Bicluster Editing with Overlaps: A Vertex Splitting Approach

Faisal N. Abu-Khzam, Lucas Isenmann and Zeina Merchad

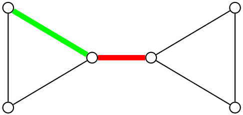
Department of Computer Science and Mathematics
Lebanese American University
Beirut, Lebanon.

Overview

- Why Bicluster Editing?
- Why Vertex Splitting?
- One-sided versus two-sided vertex splitting
- (Polynomial) Computational Complexity
- Fixed-parameter tractability
- Other results and future work

Cluster Editing

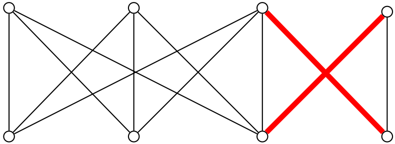
- Given a graph $G = (V, E)$ and an integer k, the objective is to transform G into a disjoint union of cliques (the clusters) via at most k edge editing/modification operations (add/delete)?



- Models correlation clustering: partition the input data set so that elements of the same set are close-enough and pairs from different sets are not close according to a given similarity measure (represented by edges of a graph...)

Bicluster Editing

- Given a bipartite graph $G = (A \cup B, E)$ and an integer k, the objective is to transform G into a disjoint union of **bi**cliques (the bi-clusters) via at most k edge editing/modification operations?



Why bi-clustering?

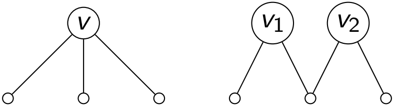
- When data elements are given along with features/attributes, using correlations to build a graph can leads to inaccurate clusters.

  Example:

  A: 11101
  B: 10001
  C: 01001

Vertex Splitting

- A vertex splitting consists of replacing a vertex by two vertices such that the union of the neighborhood the new vertices is the neighborhood of the initial vertex.



Example of a split of v into v_1 and v_2. In general it is possible that the copies share some neighbors.

Bicluster Editing with Vertex Splitting (BCEVS)

- Given a bipartite graph $G = (A \cup B, E)$ and an integer k, the objective is to transform G into a disjoint union of **bi**cliques (the bi-clusters) via at most k edge editions and vertex splittings?

Bicluster Editing with One-Sided Vertex Splitting (BCEOVS)

- Given a bipartite graph $G = (A \cup B, E)$ and an integer k, the objective is to transform G into a disjoint union of **bi**cliques (the bi-clusters) via at most k edge editions and vertex splittings only occurring on the B vertices?

By splitting only one vertex in B (the top vertices), we can get an union of bicliques.

Why Vertex Splitting

- Allows data elements to belong to more than one cluster/group.
- Allows clustering of data that is hard to cluster (e.g. due to hubness).

Main Results

- Both BCEVS and BCEOVS are NP-complete
- Hardness of approximation (both problems)
- Polynomial-time algorithm on trees (both problems)
- Polynomial kernel for BCEOVS

Reduction from 3-SAT

Using a reduction from 3-SAT, we obtain the following:

Theorem

BCEVS and BCEOVS are NP-complete even when restricted to bipartite planar graphs of maximum degree three.

Theorem

Assuming the ETH, there is no $O^*(2^{o(n)})$-time (resp. $O^*(2^{o(\sqrt{n})})$-time) algorithm for BCEVS and BCEOVS on bipartite (resp. planar) graphs with maximum degree three where n is the number of vertices of the graph.

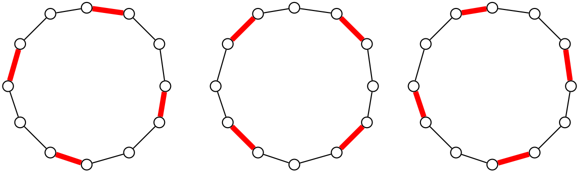
Theorem

BCEVS and BCEOVS are APX-hard.

Reduction from 3-SAT

Observation

Transforming a cycle of length $6k$ into a disjoint union of bicliques requires at least $2k$ operations. This is realized by three possible sequences of $2k$ edge deletions.

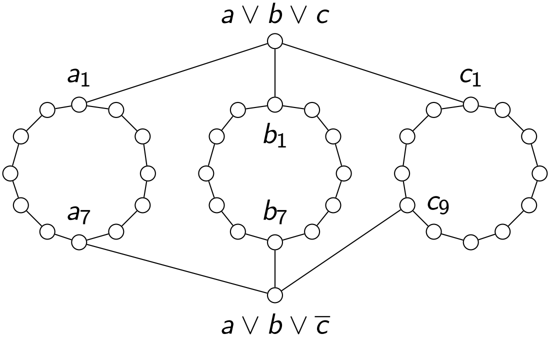


Applying vertex splitting or edge addition (in this case) increases the number of operations.

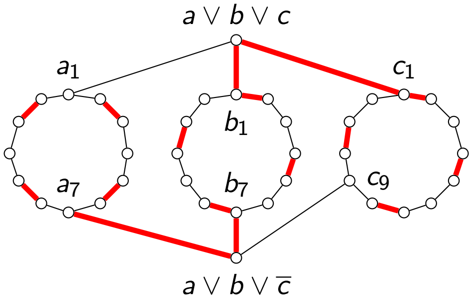
Reduction from 3-SAT

Sketch:

- For every variable we create a cycle
- For every clause we create a vertex
- For every variable appearing in a clause we connect the vertex of the clause to a specific vertex of the cycle of the variable



Reduction from 3-SAT



From a satisfying assignment $a = True$, $b = False$, $c = False$.

Theorem

A formula ϕ with n variables and m clauses is satisfiable if and only if G_ϕ can be turned into a union of bicliques with a sequence of length at most $2m + \sum_{v \in V} 2d(v)$ of operations (where $d(v)$ denotes the number of clauses where v appears).

Sequence to assignment

Given a sequence of length $2m + \sum_{v \in V} 2d(v)$, we prove that

- Each variable v cycle requires $2d(v)$ operations
- Each clause gadget requires 2 operations

The first result comes from the previous observation and the second from a disjunction of cases (to show that exactly two operations per clause suffice).

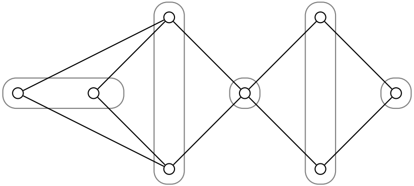
Because of the previous Lemma, if we delete all the edges $v_{2+3k}v_{3+3k}$ for every k for every variable v, then we set v to True. Otherwise we set v to False.

Kernelization

Theorem
BCEOVS has a $O(k^5)$ kernel.

Main idea/sketch of proof: reduce the graph by removing twins (vertices having the same neighborhood).

A twin class is a maximal subset of twins.

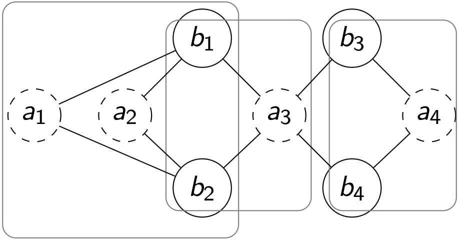


Definition

An A-partitioning cover C of a bipartite graph $G = (A, B, E)$ is a set of subsets (called bags) of $A \cup B$ covering the vertices of G such that the restrictions to A of the bags is a partition of A.

The cost of a cover is the sum of the following quantities:

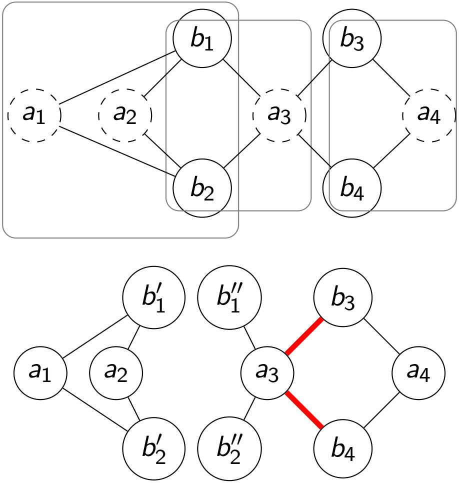
- $\forall b \in B$, $cost(b)$ is the number of bags containing b minus 1
- $\forall a \in A$, $cost(a)$ is the number of edited edges incident to a



$cost(b_1) = cost(b_2) = 2 - 1 = 1$; $cost(a_3) = 2$.

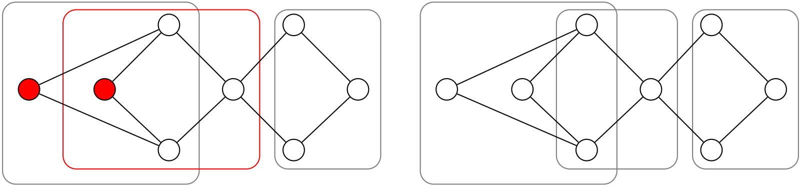
Lemma

A bipartite graph has an A-partitioning cover of cost at most k if and only if there exists a sequence of length at most k of edge editing and vertex splitting operations on B.



Lemma

There exists an A-partitioning cover C of G of minimum cost such that for every twin class T and every subset X of C, then either $T \cap X = \emptyset$ *or* $T \subseteq X$.

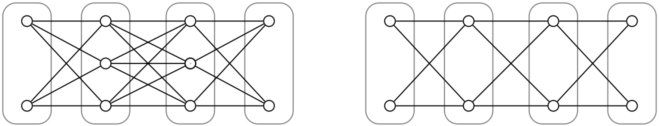


Example: The first cover is not twin adapted while the second is.

Reductions

Lemma (Reduction)

Consider the twin classes $T_1, \ldots, T_p$ of G. For every $i \in [p]$, we consider a subset T_i' of T_i of size $k+1$ if $|T_i| \geq k+1$, otherwise we set T_i' to T_i. Then G has an A-partitioning cover of cost at most k if and only $G[\cup T_i']$ has an A-partitioning cover of cost at most k.



This simply means: If a twin class has more than $k+1$ vertices then delete all but $k+1$ of them.

More Lemmas...

Lemma

Suppose that G has an A-partitioning cover of cost k. Let X be a bag, then $Atc(X) \leq k+1$ where $Atc(X)$ is the number of A-twin class in X.

Lemma

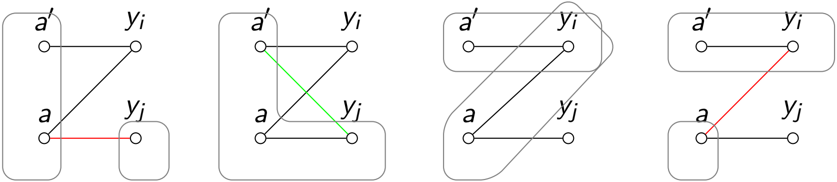
If G has an A-partitioning cover of cost k, then there are at most $(k+1)^2$ twin classes in A.

Lemma

If all twin classes are of size at most k and if G has an A-partitioning cover of cost at most k, then there are at most $4k^4$ twin classes in B.

Proof

We show that, in a yes-instance, at most $2k+1$ B-twin classes are connected to a single vertex in A?
In fact, if $y_1, \ldots, y_{2k+2}$ are B vertices in different twin classes connected to a same vertex $a \in A$.



Then, obviously, each pair of these vertices requires an operation, which requires at least $k+1$ operations leading to a contradiction.

Kernal Bound

There are $O(k^2)$ A-twin classes each of them is of size at most $O(k)$. Each A-vertex is connected to $O(k)$ B-twin classes. Therefore there is at most $O(k^4)$ B-twin classes.

To sum up:

- The vertices of G' are partitioned into twin classes;
- The twin classes are of size at most k;
- There are at most $O(k^2)$ twin classes in A and $O(k^4)$ twin classes in B.

We conclude that G' is of size $O(k^5)$.

Other/Future Results/Work

- We further show that both BCEVS and BCEOVS are solvable in polynomial-time on trees.
- Our kernel bound shows that BCEOVS is FPT but it's still open whether it can be solved in $O^*(c^k)$.
- Better kernel bound? Also for BCEVS?
  - Yes, as also shown independently by [Bentert-Drange-Haugen].
- How about other parameterizations?

Thank you for your attention!