

Three NP-Complete Optimization Problems in Seidel's Switching

Eva Jelínková

Department of Applied Mathematics
Faculty of Mathematics and Physics, Charles University
Malostranské nám. 25, 118 00 Praha 1, Czech Republic
eva@kam.mff.cuni.cz

Abstract. Seidel's switching is a graph operation which makes a given vertex adjacent to precisely those vertices to which it was non-adjacent before, while keeping the rest of the graph unchanged. Two graphs are called switching-equivalent if one can be made isomorphic to the other by a sequence of switches.

In this paper, we show the NP-completeness of the problem SWITCH- cn -CLIQUE for each $c \in (0, 1)$: determine if a graph G is switching-equivalent to a graph containing a clique of size at least cn , where n is the number of vertices of G . We also prove the NP-completeness of the problems SWITCH-MAX-EDGES and SWITCH-MIN-EDGES which decide if a given graph is switching-equivalent to a graph having at least or at most a given number of edges, respectively.

1 Introduction

The concept of Seidel's switching was introduced by a Dutch mathematician J. J. Seidel in connection with algebraic structures, such as systems of equiangular lines, strongly regular graphs, or the so-called two-graphs. For more structural properties of two-graphs, cf. [11–13]. Since then, switching has been studied by many others. Apart from the algebraic structures, consequences of switching arise in other research fields as well; for example, Seidel's switching plays an important role in Hayward's polynomial-time algorithm for solving the P_3 -structure recognition problem [6].

As proved by Colbourn and Corneil [4] (and independently by Kratochvíl et al. [9]), deciding whether two given graphs are switching equivalent is an isomorphism-complete problem.

In this paper, we prove the NP-completeness of several problems related to Seidel's switching. We examine the complexity of deciding if a given graph is switching-equivalent to a graph having a certain desired property. This paradigm has already been addressed by several authors. As observed by Kratochvíl et al. [9] and also by Ehrenfeucht et al. [5], there is no correlation between the complexity of the problem and the complexity of the property P itself. For example, Kratochvíl et al. [9] proved that any graph is switching-equivalent to a graph containing a Hamiltonian path, and it is polynomial to decide if a graph

is switching-equivalent to a graph containing a Hamiltonian cycle. However, the problems to decide if a graph itself contains a Hamiltonian path or cycle are well known to be NP-complete [7].

On the other hand, the problem of deciding switching-equivalence to a regular graph was proven to be NP-complete by Kratochvíl [10], and switching-equivalence to a k -regular graph for a fixed k is polynomial, while both the regularity and k -regularity of a graph can be tested polynomially. Three-colorability and switching-equivalence to a three-colorable graph are both NP-complete [5].

One of the problems we address in this paper is deciding switching-equivalence to a graph containing a clique of a certain size. It is well known that deciding for instances (G, k) if the graph G itself contains a clique of size at least k is NP-complete [7]. The corresponding switching problem, to decide for instances (G, k) if G is switching-equivalent to a graph with a clique of size at least k , was shown by Ehrenfeucht et al. [5] to be NP-complete as well. If k is fixed (not part of the input), then the problem can be solved by testing all induced subgraphs of size k . The whole graph G is switching-equivalent to a graph with a k -clique if and only if at least one induced subgraph of G on k vertices is switching-equivalent to a clique, and that can be determined in polynomial time. In Section 3 we extend the results of Ehrenfeucht et al. [5] by proving the NP-completeness of deciding switching-equivalence to a graph with a clique of size at least cn , where n is the number of vertices of G , for every fixed constant c in $(0, 1)$.

We further examine the complexity of problems SWITCH-MIN-EDGES and SWITCH-MAX-EDGES which for instances (G, k) decide if G is switching-equivalent to a graph with at most or at least k edges, respectively. We prove that both problems are NP-complete. Such a result may be unexpected, because switching a vertex affects the number of edges in a simple way.

On the other hand, Suchý [14] recently proved that the problems SWITCH-MIN-EDGES and SWITCH-MAX-EDGES are fixed-parameter tractable. Hence, for fixed k they are polynomial, which complements our result.

This paper is organized as follows. In Section 2, we introduce the notation and definitions used throughout the paper. In Section 3, we prove the NP-completeness of SWITCH- cn -CLIQUE. In Section 4 we prove the NP-completeness of SWITCH-MIN-EDGES and SWITCH-MAX-EDGES, and describe a connection of these problems to graph theoretic codes and the MAXIMUM LIKELIHOOD DECODING problem.

2 Basic Definitions

2.1 Preliminaries

In this paper, we use the standard graph theoretic notation. Unless defined otherwise, by n we denote the number of vertices of the currently discussed graph. The graph $G = (V, \binom{V}{2})$ is called a *complete graph* and denoted by K_n . A complete subgraph on k vertices is called a *k -clique*. A path with n vertices is denoted by P_n , and a graph with n vertices and no edges is called *discrete* and denoted by I_n . The symmetric difference of sets A and B is denoted by $A \triangle B$.

2.2 Seidel's Switching

Definition 1. Let G be a graph. Seidel's switch of a vertex $v \in V_G$ results in the graph called $S(G, v)$ whose vertex set is the same as of G and the edge set is the symmetric difference of E_G and the full star centered in v , i. e.,

$$V_{S(G,v)} = V_G$$

$$E_{S(G,v)} = E_G \setminus \{xv : x \in V_G, xv \in E_G\} \cup \{xv : x \in V_G, x \neq v, xv \notin E_G\}.$$

It is easy to observe that the result of a sequence of vertex switches in G depends only on the parity of the number of times each vertex is switched. This allows generalizing switching to vertex subsets of G .

Definition 2. Let G be a graph. Then the Seidel's switch of a vertex subset $A \subseteq V_G$ is called $S(G, A)$ and

$$S(G, A) = (V_G, E_G \Delta \{xy : x \in A, y \in V_G \setminus A\}).$$

We say that two graphs G and H are switching equivalent (denoted by $G \sim H$) if there is a set $A \subseteq V_G$ such that $S(G, A)$ is isomorphic to H .

3 Searching for a Switch with a cn -Clique

In this section, we consider the following problem.

Problem: SWITCH- cn -CLIQUE

Input: A graph G on n vertices

Question: Is G switching-equivalent to a graph containing a clique of size at least cn ?

Theorem 1. The problem SWITCH- cn -CLIQUE is NP-complete for any $c \in (0, 1)$.

Proof. We prove the theorem in two steps: first we prove the statement for rational numbers c only; then we extend it to numbers c which are irrational. For rational c , it is clear that the problem is in NP—a polynomial-size certificate contains vertex subsets A and C such that $S(G, A)[C]$ is a clique of the desired size. In the case of irrational c , we assume that an oracle can be used for querying the bits of c in constant time. This ensures that the certificate can be checked in time polynomial in n .

In the first step, we show the NP-hardness of the problem by reducing SAT to it, whereas in the second step we reduce 3-SAT. Both SAT and 3-SAT are well known to be NP-complete [7].

Suppose that c is rational and equal to $\frac{p}{q}$, where $p, q \in \mathbb{N}$, and $p < q$. We have an instance of SAT: a formula φ in CNF with k clauses and l occurrences of literals, and ask if φ is satisfiable. Without loss of generality we can assume that $k < l$ and $k \geq 2$.

Let $G = G_{p,q}(\varphi)$ be a graph constructed in the way illustrated in Fig. 1. The vertices of G are $V_G = L \cup K \cup Z$, where L, K, Z are pairwise disjoint and

$$\begin{aligned} |L| &= l, \\ |K| &= pl + p - k, \\ |Z| &= (q - p - 1)(l + 1) + k + 1. \end{aligned}$$

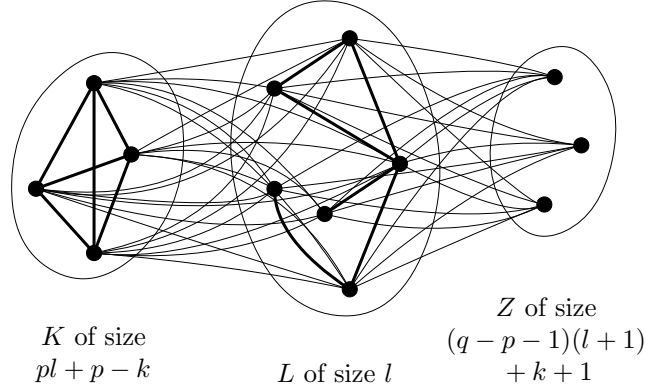


Fig. 1. The graph $G_{p,q}(\varphi)$.

The edges of G are defined as follows:

- K induces a clique and every vertex in K is adjacent to all vertices in L and no vertex in Z .
- Every vertex in Z is adjacent to all vertices in L and nothing more.
- Vertices of L represent occurrences of literals in φ . Two vertices $l_1, l_2 \in L$ are adjacent if and only if
 - l_1 and l_2 occur in different clauses and
 - they are not in the form $l_1 = \neg l_2$ nor $l_2 = \neg l_1$.

Lemma 1. *Let j be an integer. The graph $G_{p,q}(\varphi)[L]$ contains a j -clique if and only if the formula φ contains j simultaneously satisfiable clauses.*

Proof. Mutually adjacent vertices of $G_{p,q}(\varphi)[L]$ correspond to simultaneously satisfiable literals in distinct clauses. \square

Corollary 1. *The formula φ is satisfiable if and only if $G_{p,q}(\varphi)[L]$ contains a clique of size k (where k is the number of clauses in φ).*

Let us now consider cliques of size $pl + p$ in the whole graph—either in the original graph G or in its switches. The reader can verify that $n = |L \cup K \cup Z| = ql + q$ and $(pl + p)/n = c$, therefore cliques of size $pl + p$ are exactly cn -cliques.

Lemma 2. *The following statements are equivalent for $G = G_{p,q}(\varphi)$.*

- (a) *The graph $G[L]$ contains a k -clique.*
- (b) *The graph G contains a $(pl + p)$ -clique.*
- (c) *There exists a set $A \subseteq V_G$ such that $S(G, A)$ contains a $(pl + p)$ -clique.*

Proof. First we prove that (a) implies (b). Any clique in $G[L]$ forms a larger clique together with all vertices of K . So, if $G[L]$ contains a k -clique, then $G[L \cup K]$ contains a clique of size $k + (pl + p - k) = pl + p$.

The implication from (b) to (c) is obvious.

To prove that (c) implies (a), suppose that there is a set $A \subseteq V_G$ such that $S(G, A)$ contains a $(pl + p)$ -clique on a vertex set C .

The set C does not contain more than two vertices of Z , because they are pairwise non-adjacent in G and in $S(G, A)$ they induce a bipartite graph. From the assumptions $k < l$ and $k \geq 2$ it follows that $l > 2$, and $p \geq 1$, so $pl + p > 2$. Therefore C contains some vertices of L or K . But all vertices of Z are non-adjacent in G and have the same neighborhood in $G[L \cup K]$; surely all vertices in $Z \cap C$ have the same neighborhood in $S(G, A)[C]$ (otherwise C would not induce a clique). But then either $(Z \cap C) \subseteq A$ or $(Z \cap C) \cap A = \emptyset$, so switching A does not affect edges inside $S(G, A)[Z \cap C]$ and any two vertices in $S(G, A)[Z \cap C]$ are non-adjacent. Therefore C contains at most one vertex of Z .

Since $1 + |K| = 1 + (pl + p - k) < pl + p$, the clique C contains at least one vertex of L . But then it cannot contain both vertices of K and Z , because in the graph G they have the same neighborhood in L and there is no edge between K and Z . Also, the set C cannot consist only of vertices of L , because $pl + p > l$. Therefore C contains one of the following:

- $pl + p - 1$ (which is at least k) vertices of L and one vertex of Z
- at least k vertices of L and at least one vertex of K .

In both cases, C contains k vertices of L , and a vertex v of $K \cup Z$. Since C induces a clique in $S(G, A)$, the vertex v is adjacent to all other vertices in C . But in G , by definition, the vertex v is adjacent to all vertices of L , too. So switching A cannot have changed any edge connecting v and the k vertices, which means that either all these $k + 1$ vertices are in A or none of them is. But then they induce a $(k + 1)$ -clique in G as well, and $G[L]$ contains a k -clique, which we wanted to prove. \square

Corollary 1 and Lemma 2 together give us that φ is satisfiable if and only if there exists a set $A \subseteq V_G$ such that $S(G, A)$ contains a $(pl + p)$ -clique. But we have already shown that $pl + p = cn$; and clearly a graph contains a clique of size *exactly* cn if and only if it contains a clique of size *at least* cn . That concludes the reduction. The graph $G_{p,q}(\varphi)$ with $q(l + 1) = \mathcal{O}(l)$ vertices and $\mathcal{O}(l^2)$ edges can be constructed in time polynomial in the size of φ . Hence the problem SWITCH- cn -CLIQUE is NP-complete for every rational constant $c \in (0, 1)$.

Proving the NP-hardness of SWITCH- cn -CLIQUE for irrational numbers c is slightly more complicated. We use a theorem of Arora et al. [1] and certain

number theoretic results to get suitable numbers p, q , and then, analogously to the rational case, we reduce an instance of 3-SAT to SWITCH- cn -CLIQUE for the graph $G_{p,q}$. Due to space limitations, the rest of the proof is placed in the Appendix. \square

4 Minimizing the Number of Edges

In this section, we prove the NP-completeness of the following problem:

Problem: SWITCH-MIN-EDGES

Input: A graph G , an integer k .

Question: Is G switching-equivalent to a graph with at most k edges?

The problem SWITCH-MAX-EDGES is defined analogously with “at most” replaced by “at least”. It is easy to observe that these two problems are polynomially equivalent. Therefore it suffices to show the NP-completeness of SWITCH-MIN-EDGES only, which is done in the proof of Theorem 3.

4.1 The Connection to Maximum Likelihood Decoding

Let V be a fixed set of n vertices. For an edge set $E \subseteq \binom{V}{2}$, by χ_E we denote the characteristic vector of E , i. e., the element of $\mathbb{Z}_2^{\binom{n}{2}}$ such that $\chi_E(e) = 1$ if and only if $e \in E$. Thus any graph on the vertex set V can be represented by a vector of length $\binom{n}{2}$. The following observation expresses how switching works by means of characteristic vectors.

Observation 2 *Let $K_n = (V, \binom{V}{2})$ be the complete graph, let V_1, V_2 be a partition of V and let $S = \{\{x, y\}, x \in V_1, y \in V_2\}$ be the corresponding cut in K_n . Then for any $G = (V, E)$,*

$$\chi_{S(G, V_1)} = \chi_{S(G, V_2)} = \chi_E + \chi_S.$$

(Note that the summation is done over \mathbb{Z}_2 .) Therefore, if we seek a switch of G with the minimum number of edges, we seek a characteristic vector $\chi_E + \chi_S$ with the minimum Hamming weight. Or, equivalently, we seek a cut S in K_n with the minimum Hamming distance between χ_S and χ_E .

It is a well-known fact that the cut space $\mathcal{C}^*(G)$ of a graph G is a vector space, and the cycle space $\mathcal{C}(G)$ is also a vector space orthogonal to $\mathcal{C}^*(G)$. The dimension of $\mathcal{C}^*(G)$ is $|V| - 1$, and $\mathcal{C}^*(G)$ can also be viewed as a linear $[|E|, |V| - 1]$ code with a parity-check matrix C whose rows are $|E| - |V| + 1$ linearly independent characteristic vectors of cycles in G . Such a code is called a *graph theoretic code*; the concept of graph theoretic codes has been introduced by Hakimi and Frank [8].

The problem of finding a codeword in a linear code that is closest to a given vector is an important problem in coding theory. It can be formulated as a decision problem in the following way.

Problem: MAXIMUM LIKELIHOOD DECODING

Input: A binary $p \times q$ matrix H , a vector $r \in \mathbb{Z}_2^p$, and an integer $w > 0$.

Question: Is there a vector $e \in \mathbb{Z}_2^q$ of Hamming weight at most w such that $He = r$?

This problem was proven to be NP-complete by Berlekamp et al. [2]. Note that SWITCH-MIN-EDGES is indeed its special case, which we formalize in the following lemma (proved in the Appendix).

Lemma 3. SWITCH-MIN-EDGES is a special case of MAXIMUM LIKELIHOOD DECODING, where H is the parity check matrix of the code of cuts in a complete graph.

Special cases of MAXIMUM LIKELIHOOD DECODING have been studied. It is known that the problem is NP-complete even if we allow unbounded time for preprocessing the code H . This was proven by Bruck and Naor [3] by showing that MAXIMUM LIKELIHOOD DECODING is NP-complete for the cut code of a special fixed graph, and therefore no preprocessing can help because this fixed code can be known in advance. Our proof in Subsection 4.2 provides an alternative proof of Bruck and Naor's result by using K_n as the fixed graph.

4.2 Proof of NP-Completeness

We use a reduction of the following well-known NP-complete problem [7].

Problem: SIMPLE-MAX-CUT

Input: A graph G , an integer j .

Question: Does there exist a partition V_1, V_2 of V_G such that the cut between V_1 and V_2 in G contains at least j edges?

Theorem 3. SWITCH-MIN-EDGES is NP-complete.

Proof. From an instance (G, j) of SIMPLE-MAX-CUT we create an instance (G', k) of SWITCH-MIN-EDGES in the following way. For each vertex of G we create a corresponding non-adjacent vertex pair in G' . An edge in G is represented by four edges completely interconnecting the two pairs, and a non-edge in G is represented by only two edges connecting the two pairs in a parallel way. More formally, we set

$$V_{G'} = \{v', v'' : v \in V_G\}$$

$$E_{G'} = \{\{u', v'\}, \{u'', v''\} : u, v \in V_G, u \neq v\} \cup \{\{u', v''\}, \{u'', v'\} : \{u, v\} \in E_G\}.$$

The following lemma relates cuts in G with switches of G' .

Lemma 4. The following statements are equivalent:

- (a) There is a cut in G having at least j edges,
- (b) there exists a set $A \subseteq V_{G'}$ such that $S(G', A)$ contains at most $2\binom{|V_G|}{2} + 2|E_G| - 4j$ edges.

Proof. For a cut C in G we define a corresponding vertex subset $A = A(C)$ of G' . Suppose that C separates a vertex set V_1 from the remaining vertices of G . We set A to be the set $\{v', v'' : v \in V_1\}$. Note that such a set A satisfies the following condition.

Definition 3. *We say that a vertex subset of G' is legal if it contains an even number of vertices out of the pair v', v'' for every $v \in V_G$. Otherwise, we say that it is illegal.*

Legality is a desired property, because there is an obvious correspondence between legal sets and cuts in G . Also, the number of edges in $S(G', A)$ is determined by the size of the cut C . The original graph G contains two edges per every vertex-pair of G and two more edges per every edge in G , which is $2\binom{|V_G|}{2} + 2|E_G|$ edges altogether. Since A is legal, it can easily be checked that every non-edge $\{u, v\}$ in G corresponds to two edges in both G' and $S(G', A)$, regardless of the cut C . For every edge $\{u, v\}$ in the cut C we have $u', u'' \in A$ and $v', v'' \notin A$ (or vice versa), so switching A destroys all the four corresponding edges and creates none. For an edge not present in the cut C , switching A does not modify the corresponding edges, so there are still four of them in $S(G', A)$. To sum it up, $S(G', A)$ has

$$2\binom{|V_G|}{2} + 2|E_G| - 4|C|$$

edges. This proves that the statement (a) implies (b). As for the other implication, by reverting the construction of $A(C)$ from a cut C , we get that it holds for legal sets A . It remains to deal with possible illegal switches. For that purpose, we introduce another definition.

Definition 4. *We say that a vertex $u \in V_G$ is broken in A if A contains exactly one vertex of u', u'' . We say that a vertex set $\{u, v\} \subseteq V_G$ is broken in A if at least one of its vertices u, v is broken. Otherwise, we say that it is legal in A .*

Lemma 5. *For every illegal set A there is a legal set A' such that $S(G', A')$ contains at most the same number of edges as $S(G', A)$.*

Proof. Let A be an illegal set. As can be seen in Fig. 2, a broken non-edge never decreases the resulting number of edges, and thus is no more profitable than a legal non-edge. Therefore, if all the broken pairs in A correspond to non-edges, then the set A minus the union of all broken pairs is legal, and it yields a switch with at most the same number of edges as A does.

Assume that there are m broken edges, where $m > 0$. As shown in Fig. 2, a broken edge could in certain cases decrease the number of edges in G' by more than a legal edge not present in the cut would. We create a legal set A' from A using the following greedy algorithm.

1. $A' := A$.

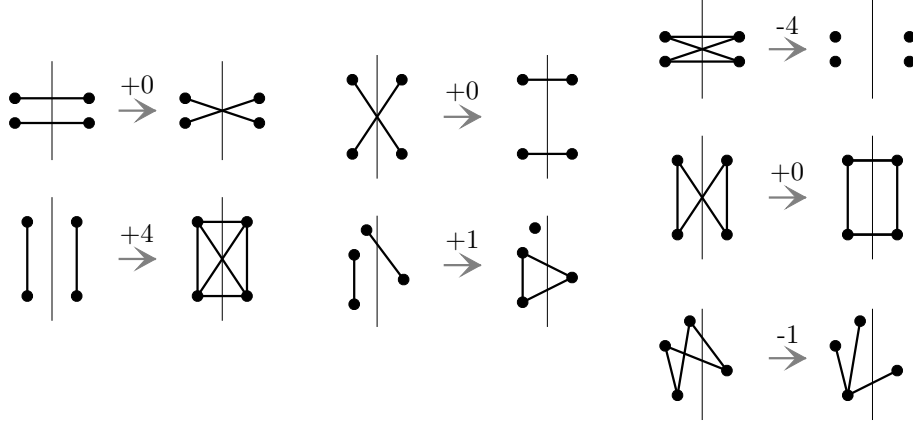


Fig. 2. The possible legal and illegal switches of non-edges (on the left, in the middle) and edges (on the right) in G and their influence on the number of edges in G' .

2. Find a vertex v broken in A' . If there is no such, STOP.
3. Look for vertices u that are legal in A' and such that $\{u, v\}$ is an edge in G . If there are more such vertices in A' than in $V_G \setminus A'$, set $A' := A' \setminus \{v', v''\}$, otherwise set $A' := A' \cup \{v', v''\}$.
4. Go back to step 2.

It remains to prove that the algorithm finds a legal set that is better than A .

In each iteration of step 3, the algorithm legalizes one vertex. It clearly finishes after a finite number of steps and creates a legal set. It does not modify legal vertices nor legal edges in A , therefore the number of edges in $S(G', A')$ corresponding to legal sets in A remains unchanged. Also, legalizing a non-edge does not increase the number of edges in $S(G', A')$ in comparison to $S(G', A)$.

As we already know, any legal set gives us a cut in G ; consider the cut obtained from A' . In each iteration of Step 3, at least half of the newly legalized edges became cut edges with one endpoint in A' and the other not in A' . Since each broken edge was legalized exactly once, we know that at least $m/2$ legalized edges became cut edges (and decreased the edge number by at least 3), and at most $m/2$ legalized edges became out of the cut (and increased the edge number by at most 1). Therefore, the edge number in $S(G', A')$ is lower by at least $3m/2 - m/2 = m$, which we assumed to be positive. \square

To finish the proof of Lemma 4, it remains to prove that (b) implies (a). Let $S(G', A)$ be a switch with at most $2\binom{|V_G|}{2} + 2|E_G| - 4j$ edges. If A is illegal, Lemma 5 assures that there is a legal set A' such that $S(G', A')$ has even less edges. The legal set yields a partition $V_1 = \{v : v', v'' \in A'\}$ and $V_2 = V_G \setminus V_1$ such that the cut between V_1 and V_2 contains at least j edges. \square

According to Lemma 4, the graph G contains a cut of size at least j if and only if G' is switching-equivalent to a graph with at most $k = 2\binom{|V_G|}{2} + 2|E_G| - 4j$

edges. The size of (G', k) is surely polynomial in the size of (G, j) . That concludes the proof of NP-hardness of SWITCH-MIN-EDGES. To prove that SWITCH-MIN-EDGES is in NP, it suffices to note that the positive answer can be certified by a vertex set A that gives us a switch with the desired number of edges. \square

Acknowledgement

I am grateful to Jan Kratochvíl for numerous useful tips and valuable discussions; and to Jiří Sgall for advice concerning some relevant references.

References

1. S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy: *Proof verification and the hardness of approximation problems*, Journal of ACM **45(3)** (1998), pp. 501–555. An extended abstract appeared in FOCS 1992.
2. E. R. Berlekamp, R. J. McEliece, H. C. A. van Tilborg, *On the inherent intractability of certain coding problems*, IEEE Trans. Inform. Theory, vol **IT-24** (1978), pp. 384–386.
3. J. Bruck, M. Naor, *The hardness of decoding linear codes with preprocessing*, IEEE Trans. Inform. Theory, vol **36** (1990), pp. 381–384.
4. C. J. Colbourn, D. G. Corneil, *On deciding switching equivalence of graphs*, Discrete Appl. Math, vol **2** (1980), pp. 181–184.
5. A. Ehrenfeucht, J. Hage, T. Harju, G. Rozenberg: *Complexity issues in switching classes*, in: Theory and Application to Graph Transformations, LNCS 1764, Springer-Verlag (2000), pp. 59–70.
6. R. B. Hayward: *Recognizing P_3 -structure: A switching approach*, J. Combin. Th. Ser. B **66** (1996), pp. 247–262.
7. M. R. Garey, D. S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, 1979.
8. S. L. Hakimi, H. Frank: *Cut-set matrices and linear codes*, IEEE Trans. Inform. Theory, vol. **IT-11** (1965), pp. 457–458.
9. J. Kratochvíl, J. Nešetřil, O. Zýka: *On the computational complexity of Seidel's switching*, Proc. 4th Czech. Symp., Prachatice 1990, Ann. Discrete Math. **51** (1992), pp. 161–166.
10. J. Kratochvíl: *Complexity of hypergraph coloring and Seidel's switching*, in: Graph Theoretic Concepts in Computer Science (H. Bodlaender, ed.), Proceedings WG 2003, Elspeet, June 2003, LNCS 2880, Springer Verlag (2003), pp. 297–308.
11. J. J. Seidel: *Graphs and two-graphs*, in: Proc. 5th. Southeastern Conf. on Combinatorics, Graph Theory, and Computing, Winnipeg, Canada (1974).
12. J. J. Seidel: *A survey of two-graphs*, Teorie combinatorie, Atti Conv. Lincei, Vol 17, Accademia Nazionale dei Lincei, Rome (1973), pp. 481–511.
13. J. J. Seidel, D. E. Taylor: *Two-graphs, a second survey*, Algebraic methods in graph theory, Vol. II, Conf. Szeged 1978, Colloq. Math. Janos Bolyai **25** (1981), pp. 689–711.
14. O. Suchý: *Some Parametrized Problems Related to Seidel's Switching*, to appear in Proceedings of IWOCA 2007.
15. V. V. Vazirani: *Approximation Algorithms*, Springer-Verlag, 2001.

Appendix

Proof (of Theorem 1, continued). To prove the NP-hardness of SWITCH- cn -CLIQUE for irrational numbers c , we use a theorem of Arora et al. proved in [1] (and restated in this way in [15] as Lemma 29.10).

Theorem 4. (Arora, Lund, Motwani, Sudan, Szegedy) *There exists a polynomial time transformation T from 3-CNF to 3-CNF and a constant $\varepsilon > 0$ such that*

- *If ψ is satisfiable, then $T(\psi)$ is satisfiable.*
- *If ψ is not satisfiable, then at most $1 - \varepsilon$ fraction of the clauses of $T(\psi)$ are simultaneously satisfiable.*

Our aim is to do a reduction from an instance ψ of 3-SAT. We will use the graph $G_{p,q}$, like in the previous part of the proof; this time for the transformed formula $T(\psi)$ and for numbers p and q such that $\frac{p}{q}$ is sufficiently close to the irrational number c . Then we examine the relationship between cn -cliques and $\frac{p}{q}$ -cliques in the resulting graph. To show that some suitable numbers p and q exist, we will make use of Lemma 6, which is a variant of Dirichlet's Theorem, and Lemma 7.

Lemma 6. *For any real number $\alpha \in [0, 1]$, any $\varepsilon > 0$ and $r \in \mathbb{R}$ there exists $n \in \mathbb{N}$ such that $n > r$ and $\{n\alpha\} < \varepsilon$ (where $\{n\alpha\}$ stands for the decimal fraction of $n\alpha$).*

Proof. Without loss of generality we can assume that $\varepsilon < \alpha$ and $\alpha \in (0, 1)$. We prove by induction that for each $k \in \mathbb{N}_0$ there exists $n_k \in \mathbb{N}$ such that

$$\{n_k \alpha\} \leq \frac{\alpha}{2^k}$$

and $n_{k+1} > n_k$ for all k . Then we take $n = n_k$ for $k \geq \max\{r, \log_2(\frac{\alpha}{\varepsilon})\}$.

We set $n_0 = 1$, because $\{1 \cdot \alpha\} \leq \frac{\alpha}{1}$. Now assume that we already have n_k for some $k \geq 0$ and want to find n_{k+1} . Let $\beta = \{n_k \alpha\}$; we want to get an integer m so that $\{m\beta\} \leq \frac{\beta}{2}$ and $m > 1$. If $\beta = 0$, then clearly the inequality $\{m\beta\} \leq \frac{\beta}{2}$ holds for any integer m , so we can set $m = 2$. Otherwise we consider the number $\lfloor \frac{1}{\beta} \rfloor \beta$. It is clear that $\lfloor \frac{1}{\beta} \rfloor \beta \leq 1$; in case of an equality we have that $\{\lfloor \frac{1}{\beta} \rfloor \beta\} = 0$, while $\lfloor \frac{1}{\beta} \rfloor$ is nonzero. Hence m can be either $\lfloor \frac{1}{\beta} \rfloor$ or any its integral multiple larger than 1.

The remaining case is that $\beta > 0$ and $\lfloor \frac{1}{\beta} \rfloor \beta < 1$. Then $1 < \lceil \frac{1}{\beta} \rceil \beta < \beta + 1$, and after subtracting 1 we get that

$$\left\{ \left\lceil \frac{1}{\beta} \right\rceil \beta \right\} < \beta. \tag{1}$$

We want m to be an integer such that $\{m\beta\} \leq \frac{\beta}{2}$. Note that $\lceil \frac{1}{\beta} \rceil > 1$, since $\lfloor \frac{1}{\beta} \rfloor > 0$ for any $\beta \in (0, 1)$. So suppose that m cannot be $\lceil \frac{1}{\beta} \rceil$, because

$\{\lceil \frac{1}{\beta} \rceil \beta\} > \frac{\beta}{2}$. Then we define

$$\delta = \beta + 1 - \left\lceil \frac{1}{\beta} \right\rceil \beta.$$

The assumption $\{\lceil \frac{1}{\beta} \rceil \beta\} > \frac{\beta}{2}$ together with (1) imply that $\delta \in (0, \frac{\beta}{2})$. Similarly like before we obtain the inequalities $\lfloor \frac{\beta}{2\delta} \rfloor \delta \leq \frac{\beta}{2}$ and $\frac{\beta}{2} \leq \lceil \frac{\beta}{2\delta} \rceil \delta$. Moreover, it is surely true that $\lceil \frac{\beta}{2\delta} \rceil \leq (1 + \lfloor \frac{\beta}{2\delta} \rfloor)$; hence

$$\frac{\beta}{2} \leq \left\lceil \frac{\beta}{2\delta} \right\rceil \delta \leq \beta. \quad (2)$$

Now we set

$$m = \left\lceil \frac{\beta}{2\delta} \right\rceil \left(\left\lceil \frac{1}{\beta} \right\rceil - 1 \right) + 1.$$

It is clear that such an m is larger than one, and by substituting δ according to its definition, it can be easily verified that

$$m\beta = \left\lceil \frac{\beta}{2\delta} \right\rceil - \left\lceil \frac{\beta}{2\delta} \right\rceil \delta + \beta. \quad (3)$$

By plugging the inequalities of (2) into (3), we obtain

$$\left\lceil \frac{\beta}{2\delta} \right\rceil \leq m\beta \leq \left\lceil \frac{\beta}{2\delta} \right\rceil + \frac{\beta}{2},$$

which immediately gives us that $\{m\beta\} \leq \frac{\beta}{2}$, and that is what we wanted. It now remains to set $n_{k+1} = mn_k$ and verify that $\{n_{k+1}\alpha\} \leq \frac{\alpha}{2^{k+1}}$. Indeed, we have that

$$\{n_{k+1}\alpha\} = \{mn_k\alpha\} = \{m\beta\} < \frac{\beta}{2} = \frac{\{n_k\alpha\}}{2} \leq \frac{\frac{\alpha}{2^k}}{2} = \frac{\alpha}{2^{k+1}},$$

where the last inequality holds by the induction hypothesis. In all cases considered we chose m to be larger than one, hence $n_{k+1} > n_k$, and we are done. \square

Lemma 7. *For each irrational $c \in (0, 1)$ and $\varepsilon > 0$, there exist $p, q \in \mathbb{N}$ such that $\frac{p}{q} \in (0, 1)$ and*

$$c \in \left(\left(1 - \frac{\varepsilon}{4p} \right) \frac{p}{q}, \frac{p}{q} \right).$$

Proof. We shall find an integer p such that the interval $(\frac{p}{c} - \frac{\varepsilon}{4c}, \frac{p}{c})$ contains another integer q . We want p to satisfy the condition

$$\left\{ \frac{p}{c} \right\} < \frac{\varepsilon}{4c}, \quad (4)$$

and additionally we request that

$$p > \frac{\varepsilon}{4(1-c)}. \quad (5)$$

It is true that $\{\frac{p}{c}\} = \{p\{\frac{1}{c}\}\}$, the number $\{\frac{1}{c}\}$ lies in the interval $(0, 1)$, and surely $\frac{\varepsilon}{4c} > 0$; hence Lemma 6 for $\alpha = \{\frac{1}{c}\}$, $\varepsilon' = \frac{\varepsilon}{4c}$ and $r = \frac{\varepsilon}{4(1-c)}$ ensures the existence of such a p .

Then we set $q = \lfloor \frac{p}{c} \rfloor$ and verify that it is really an integer in the interval $(\frac{p}{c} - \frac{\varepsilon}{4c}, \frac{p}{c})$. The number $\frac{p}{c}$ is irrational, so we have $q < \frac{p}{c}$. The fact that $\lfloor \frac{p}{c} \rfloor = \frac{p}{c} - \{p\{\frac{1}{c}\}\}$, and (4) together give us the other inequality $q > \frac{p}{c} - \frac{\varepsilon}{4c}$.

Moreover, from (5) we obtain

$$\frac{p}{c} - \frac{\varepsilon}{4c} > p,$$

so any integer q in the interval $(\frac{p}{c} - \frac{\varepsilon}{4c}, \frac{p}{c})$ is larger than p , and thus $\frac{p}{q} \in (0, 1)$. Also, by rewriting the inequalities $q < \frac{p}{c}$ and $q > \frac{p}{c} - \frac{\varepsilon}{4c}$ we get the desired inequality

$$\left(1 - \frac{\varepsilon}{4p}\right) \frac{p}{q} < c < \frac{p}{q}.$$

□

Let c be an irrational number in $(0, 1)$, let ε be the constant from Theorem 4, and p, q the integers given by Lemma 7 for ε and c . We take an instance ψ of 3-SAT and construct the graph $G = G_{p,q}(T(\psi))$ in the same way as in the previous part of the proof. Let us again denote the number of clauses of $T(\psi)$ by k . The number of occurrences of literals is $l = 3k$ and n stands for the number of vertices of G .

If ψ is satisfiable, we have again by Corollary 1 that $G[L]$ contains a k -clique, and by Lemma 2 the graph G contains a $(pl + p)$ -clique, which is a $\frac{p}{q}n$ -clique. We shall show that if ψ is not satisfiable, then for any set $A \subseteq V_G$ the graph $S(G, A)$ does not contain a clique of size larger than $(1 - \frac{\varepsilon}{4p})\frac{p}{q}n$. We limit ourselves to instances ψ such that $(1 - \varepsilon)k > 1$ and $pl + p - \varepsilon k \geq 2$, which we can do without loss of generality. Let us first show the following lemma.

Lemma 8. *Let ψ be a formula such that $(1 - \varepsilon)k > 1$ and $pl + p - \varepsilon k \geq 2$. If ψ is not satisfiable, then for any set $A \subseteq V_G$ the graph $S(G, A)$ does not contain a clique of size larger than $pl + p - \varepsilon k$.*

Proof. Suppose that for some $A \subseteq V_G$ we have a clique on a vertex set C in $S(G, A)$ and the size of the clique is larger than $pl + p - \varepsilon k$. Then (similarly as in the proof of Lemma 2) we get that the set C does not contain more than two vertices of Z , because they are pairwise non-adjacent in G and in $S(G, A)$ they induce a bipartite graph.

Since $|C|$ is more than two, C contains some vertices of L or K . But all vertices of Z are non-adjacent in G and have the same neighborhood in $G[L \cup K]$; surely all vertices in $Z \cap C$ have the same neighborhood in $S(G, A)[C]$ (otherwise C would not be a clique). But then either $(Z \cap C) \subseteq A$ or $(Z \cap C) \cap A = \emptyset$, so switching A does not affect edges inside $S(G, A)[Z \cap C]$, and any two vertices in $S(G, A)[Z \cap C]$ are non-adjacent. Therefore C contains at most one vertex of Z .

The set C contains at least one vertex of L , because

$$1 + |K| = 1 + (pl + p - k) < (1 - \varepsilon)k + pl + p - k = pl + p - \varepsilon k.$$

But then C cannot contain both vertices of K and Z , because in G they have the same neighborhood in L and there is no edge between K and Z .

By Lemma 1, every clique in L corresponds to $|C \cap L|$ clauses which are simultaneously satisfiable. Hence by Theorem 4, the maximum clique size in L is $(1 - \varepsilon)k$, which is not enough for C , since

$$(1 - \varepsilon)k < (1 - \varepsilon)k + pl + p - k = pl + p - \varepsilon k,$$

hence C cannot consist only of vertices of L . Therefore C consists of one of the following:

- more than $pl + p - \varepsilon k - 1$ (which is larger than $(1 - \varepsilon)k$) vertices of L , and one vertex of Z
- more than $(1 - \varepsilon)k$ vertices of L , and $pl + p - k$ vertices of K .

In both cases, C contains more than $(1 - \varepsilon)k$ vertices of L , and a vertex v from K or Z . Since C induces a clique in $S(G, A)$, the vertex v is adjacent to all other vertices in C . But in G , by definition, v is adjacent to all vertices of L , too. So switching A cannot have changed any edge connecting v and the other vertices, which means that either all the vertices are in A or none of them is. But then they induce a clique of size larger than $(1 - \varepsilon)k$ in $G[L]$ as well. As we have already shown, the maximum clique size in $G[L]$ is $(1 - \varepsilon)k$, which is a contradiction. \square

By Lemma 8, if ψ is not satisfiable, then the maximum clique size in $S(G, A)$ for any A is $pl + p - \varepsilon k$. But

$$\frac{\varepsilon k}{n} = \frac{\varepsilon k}{q(l+1)} = \frac{\varepsilon k}{q(3k+1)} \geq \frac{\varepsilon}{4q} = \frac{\varepsilon}{4p} \cdot \frac{p}{q},$$

so the maximum clique size divided by n is

$$\frac{pl + p - \varepsilon k}{n} = \frac{p(l+1)}{q(l+1)} - \frac{\varepsilon k}{q(l+1)} \leq \frac{p}{q} - \frac{\varepsilon}{4p} \cdot \frac{p}{q} = \left(1 - \frac{\varepsilon}{4p}\right) \frac{p}{q}.$$

We have chosen the numbers p, q so that

$$c \in \left(\left(1 - \frac{\varepsilon}{4p}\right) \frac{p}{q}, \frac{p}{q} \right),$$

hence the maximum clique ratio matches the lower bound of the interval containing c .

To sum it all up, we have shown that

- if ψ is satisfiable, then there exists an $A \subseteq V_G$ such that $S(G, A)$ contains a clique of size $\frac{p}{q}n$, which is at least cn ,

- if ψ is not satisfiable, then for no set $A \subseteq V_G$ the graph $S(G, A)$ contains a clique of size more than $(1 - \frac{\varepsilon}{4p})^{\frac{2}{q}}n$, especially of size at least cn .

Hence ψ is satisfiable if and only if G can be switched to contain a clique of size at least cn . The graph $G_{p,q}(T(\psi))$ with $q(l+1) = \mathcal{O}(l)$ vertices and $\mathcal{O}(l^2)$ edges can be constructed in polynomial time. That concludes the polynomial-time reduction of 3-SAT to SWITCH- cn -CLIQUE for an irrational constant c , and also the proof that the problem is NP-complete. \square

Proof (of Lemma 3). Having a graph G with edge set E , we set $H = C(K_n)$ (the parity-check matrix of $\mathcal{C}^*(G)$), $r = H\chi_E$, and $w = k$. Then a vector e is a solution of MAXIMUM LIKELIHOOD DECODING if and only if $H(e + \chi_E) = \mathbf{0}$, which means that the vector $e + \chi_E$ is an element of the cut space $\mathcal{C}^*(G)$ and its Hamming distance from χ_E is at most k .

Therefore, by Observation 2 there exists a switch of $S(G, A)$ whose characteristic vector is e . Since the Hamming weight of e is at most k , the switch $S(G, A)$ has at most k edges and we are done.