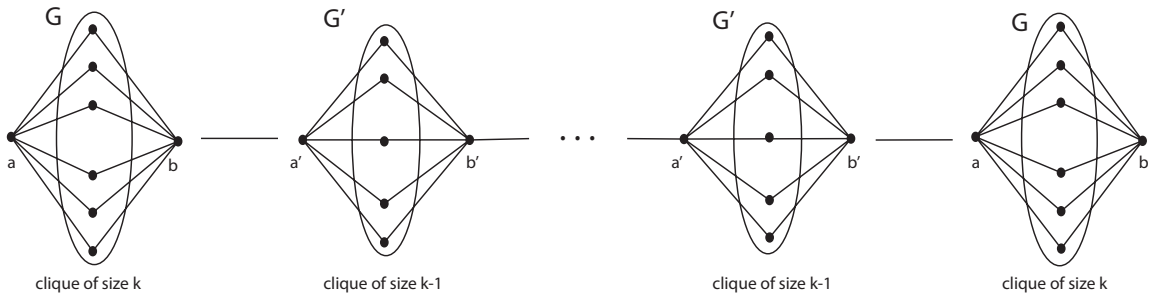

Fig. 3.2. Illustration of G_1 construction.

Lemma 3.5.1 *If there exists vertices u, ν in a connected k-core such that $d_G(u, \nu) \geq 3d$, for $d \geq 1$, then the size of the k-core $\omega'_k(G) \geq (d+1)(k+1)$.*

Proof Label the vertices along the shortest path connecting u and ν as $u = x_0, x_1, \ldots, x_{3d} = \nu$ and consider the subset $S = \{x_0, x_3, \ldots, x_{3d}\}$, $|S| = d+1$. Each

$x_i \in S$ must have k neighbors to be a member of the k-core. No vertex $x_i \in S$ can be connected to $\{x_j\} \cup N(x_j)$, for any $x_j \in S$ with $j \neq i$, or else the considered path would not have length $3d$. This means that each vertex in S is connected to k distinct vertices outside S, themselves not connected to any other vertex within S. Each of the $(d+1)$ vertices in S along with its corresponding k neighbors represent a set of at least $k+1$ distinct vertices that must be in the k-core. Thus $\omega'_k(G) \geq (d+1)(k+1)$. ■

Next, we present a proposition providing a tight upper bound on pairwise distance within G using construction G_1.

Proposition 3.5.1 *If $\omega'_k(G)$ is the size of the k-core, the maximum distance between vertices is given by the function:*

$$d'_k = \begin{cases} 1 & \text{if } \omega'_k(G) = k+1 \\ 2 & \text{if } \omega'_k(G) = k+2 \\ 3c & \text{if } \omega'_k(G) = (c+1)(k+1) \ \& \ c \geq 1 \\ 3c+1 & \text{if } \omega'_k(G) = (c+1)(k+1)+1 \ \& \ c \geq 1 \\ 3c+2 & \text{if } (c+1)(k+1)+2 \leq \omega'_k(G) < (c+2)(k+1) \ \& \ c \geq 0. \end{cases} \tag{3.1}$$

Also, d'_k provides a tight bound.

Proof By Lemma 3.5.1, we know the distance d between any two vertices satisfies $d \leq 3c$ when $\omega'_k(G) = (c+1)(k+1)$. Clearly the bounds $3c+1$ when $\omega'_k(G) = (c+1)(k+1)+1$ and $3c+2$ when $\omega'_k(G) = (c+1)(k+1)+2$ hold since by adding one vertex to a graph, the distance can only increase by one. The first two bounds are special cases. A k-core can only have distance one when it has $k+1$ vertices since it must be a clique, and adding one vertex can only increase the maximum distance to two.

To show that these bounds are tight, consider our construction G_1. The portion of the graph we labeled G alone shows that a k-core of $k+2$ vertices can achieve

distance two. The bound $3c+2$ is obtained for $\omega'_k(G) = (c+1)(k+1)+2$ by G_1 where we include two copies of G, unless $c = 0$, and $c-1$ copies of G' when $c \geq 1$, as given in the description of G_1. The bound $3c+1$ is achieved for $\omega'_k(G) = (c+1)(k+1)+1$ by taking the previous exact construction and contracting the vertex a in the second copy of G with its immediately preceding vertex. The bound $3c$ is achieved for $\omega'_k(G) = (c+1)(k+1)$ by taking the same construction with the edge between a and its preceding vertex contracted but then also removing vertex b from the second copy of G. ■

γ-Quasi-clique model. Similarly to the k-core model, no guarantee can be made about the distance between vertices in a γ-quasi-clique. To show this, we consider construction G_2 in Fig. 3.3, for which we denote the maximum distance between its vertices by L, such that L can be chosen arbitrarily large. Consider a clique of size $|V|$ missing only one edge between vertices x_0 and x_2 in V. Attach to the graph a sequence of $L-2$ vertices and build a single path between vertex x_2 and the added sequence. The number of edges in G_2 is the total number of edges in the clique at the exception of edge (x_0, x_2) plus the added path length $L-2$, i.e., $\binom{|V|}{2} - 1 + (L-2)$. With the total number of vertices in G_2 reaching $|V|+(L-2)$, the possible number of edges among vertices in G_2 grows up to $\binom{|V|+L-2}{2}$. Assuming at least one additional vertex is added to the original clique, i.e., $L \geq 3$, it is easy to see that, for fixed $0 \leq \gamma < 1$,

$$\gamma\binom{|V|+L-2}{2} \leq \binom{|V|}{2} \leq \binom{|V|}{2} - 1 + (L-2) \tag{3.2}$$

holds for large enough $|V|$. Therefore, considering the outer inequality, we observe that G_2 is a γ-quasiclique. Note that the longest path in G_2 corresponds to the path between vertex x_0 and the tip of the added vertex sequence, whose length is L. Since L can be chosen to be arbitrarily large, no fixed bounds can be set on the pairwise distances between vertices in the γ-quasiclique. While no absolute

bound on the distance may be claimed for a γ-quasiclique model, we could however generate a tight upper bound as a function of the γ-quasiclique size $\omega_\gamma(G)$, as shown in Proposition 3.5.2.

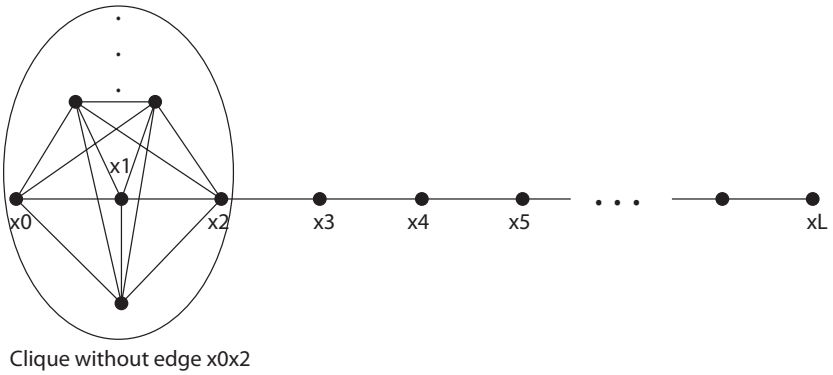

Fig. 3.3. Illustration of G_2 construction.

Proposition 3.5.2 *If $\omega_\gamma(G)$ is the size of the γ-quasiclique, the maximum distance between vertices is given by the function:*

$$d_\gamma = \left\lfloor \omega_\gamma(G) + \frac{1}{2} - \sqrt{\gamma.\omega_\gamma^2(G) - (2+\gamma)\omega_\gamma(G) + \frac{17}{4}} \right\rfloor . \tag{3.3}$$

Also, d_γ provides a tight bound.

Proof Suppose Q is a γ-quasi-clique of size $\omega_\gamma(G)$ in G and let d denote the longest distance between any two of its vertices. Label the sequence of vertices along the corresponding path in order, $\{x_i\}_{i=0,...,d}$, i.e., $x_0, x_1, ..., x_d$. Note that for all vertices $x_0, \ldots, x_d$ along the path, $N(x_i) \cap N(x_j) = \emptyset$ unless $i \in \{j-2, \ldots, j+2\}$. Indeed, if vertices more than a distance two apart on this path shared a common neighbor, the path could be shortened. However this would contradict the assumption that this path is the shortest between x_0 and x_d. If vertex y connects to x_i, it cannot connect to x_{i-3}.

Now with this remark, we would like to transform $G[Q]$ into a subgraph of our

earlier construction G_2, as illustrated in Fig. 3.3. Starting with $i = d$, we proceed as follows: For any vertex $y \notin \{x_0, \ldots, x_d\}$ and connected to x_i, remove this edge and replace it with an edge between y and x_{i-3}. Index i is then decreased and the procedure is repeated until $i = 2$. Note that, by applying this reduction, both the longest distance and the number of edges are not modified. The resulting graph G' is a subgraph of G_2. The number of edges in $G'[Q]$ is at most equal to the number of edges in G_2, where now $|V| = \omega_\gamma(G) - d + 2$. Thus, if a path of length d exists in any γ-quasi-clique, then G_2 must also have a density γ setting $L = d$. Applying (3.2), we obtain

$$\gamma \binom{\omega_\gamma(G)}{2} \leq \binom{\omega_\gamma(G) - L + 2}{2} - 1 + (L - 2). \tag{3.4}$$

If we solve (3.4) at equality for the largest value of L, we obtain the longest distance in G_2, which is the longest distance any gamma quasi-clique can achieve. Solving (3.4) for L reduces to solving for the root of a quadratic function yielding $L \leq \left\lfloor \omega_\gamma(G) + \frac{1}{2} - \sqrt{\gamma.\omega_\gamma^2(G) - (2+\gamma)\omega_\gamma(G) + \frac{17}{4}} \right\rfloor$.

This bound is sharp since construction G_2 has the distance

$$L = \left\lfloor \omega_\gamma(G) + \frac{1}{2} - \sqrt{\gamma.\omega_\gamma^2(G) - (2+\gamma)\omega_\gamma(G) + \frac{17}{4}} \right\rfloor$$

and density γ. ■

k-Connected subgraph model. The following proposition presents a bounding value to the pairwise distance within the graph as a function of the size of the k-connected subgraph.

Proposition 3.5.3 *If $\breve{\omega}_k(G)$ is the size of the k-connected subgraph, then G has distance at most $\left\lfloor \frac{\breve{\omega}_k(G)-2}{k} + 1 \right\rfloor$. This bound is tight for k even.*

Proof By Menger's theorem [26], every pair of vertices in a k-connected graph must have k vertex-independent paths between them. Consider the most distant vertices

x and y in a k-connected graph and denote by d the distance between them. Each of the k paths between x and y must have a distance of at least d. This means that $\breve{\omega}_k(G) \geq k(d-1)+2$. In other terms, the subgraph consists of at least $k(d-1)+2$ vertices. Solving for d gives the desired bound.

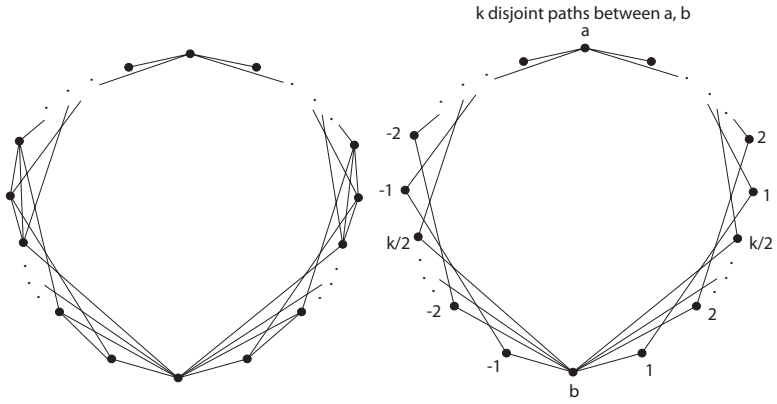

Fig. 3.4. Illustration of G_3 construction.

To show the tightness of this bound, consider the following construction denoted by G_3 (see Fig. 3.4). Place on a circle $\breve{\omega}_k(G)$ independent vertices labeled $\{x_0, \ldots, x_z\}$ where $z = \breve{\omega}_k(G) - 1$. Connect each vertex to its $\lceil \frac{k}{2} \rceil$ immediate neighbors on each side. The resulting graph is k-connected since every pair of vertices x_i and x_j has k disjoint paths between them. To form these paths, connect x_i to a neighbor x_{i+a} or x_{i-a} where $a \in \{0, \ldots, \lceil \frac{k}{2} \rceil\}$. Proceed by connecting x_{i+a} to $x_{i+a+\lceil \frac{k}{2} \rceil}$ then to $x_{i+a+2\lceil \frac{k}{2} \rceil}$ (or alternatively x_{i-a} to $x_{i-a-\lceil \frac{k}{2} \rceil}$ and then to $x_{i-a-2\lceil \frac{k}{2} \rceil}$) until eventually the path reaches vertex y. These k paths will remain disjoint at all steps yielding a k-connected graph. The furthest distance between two vertices in this graph is $\left\lfloor \frac{\breve{\omega}_k(G)-2}{2\lceil k/2 \rceil} + 1 \right\rfloor$, which shows that the given bound is tight when k is even. ■

While a relative bound can be generated, we cannot guarantee a fixed bounding value for distance in k-connected subgraphs. In fact, for a fixed k, the size of construction G_3 can be chosen arbitrarily large suggesting that no absolute bound exists.

3.5.2 Bounds on Domination

In this section, we explore which set size is guaranteed to be a dominating set for each of the canonical clique relaxations. The goal is to find the threshold set size above which any set of vertices that is selected from the subgraph of size at least the threshold size indubitably forms a dominating set. While cliques guarantee that any one vertex dominates the entire subgraph, the s-plex relaxation ensures the subgraph is dominated by any set of s vertices. Since any s-club must be connected, any set of size $\overline{\omega}_s(G)-1$ dominates the subgraph. To prove this bound is tight, let us consider the special case of a star graph. To guarantee a dominating set, one should choose at least all but one vertex from the subgraph. Otherwise, selecting a set of size $\overline{\omega}_s(G)-2$ could overlook the hub vertex and one of its peripheral vertices, which would not result in a dominating set.

No absolute bound on the domination for a k-core can be made knowing that construction G_1 remains a k-core for arbitrarily large sizes. However, a relative bound can be deduced in terms of the size of the k-core. The initial vertex along the major chain in G_1 can only be dominated by a set if this vertex or one of its k neighbors is contained in the set. Hence, we can only guarantee any set of size at least $\omega'_k(G)-k$ to be a dominating set.

For a γ-quasi-clique, no relative bound better than $\omega_\gamma(G)$ can guarantee a resulting dominating set. The reason is that for a γ-quasi-clique S, $\gamma\binom{|S|+l}{2} \leq \binom{|S|}{2}$ may hold for some fixed $\gamma < 1$, l and large enough $|S|$. A γ-quasi-clique could then contain an independent vertex accompanied by a large enough clique S. In this case, only sets containing all vertices in the subgraph are guaranteed to be dominating.

Knowing that a k-connected subgraph is also a k-core, any set of size at least $\breve{\omega}_k(G)-k$ is a dominating set. This bound is indeed tight, since a k-connected subgraph could contain a clique of size $\breve{\omega}_k(G)-1$ with an additional vertex linked to exactly k vertices from the clique. In this special case, excluding more than k vertices

from the set would no longer guarantee that the additional vertex is dominated by the set.

3.5.3 Bounds on Degree

The k-core relaxation ensures that each vertex in the induced subgraph has degree at least k. Observing construction G_1, it can be noted that guaranteeing a high minimum degree within the induced subgraph does not necessarily result in a cohesive subgroup on a global scale, especially for relatively large graphs.

The s-club model guarantees little regarding the minimum vertex degree. For instance, a star graph is an s-club for any $s \geq 2$ with a degree 1 for all of its peripheral vertices.

The s-plex model restricts each vertex within the induced subgraph to have at most $s-1$ non-neighbors. The best lower bound on vertex degree within an s-plex is found to be $\tilde{\omega}_s(G) - s$.

While a γ-quasi-clique guarantees a dense subgraph, it may allow independent vertices within its structure. For instance, a structure consisting of a clique and an independent vertex is a dense graph, however with zero minimum degree. Knowing that the minimum possible degree is no less than the graph's connectivity, a formal bound on the minimum degree for γ-quasi-cliques can be deduced from the lower bound on connectivity, i.e., $max\{0, \gamma\binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G)-1}{2}\}$, as will be discussed in Subsection 3.5.5.

Along the same direction, the minimum vertex degree in a k-connected subgraph is at least k.

3.5.4 Bounds on Density

By definition, the γ-quasiclique ensures a minimum subgraph density of γ.

Given its connectivity requirement, an s-club should comprise at least $\overline{\omega}_s(G) - 1$ edges, and hence possesses a density of at least $\frac{2}{\overline{\omega}_s(G)}$. This bound is tight in the case of a star graph.

For k-cores and k-connected subgraphs, each vertex is at least connected to k other vertices, summing to a total of at least $\frac{k\omega'_k(G)}{2}$ and $\frac{k\check{\omega}_k(G)}{2}$ edges, guaranteeing a minimum density of $\frac{k}{\omega'_k(G)-1}$ and $\frac{k}{\check{\omega}_k(G)-1}$ respectively. Note that construction G_3, which is both a k-core and a k-connected subgraph, contains precisely this density, implying the tightness of the given bounds.

By a similar argument, the largest s-plex is guaranteed a density of at least $1 - \frac{s-1}{\tilde{\omega}_s(G)-1}$. While for a fixed k a smaller size k-core would result in a higher density, the same is true for larger sized s-plex with a fixed s.

3.5.5 Bounds on Connectivity

The k-connected subgraph requires the deletion of k vertices to disconnect the graph.

In the case of s-clubs, the removal of only one vertex could potentially disconnect the graph, as illustrated by star graphs. No matter how large these relaxations could be, they do not ensure connectivity, especially if they comprise a star subgraph within their structure.

On the other hand, the larger the size of an s-plex, the higher is its connectivity level. Seidman and Foster [52] have shown that any s-plex ensures a connectivity of at least $\tilde{\omega}_s(G) - 2s + 2$, linearly increasing with its size. This expression suggests that an s-plex is connected when its size exceeds $2(s-1)$. Two independent cliques of size $s-1$ form a set of $2(s-1)$ vertices that are not connected, showing this bound is tight.

Oppositely, the smaller the size of a k-core, the higher is its connectivity level. The following proposition gives a lower bound on connectivity in terms of the k-core size $\omega'_k(G)$.

Proposition 3.5.4 *If $\omega'_k(G)$ is the size of k-core S, then*

$$\kappa(G[S]) \geq 2k + 2 - \omega'_k(G). \tag{3.5}$$

This bound is also tight.

Proof Consider an arbitrary graph $G = (V, E)$ with the minimum degree $\delta(G)$. G will be connected as long as $2(\delta(G) + 1) - |V| > 0$, since in this case, any two vertices that are not directly connected must have a common neighbor. Therefore, for a k-core S with minimum degree k and size $\omega'_k(G)$, S is connected provided $2(k + 1) - \omega'_k(G) > 0$.

When removing one vertex from G, $|V|$ is reduced by exactly one and $\delta(G)$ may decrease by at most one, causing the expression $2(\delta(G) + 1) - |V| > 0$ to decrease by at most one, and so does its connectivity. Thus, the connectivity of the k-core satisfies $\kappa(G[S]) \geq 2(k + 1) - n = 2k + 2 - n$.

The bound is sharp since two separate cliques of size $(k+1)$ form a k-core whose connectivity is zero. ■

For γ-quasi-cliques, the following proposition provides the lower bounds on connectivity.

Proposition 3.5.5

$$\kappa(G[Q]) \geq \left\lceil \gamma \binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G) - 1}{2} \right\rceil. \tag{3.6}$$

This bound is also tight.

Proof Let $a = \gamma\binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G)-1}{2}$ define the number of edges necessary beyond a clique of size $(\omega_\gamma(G) - 1)$ to achieve a density γ. By definition, any γ-quasi-clique Q of size $\omega_\gamma(G)$ comprises at least $\gamma\binom{\omega_\gamma(G)}{2}$ edges. Q can then be represented as a clique of size $\omega_\gamma(G)$ missing at most $\binom{\omega_\gamma(G)}{2} - \gamma\binom{\omega_\gamma(G)}{2}$ edges. Knowing that,

$\binom{\omega_\gamma(G)}{2} = \omega_\gamma(G) - 1 + \binom{\omega_\gamma(G)-1}{2}$, Q is therefore a clique $K_{\omega_\gamma(G)}$ missing at most $[\omega_\gamma(G) - 1 + \binom{\omega_\gamma(G)-1}{2}] - \gamma\binom{\omega_\gamma(G)}{2} = [\omega_\gamma(G) - 1 - a]$ edges.

$K_{\omega_\gamma(G)}$ being $(\omega_\gamma(G) - 1)$-connected, the removal of $(\omega_\gamma(G) - 1 - a)$ edges could destroy at most $(\omega_\gamma(G) - 1 - a)$ edge disjoint paths. The resulting graph Q has at least $\omega_\gamma(G) - 1 - (\omega_\gamma(G) - 1 - a) = a$ disjoint paths. By Menger's theorem, Q is at least a-connected, yielding $\kappa(Q) \geq \gamma\binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G)-1}{2}$. Knowing the integrality of a graph connectivity, the desired result is obtained by taking the ceiling of the latter expression.

To show that the above bound is tight, let us consider a clique of size $\omega_\gamma(G) - 1$ and a single vertex. Connecting this vertex to $a = \gamma\binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G)-1}{2}$ vertices in the clique, the resulting graph becomes a γ-quasi-clique of size $\omega_\gamma(G)$. Its connectivity is exactly equal to the number of edges connecting that single vertex to the clique, i.e., $\gamma\binom{\omega_\gamma(G)}{2} - \binom{\omega_\gamma(G)-1}{2}$. ■

All the bounds developed in this section are summarized in Table 3.3.

Table 3.3
Bounds on guaranteed cohesiveness of canonical clique relaxations.

$S \subseteq V$	Diameter	Dominating Set	Minimum Degree	Edge Density	Connectivity
Clique	"one"	"one"	"all"	"one"	"all"
s-club	$\underline{s}$	$\tilde{\omega}_s - 1$	1	$\frac{2}{\tilde{\omega}_s}$	1
s-plex	s	$\underline{s}$	$\tilde{\omega}_s - s$	$1 - \frac{s-1}{\tilde{\omega}_s - 1}^*$	$\tilde{\omega}_s - 2s + 2^*$
k-core	d'_k	$\omega'_k - k$	$\underline{k}$	$\frac{k}{\omega'_k - 1}^*$	$2k + 2 - \omega'^*_k$
γ-quasiclique	d_γ	ω_γ	$\lceil \gamma\binom{\omega_\gamma}{2} - \binom{\omega_\gamma - 1}{2} \rceil$	$\underline{\gamma}$	$\lceil \gamma\binom{\omega_\gamma}{2} - \binom{\omega_\gamma - 1}{2} \rceil$
k-connected	$\lfloor \frac{\omega_c - 2}{k} + 1 \rfloor$	$\omega_c - k$	k	$\frac{k}{\omega_c - 1}$	$\underline{k}$

3.6 Cohesiveness Properties of Weak First-Order Clique Relaxations

To illustrate the lack of cohesiveness in weak clique relaxation models, we examine the properties of s-clique to see how they compare with s-club. Since s-clique exhibits weak heredity and s-club does not exhibit heredity to any extent, k-clique offers an attractive alternative in applications where computational speed is critical.

We first present the diameter bounds of s-clique. Consider an independent set of vertices $S = \{x_1, ..., x_z\}$, where $z \geq 2s + 3$. Connect each pair of vertices x_i and x_j by a path $\{x_i, c_{ij1}, c_{ij2}, ..., c_{ijs-1}, x_j\}$ of length exactly s, where c_{ijt}, $t = 1, .., s - 1$ denote the linking vertices, while ensuring that paths between any two pairs (x_i, x_j) and (x_i, x_s) are disjoint for any i, and $j \neq s \neq i$. The resulting construction is represented in Fig. 3.5. Clearly, the distance between any two vertices x_i and x_j is s, and S therefore constitutes an s-clique. From the set S, any linking vertex c_{ijt}, $t = 1, ..., s - 1$ can only be connected to x_i and x_j through a path of length at most s. Since no vertices could be added to S while remaining an s-clique, S is maximal and is an independent set.

Since an s-clique can be an independent set, it can have infinite diameter. As we did in Section 3.5.1, we will force the s-clique to be connected and repeat the analysis. We will show that even with enforcing connectivity, an s-clique does not necessarily have a low diameter. We will show that we cannot guarantee a diameter less than $\omega_s - 1$.

Proposition 3.6.1 *There exists a connected maximal s-clique S of size $\omega_s(G)$ such that the induced subgraph $G[S]$ has a diameter $\omega_s(G) - 1$.*

Proof To show this, we consider the construction in Fig. 3.5. As shown previously, the independent set $S = \{x_1, ..., x_z\}$ constitutes a maximal s-clique, with $z = \omega_s(G)$. To form a connected subgraph, we replace the paths of length s between x_i and x_{i+1}, $i = 1, ..., z - 1$, with edges. Note that x_1 and x_z are still connected through a path of length s. The resulting subgraph $G[S]$ consists of an acyclic path $\{x_1, ..., xz\}$

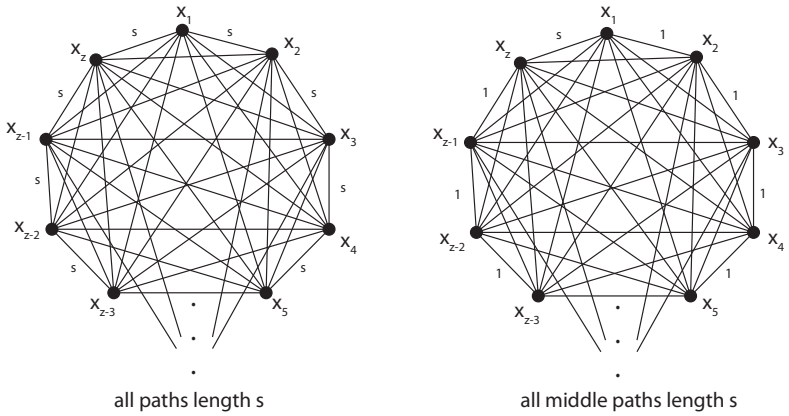

Fig. 3.5. Illustration of G_4 construction.

forming a connected s-clique (refer to Fig. 3.5). In the construction, any linking vertex c_{ijt}, $t = 1, ..., s-1$ can be connected to at most $2s+2$ vertices from the set S. To see this, note that paths of length at most s exist between c_{ijt} and each of $x_{i-(s-t)}, ..., x_{i-1}, x_i, x_{i+1}, ..., x_{i+(s-t)}$ and $x_{j-t}, ..., x_{j-1}, x_j, x_{j+1}, ..., x_{j+t}$. With the set S consisting of at least $2s+3$ vertices, no c_{ijt} can be connected to all elements in S with paths of length at most s, yielding the maximality of the s-clique S. Furthermore, the subgraph $G[S]$, consisting of the acyclic path $\{x_1, ..., xz\}$, has a diameter of $\omega_s(G) - 1$, corresponding to the distance between x_1 and x_z. ■

In analyzing our other structural properties for s-cliques, we see the only guaranteed dominating set is one that includes all vertices. This conclusion stems from the special case of an independent set of vertices that could form an s-clique, as demonstrated in construction G_4. As a result, we could only make the obvious observation in this case that every set of size $\omega_s(G)$ in an s-clique dominates the s-clique. Furthermore, an s-clique can also be formed while excluding the hub vertex within the star graph, resulting in independent vertices with zero minimum degree. Since an s-clique may consist of an independent set of vertices, no minimum density

can be guaranteed. Finally no connectivity can be guaranteed because once again, an s-clique can be an independent set.

3.7 Applications

Table 3.3 can be very useful in identifying which clique relaxation is particularly fit for a given application. When choosing a clique relaxation, the properties essential for a set to be considered a group in a particular application must first be chosen. The clique relaxations demonstrating these properties in the table are candidates for grouping. However, to decide between candidates, the properties that cannot be avoided by a relaxation should also be considered. A set S that should be considered a group will be missed if the clique relaxation necessarily requires extra structure that S does not demonstrate. While a property gained "for free" may indicate that a relaxation more closely resembles a clique, it also means the relaxation cannot avoid having that property. Thus it is crucial to pay attention to properties both ensured and omitted in the table to choose the appropriate clique relaxation for a given application.

In the discussion that follows, we attempt to highlight the important characteristics for each clique relaxation in Table 3.3. We demonstrate applications for which each clique relaxation is particularly fit because of its characteristics as evidence of the importance of the properties.

The s-clique and s-club relaxations were designed to guarantee reachability. A unique feature of these relaxations from the table is they require little for minimum degree, dominating set size, density, and connectivity. These clique relaxations are particularly adept when data should be clustered with low diameter, but also low density. The s-clubs have had success in clustering topically related information on the internet to facilitate faster searches for this reason [55]. The internet, along with numerous other networks, demonstrates preferential attachment, meaning new edges tend to appear at nodes that already have high degree. Sets of nodes with low

diameter but also low density permeate such graphs and often should be clustered despite sparsity. When this is the case, s-clubs or s-cliques are the appropriate choices. We refer the reader to [10, 44] for more information.

The cohesiveness properties of s-club are stronger than those of s-clique. However, the s-clique relaxation has two distinct properties that keep it from being obsolete. First, it demonstrates weak heredity, which makes constructing s-cliques by heuristics much simpler. Second, the largest s-clique in a graph coincides with a clique in the s^{th} power of a graph. The clique problem has been so well studied that this immediately gives a trove of tools with which to identify the largest s-clique.

The key property of the k-core relaxation is that it is solvable in polynomial time. It has proven a useful tool for pruning a graph in order to find cliques and clique relaxations where a lower bound is known on the degree of the vertices in the induced subgraph. At times, it has removed enough vertices that the optimal solution can be found, as was the case with the market graph [15]. In addition, k-core has been used to detect molecular complexes and predict protein functions [5, 7].

The s-plex relaxation is unique in that it ensures nearly every property in Table 3.3 to an extent. It was specifically designed to be an alternative to s-clique and s-club with more guaranteed structure because the internal structure of diameter 2 graphs was so "poorly understood" [52]. Accordingly, it is often useful in applications where cliques are desired but a few missing edges are tolerated, perhaps caused by errors in data collection such as noise. Because it ensures high interaction by **all** members, it has become a key tool in protein interaction networks for grouping proteins. Proteins that are part of a s-plex in a protein interaction network, which is often called a core, tend to have similar structure since they have many interactions in common [41].

Quasi-cliques, like the s-plex relaxation, demonstrate a high level of interaction between all members. This inevitably results in numerous other properties, as was true with s-plex. What makes it unique from s-plex, however, is that depending on

size, no minimum degree is required. This makes it useful in data mining applications where high density sets should be grouped regardless of structure. Quasi-cliques were successfully used to mine massive sets of telecommunications data in order to find a good way of organizing it [2]. By storing high density sets, regardless of structure, in the same section of memory, fewer calls to memory are necessary for data analysis. This can save significant time. A relaxation similar to the quasi-clique proved useful in mining biological data for functional relationships between attributes [22]. They found cohesive subgroups that dwarfed the largest cliques and helped reveal relationships previously missed due to a small subset of missing edges. It is likely in both applications that high density groups would be missed by s-plex, due to the extra structure it requires.

The k-connected subgraph is unique because it is specifically defined to guarantee communication that can survive breakdowns in the network. It is often referred to as a "survivable" or "redundant" network outside of graph theory and is more often used in design rather than analysis of a network. It has been proposed as an alternative to density-based relaxations in [32] for identifying complexes in protein interaction networks. Further research on uses for this clique relaxation could prove extremely valuable, especially in applications where network survivability is key.

3.8 Conclusion

We introduced a taxonomy of clique relaxation models that encompasses many of the popular models studied in the literature and establishes foundations for a systematic study of the corresponding optimization problems and their applications. The section opens the door for many interesting research directions that can be undertaken in exploring the existing, as well as newly identified clique relaxation models. In particular, the established bounds on cohesiveness properties of the canonical clique relaxation models should help to identify higher-order relaxations that are worth investigating. The relationship between optimization problems dealing with abso-

lute and relative relaxations based on relaxing the same elementary clique-defining property is another interesting question to study.

4. ON THE MAXIMUM QUASI-CLIQUE PROBLEM

The previous section introduced us to a clique relaxation built around density known as quasi-clique. A γ-quasi-clique requires at least the fraction γ of all possible edges between vertices to be present. As a density-based relaxation, γ-quasi-cliques, also sometime referred to as γ-cliques, provide a reasonable way for grouping objects that possess no inherent reason to display the structure of many other clique relaxations. The previous literature related to the MAXIMUM γ-CLIQUE problem concentrated on heuristic detection of large quasi-cliques in various application scenarios. The goal of this section is to start examining the MAXIMUM γ-CLIQUE problem from the mathematical perspective, including establishing the computational complexity of the problem for any fixed γ, exploring structural properties of γ-cliques, deriving analytical upper bounds, and developing mixed-integer programming (MIP) formulations.

The remainder of this section is organized as follows. Section 4.1 introduces the necessary definitions and notations. The NP-completeness of the decision version of the MAXIMUM γ-CLIQUE problem is proved in Section 4.2. Section 4.3 defines the quasi-heredity property and establishes analytical bounds on the γ-clique number of a graph. Mathematical programming formulations are derived in Section 4.4 and results of preliminary numerical experiments are reported in Section 4.5. Finally, Section 4.6 concludes the section.

4.1 Definitions, Notations, and Motivation

Let $G = (V, E)$ be a simple undirected graph with the set V of n vertices and the set E of m edges. Recall that a clique is a set of vertices that induce a complete subgraph. A clique is *maximal* if it is not a subset of a larger clique, and *maximum* if there is no larger clique in the graph. The MAXIMUM CLIQUE problem is to

find a clique of maximum cardinality in G, which is called the *clique number* and is denoted by $\omega(G)$.

Given $G = (V, E)$ and fixed real γ satisfying $0 < \gamma < 1$, a subset of vertices Q is called a γ-clique if the edge density of the induced subgraph $G[Q]$, which is given by the ratio of the number of edges in $G[Q]$ to $\binom{|Q|}{2}$, is at least γ. The MAXIMUM γ-CLIQUE problem asks for a γ-clique with the maximum possible number of vertices in G. We will denote the γ-clique number of a graph G, which is the cardinality of a largest γ-clique in G, by $\omega_\gamma(G)$. Note that for $\gamma = 1$ the MAXIMUM γ-CLIQUE problem would become the classical MAXIMUM CLIQUE problem, while for $\gamma = 0$ the problem would be trivial. For a fixed $\gamma \in (0, 1)$ the problem has not been well studied. The earliest publication on the topic is attributed to Abello et al. [1] who defined the concept of γ-quasi-clique and proposed greedy randomized adaptive search procedures (GRASP) for detecting large quasi-cliques in graphs representing telecommunications data. Similar approaches were implemented in semi-external memory algorithms that handled massive graphs with hundreds of millions of vertices [2]. Several other papers, some of which use modified definitions of quasi-cliques, presented heuristic approaches to detecting large quasi-cliques in graphs arising in various applications [14, 18, 38, 45, 49, 60]. In summary, the previous work on the problem of interest concentrated mainly on heuristic detection of large quasi-cliques in graphs arising in a diverse set of applications. This section provides a formal study of the computational complexity of the γ-clique problem, establishes analytical bounds on the γ-clique number and proposes mathematical programming formulations of the problem that can be used for finding provably optimal solutions.

4.2 Computational Complexity

This section presents the computational complexity analysis for the the MAXIMUM γ-CLIQUE problem for any fixed density γ between 0 and 1. To simplify the analysis, we first replace a real γ in the definition of a γ-clique with a rational

$\frac{p}{q}$, where positive integers p and q are given, resulting in the $\frac{p}{q}$-clique model. Afterwards, the results obtained for $\frac{p}{q}$-cliques will be extended to the general γ-clique case. Following the standard approach [31], we define the recognition version of the problem, $\frac{p}{q}$-CLIQUE, as follows: Given a graph $G = (V, E)$ and positive integers p, q and k, does there exist a $\frac{p}{q}$-clique of size at least k in G?

Proposition 4.2.1 *The $\frac{p}{q}$-CLIQUE problem is NP-complete for any positive integer constants $p, q,\ p < q$.*

Proof The proof is done by observing that $\frac{p}{q}$-CLIQUE is, obviously, in the class NP and by reducing the classical CLIQUE problem to $\frac{p}{q}$-CLIQUE. Namely, for the given k and $\frac{p}{q}$, we will construct an auxiliary graph $G' = (V', E')$ and prove that G has a clique of size k if and only if $G \cup G'$ has a $\frac{p}{q}$-clique of size $|V'| + k$.

The construction proceeds as follows. We build the set of vertices V' with $|V'| = 4(|V|^2 + k^2)q - k$ and construct edges to obtain a $2|V|$-regular graph. It is easy to observe that one can always construct a graph with any specified even regularity, provided there are enough vertices. This can be done by, e.g., placing all the vertices of V' on a circle and connecting each vertex to its immediate $|V|$ neighbors on each side in the circle. Next we randomly place edges so that we have $\frac{p}{q} \cdot \binom{|V'|+k}{2} - \binom{k}{2}$ edges between the $|V'|$ vertices. The value of $\frac{p}{q} \cdot \binom{|V'|+k}{2} - \binom{k}{2}$ is integer since $|V'|+k$ is a multiple of $2q$. $|V'|$ is sufficiently large to guarantee that the following inequalities hold:

$$\binom{|V'|}{2} \geq \frac{p}{q}\binom{|V'|+k}{2} - \binom{k}{2} \geq |V||V'|.$$

The first inequality can be verified by multiplying through by 2q, combining terms to one side of the inequality, factoring out $4(|V|^2 + k^2)q$ from all terms, and finally by replacing $(q - p)$ with 1 since it must be at least 1. The second inequality is relatively simple to verify. The first inequality ensures that we can fit in the number of edges needed for a $\frac{p}{q}$-clique of size $|V'| + k$ in $G \cup G'$, where k vertices would come

from a clique in G. The second inequality allows us to build a $2|V|$-regular graph on $|V'|$ vertices with no more edges than the desired value of $\frac{p}{q}\binom{|V'|+k}{2} - \binom{k}{2}$.

Consider the union $G \cup G'$ of the two graphs. Then we can show that for any $\frac{p}{q}$-clique Q with $|Q| > |V'|$ in $G \cup G'$ there exists a $\frac{p}{q}$-clique Q' in $G \cup G'$ such that $|Q'| = |Q|$ and $V' \subset Q'$. Indeed, suppose that there is a $\frac{p}{q}$-clique Q that has more than $|V'|$ vertices and does not include the entire V'. Define V_{in} to be the vertices from G' that are included in this $\frac{p}{q}$-clique and V_{out} to be the ones missing. Then $|V_{out}| \leq |V|$, so a vertex from V_{out} cannot be connected to more than $|V| - 1$ vertices of V_{out}. Since every vertex in G' has degree at least $2|V|$, each vertex in V_{out} must be connected to at least $|V| + 1$ vertices in V_{in}. Therefore, any vertex from $Q \setminus V_{in}$ can be replaced with any vertex from V_{out} in Q with no reduction in the edge density of the subgraph induced by Q. Substituting arbitrary $|V_{out}|$ vertices from $Q \setminus V_{in}$ with V_{out}, we obtain a $\frac{p}{q}$-clique Q' of the same size as Q that includes the entire V'.

To complete the proof, we will show that G has a clique of size k if and only if $G \cup G'$ has a $\frac{p}{q}$-clique of size $|V'| + k$. Given a clique C of size k in G, combining $G[C]$ with all of G' we have $|V'| + k$ vertices and $\frac{p}{q} \cdot \binom{|V'|+k}{2} - \binom{k}{2} + \binom{k}{2} = \frac{p}{q} \cdot \binom{|V'|+k}{2}$ edges, making this collection of vertices a $\frac{p}{q}$-clique by definition. On the other hand, assuming that $G \cup G'$ has a $\frac{p}{q}$-clique of $|V'| + k$ vertices, we know that there is $\frac{p}{q}$-clique Q' of size $|V'| + k$ in $G \cup G'$ that contains all of the vertices from G' and hence precisely k of the vertices come from G. To see that the k vertices in $Q \setminus V'$ form a clique, consider the density of G'. It is precisely $\frac{p}{q} \cdot \binom{|V'|+k}{2} - \binom{k}{2}$ by construction. If the k vertices from G don't contribute $\binom{k}{2}$ edges, then the set of $|V'| + k$ vertices forming Q cannot have density $\frac{p}{q}$. Thus the existence of a $\frac{p}{q}$-clique of size $|V'| + k$ means the set of k vertices in the original graph induces a subgraph with $\binom{k}{2}$ edges and hence forms a clique. This establishes the NP-completeness of the $\frac{p}{q}$-Clique problem.

■

Corollary 4.2.1 *For any given fixed real $\gamma \in (0,1)$, the γ-*Clique *problem is NP-complete.*

Proof First observe that instead of assuming that $\frac{p}{q}$ is a fixed value, we could allow p and q to be given with the problem instance, only placing the restrictions that $p = p(n)$ and $q = q(n)$ be functions of the order $O(n^2)$ and $0 < p(n) < q(n)$. Clearly, any possible edge density for a graph on n vertices can be specified by $\frac{p(n)}{q(n)}$ with $p(n), q(n) \leq n(n-1)/2$. The resulting $\frac{p(n)}{q(n)}$-Clique problem is also NP-complete, using the same proof we have given above. Based on this observation, it suffices to show that for any $\gamma \in (0,1)$ there exist two functions $p(n), q(n)$, of the order $O(n^2)$ such that $0 < p(n) < q(n)$ for any n, and the resulting $\frac{p(n)}{q(n)}$-Clique problem and γ-Clique problem are equivalent. Showing $p(n)$ and $q(n)$ of order $O(n^2)$ ensures they do not need to grow too large in size in order to sufficiently approximate any irrational value $\gamma \in (0,1)$.

Let γ be fixed. Define for any positive integer s

$$\bar{p}(s) = \left\lceil \gamma \frac{s(s-1)}{2} \right\rceil, \quad \bar{q}(s) = \frac{s(s-1)}{2}.$$

Note that $\bar{p}(s)$ defines the minimum possible number of edges in the induced subgraph of any γ-clique of size s, so $\bar{p}(s)/\bar{q}(s) \geq \gamma$ and the actual edge density of any γ-clique of size s is at least $\bar{p}(s)/\bar{q}(s)$. Hence, any γ-clique of size s is also a $\frac{\bar{p}(s)}{\bar{q}(s)}$-clique and vice versa. Next, for any positive integer n we define

$$p(n) = \bar{p}(s^*),\ q(n) = \bar{q}(s^*),\ \text{where } s^* = \arg\min_{1 \leq s \leq n} \frac{\bar{p}(s)}{\bar{q}(s)}.$$

Then $0 < p(n) < q(n)$ for any n and $p(n), q(n)$ are of order $O(n^2)$. Moreover, for any $s \leq n$, any s-vertex subgraph $G_s = (V_s, E_s)$ of G is $\frac{p(n)}{q(n)}$-clique if and only if it is a γ-clique, so $\frac{p(n)}{q(n)}$-Clique problem is equivalent to γ-Clique problem. ■

4.3 Properties of γ-cliques

In this section we will discuss some properties of γ-cliques that may be useful in designing solution procedures for the MAXIMUM γ-CLIQUE problem. Unlike cliques, the γ-cliques fail to display a key property used in successful algorithms for the MAXIMUM CLIQUE problem: *heredity*. Recall from Section 3, a property is called *hereditary* if, when it exists in a graph, it exists in all its induced subgraphs. It is easy to identify γ-cliques containing subsets of vertices that induce subgraphs with edge density less than γ. Because of this, even checking maximality by inclusion is a non-trivial task for quasi-cliques. However, γ-cliques do display a related property, which we will call *quasi-heredity*.

4.3.1 Quasi-Heredity

If, given any graph $G = (V, E)$ satisfying a property P, there exists $v \in V$ such that $G-v := G[V \setminus \{v\}]$ also has property P, we call the property P a quasi-hereditary property and say that the property P displays *quasi-heredity* or *quasi-inheritance*.

Proposition 4.3.1 *The graph property of having edge density of at least γ displays quasi-inheritance. In other words, any γ-clique with $s > 1$ vertices is a strict superset of a γ-clique with $s - 1$ vertices.*

Proof Consider a γ-clique Q that induces a subgraph with $s > 2$ vertices and e edges (the statement is trivially true for $s = 2$). Then a smaller γ-clique of size $s-1$ can always be formed by removing a vertex v with the lowest degree within Q. Since this vertex will have the degree less than or equal to the average, which is given by $2e/s$, the edge density of the subgraph induced by $Q \setminus \{v\}$ will be at least

$$\frac{2e - 4e/s}{(s-1)(s-2)} = \frac{2e}{s(s-1)},$$

i.e., no less than that of $G[Q]$, and hence $Q \setminus \{v\}$ is a γ-clique. ■

The quasi-heredity property implies that, provided the vertices are placed in the right order, a maximum γ-clique can be found by starting with the first vertex in the list and sequentially adding next vertex if the resulting subset of vertices is still a γ-clique. This gives us a hope that if we apply some "smart" vertex ordering rules, perhaps based on vertex degrees, there is a chance that we will be able to find large γ-cliques quickly (even though we cannot say how far their size will be from optimal). This observation suggests metaheuristic procedures such as GRASP [2] as a natural choice for solving the problem of interest. The successful computational experience reported by Abello et al. [2] provides a practical evidence in support of this hypothesis.

4.3.2 Upper Bounds

The proposed upper bound on the γ-clique number is a generalization of the classical Amin-Hakimi bound on the clique number [6].

Proposition 4.3.2 *The γ-clique number $\omega_\gamma(G)$ of a graph G with n vertices and m edges satisfies the following inequality:*

$$\omega_\gamma(G) \leq \frac{\gamma + \sqrt{\gamma^2 + 8\gamma m}}{2\gamma}. \tag{4.1}$$

Moreover, if a graph G is connected then

$$\omega_\gamma(G) \leq \frac{\gamma + 2 + \sqrt{(\gamma+2)^2 + 8(m-n)\gamma}}{2\gamma}. \tag{4.2}$$

Proof The first bound is obtained by solving the quadratic inequality

$$\gamma \frac{\omega_\gamma(G)(\omega_\gamma(G) - 1)}{2} \leq m.$$

Assuming that the graph G is connected and has a γ-clique of size $\omega_\gamma(G)$, the following inequality must hold:

$$\gamma\frac{\omega_\gamma(G)(\omega_\gamma(G)-1)}{2}+n-\omega_\gamma(G)\leq m.$$

Solving this quadratic inequality for $\omega_\gamma(G)$, we obtain the second bound. ■

For $\gamma = 1$ the second of the bounds becomes the Amin-Hakimi bound on the clique number, which is the only constant-time computable upper bound used in the comparison performed by Budinich [19].

4.3.3 Relation Between $\omega_\gamma(G)$ and $\omega(G)$

Next we derive an inequality that will relate the γ-clique number to the clique number of G. We will need the following classical lower bound on the clique number that can be easily obtained from the Motzkin-Straus [47] formulation for the MAXIMUM CLIQUE problem:

$$\omega(G)\geq\frac{1}{1-\delta}, \tag{4.3}$$

where $\delta = 2m/n^2$.

Proposition 4.3.3 *The γ-clique number $\omega_\gamma(G)$ and the clique number $\omega(G)$ of graph G satisfy the following inequalities:*

$$\frac{\omega(G)-1}{\omega(G)}\leq\frac{\omega_\gamma(G)-1}{\omega_\gamma(G)}\leq\frac{1}{\gamma}\frac{\omega(G)-1}{\omega(G)}. \tag{4.4}$$

Proof The first inequality is trivial due to the fact that $\omega(G)\leq\omega_\gamma(G)$. To prove the second inequality, consider a γ-clique C of largest size $\omega_\gamma(G)$ in G. Then, according

to (4.3), the size $\omega(G[C])$ of the largest clique in the induced subgraph $G[C]$ satisfies the inequalities

$$\omega(G) \geq \omega(G[C]) \geq \frac{1}{1-\delta_C},$$

where $\delta_C = 2m_C/n_C^2$, m_C is the number of edges in $G[C]$, and $n_C = \omega_\gamma(G)$ is the number of vertices in $G[C]$. Since C is a γ-clique, we have

$$\delta_C = \frac{2m_C}{n_C^2} = \frac{2m_C}{n_C(n_C-1)}\frac{n_C-1}{n_C} \geq \gamma\frac{n_C-1}{n_C}.$$

Therefore,

$$\omega(G) \geq \frac{1}{1-\gamma\frac{n_C-1}{n_C}},$$

which, taking into account that $n_C = \omega_\gamma(G)$, is equivalent to

$$\frac{\omega_\gamma(G)-1}{\omega_\gamma(G)} \leq \frac{1}{\gamma}\frac{\omega(G)-1}{\omega(G)}.$$

■

Corollary 4.3.1 *If* $\gamma > 1 - \frac{1}{\omega(G)}$ *then*

$$\omega_\gamma(G) \leq \frac{\omega(G)\gamma}{1-\omega(G)+\omega(G)\gamma}. \tag{4.5}$$

Proof The result follows directly from the second inequality in (4.4). ■

Bound (4.5) can be especially useful for large sparse networks that often arise in real-life applications. Such networks typically have very small clique number compared to the total number of vertices and the size of their largest clique can be computed using effective preprocessing procedures. Table 4.1 provides the value of bound (4.5) with $\gamma = 0.95, 0.9, 0.85$ for graphs with the clique number between 3 and 10. As can be seen from this table, in some cases the bound allows to claim that a maximum clique of G is also a maximum γ-clique of the same graph.

Table 4.1
The value of upper bound (4.5) on γ-clique number with $\gamma = 0.95, 0.9, 0.85$ for graphs with small clique number.

$\omega(G)$	$1 - \frac{1}{\omega(G)}$	0.95	0.9	0.85
3	0.667	3.35	3.86	4.64
4	0.75	4.75	6	8.5
5	0.8	6.33	9	17
6	0.83	8.14	13.5	51
7	0.86	10.23	21	–
8	0.88	12.67	36	–
9	0.89	15.55	81	–
10	0.9	19	–	–

4.4 MIP Formulations of the Maximum γ-Clique Problem

The lack of structure in γ-cliques as opposed to cliques and some other clique relaxations, such as k-plex [9], makes this problem extremely difficult to solve to optimality. Indeed, the most successful combinatorial algorithms for the MAXIMUM CLIQUE and MAXIMUM k-PLEX problems rely on the heredity property of these structures, which is not an option in our case. Tight bounds and effective pruning strategies within a branch-and-bound framework are not easy to develop for the MAXIMUM γ-CLIQUE problem. This section develops mixed integer programming formulations for the MAXIMUM γ-CLIQUE problem.

We consider a graph $G = (V, E)$ with the set $V = \{1, \ldots, n\}$ of n vertices. We denote by $A = [a_{ij}]_{i,j=1}^{n}$ its adjacency matrix, which is an $n \times n$ matrix with a_{ij} equal to one if $(i, j) \in E$, and zero otherwise. We introduce n binary decision variables $x_i, i = 1, \ldots, n$, one for each vertex, such that the value x_i^* assigned to the variable x_i in the output optimal solution will indicate whether the corresponding vertex i is

a part of the maximum γ-clique C^* computed. Namely, $i \in C^*$ if and only if $x_i^* = 1$. Then the MAXIMUM γ-CLIQUE problem can be formulated as follows:

$$\omega_\gamma(G) = \max \sum_{i=1}^{n} x_i \tag{4.6}$$

subject to

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} a_{ij} x_i x_j \geq \gamma \sum_{i=1}^{n} \sum_{j=i+1}^{n} x_i x_j, \tag{4.7}$$

$$x_i \in \{0, 1\},\ i = 1, \ldots, n. \tag{4.8}$$

This problem has a linear objective, but its single constraint is quadratic. Next, we introduce new variables to make this problem linear. We define $x_{ij} = x_i x_j$. We need only $n(n-1)/2 - n$ new variables since $x_{ij} = x_{ji}$. The quadratic constraint $x_{ij} = x_i x_j$ with binary variables is equivalent to the following three linear constraints:

$$x_{ij} \leq x_i,\ \ x_{ij} \leq x_j,\ \ x_{ij} \geq x_i + x_j - 1. \tag{4.9}$$

Therefore, we can formulate our graph problem as a mixed integer linear optimization problem:

$$\omega_\gamma(G) = \max \sum_{i=1}^{n} x_i, \tag{4.10}$$

subject to

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} (\gamma - a_{ij}) x_{ij} \leq 0, \tag{4.11}$$

$$x_{ij} \leq x_i,\ \ x_{ij} \leq x_j,\ \ x_{ij} \geq x_i + x_j - 1,\ \ j > i = 1, \ldots, n \tag{4.12}$$

$$x_{ij} \geq 0,\ x_i \in \{0, 1\},\ j > i = 1, \ldots, n. \tag{4.13}$$

This formulation contains $n(n-1)/2$ variables and $\frac{3}{2}n(n-1) + 1$ constraints.

Next we consider an alternative linearization. Recall that the original formulation (4.6)-(4.8) had a single constraint that can be replaced with

$$\sum_{i=1}^{n} x_i \left(\gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j \right) \geq 0. \tag{4.14}$$

Let us define a new variable y_i for $i = 1, \ldots, n$ as follows:

$$y_i = x_i \left(\gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j \right). \tag{4.15}$$

Let us denote by

$$u_i = (1 - \gamma) \sum_{j=1}^{n} a_{ij}; \qquad l_i = -(n - 1 - \sum_{j=1}^{n} a_{ij})\gamma, \tag{4.16}$$

where u_i is the sum of all the positive coefficients and l_i is the sum of all the negative coefficients for the variables in the expression in parenthesis of (4.15). Since all variables are binary, the constants u_i and l_i satisfy the following inequalities:

$$l_i \leq \gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j \leq u_i, \quad i = 1, \ldots, n, \tag{4.17}$$

$$l_i \leq y_i \leq u_i, \quad i = 1, \ldots, n. \tag{4.18}$$

Thus, the quadratic equality (4.15) with binary variables is equivalent to the following four linear inequalities:

$$y_i \leq u_i x_i, \tag{4.19}$$

$$y_i \geq l_i x_i, \tag{4.20}$$

$$y_i \geq \gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j - u_i(1 - x_i), \tag{4.21}$$

$$y_i \leq \gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j - l_i(1 - x_i). \tag{4.22}$$

Therefore, the problem of finding a maximum γ-clique can be represented as the following mixed integer linear optimization problem with $2n$ variables, n of which are 0-1 variables and n – continuous, and $4n + 1$ constraints:

$$\omega_\gamma(G) = \max \sum_{i=1}^{n} x_i \tag{4.23}$$

subject to

$$\sum_{i=1}^{n} y_i \geq 0, \tag{4.24}$$

$$y_i \leq u_i x_i, \;\; y_i \geq l_i x_i, \;\; i = 1, \ldots, n, \tag{4.25}$$

$$y_i \geq \gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j - u_i(1 - x_i), \;\; i = 1, \ldots, n, \tag{4.26}$$

$$y_i \leq \gamma x_i + \sum_{j=1}^{n} (a_{ij} - \gamma) x_j - l_i(1 - x_i), \;\; i = 1, \ldots, n, \tag{4.27}$$

$$x_i \in \{0, 1\}; \;\; y_i \in \mathbb{R}, \; i = 1, \ldots, n. \tag{4.28}$$

The proposed formulations allow to use standard optimization solvers to find optimal γ-cliques in graphs that are not very large.

4.5 Results of Numerical Experiments

To provide a preliminary evaluation of the relative practical efficacy of the proposed mathematical programming formulations, sample numerical experiments have been conducted using a state-of-the-art commercial solver. All experiments were performed on a Dell Optiplex 980 PC with Intel Core i7 CPU 860 2.80 GHz processor, 8 GB RAM, running 64-bit Windows 7 Professional operating system. The proposed formulations were used in conjunction with FICO Xpress-IVE Version 1.21.02 solver on a number of instances.

The testbed used included two types of instances: uniform random graphs on 50 and 100 vertices and power-law random graphs on 100 vertices. A uniform random graph $G(n, p)$ has n vertices, where each pair of vertices is connected by an edge independently with the probability p, whereas in a power-law graph the probability that a node has a degree k is proportional to $k^{-\beta}$. Generating test instances of uniform random graphs with given n and p is straightforward, whereas in the case of power-law graphs one can use the procedure described in [23], which essentially assigns the probabilities p_{ij} for each pair of nodes (i, j) to be connected, using the extended random graph model for a general degree distribution and then adjusting that model so that the resulting graph follows the power-law degree distribution. Tables A.1 and A.2 in Appendix A present the description of the uniform random graphs and the power law random graphs used, respectively. In Table A.1, the first column specifies the name of the graph, while the second and third columns marked by "n" and "p" contain the number of vertices and probability used to generate the corresponding graph $G(n, p)$, respectively. The next column "m" contains the actual number of edges in the corresponding graph. The remaining columns show the computed γ-clique number $\omega_\gamma(G)$ for $\gamma = 1, 0.95, 0.9, 0.85, 0.8$, and 0.75. The only difference in notations used in Table A.2 compared to Table A.1 is in the third column, where the parameter β needed to generate a power law random graph is used instead of p.

The running times for the two proposed formulations applied to the above described uniform and power law random graphs are compared in Tables A.3 and A.4 of Appendix A, respectively. The first column of these tables, again, contains the graph name. The remaining eight columns are subdivided into four pairs corresponding to four reported values of γ; $\gamma = 1, 0.95, 0.85$, and 0.75, respectively. In each of the four pairs of columns, the first column, marked with "F1", shows the running time for formulation (4.10)-(4.13), and the second column, "F2", reports the running time for formulation (4.23)-(4.28). All running times are reported in seconds. If the MIP gap did not show much improvement after 50,000 seconds, the corresponding run was terminated with "$> 50,000$" reported in the respective table entry. One can observe that, in most cases, the running times for both formulations grow significantly with the increase of graph density and the decrease in γ value. While the second formulation consistently outperforms the first one for higher values of γ on the considered uniform random graphs, the first formulation takes over when $\gamma \leq 0.85$ and the graph's edge density is at least 0.15. The difference becomes dramatic on the last 10 graphs in Table A.3 and the last 5 instances in Table A.4, for which the second formulation requires over 50,000 seconds in multiple cases, while the first one often finds the solution in much shorter time spans and never takes more than 40,000 seconds.

To provide a deeper insight into the performance of the two formulations, Table A.5 of Appendix A presents a comparison of upper bounds for the MAXIMUM γ-CLIQUE problem. The first bound is based on analytical expression (4.1) (if the graph is not connected; such graphs are marked with *) or (4.2) (for connected graphs). The other two bounds are given by the optimal objective function value of LP relaxations for the first formulation (LPRF1), and the second formulation (LPRF2). Solving times for the LP relaxations are given in seconds. One representative problem instance from each subtype included in the testbed is used for the comparison. All uniform random graphs generated for the experiments were verified

to be connected, therefore, upper bound (4.2) applies. On the other hand, none of the power law random graphs in the testbed were connected, therefore, bound (4.1) was used for the corresponding two instances included in the table. One can observe that both LP bounds are of rather poor quality and could be improved by adding the constraints corresponding to the proposed analytical bounds. However, adding such constraints results in even higher running times, as finding high-quality feasible solutions becomes more challenging for the MIP solver. The LP bounds obtained from the MIP formulations are comparable, with the second formulation being slightly tighter in most cases. Moreover, the second formulation requires less time to compute. Surprisingly, the first formulation still comprehensively outperforms the second one on several instances, as reported in Tables A.3 and A.4. This is due to the fact that typically the number of branch-and-bound nodes explored by the solver for the first formulation is significantly lower than for the second formulation. For example, for graph u50-1 with $\gamma = 0.75$ the first formulation terminates at node 541, while the second formulation – at node 17,459. It should be noted that in the reported preliminary experiments we just used default solver settings. Perhaps more advanced branch-and-bound strategies, tailored specifically for the MAXIMUM QUASI-CLIQUE problem, may lead to significant speedups.

4.6 Conclusion

This section is the first attempt to establish rigorous mathematical foundations for the MAXIMUM γ-CLIQUE problem that finds numerous practical applications. We show that the decision version of the problem is NP-complete, develop analytical bounds on the γ-clique number of a graph, and provide mixed-integer programming formulations for the problem of interest. In addition, we report the results of preliminary computational study employing the proposed formulations in conjunction with a modern commercial MIP solver. The lack of well-defined structure in γ-cliques makes the problem extremely challenging for exact solution methods. The results on

small graph instances with up to 100 vertices underline the necessity of developing more advanced techniques in order to be able to solve larger-scale instances to optimality. The analytical bounds and MIP formulations proposed in this section could motivate future research on exact algorithms for the MAXIMUM QUASI-CLIQUE problem.

5. 2-CLIQUES ON UNIT DISK GRAPHS

In this section we investigate the problem of finding 2-cliques on unit disk graphs. The section is organized as follows. In Section 5.1, we motivate the problem by examining potential applications and by demonstrating how it can contribute to already completed research. In Section 5.2, we give the necessary definitions, notation, and background to understand the paper. In Section 5.3, we outline the proof that 2-cliques are 4-dominated on unit disk graphs, which is completed in Section 5.5, and provide an example to show this bound is nearly tight. In Section 5.4, we discuss how to solve the 2-clique problem effectively on unit disk (UD) graphs using our proof, ultimately establishing a $\frac{1}{2}$-approximation ratio for our polynomial time algorithm, as well as how to the solution performs on random unit disk graphs, showing both theoretical and computational results.

5.1 Related Research and Applications

As discussed in the introduction, clique relaxations and unit disk graphs merit research independent of each other. However, at the intersection of these two areas of research are some interesting problems. Routing on dynamic broadcasting networks is one such problem. A proposed method for highly stable routing in a dynamic network is to partition a graph into cliques, treating each clique as a super-vertex in the graph [37]. As nodes move around, it is unlikely they break communication with all vertices in their group before the graph can be re-examined and re-partitioned into cliques. When this is the case, routing tables only have to be updated within each group, since the general route remains fixed, consuming less bandwidth and allowing the network to scale to handle a higher capacity of nodes. Clique relaxations could be substituted for cliques in this algorithm and would constitute a trade of slightly less stability in routing for the ability to scale the network to even larger sizes.

Another problem that might be assisted by identifying 2-cliques in a unit disk graph is flight scheduling. The hub and spoke model has been the dominant route structure for most airlines since flights were deregulated in 1978. This model helps ensure full flights and gets most customers to their destination with only one layover, but it lacks flexibility. Inclement weather at a hub or other delays can make re-scheduling quite difficult. A hub and all its spokes, in the context of graph theory, constitute a 2-clique in a unit disk graph, where the radius of the disks represent the flight capabilities of a plane or perhaps restrictions on a crew. To take pressure off some of the hubs, replace the centralized 2-cliques, consisting of a hub and its spokes, in the unit disk graph with 2-cliques found by the method in this paper. All flights scheduled between cities in a 2-clique still get customers to their destination with one layover. Further, our 2-cliques will have at most two dominating vertices, which can be used as hubs for the traditional hub and spoke method when there are no delays.

The solutions to both the routing and airline scheduling problems are better when the clique relaxation found is large. In the routing problem, the more vertices we place in each super-vertex the larger the network can scale with fixed bandwidth capabilities. Having more cities included in our diameter 2 clique relaxation in flight scheduling gives us more flexibility in scheduling and more insurance of full flights while still getting customers to their destination with only one layover. We suggest 2-cliques specifically to solve these problems because we can find large 2-cliques on unit disk graphs very effectively, as we will show in this section.

Considerable research has already been invested into clique relaxations on unit disk graphs. In [8] the author emphasizes the importance of identifying the complexity of this set of problems:

> *Complexity of maximum k-plex, k-club, and k-clique problems on restricted graph classes such as planar and perfect graphs is important,*

> *even more so on graph classes that have practical applicability such as unit disk graphs.*

One reason identifying the complexity of the aforementioned problems is so important is assumptions are being made that may or may not be true. For instance, in [36] the author writes of the k-clique problem:

> *There are some strong indications that this problem is NP-hard in unit disk graphs. We claim this as a conjecture and give complexity results for some geometric graphs which may be helpful in verifying our conjecture.*

The complexity of the k-clique problem on unit disk graphs remains open. Research into k-cliques on unit disk graphs would most certainly help give validity to many of the methods employed in [36]. Instead of tackling the whole problem, we will focus our attention on the 2-clique problem for unit disk graphs, with the hope that our findings will alleviate the proof of the complexity of k-cliques on unit disk graphs in general.

5.2 Definitions and Notation

We introduce a few definitions not already in the literature that will help in the discussion that follows.

Definition 5.2.1 *For any set of pairwise overlapping disks centered at c_1, c_2, and c_3, define their circular triangle $\triangle c_1c_2c_3$ to be the shape outlined by p_1, p_2, and p_3 and the boundaries of the disks, where p_i is the intersection of the disks centered at c_j and c_k closest to the disk centered at c_i as in Fig. 5.1. We will refer to p_i as the vertices of this triangle and the pairwise intersections of disks with the circular triangle removed as leaves.*

Note that the circular triangle may not be contained within any of the disks if they do not share a mutual intersection. We will often call this a *concave circular triangle*,

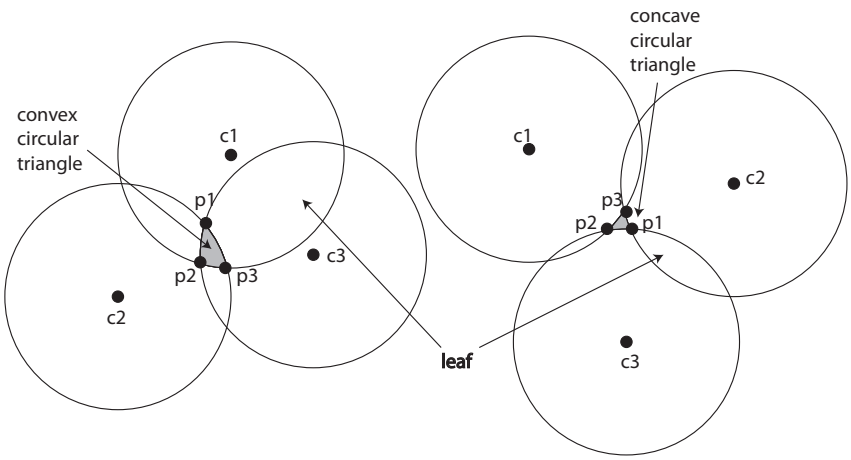

Fig. 5.1. Circular triangles.

just to clarify we are dealing with the case where three circles do not have a mutual overlap.

Part of our proof will make use of a set of points in space S overlapped by all disks in a set of disks. We next define an object relative to this set S to help in our proof.

Definition 5.2.2 *Suppose S is a set of points in space overlapped by all disks and suppose the boundary of S is formed by n different circles. We will refer to these circles as border circles, as in Fig. 5.2.*

Definition 5.2.3 *The intersection $A \cap B$ of any pair of disks $\{A, B\}$ will be referred to as a lens. The extreme points of this intersection will be referred to as the vertices of the lens.*

Note that a lens is always the full intersection of a pair of disks and thus in some cases differs from a leaf.

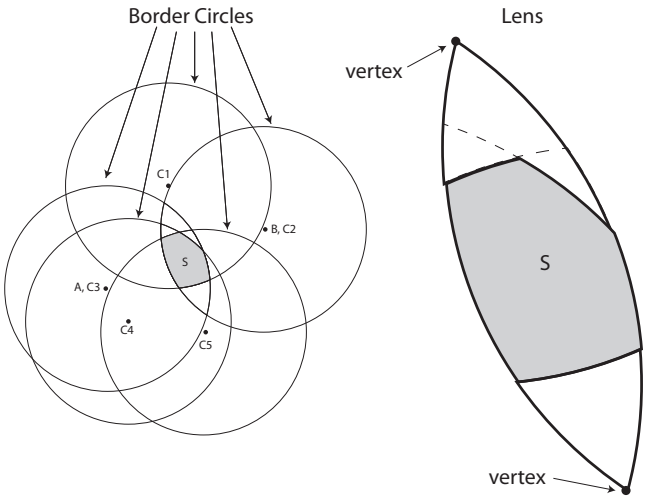

Fig. 5.2. Border circles and lens.

5.3 Domination for 2-Cliques on Unit Disk Graphs

We outline a proof that 2-cliques are 4-dominated on unit disk graphs, which is completed in Section 5.5. This is not true for 2-cliques on general graphs. It is shown in [42] how to construct graphs of diameter 2 with minimum dominating set exceeding any size n. This means that in order to prove 2-cliques are 4-dominated on unit disk graphs, we must take advantage of the extra structure we have with unit disk graphs.

Let K be any 2-clique in a unit disk graph. A key detail to note is that we do not require the elements in a dominating set for K to be members of K. Another important note is that we will be working exclusively with the containment model for UD graphs. By working with the containment model, every pair of disks A and B in our 2-clique K must intersect and there must be a vertex in their intersection to ensure they have distance at most 2. Recognizing all intersections of disks in K must contain a vertex of the graph, we will break our proof down into two cases. The first case is where there exist three disks A, B, and C in K that intersect pairwise

but $A \cap B \cap C = \emptyset$, meaning there is not a mutual intersection for all three disks. The other case is where all disks A, B, and C in K satisfy $A \cap B \cap C \neq \emptyset$.

5.3.1 Case 1: $\exists\ A, B, C \in K\ s.t\ A \cap B \cap C = \emptyset$

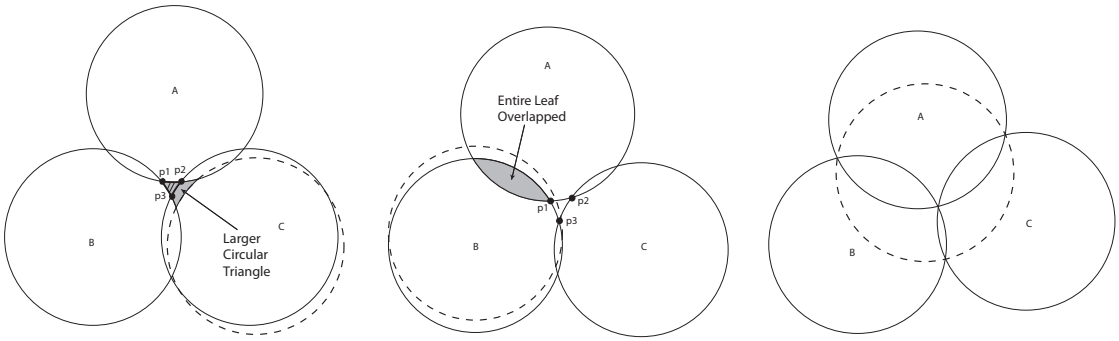

Fig. 5.3. An illustration to the proof of Case 1.

The general idea of the proof in this case is first to take the concave circular triangle between three circles of the 2-clique that has largest area. Suppose A, B, and C are the circles forming the border of this circular triangle. It is clear from Fig. 5.3 that every other disk in the 2-clique must overlap at least one of its three vertices p_1, p_2, or p_3. Otherwise it would produce a circular triangle with larger area as in the figure, a contradiction to how we chose A, B, and C. We then prove that, since every disk must overlap one vertex of this triangle, they must in fact overlap at least one entire leaf $A \cap B$, $A \cap C$, or $B \cap C$. This is given credibility by the fact that even when the three vertices p_1, p_2, and p_3 are concurrent, making the outer tips of the leaves as spread apart as possible in Case 1, a disk cannot squeeze in between the 3 leaves. This special case, as pictured at right in Fig. 5.3, is proven in [35]. Since the leaves $A \cap B$, $A \cap C$, or $B \cap C$ must contain vertices of the graph for A, B, and C to be distance 2 apart, and since all disks in K must overlap one of these

three leaves entirely, all disks are connected to one of these 3 vertices. Hence, K is 3-dominated in this case.

5.3.2 Case 2: $A \cap B \cap C \neq \emptyset \ \forall \ A, B, C \in K$

A key aspect to the proof in this case is a well-known theorem from convex geometry.

Theorem 5.3.1 (Helly's Theorem in Two Dimensions [25]) *Suppose F is a finite family of at least 3 convex sets in $\Re^2$. Then if every 3 members of F have a common point, there is a point common to all members of F.*

Since in case 2 we are assuming $A \cap B \cap C \neq \emptyset \ \forall A, B, C \in K$, by Helly's theorem this means there exists a set of points in space S that all members of K overlap, as shown in Fig. 5.4.

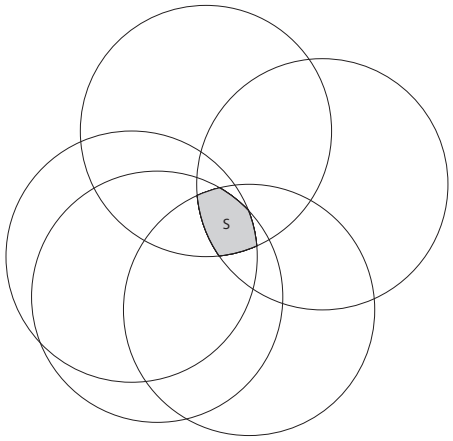

Fig. 5.4. The area guaranteed by Helly.

Clearly, if there is a vertex in S, the 2-clique is 1-dominated. If instead there are two vertices in a lens formed from two discs on the border of S that are separated by S, as in Fig. 5.5, the 2-clique is 2-dominated. There is no way to squeeze between these two vertices without changing the border of S, which would be a contradiction

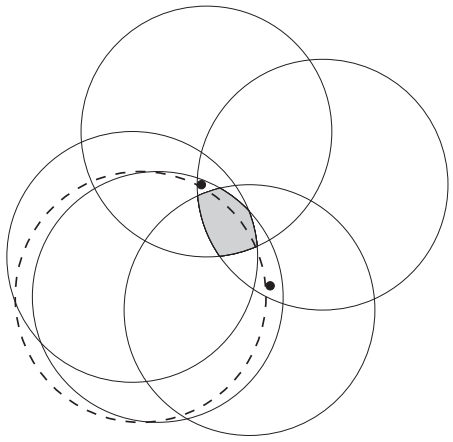

Fig. 5.5. A border lens with 2 vertices.

since all disks must overlap S entirely. Assuming neither of these are true, we are in a third case, pictured in Fig. 5.6. In this case, we will use a finite sequence of steps

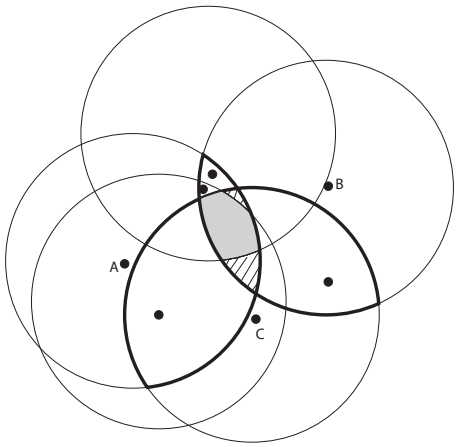

Fig. 5.6. Discs A, B, and C with no mutual neighbor.

to identify three disks A, B, C that border S in K such that no vertex of the graph lies inside $A \cap B \cap C$, that is $N(A) \cap N(B) \cap N(C) = \emptyset$. To produce this set of disks, we choose a pair of disks A and B that do not make up consecutive pieces of the border of S and consider their intersection, as in Fig. 5.7. Since A and B belong to

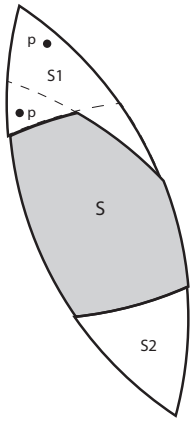

Fig. 5.7. The lens formed from A and B.

the 2-clique K, there must be a node(s) p in $A \cap B$ that cannot be in S. Further, since A and B are border disks for S that are not consecutive, S will divide $A \cap B$ into two pieces, S_1 and S_2, only one of which can contain a node p or else the graph would be 2-dominated, as we established previously. Suppose these points lie in S_1. Starting at any node p, we choose a disk C bordering S that does not contain p, hoping that $N(A) \cap N(B) \cap N(C) = \emptyset$. If that is not the case, we replace either A or B with C, and repeat. Since every repetition reduces the number of border disks between A and B by at least one, we eventually will produce two border disks A and B close enough together on the border of S such that there exists a point p in $A \cap B$ but the disk C separating the node p from S does not contain any node in $A \cap B \cap S_1$. But the set of points $\{p \mid p \in A \cap B \cap S_1\} = \{p \mid p \in A \cap B\}$ since the set is not 2-dominated and hence C cannot overlap any node of $A \cap B$. Thus we have produced $N(A) \cap N(B) \cap N(C) = \emptyset$, which we needed to know so that each of $A \cap B$, $A \cap C$, and $B \cap C$ must contain a vertex of the graph, similar to Case 1. Call these vertices v_1, v_2, and v_3 respectively.

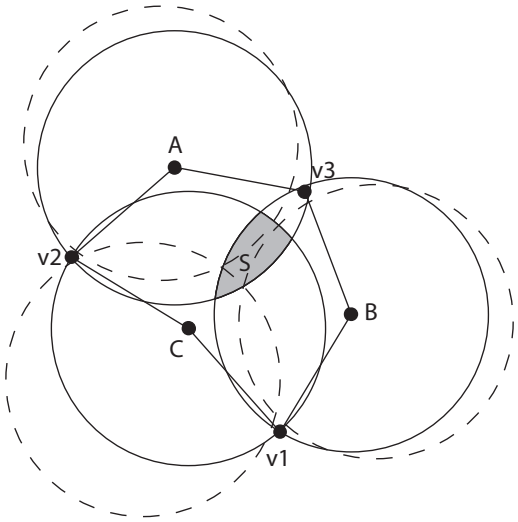

Fig. 5.8. An illustration of the hexagon forming a border around S.

It is obvious there is no way to "squeeze" a disk between v_1, v_2, and v_3 with a point outside the hexagon pictured in Fig. 5.8 without making either A, B, or C be no longer a part of the border of S. Since every point must overlap S, this means every point in K is either connected to v_1, v_2 or v_3 directly or else inside this hexagon. It is possible that a point D inside this hexagon avoids v_1, v_2, and v_3. The disk centered at D in Fig. 5.9 is one such example.

We show that any point that does not connect to one of v_1, v_2, or v_3 inside the hexagon is connected to one other vertex, making the graph 4-dominated.

5.3.3 A Lower Bound on the Domination Number

In [33] it is proven that for a set of congruent disks that intersect pairwise, the piercing number, which is the fewest points in space that intersect every object in a given set, is precisely 3. They give an example similar to the one in Fig. 5.10, which is discussed in detail in [21].

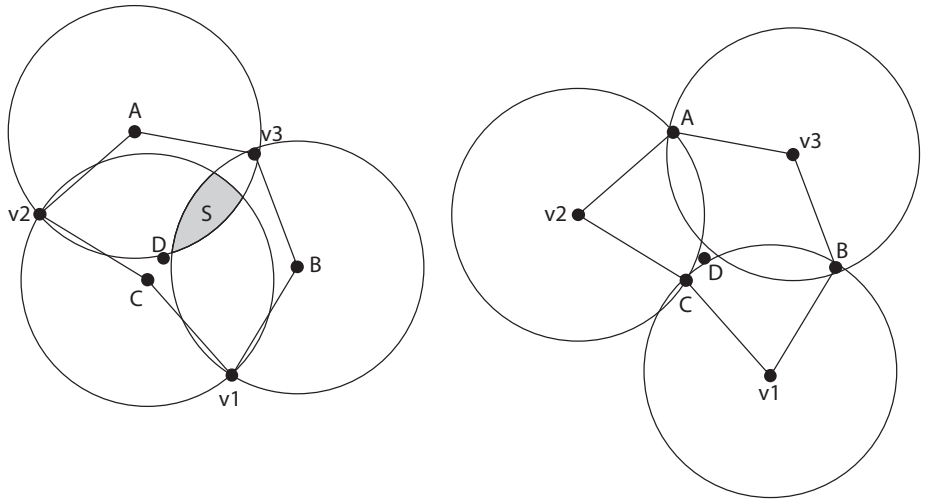

Fig. 5.9. A disc inside the hexagon that avoids vertices v_1, v_2, and v_3.

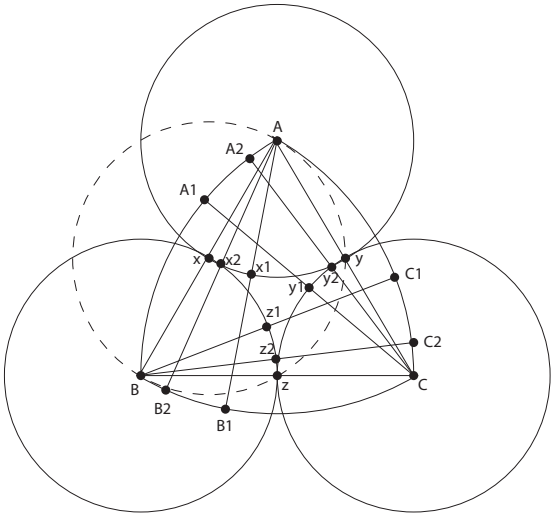

Fig. 5.10. A 2-clique with a minimum dominating set of 3 vertices.

In [21], the sequences of points $\{A_i\}_{i=1}^n$, $\{B_i\}_{i=1}^n$, and $\{C_i\}_{i=1}^n$, are created such that no unit disk contains more than $\frac{1}{3}|\{\{A_i\}_{i=1}^n, \{B_i\}_{i=1}^n, \{C_i\}_{i=1}^n\}|+1$ of them. That example is adapted here so that the path AB_i contains midpoint x_i, the path BC_i contains midpoint z_i, and the path CA_i contains midpoint y_i. These midpoints are

necessary because we are dealing with a set of disks that form a 2-clique rather than a clique and we are using the containment model for unit disk graphs. If $V_n = \{\{A_i\}_{i=1}^n, \{B_i\}_{i=1}^n, \{C_i\}_{i=1}^n, \{x_i\}_{i=1}^n, \{y_i\}_{i=1}^n, \{z_i\}_{i=1}^n\}$, then in the adapted example it is true that no two disks contain and hence dominate more than $\frac{5}{6}|V_n| + 1$ points. The set as a whole forms a 2-clique and so the lower bound on domination number for 2-cliques in UD graphs is 3.

5.4 Effectively Finding 2-Cliques on Unit Disk Graphs

Theorem 5.4.1 *The maximum 2-clique problem can be solved in polynomial time on unit disk graphs with a $\frac{1}{2}$-approximation ratio.*

Proof Let G be our unit disk graph. We claim we can find the largest 2-clique dominated by 2 elements in any graph in polynomial time. First, extract a pair $\{v_1, v_2\}$ of vertices and their neighbors and square the extract to produce a co-bipartite graph with partitions $N[v_1]$ and $N[v_2]$. Note that as we square the extract, we connect vertices that are distance 2 or less in G, even if they are not distance 2 in the subgraph induced by our extract. The result is still a co-bipartite graph, and the maximum clique problem can be solved on such graphs in polynomial time [24]. A clique in the square of a graph by definition is a 2-clique in the original graph and hence by extracting the closed neighborhoods of all subsets $\{v_1, v_2\}$ of size 2, we can identify the largest 2-clique dominated by 2 elements in the graph in polynomial time. For ease below, we will define this method to be called the *extraction method*.

We claim that the largest 2-clique dominated by 2 elements must be at least half the size of the largest 2-clique. To see this, note k-cliques are weakly hereditary, meaning every subgraph of a k-clique will be a k-clique. This is because the distance between two elements in a subset is the shortest path between the elements, and that path is not restricted to only use elements in the subset. Thus, while each subset of a k-clique will have a different induced subgraph, the distance between

elements in those subsets remains k, and hence, every subset is a k-clique. Next note that at least half of the vertices of the largest 2-clique will be dominated by 2 elements in the graph. We showed in Section 5.3 that all 2-cliques are 4-dominated on unit disk graphs, and hence, at least half of the elements must be connected to two of the members of the dominating set. Since these two dominating points and all their neighbors form a 2-clique independently by weak heredity, and such a 2-clique would be detected by our extraction method, the largest 2-clique produced by the extraction method must be at least half as large as the largest 2-clique, and the proof is done. ■

Theorem 5.4.2 *With asymptotic probability 1, the largest 2-clique in random unit disk graphs can be found with $\frac{2}{3}$-approximation ratio in polynomial time.*

Proof In [34], it is proved that given a set of random points in a punctured unit disk, with asymptotic probability 1, there exist two points that will cover all the points in the unit disk. In the case where Helly's theorem established the existence of a set S overlapped by all disks, we can take any point in S as the center of a punctured unit disk that will cover all points of the 2-clique, since all members of the 2-clique are within the circle of radius 1 of every point in S. In this case, with asymptotic probability 1, the set of disks in such a 2-clique are 2-dominated. In the other case, where $A \cap B \cap C = \emptyset$, we proved the graph 3-dominated, rather than 4-dominated. Combining these results we can say that our solution to the 2-clique problem is, with asymptotic probability 1, a solution with $\frac{2}{3}$-approximation ratio to the 2-clique problem on random unit disk graphs. ■

While we can guarantee with asymptotic probability 1 that the largest 2-clique in a random unit disk graph can be found with 2/3-approximation ratio, the algorithm actually performs even better in practice. We generated 3500 random unit disk graphs of 50 nodes and 100 random unit disk graphs of 100 nodes for each density in the range from .05 to 1 in increments of .05. In order to build our random unit

disk graphs $G(n, p)$ with the specified edge probability p, we made use of the results established in [50]. In [50] a formula for the probability distribution for distance between random points in a box is established. We use the formula in reverse, starting with our fixed probability p and using the distribution to identify the radius necessary for any pair of disks to overlap in a fixed size box with probability p. In all 70,000 experiments with 50 nodes and all 2000 experiments with 100 nodes, the size of the largest 2-clique and the largest 2-clique generated by our algorithm matched. Practically speaking, the 2-clique problem has been solved for random unit disk graphs in polynomial time.

5.5 Proof that 2-Cliques are 4-Dominated on UD Graphs

Our proof assumes we have a 2-clique K and is broken into the two cases as outlined in Section 5.3. The following lemmas are basic results about intersecting circles with equal radius that are needed in both cases.

Lemma 5.5.1 *In Fig. 5.11, disks A, X, and Y all have the same radius. If the intersection points $\{x_1, x_2\}$ of X and A are both inside circle Y, then $A \cap X \subseteq A \cap Y$. If the intersection points $\{y_1, y_2\}$ of circles A and Y are both outside of circle X, then either $A \cap Y \supseteq A \cap X$ or else these intersections don't overlap, i.e. $(A \cap X) \cap (A \cap Y) = \emptyset$.*

Proof Suppose the intersection points $\{x_1, x_2\}$ of X and A are both inside circle Y. First note that given the radius of a circle and two points on the circle, we can set up a system of two equations to solve for two unknowns (h, k) representing the center of the circle. It is not difficult to see that since they have the same radius, this will yield at most two potential solutions for the centers, one on each side of the line between the two given points on the circle and curving in opposite directions.

Next note that disks X and Y either do not intersect at all or intersect twice within A. Two circles can intersect 0, 1, or 2 times. In order for extreme points

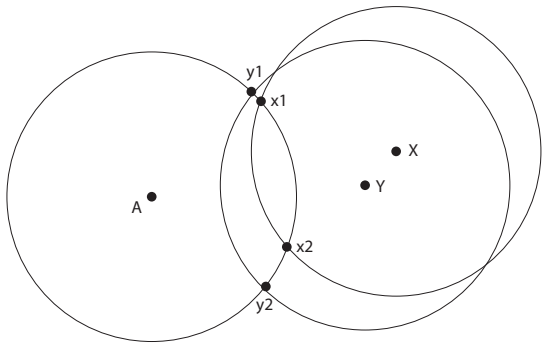

Fig. 5.11. An illustration to the proof of Lemma 5.5.1.

x_1 and x_2 of the intersection of A with X to be both within Y, the disk X must cross the boundary of Y an even number of times within A, which means either 0 or 2 intersections. Suppose X and Y intersect twice within A, at points p_1 and p_2. Since X and Y have the same radius, this means they must be centered on opposite sides of the line between p_1 and p_2 as we argued above. But this means X and Y are curving in opposite directions, which means $\{x_1, x_2\}$, the intersections of X with A, cannot be contained within Y, a contradiction. If they were contained within Y, then X must intersect $A \cap Y$ four times and hence have diameter less than the maximum distance of points in $A \cap Y$. This contradiction means X and Y must not intersect within A. Since the boundaries of disks are continuous and X and Y do not intersect within A, this means either $A \cap X \subseteq A \cap Y$, $A \cap Y \subseteq A \cap X$, or $(A \cap X) \cap (A \cap Y) = \emptyset$. Since $x_1 \in A \cap X$ also belongs to $A \cap Y$ and $y_1 \notin A \cap Y$, it must be that $A \cap X \subseteq A \cap Y$.

The proof for the case that the intersection points $\{y_1, y_2\}$ of circles A and Y are both outside of circle X is analogous, with the conclusion again that either $A \cap X \subseteq A \cap Y$, $A \cap Y \subseteq A \cap X$, or $(A \cap X) \cap (A \cap Y) = \emptyset$. In this case, however, we can only include either $A \cap X \subseteq A \cap Y$ or $(A \cap X) \cap (A \cap Y) = \emptyset$ because we are not aware of any point in $A \cap X$ necessarily in $A \cap Y$. ■

Lemma 5.5.2 *Suppose two disks A and B of radius r intersect to form a lens $A \cap B$. Then any disk C of radius r intersects the boundary of the lens $A \cap B$ at most twice.*

Proof We showed in the proof of Lemma 5.5.1 that if C intersects the boundary of $A \cap B$ twice along A, then C cannot intersect B; or if C intersects it twice along B, it cannot intersect A. But in order to intersect the lens more than twice it must intersect one of A or B twice, which contradicts the proof of Lemma 5.5.1. ■

Corollary 5.5.3 *Suppose two disks A and B of radius r intersect to form a lens $A \cap B$. Let v_1 and v_2 be the intersections of the circles A and B. Any disk C of radius r that does not overlap v_1 and v_2 cannot intersect the boundaries of both A and B within $A \cap B$.*

Proof By Lemma 5.5.2, C can only intersect the lens $A \cap B$ twice. If it intersects the boundaries of both A and B within $A \cap B$, it by necessity must contain one of the vertices of the lens to be a closed object. ■

The previous result shows there is no way to "squeeze" a disk between the vertices of a lens and intersect both disks forming the lens beyond those points contained in the lens.

5.5.1 Case 1: $\exists\ A, B, C \in K\ s.t\ A \cap B \cap C = \emptyset$

We begin with a simple observation for Case 1, since we know there exist three disks without a mutual intersection.

Lemma 5.5.4 *Let A, B, and C be the members of K that form the concave circular triangle with largest area. Let p_1, p_2, and p_3 be the vertices of this triangle. Then any disk centered outside of triangle $p_1p_2p_3$ not coinciding with one of A, B, or C must properly contain at least one of p_1, p_2, or p_3.*

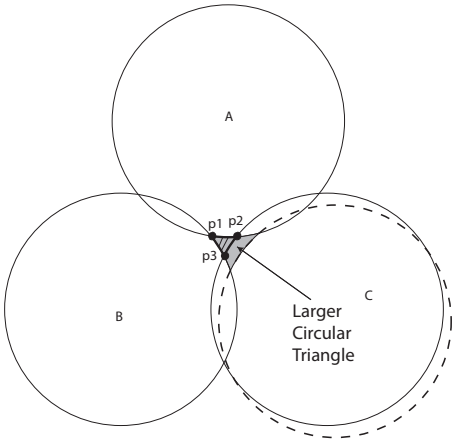

Fig. 5.12. An illustration to the proof of Lemma 5.5.4.

Proof We claim any disk X that does not properly contain p_1, p_2, or p_3 will form a concave circular triangle with larger area, which contradicts our choice of A, B, and C. By Lemma 5.5.1, a disk centered outside of triangle $p_1p_2p_3$ cannot intersect any side of triangle $\triangle p_1p_2p_3$, provided the curvature is oriented in the same direction, without overlapping at least one of the intersection points p_1, p_2, or p_3. But this means either X has opposite curvature from all sides of the concave circular triangle and is centered within $\triangle p_1p_2p_3$ or else it must form a concave circular triangle that contains triangle $\triangle p_1p_2p_3$ within it as in Fig. 5.12. This is because X must intersect all of A, B, and C as a member of K and hence forms a circular triangle of its own, but cannot intersect $\triangle p_1p_2p_3$ except at p_1, p_2, or p_3 since it cannot intersect a side of $\triangle p_1p_2p_3$ as explained above. Since it does not properly contain any of p_1, p_2, or p_3 and does not coincide with any of A, B, or C, the disk X must form a concave circular triangle with strictly larger area than that formed by A, B, and C. But this contradicts our choice of A, B, and C as forming the concave circular triangle with largest area, and the proof is done. ■

We now present a few more results about intersecting circles with equal radius.

Lemma 5.5.5 *If three disks A, B, and C overlap pairwise but $A \cap B \cap C = \emptyset$, then the triangle formed by connecting their centers will be acute.*

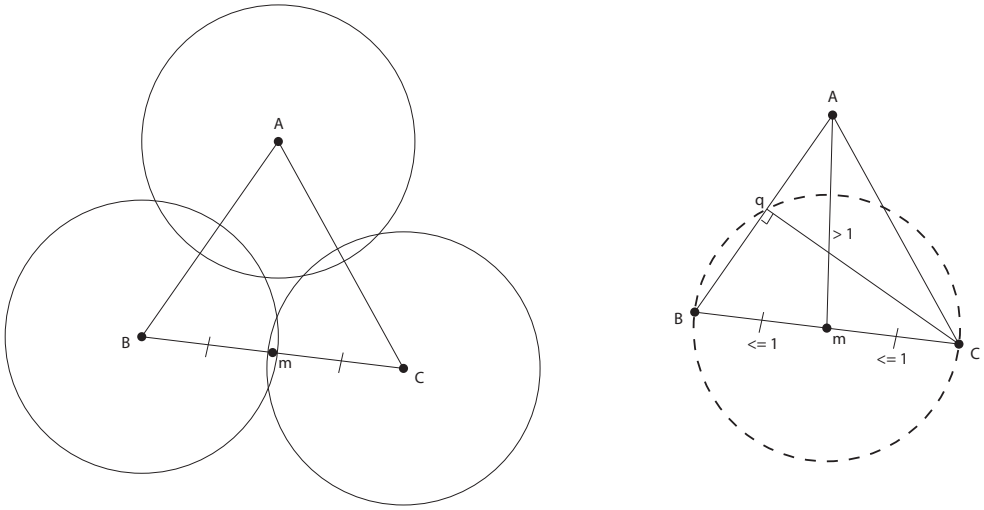

Fig. 5.13. An illustration to the proof of Lemma 5.5.5.

Proof Consider the midpoint m of side BC, which lies in the intersection $B \cap C$ as in Fig. 5.13. Since $A \cap B \cap C = \emptyset$, it must be that A is greater than distance 1 from m. Consider the circle centered at m of radius $|mC|$. Then the center of A cannot be in this circle. Consider the point q where side BA intersects disk m. The angle $\angle BqC$ then is a right angle. But this means $\angle BAC$ is acute because triangle $\triangle CqA$ has right angle $\angle CqA$. By similar arguments we can conclude angles $\angle BCA$ and $\angle ABC$ are acute and hence $\triangle ABC$ is acute. ■

We next give a lemma describing what happens as a circle rotates around an intersection point.

Lemma 5.5.6 *In Fig. 5.14, all circles have the same radius, and circles c and d both intersect circle a at a shared point s. If the center of circle d is on or inside $\angle csx$, then x is on or inside circle d, and y, z are on or inside circle c.*

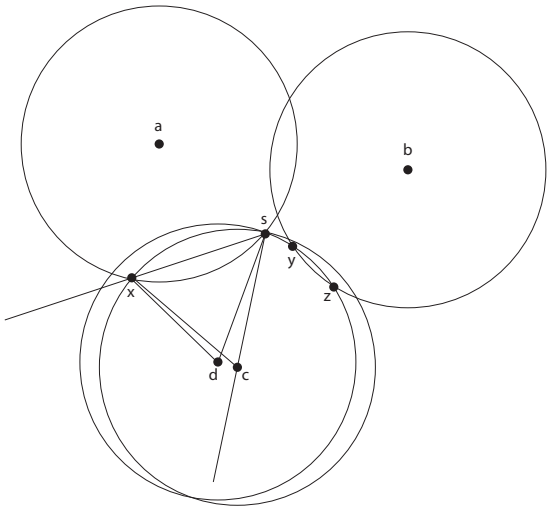

Fig. 5.14. An illustration to the proof of Lemma 5.5.6.

Proof $\angle csx = \angle cxs$ since $|cs| = |cx|$. If d is outside $\triangle csx$ then since d is on or inside $\angle csx$, this implies $\angle dxs > \angle cxs = \angle csx > \angle xsd$. Thus it must be that $|dx| < |ds| = 1$ since side ds is across from the larger angle. On the other hand, if d is on or inside $\triangle csx$, then clearly $|ds| + |dx| \leq |cs| + |cx|$ since the triangles share side sx, and since $|cs| = |ds| = 1$, it must be that $|dx| \leq |cx| = 1$. Thus in either case x is on or inside circle d since it has distance less than or equal to 1 from x.

We can conclude y and z are on or inside circle c using the exact same argument with c on or inside $\angle ds*$, where $*$ represents either y or z, which must be true since $\angle csx + \angle cs* = \angle xs* = \angle dsx + \angle ds*$ and we hypothesized $\angle csx > \angle dsx$. ■

We break down the remainder of our proof into three cases. First we examine the case where a disk properly contains none of p_1, p_2, or p_3. Following that, we handle the case where the disk contains at least one of p_1, p_2, or p_3 properly inside it but only overlaps part of $\triangle p_1p_2p_3$. Finally we will handle the case where a disk overlaps all of $\triangle p_1p_2p_3$. We will prove in all possible scenarios that an entire leaf is overlapped, implying that K is 3-dominated.

Lemma 5.5.7 *Suppose A, B, and C are three circles in a 2-clique such that $A \cap B \cap C = \emptyset$ and produce the concave circular triangle with largest area. Let the vertices of this concave circular triangle be p_1, p_2, and p_3 as in Fig. 5.15. Then it is impossible for a disk Q of identical radius not coinciding with A, B, or C to not properly contain at least one of p_1, p_2, or p_3 inside of it.*

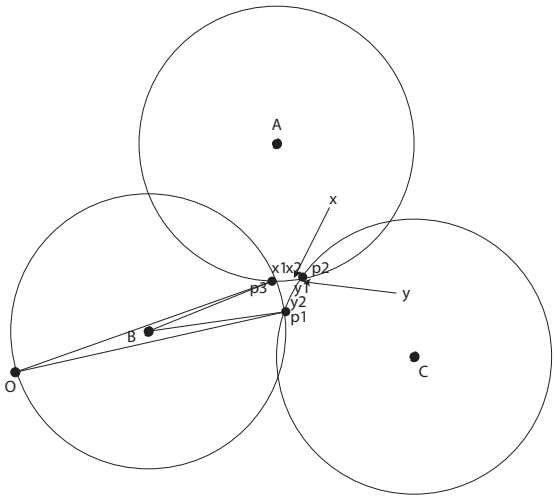

Fig. 5.15. Setup for the proof of Lemma 5.5.7.

Proof Since none of p_1, p_2, or p_3 are properly contained in Q, by Lemma 5.5.4 it must be that Q is centered within circular triangle $\triangle p_1 p_2 p_3$. Our strategy will be to build a sequence of disks that increase in area, starting with Q and ending with a disk of maximum possible size that still does not properly contain p_1, p_2, or p_3. We will show this final disk to have radius less than 1 and the proof will be done.

Let Q be an arbitrary circle with center inside $\triangle p_1 p_2 p_3$. Suppose Q intersects side $p_3 p_2$ at points x_1 and x_2 and intersects side $p_1 p_2$ at y_1 and y_2 as labeled in Fig. 5.15. Note that if x_1 coincides with p_3 or y_2 coincides with p_1, we skip one of the next two steps in our construction. Because x_1, x_2, y_1, and y_2 are all on one circle, we can conclude $\angle x_2 x_1 y_2 + \angle x_2 y_1 y_2 = \frac{\angle x_2 Q y_2}{2} + \frac{2\pi - \angle x_2 Q y_2}{2} = \pi$.

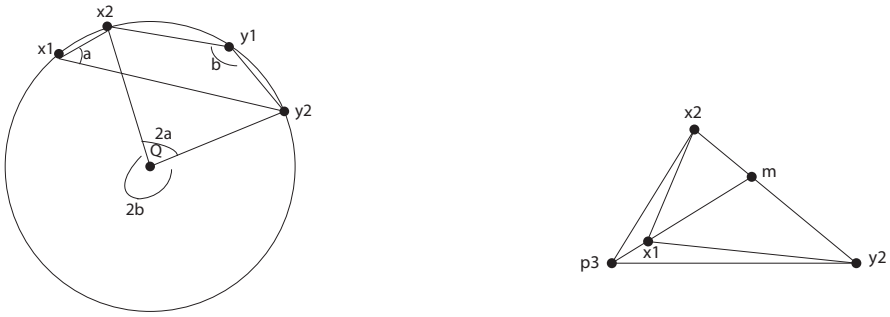

Fig. 5.16. The start of a sequence of circles with increasing area.

Without loss of generality (WLOG), we can assume $\angle x_2x_1y_2 \leq \frac{\pi}{2}$. We can calculate the radius of the circle through the four points to be $r_0 = \frac{x_2y_2}{\sin\angle x_2x_1y_2}$. Note from Fig. 5.16 that $\angle x_2p_3y_2 = \angle x_2p_3m + \angle mp_3y_2 < \angle x_2x_1m + \angle mx_1y_2 = \angle x_2x_1y_2$, so we can conclude the radius of the circle through x_2, y_2 and p_3 to be $r_1 = \frac{x_2y_2}{\sin\angle x_2p_3y_2} > \frac{x_2y_2}{\sin\angle x_2x_1y_2} = r_0$, that is greater than the radius of the circle through x_1, x_2, y_1, and y_2.

The circle through x_2, y_2 and p_3 must intersect circle C in a second location. Let y be this point. Note that since this circle intersects p_3p_2 twice, at p_3 and x_2, it cannot overlap p_2. Thus y must be on p_1p_2. Consider the circle through x_2, y_2, p_3 and y. Repeating the analysis from above, we can conclude that $\angle p_3y_2y + \angle p_3x_2y = \pi$. Assume WLOG that $\angle p_3y_2y \leq \frac{\pi}{2}$. Since $\angle p_3p_1y < \angle p_3y_2y$ using the same argument as before, we can conclude $r_2 = \frac{p_3y}{\sin\angle p_3p_1y} > \frac{p_3y}{\sin\angle p_3y_2y} = r_1$, since both angles $\angle p_3y_2y$ and $\angle p_3p_1y$ are between 0 and $\frac{\pi}{2}$ where sin is increasing. Thus the circle through p_3, p_1, and y will have greater radius than the circle through x_2, y_2, p_3 and y.

We now break down the argument into two cases. The first case is where $\angle p_1yp_3 \geq \frac{\pi}{2}$. Let O and I be points on B as pictured in Fig. 5.17, separated by p_1p_3. We know that $\angle p_1Ip_3 + \angle p_1Op_3 = \frac{\angle p_1Bp_3}{2} + \frac{2\pi - \angle p_1Bp_3}{2} = \pi$. Using the same trick as above, we see that $\angle p_1Ip_3 = \angle p_1Ik + \angle kIp_3 > \angle p_1yk + \angle kyp_3 = \angle p_1yp_3$. Thus $\angle p_1yp_3 + \angle p_1Op_3 < \pi$. Since $\angle p_1yp_3 \geq \frac{\pi}{2}$, this means $\angle p_1Op_3 < \frac{\pi}{2}$. It also means

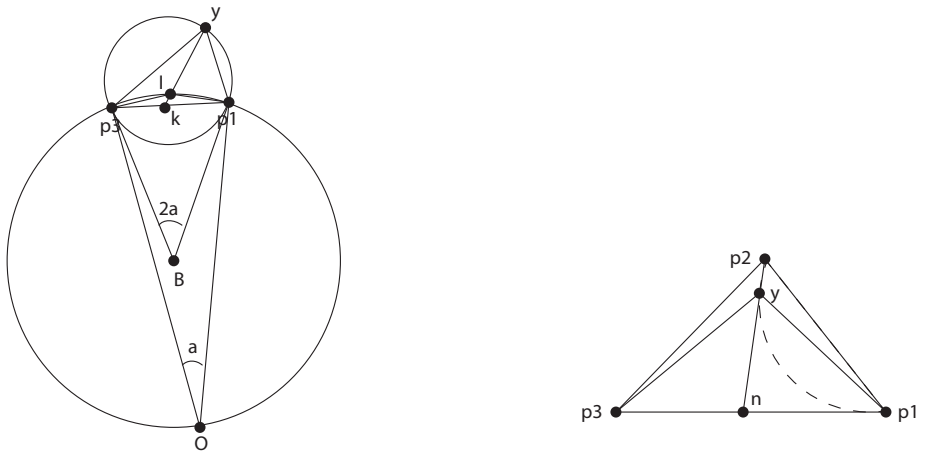

Fig. 5.17. Continuing the sequence of circles with increasing area.

$\angle p_1yp_3 < \pi - \angle p_1Op_3$. Thus we have $0 < \angle p_1Op_3 < \angle p_1yp_3 < \pi - \angle p_1Op_3 < \pi$, which means $\sin(\angle p_1yp_3) > \sin(\angle p_1Op_3)$. Thus $r_2 = \frac{p_1p_3}{\sin(\angle p_1yp_3)} < \frac{p_1p_3}{\sin(\angle p_1Op_3)} = 1$ since this last circle is B. The proof is done in this case because the circle through x_1, x_2, y_1, and y_2 had radius r_0 and we have shown $r_0 < r_1 < r_2 < 1$.

To handle the case $\angle p_1yp_3 < \frac{\pi}{2}$, note that since y is on arc p_1p_2, it must be inside $\triangle p_1p_2p_3$. Then, as in Fig. 5.17, $\angle p_1yp_3 = \angle p_1yn + \angle nyp_3 > \angle p_1p_2n + \angle np_2p_3 = \angle p_1p_2p_3$. Thus $0 < \angle p_1p_2p_3 < \angle p_1yp_3 < \frac{\pi}{2}$ in this case, so $\sin(\angle p_1p_2p_3) < \sin(\angle p_1yp_3)$. Thus the radius of the circle through p_1, p_2, and p_3 satisfied $r_3 = \frac{p_1p_3}{\sin(\angle p_1p_2p_3)} > \frac{p_1p_3}{\sin(\angle p_1yp_3)} = r_2$.

Assume that $\triangle p_1p_2p_3$ is acute. If not and $\angle p_1p_2p_3 \geq \frac{\pi}{2}$, then choose points I and O on circle B just as before, establish $\angle p_1p_2p_3 < \angle p_1Ip_3$ so that $\angle p_1p_2p_3 + \angle p_1Op_3 < \pi$, and the argument proceeds exactly as in the first case we argued above. The same could be done with circle A if $\angle p_3p_1p_2 \geq \frac{\pi}{2}$ and with circle C if $\angle p_1p_3p_2 \geq \frac{\pi}{2}$. Thus assume $\triangle p_1p_2p_3$ is acute. Assume WLOG that $\angle p_2Ap_3$ is the smallest in $\{\angle p_2Ap_3, \angle p_1Bp_3, \angle p_1Cp_3\}$. We want to prove $\angle p_2p_1p_3 > \angle p_2Ap_3$. If we can do this, then because $\angle p_2p_1p_3 < \frac{\pi}{2}$ as part of an acute triangle, we can

establish $\sin(\angle p_2p_1p_3) > \sin(\angle p_2Ap_3) > \sin(\frac{\angle p_2Ap_3}{2})$, which means $r_3 = \frac{p_2p_3}{\sin(\angle p_2p_1p_3)} < \frac{p_2p_3}{\sin(\frac{\angle p_2Ap_3}{2})} = 1$, and the proof will be finished by the chain $r_0 < r_1 < r_2 < r_3 < 1$.

To prove $\angle p_2p_1p_3 > \angle p_2Ap_3$, note that $\angle p_2p_1p_3 = 2\pi - \angle Bp_1C - \angle Bp_1p_3 - \angle Cp_1p_2$. Using $\angle Bp_1C = \pi - 2\angle p_1BC$, $\pi = 2\angle Bp_1p_3 + \angle p_1Bp_3$, which yields $\angle Bp_1p_3 = \frac{\pi}{2} - \frac{\angle p_1Bp_3}{2}$, and $\pi = 2\angle Cp_1p_2 + \angle p_1Cp_2$, which yields $\angle Cp_1p_2 = \frac{\pi}{2} - \frac{\angle p_1Cp_2}{2}$, we see that $\angle p_2p_1p_3$ simplifies to $2\angle p_1BC + \frac{\angle p_1Bp_3}{2} + \frac{\angle p_1Cp_2}{2}$. Then we have $\angle p_2p_1p_3 = 2\angle p_1BC + \frac{\angle p_1Bp_3}{2} + \frac{\angle p_1Cp_2}{2} \geq \frac{\angle p_1Bp_3}{2} + \frac{\angle p_1Cp_2}{2} \geq \angle p_2Ap_3$ since $\angle p_2Ap_3$ is the smallest in $\{\angle p_2Ap_3, \angle p_1Bp_3, \angle p_1Cp_3\}$. Since $\angle p_2p_1p_3 > \angle p_2Ap_3$, the proof is done. ■

Note that in our above proof, we showed that a circle that contains two of the three vertices from the set $\{p_1, p_2, p_3\}$ on its border, as well as a point y that could be the third point from this set, cannot have radius 1.

Lemma 5.5.8 *Let A, B, and C be three circles in our 2-clique K with $A \cap B \cap C = \emptyset$ and which produce a concave circular triangle with largest area. Then any other disk Q in K that does not strictly contain the entire circular triangle must strictly contain one of $A \cap B$, $A \cap C$, or $B \cap C$ entirely.*

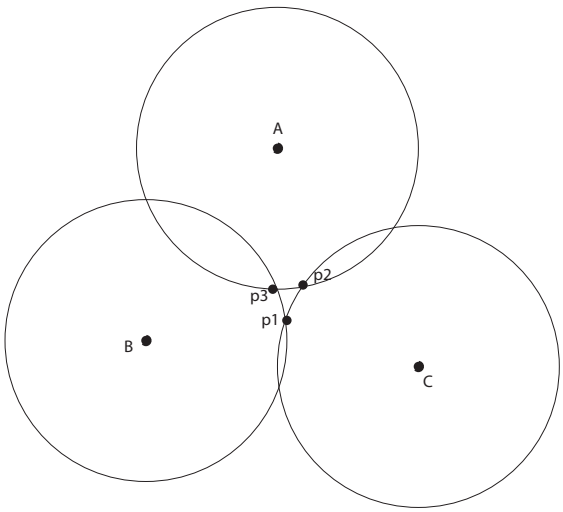

Fig. 5.18. The general setup for the proof of Lemma 5.5.8.

Proof Suppose the setup is as seen in Fig. 5.18, where A, B, and C are the 3 circles in our 2-clique K that form a concave circular triangle with most area, and p_1, p_2, and p_3 are the vertices of this triangle. Since the disk being added cannot strictly contain the entire circular triangle, there will always be at least one vertex that it will not strictly contain. We will let that vertex be p_2. In order to contain one of p_1 or p_3, any disk Q in K must intersect at least one of the sides p_1p_2 or p_2p_3 of triangle $\triangle p_1p_2p_3$ an odd number of times. Since disks can intersect each other at most twice, we can break down the proof into three essential cases, based off the number of times Q intersects each of p_1p_2 and p_2p_3.

In all cases we will let Q denote the disk we wish to add. We will let x_1 and x_2 denote the intersections of Q with A and y_1 and y_2 denote the intersections of Q with C. Note that x_1 can coincide with p_3; x_2 and y_1 can coincide with p_2; and y_3 can coincide with p_1. However, x_1 cannot coincide with p_3 at the same time when y_3 coincides with p_1 since one of p_1 or p_3 must be strictly contained in Q.

Subcase 1: Q intersects one of the edges p_1p_2 or p_2p_3 once and the other not at all.

Suppose WLOG that Q does not intersect p_1p_2 at all and hence intersects p_2p_3 exactly once. These together imply Q cannot overlap p_1 and thus must overlap p_3 only, as in Fig. 5.19. Since Q is part of the 2-clique, it must overlap disks A and C, and since it cannot overlap p_2, it forms a concave circular triangle with these disks as in the picture. This means $\triangle AQC$ will be acute by Lemma 5.5.5.

Note that $\angle x_1Qy_2$ is less than π. This is because $\angle x_1Qy_2 = \angle x_1Qx_2 + \angle x_2Qy_1 + \angle y_1Qy_2 = 2\angle x_1QA + \angle x_2Qy_1 + 2\angle y_2QC \leq 2\angle AQC < \pi$ since $\triangle AQC$ is acute. Clearly x_2 is outside of B, since circle Q must overlap p_3. If we can show x_1 is outside of B, we will be done by Lemma 5.5.1, because $\{p_3\} \in (A \cap Q) \cap (A \cap B)$ which implies that $A \cap Q \supseteq A \cap B$ and thus an entire leaf is overlapped.

To show x_1 is outside of B, first note $|Bx_2| > 1$. To make the notation less tedious in specifying angles for our argument, we label the vertices involved with a

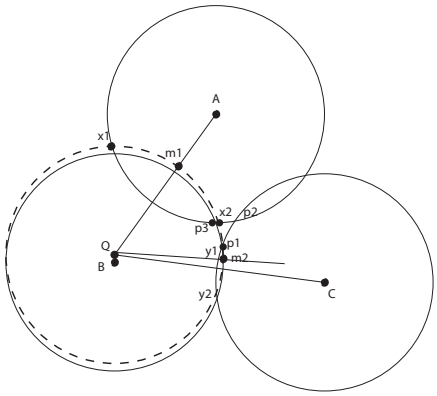

Fig. 5.19. An illustration for the proof of Subcase 1.

new polar coordinate system as in Fig. 5.20. We let Q be the origin of a new polar

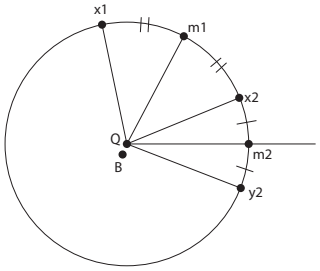

Fig. 5.20. Polar coordinate system in Subcase 1.

coordinate system, and the midpoint of the arc between x_2 and y_2 on circle Q, which is m_2, will be the point $(1, 0)$. Let x_1 be the point $(1, \alpha_1)$, y_2 be the point $(1, \alpha_2)$, m_1 be the point $(1, \beta)$, and B be the point (r, γ).

Since $\angle x_1 Q y_2 < \pi$, we can conclude $0 < \beta = \frac{\angle x_1 Q y_2}{2} < \frac{\pi}{2}$. Suppose by way of contradiction $|Bx_1| \leq 1$ and hence x_1 is not outside of B. Since we know $|Bx_2| > 1$, this means $|Bx_2| > |Bx_1|$. In order to be closer to x_1 than x_2, B must be on the same side of the line through m_1 and Q as x_1. Further $|Bx_2| > |By_2|$, since $y_2 \in B$. This means B must be on the same side of the line through m_2 and Q as y_2, that is, below the x-axis in our new coordinate system. Together these imply

that B must be in the third quadrant, since $\beta < \frac{\pi}{2}$. That is $\pi < \gamma < \frac{3\pi}{2}$. Note $0 < \alpha_1 = \angle x_1 Q m_2 < \angle x_1 Q y_2 < \pi$, and $0 < \angle m_2 Q y_2 = \frac{\angle x_2 Q y_2}{2} < \frac{\angle x_1 Q y_2}{2} < \frac{\pi}{2}$, which implies $\frac{3\pi}{2} < \alpha_2 < 2\pi$. Further note that $\angle BQy_2 = \alpha_2 - \gamma$ and $\angle x_1 QB = \gamma - \alpha_1$. Adding these last two equations together we get $\alpha_2 - \alpha_1 = \angle BQy_2 + \angle x_1 QB$. But $\alpha_2 - \alpha_1 = 2\pi - \angle x_1 Q y_2 > \pi$, so either $\angle BQy_2 > \frac{\pi}{2}$ or $\angle BQx_1 > \frac{\pi}{2}$. But $\angle BQy_2 > \frac{\pi}{2}$ would mean $|By_2| > |Qy_2| = 1$, which would be contradiction since y_2 is in B. Further $\angle BQx_1 > \frac{\pi}{2}$ would mean $|Bx_1| > |Qx_1| = 1$, which is a contradiction to what we supposed by way of contradiction. Thus it must be that $|Bx_1| > 1$ and then by Lemma 5.5.1, since x_1 is outside of B, all of the leaf $A \cap B$ is overlapped by Q.

Subcase 2: Q intersects p_1p_2 once and p_2p_3 once.

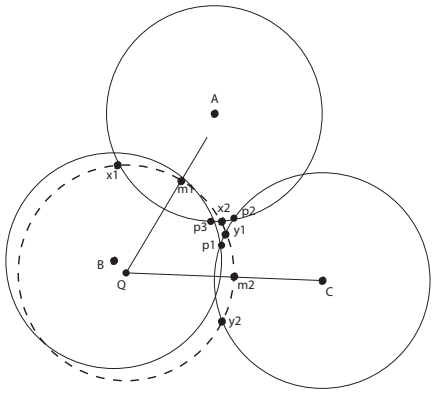

Fig. 5.21. An illustration for the proof of Subcase 2.

This case has very similar argument to Subcase 1 and is pictured in Fig. 5.21. Since Q cannot strictly overlap p_2, it forms a concave circular triangle with disks A and C. This means $\triangle AQC$ will be acute by Lemma 5.5.5.

Note that $\angle x_1 Q y_2$ is less than π because $\angle x_1 Q y_2 = \angle x_1 Q x_2 + \angle x_2 Q y_1 + \angle y_1 Q y_2 = 2\angle x_1 QA + \angle x_2 Q y_1 + 2\angle y_2 QC \leq 2\angle AQC$, and $2\angle AQC < \pi$ since $\triangle AQC$ is acute. Clearly x_2 is outside of B, since Q must overlap p_3, and y_1 is outside of B, since circle Q must overlap p_1. If we can show either x_1 or y_2 is outside of B, we will be done

by Lemma 5.5.1, because $\{p_3\} \in (A \cap Q) \cap (A \cap B)$ and $\{p_1\} \in (C \cap Q) \cap (C \cap B)$ implies either $A \cap Q \supseteq A \cap B$ or $C \cap Q \supseteq C \cap B$ and thus an entire leaf is overlapped.

The only way an entire leaf will not be overlapped is if $|Bx_1| \leq 1$ and $|By_2| \leq 1$. To make the notation less tedious in specifying angles for our argument, we again label the vertices involved with a polar coordinate system as in Fig. 5.22. We let Q

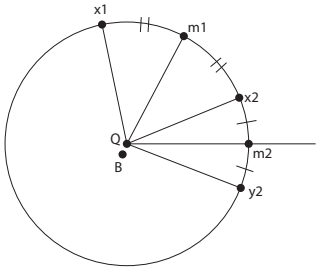

Fig. 5.22. Polar coordinate system in Subcase 2.

be the origin of a new polar coordinate system, and the midpoint of the arc between x_2 and y_2 on circle Q, which is m_2, will be the point $(1, 0)$. Let x_1 be the point $(1, \alpha_1)$, y_2 be the point $(1, \alpha_2)$, m_1 be the point $(1, \beta)$, and B be the point (r, γ).

Since $\angle x_1 Q y_2 < \pi$, we can conclude $0 < \beta = \frac{\angle x_1 Q y_2}{2} < \frac{\pi}{2}$. Suppose by way of contradiction both $|Bx_1| \leq 1$ and $|By_2| \leq 1$. Since we know $|Bx_2| > 1$, this means $|Bx_2| > |Bx_1|$ and $|Bx_2| > |By_2|$. In order to be closer to x_1 than x_2, B must be on the same side of the line through m_1 and Q as x_1. Likewise B must be on the same side of the line through m_2 and Q as y_2, which means it must be below the x-axis in our new coordinate system. Together these imply that B must be in the third quadrant, since $\beta < \frac{\pi}{2}$. That is $\pi < \gamma < \frac{3\pi}{2}$. Note $0 < \alpha_1 = \angle x_1 Q m_2 < \angle x_1 Q y_2 < \pi$, and $0 < \angle m_2 Q y_2 = \frac{\angle x_2 Q y_2}{2} < \frac{\angle x_1 Q y_2}{2} < \frac{\pi}{2}$, which implies $\frac{3\pi}{2} < \alpha_2 < 2\pi$. Further note $\angle BQy_2 = \alpha_2 - \gamma$ and $\angle x_1 QB = \gamma - \alpha_1$. Adding these last two equations together we get $\alpha_2 - \alpha_1 = \angle BQy_2 + \angle x_1 QB$. But $\alpha_2 - \alpha_1 = 2\pi - \angle x_1 Q y_2 > \pi$ so either $\angle BQy_2 > \frac{\pi}{2}$ or $\angle BQx_1 > \frac{\pi}{2}$. But $BQy_2 > \frac{\pi}{2}$ would mean $|By_2| > |Qy_2| = 1$, which would be contradiction since y_2 is in B. Further $\angle BQx_1 > \frac{\pi}{2}$ would mean

$|Bx_1| > |Qx_1| = 1$, which is also a contradiction. Thus neither $\angle BQy_2 > \frac{\pi}{2}$ nor $\angle BQx_1 > \frac{\pi}{2}$ can hold, and we have a contradiction. This means either x_1 or y_2 is outside circle B and then by Lemma 5.5.1, since x_2 and y_1 are outside of B, either all of the leaf $A \cap B$ or $B \cap C$ is overlapped by Q.

Subcase 3: Q intersects one of the edges p_1p_2 and p_2p_3 once and the other twice.

Suppose WLOG Q intersects p_1p_2 once and p_2p_3 twice, as in Fig. 5.23.

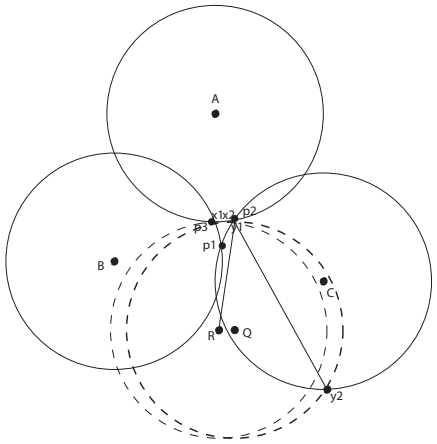

Fig. 5.23. An illustration for the proof of Subcase 3.

For small ϵ, draw a new circle R passing through $p_3 + \epsilon$ and y_1 whose center is within angle $\angle p_3y_1y_2$, as in Fig. 5.23. It will either be the case that the center of Q is inside $\angle Ry_1y_2$ or the center of R is inside $\angle Qy_1y_2$. By the contrapositive of Lemma 5.5.6, since both Q and R pass through the point y_1 and p_3 is not inside Q, we can conclude that R cannot be on or inside $\angle Qy_1y_2$. Thus it must be that Q is inside angle $\angle Ry_1y_2$. Using Lemma 5.5.6 in the forward direction, with knowledge that Q is inside $\angle Ry_1y_2$ and both Q and R pass through y_1, we conclude that both intersections of Q with C are on or outside R and so by Lemma 5.5.1, $R \cap C \subseteq Q \cap C$.

Note that R intersects the arc p_2p_3 once and p_1p_2 once, namely at y_1. If it intersected p_1p_2 twice, then at some point in the process of rotating around y_1,

it must have intersected both p_1p_2 and p_2p_3 each twice. But that would mean R overlaps none of p_1, p_2, and p_3 and hence must be centered inside triangle $\triangle p_1p_2p_3$. But this is impossible, as we established in the proof of Lemma 5.5.7.

Since R intersects each of p_1p_2 and p_2p_3 once, then, for small enough ϵ, R matches the setup as in Subcase 2. From that case we can conclude R overlaps $B \cap C$, since having an intersection at $p_3 + \epsilon$ precludes it from overlapping $A \cap B$ for small enough ϵ. Thus since $B \cap C \subseteq R$, we can conclude that $B \cap C \subseteq R \cap C \subseteq Q \cap C \subseteq Q$ and hence Q covers an entire leaf, the leaf $B \cap C$. ■

To finish Case 1, we need to show that even when the entire concave circular triangle formed by A, B, and C in the above scenario gets overlapped by a disk Q in the 2-clique, Q still overlaps one of the leaves $A \cap B$, $A \cap C$, or $B \cap C$. First, one more theorem and one more lemma are in order.

Theorem 5.5.9 *For $i = 1, 2, 3$, let the circles γ_i have center C_i and equal radius r. Assume that $\triangle C_1C_2C_3$ is acute, and has signed circumradius $c > 0$. Let $\gamma_1 \cap \gamma_2 = \{A_3, B_3\}$, $\gamma_1 \cap \gamma_3 = \{A_2, B_2\}$, and $\gamma_2 \cap \gamma_3 = \{A_1, B_1\}$ be their intersection points. Let A_1, A_2, and A_3 be the three outer intersection points, and let $\delta = sign(r^2 - c^2)$. Let a be the signed circumradius of $\triangle A_1A_2A_3$ and let b be the signed circumradius of $\triangle B_1B_2B_3$. Then*

$$0 \leq a, b \leq r \leq c \leq a + b \quad \textit{if} \quad \delta < 0$$
$$0 \leq b, c \leq r \leq a \leq b + c \quad \textit{if} \quad \delta > 0.$$

Proof This is just an adaptation of Theorems 3 and 4 in [43]. ■

We use this theorem in the proof of Lemma 5.5.11.

Lemma 5.5.10 *Suppose three circles A, B, and C overlap pairwise but $A \cap B \cap C = \emptyset$. Let x, y, and z be the outer vertices of the leaves $A \cap B$, $A \cap C$, and $B \cap C$ respectively. Then $\triangle xyz$ is acute.*

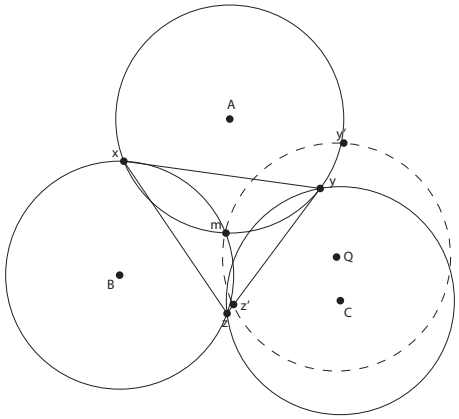

Fig. 5.24. An illustration to the proof of Lemma 5.5.10.

Proof Draw another circle Q as in Fig. 5.24 such that $A \cap B \cap Q = \{m\}$. Suppose that it intersects circle C at point z' and does not overlap the point z. Note that Lemma 5.5.8 can be extended to the case where p_1, p_2, and p_3 coincide, provided the requirement of strict containment is dropped. This means that since Q cannot overlap $B \cap C$, it must be that Q contains $A \cap C$. Thus y is inside Q and $\angle zxy \leq \angle zxy'$. Let $\epsilon = \angle z'xz$ so that $\angle zxy \leq \angle zxy' = z'xy' + \epsilon$. This will hold no matter how small ϵ is made.

By Johnson's Theorem [35], $\triangle z'xy'$ is similar to the triangle made of the centers of the three circles, which is acute by Lemma 5.5.5. Thus $\angle z'xy'$ is acute, and no matter what degree less than 90 this angle is, ϵ can be made small enough such that $\angle zxy$ is also acute.

With the same method, we can conclude $\angle zyx$ and $\angle xzy$ are acute, so the triangle is acute. The angle $\angle zxy$ was an arbitrary angle in the triangle when we showed it acute. ■

Theorem 5.5.11 *Suppose K is a 2-clique in a unit disk graph and $\exists\ A, B, C \in K$ suchthat $A \cap B \cap C = \emptyset$. Then K is 3-dominated.*

Proof Because of Lemma 5.5.8, we need only show that a disk that strictly overlaps the entire concave circular triangle formed from A, B, and C must also overlap one of the leaves $A \cap B$, $A \cap C$, or $B \cap C$ entirely, making our graph 3-dominated by the vertices inside these leaves. Suppose by way of contradiction that a circle Q could overlap the entire concave circular triangle formed by A, B, and C and not overlap a leaf entirely. Then it must overlap with each leaf at two points. Call these points j,k,m,n,r, and s, as in Fig. 5.25.

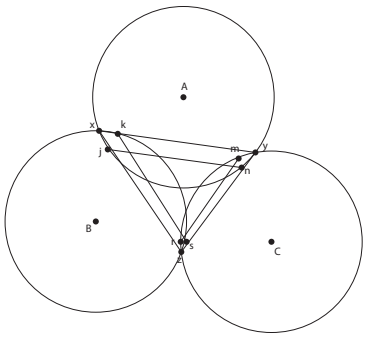

Fig. 5.25. Hypothetical intersections of Q with leaves.

From the picture, if Q is to overlap the entire concave circular triangle, it must be on the opposite side of the line jn from A. Both A and Q go through the points j and n and if they were on the same side of jn then Q could not cover the entire concave circular triangle. Similarly Q must be on the opposite side of the line ks from B and the line rm from C. This is more than enough to ensure that Q must be centered strictly within the triangle $\triangle xyz$.

If any of x, y, or z is inside Q then Q covers an entire leaf and we have contradicted our assumption above. Thus it must be that $|Qx| > 1$, $|Qy| > 1$, and $|Qz| > 1$. Suppose WLOG the minimum of $\{|Qx|, |Qy|, |Qz|\}$ is $|Qx|$. If we draw a circle centered at Q with radius Qx, then y and z must still be outside this circle with enlarged radius, since $|Qx|$ is smallest. Thus this circle must have intersection points on both xy and xz, which we will call u and v respectively.

We wish to show that $|uv| < |yz|$. If that is the case, then we can say that if R is the radius of the circumcircle through x, y, and z, then $R = \frac{|yz|}{\sin \angle yxz} > \frac{|uv|}{\sin \angle yxz} = \frac{|uv|}{\sin \angle uxv} = |Qx| > 1$. But we can show this contradicts Theorem 5.5.9. To apply this theorem, let c be the radius of the circle through the centers of circles A, B, and C. Clearly $c > 1$ since otherwise $A \cap B \cap C$ would not be empty because there would be a point within unit distance of all three. Under this definition of c, Theorem 5.5.9 says that since $\delta = \text{sign}(1 - c^2) < 0$, it must be that $0 \leq R \leq 1$, where R is the circumradius of the circle passing through the outer intersections of circles A, B, and C. But this is a contradiction to $R > 1$ above and thus it must be impossible that a circle Q exists as described.

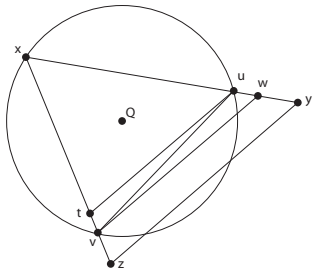

Fig. 5.26. An illustration for the proof of Theorem 5.5.11.

To show $|uv| < |yz|$, note $\triangle uxv$ must be acute. Angle $\angle uxv = \angle yxz$ is acute since $\triangle xyz$ is acute by Lemma 5.5.10. If one of $\angle xuv$ or $\angle xvu$ were right or obtuse, then Q would not lie strictly within $\angle uxv$. But this would mean Q cannot lie within $\triangle xyz$, a contradiction. Thus $\triangle uxv$ must be acute. Construct lines parallel to yz through u and v, as in Fig. 5.26. Call where these lines intersect triangle xyz the points t and w respectively. One of tu and vw will be outside of $\triangle uxv$. Suppose WLOG vw is outside $\triangle uxv$. Then $\angle vuw$ is supplementary to angle vux, which is acute, and hence $\angle vuw$ is obtuse. Since $\angle vuw$ is obtuse, this means $|vw| > |uv|$.

But clearly $|yz| > |vw|$. Hence $|uv| < |yz|$ and the proof of the lemma is complete. ■

5.5.2 Case 2: $A \cap B \cap C \neq \emptyset \ \forall \ A, B, C \in K$

Since in Case 2 we are assuming $A \cap B \cap C \neq \emptyset \ \forall A, B, C \in K$, by Helly's theorem this means there exists a set S of points in space that all members of K overlap. We prove a few lemmata before proving our main result.

Lemma 5.5.12 *Suppose A, B, and C are three disks that intersect pairwise, but $A \cap B \cap C = \emptyset$. Suppose WLOG that $B \cap C$ has area less than or equal to that of $A \cap B$ and $A \cap C$. Let p_1, p_2, and p_3 be the vertices of the concave circular triangle $A \cap B \cap C$. Then any circle D with center in $B \cap C$ will cover the entire circular triangle of A, B, and C.*

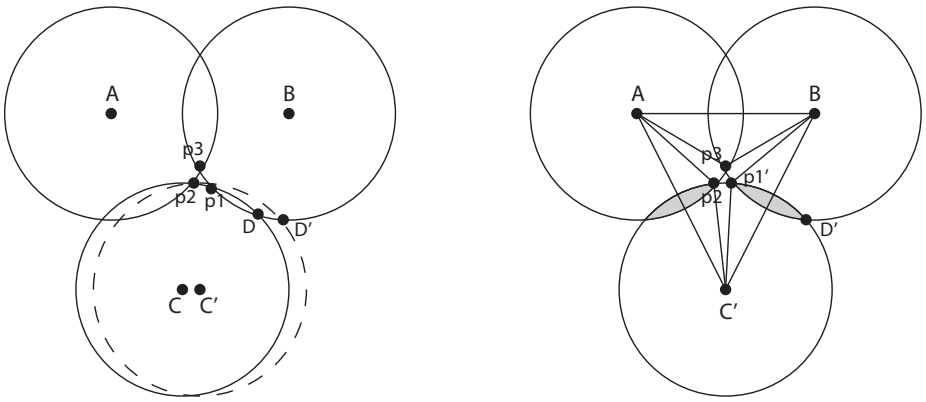

Fig. 5.27. An illustration to the proof of Lemma 5.5.12.

Proof It is sufficient to show the circle D centered at the outer vertex of $B \cap C$ covers the vertices of the circular triangle, specifically p_2 and p_3 since they are most distant from $A \cap C$. Assume WLOG that $A \cap C$ has less area than $A \cap B$, in which

case we rotate circle C around p_2 to create a new circle C' such that $A \cap C' = B \cap C'$, as in Fig. 5.27. If this is false, we rotate B around p_3 instead. By Lemma 5.5.6, it is clear that $A \cap C \subseteq A \cap C'$. Since D is on arc $p_1'D'$, it is clear that $|p_2D'| \geq |p_2D|$ and $|p_3D'| \geq |p_3D|$. Further $|AC'| = |BC'| \geq |AB|$ because the area of leaf $A \cap B$ is larger than that of leaf $A \cap C$, which is larger than the area of leaf $A \cap C'$.

Note that $|Ap_3| = |Bp_3| = |Bp_1'| = |C'p_1'| = |C'p_2| = |Ap_2| = 1$. Since $\angle p_3AB = \angle p_3BA$, we will call this angle α. Similarly $\angle p_3Ap_2 = \angle p_3Bp_1'$ so we refer to these angles as β, and $\angle p_2AC' = \angle p_2C'A = \angle p_1'C'B = \angle p_1'BC'$, which will be referred to as γ. Lastly we will refer to $\angle p_2C'p_1'$ as δ.

The key to the proof is showing that $\beta \leq \delta$. To that end, first note that $\gamma \leq \alpha$. Because $|AC'| = |BC'| \geq |AB|$, which implies that $\angle C'p_1'B$ and $\angle Cp_2A$ are greater than $\angle Ap_3B$, there are fewer degrees available for γ in $\triangle p_1'BC'$ so that $\gamma \leq \alpha$. Suppose for contradiction that $\delta < \beta$. Then $|p_2p_1'| < |p_2p_3|$. But at the same time $\angle p_2p_3p_1' = 2\pi - \angle Ap_3B - \angle Ap_3p_2 - \angle Bp_3p_1' = 2\pi - (\pi - 2\alpha) - \angle Ap_3p_2 - \angle Bp_3p_1'$. But $\angle Ap_3p_2 = \angle Ap_2p_3$ and $\angle Bp_3p_1' = \angle Bp_1'p_3$ since they are isosceles triangles, implying that $2\angle Ap_3p_2 + \angle p_3Ap_2 = \pi$ and $2\angle Bp_3p_1' + \angle p_3Bp_1' = \pi$. Thus $\angle p_2p_3p_1' = 2\pi - (\pi - 2\alpha) - (\frac{\pi}{2} - \frac{\beta}{2}) - (\frac{\pi}{2} - \frac{\beta}{2}) = 2\alpha + \beta$. Similarly $\angle p_3p_1'p_2 = 2\pi - \angle Bp_1'C' - \angle Bp_1'p_3 - \angle C'p_1'p_2 = 2\pi - (\pi - 2\gamma) - (\frac{\pi}{2} - \frac{\beta}{2}) - (\frac{\pi}{2} - \frac{\delta}{2}) = 2\gamma + \frac{\beta}{2} + \frac{\delta}{2}$. If $\delta < \beta$ then $\angle p_3p_1'p_2 = 2\gamma + \frac{\beta}{2} + \frac{\delta}{2} < 2\alpha + \beta = \angle p_2p_3p_1'$ since we know $\gamma \leq \alpha$. But $\angle p_3p_1'p_2 < \angle p_2p_3p_1'$ implies that $|p_2p_3| < |p_2p_1'|$. But $|p_2p_1'| < |p_2p_3|$ was already established above. Because of this contradiction, it must be that $\delta \geq \beta$.

Since $\beta \leq \delta$, we can conclude that $\angle D'Bp_3 = 2\gamma + \beta \leq 2\gamma + \delta = \angle BC'A$, and $\angle BC'A \leq \frac{\pi}{3}$ since $|AC'| = |BC'| \geq |AB|$. Similarly $\angle D'C'p_2 = \angle BC'A \leq \frac{\pi}{3}$. Recall $|p_2D'| \geq |p_2D|$ and $|p_3D'| \geq |p_3D|$ as we established above. Further $|p_3D'| \leq 1$ and $|p_2D'| \leq 1$ since these are the edges across from an angle at most $\frac{\pi}{3}$ in triangles $\triangle p_3D'B$ and $\triangle p_2D'C'$ that have the other sides of length 1. Thus, by the transitive property, we can conclude $|p_3D| \leq 1$ and $|p_2D| \leq 1$ so that the circle centered at D covers both p_2 and p_3 and the proof is done. ■

Lemma 5.5.13 *Suppose a 2-clique K in a unit disk graph is such that $A \cap B \cap C \neq \emptyset$ $\forall A, B, C \in K$ and suppose S is the convex area overlapped by all elements of K, as guaranteed by Helly's theorem. Then if K is not 1-dominated or 2-dominated, there exists a set of three disks A, B, C in K that help to form the border of S such that the leaves $A \cap B$, $A \cap C$, and $B \cap C$ each contain vertices of the graph in their interiors but no vertex of the graph lies inside $A \cap B \cap C$.*

Proof Let S be the convex space overlapped by all members of K as guaranteed by Helly's theorem. If K is not 1-dominated then S must contain no vertices of the graph in its interior since every disk must overlap S. If the border of S is formed by 1 or 2 disks, then K must be 1-dominated, which would be a contradiction. Every pairwise intersection of disks must contain a vertex, since K is a 2-clique, and thus an area bordered by 1 or 2 disks must have a vertex of the graph inside. If the border of S is formed by 3 disks, there must be no vertices in the interior of S since it is not 1-dominated, which means the three leaves surrounding S must contain vertices, and the lemma is true. Thus we can assume the border of S is formed by 4 or more disks.

Let A and B be any two border circles that do not form consecutive pieces of the border of S. Two such disks must exist since the border is defined by 4 or more disks. Note that $(A \cap B) - S$ is separated into two pieces, S_1 and S_2. Since K is a 2-clique, there must be a vertex of the graph in $A \cap B$, and since we are assuming K is not 1-dominated, this vertex must not be in S. Thus either S_1 or S_2 must contain a vertex of the graph. If both S_1 and S_2 contain vertices of the graph, K must be 2-dominated by Corollary 5.5.3, where a point in S_1 is substituted for v_1 and a point in S_2 is substituted for v_2. It is clear that there is no way to "squeeze" a disk between two such points and overlap all of S since A and B form the border of S. Thus we can assume WLOG there is a vertex of the graph in S_1 and none in S_2. Call this vertex p_1.

Since p_1 cannot be in S, there must be a disk bordering S_1 and S that does not contain p_1. Call this disk D_1. If there is no point of the graph inside $D_1 \cap S_1$ then $N(D_1) \cap N(A) \cap N(B) = \emptyset$ and the proof is done. Suppose there is a point of the graph inside $D_1 \cap S_1$. Call this point p_2. Again there must be a disk on the border of S_1 and S that does not contain p_2. Call this disk D_2. If this disk is between D_1 and A on the border of S then replace B with D_1. If D_2 is between D_1 and B then replace A with D_1. Assume WLOG we replaced B with D_1. We now consider $A \cap D_2 \cap D_1$. $A \cap D_1 \cap S_1$ is non-empty since it contains p_2. No graph point can exist in $(A \cap D_1) - S_1$ or else the graph would be 2-dominated because $A \cap D_1 \cap S_1 \subset A \cap B \cap S_1$ and $A \cap B \cap S_1$ has S separating the space from S_2 and hence S must also separate $A \cap D_1 \cap S_1$ from space between $A \cap D_1$ but outside S_1, which we will call S_2'. If S_2' contained a vertex of the graph, the graph would be 2-dominated by the same argument as before. Since D_2 does not contain p_2, it might be that $N(A) \cap N(D_2) \cap N(D_1) = \emptyset$. If so, the proof is done. Otherwise, begin the process again by calling this point inside D_2 to be p_3, finding D_3, and then replacing either A or D_1 with D_2.

This process must terminate with a set of 3 disks with intersection containing no vertices from the graph. Every iteration reduces the number of disks on the boundary of S and S_1 between the two disks forming the lens around p_i and there are only a finite number of disks such as this between the original disks forming the lens, A and B. Eventually, once the edges of the lens are close enough together, there has to be a disk on the boundary of S and S_1 that does not contain any of the vertices of the graph that are also in that lens, since every point in S_1 has some disk separating it from S. Hence this process will terminate in a lens with a vertex and a disk that does not contain any vertex inside that lens. This is a set of 3 disks with empty intersection and the proof is done. ■

Theorem 5.5.14 *If a 2-clique K on a unit disk graph satisfies $A \cap B \cap C \neq \emptyset$ $\forall A, B, C \in K$, then K is 4-dominated.*

Proof We know by the previous lemma there is a set of three border disks A, B, C in K such that no vertex of the graph lies inside $A \cap B \cap C$. We know that each of $A \cap B$, $A \cap C$, and $B \cap C$ must contain vertices of the graph since all points in K have distance 2 or less. Call these vertices v_1, v_2, and v_3 respectively. By Corollary 5.5.3,

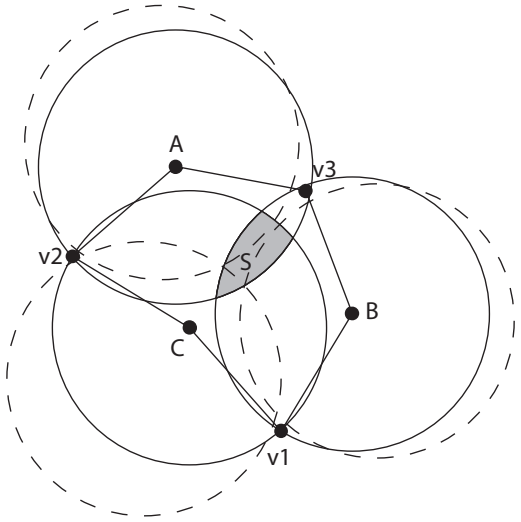

Fig. 5.28. The hexagon border of S in Theorem 5.5.14.

there is no way to "squeeze" a disk between v_1, v_2, and v_3 with a point outside this hexagon without making either A, B, or C be no longer a part of the border of S (see Fig. 5.28), a contradiction. Thus every point of the 2-clique must be either directly connected to v_1, v_2, or v_3 or inside the hexagon. It is possible that a point inside this hexagon avoids being directly connected to v_1, v_2, and v_3, as can be seen in Fig. 5.29. We claim that any point that does not connect to one of v_1, v_2, or v_3 is connected to one other vertex. If there exists a vertex D inside the hexagon not directly connected to v_1, v_2, or v_3, it must be that $v_1 \cap v_2 \cap v_3 = \emptyset$, and the vertex must fall in the gap between disks v_1, v_2 and v_3. But by Lemma 5.5.12, one of A, B, or C, which exist in $v_1 \cap v_2$, $v_1 \cap v_3$, and $v_2 \cap v_3$, must cover the entire middle area and hence the proof is done. ■

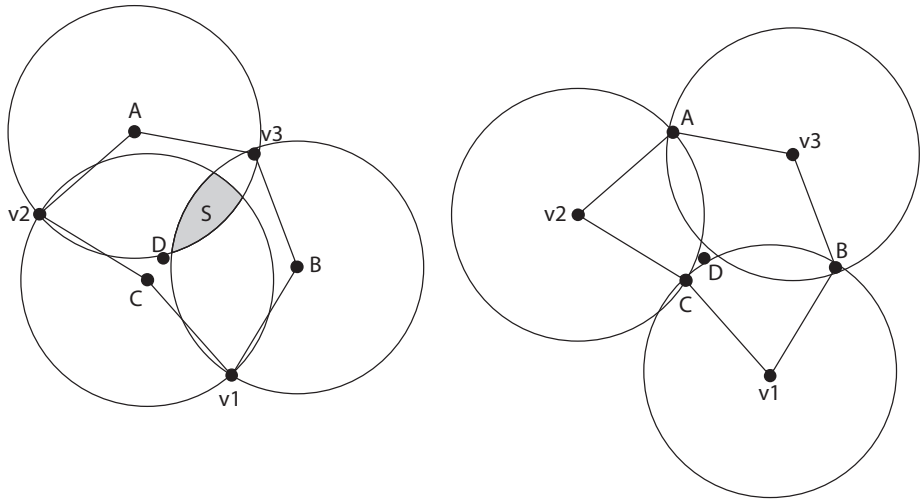

Fig. 5.29. A disk that avoids vertices of the hexagon in Theorem 5.5.14.

5.5.3 Putting It All Together

With all the groundwork done above, we easily prove the following theorem.

Theorem 5.5.15 *All 2-clubs on unit disk graphs are 3-dominated and all 2-cliques on unit disk graphs are 4-dominated.*

Proof We conclude that 2-cliques are 4-dominated by combining Theorem 5.5.11 and Theorem 5.5.14. To see why 2-clubs are in fact 3-dominated, note that the reason we could not conclude 2-cliques are 3-dominated is because in the case where A, B, and C are in a 2-clique K and $A \cap B \cap C \neq \emptyset$, it was possible that $v_1 \cap v_2 \cap v_3 = \emptyset$ and a vertex fell in the gap between v_1, v_2, and v_3. However, by definition of a 2-club, v_1, v_2, and v_3 must be in the 2-club, which is not the case with 2-cliques. Thus if v_1, v_2, and v_3 in the 2-club satisfy $v_1 \cap v_2 \cap v_3 = \emptyset$, we are in fact in Case 1, where we have already proven 3-domination. If $v_1 \cap v_2 \cap v_3 \neq \emptyset$, then our proof in Case 2 shows in fact 3-domination. Thus 2-clubs are 3-dominated no matter which case characterizes the set. ■

6. CONCLUSION

This dissertation explored and compared the structural and computational properties of clique relaxations. The overarching purpose for this research is to assist researchers in making informed decisions as to the best clique relaxation and best algorithm to use for grouping in a given application. We concentrated on first-order clique relaxations, investigating the secondary properties that are a consequence of the primary property defining each structure. We helped to resolve an open question as to the number of steps required to identify the maximum quasi-clique within a network and identified an effective algorithm for finding 2-cliques in unit disk graphs, proving its merit. We now go into more detail about our precise contributions and discuss areas for possible future research.

6.1 Foundations

The first contribution of this dissertation is in laying a foundation for creating or choosing an appropriate clique relaxation based on its structure. We equipped researchers with knowledge of both the primary and secondary properties of first-order clique relaxations by filling in a table of structural properties. Table 3.3 reveals not only which clique relaxations exhibit a pre-determined set of essential properties, but also any extraneous structural requirements that may not be obvious from the definition. If no clique relaxation exhibits the essential set of properties without extraneous requirements that are problematic, we gave the first ever methodology for how a researcher could create a new clique relaxation in hopes of matching the desired set of properties.

All bounds on the properties in Table 3.3 were proven sharp, laying an excellent foundation for making informed decisions about clique relaxations. There are, however, still plenty of opportunities for future research. We examined properties of first-order clique relaxations only. Second order clique relaxations such as the

k-robust s-club and (λ, γ)-quasi-clique should have their structure studied to give researchers even more information. In addition, there may be properties beyond the key defining properties for clique relaxations that would be good to add to our table. We examined the first-order clique relaxations for heredity because it was important computationally. There may be similar properties with significant computational implications that would be a good addition to our table of first-order clique relaxations.

A classical result from complexity theory is that node deletion to find a set defined by a nontrivial, interesting, hereditary property is NP-hard, which was Theorem 3.4.1 in this dissertation. The list of problems proven NP-complete by this general result include node-deletion to find the largest acyclic subgraph, the largest symmetric subgraph, largest planar subgraph, largest outerplanar subgraph, largest bipartite subgraph, and largest chordal graph among others [58]. More significantly, many of these problems were proven NP-complete when restricted to planar and acyclic graphs. Restricting problems to a specific set of subgraphs often makes proving complexity much more difficult and so the ramifications of this result cannot be overstated. We believe the proof might be extended to sets exhibiting weak heredity. If such a proof can be completed, it might prove the k-clique problem, among others, to be NP-complete on various restricted graphs and could be a significant contribution to complexity theory.

We were the first to label k-connected subgraphs as a clique relaxation, though it had been used in that capacity previously [32]. For this reason, the computational complexity of the k-CONNECTED SUBGRAPH problem has not yet been established. It would be useful to explore this, as well as the complexity of the WEAK k-CONNECTED SUBGRAPH problem, which has nicer computational properties than k-connected subgraphs but deviates further from the structure of clique. Both problems would benefit from a polyhedral study and the creation of heuristics.

6.2 Quasi-clique

The second contribution of this dissertation is in resolving the complexity of the maximum quasi-clique problem. By proving the maximum quasi-clique problem NP-hard, we terminated the need to search for fundamentally faster algorithms to solve the problem than those already in existence, assuming $P \neq NP$. While studying the problem, we formulated it as an integer programming problem and helped to validate GRASP as an effective heuristic for solving the problem. We showed there always exists an ordering of vertices such that any γ-quasi-clique can be built one vertex at a time and maintain density γ at every step. This means that any algorithm that concentrates on vertex ordering, such as GRASP, has the potential to find the optimal solution.

Although it likely cannot be solved in polynomial time, there is much work that can still be done to improve exponential time algorithms designed to solve the MAXIMUM QUASI-CLIQUE problem. A polyhedral study of quasi-cliques could prove effective in branch & cut methods for solving the problem. Establishing valid inequalities for quasi-cliques has proven to be difficult, however. This is because quasi-clique was specifically designed to be a clique relaxation with very flexible structure so that it could be used in applications with randomly distributed noise. We established valid inequalities for connected quasi-cliques based on the maximum possible distance between vertices in Table 3.3. If the maximum distance between two vertices in a connected quasi-clique is k, we could use these distance results to build inequalities identical to the *independent set inequalities* of [8] when applied to the graph G^k. An opportunity for future research would be to find other small assumptions, like connectivity, to extend more of the cuts in [8] to quasi-clique and then explore how they perform in a branch & cut method.

In addition to general studies of the QUASI-CLIQUE problem, it would be beneficial to study the problem in the settings of random graphs and power law graphs. The graphs in many applications exhibit one of these two distributions of edges and

the extra structure can make some problems significantly easier. Asymptotics for the size of the largest quasi-clique in a random graph have already been developed [57]. Similar results for power law graphs would be extremely beneficial. As a clique relaxation with very few structural requirements, quasi-clique is ideal for settings where little is known about the structure of the groups desired. However, as a clique relaxation that does not demonstrate heredity or pseudo-heredity, the problem can be very difficult to solve. For this reason, provably efficient algorithms seem unlikely in the setting of general graphs but might be possible in settings where more is known about structure. Such algorithms would constitute a significant contribution to research.

6.3 2-Cliques on Unit Disk Graphs

The third contribution of this dissertation is in establishing a highly effective algorithm for solving the maximum 2-clique problem on unit disk graphs. Unit disk graphs are a topic of much research because they can be used to model many problems in wireless communication. While we prove our algorithm to have a 1/2-approximation ratio, meaning the solution is at least half the size of the largest 2-clique in the graph, we demonstrate the algorithm to be much more effective than that. We generated many instances of random unit disk graphs and in all cases our algorithm returned the exact solution.

There is much more to be gained by future research into k-cliques on unit disk graphs. The computational complexity of the problem still needs to be established. The clique problem, which is highly related to k-clique but slightly simpler on general graphs, is solvable in polynomial time on unit disk graphs. This was proven as part of a fundamental paper on unit disk graphs [24] and helped solidify the value of independent research into unit disk graphs. A similar result for k-cliques would constitute a significant contribution to the theory of unit disk graphs. The diametric

result would provide significant insight into some of the fundamental differences between cliques and k-cliques.

While we prove our algorithm to have a $\frac{1}{2}$-approximation ratio, this ratio may be improved to $\frac{2}{3}$. The example in Section 5.3.3 shows a unit disk graph with a 2-clique that is 3-dominated. We prove that 2-cliques are at most 4-dominated on unit disk graphs. Finding an example of a 2-clique with a minimum dominating set of size 4 would indicate a $\frac{1}{2}$-approximation is the best possible. On the other hand, proving 2-cliques to, in fact, need only 3 vertices for a dominating set would give us a $\frac{2}{3}$-approximation ratio. We have proven this to be so when no area of common overlap exists for the disks in a given 2-clique. Thus it is only necessary to prove 3-domination in the case where the 2-clique has an area of common overlap.

We did show our algorithm has a $\frac{2}{3}$-approximation ratio with asymptotic probability 1 on random unit disk graphs. In showing this result, we made use of a proof that with asymptotic probability 1, points scattered within a punctured unit disk are 2-dominated. It is not difficult to see this implies a 2-clique on unit disk graphs is 2-dominated with asymptotic probability 1 when there is an area of common overlap for all members in the 2-clique. It would be a nice theoretical result to extend this to the case of a 2-clique without an area of common overlap for all its members. This would give theoretical justification to our computational results, where we found our algorithm to always return the exact solution on randomly distributed unit disk graphs.

A consequence of our proof that 2-cliques are 4-dominated on unit disk graphs is that 2-clubs are 3-dominated on unit disk graphs. We mentioned it only in passing in the dissertation because it is not as obvious how to use this information as part of an algorithm to find a largest 2-club. With 2-cliques we had the advantage that they can be solved as a clique on the square of the graph, and that information is what motivated us to try to identify the size of the smallest dominating set for 2-cliques. It would be beneficial to explore how this information about 2-clubs on unit

disk graphs might assist algorithms to solve the problem. Beyond that, it would be beneficial to analyze the entire spectrum of clique relaxations on unit disk graphs, both for the insight they provide to unit disk graphs and for the many applications to which they could provide valuable information.

REFERENCES

[1] Abello, J., Pardalos, P., Resende, M.: On maximum clique problems in very large graphs. In: Abello, J., Vitter, J. (eds.) External Memory Algorithms and Visualization, pp. 119–130. American Mathematical Society, Boston (1999)

[2] Abello, J., Resende, M., Sudarsky, S.: Massive quasi-clique detection. In: Rajsbaum, S. (ed.) LATIN 2002: Theoretical Informatics, pp. 598–612. Springer-Verlag, London (2002)

[3] Alba, R.: A graph-theoretic definition of a sociometric clique. Journal of Mathematical Sociology **3**(1), 113–126 (1973)

[4] Almaas, E., Barabási, A.L.: Power laws in biological networks. In: Koonin, E., Wolf, Y.I., Karev, G.P. (eds.) Power Laws, Scale-Free Networks and Genome Biology, pp. 1–11. Springer Science + Business Media, New York (2006)

[5] Altaf-Ul-Amin, M., Nishikata, K., Koma, T., Miyasato, T., Shinbo, Y., Arifuzzaman, M., Wada, C., et al., M.M.: Prediction of protein functions based on k-cores of protein-protein interaction networks and amino acid sequences. Genome Informatics **14**, 498–499 (2003)

[6] Amin, A.T., Hakimi, S.L.: Upper bounds on the order of a clique of a graph. SIAM Journal on Applied Mathematics **22**, 569–573 (1972)

[7] Bader, G., Hogue, C.: An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics **4**(1), 2 (2003). DOI 10.1186/1471-2105-4-2. URL http://www.biomedcentral.com/1471-2105/4/2

[8] Balasundaram, B.: Graph theoretic generalizations of clique: Optimization and extensions. Ph.D. dissertation, Texas A&M University, College Station (2007)

[9] Balasundaram, B., Butenko, S., Hicks, I.V.: Clique relaxations in social network analysis: The maximum k-plex problem. Operations Research **59**, 133–142 (2011)

[10] Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)

[11] Barabási, A.L., Albert, R., Jeong, H.: Scale-free characteristics of random networks: The topology of the World Wide Web. Physica A **281**, 69–77 (2000)

[12] Barnes, J.: Networks and political process. In: Swart, M. (ed.) Local-Level Politics, pp. 107–130. Aldine, Chicago (1968)

[13] Ben-Dor, A., Shamir, R., Yakhini, Z.: Clustering gene expression patterns. Journal of Computational Biology **6**(3), 281–297 (1999)

[14] Bhattacharyya, M., Bandyopadhyay, S.: Mining the largest quasi-clique in human protein interactome. In: Proceedings of IEEE International Conference on Artificial Intelligence Systems, pp. 194–199. IEEE Computer Society, Los Alamitos, CA (2009)

[15] Boginski, V., Butenko, S., Pardalos, P.: Statistical analysis of financial networks. Computational Statistics & Data Analysis **48**, 431–443 (2005)

[16] Boginski, V., Butenko, S., Pardalos, P.M.: On structural properties of the market graph. In: A. Nagurney (ed.) Innovation in Financial and Economic Networks. Edward Elgar Publishers, London (2003)

[17] Broido, A., Claffy, K.C.: Internet topology: Connectivity of ip graphs. In: Fahmy, S., Park, K. (eds.) Scalability and Traffic Control in IP Networks, pp. 172–187. SPIE Publications, Bellingham, WA (2001)

[18] Brunato, M., Hoos, H., Battiti, R.: On effectively finding maximal quasi-cliques in graphs. In: Maniezzo, V., Battiti, R., Watson, J. (eds.) Proc. 2nd Learning and Intelligent Optimization Workshop, LION 2, *LNCS*, vol. 5313. Springer Verlag (2008)

[19] Budinich, M.: Exact bounds on the order of the maximum clique of a graph. Discrete Applied Mathematics **127**, 535–543 (2003)

[20] Carraghan, R., Pardalos, P.: An exact algorithm for the maximum clique problem. Operations Research Letters **9**, 375–382 (1990)

[21] Ceroi, S.: Clique number and ball containment number of unit ball graphs. Electronic Notes in Discrete Mathematics pp. 22–25 (2001)

[22] Chesler, E.J., Langston, M.A.: Combinatorial genetic regulatory network analysis tools for high throughput transcriptomic data. Tech. Rep. ut-cs-06-575, CS Technical Reports, University of Tennessee, Knoxville (2006)

[23] Chung, F., Lu, L., Vu, V.: The spectra of random graphs with given expected degrees. Internet Mathematics **1**, 257–275 (2004)

[24] Clark, B.N., Colbourn, C.J., Johnson, D.S.: Unit disk graphs. Discrete Mathematics **86**, 165–177 (1990)

[25] Danzer, L., Grunbaum, B., Klee, V.: Helly's theorem and its relatives. In: V. Klee (ed.) Convexity, pp. 101–180. AMS, Providence, RI (1963)

[26] Diestel, R.: Graph Theory. Springer-Verlag, Berlin (1997)

[27] Erdös, P., Rényi, A.: On random graphs. Publicationes Mathematicae **6**, 290–297 (1959)

[28] Erdös, P., Rényi, A.: On the evolution of random graphs. Publ. Math. Inst. Hungar. Acad. Sci. **5**, 17–61 (1960)

[29] Freeman, L.C.: The sociological concept of "group": An empirical test of two models. American Journal of Sociology **98**(1), 152–166 (1992)

[30] Gagneur, J., Krause, R., Bouwmeester, T., Casari, G.: Modular decomposition of protein-protein interaction networks. Genome Biology **5**(8), R57.1–R57.12 (2004)

[31] Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-completeness. W.H. Freeman and Company, New York (1979)

[32] Habibi, M., Eslahchi, C., Wong, L.: Protein complex prediction based on k-connected subgraphs in protein interaction network. BMC Systems Biology **4**(1), 129 (2010). DOI 10.1186/1752-0509-4-129. URL http://www.biomedcentral.com/1752-0509/4/129

[33] Hadwiger, H., Debrunner, H., Klee, V.: Combinatorial Geometry in the Plane. Holt, Rinehart, and Winston, New York (1964)

[34] Hansen, J., Schmutz, E., Sheng, L.: Covering random points in a unit disk. Advances in Applied Probability **40**, 22–30 (2008)

[35] Johnson, R.: A circle theorem. American Mathematical Monthly **23**, 161–162 (1916)

[36] Kahruman-Anderoglu, S.: Optimization in geometric graphs: Complexity and approximation. Ph.D. dissertation, Texas A&M University (2009)

[37] Krishna, P., Vaidya, N., Chatterjee, M., Pradhan, D.K.: A cluster-based approach for routing in dynamic networks. ACM SIGCOMM Computer Communication Review **27**, 49–65 (1997)

[38] Liu, G., Wong, L.: Effective pruning techniques for mining quasi-cliques. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases - Part II, ECML PKDD '08, pp. 33–49. Springer-Verlag, Berlin (2008)

[39] Luce, R.: Connectivity and generalized cliques in sociometric group structure. Psychometrika **15**, 169–190 (1950)

[40] Luce, R., Perry, A.: A method of matrix analysis of group structure. Psychometrika **14**(2), 95–116 (1949)

[41] Luo, F., Li, B., Wan, X.F., Scheuermann, R.: Core and periphery structures in protein interaction networks. BMC Bioinformatics **10**(Suppl 4), S8 (2009). DOI 10.1186/1471-2105-10-S4-S8. URL http://www.biomedcentral.com/1471-2105/10/S4/S8

[42] MacGillivray, G., Seyffarth, K.: Domination numbers of planar graphs. Journal of Graph Theory **22**, 213–229 (1996)

[43] Mackenzie, D.: Triquetras and porisms. The College Mathematics Journal **23**, 118–131 (1992)

[44] Mahdavi, F., Balasundaram, B.: On inclusionwise maximal and maximum cardinality k-clubs in graphs. Working paper

[45] Matsuda, H., Ishihara, T., Hashimoto, A.: Classifying molecular sequences using a linkage graph with their pairwise similarities. Theoretical Computer Science **210**(2), 305 – 325 (1999)

[46] Moody, J., White, D.R.: Structural cohesion and embeddedness: A hierarchical concept of social groups. American Sociological Review **68**, 103–127 (2003)

[47] Motzkin, T.S., Straus, E.G.: Maxima for graphs and a new proof of a theorem of Turán. Canad. J. Math. **17**, 533–540 (1965)

[48] Östergård, P.R.J.: A fast algorithm for the maximum clique problem. Discrete Applied Mathematics **120**, 197–207 (2002)

[49] Pei, J., Jiang, D., Zhang, A.: On mining cross-graph quasi-cliques. In: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, KDD '05, pp. 228–238. ACM, New York, NY, USA (2005)

[50] Philip, J.: The probability distribution of the distance between two random points in a box. TRITA MAT **7** (2007)

[51] Seidman, S.B.: Network structure and minimum degree. Social Networks **5**, 269–287 (1983)

[52] Seidman, S.B., Foster, B.L.: A graph theoretic generalization of the clique concept. Journal of Mathematical Sociology **6**, 139–154 (1978)

[53] Sherali, H.D., Smith, J.C.: A polyhedral study of the generalized vertex packing problem. Mathematical Programming **107**(3), 367 – 390 (2006)

[54] Singh, A., Mai, D., Kumar, A., Steyn, A.: Dissecting virulence pathways of mycobacterium tuberculosis through protein-protein association. In: Proceedings of the National Academy of Sciences of the United States of America, vol. 103, pp. 11,346–11,351 (2006)

[55] Terveen, L., Hill, W., Amento, B.: Constructing, organizing, and visualizing collections of topically related, web resources. ACM Transactions on Computer-Human Interaction **6**, 67–94 (1999)

[56] Veremyev, A., Boginski, V.: Provably optimal design and enhancement strategies for low-diameter networks with guaranteed attack tolerance properties. Working paper

[57] Veremyev, A., Boginski, V., Krokhmal, P., Jeffcoat, D.: On asymptotic behavior of clique relaxations in random graphs. Working paper

[58] Yannakakis, M.: Node-and edge-deletion NP-complete problems. In: STOC '78: Proceedings of the 10th Annual ACM Symposium on Theory of Computing, pp. 253–264. ACM Press, New York (1978)

[59] Yu, H., Paccanaro, A., Trifonov, V., Gerstein, M.: Predicting interactions in protein networks by completing defective cliques. Bioinformatics **22**, 823–829 (2006)

[60] Zeng, Z., Wang, J., Zhou, L., Karypis, G.: Out-of-core coherent closed quasi-clique mining from large dense graph databases. ACM Trans. Database Syst. **32** (2007)

APPENDIX A

COMPUTATIONAL RESULTS FOR THE MAXIMUM QUASI-CLIQUE PROBLEM

Table A.1
Description of the uniform random graphs used in experiments.

Name	n	p	m	$\omega_\gamma(G)$ for $\gamma = ...$					
				1	0.95	0.9	0.85	0.8	0.75
u50-1	50	0.2	215	4	4	5	5	6	7
u50-2	50	0.2	231	4	4	5	5	6	6
u50-3	50	0.2	242	4	4	5	5	6	6
u50-4	50	0.2	221	4	4	5	6	6	7
u50-5	50	0.2	253	4	4	5	6	6	7
u50-6	50	0.3	355	5	5	6	7	8	9
u50-7	50	0.3	379	5	5	5	6	7	8
u50-8	50	0.3	367	5	5	6	7	8	9
u50-9	50	0.3	340	5	5	6	7	7	8
u50-10	50	0.3	354	5	5	6	7	8	9
u100-1	100	0.05	244	3	3	3	3	4	4
u100-2	100	0.05	248	3	3	3	3	3	3
u100-3	100	0.05	217	4	4	4	4	5	5
u100-4	100	0.05	249	3	3	3	3	4	4
u100-5	100	0.05	280	3	3	3	3	4	4
u100-6	100	0.1	536	4	4	4	4	5	5
u100-7	100	0.1	485	4	4	4	4	5	5
u100-8	100	0.1	500	4	4	5	5	6	6
u100-9	100	0.1	469	4	4	4	4	5	5
u100-10	100	0.1	490	4	4	4	4	5	5
u100-11	100	0.15	737	4	4	5	6	6	7
u100-12	100	0.15	711	4	4	5	5	6	6
u100-13	100	0.15	741	4	4	5	5	6	7
u100-14	100	0.15	746	4	4	5	6	6	7
u100-15	100	0.15	760	5	5	6	6	7	8
u100-16	100	0.2	974	5	5	5	7	7	8
u100-17	100	0.2	934	5	5	6	7	7	8
u100-18	100	0.2	977	5	5	6	6	7	8
u100-19	100	0.2	992	5	5	6	7	7	9
u100-20	100	0.2	1010	5	5	6	7	7	8

Table A.2
Description of the power law random graphs used in experiments.

Name	n	β	m	$\omega_\gamma(G)$ for $\gamma = ...$					
				1	0.95	0.9	0.85	0.8	0.75
pl100-1	100	0.1	312	6	7	7	9	11	12
pl100-2	100	0.1	339	6	7	7	9	10	12
pl100-3	100	0.1	313	6	7	8	9	11	12
pl100-4	100	0.1	334	6	7	7	9	11	12
pl100-5	100	0.1	333	6	7	7	9	10	12
pl100-6	100	0.1	340	6	6	7	8	10	11
pl100-7	100	0.1	323	6	7	8	9	11	13
pl100-8	100	0.1	327	6	7	8	9	10	11
pl100-9	100	0.1	347	6	7	7	9	10	11
pl100-10	100	0.1	325	6	6	7	8	9	10
pl100-11	100	0.2	974	15	20	24	27	31	34
pl100-12	100	0.2	986	16	19	23	26	31	34
pl100-13	100	0.2	985	16	20	23	27	31	35
pl100-14	100	0.2	1021	16	21	25	28	32	35
pl100-15	100	0.2	991	16	20	24	28	31	34

Table A.3
Comparison of running times for the experiments with uniform random graphs.

Graph	Running times, in seconds, for $\gamma = ...$							
	1		0.95		0.85		0.75	
	F1	F2	F1	F2	F1	F2	F1	F2
u50-1	0.7	0.1	2.6	1.5	2.9	3.9	5.0	22.2
u50-2	0.2	0.1	2.7	1.6	3.6	4.5	5.9	48.9
u50-3	0.6	0.2	2.5	1.7	4.0	5.0	6.6	120.3
u50-4	0.2	0.2	2.4	1.5	2.9	3.2	5.4	21.6
u50-5	0.7	0.2	2.6	1.7	3.5	6.5	10.6	108.7
u50-6	0.3	0.4	3.3	3.1	6.3	94.7	34.9	13,047.5
u50-7	1.1	0.6	4.0	2.9	8.4	158.2	61.4	41,796.1
u50-8	1.0	0.2	3.1	3.2	6.9	118.7	38.2	24,760.9
u50-9	1.2	0.3	9.9	2.1	6.6	42.7	39.3	4,971.8
u50-10	1.1	0.5	3.1	3.0	6.6	78.4	40.8	12,087.4
u100-1	2.4	0.3	15.5	6.1	146.0	12.8	128.7	36.6
u100-2	2.6	0.3	17.1	5.2	139.6	13.1	135.3	43.1
u100-3	1.1	1.2	17.6	6.1	137.9	9.3	123.1	28.5
u100-4	2.8	0.3	140.1	3.6	152.9	11.7	117.6	36.2
u100-5	2.3	0.5	26.0	6.9	141.8	16.5	106.4	44.5
u100-6	6.8	0.7	152.7	8.8	94.6	99.2	864.2	21,187.1
u100-7	6.3	1.6	139.1	10.7	102.8	72.5	773.8	3,925.0
u100-8	3.0	1.9	143.3	8.9	136.3	70.6	816.5	19,596.6
u100-9	4.8	2.2	138.6	8.4	143.4	45.6	689.5	2,188.1
u100-10	4.9	1.8	131.3	8.4	111.8	52.9	739.3	$> 50,000$
u100-11	5.6	2.1	98.6	15.9	584.4	2,167.1	4,600.1	$> 50,000$
u100-12	6.9	2.4	132.7	14.9	560.4	4,272.2	3,575.9	$> 50,000$
u100-13	6.2	2.5	99.4	17.6	629.0	1,750.1	5,162.8	$> 50,000$
u100-14	5.9	2.5	106.5	16.5	594.8	1,898.3	6,203.0	$> 50,000$
u100-15	5.3	1.7	99.1	16.6	616.0	5,798.9	5,199.7	$> 50,000$
u100-16	6.0	38.8	107.3	23.3	894.2	$> 50,000$	31,544.5	$> 50,000$
u100-17	5.2	2.1	109.7	22.8	801.5	$> 50,000$	34,690.5	$> 50,000$
u100-18	6.3	2.3	114.2	24.1	934.1	$> 50,000$	37,704.0	$> 50,000$
u100-19	5.8	2.2	116.5	27.4	1,033.0	$> 50,000$	33,289.7	$> 50,000$
u100-20	6.1	2.3	102.4	30.3	1,184.9	$> 50,000$	35,457.6	$> 50,000$

Table A.4
Comparison of running times for the experiments with power law random graphs.

Graph	Running times, in seconds, for $\gamma = ...$							
	1		0.95		0.85		0.75	
	F1	F2	F1	F2	F1	F2	F1	F2
pl100-1	2.0	0.1	120.1	7.4	100.5	38.5	84.3	127.9
pl100-2	1.2	0.1	94.0	8.0	104.6	55.0	378.4	165.1
pl100-3	1.5	0.1	94.9	7.0	97.3	27.4	94.9	131.2
pl100-4	2.2	0.1	17.5	8.3	91.7	24.9	240.5	98.5
pl100-5	2.1	0.1	99.5	10.2	114.4	23.6	411.6	201.5
pl100-6	1.8	0.1	112.2	8.5	87.7	34.9	158.3	1,401.1
pl100-7	1.6	0.1	106.7	8.0	93.6	29.2	169.1	101.5
pl100-8	2.4	0.1	123.9	7.1	103.3	28.9	299.1	229.1
pl100-9	2.4	0.1	104.0	7.3	92.4	33.5	347.1	301.8
pl100-10	1.9	0.1	123.4	7.8	118.5	25.9	266.7	276.1
pl100-11	1.2	0.1	96.7	289.8	1,429.9	$> 50,000$	29,272.6	$> 50,000$
pl100-12	1.1	0.1	80.4	125.4	1,649.9	$> 50,000$	26,196.4	$> 50,000$
pl100-13	1.4	0.1	85.4	149.1	1,859.4	$> 50,000$	26,335.3	$> 50,000$
pl100-14	1.1	0.1	77.1	130.2	2,278.4	$> 50,000$	42,761.2	$> 50,000$
pl100-15	1.5	0.1	104.9	178.7	1,581.5	$> 50,000$	39,742.2	$> 50,000$

Table A.5
Comparison of upper bounds for the maximum γ-clique problem.

Graph	γ	bound	LPRF1		LPRF2	
		(4.1) or (4.2)	bound	time	bound	time
u50-1	0.75	22.8895	26.0652	0.187	25.9913	0.032
	0.85	21.4513	25.5849	0.187	25.5126	0.032
	0.95	20.2550	25.1768	0.218	25.1501	0.031
	1	19.7277	25.0000	0.078	25.0000	0.031
u50-6	0.75	30.4112	27.056	0.234	26.9876	0.032
	0.85	28.5178	26.1010	0.202	26.0082	0.032
	0.95	26.9399	25.3336	0.202	25.2911	0.032
	1	26.2437	25.0000	0.093	25.0000	0.031
u100-1	0.75	21.5148	50.4621	0.951	50.4469	0.094
	0.85	20.1598	50.2447	0.999	50.2352	0.093
	0.95	19.0332	50.073	0.983	50.0698	0.093
	1	18.5367	50.0000	0.328	50.0000	0.093
u100-6	0.75	35.9805	51.0966	1.123	51.0581	0.093
	0.85	33.7497	50.5881	1.186	50.5538	0.109
	0.95	31.8892	50.1763	1.373	50.1637	0.109
	1	31.0677	50.0000	0.296	50.0000	0.110
u100-11	0.75	43.0890	51.6616	1.529	51.5565	0.109
	0.85	40.4274	50.9068	1.716	50.8102	0.110
	0.95	38.2059	50.2771	2.169	50.2385	0.093
	1	37.2246	50.0000	0.312	50.0000	0.093
u100-16	0.75	50.1451	52.2528	2.199	52.1858	0.110
	0.85	47.0558	51.2018	1.451	51.1319	0.109
	0.95	44.4759	50.3610	1.544	50.3319	0.125
	1	43.336	50.0000	0.312	50.0000	0.093
pl100-1*	0.75	29.3487	50.6982	2.277	50.6889	0.109
	0.85	27.5992	50.3720	2.463	50.3572	0.110
	0.95	26.1338	50.1110	2.292	50.1048	0.109
	1	25.4850	50.0000	0.296	50.0000	0.094
pl100-11*	0.75	51.4665	53.2654	4.054	53.7225	0.140
	0.85	48.3750	51.7716	3.660	51.8166	0.100
	0.95	45.7855	50.5436	3.809	50.5103	0.100
	1	44.6390	50.0000	0.310	50.0000	0.100

VITA

Jeffrey Lee Pattillo received his Bachelor of Science degree in Mathematics and Computer Science from Wheaton College in 2005. He joined the Mathematics program at Texas A&M in August of 2005, where he received his Masters of Science degree in 2007. He then continued in the program and received his Doctor of Philosophy degree in December 2011. His research interests are in graph theory and discrete optimization, specifically focusing on grouping nodes within networks.

Dr. Pattillo may be reached at the address: Department of Mathematics, Mailstop 3368, Texas A&M University, College Station, TX 77843-3368. His email address is jeff.pattillo@gmail.com.