(In)approximability of the Maximum 2-Clique Problem for Chordal Graphs*

Yuichi Asahiro[1], Eiji Miyano[2], and Shunta Nakamura[1]

[1] Kyushu Sangyo University, Fukuoka, Japan
{asahiro@is,k25gjk09@st}.kyusan-u.ac.jp
[2] Kyushu Institute of Technology, Iizuka, Japan
miyano@ai.kyutech.ac.jp

Abstract. We study the complexity of a distance-based relaxed variant of the maximum clique problem, known as MAX 2-CLIQUE: A 2-*clique* in a graph $G = (V, E)$ is a subset $S \subseteq V$ of vertices such that the distance between any pair of vertices in S is at most two in G. Given a graph G with n vertices, the goal of MAX 2-CLIQUE is to find a 2-clique of maximum cardinality. It is known that MAX 2-CLIQUE cannot be approximated in polynomial time within a factor of $n^{1/2-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. On the positive side, a polynomial-time $O(n^{1/2})$-approximation algorithm exists for MAX 2-CLIQUE. In this paper, we establish tight bounds on the approximability of MAX 2-CLIQUE for chordal graphs. Specifically, our main contributions are twofold: (i) There exists a polynomial-time $O(n^{1/3})$-approximation algorithm for MAX 2-CLIQUE on chordal graphs; and (ii) MAX 2-CLIQUE on split graphs cannot be approximated in polynomial time within a factor of $n^{1/3-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$, where the class of split graphs is a well-known subclass of the class of chordal graphs.

Keywords: 2-clique · 2-club · Chordal graph · Approximability · Inapproximability

1 Introduction

1.1 Problems

The Maximum Clique Problem (MAX CLIQUE) is one of the central problems in graph theory, operations research, and combinatorial optimization. As such, it has attracted significant research attention [5]: Let $G = (V, E)$ be an unweighted undirected graph, where V and E denote the set of vertices and the set of edges, respectively. A *clique* in a graph G is a subset $Q \subseteq V$ of pairwise adjacent vertices. Notably, the decision version

* This work was supported by JSPS KAKENHI Grant Numbers JP22K11915 and JP24K02902.

of Max Clique was included in Karp's original list of 21 $\mathcal{NP}$-complete problems [14], marking it as a seminal result in computational complexity theory. Cliques have been used to model a wide range of problems across diverse fields, as they represent extremely cohesive groups, i.e., all members are pairwise adjacent. However, this strict requirement can be overly restrictive in many practical applications, where it may suffice for each member to be reachable within at most d hops for some d. To address this, various relaxations of the clique concept have been proposed [4, 18, 19].

In this paper, we consider a distance-based relaxed variant of Max Clique, known as the maximum distance-d clique problem (Max d-Clique) for a positive integer d [1, 15, 16]: A *distance-d clique* (*d-clique* for short) in a graph $G = (V, E)$ is a subset $S \subseteq V$ of vertices such that for pairs of vertices $u, v \in S$, the distance between u and v in G is at most d. Given a graph G, the goal of Max d-Clique is to find a d-clique of maximum cardinality in G. Another related problem is the maximum diameter-d club problem (Max d-Club), which resembles but is distinct from Max d-Clique: A *diameter-d club* (*d-club* for short) in a graph G is a subset $S' \subseteq V$ such that a subgraph of G induced by S' has diameter at most d. Given a graph G, the goal of Max d-Club is to find a d-club of maximum cardinality in G. The key difference between Max d-Clique and Max d-Club lies in how distances are measured: In a d-clique, every pair of vertices has distance at most d **in the entire graph** G, whereas in a d-club, the diameter of the induced subgraph is at most d, i.e., every pair of vertices has distance at most d **within the d-club itself**. Thus, a d-club is always a d-clique, but the converse does not necessarily hold, since the shortest path between two vertices included in the d-clique may pass through vertices outside the d-clique.

1.2 Previous Results

In the following we assume that the input graph $G = (V, E)$ has $|V| = n$ vertices and $|E| = m$ edges. For Max Clique, that is, for Max 1-Clique and Max 1-Club, the best known polynomial-time approximation algorithm guarantees a solution within a factor of $n(\log \log n)^2/(\log n)^3$ of the optimum [10]. On the negative side, Max Clique cannot be approximated in polynomial time within a factor of $n^{1-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$ [21]. For $d \geq 2$, it is known that both Max d-Clique and Max d-Club cannot be approximated in polynomial time within a factor of $n^{1/2-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$ [2]. On the positive side, polynomial-time $O(n^{1/2})$-approximation algorithms have been developed for both Max d-Clique and Max d-Club with $d \geq 2$ [2].

Although Max d-Clique and Max d-Club (including Max Clique) are difficult to approximate as described above, it is known that Max Clique is polynomial-time solvable when the input graph is restricted to certain classes such as planar graphs, chordal graphs, or graphs with bounded degrees [12]. This line of research, i.e., restricting the input to specific graph classes, is a widely studied approach for addressing computationally hard problems. Here we summarize some known results of Max d-Clique and Max d-Club for various graph classes (refer, e.g., to [9] for graph classes and related parameters). Both Max d-Clique and Max d-Club are $\mathcal{NP}$-hard for graphs with degeneracy at most three when $d \geq 3$, or at most two when $d \geq 5$ [7]. On the positive side, for trees that have representative simple structures, Max d-Club can be solved in polynomial time [11]. Since an optimal solution for Max d-Clique is a d-club for trees, the two problems Max d-Club and Max d-Clique are equivalent on trees. Then, for split graphs, Max 2-Club cannot be approximated in polynomial time within a factor of $O(n^{1/3-\varepsilon})$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$, while a polynomial-time $O(n^{1/3})$-approximation algorithm is known [3]. On the other hand, both Max d-Club and Max d-Clique become trivial for split graphs when $d = 1$ or $d \geq 3$. To the best of our knowledge, whether Max 2-Clique can be solved in polynomial time or not for split graphs is unknown. Since every split graph is a chordal graph, the class of chordal graphs forms a superclass of the class of split graphs. (Note that a tree is also a chordal graph.) Thus, the inapproximability of Max 2-Club on split graphs is transferred to Max 2-Club on chordal graphs; for chordal graphs, Max 2-Club cannot be approximated in polynomial time within a factor of $O(n^{1/3-\varepsilon})$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. However, whether Max 2-Club on chordal graphs can be approximated within a factor of $O(n^{1/3})$ in polynomial time is also unknown. Besides, Max d-Club is solvable in polynomial time when the input is (i) interval, (ii) strongly chordal, (iii) weakly chordal, or (iv) chordal for odd $d \geq 3$ [13]. It has been also unknown so far whether Max 2-Clique is solvable in polynomial time or not for chordal graphs.

1.3 Contributions

This paper addresses the gaps identified in the previous section for chordal (including split) graphs and $d = 2$. Below, we summarize the obtained results:

– MAX 2-CLIQUE and also MAX 2-CLUB can be approximated in polynomial time within a factor of $O(n^{1/3})$ for chordal graphs (and hence also for split graphs).
– MAX 2-CLIQUE on split graphs cannot be approximated in polynomial time within a factor of $n^{1/3-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. (Hence the same inapproximability bound also holds for chordal graphs.)

1.4 Organization

Section 2 introduces the notation and problem definitions. In Sect. 3, we present the approximability result for MAX 2-CLIQUE on chordal graphs, followed by the corresponding result for MAX 2-CLUB. Section 4 is devoted to establishing the inapproximability of MAX 2-CLIQUE on chordal graphs. Finally, Sect. 5 concludes the paper with several open problems.

2 Preliminaries

2.1 Notation

Let $G = (V, E)$ be a connected undirected graph, where V and E are the set of vertices and the set of edges, respectively. $V(G)$ and $E(G)$ also denote the vertex set and the edge set of G, respectively. We denote an edge with endpoints u and v by $\{u, v\}$. For a vertex v, the set of vertices adjacent to v in G, i.e., the open neighborhood of v is denoted by $N_G(v)$, and $N_G[v]$ denotes the closed neighborhood of v, i.e., $N_G[v] = N_G(v) \cup \{v\}$. We generalize the notions of $N_G(v)$ and $N_G[v]$ to distance two. Let $N_G^2(v)$ denote the set of all vertices that are of distance **exactly** two from a vertex v, that is, $N_G^2(v) = \{w \mid w \in N_G(u) \setminus N_G(v) \text{ such that } u \in N_G(v)\}$. The set of all vertices that are of distance **at most** two from a vertex v is denoted by $N_G^2[v]$, namely, $N_G^2[v] = N_G[v] \cup N_G^2(v)$. The degree $d_G(v)$ of a vertex v in G is defined as $|N_G(v)|$. Then the maximum degree among all vertices in a graph G is denoted by $\Delta(G)$, i.e., $\Delta(G) = \max_{v \in V(G)}\{d_G(v)\}$.

A graph H is a subgraph of a graph $G = (V, E)$ if $V(H) \subseteq V$ and $E(H) \subseteq E$. For a subset of vertices $U \subseteq V$, $G[U]$ denotes the subgraph of G induced by U, where the edge set of $G[U]$ includes all the edges of G connecting two vertices in U.

A path P of length ℓ from a vertex v_0 to a vertex v_ℓ in G is represented as a sequence of vertices such as $P = \langle v_0, v_1, \ldots, v_\ell \rangle$, where $v_i \in V(G)$ for $0 \leq i \leq \ell$ and $\{v_i, v_{i+1}\} \in E(G)$ for $0 \leq i \leq \ell - 1$. For a pair of vertices u and v in G, the length of a shortest path from u to v, i.e., the distance between u and v is denoted by $dist_G(u,v)$, and the diameter of G is defined as $diam(G) = \max_{u,v \in V}\{dist_G(u,v)\}$.

Let $G = (V, E)$ be a graph and $H = (V', E')$ be its subgraph, where $V' \subseteq V$ and $E' \subseteq E$. Then, the subgraph H or the subset V' of vertices is called a *clique*, if $E' = \{\{u,v\} \mid u, v \in V'\}$. When $E' = \emptyset$, H or V' is called an *independent set*. A graph G is *chordal* if every cycle $\langle v_0, v_1, v_2, \ldots, v_{\ell-1}, v_0\rangle$ in G of length $\ell \geq 4$ has at least one chord, where a chord is an edge connecting two vertices v_i and v_j such that $j \neq (i+1) \bmod \ell$. If a cycle does not have any chord, it is called an *induced cycle*. A graph G is a *split graph* if there is a partition of $V(G)$ into a clique C and an independent set I such that $C \cap I = \emptyset$ and $C \cup I = V(G)$. Using these C and I, we sometimes write a split graph G as $G = (C, I, E)$, where E is the edge set of G. It is known that every split graph is a chordal graph; the class of split graphs is a subclass of the class of chordal graphs [9].

For the maximization problems, an algorithm `ALG` is called a σ-approximation algorithm and `ALG`'s approximation ratio is σ if $OPT(G)/ALG(G) \leq \sigma$ holds for every input G, where $ALG(G)$ and $OPT(G)$ are the numbers of vertices of obtained subgraphs by `ALG` and an optimal algorithm, respectively. A gap-preserving reduction Γ from a maximization problem Π_1 to another maximization problem Π_2 with four parameters (functions) f_1, f_2, α, and β is defined as follows: Given an instance x of Π_i, Γ computes an instance y of Π_2 in polynomial time such that (i) if $OPT_{\Pi_1}(x) \geq f_1(x)$, then $OPT_{\Pi_2}(y) \geq f_2(y)$; and (ii) if $OPT_{\Pi_1}(x) < f_1(|x|)/\alpha(|x|)$, then $OPT_{\Pi_2}(y) < f_2(y)/\beta(|y|)$, where $OPT_{\Pi_1}(x)$ and $OPT_{\Pi_2}(y)$ are optimal values for the instance x of Π_1 and the instance y of Π_2, respectively. If Π_1 can not be approximated in polynomial time within a factor of $\alpha(|x|)$ unless $\mathcal{P} = \mathcal{NP}$ and a gap-preserving reduction from Π_1 to Π_2 can be constructed, then Π_2 can not be approximated within a factor of $\beta(|y|)$ in polynomial time unless $\mathcal{P} = \mathcal{NP}$ [20].

2.2 Problems

We first provide the formal definition of a d-clique.

Definition 1. *A subgraph H of a graph G is a* d-clique *if $dist_G(u,v) \leq d$ holds for every pair of $u, v \in V(H)$.*

The diameter of a d-clique may be greater than d, and also a d-clique may be disconnected. A d-clique with diameter at most d is called d-club.

Definition 2. *A subgraph H of a graph G is a d-*club *if $dist_H(u, v) \leq d$ holds for every pair of $u, v \in V(H)$, i.e., $diam(H) \leq d$.*

A d-club (or a d-clique) is originally defined as a maximal subgraph in [16], i.e., no super set of vertices of a d-club (or a d-clique) forms a d-club (or a d-clique). However, according to the modified definitions in [6, 8], we do not restrict our attention to maximal ones in this paper. Namely, a non-maximal d-club (or a non-maximal d-clique) can also be treated as a d-club (or a d-clique). As for the maximality, it is known to be $\mathcal{NP}$-complete to answer whether a given d-club is maximal or not [17].

The maximum d-clique problem (MAX d-CLIQUE), and the maximum d-club problem (MAX d-CLUB) are formulated as follows:

Problem: MAXIMUM d-CLIQUE (MAX d-CLIQUE)
Input: A connected undirected graph $G = (V, E)$.
Output: A maximum d-clique in G.

Problem: MAXIMUM d-CLUB (MAX d-CLUB)
Input: A connected undirected graph $G = (V, E)$.
Output: A maximum d-club in G.

Throughout the paper, we use the following notation: $n = |V|$ and $m = |E|$ for the input graph G. In the case $d = 1$, the above two problems are equivalent; the problems MAX 1-CLUB and MAX 1-CLIQUE are identical to MAX CLIQUE which is to find a maximum clique in a given graph, and hence they are generalizations of MAX CLIQUE in terms of distance and diameter of output subgraphs.

3 Approximability

In this section, we show approximability results. First we show that MAX 2-CLIQUE on chordal graphs can be approximated in polynomial time within a factor of $O(n^{1/3})$ in Sect. 3.1. Then, in Sect. 3.2, we see that MAX 2-CLUB on chordal graphs is also approximated in polynomial time within the same factor $O(n^{1/3})$.

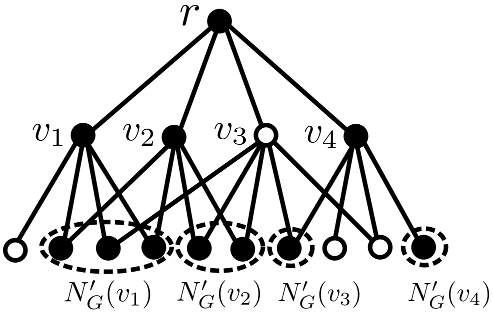

Fig. 1. A solution S is represented by filled vertices. Within S, there exists a vertex whose distance from a designated vertex r is two. This figure highlights the vertex r along with $N^2_G[r]$, where $N_G(r) = \{v_1, v_2, v_3, v_4\}$ and $N^*_G(r) = \{v_1, v_2, v_4\}$. Dashed circles indicate the sets $N'_G(v_1)$ through $N'_G(v_4)$. Note that only the edges between r and $N_G(r)$, and between $N_G(r)$ and $N^2_G(r)$, are shown. As a result, the subgraph induced by the filled vertices may not appear to form a 2-clique in the figure.

3.1 Max 2-Clique on chordal graphs

This section discusses the approximability of Max 2-Clique for chordal graphs. Since a maximum clique in a chordal graph can be found in polynomial time [12], we assume that an optimal solution for Max 2-Clique is not a clique.

Let S be an optimal set of vertices for Max 2-Clique on chordal graph G such that $G[S]$ does not form a clique. The diameter of $G[S]$ is unclear, since it can be a disconnected graph for Max 2-Clique. However, for any vertex $r \in S$, $S \subseteq N^2_G[r]$ holds. For simplicity, $G[S]$ is denoted by H. Let $r \in S$ denote a vertex satisfying that there exists a vertex $v \in S$ with $dist_G(r, v) = 2$ (note that it is not "$dist_H(r, v) = 2$").

Let $d = deg_G(r)$, $N_G(r) = \{v_1, v_2, \ldots, v_d\}$, and $N^*_G(r) = N_G(r) \cap S$. Note that $N^*_G(r) = \emptyset$ can happen, since H may be a disconnected 2-clique. Let $N'_G(v_i)$ denote a set of vertices in $N^2_G(r) \cap S$ which belong to $N_G(v_i)$ but not belong to $N_G(v_j) \cap S$ for $1 \le j \le i-1$. That is, formally, $N'_G(v_i) = \{w \in N_G(v_i) \cap S \cap N^2_G(r) \mid w \notin \bigcup_{j=1}^{j=i-1} N'_G(v_j)\}$ for $1 \le i \le d$. See Fig. 1 for an illustration. We can assume $|N'_G(v_1)| \ge |N'_G(v_2)| \ge \cdots \ge |N'_G(v_d)|$ without loss of generality: To obtain a contradiction, suppose that $|N'_G(v_i)| < |N'_G(v_{i+1})|$, and let G' be the graph obtained by swapping the indices i and $i+1$. Then, in G', $N'_{G'}(v_i) = N'_G(v_{i+1}) \cup T$ and $N'_{G'}(v_{i+1}) = N'_G(v_i) \setminus T$, where T is a set of vertices in $N'_G(v_i) \cap N'_G(v_{i+1})$. Thus, $|N'_{G'}(v_i)| > |N'_{G'}(v_{i+1})|$, since $|N'_G(v_i)| < |N'_G(v_{i+1})|$.

The algorithm we use in this section is `FindStar` which simply finds a vertex u of the maximum degree $\Delta(G)$, and outputs its closed neighborhood $N_G[u]$. The subgraph induced by $N_G[u]$ is clearly a 2-clique (more precisely, a 2-club), and obviously `FindStar` runs in polynomial time. It is known that `FindStar` is an $O(n^{1/2})$-approximation algorithm for Max 2-Clique (and also for Max 2-Club) on general graphs [2]. We first consider the case that $N'_G(v_2) = \emptyset$.

Lemma 1. *If $N'_G(v_2) = \emptyset$, then the size $|S|$ of the maximum 2-clique is at most $2\Delta(G)$.*

Proof. First, the assumption $N'_G(v_2) = \emptyset$ implies that $N'_G(v_i) = \emptyset$ also for $i \geq 3$ by the assumption $|N'_G(v_i)| \geq |N'_G(v_{i+1})|$.

Since $|N'_G(v_1)| \leq \Delta(G) - 1$,

$$\begin{aligned}|S| &\leq |\{r\} \cup N_G(r) \cup N'_G(v_1)| \\ &\leq |\{r\} \cup N_G(r) \cup (N_G(v_1) \setminus \{r\})| \\ &\leq 1 + \Delta(G) + (\Delta(G) - 1) \\ &= 2\Delta(G),\end{aligned}$$

where the first inequality follows from the facts that $N_G(r) \cap S \subseteq N_G(r)$ and $N'_G(v_i) = \emptyset$ for $i \geq 2$. $\square$

Next, we consider the case $N'_G(v_2) \neq \emptyset$ in the following, i.e., there are at least two v_i's such that $|N'_G(v_i)| > 0$. The following two lemmas demonstrate existence of certain types of edges.

Lemma 2. *Suppose that $u_i \in N^2_G(r) \cap N_G(v_i)$ and $u_j \in N^2_G(r) \cap N_G(v_j)$. If $\{u_i, u_j\} \in E(G)$, then $E(G)$ includes either or both $\{v_i, u_j\}$ and $\{v_j, u_i\}$.*

Proof. If $v_i = v_j$, then $\{v_j, u_j\} = \{v_i, u_j\}$ and $\{v_i, u_i\} = \{v_j, u_i\}$. By definition $u_j \in N^2_G(r) \cap N_G(v_j)$, $\{v_j, u_j\} \in E(G)$. Analogously, $\{v_i, u_i\}$ exists in $E(G)$. Thus, the lemma holds.

Consider the case $v_i \neq v_j$. For obtaining a contradiction, assume that neither $\{v_i, u_j\}$ nor $\{v_j, u_i\}$ exists in a chordal graph G. There is a cycle $C = \langle r, v_i, u_i, u_j, v_j, r\rangle$ of length five in G, and neither $\{r, u_i\}$ nor $\{r, u_j\}$ exists in G by definitions of u_i and u_j. See Fig. 2 for an illustration. Thus, we can have $\{v_i, v_j\}$ only as a chord of the cycle C. Even though, still there is an induced cycle $\langle v_i, u_i, u_j, v_j, v_i\rangle$ of length four. Namely, by the assumption, there is an induced cycle of length four or five, implying that G is not chordal. This is a contradiction, and thus $E(G)$ includes either or both $\{v_i, u_j\}$ and $\{v_j, u_i\}$. $\square$

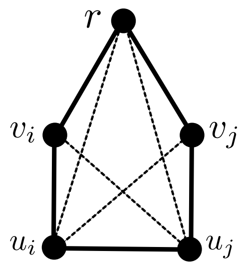

Fig. 2. Cycle $\langle r, v_i, u_i, u_j, v_j, r\rangle$ of length five. Dashed lines indicate where edges can not exist by assumption of the proof of Lemma 2.

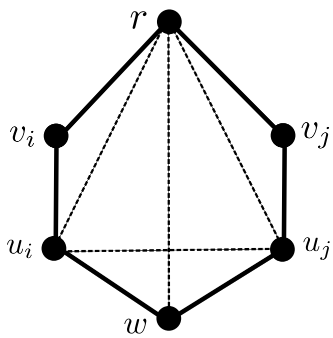

Fig. 3. Cycle $\langle r, v_i, u_i, w, u_j, v_j, r\rangle$ of length six. Dashed lines indicate where edges can not exist by assumption of the proof of Lemma 3.

Lemma 3. *Suppose that $u_i \in N_G^2(r) \cap N_G(v_i)$ and $u_j \in N_G^2(r) \cap N_G(v_j)$. If there is a path $\langle u_i, w, u_j\rangle$, for some $w \notin N_G[r]$ and $\{u_i, u_j\} \notin E(G)$, then $E(G)$ includes both $\{w, v_i\}$ and $\{w, v_j\}$.*

Proof. If $v_i = v_j$, then there is a cycle $C = \langle v_i, u_i, w, u_j, v_i\rangle$. Since $\{u_i, u_j\} \notin E(G)$ and G is chordal, $\{w, v_i\}(= \{w, v_j\})$ must exist as a chord of C.

Consider the case $v_i \neq v_j$. First, there exists a cycle $C = \langle r, v_i, u_i, w, u_j, v_j, r\rangle$ in G, where none of $\{r, u_i\}$, $\{r, u_j\}$, $\{r, w\}$, and $\{u_i, u_j\}$ exists. See Fig. 3 for an illustration. Look at the subpath $\langle v_i, u_i, w, u_j\rangle$ of C, whose length is three. To obtain a contradiction, assume that $\{w, v_i\} \notin E(G)$. Even if we add all remaining (possible) edges $\{v_i, v_j\}$, $\{v_i, u_j\}$, $\{u_i, v_j\}$, and $\{w, v_j\}$, there exists an induced cycle $\langle v_i, u_i, w, u_j, v_i\rangle$ of length four. This contradicts the assumption that G is chordal. Hence $\{w, v_i\} \in E(G)$. By a similar argument, $\{w, v_j\} \in E(G)$, too. $\square$

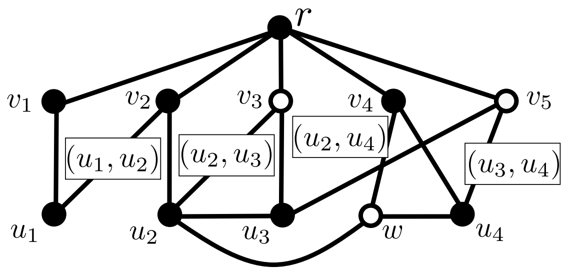

Fig. 4. Assignment of each pair of vertices in $N_G^2(r) \cap S$ to an edge between $N_G(r)$ and $N_G^2(r)$ in the proof of Lemma 4. The pairs (u_1, u_2), (u_2, u_3), (u_2, u_3), and (u_3, u_4) are assigned to the edges $\{v_2, u_1\}$, $\{v_3, u_2\}$, $\{v_4, w\}$, and $\{v_5, u_4\}$, respectively. Unfilled vertices are assumed not to be included in the solution. However, this assumption is immaterial, since the proof relies only on the edges between $N_G(r)$ and $N_G^2(r)$.

Next, we bound the size of $\bigcup_{j=1}^{d} N'_G(v_j)$, which plays an important role to show the approximability of `FindStar`. Let $q = |\bigcup_{j=1}^{d} N'_G(v_j)|$.

Lemma 4. *If $N'_G(v_2) \neq \emptyset$, then $q \leq 4 \cdot \Delta(G)^{3/2} + 1$.*

Proof. For simplicity, let $V = V(G)$, $E = E(G)$, and $\Delta = \Delta(G)$. We estimate q by analyzing the number of edges between $N_G(r)$ and $N_G^2(r)$.

The proof strategy. The proof proceeds as follows.

(I) For a pair of vertices $u_i \in N'_G(v_i)$ and $u_j \in N'_G(v_j)$, we *assign* the pair (u_i, u_j) to an existing edge between $N_G(r)$ and $N_G^2(r)$, utilizing Lemmas 2 and 3. See Fig. 4 for an illustration.

(II) A single edge may be assigned to multiple such pairs by (I). We observe that the number of pairs assigned to any edge is at most 8Δ.

(III) Since the number of edges emanating from $N_G(r)$ is at most $\Delta \cdot |N_G(r)|$, the total number $q(q-1)/2$ of pairs in $\bigcup_{j=1}^{d} N'_G(v_j)$ is bounded above by $8\Delta^2 \cdot |N_G(r)|$. This yields the desired bound on q.

(I) We begin by selecting an edge to which the pair (u_i, v_j) will be assigned, where $i < j$. To ensure that $dist_G(u_i, u_j) \leq 2$, there must exist additional edges beyond the four fixed ones: $\{r, v_i\}$, $\{v_i, u_i\}$, $\{r, v_j\}$, and $\{v_j, u_j\}$. Note that neither $\{r, u_i\}$ nor $\{r, u_j\}$ can exist; otherwise $u_i \in N_G(r)$ or $u_j \in N_G(r)$, contradicting the assumptions that $u_i \in N'_G(v_i)$ and $u_j \in N'_G(v_j)$. Therefore, to satisfy the distance condition, one of the following must hold: (A) $\{v_i, u_j\} \in E$, (B) $\{v_j, u_i\} \in E$, (C) $\{u_i, u_j\} \in E$, or (D) there exists a path $\langle u_i, w, u_j \rangle$ for some vertex $w \in V \setminus \{r, v_i, v_j\}$.

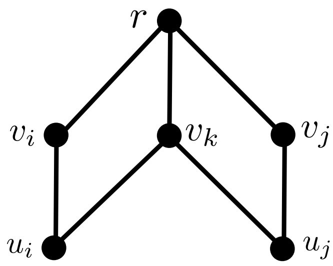

Fig. 5. (D-1) A path $\langle u_i, w, u_j \rangle$, where $w = v_k$ for some $k \notin \{i, j\}$.

(Note that, for example, if $w = v_j$ then $\{u_i, v_j\} \in E$). In the following, we examine each of these cases (A)-(D) to determine an appropriate edge to which the pair (u_i, u_j) can be assigned.

(A) If $\{v_i, u_j\} \in E$, it contradicts the assumption $u_j \in N'_G(v_j)$ for $i < j$ by the definition of $N'_G(v_j)$. Thus, we assume $\{v_i, u_j\} \notin E$ in the following.

(B) If ($\{v_i, u_j\} \notin E$ and) $\{v_j, u_i\} \in E$, we assign the pair (u_i, u_j) to this edge $\{v_j, u_i\}$. By this type, pairs (u_i, u) for $u \in N'_G(v_j)$ may be assigned to $\{v_j, u_i\}$. (In Fig. 4, the pair (u_1, u_2) is assigned to the edge $\{v_2, u_1\}$.)

(C) Suppose that ($\{v_i, u_j\} \notin E$ and) $\{u_i, u_j\} \in E$. From Lemma 2, $\{\{v_i, u_j\}, \{v_j, u_i\}\} \cap E \neq \emptyset$. However, from (A), $\{v_i, u_j\} \notin E$, and thus $\{v_j, u_i\} \in E$. Therefore, this case is included in (B), and hence we assign the pair (u_i, u_j) to this edge $\{v_j, u_i\}$. (In Fig. 4, the pair (u_2, u_3) is assigned to the edge $\{v_3, u_2\}$.)

(D) Suppose that ($\{v_i, u_j\} \notin E$, $\{v_j, u_i\} \notin E$, $\{u_i, u_j\} \notin E$, and) a path $\langle u_i, w, u_j \rangle$ for some vertex $w \in V \setminus \{r, v_i, v_j\}$ exists. Note that $w \notin S$ can happen. We further distinguish this case into five subcases, (D-1) $w = v_k$ for some $k \notin \{i, j\}$, (D-2) $w \in N^2_G(r) \cap N_G(v_k)$ for some $k \notin \{i, j\}$, (D-3) $w \in N^2_G(r) \cap N_G(v_i)$, (D-4) $w \in N^2_G(r) \cap N_G(v_j)$, and (D-5) $w \notin N^2_G[r]$.

  (D-1) The case $w = v_k$ for some $k \notin \{i, j\}$: See Fig. 5 for an illustration. We assign the pair (u_i, u_j) to the second edge $\{v_k, u_j\}$ of the path $\langle u_i, v_k, u_j \rangle$. (In Fig. 4, the pair (u_3, u_4) is assigned to the edge $\{v_5, u_4\}$.)

  (D-2) The case $w \in N^2_G(r) \cap N_G(v_k)$ for some $k \notin \{i, j\}$: See Fig. 6 for an illustration. Look at the second edge $\{w, u_j\}$ of the path $\langle u_i, w, u_j \rangle$. From Lemma 2, there must exist $\{v_k, u_j\}$ and/or $\{v_j, w\}$ in E. If $\{v_k, u_j\} \in E$, we assign the pair (u_i, u_j) to $\{v_k, u_j\}$, otherwise, we assign the pair (u_i, u_j) to $\{w, v_j\}$.

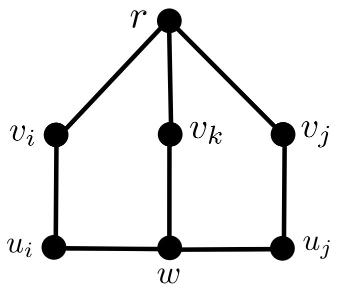

Fig. 6. (D-2) A path $\langle u_i, w, u_j \rangle$, where $w \in N_G^2(r) \cap N_G(v_k)$ for some $k \notin \{i, j\}$. From Lemma 2, $\{v_k, u_j\} \in E$ and/or $\{v_j, w\} \in E$.

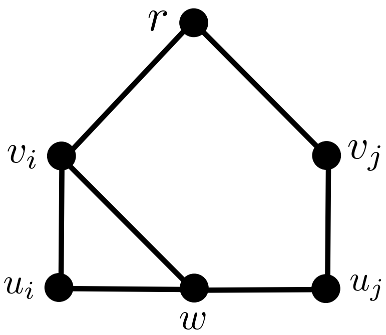

Fig. 7. (D-3) A path $\langle u_i, w, u_j \rangle$, where $w \in N_G^2(r) \cap N_G(v_i)$. From Lemma 2, $\{v_j, w\} \in E$.

(D-3) The case $w \in N_G^2(r) \cap N_G(v_i)$: See Fig. 7 for an illustration. As in (D-2), from Lemma 2, there must exist $\{v_j, w\}$ in E.

(D-4) The case $w \in N_G^2(r) \cap N_G(v_j)$: In this case, the edge $\{v_j, w\}$ exists in E. Thus, we assign the pair (u_i, u_j) to the edge $\{v_j, w\}$.

(D-5) The case $w \notin N_G^2[r]$: See Fig. 3 again for an illustration. By Lemma 3, there exists an edge $\{v_j, w\}$ in E, i.e., $w \in N_G(v_j)$. This contradicts the assumption $w \notin N_G^2[r]$, that is, this case does not occur.

By the above cases (A)-(D), every pair (u_i, u_j) for $u_i \in N'_G(v_i)$ and $u_j \in N'_G(v_j)$ for $i < j$ is assigned to an edge that emanates from $N_G(r)$.

(II) Next we estimate how many pairs can be assigned to a single edge between $N_G(r)$ and $N_G^2(r)$. Consider an edge $\{v_i, u_i\}$, where $v_i \in N_G(r)$ and $u_i \in N'_G(v_i)$. Recall that Case (A) does not assign any pair to $\{v_i, u_i\}$, and Case (C) is subsumed by Case (B). Therefore, we focus on Cases (B) and (D) in the following analysis.

(B) When $\{v_j, u_i\} \in E$, the pair (u_i, u_j) is assigned to the edge $\{v_j, u_i\}$: In this case, multiple pairs (u_i, u_h), where $h \in N'_G(v_j)$, may be assigned to the same edge $\{v_j, u_i\}$. Since $deg_G(v_j) \leq \Delta$, the number of pairs assigned to $\{v_j, u_i\}$ under Case (B) is at most Δ.

(D) As in the above discussion, we consider each of the distinguished cases:

(D-1) When $\{v_k, u_j\}$ is the second edge of a path $\langle u_i, v_k, u_j \rangle$, the pair (u_i, u_j) is assigned to the edge $\{v_k, u_j\}$: This edge $\{v_k, u_j\}$ may serve as the second edge of such a path $\langle u, v_k, u_j \rangle$ for some $u \in N^2_G(r) \cap S$. Since $deg_G(v_k) \leq \Delta$, the number of pairs assigned to $\{v_k, u_j\}$ under Case (D-1) is at most Δ.

(D-2) When $\{w, u_j\}$ is the second edge of a path $\langle u_i, w, u_j \rangle$, where $w \in N'_G(v_k)$ and $k \notin \{i, j\}$, the pair (u_i, u_j) is assigned to either $\{v_k, u_j\}$ or $\{v_j, w\}$: The edge $\{w, u_j\}$ may serve as the second edge of a path $\langle u, w, u_j \rangle$ or of a path $\langle u, u_j, w \rangle$ when $w \in S$, for some $u \in N^2_G(r) \cap S$. Since $deg_G(w) \leq \Delta$ and $deg_G(u_j) \leq \Delta$, at most 2Δ pairs can be assigned to each of the edges $\{v_k, u_j\}$ and $\{w, v_j\}$ under Case (D-2).

(D-3) When $\{w, u_j\}$, with $w \in N'_G(v_i)$, is the second edge of a path $\langle u_i, w, u_j \rangle$, the pair (u_i, u_j) is assigned to the edge $\{v_j, w\}$: Similar to Case (D-2), the edge $\{w, u_j\}$ may serve as the second edge of at most 2Δ paths of the form $\langle u, w, u_j \rangle$ and $\langle u, u_j, w \rangle$ (when $w \in S$), for some $u \in N^2_G(r) \cap S$. Thus, at most 2Δ pairs are assigned to the edge $\{v_j, w\}$ under Case (D-3).

(D-4) When $\{w, u_j\}$, with $w \in N'_G(v_j)$, is the second edge of a path $\langle u_i, w, u_j \rangle$, the pair (u_i, u_j) is assigned to the edge $\{v_j, w\}$: Similar to Cases (D-2) and (D-3), at most 2Δ pairs are assigned to the edge $\{v_j, w\}$ under Case (D-4).

In total, the number of pairs assigned to any single edge between $N_G(r)$ and $N^2_G(r)$ is at most

$$\Delta + \Delta + 2\Delta + 2\Delta + 2\Delta = 8\Delta,$$

where the first two terms Δ come from Cases (B) and (D-1), and the remaining three terms 2Δ correspond to Cases (D-2), (D-3), and (D-4), respectively.

(III) Since $|N_G(r)| \leq \Delta$, the number of edges between $N_G(r)$ and $N^2_G(r)$ is at most $|N_G(r)| \cdot \Delta \leq \Delta^2$. The number of pairs (u, u')'s, where $u, u' \in \bigcup_{j=1}^{d} N'_G(v_j)$, is $q(q-1)/2$. Thus, it must hold (as a necessary condition)

$$\frac{(q-1)^2}{2} \leq \frac{q(q-1)}{2} \leq \Delta^2 \cdot 8\Delta,$$

which implies

$$q \leq 4 \cdot \Delta^{3/2} + 1.$$

$\square$

Based on the above lemma, we obtain an upper bound of the size of the maximum 2-clique:

Corollary 1. *The size of the maximum* 2*-clique in a chordal graph* G *is at most* $4 \cdot \Delta(G)^{3/2} + \Delta(G) + 2$.

Proof. Let G be an input chordal graph, and S be the vertex set of the maximum 2-clique in G. If $N'_G(v_2) = \emptyset$, then $|S| \leq 2\Delta(G)$ from Lemma 1. If $N'_G(v_2) \neq \emptyset$, then

$$\begin{aligned}
|S| &= \left| \{r\} \cup N^*_G(r) \cup \bigcup_{j=1}^{d} N'_G(v_j) \right| \\
&\leq \left| \{r\} \cup N_G(r) \cup \bigcup_{j=1}^{d} N'_G(v_j) \right| \\
&\leq 1 + \Delta(G) + q \\
&\leq 4 \cdot \Delta(G)^{3/2} + \Delta(G) + 2,
\end{aligned}$$

where the first inequality is based on the fact $N^*_G(r) \subseteq N_G(r)$, the second inequality holds since $|\{r\} \cup N_G(r)| \leq 1 + \Delta(G)$, and then the last inequality is from Lemma 4. The lemma follows from the observation that $2\Delta(G) \leq 4 \cdot \Delta(G)^{3/2} + \Delta(G) + 2$. $\square$

Based on the above lemma, we can show that `FindStar` is an $O(n^{1/3})$-approximation algorithm:

Theorem 1. *`FindStar` is an* $O(n^{1/3})$*-approximation algorithm for* Max 2-Clique *if the input is a chordal graph.*

Proof. Let S^* denote an optimal set of vertices, and let $\Delta = \Delta(G)$. Then, `FindStar` outputs a 2-clique of $\Delta + 1$ vertices.

Consider the case $\Delta \geq n^{2/3}$. Since it is obvious that $|S^*| \leq n$, the approximation ratio of `FindStar` is at most $n/(\Delta + 1) \leq n^{1/3}$.

Next, consider the case $\Delta < n^{2/3}$. It holds $|S^*| \leq 4 \cdot \Delta^{3/2} + \Delta + 2$ from Corollary 1, and thus, the approximation ratio for this case is at most

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

Therefore, in every case, the approximation ratio of `FindStar` is $O(n^{1/3})$. $\square$

3.2 Max 2-Club on chordal graphs

Since a 2-club is a 2-clique, the maximum 2-clique is larger than the maximum 2-club. Thus, the upper bound of the maximum 2-clique size shown in Corollary 1 is transferred to Max 2-Club:

Corollary 2. *The size of the maximum* 2*-club in a chordal graph* G *is at most* $4 \cdot \Delta(G)^{3/2} + \Delta(G) + 2$.

Then, the proof of Theorem 1 also gives the next theorem:

Theorem 2. *`FindStar` is an* $O(n^{1/3})$*-approximation algorithm for Max 2-Club if the input is a chordal graph.*

4 Inapproximability

This section investigates the intractability of Max 2-Clique on chordal graphs. It is known that Max 2-Club on split graphs cannot be approximated in polynomial time within a factor of $O(n^{1/3-\varepsilon})$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$ [3]. Since every split graph is a chordal graph, the inapproximability result directly extends to chordal graphs: Max 2-Club on chordal graphs also cannot be approximated in polynomial time within a factor of $O(n^{1/3-\varepsilon})$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. In the previous section, we showed that both Max 2-Club and Max 2-Clique admit polynomial-time $O(n^{1/3})$-approximation algorithms for chordal graphs. Thus, the remaining task in this section is to show that Max 2-Clique on chordal graphs cannot be approximated in polynomial time within a factor of $O(n^{1/3-\varepsilon})$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. To this end, we prove the same inapproximability bound, $O(n^{1/3-\varepsilon})$ for split graphs, which immediately implies the corresponding inapproximability result for chordal graphs as well.

The next lemma shows that a maximum 2-clique in a split graph $G = (C, I, E)$ contains C as its vertex subset.

Lemma 5. *A maximum* 2*-clique in a split graph* $G = (C, I, E)$ *contains* C.

Proof. Let $S^* = C^* \cup I^*$ be a maximum 2-clique in a split graph $G = (C, I, E)$ such that $C^* \subseteq C$ and $I^* \subseteq I$. For the sake of contradiction, assume $C - C^* \neq \emptyset$. If $I^* = \emptyset$, it contradicts the assumption that S^* is a maximum 2-clique, since C is a clique and thus it is a 2-clique including more vertices than S^*. Hence, if $I^* = \emptyset$, $C^* = C$. Assume that $I^* \neq \emptyset$ in the following.

Let $v \in C \setminus C^*$. Since C is a clique, we have $dist_G(v, u) = 1$ for any $u \in C^*$. Now consider a vertex $u \in I^*$. If $v \in N_G(u)$, then clearly $dist_G(v, u) = 1$. Assume instead that $v \notin N_G(u)$, and let $v' \in C$ be a vertex in $N_G(u)$. Because C is a clique, we have $\{v', v\} \in E(G)$, and hence $dist_G(v, u) = 2$. Therefore, $S^* \cup \{v\}$ is also a 2-clique with $(|S^*|+1)$ vertices, contradicting the assumption that S^* is a maximum 2-clique. Consequently, the assumption $C - C^* \neq \emptyset$ is false, and thus $C^* = C$ holds. □

The next lemma shows that a maximum 2-clique is a 2-club in a split graph.

Lemma 6. *A maximum 2-clique in a split graph is a 2-club.*

Proof. Let $S^* = C \cup I^*$ be a maximum 2-clique in a split graph $G = (C, I, E)$ such that $I^* \subseteq I$, where S^* contains C, based on Lemma 5. Also let $H = G[S^*]$.

If $|I^*| = 0$, i.e., $I^* = \emptyset$, then $H = G[C]$ is a clique and hence a 2-club. Suppose that $|I^*| = 1$, and let $I^* = \{u\}$. Consider any vertex $v \in C$. If $v \in N_G(u)$, then $dist_H(v, u) = 1$. Otherwise if $v \notin N_G(u)$, there is a vertex $v' \in N_G(u)$ $(\subseteq C)$ such that there is a path $\langle u, v', v\rangle$ of length two in H, i.e., $dist_H(u, v) = 2$. Also, since C is a clique, for any $u, v \in C$, $dist_H(u, v) = 1$. Thus, the distance between any pair of two vertices is at most two in H, i.e., H is a 2-club.

As the last case, assume that $|I^*| \geq 2$. It holds $dist_H(u, v) \leq 2$, if either or both u and v are in C, according to the discussion for the case $|I^*| = 1$. Now, we consider two distinct vertices $u, v \in I^*$. By the definition of $I^* \subseteq I$, there is no edge between u and v. Since H is a 2-clique, there is a vertex w in G such that there is a path $\langle u, w, v\rangle$ of length two in G. It is clear that $w \in C$. Since $C \subseteq S^*$, $w \in S^*$ which ensures that the path $\langle u, w, v\rangle$ also exists in H implying that $dist_H(u, v) = 2$. Therefore H is a 2-club. □

Based on the above lemma, we show the following theorem.

Theorem 3. *In a split graph, a maximum 2-club is also a maximum 2-clique, and vice versa.*

Proof. Let H and H' be a maximum 2-club and a maximum 2-clique in a split graph, respectively. From Lemma 6, H' is also a 2-club. Thus, $|H'| \leq |H|$ holds. It is clear that a 2-club is also a 2-clique, which implies that $|H| \leq |H'|$. Therefore $|H| = |H'|$ holds, i.e., H' (or H) is a maximum 2-club (or a maximum 2-clique), too. $\square$

Since finding a maximum 2-clique is equivalent to finding a maximum 2-club in a split graph from the above theorem, the inapproximability of Max 2-Club on split graphs [3] is transferred to Max 2-Clique on split graphs, which also gives an inapproximability of Max 2-Clique on chordal graphs.

Corollary 3. *Max 2-Clique on split graphs (a subclass of chordal graphs) cannot be approximated in polynomial time to within a factor of* $n^{1/3-\varepsilon}$ *for any* $\varepsilon > 0$ *unless* $\mathcal{P} = \mathcal{NP}$.

5 Conclusion

We have shown that Max 2-Clique on chordal graphs admits a polynomial-time $O(n^{1/3})$-approximation algorithm. On the negative side, we also proved that Max 2-Clique on chordal graphs cannot be approximated in polynomial time within a factor of $n^{1/3-\varepsilon}$ for any $\varepsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. As a secondary result, we further established that Max 2-Club also admits a polynomial-time $O(n^{1/3})$-approximation algorithm for chordal graphs.

We conclude by listing several open problems:

- Is Max d-Clique on chordal graphs $\mathcal{NP}$-hard for $d \geq 3$?
  - It is known that Max d-Club on chordal graphs can be solved in polynomial time for odd $d \geq 3$ [13], and is $\mathcal{NP}$-hard for even $d \geq 4$ [3].
- What is the complexity of Max 2-Clique on bipartite graphs?
  - While Max 2-Club can be solved in polynomial time for bipartite graphs [3], it remains open whether Max 2-Clique is also polynomial-time solvable.
- Are there graph classes for which the complexities of Max d-Clique and Max d-Club differ?
  - To the best of our knowledge, the complexities of these two problems coincide for all graph classes studied so far (from the viewpoint of the polynomial-time solvability). Bipartite graphs may serve as a candidate for identifying such a separation.

Acknowledgments. This version of the contribution has been accepted for publication, after peer review (when applicable) but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: `http://dx.doi.org/10.1007/978-3-032-30521-3_15`.
Use of this Accepted Version is subject to the publisher's Accepted Manuscript terms of use `https://www.springernature.com/gp/open-science/policies/accepted-manuscript-terms`.

References

1. Alba, R.: A graph-theoretic definition of a sociometric clique. J. Math Sociol., **3**, 113–126(1973)
2. Asahiro, Y., Doi, Y., Miyano, E., Samizo, K., Shimizu, H.: Optimal approximation algorithms for maximum distance-bounded subgraph problems. Algorithmica, **80**(6), 1834–1856(2018)
3. Asahiro, Y., Miyano, E., Samizo., K.: Approximating maximum diameter-bounded subgraphs. In: LATIN 2010, LNCS, vol. 6034, pp.616–627(2010)
4. Abello, J., Resende, M.G., Sudarsky, S.: Massive quasi-clique detection. In: LATIN 2002, LNCS, vol. 2286, pp.598–612(2002)
5. Bomze, I.M., Budinich, M., Pardalos, P.M., Pelillo, M.: The maximum clique problem. In: Handbook of Combinatorial Optimization, pp.1–74, Kluwer Academic(1999)
6. Balasundaram, B., Butenko, S., Trukhanov, S.: Novel approaches for analyzing biological networks. J. Combinatorial Optimization, **10**(1), 23–39(2005)
7. Baril, A., Castillon, A., Oijid, N.: On the parameterized complexity of non-hereditary relaxations of clique, Theoretical Computer Science, **1003**, Article 114625(2024)
8. Bourjolly, J.-M., Laporte, G., Pesant, G.: An exact algorithm for the maximum k-club problem in an undirected graph. European J. Operational Research, **138**, 21–28(2002)
9. Brandstädt, A. Le, V.B., Spinrad, J.P.: Graph Classes: A Survey, SIAM, 1999.
10. Feige, U.: Approximating maximum clique by removing subgraphs. SIAM J. Discrete Mathematics, **18**, 219–225(2004)
11. Feránndez-Zepeda, J.A., Flores-Lamas, A., Hague, M., Trejo-ánchez, J.A.: A dynamic programming algorithm for the maximum s-club problem on trees, European J. Operational Research, **329**(2) 426–435(2026)
12. Gavril, F.: Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. SIAM J. Computing, **1**(2), 180–187(1972)
13. Golovach, P.A., Heggernes, P., Kratsch, D., Rafiey, A.: Finding clubs in graph classes. Discrete Applied Mathematics, **174**, 57–65(2014)
14. Karp, R.M.: Reducibility among combinatorial problems. Complexity of Computer Computations, 85–103(1972)
15. Luce, R.D.: Connectivity and generalized cliques in sociometric group structure. Psychometrika, **15**(2), 169–190(1950)

16. Mokken, R.J.: Cliques, clubs and clans. Quality and Quantity, **13**, 161–173(1979)
17. Pajouh, F.M., Balasundaram, B.: On inclusionwise maximal and maximum cardinality k-clubs in graphs. Discrete Optimization, **9**, 84–97(2012)
18. Seidman, S.B.: Network structure and minimum degree. Social Networks, **5**(3), 269–287(1983)
19. Seidman, S.B., Foster, B.L.: A graph-theoretic generalization of the clique concept. J. Mathematical Sociology, **6**(1), 139–154(1978)
20. Vazirani, V.V.: Approximation Algorithms. Springer, 2003.
21. Zuckerman, D.: Linear degree extractors and the inapproximability of max clique and chromatic number. Theory of Computing, **3**, 103–128(2007)