Recognizing Self-Clique Graphs

F. Larrión*§ V. Neumann-Lara*§ M. A. Pizaña† T. D. Porter‡

*Instituto de Matemáticas, U.N.A.M. 
Circuito Exterior, C.U. México 04510 D.F. MÉXICO.
{paco, neumann}@matem.unam.mx

†Universidad Autónoma Metropolitana, Depto. de Ingeniería Eléctrica.
Av. Michoacán y Purisima s/n México 09340 D.F. MÉXICO.
map@xanum.uam.mx, http://xamanek.uam.mx/map

‡Department of Mathematics, Southern Illinois University.
Carbondale, IL 62901-4408 USA.
tporter@math.siu.edu

Abstract

The clique graph $K(G)$ of a graph $G$ is the intersection graph of all the (maximal) cliques of $G$. A connected graph $G$ is self-clique if $G \cong K(G)$. Self-clique graphs have been studied since 1973. We propose recently a hierarchy of self-clique graphs: Type 3 \(\subset\) Type 2 \(\subset\) Type 1 \(\subset\) Type 0. Here we study the computational complexity of the corresponding recognition problems. We show that recognizing graphs of Type 0 and Type 1 is polynomially equivalent to the graph isomorphism problem. Partial results for Types 2 and 3 are also presented.

Keywords: clique graphs, self-clique graphs, vertex-clique bipartite graph, computational complexity, graph isomorphism problem.

1 Preliminaries

Self-clique graphs, discovered by Escalante in [7], have also been studied in [1, 4, 6, 11-13]. Hedman [10] asked if such graphs can be characterized. We refer to [15] for the bibliography on clique graphs. We learned recently that Balconi [2] also has related results. Our few undefined terms and symbols are standard and can be found in [5, 8, 9].

If $G$ is a (finite, simple) graph and $X \subseteq V(G)$, we denote by $G[X]$ the subgraph of $G$ induced by $X$, and we usually identify $X$ with $G[X]$. In particular we often write $x \in G$ instead of $x \in V(G)$, and identify the cliques of $G$ (which are maximal complete subgraphs) with their vertex sets.

We denote the distance between two vertices $x, y \in G$ by $d(x, y)$ or $d_G(x, y)$. The disk of radius $r$ centered at $x$ in $G$ is denoted by $D^r_G(x) = \{y \in G : d(x, y) \leq r\}$. When $r = 1$, $D^1_G(x) = N_G(x)$ is the closed neighbourhood of $x$. On the other hand, the neighbourhood $N_G(x)$ is the set of all neighbours of $x$ in $G$. 

---

*§Partially supported by CONACyT, Grant 400333-5-27668E.
We say that a vertex \( v \in G \) is dominated (by \( w \)) if \( N_G[v] \subseteq N_G[w] \) for some \( w \neq v \) in \( G \). For instance, in a triangleless graph, dominated means terminal. The \( n \)-th power graph \( G^n \) has \( V(G^n) = V(G) \), \( E(G^n) = \{x, y\} : d_G(x, y) \leq n \).

A family \( \mathcal{F} \) of subsets of a set \( X \neq \emptyset \) is \textbf{Helly} if \( \cap \mathcal{S} \neq \emptyset \) for any pairwise intersecting subfamily \( \mathcal{S} \subseteq \mathcal{F} \). A graph \( G \) is \textbf{Helly} if the family of cliques of \( G \) is Helly. For instance, every triangleless graph is Helly.

The \textbf{vertex-clique bipartite graph} (see [18]) \( BK(G) \) of \( G \) has \( V(BK(G)) = V(G) \cup V(K(G)) \) and \( E(BK(G)) = \{\{x, Q\} : x \in Q\} \). The neighbourhoods in \( BK(G) \) are as follows: \( N(Q) = Q \subseteq V(G) \) for \( Q \in K(G) \) and \( N(v) = v^* \subseteq V(K(G)) \) for \( v \in G \). Here \( v^* = \{Q \in K(G) : v \in Q\} \) is the \textit{star} of \( v \).

Let’s recall the hierarchy of self-clique graphs studied in [11]. A graph \( G \) is of \textbf{Type 0} if it is self-clique: connected and \( G \cong K(G) \). A graph \( G \) is of \textbf{Type 1} if it is a Helly self-clique graph. The distinction between Helly and non-Helly self-clique graphs was already made by Escalante in [7]. A connected graph \( G \) is \textbf{involutive} or of \textbf{Type 2} if \( B = BK(G) \) has a part-switching involution, that is, \( B \) has an automorphism \( \varphi : B \rightarrow B \) such that \( \varphi(V(G)) = V(K(G)) \), \( \varphi(V(K(G))) = V(G) \) and \( \varphi^2 = \text{id} \). It was shown in [11] that all previously published graphs of Type 1 were indeed of Type 2. Finally, a connected graph \( G \) is said to be \textbf{clique-disk} or of \textbf{Type 3} if \( G \) does not have dominated vertices and there is a graph \( R \) such that \( G = R^2 \) and the cliques of \( G \) are precisely the disks of radius 1 of \( R \), in symbols: \( V(K(G)) = \bigcup_{z \in G} \{N_R[z]\} \).

In this paper we are interested in the time complexity of recognizing whether a given graph \( G \) is of Type \( N \) for \( N = 0, 1, 2, 3 \). We shall use the following tags for the indicated decision problems:

- \textbf{ISO}: Graph isomorphism problem.
- \textbf{SELF}: Self-clique graph recognition.
- \textbf{HSELF}: Helly self-clique graph recognition.
- \textbf{INVO}: Involutive graph recognition.
- \textbf{CDISK}: Clique-disk graph recognition.

Our graphs are usually loopless, but for auxiliary purposes we also use \textbf{possibly loopy} graphs (always called \( H \)) that are allowed to have at most one loop at each vertex. Notice that under these circumstances, \( x \in N_H(x) \) iff \( H \) has a loop at \( x \). For such a possibly loopy graph we define the \textit{strict square} \( H^{[2]} \) as the (loopless) graph that has the same vertex set as \( H \) and in which two vertices \( x, y \) are adjacent iff they can be joined by two distinct edges \( \{x, u\} \) and \( \{u, y\} \) of \( H \) (here a loop counts as an edge).

We say that a possibly loopy graph \( H \) is \textbf{good} iff the family of neighbourhoods \( \{N_H(x) : x \in H\} \) is Helly and no neighbourhood is contained in another one: \( N_H(x) \subseteq N_H(y) \Rightarrow x = y \). We shall use the following theorems proved in [11]:

\textbf{Theorem 1.1} [11] \( BK(G) \) is good if and only if \( G \) is Helly without dominated vertices.

\textbf{Theorem 1.2} [11] A graph \( G \) is involutive if and only if \( G \cong H^{[2]} \) for some possibly loopy, good, connected, non-bipartite graph \( H \).

\textbf{Theorem 1.3} (The Hierarchy Theorem [11]) The following proper containment relations among the classes of self-clique graphs hold:

\[ \text{Type } 3 \subset \text{Type } 2 \subset \text{Type } 1 \subset \text{Type } 0 \]
2 Self-Clique Graphs

Let $G$ be a graph, with $p$ vertices, $q$ edges and $\mu$ maximal independent sets. Tsukiyama, Ide, Ariyoshi and Shirakawa [17] presented an algorithm (which we shall call the TIAS algorithm) that can compute all the maximal independent sets of $G$ in $O(pq\mu)$ time. Indeed this algorithm computes a new maximal independent set within every $O(pq)$ time interval.

Since we can complement a graph in $O(p^2)$ time, it follows that we can compute a polynomial number of clique in polynomial time. In particular, given a graph $G$ we can determine if it has exactly $|G|$ cliques (and compute them) in $O(p^2 (p^2 - q))$ time. Thus, in order to decide whether $G$ is self-clique or not, we can compute $K(G)$ (or stop with answer “no” if $|K(G)| \neq |G|$) in polynomial time and then apply an isomorphism test. It follows that SELF is polynomially reducible to ISO. Since we know by Szwarcflter [16] that Hellyness is polynomially verifiable, it is clear that HSELF is also polynomially reducible to ISO. We shall see here that the converses also hold.

We subdivide a graph $G$ by replacing each edge by a new path of length 2. If $\tilde{G}$ is the subdivision of $G$, then $\tilde{G}$ is bipartite and has a natural bipartition $\{X, Y\} = \{\text{old vertices, new vertices}\}$. If $G$ is connected so is $\tilde{G}$ and its bipartition is unique, so given $\tilde{G}$ and the fact that the part $X$ contains an old vertex (hence all) one recovers $G$ by $G = \tilde{G}[X]$. Note that, since every new vertex in $\tilde{G}$ has degree 2, whenever $G$ is connected and not a cycle it is quite easy to see which part contains the old vertices.

Let $G_1$ and $G_2$ be any two disjoint graphs. Take $G_1$ and add three extra vertices $\{x_1, y_1, z_1\}$, make $x_1$ adjacent to every vertex in $G_1 \cup \{y_1, z_1\}$ and make $y_1$ adjacent to every vertex in $G_1 \cup \{x_1, z_1\}$. Call the resulting graph $G_1'$. Now subdivide $G_1'$ to obtain $G_1''$. Do the same to $G_2$ with three other extra vertices $\{x_2, y_2, z_2\}$ to obtain $G_2'$ and then subdivide to get $G_2''$. Then $G_1''$ and $G_2''$ are connected, triangleless (therefore Helly) and without dominated (i.e. terminal) vertices. We also have that $G_1''$ and $G_2''$ are isomorphic iff $G_1$ and $G_2$ are so; indeed, the only maximal-degree vertices in $G_1''$ are the extra vertices $x_i$ and $y_i$, so any isomorphism $G_1'' \rightarrow G_2''$ induces an isomorphism $G_1' \rightarrow G_2'$ and so $G_1 \cong G_2$.

Now define a new graph $G_{12}$ by $V(G_{12}) = V(G_1'') \cup V(K(G_2''))$ and $E(G_{12}) = E(G_1'') \cup E(K(G_2'')) \cup \{\{z_1, Q\} : Q \in K(G_1'') \text{ and } z_2 \in Q\}$. This is just the disjoint union of $G_1''$ and $K(G_2'')$ plus 2 specific edges.

Theorem 2.1 Given any two graphs $G_1$ and $G_2$, construct $G_{12}$ as above. Then the following conditions are equivalent:

1. $G_1$ and $G_2$ are isomorphic.
2. $G_{12}$ is involutive.
3. $G_{12}$ is Helly self-clique.
4. $G_{12}$ is self-clique.

Proof. (1)⇒(2): If $G_1 \cong G_2$, there is an isomorphism $\tau : G_1'' \rightarrow G_2''$ satisfying $\tau(z_1) = z_2$. Then $\tau_K : K(G_1'') \rightarrow K(G_2'')$, defined by $\tau_K(Q) = \{\tau(x) : x \in Q\}$, is also an isomorphism. We know by 1.1 that $BK(G_1'')$ is good. Now attach a
loop at \( z_1 \) to \( BK(G''_1) \) to obtain \( H \). It is easy to check that \( H \) is still good, and it is clearly connected and non-bipartite. Since \( H^{[2]} \cong G_{12} \) via the isomorphism defined by \( \varphi(x) = x \) for \( x \in G''_1 \) and \( \varphi(Q) = \tau_K(Q) \) for \( Q \in K(G''_1) \), \( G_{12} \) is involutive by 1.2.

\((2) \Rightarrow (3) \Rightarrow (4)\): This follows from the Hierarchy Theorem 1.3.

\((4) \Rightarrow (1)\): Define \( G_{21} \) by \( V(G_{21}) = V(G''_2) \cup V(K(G''_1)) \) and \( E(G_{21}) = E(G''_2) \cup E(K(G''_1)) \cup \{z_2, Q \} : Q \in K(G''_1) \text{ and } z_1 \in Q \}. \) It is a routine verification to check that \( G_{21} \cong K(G_{12}) \) via the isomorphism defined by \( \varphi(z_2) = \{Q \in K(G''_2) : z_2 \in Q\} \cup \{z_1\}, \varphi(x) = \{Q \in K(G''_1) : x \in Q\} \) for \( x \neq z_2, x \in G''_2 \subseteq G_{21} \) and \( \varphi(Q) = Q \) for \( Q \in K(G''_1) \subseteq G_{21} \).

Now, assuming that \( G_{12} \cong K(G_{12}) \), there is an isomorphism \( \tau : G_{12} \to G_{21} \). By construction, \( G''_1 \) and \( G''_2 \) do not have cutpoints. Since the cliques of \( G''_1 \) are its edges, also \( K(G''_1) \) and \( K(G''_2) \) are cutpoint-free. Then \( z_1 \) (resp. \( z_2 \)) is the only cutpoint of \( G_{12} \) (resp. \( G_{21} \)). Now \( \tau(z_1) = z_2 \), so \( G''_1 \subseteq G_{12} \) must be mapped by \( \tau \) onto \( G''_2 \subseteq G_{21} \) or onto \( K(G''_1) \cup \{z_2\} \subseteq G_{21} \). Since \( G''_1 \) and \( G''_2 \) are triangleless but \( K(G''_1) \cup \{z_2\} \) is not, \( \tau(G''_1) = G''_2 \). Thus \( G''_1 \) and \( G''_2 \) are isomorphic, and so are \( G_1 \) and \( G_2 \). \( \square \)

Since \( G''_2 \) has \( |E(G''_2)| = 2|E(G_2)| + 4|V(G_2)| + 6 \) cliques, we can construct \( K(G''_2) \) and hence \( G_{12} \) in polynomial time. Then we have proved the following:

**Theorem 2.2** ISO is polynomially reducible to SELF, HSELF and INVO. Furthermore, SELF and HSELF are polynomially equivalent to ISO. \( \square \)

The authors of [4] have recently informed us that they also independently proved that ISO and SELF are polynomially equivalent.

**Problem 2.3** Determine the time complexity of INVO and CDISK.

### 3 Clique-Disk Graphs

By the previous section we only know that INVO is (up to a polynomial transformation) at least as difficult as ISO. But we know even less about the clique-disk recognition problem: We know nothing, apart from the obvious CDISK \( \in \mathcal{NP} \). Motwani and Sudan [14] showed that computing square roots of graphs is \( \mathcal{NP} \)-hard, which seems to suggest that CDISK could be \( \mathcal{NP} \)-complete. However, all the graphs constructed by Motwani and Sudan in their proof have exponentially many cliques, so those graphs are “highly non self-clique”, very far from our domain.

In [4], Bondy, Durán, Lin and Szwarzfiter introduced an important and large subclass of Type 3 (which indeed motivated the definition of Type 3 in [11]). The purpose of this section is to prove that the graphs in this subclass (which we shall call BDLS graphs) are recognizable in polynomial time.

A connected graph \( G \) is a **BDLS graph** if \( G = R^{2k} \) for some graph \( R \) with \( \delta(R) \geq 2, g(R) \geq 6k + 1 \) and \( k \geq 1 \). Here \( g(R) \) is the girth of \( R \).
**Theorem 3.1** Let $G$ be a graph. For each vertex $x \in G$ define recursively the sets $F_0 \supseteq F_1 \supseteq F_2 \supseteq \cdots$ by:

$$F_0(x) = x^* = \{Q \in K(G) : x \in Q\}$$

$$F_j(x) = \bigcup\{Q \in F_{j-1}(x) : Q \subseteq \bigcup(F_{j-1}(x) \setminus \{Q\})\}.$$ 

If $G = R^{2k}$ is a BDLS graph, then for all $j \geq 0$ and $x \in G$ we have

$$F_j(x) = \{D^k_R(y) : y \in D^{k-j}_R(x)\}.$$ 

Thus: $F_{k-1}(x) = \{D^k_R(y) : y \in N_R[x]\}$, $F_k(x) = \{D^k_R(x)\}$ and $F_{k+1}(x) = \emptyset$.

**Proof.** Let $G = R^{2k}$ be a BDLS graph. Recall from [4] (see also [3,11]) that: The cliques of $G$ are precisely the disks of radius $k$ of $R$, the rule $x \mapsto D^k_R(x)$ is an isomorphism from $G$ to $K(G)$ and each $D^k_R(x)$ induces in $R$ a tree of radius $k$ with all the leaves at distance $k$ from the center $x$.

Since $x \in D^k_R(y)$ if and only if $y \in D^k_R(x)$, we have $F_0(x) = \{D^k_R(y) : y \in D^k_R(x)\}$ as required for $j = 0$.

By induction, assume that $F_j(x) = \{D^k_R(y) : y \in D^{k-j}_R(x)\}$ for some $j$.

The set $D^{k-j}_R(x)$ induces a tree $T_x$ in $R$, and a vertex $y \in R$ is a leaf of $T_x$ if and only if $d_R(y,x) = k-j$. Now $y \in D^{k-j-1}_R(x) \iff N_R[y] \subseteq T_x \iff D^k_R(y) \subseteq \bigcup\{D^k_R(z) : z \in N_R[y] \cap T_x; z \neq y\} \iff D^k_R(y) \in F_{j+1}(x)$. \hfill \Box

Therefore, if $G = R^{2k}$ is a BDLS graph, $R$ and $k$ are determined by $G$. Indeed: $k$ is the number for which $|F_k(x)| = 1$ for all (or just one) $x \in G$ and we can reconstruct $R$ by $V(R) = V(G)$ and $\{x, y\} \in E(R)$ iff $x \neq y$ and $F_k(y) \subseteq F_{k-1}(x)$.

Now assume we want to determine whether a graph $G$ is a BDLS graph. Thanks to the TIAS algorithm [17], we can construct each $F_0(x)$ in polynomial time (or determine that $G$ does not have exactly $|V(G)|$ cliques, thus answering “no” and stopping computation). Then, as described above, we can also reconstruct $k$ and $R$ (or determine that there are no such $k$ and $R$) in polynomial time: Since we always have $F_j(x) = F_{j+1}(x)$ for some $j \leq |V(G)|$ we only have to compute (at worst) $|V(G)|^2$ of the $F_j(x)$’s. Finally, we just have to check that $G = R^{2k}$ (equality, not isomorphism!) $\delta(R) \geq 2$, $g(R) \geq 6k + 1$ and that $R$ is connected. It is clear that all these operations can be carried out in polynomial time, so we have proved:

**Theorem 3.2** BDLS graphs are recognizable in polynomial time. \hfill \Box

4 Final Remarks

Given two graphs $A$ and $B$, the strong product $A \boxtimes B$ is the loopless graph with vertex set $V(A \boxtimes B) = V(A) \times V(B)$ where $\{(a_i, b_i), (a_2, b_2)\} \in E(A \boxtimes B)$ iff $a_1$ and $a_2$ are adjacent or equal AND $b_1$ and $b_2$ are adjacent or equal.

Now, take $m, n \geq 7$ and $P = C_n \boxtimes C_m$ (here $C_n$ is a cycle of length $n$). A direct verification shows that $G = P^2$ satisfies $K(G) = \{N_P[v] : v \in P\}$, so it is
clique-disk. If we try our BDLS graph recognizing algorithm on this one, we get that for all \( v \in G \):

\[
F_0(v) = \{N_P[v + \alpha] : \alpha \in \{-1,0,1\} \times \{-1,0,1\}\},
F_1(v) = \{N_P[v + \alpha] : \alpha \in \{(0,1),(0,-1),(0,0),(1,0),(-1,0)\}\} \text{ and}
F_2(v) = \{N_P[v]\}.
\]

Then we define \( R \) by \( V(R) = V(G) = V(P) \) and \( \{u,v\} \in E(R) \) if and only if \( u - v \in \{(0,1),(0,-1),(1,0),(-1,0)\} \). Since \( k \) should be 2, we observe that \( \delta(R) = 4 \geq 2 \), but \( g(R) = 4 < 6k + 1 = 13 \) and \( G \neq R^4 \).

We conclude that the BDLS class is properly contained in Type 3, and that the final verifications in our algorithm are not superfluous (at least these two; \( g(R) \geq 6k + 1 \) and \( G = R^{2k} \)).

On the other hand we note that, in this case, computing \( F_2(v) = \{N_P[v]\} \) gives us the isomorphism \( v \leftrightarrow N_P[v] \) between \( G \) and \( K(G) \). If this were always the case for a clique-disk graph, we would have a polynomial time algorithm for CDISK. Unfortunately this is not so, since the clique-disk graph \( G = (R_8)^2 \) (see Fig. 1) has

\[
F_0(a_i) = \{N_{R_8}[v] : v \in \{a_{i-1}, a_i, a_{i+1}, x_{i-1}, x_i, b_i\}\},
F_1(a_i) = \{N_{R_8}[v] : v \in \{a_i, x_{i-1}, x_i, b_i\}\},
F_2(a_i) = \{N_{R_8}[a_i], N_{R_8}[b_i]\} \text{ and}
F_3(a_i) = \emptyset = F_4(a_i) = F_5(a_i) = \ldots
\]

Figure 1: The graph \( R_8 \) (identify vertices with same labels).

References


