Discrete Applied Mathematics 161 (2013) 244–257

Contents lists available at SciVerse ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

# On the maximum quasi-clique problem

Jeffrey Pattillo [a], Alexander Veremyev [b], Sergiy Butenko [c,*], Vladimir Boginski [b]

[a] Texas A&M University, Department of Mathematics, College Station, TX 77843, United States
[b] University of Florida, Department of Industrial and Systems Engineering, 303 Weil Hall, P.O. Box 116595 Gainesville, FL 32611-6595, United States
[c] Texas A&M University, Department of Industrial and Systems Engineering, College Station, TX 77843-3131, United States

ARTICLE INFO

*Article history:*
Received 27 April 2011
Received in revised form 14 April 2012
Accepted 29 July 2012
Available online 17 August 2012

*Keywords:*
Clique
Quasi-clique
Optimization

ABSTRACT

Given a simple undirected graph $G = (V, E)$ and a constant $\gamma \in (0, 1)$, a subset of vertices is called a $\gamma$-quasi-clique or, simply, a $\gamma$-clique if it induces a subgraph with the edge density of at least $\gamma$. The maximum $\gamma$-clique problem consists in finding a $\gamma$-clique of largest cardinality in the graph. Despite numerous practical applications, this problem has not been rigorously studied from the mathematical perspective, and no exact solution methods have been proposed in the literature. This paper, for the first time, establishes some fundamental properties of the maximum $\gamma$-clique problem, including the NP-completeness of its decision version for any fixed $\gamma$ satisfying $0 < \gamma < 1$, the quasi-heredity property, and analytical upper bounds on the size of a maximum $\gamma$-clique. Moreover, mathematical programming formulations of the problem are proposed and results of preliminary numerical experiments using a state-of-the-art optimization solver to find exact solutions are presented.



## 1. Introduction

Networks or graphs, consisting of vertices (nodes, dots) and edges (links, arcs) connecting pairs of vertices, provide a convenient modeling tool for studying objects of diverse origins, including the Internet and the World Wide Web, natural ecosystems, cellular processes in biology, and phone call records in telecommunications, among other applications [21]. One of the common issues of interest to many such applications is finding large *clusters*, or "tightly knit" subsets of vertices. The ideal description of a cluster of similar elements is given by the concept of a *clique*, defined as a subset of vertices any two of which are connected by an edge. For example, in social networks, where the vertices correspond to "actors" and an edge indicates a relationship between two actors, a clique represents a group of people, any two of which have a certain kind of relationship (friendship, acquaintance, etc.) with each other [24]. In fact, some of the earliest work addressing the concept of cliques and methods of their detection was motivated by applications in sociometry [17,16,14]. Due to its numerous applications, the maximum clique problem is one of the most important NP-hard problems and it has been extensively studied in the literature. A detailed survey of results on the maximum clique problem and related references can be found in [8].

While the clique model is, undoubtedly, a very reasonable formalization of a cluster, the requirement for every two vertices to be connected is rather restrictive for some applications, especially for those relying on experimental data. This motivated research on a variety of related concepts, some of which can be viewed as "relaxations" of the idea of clique. Not surprisingly, several such concepts first appeared in studying *cohesive subgroups* in social networks and were based

* Corresponding author. Fax: +1 979 847 9005.
*E-mail addresses:* jeff.pattillo@gmail.com (J. Pattillo), averemyev@ufl.edu (A. Veremyev), butenko@tamu.edu (S. Butenko), boginski@reef.ufl.edu (V. Boginski).

0166-218X/$ – see front matter 
doi:10.1016/j.dam.2012.07.019

on relaxing some of the desirable properties that a clique idealizes. The clique relaxation models originating from social network analysis include the notions of $k$-clique [16], $k$-club [19], and $k$-plex [23]; see also [6,5]. Abello et al. [1,2] proposed a density-based relaxation called *quasi-clique*, which requires at least the fraction $\gamma$ of all possible edges between vertices of a $\gamma$-quasi-clique to be present. As a density-based relaxation, $\gamma$-cliques provide a reasonable way for grouping objects that possess no inherent reason to display the structure of other clique relaxations. The previous literature related to the maximum $\gamma$-clique problem concentrated on heuristic detection of large quasi-cliques in various application scenarios. The goal of this paper is to start examining the maximum $\gamma$-clique problem from the mathematical perspective, including establishing the computational complexity of the problem for any fixed $\gamma$, exploring structural properties of $\gamma$-cliques, deriving analytical upper bounds, and developing mixed-integer programming (MIP) formulations.

The remainder of this paper is organized as follows. Section 2 introduces the definitions and notations used throughout the paper. The NP-completeness of the decision version of the maximum $\gamma$-clique problem is proved in Section 3. Section 4 defines the quasi-heredity property and establishes analytical bounds on the $\gamma$-clique number of a graph. Mathematical programming formulations are derived in Section 5 and results of preliminary numerical experiments are reported in Section 6. Finally, Section 7 concludes the paper.

## 2. 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. For $u, v \in V$, $d_G(u, v)$ is the length of a shortest path between $u$ and $v$ in $G$ and $d(G) = \max_{u,v\in V} d_G(u, v)$ is the diameter of $G$. Given $S \subseteq V$, $G[S] = (S, E \cap S \times S)$ denotes the subgraph induced by $S$ on $G$. A graph $G = (V, E)$ is called *complete* if all its vertices are pairwise *adjacent*, i.e. if $\forall i, j \in V$, $i \neq j$, we have $(i, j) \in E$. A *clique* $C$ is a subset of $V$ such that the subgraph $G[C]$ induced by $C$ on $G$ is complete. 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 $\gamma$ satisfying $0 < \gamma < 1$, a subset of vertices $Q$ is called a $\gamma$-quasi-clique or, simply, a $\gamma$-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 $\gamma$. The maximum $\gamma$-clique problem asks for a $\gamma$-clique with the maximum possible number of vertices in $G$. We will denote the $\gamma$-clique number of a graph $G$, which is the cardinality of a largest $\gamma$-clique in $G$, by $\omega_\gamma(G)$. Note that for $\gamma = 1$ the maximum $\gamma$-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 $\gamma$-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 [9,7,18,22,25,15]. 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 paper provides a formal study of the computational complexity of the $\gamma$-clique problem, establishes analytical bounds on the $\gamma$-clique number and proposes mathematical programming formulations of the problem that can be used for finding provably optimal solutions.

## 3. Computational complexity

This section presents the computational complexity analysis for the maximum $\gamma$-clique problem for any fixed density $\gamma$ between 0 and 1. To simplify the analysis, we first replace a real $\gamma$ in the definition of a $\gamma$-clique with a rational $\frac{p}{q}$, where positive integers $p$ and $q$ ($p < q$) 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 $\gamma$-clique case. Following the standard approach [13], 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 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 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 one 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 1.** *For any given fixed real $\gamma \in (0, 1)$, the $\gamma$-Clique problem is NP-complete.*

**Proof.** First observe that instead of assuming that $\frac{p}{q}$ is a fixed value, we could allow $p = p(n)$ and $q = q(n)$ to be functions of $n$ such that $0 < p(n) < q(n)$ and both $p(n)$ and $q(n)$ are $O(n^2)$. 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 $\gamma$-Clique problem are equivalent.

Let $\gamma$ be fixed. Define for any positive integer $s$

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

Note that $\bar{p}(s)$ defines the minimum possible number of edges in the induced subgraph of any $\gamma$-clique of size $s$, so $\bar{p}(s)/\bar{q}(s) \geq \gamma$ and the actual edge density of any $\gamma$-clique of size $s$ is at least $\bar{p}(s)/\bar{q}(s)$. Hence, any $\gamma$-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

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

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 $\gamma$-clique, so $\frac{p(n)}{q(n)}$-Clique problem is equivalent to $\gamma$-Clique problem. □

## 4. Properties of $\gamma$-cliques

In this section we will discuss some properties of $\gamma$-cliques that may be useful in designing solution procedures for the maximum $\gamma$-clique problem. Unlike cliques, the $\gamma$-cliques fail to display a key property used in successful algorithms for the maximum clique problem: *heredity*. A property is called *hereditary* if, when it exists in a graph, it exists in all its induced subgraphs. It is easy to identify $\gamma$-cliques containing subsets of vertices that induce subgraphs with edge density less than $\gamma$.

Because of this, even checking maximality by inclusion is a non-trivial task for quasi-cliques. However, $\gamma$-cliques do display a related property, which we will call *quasi-heredity*.

### 4.1. 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 2.** *The graph property of having edge density of at least $\gamma$ displays quasi-inheritance. In other words, any $\gamma$-clique with $s > 1$ vertices is a strict superset of a $\gamma$-clique with $s - 1$ vertices.*

**Proof.** Consider a $\gamma$-clique $Q$ that induces a subgraph with $s > 2$ vertices and $e$ edges (the statement is trivially true for $s = 2$). Then a smaller $\gamma$-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 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 $\gamma$-clique. □

The quasi-heredity property implies that, provided the vertices are placed in the right order, a maximum $\gamma$-clique can be found by starting with the first vertex in the list and sequentially adding the next vertex if the resulting subset of vertices is still a $\gamma$-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 $\gamma$-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 practical evidence in support of this hypothesis.

### 4.2. Upper bounds

The proposed upper bound on the $\gamma$-clique number is a generalization of the classical Amin–Hakimi bound on the clique number [4].

**Proposition 3.** *The $\gamma$-clique number $\omega_\gamma(G)$ of a graph $G$ with $n$ vertices and $m$ edges satisfies the following inequality:*

$$\omega_\gamma(G) \le \frac{\gamma + \sqrt{\gamma^2 + 8\gamma m}}{2\gamma}. \tag{1}$$

*Moreover, if a graph $G$ is connected then*

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

**Proof.** The first bound is obtained by solving the quadratic inequality

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

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

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

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 [10].

### 4.3. Relation between $\omega_\gamma(G)$ and $\omega(G)$

Next we derive an inequality that will relate the $\gamma$-clique number to the clique number of $G$. We will need the following classical lower bound on the clique number that can be easily obtainedfrom the Motzkin–Straus [20] formulation for the

**Table 1**
The value of upper bound (5) on $\gamma$-clique number $\omega_\gamma(G)$ ($\gamma = 0.95, 0.9, 0.85$) for graphs with known clique number $\omega(G) \in \{2, 3, \dots, 10\}$. The bound does not depend on the size of $G$ and only assumes that $\omega(G)$ is given by the value specified in the first column. Cases where the bound is not applicable (i.e., $\gamma \le 1 - \frac{1}{\omega(G)}$) are marked with "–".

| $\omega(G)$ | $1 - \frac{1}{\omega(G)}$ | Bound (5) on $\omega_\gamma(G)$ for $\gamma = \cdots$ | | |
|---|---|---|---|---|
| | | 0.95 | 0.9 | 0.85 |
| 2 | 0.5 | 2.111 | 2.25 | 2.429 |
| 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 | – | – |

maximum clique problem:

$$\omega(G) \ge \frac{1}{1-\delta}, \tag{3}$$

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

**Proposition 4.** *The $\gamma$-clique number $\omega_\gamma(G)$ and the clique number $\omega(G)$ of graph $G$ satisfy the following inequalities:*

$$\frac{\omega(G)-1}{\omega(G)} \le \frac{\omega_\gamma(G)-1}{\omega_\gamma(G)} \le \frac{1}{\gamma}\frac{\omega(G)-1}{\omega(G)}. \tag{4}$$

**Proof.** The first inequality is trivial due to the fact that $\omega(G) \le \omega_\gamma(G)$. To prove the second inequality, consider a $\gamma$-clique $C$ of largest size $\omega_\gamma(G)$ in $G$. Then, according to (3), the size $\omega(G[C])$ of the largest clique in the induced subgraph $G[C]$ satisfies the inequalities

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

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 $\gamma$-clique, we have

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

Therefore,

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

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

$$\frac{\omega_\gamma(G)-1}{\omega_\gamma(G)} \le \frac{1}{\gamma}\frac{\omega(G)-1}{\omega(G)}. \quad \square$$

**Corollary 2.** *If $\gamma > 1 - \frac{1}{\omega(G)}$ then*

$$\omega_\gamma(G) \le \frac{\omega(G)\gamma}{1-\omega(G)+\omega(G)\gamma}. \tag{5}$$

**Proof.** The result follows directly from the second inequality in (4). $\square$

Observe that bound (5) on the $\gamma$-clique number $\omega_\gamma(G)$ involves only one graph invariant, which is the clique number $\omega(G)$. Unlike more traditional bounds, such as (1) or (2), it does not explicitly depend on the number of vertices and edges in graph $G$, but instead assumes that the clique number of $G$ is known. Such a bound can be especially useful for large sparse networks with a small clique number $\omega(G)$, assuming that $\omega(G)$ can be computed using effective scale-reduction procedures such as "peeling" used in [1]. As supporting evidence, Table 1 provides the value of bound (5) with $\gamma = 0.95, 0.9, 0.85$ for graphs with clique number in the range between 2 and 10. The first column of this table contains the value of the graph's clique number $\omega(G)$ (which is all one needs to know about $G$ in order to compute bound (5) on $\omega_\gamma(G)$); the second column shows the value of threshold $1-\frac{1}{\omega(G)}$ needed to determine whether bound (5) is applicable for given values of $\omega(G)$ and $\gamma$; and the remaining three columns provide the value of bound (5) for $\gamma = 0.95, 0.9$, and 0.85, respectively. Since the bound is available only if $\gamma > 1-\frac{1}{\omega(G)}$, the cases when this condition is not met are marked with a dash in the table. As can be seen from the table, in some cases the bound appears to be extremely useful. For example, if one is given a very large, sparse graph $G$ with $\omega(G) = 4$, it immediately follows from the bounds provided in the table that $\omega_{0.95}(G) = 4$, $4 \le \omega_{0.9}(G) \le 6$, and $4 \le \omega_{0.85}(G) \le 8$.

## 5. MIP formulations of the maximum $\gamma$-clique problem

The lack of structure in $\gamma$-cliques as opposed to cliques and some other clique relaxations, such as $k$-plex [5], 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 $\gamma$-clique problem. This section develops mixed integer programming formulations for the maximum $\gamma$-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 $\gamma$-clique $C^*$ computed. Namely, $i \in C^*$ if and only if $x_i^* = 1$. Then the maximum $\gamma$-clique problem can be formulated as follows:

$$\omega_\gamma(G) = \max \sum_{i=1}^n x_i \tag{6}$$

subject to

$$\sum_{i=1}^n \sum_{j=i+1}^n a_{ij} x_i x_j \ge \gamma \sum_{i=1}^n \sum_{j=i+1}^n x_i x_j, \tag{7}$$

$$x_i \in \{0, 1\}, \quad i = 1, \ldots, n. \tag{8}$$

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

$$w_{ij} \le x_i, \qquad w_{ij} \le x_j, \qquad w_{ij} \ge x_i + x_j - 1. \tag{9}$$

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

$$\omega_\gamma(G) = \max \sum_{i=1}^n x_i, \tag{10}$$

subject to

$$\sum_{i=1}^n \sum_{j=i+1}^n (\gamma - a_{ij}) w_{ij} \le 0, \tag{11}$$

$$w_{ij} \le x_i, \qquad w_{ij} \le x_j, \qquad w_{ij} \ge x_i + x_j - 1, \quad j > i = 1, \ldots, n \tag{12}$$

$$w_{ij} \ge 0, \qquad x_i \in \{0, 1\}, \quad j > i = 1, \ldots, n. \tag{13}$$

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 (6)–(8) had a single constraint that can be written as

$$\sum_{i=1}^n x_i \left(\gamma x_i + \sum_{j=1}^n (a_{ij} - \gamma) x_j\right) \ge 0. \tag{14}$$

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

$$y_i = x_i \left(\gamma x_i + \sum_{j=1}^n (a_{ij} - \gamma) x_j\right). \tag{15}$$

Let us use the following notations:

$$u_i = (1-\gamma)\sum_{j=1}^{n} a_{ij}; \qquad l_i = -\left(n-1-\sum_{j=1}^{n} a_{ij}\right)\gamma, \tag{16}$$

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 (15). Since all variables are binary, the constants $u_i$ and $l_i$ satisfy the following inequalities:

$$l_i \le \gamma x_i + \sum_{j=1}^{n}(a_{ij}-\gamma)x_j \le u_i, \quad i = 1,\ldots,n, \tag{17}$$

$$l_i \le y_i \le u_i, \quad i = 1,\ldots,n. \tag{18}$$

Thus, the quadratic equality (15) with binary variables is equivalent to the following four linear inequalities:

$$y_i \le u_i x_i, \tag{19}$$

$$y_i \ge l_i x_i, \tag{20}$$

$$y_i \ge \gamma x_i + \sum_{j=1}^{n}(a_{ij}-\gamma)x_j - u_i(1-x_i), \tag{21}$$

$$y_i \le \gamma x_i + \sum_{j=1}^{n}(a_{ij}-\gamma)x_j - l_i(1-x_i). \tag{22}$$

Therefore, the problem of finding a maximum $\gamma$-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:

$$\omega_\gamma(G) = \max \sum_{i=1}^{n} x_i \tag{23}$$

subject to

$$\sum_{i=1}^{n} y_i \ge 0, \tag{24}$$

$$y_i \le u_i x_i, \qquad y_i \ge l_i x_i, \quad i = 1,\ldots,n, \tag{25}$$

$$y_i \ge \gamma x_i + \sum_{j=1}^{n}(a_{ij}-\gamma)x_j - u_i(1-x_i), \quad i = 1,\ldots,n, \tag{26}$$

$$y_i \le \gamma x_i + \sum_{j=1}^{n}(a_{ij}-\gamma)x_j - l_i(1-x_i), \quad i = 1,\ldots,n, \tag{27}$$

$$x_i \in \{0,1\}; \qquad y_i \in \mathbb{R}, \quad i = 1,\ldots,n. \tag{28}$$

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

## 6. 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 the 64-bit Windows 7 Professional operating system. The proposed formulations were used in conjunction with the FICO Xpress-IVE Version 1.21.02 solver on a number of instances.

The testbed used included three types of instances: uniform random graphs on 50 and 100 vertices, power-law random graphs on 100 vertices, and a set of standard test instances available in the public domain. Using random graph models provides a convenient and flexible framework for generating a wide range of instances allowing one to explore the capabilities and limitations of the proposed computational approaches. In addition to the commonly used uniform random graphs, we included a set of power-law random graphs in our testbed, which is motivated by a widely reported observation that many real-life complex networks appear to have power-law structure [12,3,21]. A uniformrandom graph $G(n, p)$ has

**Table 2**
Description of the uniform random graphs used in experiments.

| Name | $n$ | $p$ | $m$ | $\omega_\gamma(G)$ for $\gamma = \cdots$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | 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 |

$n$ 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 vertex has 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 [11], which essentially assigns the probabilities $p_{ij}$ for each pair of vertices $(i, j)$ 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. As for standard test instances, we used some of the graphs from the Second and Tenth DIMACS Implementation Challenges (the corresponding links can be found at http://dimacs.rutgers.edu/Challenges/), as well as instances from Trick's graph coloring page (http://mat.gsia.cmu.edu/COLOR04/).

Tables 2–4 present the description of the uniform random graphs, the power-law random graphs, and the standard test instances used, respectively. In Table 2, the first column specifies the name of the graph, while the second and third columns labeled 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 $\gamma$-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 3 compared to Table 2 is in the third column, where the parameter $\beta$ needed to generate a power-law random graph is used instead of $p$. Finally, the column labeled by "$\rho$" in Table 4 contains the graph's edge density, $\rho = \frac{2m}{n(n-1)}$. The first six graphs in the standard instances set listed in Table 4 are from DIMACS Challenges (with the first graph being from the Second DIMACS Implementation Challenge and the other five from the Tenth DIMACS Implementation Challenge), and the remaining instances are from the graph coloring collection. The selection of this subset of well-known graphs, available in the public domain, which excludes larger-scale instances, was driven by limitations of the proposed solution approaches and can be justified by computational intractability of the maximum quasi-clique problem.

The running times for the two proposed formulations applied to the above described uniform random graphs, power-law random graphs, and standard test instances are compared in Tables 5–7, 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 $\gamma$; $\gamma = 1, 0.95, 0.85$, and 0.75, respectively. In each of the four pairs of columns, the first column, marked with "F1", shows the running time for formulation (10)–(13), and the second column, "F2", reports the running time for formulation (23)–(28). All running times are reported in seconds. If the MIP gap did not show much improvement after 50,000 s, the corresponding

**Table 3**
Description of the power law random graphs used in experiments.

| Name | $n$ | $\beta$ | $m$ | $\omega_\gamma(G)$ for $\gamma = \cdots$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | 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 4**
Description of standard test instances used in experiments.

| Name | $n$ | $m$ | $\rho$ | $\omega_\gamma(G)$ for $\gamma = \cdots$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 0.95 | 0.9 | 0.85 | 0.8 | 0.75 |
| Johnson8-2-4 | 28 | 210 | 0.556 | 4 | 4 | 4 | 4 | 5 | 5 |
| Karate | 34 | 78 | 0.139 | 5 | 5 | 6 | 6 | 6 | 7 |
| Dolphins | 62 | 159 | 0.084 | 5 | 5 | 6 | 6 | 7 | 7 |
| Polbooks | 105 | 441 | 0.081 | 6 | 7 | 7 | 9 | 10 | 11 |
| Adjnoun | 112 | 425 | 0.068 | 5 | 5 | 6 | 7 | 8 | 9 |
| Football | 115 | 613 | 0.094 | 9 | 9 | 9 | 10 | 11 | 11 |
| Games120 | 120 | 638 | 0.089 | 9 | 9 | 9 | 10 | 10 | 11 |
| Myciel3 | 11 | 20 | 0.364 | 2 | 2 | 2 | 2 | 2 | 2 |
| Myciel4 | 23 | 71 | 0.281 | 2 | 2 | 2 | 2 | 2 | 2 |
| Myciel5 | 47 | 236 | 0.218 | 2 | 2 | 2 | 2 | 2 | 2 |
| Myciel6 | 95 | 755 | 0.169 | 2 | 2 | 2 | 2 | 2 | 2 |
| Queen5_5 | 25 | 160 | 0.533 | 5 | 5 | 5 | 6 | 7 | 9 |
| Queen6_6 | 36 | 290 | 0.460 | 6 | 6 | 6 | 7 | 7 | 9 |
| Queen7_7 | 49 | 476 | 0.405 | 7 | 7 | 7 | 8 | 8 | 9 |
| 1-FullIns_3 | 30 | 100 | 0.230 | 3 | 3 | 3 | 3 | 5 | 5 |
| 2-FullIns_3 | 52 | 201 | 0.152 | 4 | 4 | 5 | 5 | 6 | 7 |
| 3-FullIns_3 | 80 | 346 | 0.110 | 5 | 5 | 6 | 7 | 7 | 8 |
| 4-FullIns_3 | 114 | 541 | 0.084 | 6 | 7 | 7 | 8 | 9 | 10 |
| 5-FullIns_3 | 154 | 792 | 0.067 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1-FullIns_4 | 93 | 593 | 0.139 | 3 | 3 | 3 | 3 | 6 | 7 |
| 1-Insertions_4 | 67 | 232 | 0.105 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2-Insertions_3 | 37 | 72 | 0.108 | 2 | 2 | 2 | 2 | 2 | 2 |
| 3-Insertions_3 | 56 | 110 | 0.071 | 2 | 2 | 2 | 2 | 2 | 2 |
| 4-Insertions_3 | 79 | 156 | 0.051 | 2 | 2 | 2 | 2 | 2 | 2 |
| Mug88_1 | 88 | 146 | 0.038 | 3 | 3 | 3 | 3 | 4 | 4 |
| Mug88_25 | 88 | 146 | 0.038 | 3 | 3 | 3 | 3 | 4 | 4 |
| Mug100_1 | 100 | 166 | 0.034 | 3 | 3 | 3 | 3 | 4 | 4 |
| Mug100_25 | 100 | 166 | 0.034 | 3 | 3 | 3 | 3 | 4 | 4 |
| Huck | 74 | 301 | 0.111 | 11 | 11 | 11 | 12 | 13 | 14 |
| Jean | 80 | 254 | 0.080 | 10 | 11 | 12 | 13 | 14 | 14 |
| David | 87 | 406 | 0.109 | 11 | 12 | 13 | 14 | 16 | 17 |
| Anna | 138 | 493 | 0.052 | 11 | 12 | 12 | 13 | 15 | 16 |

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 $\gamma$ value. While the second formulation consistently outperforms the first one for higher values of $\gamma$ 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 5 and the last 5 instances in Table 6, for which the second formulation requires

**Table 5**
Comparison of running times for the experiments with uniform random graphs.

| Graph | Running times, in seconds, for $\gamma = \cdots$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 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 | 4971.8 |
| 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 | 3925.0 |
| 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 | 2188.1 |
| 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 | 2167.1 | 4600.1 | >50,000 |
| u100-12 | 6.9 | 2.4 | 132.7 | 14.9 | 560.4 | 4272.2 | 3575.9 | >50,000 |
| u100-13 | 6.2 | 2.5 | 99.4 | 17.6 | 629.0 | 1750.1 | 5162.8 | >50,000 |
| u100-14 | 5.9 | 2.5 | 106.5 | 16.5 | 594.8 | 1898.3 | 6203.0 | >50,000 |
| u100-15 | 5.3 | 1.7 | 99.1 | 16.6 | 616.0 | 5798.9 | 5199.7 | >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 | 1033.0 | >50,000 | 33,289.7 | >50,000 |
| u100-20 | 6.1 | 2.3 | 102.4 | 30.3 | 1184.9 | >50,000 | 35,457.6 | >50,000 |

**Table 6**
Comparison of running times for the experiments with power-law random graphs.

| Graph | Running times, in seconds, for $\gamma = \cdots$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 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 | 1401.1 |
| 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 | 1429.9 | >50,000 | 29,272.6 | >50,000 |
| pl100-12 | 1.1 | 0.1 | 80.4 | 125.4 | 1649.9 | >50,000 | 26,196.4 | >50,000 |
| pl100-13 | 1.4 | 0.1 | 85.4 | 149.1 | 1859.4 | >50,000 | 26,335.3 | >50,000 |
| pl100-14 | 1.1 | 0.1 | 77.1 | 130.2 | 2278.4 | >50,000 | 42,761.2 | >50,000 |
| pl100-15 | 1.5 | 0.1 | 104.9 | 178.7 | 1581.5 | >50,000 | 39,742.2 | >50,000 |

over 50,000 s in multiple cases, while the first one often finds the solution in much shorter time spans and never takes more than 40,000 s. Similar conclusions can be made with respect to the results reported for standard instances. Running times become prohibitively high with an increase in edge density for lower values of $\gamma$, especially for the second formulation. In the most extreme example, the execution of the second formulation for "johnson8-2-4" with $\gamma = 0.75$ has not terminated

**Table 7**
Comparison of running times for the experiments with standard test instances.

| Graph | Running times, in seconds, for $\gamma = \cdots$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | | 0.95 | | 0.85 | | 0.75 | |
| | F1 | F2 | F1 | F2 | F1 | F2 | F1 | F2 |
| Johnson8-2-4 | 0.1 | 0.0 | 0.7 | 1.1 | 1.7 | 752.7 | 20.0 | >50,000 |
| Karate | 0.1 | 0.0 | 0.6 | 0.6 | 0.7 | 0.5 | 0.9 | 0.8 |
| Dolphins | 0.4 | 0.3 | 5.1 | 1.6 | 5.2 | 3.2 | 5.3 | 7.8 |
| Polbooks | 2.1 | 0.3 | 146.2 | 10.1 | 99.8 | 46.4 | 749.5 | 1160.9 |
| Adjnoun | 2.5 | 0.7 | 132.2 | 9.7 | 147.0 | 46.9 | 462.0 | 706.4 |
| Football | 2.2 | 0.4 | 299.9 | 12.2 | 268.1 | 108.2 | 2077.0 | 22,392.4 |
| Games120 | 2.0 | 0.5 | 372.4 | 16.4 | 237.8 | 414.5 | 2968.8 | >50,000 |
| Myciel3 | 0.0 | 0.1 | 0.1 | 0.2 | 0.1 | 0.2 | 0.1 | 0.2 |
| Myciel4 | 0.1 | 0.1 | 0.3 | 1.0 | 0.3 | 0.5 | 0.4 | 1.0 |
| Myciel5 | 0.5 | 0.3 | 2.7 | 1.5 | 3.7 | 4.4 | 7.8 | 84.0 |
| Myciel6 | 3.8 | 1.2 | 65.7 | 16.3 | 409.1 | 13,530.1 | 2184.9 | >50,000 |
| Queen5_5 | 0.1 | 0.0 | 0.4 | 0.7 | 0.8 | 2.4 | 1.3 | 602.9 |
| Queen6_6 | 0.5 | 0.1 | 1.4 | 2.6 | 3.7 | 33.3 | 20.4 | 15,838.2 |
| Queen7_7 | 0.8 | 0.2 | 4.3 | 6.0 | 27.0 | 4840.6 | 359.8 | >50,000 |
| 1-FullIns_3 | 0.1 | 0.1 | 0.6 | 0.6 | 0.7 | 0.7 | 0.8 | 0.9 |
| 2-FullIns_3 | 0.3 | 0.1 | 3.0 | 1.2 | 3.5 | 3.1 | 4.4 | 9.9 |
| 3-FullIns_3 | 0.7 | 0.1 | 10.1 | 4.0 | 15.0 | 22.3 | 27.9 | 163.9 |
| 4-FullIns_3 | 1.3 | 0.1 | 224.2 | 11.5 | 203.3 | 136.7 | 1502.2 | 4819.0 |
| 5-FullIns_3 | 3.0 | 1.1 | 1115.8 | 31.3 | 791.8 | 2013.4 | 6829.2 | >50,000 |
| 1-FullIns_4 | 1.3 | 0.3 | 65.0 | 10.5 | 230.8 | 447.9 | 1100.5 | >50,000 |
| 1-Insertions_4 | 0.6 | 0.2 | 6.6 | 2.5 | 6.8 | 4.9 | 9.9 | 25.6 |
| 2-Insertions_3 | 0.1 | 0.0 | 1.1 | 0.9 | 1.0 | 0.8 | 1.2 | 0.6 |
| 3-Insertions_3 | 0.2 | 0.0 | 2.9 | 1.4 | 3.2 | 2.0 | 3.5 | 2.2 |
| 4-Insertions_3 | 0.6 | 0.1 | 8.2 | 3.0 | 7.8 | 4.2 | 10.1 | 10.0 |
| Mug88_1 | 0.7 | 0.2 | 9.8 | 4.7 | 78.1 | 6.0 | 77.6 | 9.3 |
| Mug88_25 | 0.7 | 0.2 | 7.8 | 5.0 | 72.8 | 5.9 | 73.6 | 10.1 |
| Mug100_1 | 1.0 | 0.2 | 15.1 | 6.8 | 143.9 | 5.2 | 143.3 | 15.4 |
| Mug100_25 | 2.1 | 0.2 | 14.3 | 3.8 | 146.2 | 4.8 | 141.7 | 13.2 |
| Huck | 0.7 | 0.2 | 8.7 | 3.6 | 11.9 | 13.0 | 23.4 | 527.0 |
| Jean | 0.7 | 0.1 | 9.0 | 3.5 | 13.0 | 8.2 | 18.3 | 25.5 |
| David | 0.8 | 0.1 | 49.3 | 6.6 | 45.7 | 83.6 | 281.2 | 957.5 |
| Anna | 2.2 | 0.7 | 280.1 | 22.8 | 418.6 | 1142.0 | 652.9 | 4684.3 |

after 50,000 s. The graph in question has only 28 vertices, but its edge density is over 0.5. The poor performance of the considered approaches on dense graphs is the main reason why "johnson8-2-4" was the only instance from the Second DIMACS Implementation Challenge included in our testbed. The other instances from this collection appeared to be too challenging for the studied approaches to be able to compute $\omega_\gamma(G)$ values for every $\gamma$ in the considered range within the 50,000 s time limit or have over 0.9 edge density and thus are trivial for $\gamma \leq 0.9$. The largest nontrivial standard instance from the testbed that we were able to solve to optimality for all considered $\gamma$ values was "5-FullIns_3" from the graph coloring collection. This graph consists of 154 vertices and 792 edges.

To provide a deeper insight into the performance of the two formulations, Tables 8 and 9 present a comparison of upper bounds on $\omega_\gamma(G)$ for selected random graphs and standard instances, respectively. The first bound is based on analytical expression (1) (if the graph is not connected; such graphs are marked with *) or (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. For random graphs, 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 (2) applies. On the other hand, none of the power-law random graphs in the testbed were connected, therefore, bound (1) was used for the corresponding two instances included in Table 8. The standard instances in Table 9 were selected to illustrate the fact that none of the bounds is dominated by another. Only two of all the considered standard instances are disconnected, and both of them are included in the table. One can observe that in most cases 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 LP relaxation of 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 5–7. 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

**Table 8**
Comparison of upper bounds for selected random graphs. For analytical bound, (1) is used for disconnected graphs (marked with *), and (2) is used for connected graphs.

| Graph | $\gamma$ | Analytical bound | LPRF1 | | LPRF2 | |
|---|---|---|---|---|---|---|
| | | | 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 |

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 terminates at node 17,459. It should be noted that in the reported preliminary experiments we used default solver settings. Perhaps more advanced branch-and-bound strategies, tailored specifically for the maximum quasi-clique problem, may lead to significant speedups.

## 7. Conclusion

This paper is the first attempt to establish rigorous mathematical foundations for the maximum $\gamma$-clique problem that finds numerous practical applications. We show that the decision version of the problem is NP-complete, develop analytical bounds on the $\gamma$-clique number of a graph, and provide mixed-integer programming formulations for the problem of interest. In addition, we report the results of a preliminary computational study employing the proposed formulations in conjunction with a modern commercial MIP solver. The lack of well-defined structure in $\gamma$-cliques makes the problem extremely challenging for exact solution methods. The results of numerical experiments underline the necessity of developing more advanced techniques in order to be able to solve larger-scale instances to optimality. We hope that the analytical bounds and MIP formulations proposed in this paper will motivate future research on exact algorithms for the maximum quasi-clique problem.


## Acknowledgments

We would like to thank everyone involved in refereeing and handling the manuscript for their suggestions that helped to improve the paper. Partial support of Air Force Office of Scientific Research (Awards FA9550-09-1-0154 and FA9550-08-1-0483) and Defense Threat Reduction Agency (Award HDTRA1-09-1-0056) is gratefully acknowledged.

**Table 9**
Comparison of upper bounds for selected standard instances. For analytical bound, (1) is used for disconnected graphs (marked with *), and (2) is used for connected graphs.

| Graph | $\gamma$ | Analytical bound | LPRF1 | | LPRF2 | |
|---|---|---|---|---|---|---|
| | | | Bound | Time | Bound | Time |
| Johnson8-2-4 | 0.75 | 23.9398 | 17.6842 | 0.060 | 17.6842 | 0.020 |
| | 0.85 | 22.4381 | 15.7355 | 0.055 | 15.7355 | 0.015 |
| | 0.95 | 21.1885 | 14.4762 | 0.055 | 14.4762 | 0.015 |
| | 1 | 20.6377 | 14 | 0.035 | 14 | 0.015 |
| Football | 0.75 | 38.3212 | 58.5086 | 7.755 | 58.5079 | 0.125 |
| | 0.85 | 35.9485 | 58.0312 | 8.400 | 58.0291 | 0.125 |
| | 0.95 | 33.9692 | 57.6582 | 8.295 | 57.6568 | 0.125 |
| | 1 | 33.0951 | 57.5 | 0.365 | 57.5 | 0.115 |
| Myciel4 | 0.75 | 13.2946 | 12.5244 | 0.030 | 12.3947 | 0.020 |
| | 0.85 | 12.4353 | 12.0637 | 0.030 | 11.9553 | 0.015 |
| | 0.95 | 11.7243 | 11.6699 | 0.030 | 11.6317 | 0.015 |
| | 1 | 11.4121 | 11.5 | 0.025 | 11.5 | 0.015 |
| Queen7_7 | 0.75 | 35.6272 | 27.7114 | 0.230 | 27.718 | 0.035 |
| | 0.85 | 33.4179 | 26.1819 | 0.170 | 26.1008 | 0.035 |
| | 0.95 | 31.5753 | 25.0204 | 0.185 | 24.9557 | 0.035 |
| | 1 | 30.7617 | 24.5 | 0.075 | 24.5 | 0.03 |
| 5-Fullins_3 | 0.75 | 43.1213 | 78.056 | 32.585 | 77.9992 | 0.215 |
| | 0.85 | 40.4577 | 77.557 | 39.365 | 77.5234 | 0.255 |
| | 0.95 | 38.2346 | 77.1656 | 34.585 | 77.1548 | 0.220 |
| | 1 | 37.2526 | 77 | 0.810 | 77 | 0.190 |
| Mug100_1 | 0.75 | 15.2259 | 50.2922 | 0.550 | 50.2913 | 0.090 |
| | 0.85 | 14.2504 | 50.1544 | 0.590 | 50.1538 | 0.095 |
| | 0.95 | 13.442 | 50.046 | 0.575 | 50.0458 | 0.090 |
| | 1 | 13.0866 | 50 | 0.280 | 50 | 0.085 |
| Huck* | 0.75 | 28.8358 | 38.3131 | 0.605 | 38.3104 | 0.060 |
| | 0.85 | 27.1174 | 37.8462 | 0.740 | 37.6622 | 0.055 |
| | 0.95 | 25.6781 | 37.3961 | 1.475 | 37.1905 | 0.055 |
| | 1 | 25.0408 | 37 | 0.160 | 37 | 0.050 |
| Jean* | 0.75 | 26.5304 | 40.8676 | 0.800 | 40.7582 | 0.070 |
| | 0.85 | 24.9519 | 40.516 | 0.810 | 40.3911 | 0.065 |
| | 0.95 | 23.6298 | 40.1554 | 1.325 | 40.1143 | 0.070 |
| | 1 | 23.0444 | 40 | 0.180 | 40 | 0.055 |

## References


[1] J. Abello, P.M. Pardalos, M.G.C. Resende, On maximum clique problems in very large graphs, in: J. Abello, J. Vitter (Eds.), External Memory Algorithms and Visualization, American Mathematical Society, Boston, 1999, pp. 119–130.
[2] J. Abello, M.G.C. Resende, S. Sudarsky, Massive quasi-clique detection, in: S. Rajsbaum (Ed.), LATIN 2002: Theoretical Informatics, Springer-Verlag, London, 2002, pp. 598–612.
[3] W. Aiello, F. Chung, L. Lu, A random graph model for power law graphs, Experimental Mathematics 10 (2001) 53–66.
[4] A.T. Amin, S.L. Hakimi, Upper bounds on the order of a clique of a graph, SIAM Journal on Applied Mathematics 22 (1972) 569–573.
[5] B. Balasundaram, S. Butenko, I.V. Hicks, Clique relaxations in social network analysis: the maximum *k*-plex problem, Operations Research 59 (2011) 133–142.
[6] B. Balasundaram, S. Butenko, S. Trukhanov, Novel approaches for analyzing biological networks, Journal of Combinatorial Optimization 10 (2005) 23–39.
[7] M. Bhattacharyya, S. Bandyopadhyay, Mining the largest quasi-clique in human protein interactome, in: IEEE International Conference on Artificial Intelligence Systems, IEEE Computer Society, Los Alamitos, CA, USA, 2009, pp. 194–199.
[8] I.M. Bomze, M. Budinich, P.M. Pardalos, M. Pelillo, The maximum clique problem, in: D.-Z. Du, P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, pp. 1–74.
[9] M. Brunato, H.H. Hoos, R. Battiti, On effectively finding maximal quasi-cliques in graphs, in: V. Maniezzo, R. Battiti, J.-P. Watson (Eds.), Learning and Intelligent Optimization, in: LNCS, vol. 5313, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 41–55.
[10] M. Budinich, Exact bounds on the order of the maximum clique of a graph, Discrete Applied Mathematics 127 (2003) 535–543.
[11] F. Chung, L. Lu, V. Vu, The spectra of random graphs with given expected degrees, Internet Mathematics 1 (2004) 257–275.
[12] M. Faloutsos, P. Faloutsos, C. Faloutsos, On power-law relationships of the internet topology, in: Proceedings of the ACM-SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Cambridge, MA, 1999, pp. 251–262.
[13] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York, 1979.
[14] F. Harary, I.C. Ross, A procedure for clique detection using the group matrix, Sociometry 20 (1957) 205–215.
[15] G. Liu, L. Wong, Effective pruning techniques for mining quasi-cliques, in: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases—Part II, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 33–49.
[16] R. Luce, Connectivity and generalized cliques in sociometric group structure, Psychometrika 15 (1950) 169–190.
[17] R. Luce, A. Perry, A method of matrix analysis of group structure, Psychometrika 14 (1949) 95–116.
[18] H. Matsuda, T. Ishihara, A. Hashimoto, Classifying molecular sequences using a linkage graph with their pairwise similarities, Theoretical Computer Science 210 (2) (1999) 305–325.

[19] R. Mokken, Cliques, clubs and clans, Quality and Quantity 13 (1979) 161–173.
[20] T.S. Motzkin, E.G. Straus, Maxima for graphs and a new proof of a theorem of Turán, Canadian Journal of Mathematics 17 (1965) 533–540.
[21] M.E.J. Newman, The structure and function of complex networks, SIAM Review 45 (2003) 167–256.
[22] J. Pei, D. Jiang, A. Zhang, On mining cross-graph quasi-cliques, in: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, KDD'05, ACM, New York, NY, USA, 2005, pp. 228–238.
[23] S.B. Seidman, B.L. Foster, A graph theoretic generalization of the clique concept, Journal of Mathematical Sociology 6 (1978) 139–154.
[24] S. Wasserman, K. Faust, Social Network Analysis, Cambridge University Press, New York, 1994.
[25] Z. Zeng, J. Wang, L. Zhou, G. Karypis, Out-of-core coherent closed quasi-clique mining from large dense graph databases, ACM Transactions on Database System 32 (2007). Article 13. http://dx.doi.org/10.1145/1242524.1242530.