PARAMETERIZED VERTEX COVER, HITTING SET, AND MATCHING IN DYNAMIC GRAPHS

SOFYA VOROTNIKOVA, UNIVERSITY OF MASSACHUSETTS AMHERST

JOINT WORK WITH RAJESH CHITNIS, GRAHAM CORMODE, HOSSEIN ESFANDIARI, MOHAMMADTAGHI HAJIAGHAYI, ANDREW MCGREGOR, MORTEZA MONEMIZADEH

DEFINITIONS

Vertex Cover of a graph $G = (V, E)$ is a subset of vertices $S \subseteq V$ such that every edge has at least one endpoint in S. We denote the minimum vertex cover of G by $\mathsf{vc}(G)$.

Matching in a graph $G = (V, E)$ is a subset of edges $M \subseteq E$ such that no two edges in M share an endpoint. We denote the maximum matching by $\mathsf{match}(G)$.

Hitting Set is an extension of vertex cover to hypergraphs, i.e., a subset of vertices $S \subseteq V$ such that every hyperedge contains at least one vertex from S. We denote the minimum hitting set by $\mathsf{hs}(G)$.

Dynamic Graph is defined by a stream (sequence) of edge insertions and deletions.

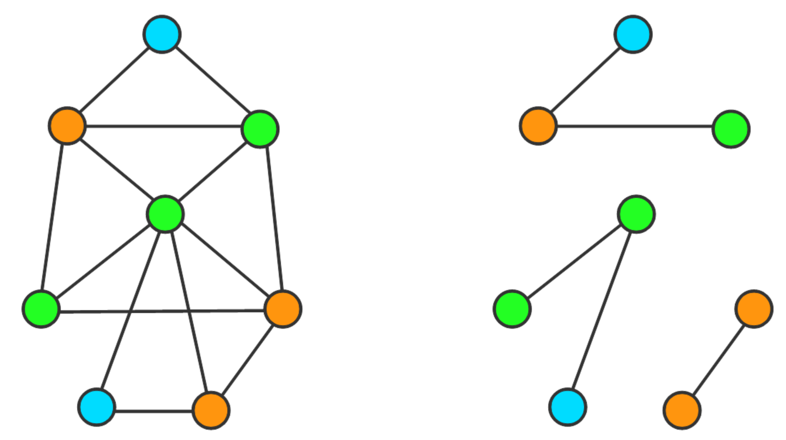
RESULTS

Assume that optimal solution is of size at most k.
In hypergraphs every edge is of size at most d.
After one pass over the stream we obtain a subgraph that preserves the problem solution using:

Space
Vertex cover, matching: $O(k^2 \operatorname{polylog} n)$
Hitting set, hyper-matching: $O(k^d \operatorname{polylog} n)$
Both bounds are tight up to $\operatorname{polylog} n$ factor.

Time: $O(\operatorname{polylog} n)$ update time

SAMPLING



- Color the vertices with b colors.
- For every color combination, sample one edge uniformly at random.
- Repeat r times and take the union of obtained subgraphs.

For our algorithms we take $b = O(k)$ and $r = O(\log k)$. Then the number of color combinations with at most d colors is $O(k^d)$ and the total number of (hyper)edges sampled is $O(k^d \log k)$. In the case of simple graphs $d = 2$ and it gives the bound of $O(k^2 \log k)$ edges.

By varying the number of colors, samples, and color classes to sample from, we can adapt this method to solve a variety of graph problems. The intuition is that when we sample in this manner, we are less likely to sample an edge incident to a high degree vertex than if we sampled uniformly at random from the edge set. For a large family of problems, it is advantageous to avoid bias towards edges whose endpoints have high degree.

LOWER BOUND

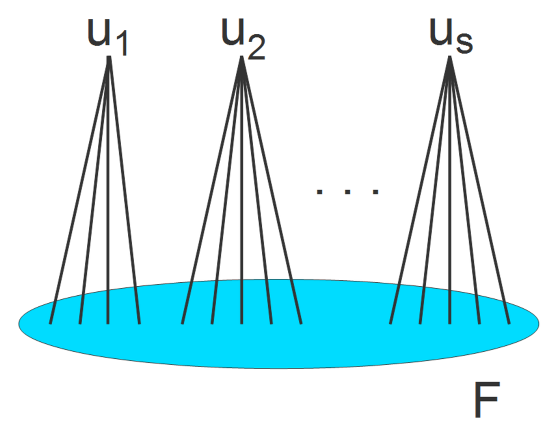
Any (randomized) parameterized streaming algorithm for the minimum d-hitting set or maximum (hyper)matching problem with parameter k requires $\Omega(k^d)$ space. Note that taking $d = 2$ gives a bound for vertex cover and matching in simple graphs.

The result can be shown via a reduction from the INDEX communication problem: Alice has a bit string $S = s_1 s_2 \ldots s_n$ and Bob has an index $x \in [n]$; Bob wants to find s_x via one message from Alice. This requires $\Omega(n)$ bits of communication.

Let $k = \sqrt[d]{n}$. The idea of the proof is the following: Alice's n bit string can be viewed as an adjacency matrix of a certain graph. Alice runs the hitting set algorithm on this graph using space $f(k)$ and sends the memory contents of the algorithm to Bob. Bob continues running the algorithm by inserting more edges based on his index x. From the size of the hitting set/matching of the resultant graph, Bob can infer whether s_x is 0 or 1.

Alice only sends $f(k)$ bits to Bob. Therefore, $f(k) = \Omega(n) = \Omega(k^d)$.

MINIMUM VERTEX COVER



Define
U = vertices with high degree (at least $10k$)
F = edges with both endpoints not in U

Every vertex in U has to be in the vertex cover. Otherwise, we would have to include all of its neighbors in the cover and there are more than k of them. Therefore, $\mathsf{vc}(G) = U \cup \mathsf{vc}(F)$. We show that with high probability we sample all edges in F and enough edges on every vertex in U to recognize its high degree. Thus, minimum vertex cover of the original graph is equal to minimum vertex cover of the sampled subgraph.

Formally, let $G' = (V, E')$ be the subgraph obtained by sampling with $b = O(k)$ colors repeated $r = O(\log k)$ times.

With probability $1 - 1/\operatorname{poly}(k)$ the following holds:

- *Lemma 1*: $F \subseteq E'$
- *Lemma 2*: $\forall u \in U, deg_{G'}(u) \geq 5k$

Therefore, $\mathsf{vc}(G) = \mathsf{vc}(G')$.

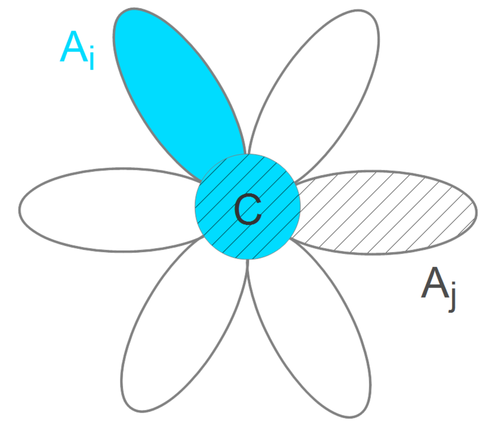
MAXIMUM MATCHING

Let $|\mathsf{match}(G)| = k/2$. Then $|\mathsf{vc}(G)| \leq k$. We run sampling as for the vertex cover algorithm above to obtain sets F and U.

Let M be a maximum matching of G. $F \cap M$ is preserved in G'. If $(u, v) \in M$ such that $u \in U$ there is a chance (u, v) was not sampled. However, at most k vertices participate in the matching, and degree of u in G' is at least $5k$. Therefore, we can always pick a "replacement" edge for the matching from the set of edges incident to u in G'.

Thus, $|\mathsf{match}(G)| = |\mathsf{match}(G')|$.

MINIMUM HITTING SET



Hitting set is a generalization of vertex cover, so the proof idea is similar. However, the combinatorial structure we need to analyze is more complicated in case of hypergraphs. We employ a result from set theory called Sunflower Lemma.

Sunflower Lemma: Let $\mathcal{F}$ be a collection of sets. Then $A_1, \ldots, A_s \in \mathcal{F}$ is an **s-sunflower** if $A_i \cap A_j = C$ for all $1 \leq i < j \leq s$. We refer to C as the **core** of the sunflower and $A_i \setminus C$ as **petals**. If each set in $\mathcal{F}$ has size at most d and $|\mathcal{F}| > d!k^d$, then $\mathcal{F}$ contains a $(k+1)$-sunflower.

Define a **large sunflower** to have more than ak petals, where a is a large constant, and a **significant sunflower** to have over k petals.
U = cores of all large sunflowers.
F = edges not containing large sunflower cores.
U' = sunflower cores in U that do not contain cores of significant sunflowers as subsets.

Similarly to the vertex cover proof, it can be shown that $\mathsf{hs}(G) = \mathsf{hs}(U \cup F) = \mathsf{hs}(U' \cup F)$.

Let $G' = (V, E')$ be the subgraph obtained by sampling with $b = O(k)$ colors repeated $r = O(\log k)$ times.

With probability $1 - 1/\operatorname{poly}(k)$ the following holds:

- *Lemma 3*: $F \subseteq E'$
- *Lemma 4*: In G' the number of petals in a maximum sunflower on core C is greater than k for all $C \in U'$.

Therefore, $\mathsf{hs}(G) = \mathsf{hs}(G')$.

Matching in a hypergraph can be found the same way as in a simple graph. Let $|\mathsf{match}(G)| = k/d$. Each edge of the matching is either contained in F or can be replaced by an edge from the same sunflower in G'.