# Cluster Graph Modification Problems [1]

Ron Shamir [a] Roded Sharan [b,2] Dekel Tsur [a]

[a] *School of Computer Science, Tel-Aviv University, Tel-Aviv 69978, Israel. Email: {rshamir,dekelts}@tau.ac.il.*

[b] *International Computer Science Institute, 1947 Center St., Suite 600, Berkeley CA 94704, USA. Email: roded@icsi.berkeley.edu.*

**Abstract**

In a clustering problem one has to partition a set of elements into homogeneous and well-separated subsets. From a graph theoretic point of view, a cluster graph is a vertex-disjoint union of cliques. The clustering problem is the task of making the fewest changes to the edge set of an input graph so that it becomes a cluster graph. We study the complexity of three variants of the problem. In the Cluster Completion variant edges can only be added. In Cluster Deletion, edges can only be deleted. In Cluster Editing, both edge additions and edge deletions are allowed. We also study these variants when the desired solution must contain a prespecified number of clusters.

We show that Cluster Editing is NP-complete, Cluster Deletion is NP-hard to approximate to within some constant factor, and Cluster Completion is polynomial. When the desired solution must contain exactly $p$ clusters, we show that Cluster Editing is NP-complete for every $p \geq 2$; Cluster Deletion is polynomial for $p = 2$ but NP-complete for $p > 2$; and Cluster Completion is polynomial for any $p$. We also give a constant factor approximation algorithm for a variant of Cluster Editing when $p = 2$.

*Key words:* Clustering, cluster graph, graph modification problem, complexity, approximation.

[1] A preliminary version of this paper appeared in the Proceedings of the 27th International Workshop Graph-Theoretic Concepts in Computer Science [1].
[2] To whom correspondence should be addressed.

## 1 Introduction

**Problem Definition and Motivation:** Clustering is a central optimization problem with applications in numerous fields including computational biology (cf. [2]), image processing (cf. [3]), VLSI design (cf. [4]), and many more. The input to the problem is typically a set of elements and pairwise similarity values between elements. The goal is to partition the elements into subsets, which are called *clusters*, so that two meta-criteria are satisfied: *Homogeneity* — elements inside a cluster are highly similar to each other; and *separation* — elements from different clusters have low similarity to each other. Concrete realizations of these criteria generate a variety of combinatorial optimization problems [5].

In the basic graph theoretic approach to clustering, one builds from the raw data a *similarity graph* whose vertices correspond to elements and there is an edge between two vertices if and only if the similarity of their corresponding elements exceeds a predefined threshold [5, 6]. Ideally, the resulting graph would be a *cluster graph*, that is, a graph composed of vertex-disjoint cliques. In practice, it is only close to being such, since similarity data is experimental and, therefore, error-prone.

Following [7] we formalize the resulting problem as the task of changing (adding or deleting) the fewest edges of an input graph so as to obtain a cluster graph. We call this problem *Cluster Editing*. In the related *Cluster Deletion* (respectively, *Cluster Completion*) problem one has to remove (respectively, add) fewest edges from (respectively, to) an input graph so that it becomes a cluster graph. Completion (respectively, deletion) problems arise when the data contains only false negative (respectively, positive) errors. The above problems belong to the class of *edge modification problems* (cf. [8]), in which one has to minimally change the edge set of a graph so as to satisfy a certain property. Another variant of these problems arises when the solution is also required to consist of a prespecified number of clusters. This variant is motivated by many real-life applications in which a partition of elements into a known number of categories is desired (see, e.g., [9, 10]).

**Previous Results:** Edge modification problems were studied extensively in [8], where earlier studies are also reviewed. Most of these problems were shown to be NP-complete. Polynomial algorithms were given for bounded degree input graphs. In particular, a constant factor approximation algorithm was given for editing and deletion problems with respect to any property that can be characterized by a finite set of forbidden induced subgraphs. Since a graph is a cluster graph if and only if it is $P_2$-free (i.e., it does not contain an induced path of two edges), this result implies a $3d$-approximation algorithm for Cluster Editing and Cluster Deletion on input graphs with degree bounded by $d$.

The Cluster Editing problem was first studied by Ben-Dor et al. [7], who presented a polynomial algorithm that solves the problem with high probability under a stochastic data model. The complexity of the problem was left open. Cluster Deletion was shown to be NP-complete by Natanzon [11].

**Contribution of this Paper:** We prove that Cluster Editing is NP-complete, Cluster Deletion is NP-hard to approximate to within some constant factor, and Cluster Completion is polynomial. We also study the $p$-Cluster versions of these problems, in which the required graph must also be a vertex-disjoint union of $p$ cliques. We show that $p$-Cluster Editing is NP-complete for every $p \geq 2$; $p$-Cluster Deletion is polynomial for $p = 2$ but NP-complete for $p > 2$; and $p$-Cluster Completion is polynomial for any $p$. We also give a 0.878-approximation algorithm for a weighted variant of 2-Cluster Editing.

**Organization of the Paper:** Section 2 contains terminology and problem definitions. In Section 3 we prove the NP-completeness of the Cluster Editing variants, and provide a 0.878-approximation algorithm for a weighted variant of 2-Cluster Editing. In Section 4 we give polynomial algorithms for the Cluster Completion variants. Finally, in Section 5 we study the complexity of the Cluster Deletion variants.

## 2 Preliminaries

All graphs in this paper are simple, i.e., contain no parallel edges or self-loops. Let $G = (V, E)$ be a graph. We denote its set of edges by $E(G)$. For a set $S \subseteq V$, we denote by $G_S$ the subgraph of $G$ induced by the vertices in $S$. For two disjoint subsets $A, B \subseteq V$, we denote by $E_{A,B}$ the set of all edges with one endpoint in $A$ and the other in $B$. The *complement graph* of $G$ is $\overline{G} = (V, \{(u, v) \in (V \times V) \setminus E : u \neq v\})$. See [12] for more definitions of graphs and hypergraphs.

A graph $G = (V, E)$ is called a *cluster graph* if every connected component of $G$ is a complete graph. $G$ is called a *$p$-cluster graph* if it is a cluster graph with $p$ connected components or, equivalently, if it is a vertex-disjoint union of $p$ cliques. If $G$ is any graph and $F \subset V \times V$ is such that $G' = (V, E \triangle F)$ is a cluster graph, then $F$ is called a *cluster editing set* for $G$ ($E \triangle F$ denotes the symmetric difference between $E$ and $F$, i.e., $(E \setminus F) \cup (F \setminus E)$). If in addition $F \subseteq E$, then $F$ is called a *cluster deletion set* for $G$. If $F \cap E = \phi$ then $F$ is called a *cluster completion set* for $G$. For a constant $p$, *$p$-cluster editing set*, *$p$-cluster deletion set*, and *$p$-cluster completion set* are similarly defined. We denote by $P(F)$ the partition of $V$ into disjoint subsets of vertices according to the connected components (cliques) of $G'$. For a partition $P = (V_1, \ldots, V_l)$ of $V$, we denote by $F(P)$ the cluster editing set implied by $P$, that is,

$$F(P) = \bigcup_{i=1}^{l} \{(u, v) \notin E : u, v \in V_i\} \cup \{(u, v) \in E : u \in V_i, v \in V_j, i \neq j\} .$$

The problems we study in this paper are of two types:

**Problem 1 (Cluster Editing/Completion/Deletion)** *Given a graph $G$ and an integer $k$, determine if $G$ has a cluster editing/completion/deletion set of size at most $k$.*

**Problem 2 ($p$-Cluster Editing/Completion/Deletion)** *Given a graph $G$ and an integer $k$, determine if $G$ has a $p$-cluster editing/completion/deletion set of size at most $k$.*

## 3 Cluster Editing

We prove in this section that Cluster Editing is NP-complete by reduction from a restriction of exact cover by 3-sets:

**Problem 3 (3-Exact 3-Cover (3X3C))** *Given a collection $C$ of triplets of elements from a set $U = \{u_1, \ldots, u_{3n}\}$, such that each element of $U$ is a member of at most 3 triplets, determine if there is a sub-collection $I \subseteq C$ of size $n$ which covers $U$.*

The 3X3C problem is known to be NP-complete [13, Problem SP2].

**Theorem 4** *Cluster Editing is NP-complete.*

**PROOF.** Membership in NP is trivial. We prove NP-hardness by reduction from 3X3C. Let $m \equiv 30n$. Given an instance $\langle C, U \rangle$ of 3X3C we build a graph $G = (V, E)$ as follows:

$$\begin{aligned}
V &= \bigcup_{S \in C} \{v_{S,1}, \ldots, v_{S,m}\} \cup U \ , \\
E &= E_1 \cup E_2 \cup E_3 \ , \\
E_1 &= \{(v_{S,i}, u) : S \in C, 1 \leq i \leq m, u \in S\} \ ,
\end{aligned}$$

$$E_2 = \{(v_{S,i}, v_{S,j}) : S \in C, 1 \le i < j \le m\} ,$$
$$E_3 = \{(u, u') : \exists S \in C \text{ s.t. } u, u' \in S\} .$$

In words, we build a clique of size $m + 3$ around each triplet $S$ by fully connecting $S$ and $m$ additional vertices. For each triplet $S \in C$ we denote $V_S = \{v_{S,1}, \ldots, v_{S,m}\}$ and call the elements of $V_S$, *$S$-vertices*. Let $q = \sum_{S \in C} |S| = 3|C|$. Define $N \equiv m(q - 3n)$ and $M \equiv |E_3| - 3n$. We prove that there is an exact cover of $U$ if and only if there is a cluster editing set for $G$ of size at most $N + M$:

($\Rightarrow$) Suppose that $I \subseteq C$ is an exact cover of $U$. Let $F_1 = \{(v_{S,i}, u) : S \notin I, 1 \le i \le m, u \in S\}$ and let $F_2 = \{(u, u') \in E_3 : \nexists S \in I \text{ s.t. } u, u' \in S\}$. It is easy to verify that $F = F_1 \cup F_2$ is a cluster editing set for $G$, whose size is $|F| = |F_1| + |F_2| = N + M$.

($\Leftarrow$) Suppose that $G$ has an editing set of size at most $N + M$. Let $F$ be an editing set of $G$ of minimum size. Clearly, $|F| \le N + M$. We shall prove that one can derive from $F$ an exact cover of $U$. Since each element of $U$ occurs in at most 3 triplets, $q \le 9n$. Thus, $|E_3| \le q \le 9n$ and $|F| \le N + M \le 6mn + 6n = 180n^2 + 6n < \frac{m}{2}(\frac{m}{2} - 2)$.

Let $G' = (V, E \triangle F)$ be the cluster graph obtained by editing $G$ according to $F$ and let $P(F)$ be the partition of $V$ according to the cliques of $G'$. We shall prove that for every subset $S \in C$ there is a unique clique in $G'$ which contains $V_S$. To this end, we first show that there is a clique $K_S$ in $G'$ such that $|K_S \cap V_S| \ge m/2 + 3$: Suppose that the vertices of $V_S$ are partitioned among $k$ cliques $X_1, \ldots, X_k$ in $G'$. Let $s(X_i) = |V_S \cap X_i|, i = 1, \ldots, k$. Suppose to the contrary that $s(X_i) \le m/2 + 2$ for all $i$. Therefore,

$$|F| \ge \frac{1}{2} \sum_{i=1}^{k} s(X_i)(m - s(X_i)) \ge \frac{1}{2} \sum_{i=1}^{k} s(X_i)(\frac{m}{2} - 2) = \frac{m}{2}(\frac{m}{2} - 2) .$$

A contradiction follows.

Let $K_S$ be the clique $X_i$ for which $s(X_i)$ is maximum ($|K_S \cap V_S| \ge m/2 + 3$). We next prove that $V_S \subseteq K_S \subseteq V_S \cup S$. Let $x = |K_S \setminus (V_S \cup S)|$. Consider a new partition $P'$ of $V$, which is obtained from $P(F)$ by splitting $K_S$ into $K_S \cap (V_S \cup S)$ and $K_S \setminus (V_S \cup S)$. Clearly, $|F| - |F(P')| \ge (m/2 + 3)x - 3x = xm/2$. Since $F$ is an optimum cluster editing set, we conclude that $x = 0$ and $K_S \subseteq V_S \cup S$. To see that $K_S \supseteq V_S$, suppose to the contrary that there is some index $1 \le i \le m$ such that $v_{S,i} \notin K_S$. Let $K'$ be the clique in $G'$ which contains $v_{S,i}$. Let $P''$ be a new partition of $V$, which is obtained from $P(F)$ by moving $v_{S,i}$ from $K'$ to $K_S$. Then $|F| - |F(P'')| \ge m/2 + 3 - (m/2 - 4 + 3) = 4$, a contradiction. We conclude that for every $S \in C$ there is a unique clique in $G'$ which contains $V_S$ and is contained in $V_S \cup S$.

Let $F_1 = F \cap E_1$. Examine an element $u \in U$ which is a member of (at least) two subsets $S_1, S_2 \in C$. By the previous claim, $V_{S_1}$ and $V_{S_2}$ are subsets of distinct cliques in $G'$. Hence, either $E_{V_{S_1},\{u\}} \subseteq F$, or $E_{V_{S_2},\{u\}} \subseteq F$ (or both). Therefore, $|F_1| \geq N$. Moreover, since $|F_1| \leq N + M$ and $M \leq 6n$, each vertex $u \in U$ must be adjacent in $G'$ to the $S$-vertices of exactly one set $S$ where $u \in S$. Call this set the *$S$-set* of $u$.

Let $F_2 = F \setminus F_1$. For every two vertices $u, u' \in U$ such that $(u, u') \in E$, and the $S$-sets of $u$ and $u'$ differ, we must have $(u, u') \in F_2$. Since each subset in $C$ contains 3 elements, $G'_U$ is a union of cliques of size at most 3. It is easy to verify that the maximum number of edges in such a $3n$-vertex graph is $3n$, and that number is obtained if and only if $G'_U$ is a union of triangles only. Therefore, $|F_2| = |E_3| - |E(G'_U)| \geq M$ with equality if and only if there is a partition of $U$ into triplets of elements, such that the elements of each triplet have the same $S$-set. Since $|F| \leq N + M$, we must have $|F| = N + M$ and the implied partition into triplets induces an exact cover of $U$. $\square$

We note, that the same construction can be used to show that Cluster Deletion is NP-complete.

### *3.1 p-Cluster Editing*

In this section we study the $p$-Cluster Editing problem. We first show that 2-Cluster Editing is NP-complete. We then conclude that $p$-Cluster Editing is NP-complete for every $p \geq 2$.

To prove the hardness of 2-Cluster Editing, we define the following problem:

**Problem 5 (Balanced 2-Coloring of a 3-Uniform Hypergraph)** *Given a 3-uniform hypergraph $G$, determine if there is a 2-coloring of $G$ such that the number of vertices that are colored by each color is the same.*

This problem can be shown to be NP-complete by a trivial reduction from 2-Coloring of a 3-Uniform Hypergraph, whose NP-completeness was proven by Lovasz [14].

**Theorem 6** *2-Cluster Editing is NP-complete.*

**PROOF.** Membership in NP is trivial. We reduce from Balanced 2-Coloring of a 3-Uniform Hypergraph. Given a hypergraph $G = (V, E)$, we build an instance of 2-Cluster Editing $\langle G' = (V', E'), k \rangle$ as follows: Let $n$ and $m$ be the number of vertices and hyperedges, respectively, in $G$, and assume that

$V = \{1, \dots, n\}$. Let $M \equiv 2n^3$. Each vertex $i$ of $G$ is associated with a set of $M$ vertices $V_i = \{v_{i,j} : j = 1, \dots, M\}$ in $G'$, which we call a *cluster*. We define $V' = \cup_{i=1}^{n} V_i$. For a triplet of indices $1 \le i < j < l \le n$ define the set $E_{i,j,l} = \{(v_{i,r}, v_{j,r}), (v_{j,r+1}, v_{l,r}), (v_{l,r+1}, v_{i,r+1})\}$, where $r = 2(n^2 i + nj + l) - 1$. The edge set of $G'$ is defined as

$$E' = \bigcup_{i<j<l,(i,j,l)\notin E} E_{i,j,l} \cup \bigcup_{i=1}^{n} \{(v_{i,j}, v_{i,k}) : j \neq k\}.$$

In words, we build a clique around each $V_i$, and add the edges of $E_{i,j,l}$ for every non-hyperedge of $G$. Finally, we set $k \equiv 2\binom{n/2}{2}(M^2 - (n-2)) + (\frac{n}{2})^2(n-2) - m$. For convenience we also define a graph $G'' = (V', E'')$, which is built like $G'$ except that it contains the edges in $E_{i,j,l}$ for every triplet $i < j < l$, that is,

$$E'' = E' \cup \bigcup_{i<j<l,(i,j,l)\in E} E_{i,j,l} \ .$$

We now prove that there is a balanced 2-coloring of $G$ if and only if there is a 2-cluster editing set of $G'$ of size at most $k$.

($\Rightarrow$) Suppose that $f: V \to \{0, 1\}$ is a balanced 2-coloring of $G$. Let $S = \cup_{i:f(i)=0} V_i$, and let $F', F''$ be the 2-cluster editing sets of $G'$ and $G''$, respectively, that correspond to the partition $P = (S, V \setminus S)$. Since $f$ is balanced, each side of $P$ consists of $\frac{n}{2}$ clusters. We first compute the size of $F''$. For two distinct clusters $V_i$ and $V_j$, $i < j$, each set of the form $E_{i,j,l}$, $E_{i,l,j}$, or $E_{l,i,j}$ contains exactly one edge between $V_i$ and $V_j$. Therefore, there are exactly $n - 2$ edges between every pair of clusters in $G''$. It follows that $F''$ contains $2\binom{n/2}{2}(M^2 - (n-2))$ edges that are not in $E''$ between clusters on the same side of the partition, and $(\frac{n}{2})^2(n-2)$ edges in $E''$ between clusters on different sides of the partition. Thus, $|F''| = 2\binom{n/2}{2}(M^2 - (n-2)) + (\frac{n}{2})^2(n-2)$. We now compute the size of $F'$: For each hyperedge $(i, j, l) \in E$, the edges of $E_{i,j,l}$ in $G''$ contribute two edges to $F''$ (as the clusters $V_i$, $V_j$, and $V_l$ are not all on the same side of the partition), while the non-existence of the edges of $E_{i,j,l}$ in $G'$ contributes only one edge to $F'$ (between the two clusters on the same side of the partition). It follows that $|F'| = |F''| - m = k$.

($\Leftarrow$) Suppose that $G'$ has a 2-cluster editing set of size at most $k$. Let $F$ be a 2-cluster editing set for $G'$ of minimum size. Clearly, $|F| \le k$. We shall prove that one can construct from $F$ a balanced 2-coloring of $G$.

Let $P(F)$ be the partition $(S, V' \setminus S)$. We say that $P(F)$ *splits* a cluster $V_i$ if $V_i \cap S \neq \phi$ and $V_i \nsubseteq S$. We first claim that $P(F)$ splits no cluster. Suppose to the contrary that $P(F)$ splits at least one cluster. If $P(F)$ splits more than one cluster then let $V_i$ be a split cluster whose intersection with $S$ has minimum cardinality, and let $V_j$ be a split cluster whose intersection with $S$ has maximum cardinality and $j \neq i$. Denote $a = |V_i \cap S|$ and $b = |V_j \cap S|$. Choose some vertex $u \in V_i \cap S$ and a vertex $w \in V_j \setminus S$. Let $S' = S \cup \{w\} \setminus \{u\}$, and let $F'$ be the 2-cluster editing set that corresponds to the partition $(S', V' \setminus S')$. We will show that $|F| - |F'| \geq 0$. Note that if $\{i, j, l\} \neq \{i', j', l'\}$ then the edges of $E_{i,j,l}$ are incident on different vertices than the edges of $E_{i',j',l'}$. Therefore, every $v \in V_i$ has at most one neighbor outside of $V_i$. If such a neighbor exists, denote it by $n_v$.

The edges in $F$ that are incident on $u$ or $w$ are

(1) $M - a$ edges (in $E'$) between $u$ and $V_i \setminus S$.
(2) A possible edge (in $E'$) between $u$ and $n_u$ (if $n_u$ exists and $n_u \in V' \setminus S$).
(3) Either $|S| - a$ or $|S| - a - 1$ edges (not in $E'$) between $u$ and $S \setminus (V_i \cap S)$ (the second term is for the case that $n_u$ exists and $n_u \in S$).
(4) $b$ edges (in $E'$) between $w$ and $V_j \cap S$.
(5) A possible edge (in $E'$) between $w$ and $n_w$ (if $n_w$ exists and $n_w \in S$).
(6) Either $nM - |S| - (M - b)$ or $nM - |S| - (M - b) - 1$ edges (not in $E'$) between $w$ and $V' \setminus S \setminus (V_j \setminus S)$ (the second term is for the case when $n_w$ exists and $n_w \in V' \setminus S$).

The total number of these edges is at least $nM - 2a + 2b - 2$.

Similarly, the edges in $F'$ that are incident on $u$ or $w$ are

(1) $a - 1$ edges (in $E'$) between $u$ and $V_i \cap S \setminus \{u\}$.
(2) A possible edge (in $E'$) between $u$ and $n_u$.
(3) Either $nM - |S| - (M - a) - 1$ or $nM - |S| - (M - a) - 2$ edges (not in $E'$) between $u$ and $V' \setminus S \setminus (V_i \setminus S) \setminus \{w\}$.
(4) $M - b - 1$ edges (in $E'$) between $w$ and $V_j \setminus S \setminus \{w\}$.
(5) A possible edge (in $E'$) between $w$ and $n_w$.
(6) Either $|S| - b - 1$ or $|S| - b - 2$ edges (not in $E'$) between $w$ and $S \setminus (V_j \cap S) \setminus \{u\}$.

The total number of these edges is at most $nM + 2a - 2b - 2$. It follows that

$$|F| - |F'| \geq (nM - 2a + 2b - 2) - (nM + 2a - 2b - 2) = 4(b - a) \geq 0.$$

If $a < b$, we have that $|F'| < |F|$, in contradiction to the minimality of $F$. If $a = b$, we have that $|F'| = |F|$. In this case we build a set $S''$ from $S'$ using the same process as above, and since $|V_l \cap S'|$ is not equal amongst the clusters, it follows that the 2-cluster editing set $F''$ that corresponds to the partition $(S'', V \setminus S'')$ satisfies $|F''| < |F'| = |F|$, and again we arrive at a contradiction.

Now suppose that the partition $P(F)$ splits exactly one cluster, and denote this cluster by $V_i$. Let $a = |V_i \cap S|$. Out of the remaining $n-1$ clusters, suppose that $r$ clusters are contained in $S$, and $n-r-1$ clusters are contained in $V' \setminus S$. W.l.o.g. suppose that $n-r-1 \le r$, and since $n$ is even we have $n-r-1 \le r-1$. Define $S' = S \setminus V_i$, and let $F'$ be the corresponding 2-cluster editing set. For each $v \in V_i \cap S$, there are at least $rM-1$ edges in $F$ between $v$ and $S \setminus V_i$ (the term $-1$ is due to the possibility that $n_v$ exists and $n_v \in S \setminus V_i$) and $M-a$ edges between $v$ and $V_i \setminus S$. Hence, the number of edges in $F$ that are incident on $v$ is at least $rM - 1 + M - a$. On the other hand, an edge in $F'$ that is incident on $v$ is either between $v$ and $n_v$, or between $v$ and $(V' \setminus S) \setminus V_i$. The number of edges of the latter type is $(n-1-r)M$, so the number of edges in $F$ that are incident on $v$ is at most $(n-1-r)M + 1 \le (r-1)M + 1$. It follows that

$$|F| - |F'| \ge a\,(rM - 1 + M - a - ((r-1)M + 1)) = a\,(2M - a - 2) > 0,$$

in contradiction to the minimality of $F$. Therefore, $F$ splits no cluster.

We now claim that $S$ contains exactly $r = \frac{n}{2}$ clusters. Conversely, suppose w.l.o.g. that $r > n/2$. Let $V_i$ be some cluster contained in $S$. Let $S' = S \setminus V_i$ and let $F'$ be the corresponding 2-cluster editing set. Similarly to the above, we have that

$$|F| - |F'| \ge M((r-1)M - 1 - ((n-r)M + 1)) \ge M(M-2) > 0,$$

a contradiction. Hence, $S$ contains $\frac{n}{2}$ clusters.

Define a coloring $f\colon V \to \{0,1\}$ by $f(i) = 0$ if and only if $V_i \subseteq S$. Clearly, $f$ is balanced. It remains to show that $f$ is a legal 2-coloring. For a hyperedge $(i,j,k) \in E$, if $i,j,k$ have the same color then $|F \cap E_{i,j,l}| = 3$. Otherwise, $|F \cap E_{i,j,l}| = 1$ since two of the edges in $E_{i,j,l}$ must cross the partition $(S, V' \setminus S)$. Hence, each monochromatic hyperedge increases $|F|$ by 2. By the first direction of the proof, the editing set that corresponds to a legal 2-coloring is of size exactly $k$. Thus, no monochromatic hyperedge is possible in $f$. It follows that $f$ is a balanced 2-coloring of $G$. $\square$

**Corollary 7** *$p$-Cluster Editing is NP-complete for any $p \ge 2$.*

**PROOF.** Fix $p > 2$. We provide a reduction from 2-Cluster Editing. Given an input instance $\langle G = (V,E), k \rangle$ of 2-Cluster Editing, $|V| = n$, we form an instance $\langle G' = (V', E'), k \rangle$ of $p$-Cluster Editing as follows: Define $V' = V \cup \bigcup_{i=1}^{p-2} V_i$, where $V_i = \{w_{i,j} : j = 1, \ldots, n^2\}$. Define $E' = E \cup \bigcup_{i=1}^{p-2} \{(w_{i,j}, w_{i,k}) : k \neq j\}$. That is, $p-2$ disjoint cliques of size $n^2$ each are added to $G$.

Clearly, every 2-cluster editing set of $G$ is a $p$-cluster editing set of $G'$ (of the same size). Conversely, suppose that $F'$ is a $p$-cluster editing set of $G'$ of size at most $k$, and let $P(F') = (S_1, \ldots, S_p)$ be the corresponding partition. We show that $F'$ is also a 2-cluster editing set for $G$.

If there is a set $V_i$ such that $V_i \cap S_j \neq \phi$ and $V_i \not\subseteq S_j$ for some $j$, then $F'$ contains $E_{V_i \cap S_j, V_i \setminus S_j}$. The number of such edges is at least $n^2 - 1 > k$, a contradiction. Therefore, every $V_i$ is contained in some set $S_j$. Furthermore, every $S_j$ contains at most one set $V_i$ since, otherwise, we have $|F'| \geq n^4 > k$, a contradiction. If $S_j \supseteq V_i$ then $S_j = V_i$ using a similar argument. It follows that all edges in $F'$ are incident on vertices of $V$, implying that $F'$ is a 2-cluster editing set of $G$. □

### *3.2 A 0.878-Approximation Algorithm*

In this section we give a polynomial approximation algorithm for a weighted variant of 2-Cluster Editing which is defined as follows:

**Problem 8 (Weighted 2-Cluster Editing)** *Given a graph $G$ and a weight function on vertex pairs $w\colon E(G) \cup E(\overline{G}) \to \mathcal{N}$, find in $G$ a 2-cluster editing set with maximum total weight of unedited vertex pairs.*

Note, that the decision version of Weighted 2-Cluster Editing reduces to that of 2-Cluster Editing when $w \equiv 1$ (i.e., $w(e) = 1$ for every $e \in E(G) \cup E(\overline{G})$).

Let $G = (V, E, w)$ be an input weighted graph with $n$ vertices. Let $S_n$ denote the $n$-dimensional unit sphere. We define the following semi-definite relaxation of Weighted 2-Cluster Editing:

$$\begin{aligned} &\max \frac{1}{2}[\sum_{(i,j)\in E} (w((i,j))(1 + v_i \cdot v_j)) + \sum_{(i,j)\notin E} (w((i,j))(1 - v_i \cdot v_j))] \\ &\text{s.t. } v_i \in S_n \ \ \forall i \end{aligned}$$

We claim that this is indeed a relaxation of Weighted 2-Cluster Editing, that is, for every partition $P = (A, B)$ of $G$ there exist vectors $v_1, \ldots, v_n \in S_n$ such that the total weight of unedited vertex pairs as implied by $P$ is $\frac{1}{2}[\sum_{(i,j)\in E}(w((i,j))(1 + v_i \cdot v_j)) + \sum_{(i,j)\notin E}(w((i,j))(1 - v_i \cdot v_j))]$. Indeed, let $(A, B)$ be a partition of $G$. Let $v_0$ be any unit vector in $S_n$. For every $i \in A$ set $v_i = v_0$, and for every $i \in B$ set $v_i = -v_0$. The claim follows.

Our approximation algorithm solves this semi-definite relaxation and then rounds the solution obtained using the random hyperplane technique [15].

**Theorem 9** *The algorithm approximates Weighted 2-Cluster Editing with an expected approximation ratio of at least* 0.878.

$S_0 = \{(0, \ldots, 0)\}$
**For** $i = 1$ to $t$ **do:**
    $S_i = S_{i-1} \cup \{v + C_i e_j : v \in S_{i-1}, j = 1, \ldots, p-1\}$.
Pick in $S_t$ a vector $v^*$ minimizing $\sum_{i=1}^{p-1} v_i^2 + (n - \sum_{i=1}^{p-1} v_i)^2$.
**Output** the completion set that corresponds to $v^*$.

Fig. 1. An algorithm for $p$-Cluster Completion. $e_j$ denotes a $(p-1)$-dimensional unit vector with 1 in position $j$.

**PROOF.** Follows directly from [15, Theorem 6.1]. $\square$

## 4 Cluster Completion

The cluster completion problem is trivially polynomial: The optimum solution is obtained by simply transforming each connected component of the input graph into a complete graph. In this section we give a polynomial algorithm for $p$-Cluster Completion, for any fixed $p \geq 2$.

Let $G = (V, E)$ be an input graph with $n$ vertices and $t$ connected components. If $t < p$ we output *False*. We assume henceforth that $t \geq p$. To find the optimum completion set we compute partitions of the $t$ components of $G$ into $p$ sets (splitting no connected components) and choose the partition which results in a minimum completion set. Using dynamic programming, we only need to consider a polynomial number of partitions. Note that since we only add edges, we seek to minimize the sum of the number of edges in each of the $p$ sets of the partition, or equivalently, the sum of the squared sizes of the sets.

Let $C_1, \ldots, C_t$ be the cardinalities of the connected components in $G$. Our algorithm will denote each possible partition by a $(p-1)$-long vector of integers, which describes the sizes of the sets in the partition (the size of the last set is the difference from $n$). We will maintain a set $S_i$ of the vectors that correspond to all possible partitions of the first $i$ connected components. The algorithm is given in Figure 1. The actual partition can be obtained by maintaining for each $v \in S_i$ a pointer to its parent vector in $S_{i-1}$.

**Theorem 10** *The algorithm correctly solves $p$-Cluster Completion in $O(tn^{p-1})$ time.*

**PROOF.** Let $F$ be the $p$-completion set returned by the algorithm. It suffices to prove that $F$ is optimum. Let $P(F) = (V_1, \ldots, V_p)$. Then

$$|F| = \sum_{i=1}^{p} \binom{|V_i|}{2} - |E| = \frac{1}{2}\sum_{i=1}^{p}(|V_i|^2 - |V_i|) - |E| = \frac{1}{2}\sum_{i=1}^{p}|V_i|^2 - \frac{n}{2} - |E|.$$

Let $F^*$ be an optimum $p$-cluster completion set of $G$, and let $P(F^*) = (V_1^*, \ldots, V_p^*)$. Then $|F^*| = \frac{1}{2}\sum_{i=1}^{p}|V_i^*|^2 - \frac{n}{2} - |E|$. By the algorithm, $|F| \leq |F^*|$, implying that $F$ is optimum. $\square$

## 5 Cluster Deletion

In this section we study the cluster deletion problem. We shall give a gap preserving reduction (cf. [16]) from a restricted version of SET-COVER to Cluster Deletion. This reduction implies that there is some constant $\epsilon > 0$ such that it is NP-hard to approximate Cluster Deletion to within a factor of $1 + \epsilon$. We begin by introducing the SET-COVER restriction.

**Problem 11 (Minimum Restricted Exact Cover (REC))** *The input is a set $U = \{u_1, \ldots, u_t\}$, and a collection $C$ of subsets of $U$ which satisfies the following conditions:*

- *There is a constant $k_1 > 0$ such that for each $S \in C$, $|S| \leq k_1$.*
- *There is a constant $k_2 > 0$ such that for all $u \in U$, $|\{S \in C : u \in S\}| \leq k_2$.*
- *If $S \in C$ and $S' \subset S$ then $S' \in C$.*
- *$\bigcup_{S \in C} S = U$.*

*The goal is to find a sub-collection $I \subseteq C$ of minimum cardinality, such that $\bigcup_{S \in I} S = U$, and the sets in $I$ are pairwise-disjoint.*

Note, that the third and fourth conditions guarantee that a solution to REC always exists. REC can be shown to be MAX-SNP complete by a simple L-reduction from a restriction of SET-COVER in which the size of every set is bounded and each element occurs in a bounded number of sets. The latter problem is known to be MAX-SNP complete [17]. Hence, there is a constant $\delta_{REC} > 0$ such that it is NP-hard to approximate REC to within a factor of $1 + \delta_{REC}$.

**Theorem 12** *There is some constant $\epsilon > 0$ such that it is NP-hard to approximate Cluster Deletion to within a factor of $1 + \epsilon$.*

**PROOF.** By a gap preserving reduction from REC (similar to the one in Theorem 4). For an instance $I_{REC}$ of REC, the reduction produces in polynomial time an instance $I_{CD}$ of Cluster Deletion such that $opt(I_{REC}) \leq c$ implies $opt(I_{CD}) \leq c'$ and $opt(I_{REC}) > (1+\delta_{REC})c$ implies $opt(I_{CD}) > (1+\epsilon)c'$, where $opt(I)$ denotes the optimal value for instance $I$.

We now describe the reduction. Let $I_{REC} = \langle U, C \rangle$, and let $|U| = t$. Suppose that each set in $C$ has size at most $k_1$, and each element occurs in at most $k_2$ sets. Let $m \equiv \frac{k_1^2 k_2}{\delta_{REC}}$ and let $q \equiv \sum_{S \in C} |S|$. We build an instance $I_{CD} = \langle G = (V, E) \rangle$ of Cluster Deletion as follows:

$$
\begin{aligned}
V &= \bigcup_{S \in C} \{v_{S,1}, \ldots, v_{S,m}, w_S\} \cup U \ , \\
E &= E_1 \cup E_2 \cup E_3 \cup E_4 \ , \\
E_1 &= \{(v_{S,i}, u) : S \in C, 1 \leq i \leq m, u \in S\} \ , \\
E_2 &= \{(v_{S,i}, v_{S,j}) : S \in C, 1 \leq i < j \leq m\} \ , \\
E_3 &= \{(u, u') : \exists S \in C \text{ s.t. } u, u' \in S\} \ , \\
E_4 &= \{(v_{S,i}, w_S) : S \in C, 1 \leq i \leq m\} \ .
\end{aligned}
$$

In words, for each $S \in C$ we form a clique on $S$ and a set of $m$ new vertices, and also connect all the new vertices to a single extra vertex $w_S$. For each subset $S \in C$ we denote $V_S = \{v_{S,1}, \ldots, v_{S,m}\}$ and call the elements of $V_S$, $S$-vertices. Note, that $|E_3| \leq (k_1 - 1)k_2 t/2 < k_1 k_2 t/2$ and $q \leq k_2 t$. Clearly, $t/k_1 \leq opt(I_{REC}) \leq t$. Let $c$ be any constant such that $t/k_1 \leq c \leq t$. Define $c' \equiv (q - t + c)m + |E_3|$ and $\epsilon \equiv \frac{\delta_{REC}}{2k_1 k_2 + \delta_{REC}}$. We prove that this reduction is gap preserving:

($\Rightarrow$) Suppose that $opt(I_{REC}) \leq c$. Let $I \subseteq C$ be an exact cover of $U$, $|I| \leq c$. For $u \in U$ denote by $I_u$ the set in $I$ which contains $u$. Let $\bar{I} = C \setminus I$.

To obtain a cluster subgraph $G'$ of $G$ we delete the following edges:

(1) For all $S \in \bar{I}, u \in S$ delete all the edges in $E_{V_S,\{u\}}$.
(2) For all $S \in I$ delete all the edges in $E_{V_S,\{w_S\}}$.
(3) For all $u \in U, u' \in U \setminus I_u$ delete the edge $(u, u')$ if it exists.

One can easily verify that $G'$ is a cluster graph and, therefore, $opt(I_{CD}) \leq (q - t + c)m + |E_3| = c'$.

($\Leftarrow$) Suppose that $opt(I_{REC}) > (1 + \delta_{REC})c$. Observe that in any cluster subgraph of $G$, every $u \in U$ is adjacent to the $S$-vertices of at most one set $S \in C$. Furthermore, there exists an optimum solution $F$ of $I_{CD}$ for which: If a vertex $u \in U$ is adjacent to an $S$-vertex in $(V, E \setminus F)$, for some $S \in C$, then $F$ contains all the edges in $E_{V_S,\{w(S)\}}$ and does not contain any edges in $E_{V_S,\{u\}}$. Indeed, if $F'$ is a cluster deletion set such that $u_1, \ldots, u_r$ $(1 \leq r \leq k_1)$ are adjacent to an $S$-vertex in $(V, E \setminus F')$, then $F'' = (F' \cup E_{V_S,\{w(S)\}}) \setminus (\bigcup_{i=1}^{r} E_{V_S,\{u_i\}} \cup \{v_{S,i}, v_{S,j} :$

$i \neq j\}$) is also such a cluster deletion set, and $|F''| \leq |F'|$.

Examine now the cluster deletion set $F$. For each $u \in U$, either $E_{V \setminus U, \{u\}} \subseteq F$ or there exists a single set $S \in C$ such that $E_{V_S, \{u\}} \not\subseteq F$ and $E_{V_S, \{w(S)\}} \subseteq F$. Let $k$ be the number of vertices $u \in U$ for which the latter case applies, and let $\mathcal{T}$ be the collection of all sets $S$ such that $(v_{S,i}, u) \in E \setminus F$ for some $u \in U, i$. It follows that $|F| \geq (q - k + |\mathcal{T}|)m$. The sets in $\mathcal{T}$ cover $k$ elements of $U$, so $|\mathcal{T}| \geq opt(I_{REC}) - (t - k)$ (since $C$ contains all singleton sets). We conclude that

$$\begin{aligned} opt(I_{CD}) &\geq (q - t + opt(I_{REC}))m > (q - t + (1 + \delta_{REC})c)m = c' + (\delta_{REC} cm - |E_3|) \\ &> c'(1 + \frac{\delta_{REC} cm - |E_3|}{qm + |E_3|}) > c'(1 + \frac{\delta_{REC}(t/k_1)m - k_1 k_2 t/2}{k_2 tm + k_1 k_2 t/2}) \\ &= c'(1 + \frac{2\delta_{REC} m/k_1 - k_1 k_2}{2k_2 m + k_1 k_2}) = c'(1 + \frac{\delta_{REC}}{2k_1 k_2 + \delta_{REC}}) = c'(1 + \epsilon) \ . \end{aligned}$$

□

### 5.1 p-Cluster Deletion

In this section we give a polynomial algorithm for the optimization version of 2-Cluster Deletion. We then show that $p$-Cluster Deletion is NP-complete for every $p > 2$.

Let $G = (V, E)$ be an input graph. W.l.o.g., $G$ is connected as, otherwise, either $G$ is already a 2-cluster graph, or we output *False*. The algorithm is described in Figure 2.

**Theorem 13** *The algorithm correctly solves 2-Cluster Deletion in $O(n + |E(\overline{G})|)$ time.*

**PROOF. Correctness**: Since the complement of a 2-cluster graph is a complete bipartite graph, a solution exists if and only if $\overline{G}$ is bipartite. Hence, the algorithm outputs *False* if and only if no solution exists. Moreover, the

Let $C_1, \ldots, C_t$ be the connected components of $\overline{G}$.
**For** $i = 1, \ldots, t$ **do:**
  **If** $C_i$ is not bipartite **then** output *False* and halt.
  **Else** find a bipartition $(A_i, B_i)$ of $C_i$ such that $|A_i| \geq |B_i|$.
**Output** $F((A_1 \cup \ldots \cup A_t, B_1 \cup \ldots \cup B_t))$.

Fig. 2. An algorithm for 2-Cluster Deletion.

partition produced by the algorithm has the property that if two vertices are assigned to the same set then they are adjacent. Therefore, the set of edges $F = F((A_1 \cup \ldots \cup A_t, B_1 \cup \ldots \cup B_t))$ returned by the algorithm is a 2-deletion set of $G$. It suffices to prove that $F$ is optimum.

Denote $S = A_1 \cup \ldots \cup A_t$. Clearly, $F$ consists of edges in $G$ with one endpoint in $S$ and the other in $V \setminus S$. Therefore, $|F| = |E_{S,V\setminus S}| = |S|(n - |S|) - E(\overline{G})$. Let $F^*$ be a smallest 2-deletion set of $G$, and let $P(F^*) = (S^*, V \setminus S^*)$, where $|S^*| \leq \frac{n}{2}$. It follows that $|F^*| = |S^*|(n - |S^*|) - E(\overline{G})$. For every $i \leq t$, either $A_i \subseteq S^*$ or $B_i \subseteq S^*$ and, therefore, $|S| \leq |S^*| \leq \frac{n}{2}$, implying that $|F| \leq |F^*|$. Hence, $F$ is an optimum 2-deletion set of $G$.

**Complexity**: The bottleneck in the complexity of the algorithm is computing the connected components of $\overline{G}$ and finding a bipartition for each of them. These tasks can be performed in $O(n + |E(\overline{G})|)$ total time. $\square$

**Theorem 14** *$p$-Cluster Deletion is NP-complete for any $p \geq 3$.*

**PROOF.** Membership in NP is trivial. We provide a reduction from $p$-Coloring. Given an input graph $G = (V, E)$, the reduction outputs its complement $\overline{G} = (V, \overline{E})$ and a bound $k = |\overline{E}|$. A $p$-coloring $f$ of $G$ trivially translates into a $p$-deletion set $\{(u, v) \notin E : f(u) \neq f(v)\}$ of $\overline{G}$ of size at most $k$. Conversely, suppose that $F$ is a $p$-deletion set of $\overline{G}$ with $|F| \leq k$, and let $C_1, \ldots, C_p$ be the cliques of $(V, \overline{E} \setminus F)$. The coloring $f$ defined by $f(v) = i$ for all $v \in C_i$ is a $p$-coloring of $G$. $\square$

Note that the reduction works with any $k \geq |\overline{E}|$ and in fact shows that even deciding whether a graph has a $p$-cluster deletion set is NP-hard, for $p \geq 3$.

## 6 Concluding Remarks

After the submission of the paper we discovered that the Cluster Editing problem was studied independently by Chen et al. [18] and Bansal et al. [19]. Chen et al. study Cluster Editing in the context of phylogeny reconstruction, and show that the problem is NP-hard. Bansal et al. show the NP-hardness of the problem and give a constant factor approximation algorithm for it.

## Acknowledgments

R. Shamir was supported in part by the Israel Science Foundation (grants number 565/99 and 309/02). R. Sharan was supported by a Fulbright grant and by an Eshkol fellowship from the Ministry of Science, Israel.

## References

[1] R. Shamir, R. Sharan, D. Tsur, Cluster graph modification problems, in: Proceedings of the 27th International Workshop Graph-Theoretic Concepts in Computer Science (WG), 2002, pp. 379–390.

[2] R. Sharan, A. Maron-Katz, R. Shamir, CLICK and EXPANDER: A system for clustering and visualizing gene expression data, Bioinformatics 19 (2003) 1787–1799. A preliminary version appeared in Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB), pages 307–316, AAAI Press, 2000.

[3] Z. Wu, R. Leahy, An optimal graph theoretic approach to data clustering: theory and its application to image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (11) (1993) 1101–1113.

[4] C. Hagen, A. Kahng, New spectral methods for ratio cut partitioning and clustering, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 11(9) (1992) 1074–1085.

[5] P. Hansen, B. Jaumard, Cluster analysis and mathematical programming, Mathematical Programming 79 (1997) 191–215.

[6] J. Hartigan, Clustering Algorithms, John Wiley and Sons, 1975.

[7] A. Ben-Dor, R. Shamir, Z. Yakhini, Clustering gene expression patterns, Journal of Computational Biology 6 (3/4) (1999) 281–297.

[8] A. Natanzon, R. Shamir, R. Sharan, Complexity classification of some edge modification problems, Discrete Applied Mathematics 113 (2001) 109–128.

[9] T. R. Golub, D. K. Slonim, et al., Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring, Science 286 (1999) 531–537.

[10] A. A. Alizadeh, M. B. Eisen, et al., Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, Nature 403 (6769) (2000) 503–511.

[11] A. Natanzon, Complexity and approximation of some graph modification problems, Master's thesis, Department of Computer Science, Tel Aviv University (1999).

[12] C. Berge, Graphs and Hypergraphs, North-Holland, Amsterdam, 1973.

[13] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Co., San Francisco, 1979.

[14] L. Lovasz, Covering and coloring of hypergraphs, in: Proc. 4th Southeastern Conf. on Combinatorics, Graph Theory, and Computing, Utilitas Mathematica Publishing, 1973.

[15] M. X. Goemans, D. P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, Journal of the ACM 42(6) (1995) 1115–1145.

[16] D. S. Hochbaum (Ed.), Approximation Alogrithms for NP-Hard Problems, PWS Publishing, Boston, 1997.

[17] C. Papadimitriou, M. Yannakakis, Optimization, approximation, and complexity classes, J. of Computer and System Science 43 (1991) 425–440.

[18] Z.-Z. Chen, T. Jiang, G. Lin, Computing phylogenetic roots with bounded degrees and errors, SIAM Journal on Computing 32 (2003) 864–879.

[19] N. Bansal, S. Chawla, A. Blum, Correlation clustering, in: Proceedings of the 43rd Annual Symposium on Foundations of Computer Science (FOCS), 2002, pp. 238–247.