# Kernelization via Sampling with Applications to Finding Matchings and Related Problems in Dynamic Graph Streams

Sofya Vorotnikova

University of Massachusetts Amherst

Joint work with Rajesh Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, and Morteza Monemizadeh

# Problem Description

Want to solve problems on massive graphs in a variety of computational models including

- streaming
- distributed systems such as MapReduce

using **edge sampling**.

- Small solution size: want the subsampled graph to preserve the size of the optimal solution exactly.
- Unbounded solution size: want to obtain a good approximation of the optimal solution.

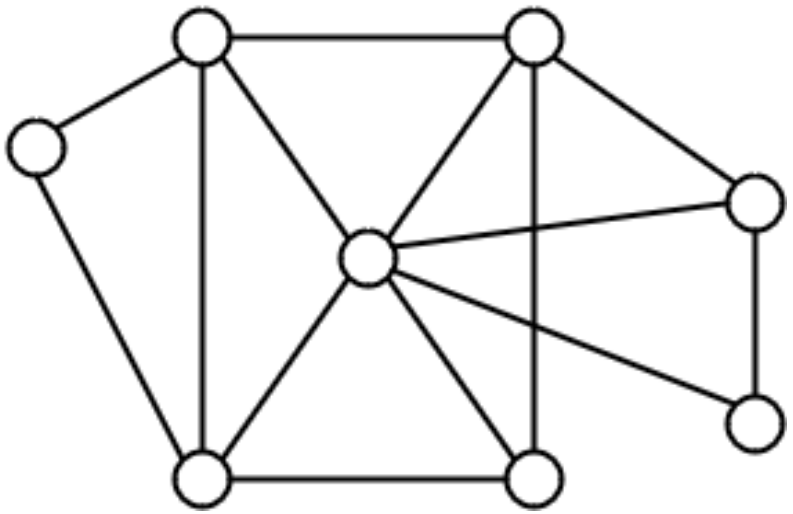
# Results

Space for finding **maximum matching** in dynamic graph streams.

| | Previous Results | Our Results |
|---|---|---|
| *Parameterized*: $\lvert\mathrm{OPT}\rvert \leq k$ | $\tilde{O}(kn)$ Chitnis et al. | $\tilde{O}(k^2)$ |
| *Unbounded* solution size, $\alpha$-approx | | $\tilde{O}(n^2/\alpha^3)$ simultaneous: Assadi et al. |

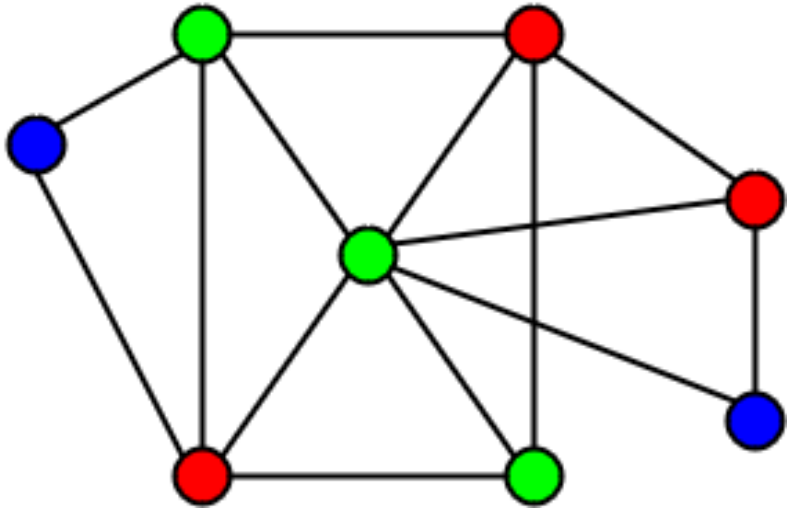
- Our results are tight up to polylog factors.
- Methods apply to other problems: vertex cover, weighted matching, b-matching, disjoint paths, vertex coloring, etc.
- Parameterized results extend to hypergraphs: hitting set, hyper-matching.
- $\tilde{O}(1)$ update time for parameterized results.

# Sampling Procedure



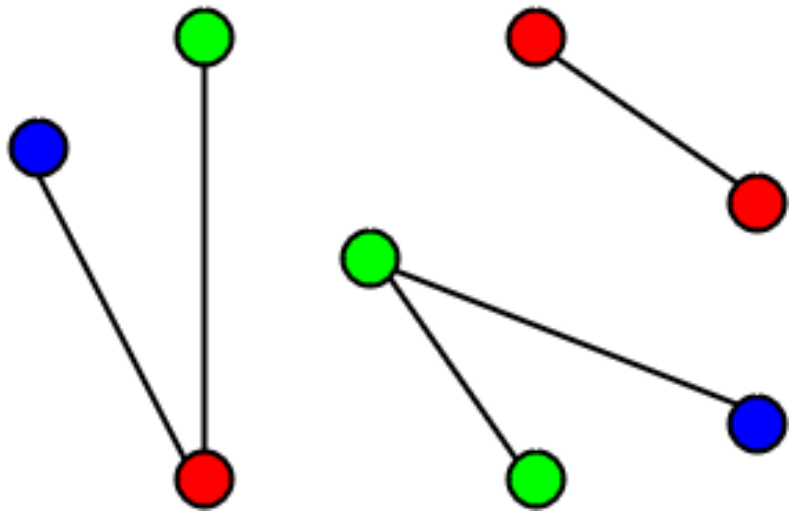
# Sampling Procedure

- Color vertices with $b$ colors.

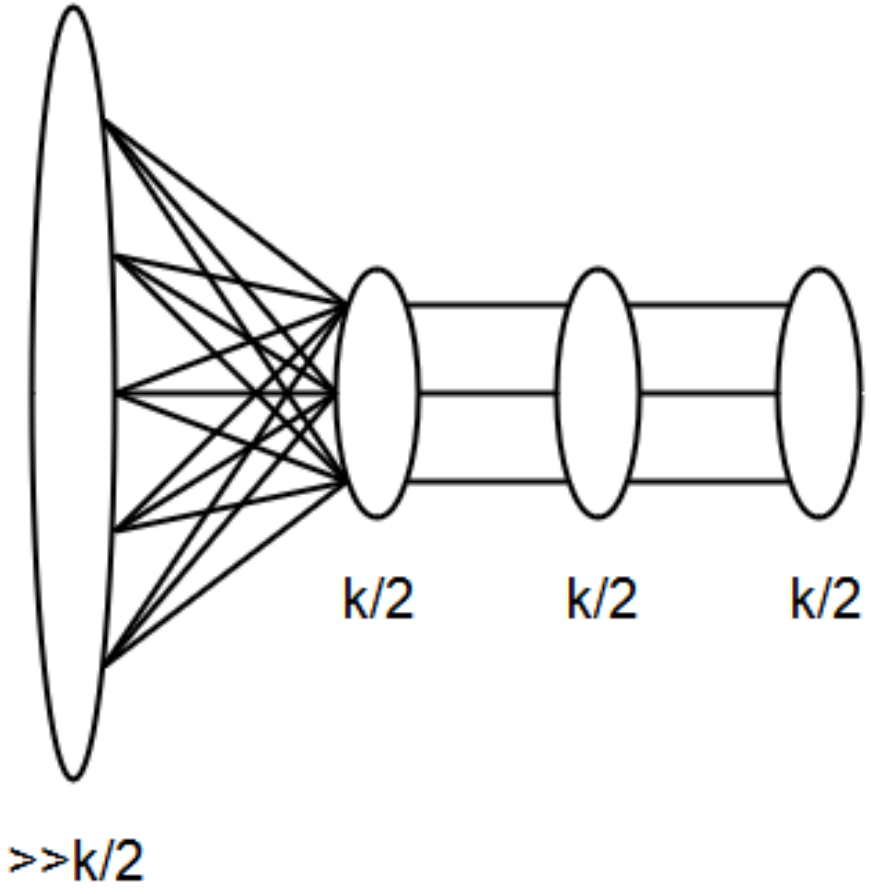


# Sampling Procedure

- Color vertices with $b$ colors.
- For every color pair, sample one edge uniformly at random.

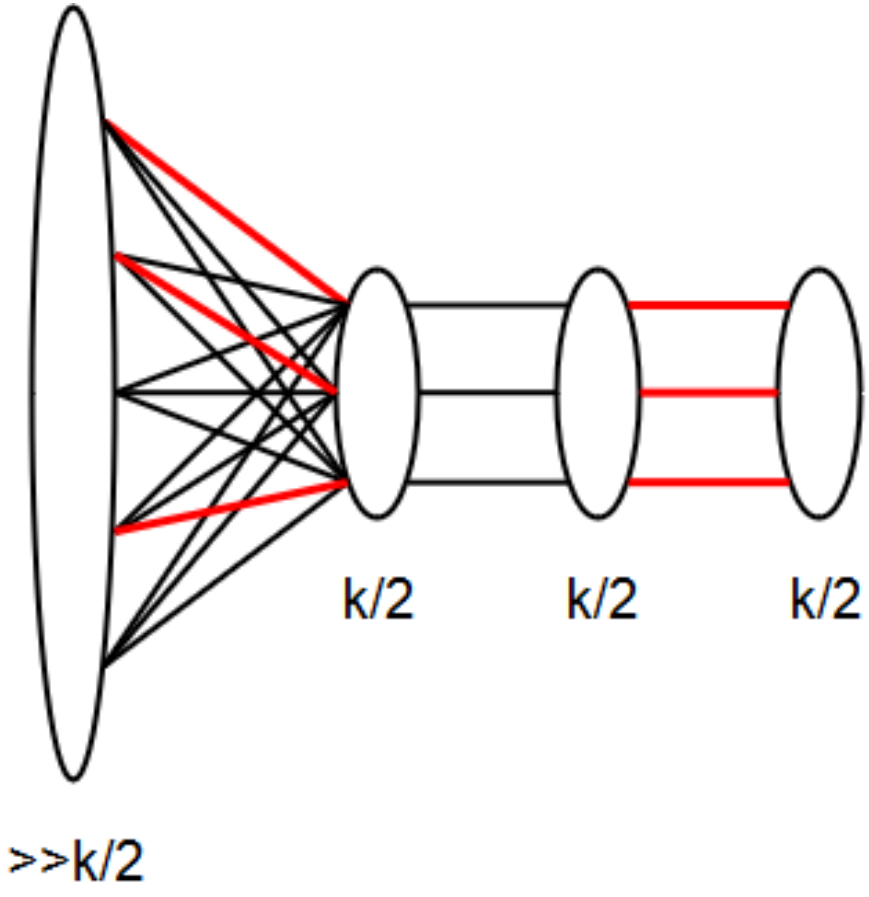


# Why not just sample uniformly?



# Why not just sample uniformly?

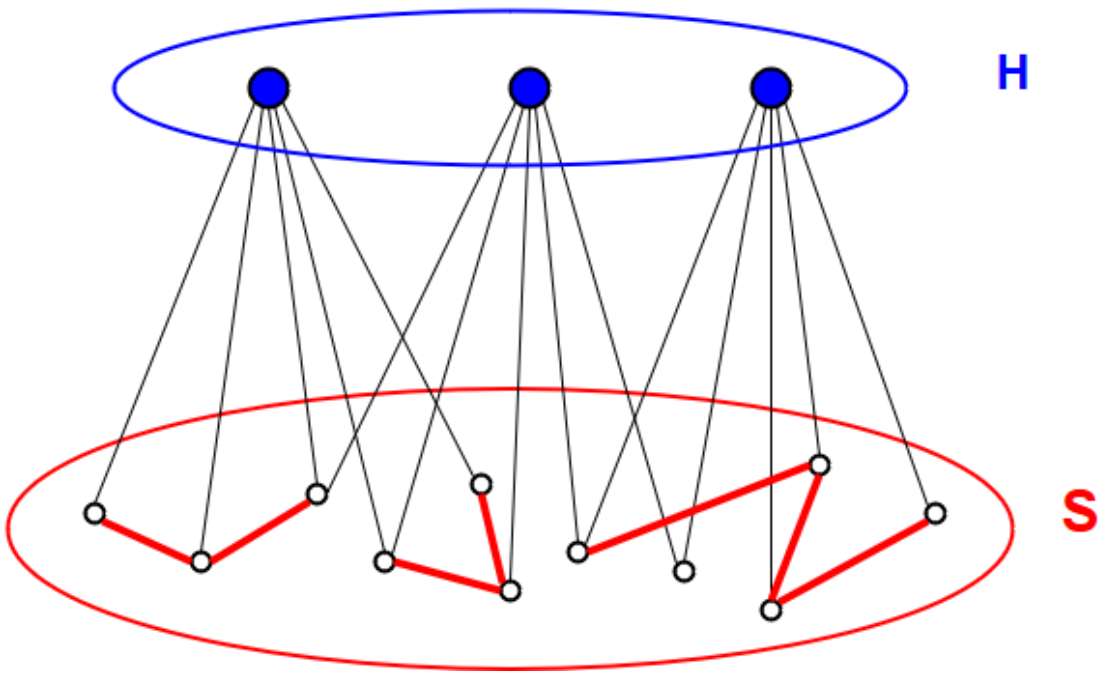
The size of maximum matching is $k$.




Oversampling from dense parts of the graph.

# Minimum Vertex Cover

Assume $|\mathrm{vc}(G)| \leq k$.
$H$ = set of **heavy** vertices with degree at least $10k$
$S$ = set of **shallow** edges s.t. both endpoints are not heavy




$\mathrm{vc}(G) = H \cup \mathrm{vc}(S)$

# Minimum Vertex Cover

Let $G'$ be the graph obtained by sampling using $\Theta(k)$ colors.

With constant probability:

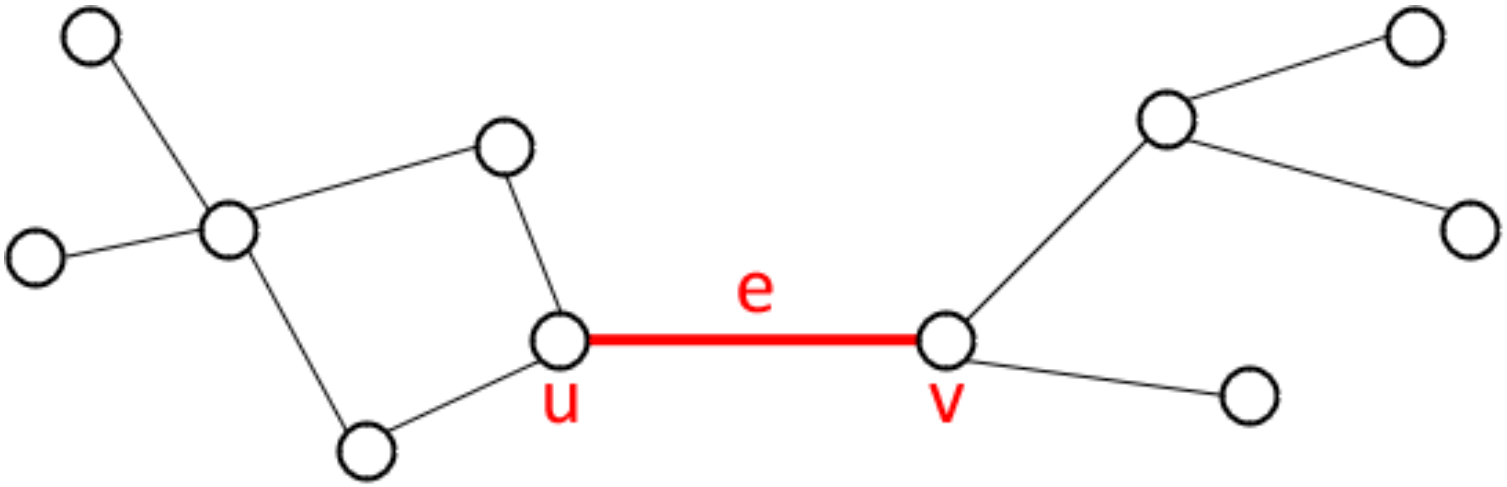
- *Lemma 1*: If edge $e$ is shallow, then $e$ is sampled
- *Lemma 2*: If vertex $v$ is heavy, its degree in $G'$ is at least $5k$

By repeating the sampling $\Theta(\log n)$ times we sample all edges in $S$ and detect all vertices in $H$ with high probability.

Thus, preserving the size of vertex cover.

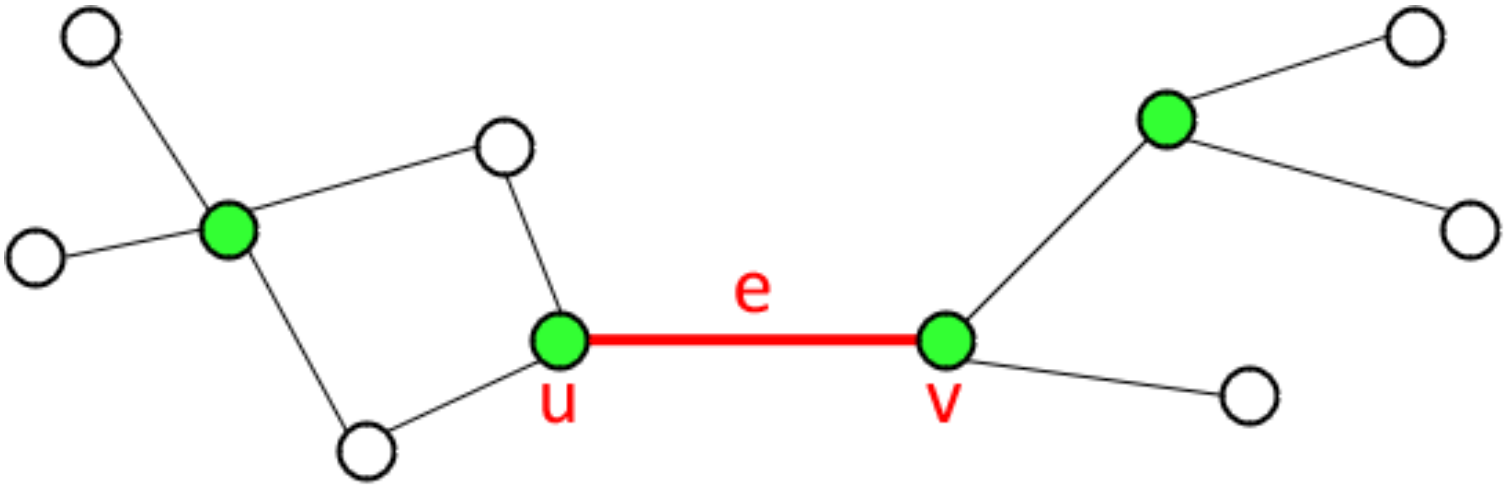
# Proof of Lemma 1: Shallow Edges

Let $e = (u, v)$ be a shallow edge.



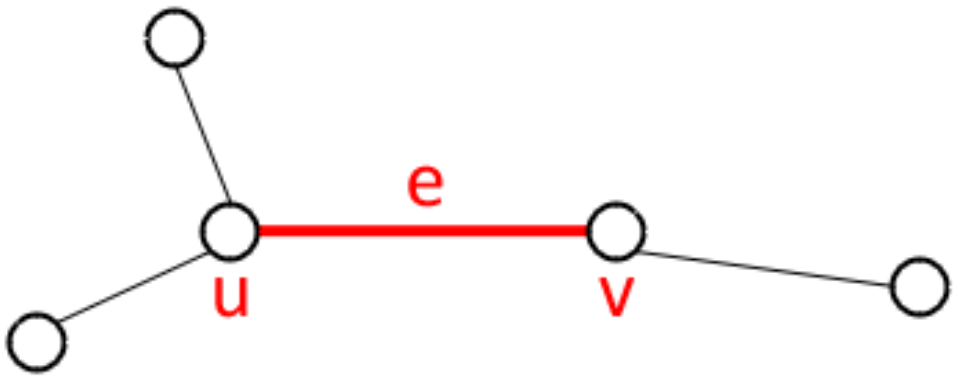
# Proof of Lemma 1: Shallow Edges

Consider minimum vertex cover $VC$ of $G$.



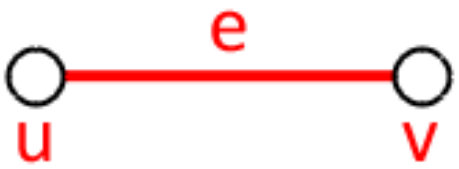
# Proof of Lemma 1: Shallow Edges

Delete vertices in $VC \setminus \{u, v\}$.



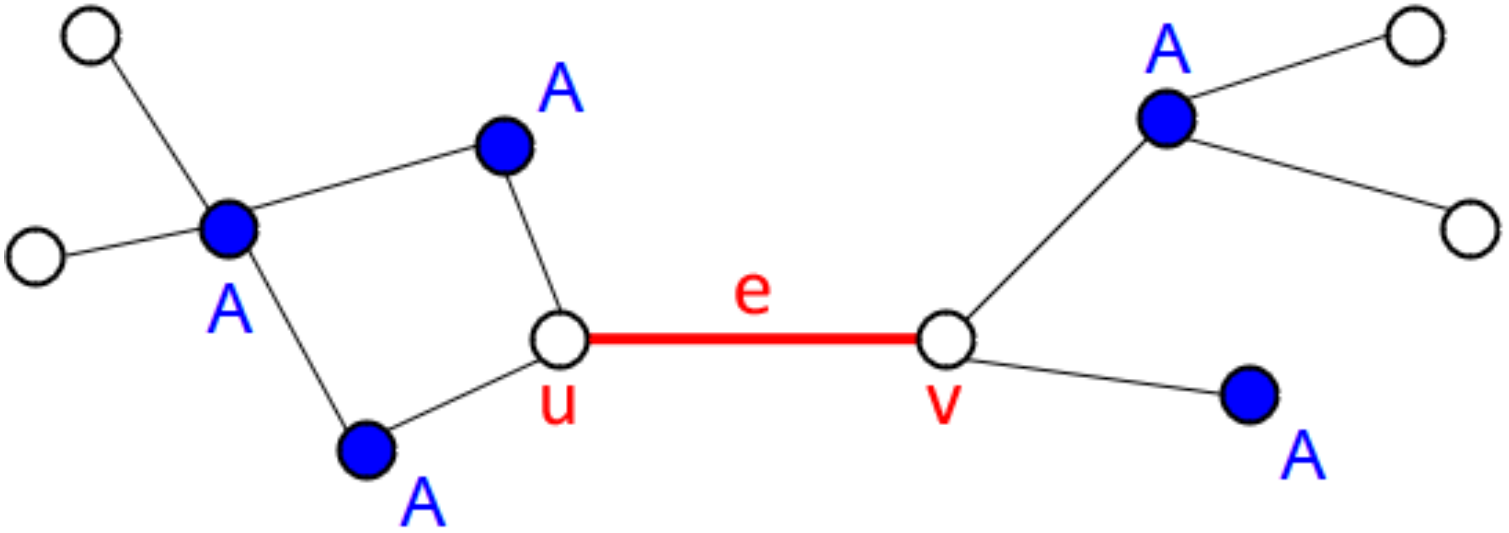
# Proof of Lemma 1: Shallow Edges

Delete vertices in $N(u) \cup N(v) \setminus \{u, v\}$.



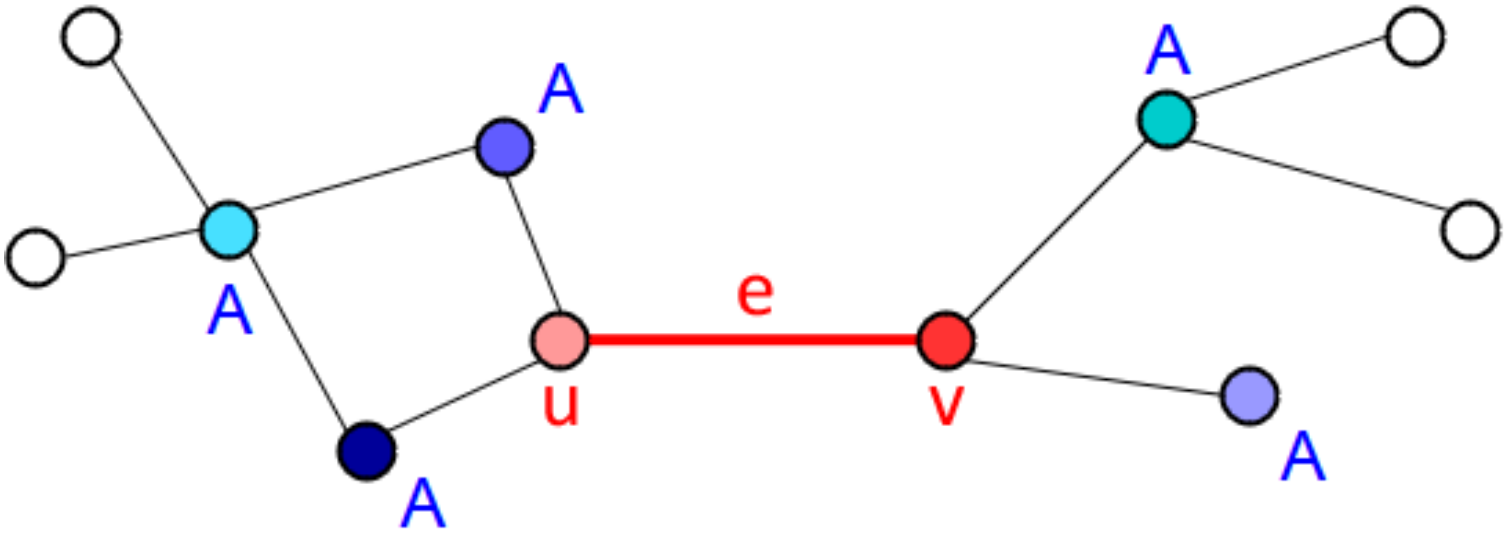
# Proof of Lemma 1: Shallow Edges

Let $A = (VC \cup N(u) \cup N(v)) \setminus \{u, v\}$ be the set of deleted vertices. Note that all edges except $e$ have an endpoint in $A$.



# Proof of Lemma 1: Shallow Edges

If $u$ and $v$ are colored differently than vertices in $A$, then $(u, v)$ is a uniquely colored edge and it will be sampled.

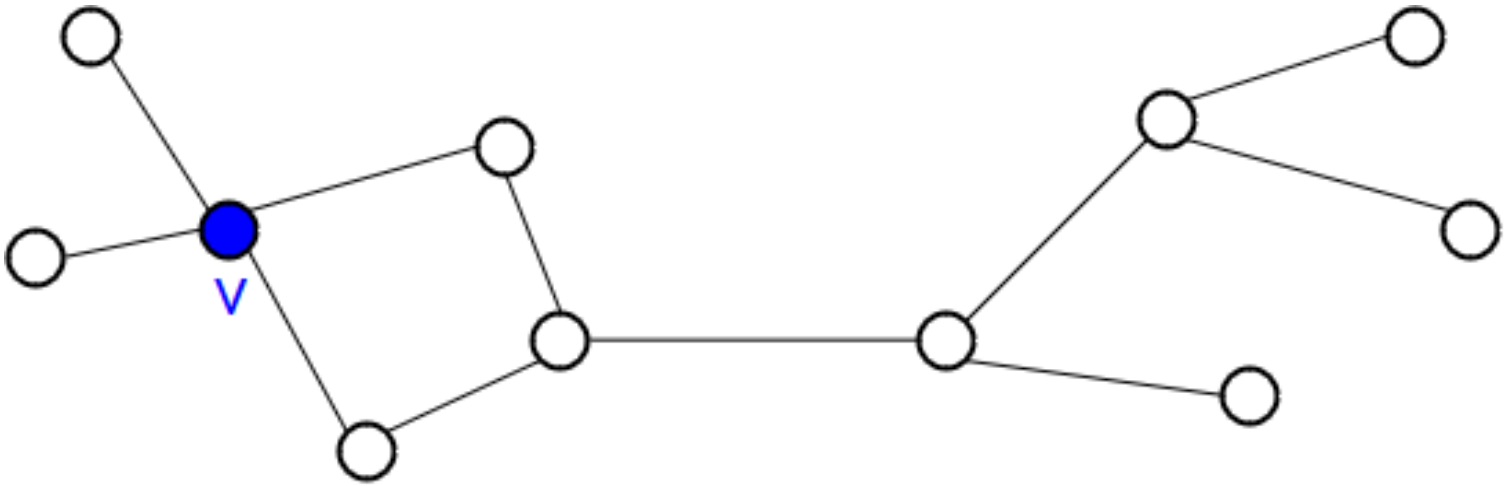



This happens with probability at least

$$1 - \frac{2|A|}{\#\text{colors}} \geq 1 - \frac{2(|VC|+|N(u)|+|N(v)|)}{\#\text{colors}} > 1 - \frac{2(k+10k+10k)}{ck} \geq \frac{3}{4}$$

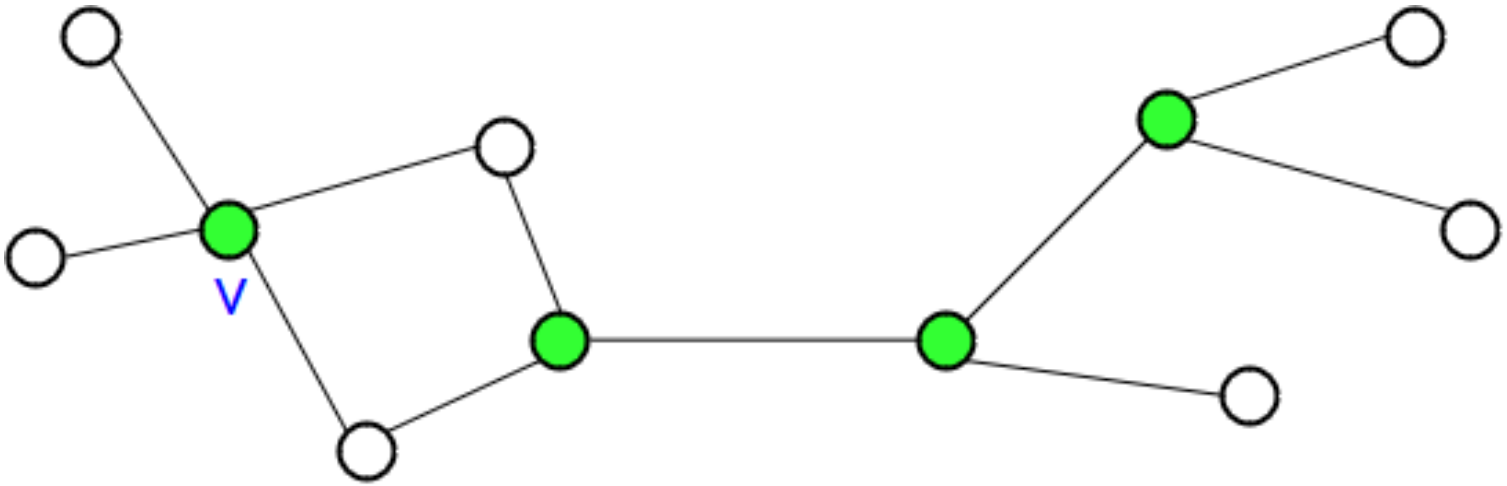
# Proof of Lemma 2: Heavy Vertices

Let $v$ be a heavy vertex.



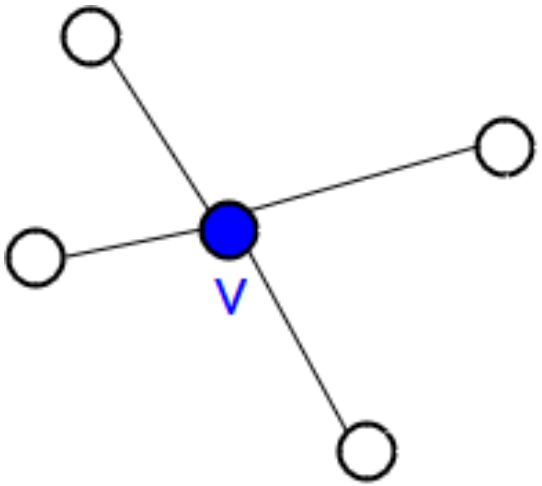
# Proof of Lemma 2: Heavy Vertices

Consider minimum vertex cover $VC$ of $G$.



# Proof of Lemma 2: Heavy Vertices

Delete vertices in $A = VC \setminus \{v\}$.
All edges not incident to $v$ have an endpoint in $A$.

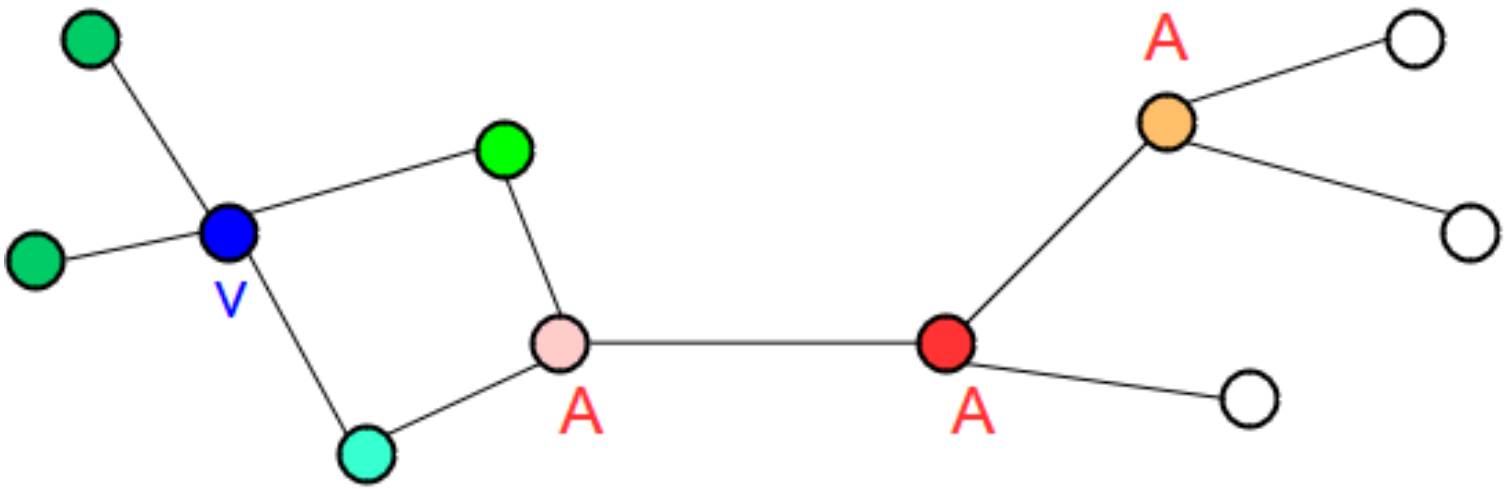


# Proof of Lemma 2: Heavy Vertices

If $v$ and $5k$ of its neighbors have colors that are

1. distinct
2. different from any color in $A$

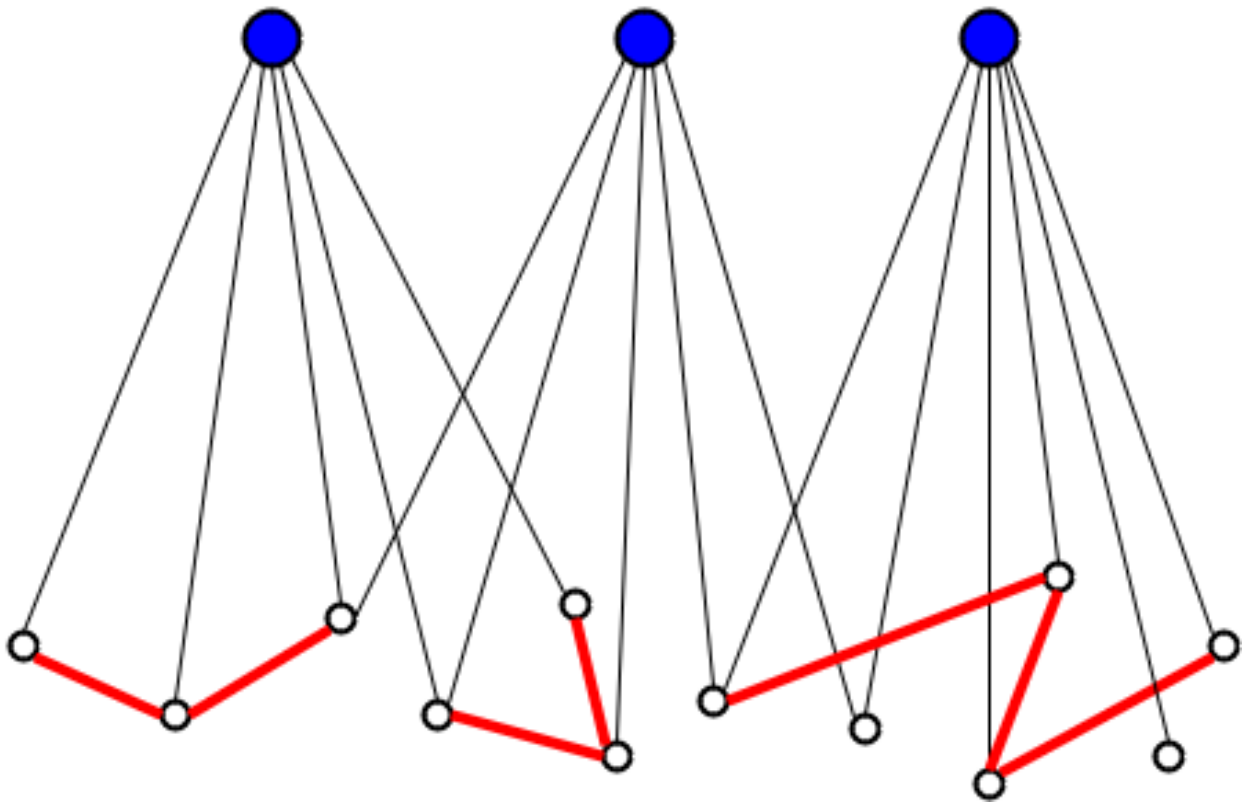
then those $5k$ edges on $v$ will be sampled.




This happens with constant probability.

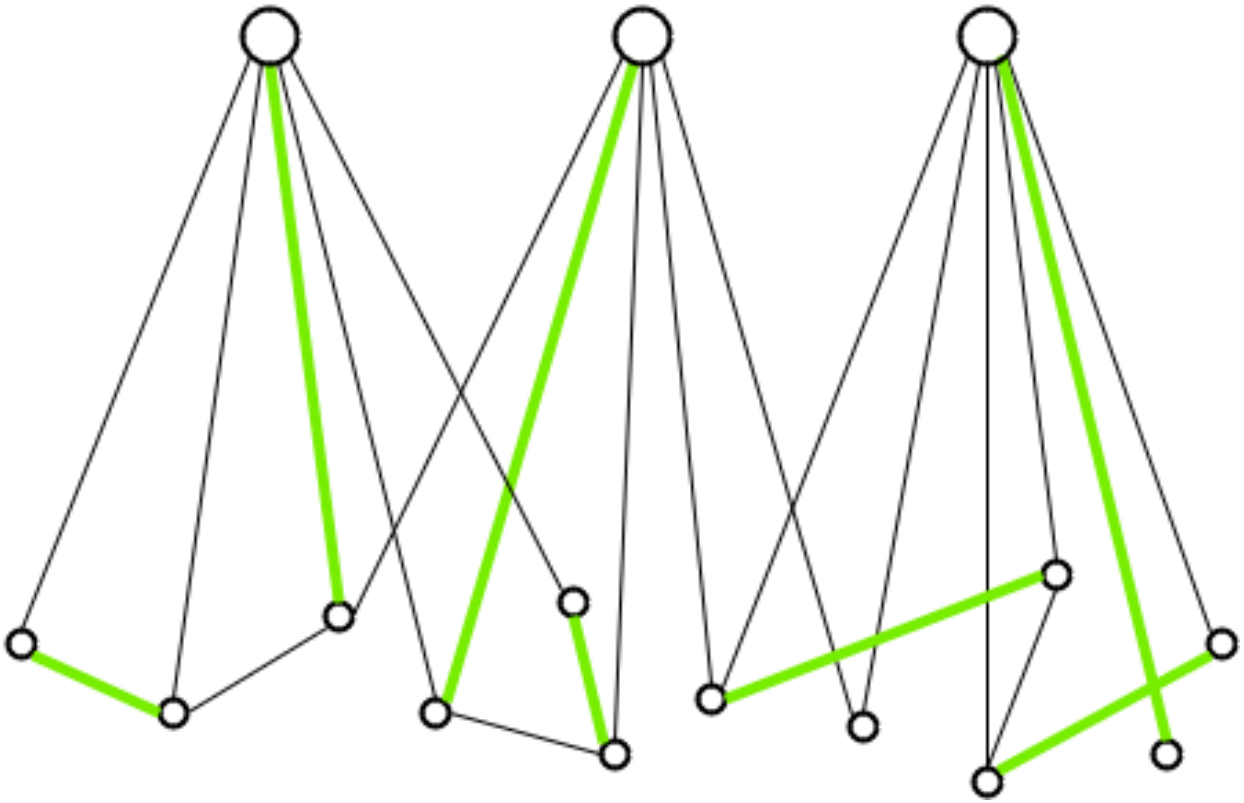
# Maximum Matching

Let $|\mathrm{match}(G)| \leq k$. Then $|\mathrm{vc}(G)| \leq 2k$.



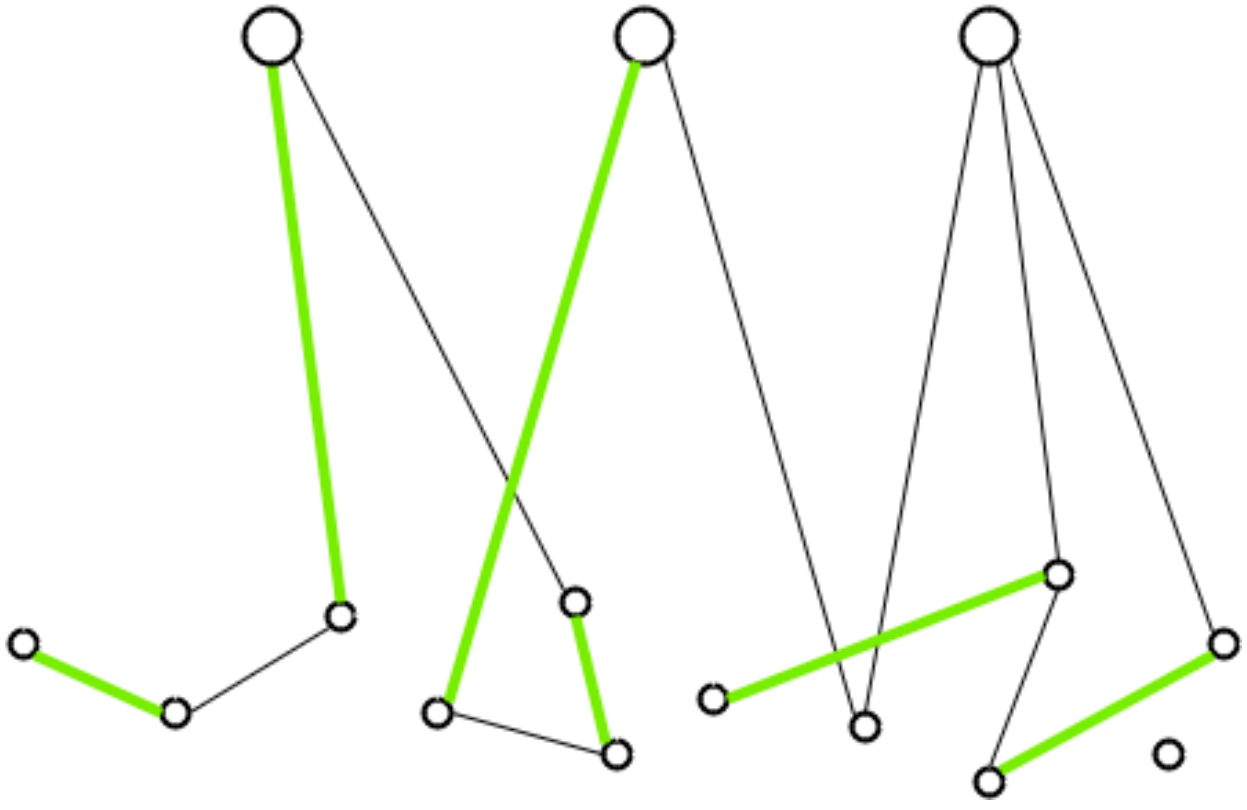
# Maximum Matching

Consider maximum matching $M$ in $G$.



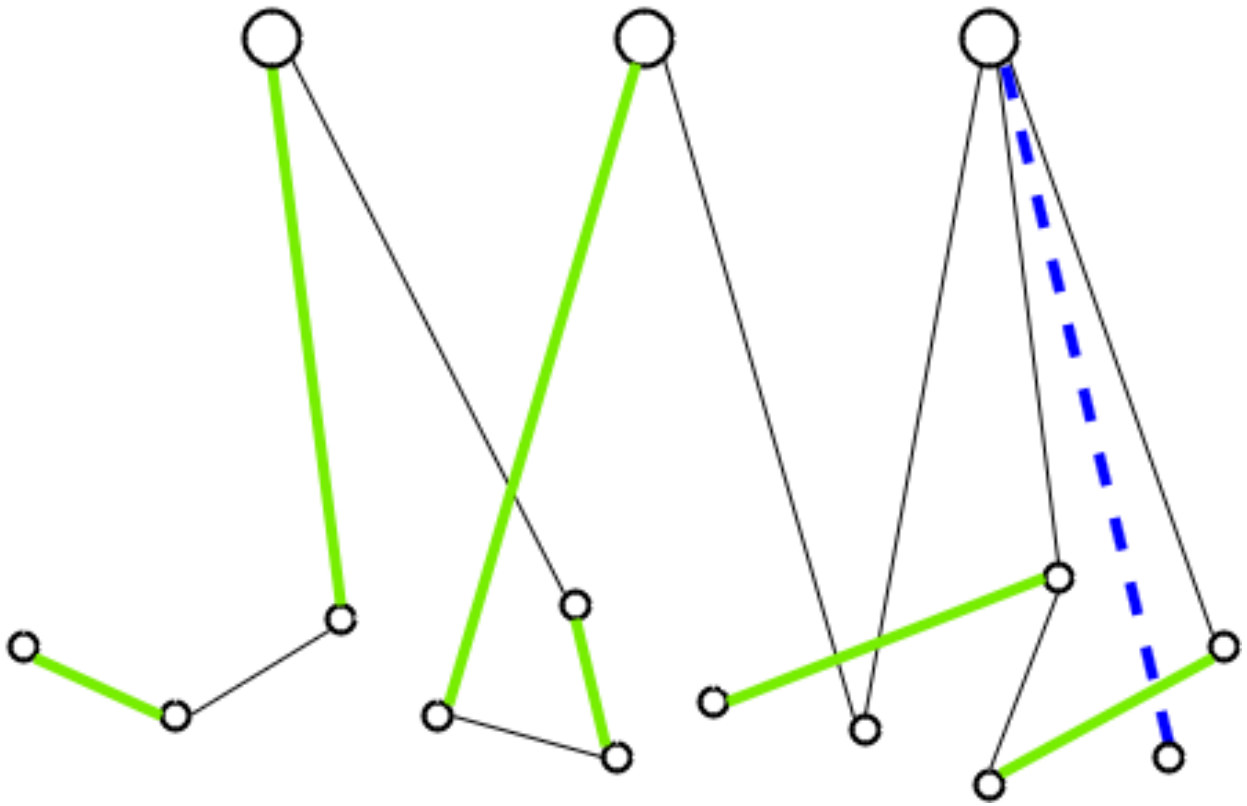
# Maximum Matching

Subsampled graph $G'$:



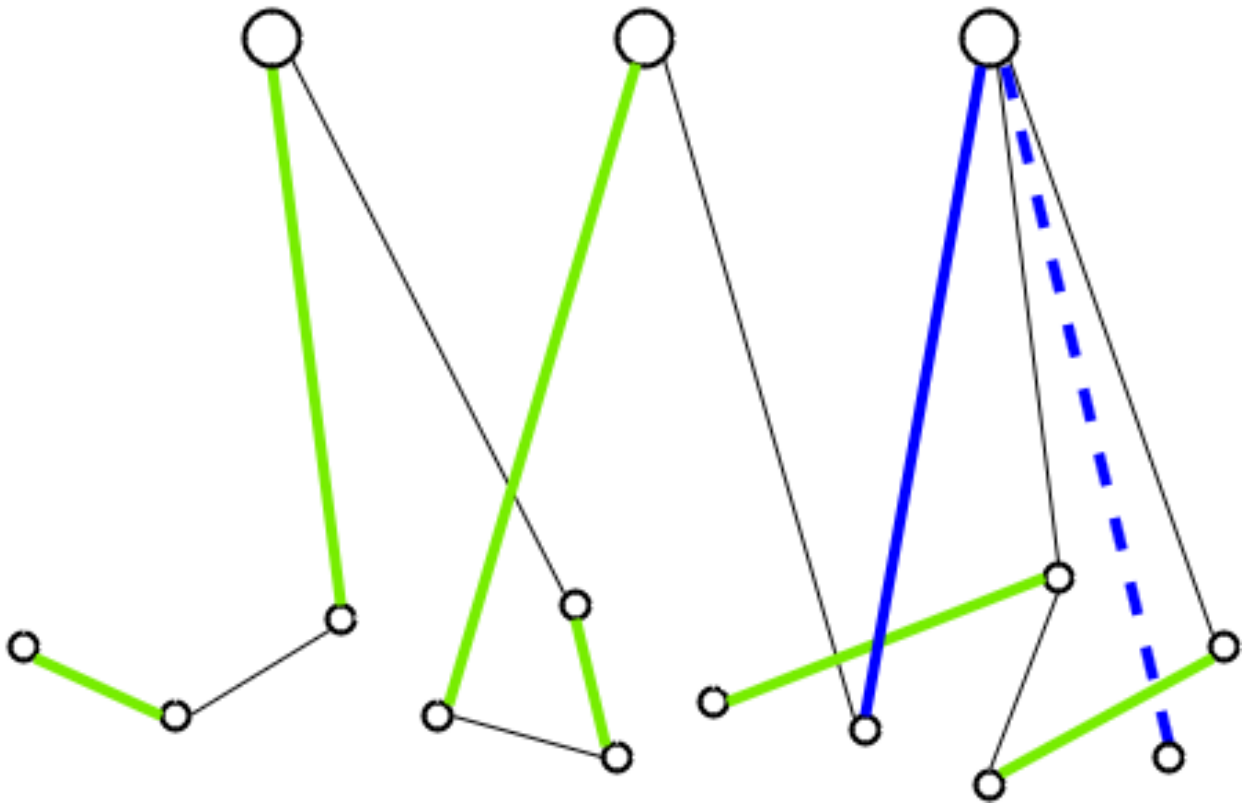
# Maximum Matching

Shallow edges of the matching are preserved in $G'$.
Suppose we did not sample an edge incident to a heavy vertex.



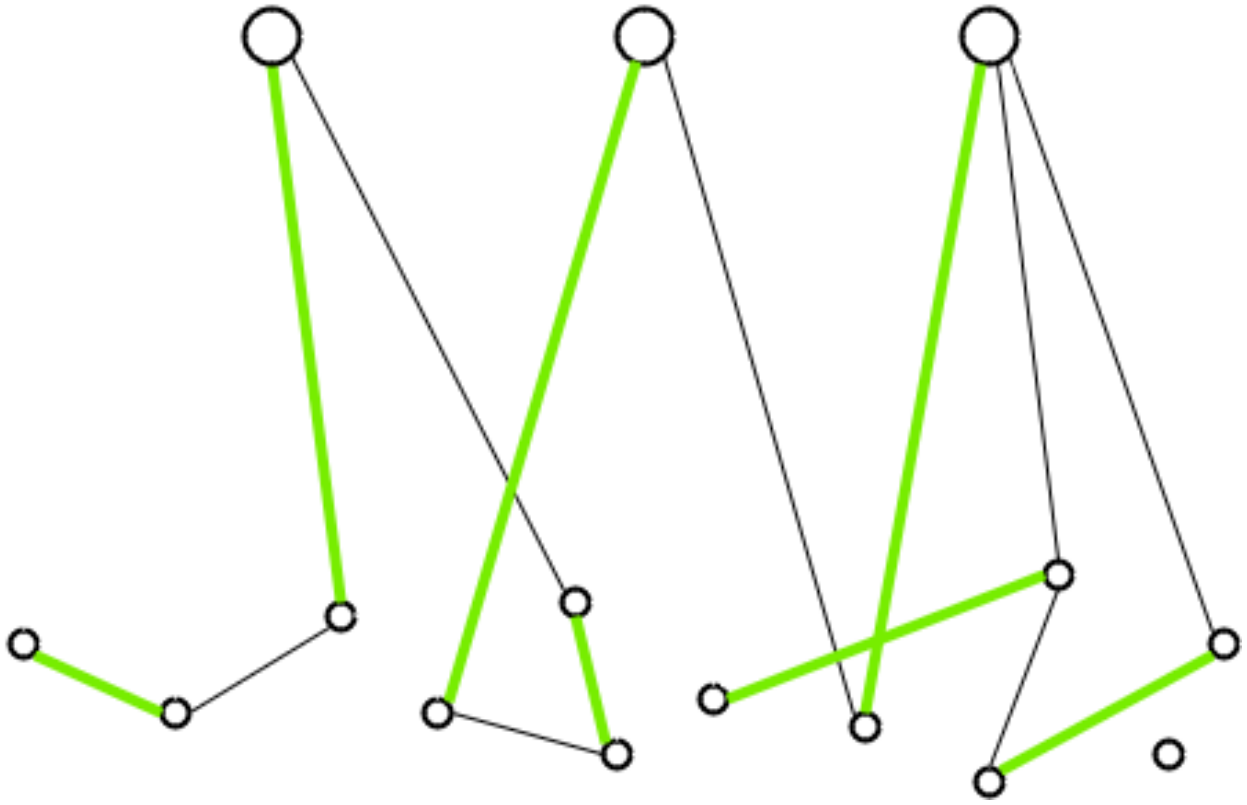
# Maximum Matching

Then we can always pick a "replacement" edge for the matching from the set of edges incident to that vertex in $G'$.



# Maximum Matching

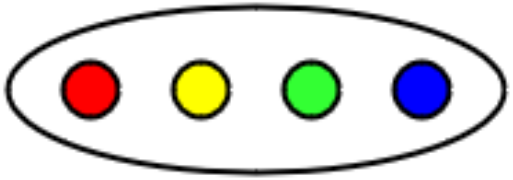
Thus, preserving the size of maximum matching.



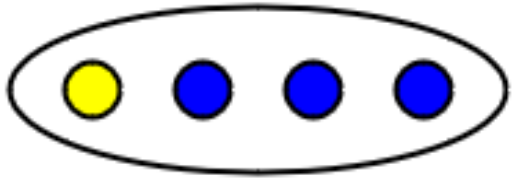
# Parameterized Results: Extension to Hypergraphs

Assume that every hyperedge has constant cardinality $d$.

Sample one hyperedge for every combination of $d$ colors.



( RED, YELLOW, GREEN, BLUE )



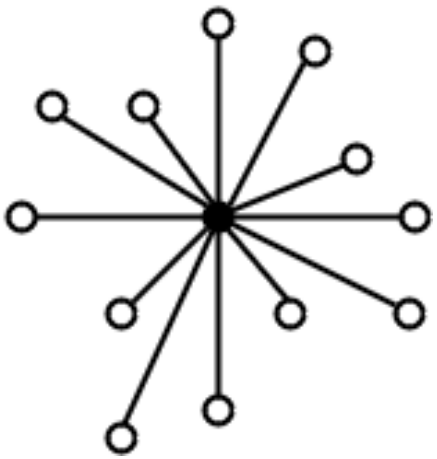
( 1 YELLOW, 3 BLUE )

Using $\Theta(k)$ colors preserves the size of minimum hitting set and maximum hyper-matching in the subsampled hypergraph.

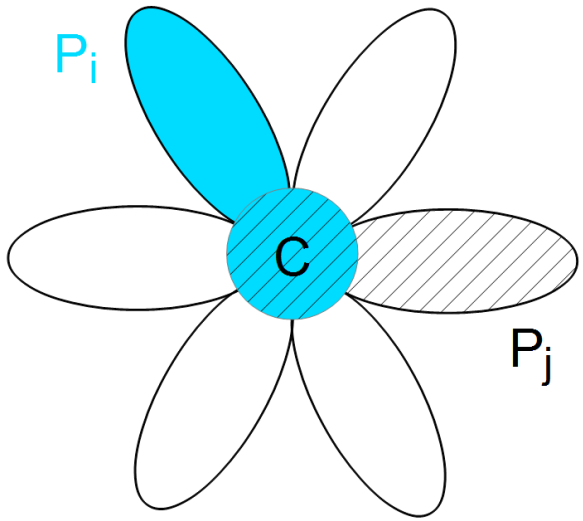
Space used by the algorithm is $\tilde{O}(k^d)$.

# Sunflowers

**Sunflower**: collection of sets with the same pairwise intersection.
Sets are referred to as **petals** and common intersection as the **core**.



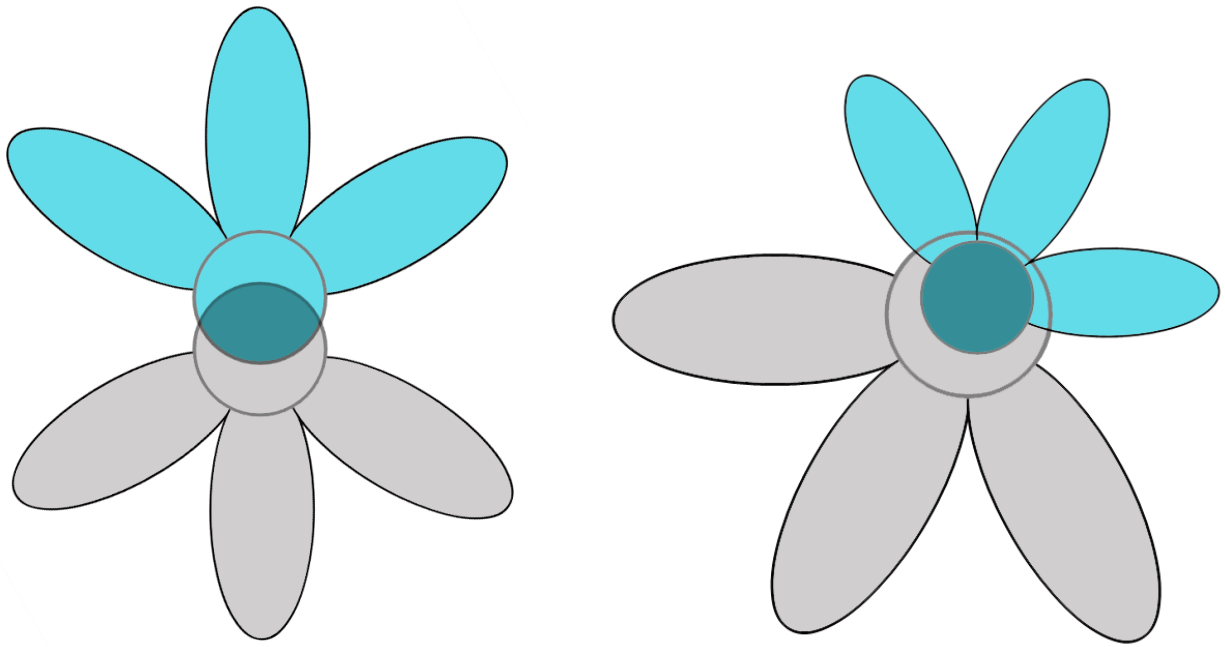
Heavy vertex: high degree
Shallow edge: both endpoints are not heavy




Heavy sunflower: many petals
Shallow hyperedge: does not contain heavy sunflower cores

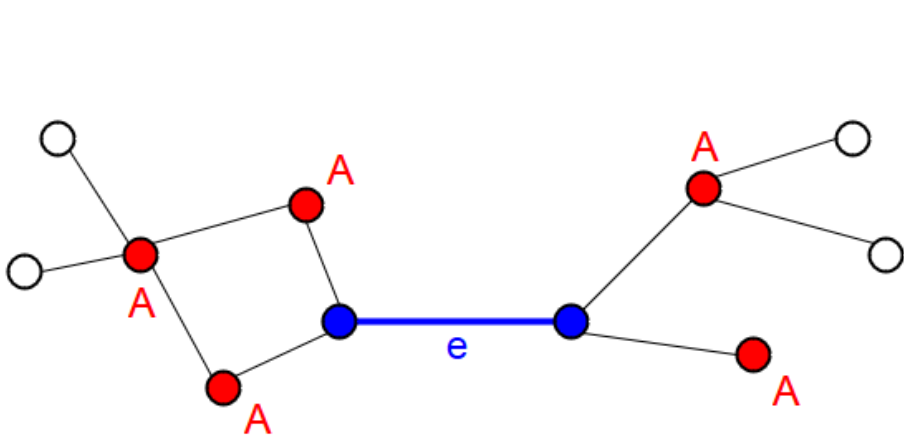
# Sunflowers

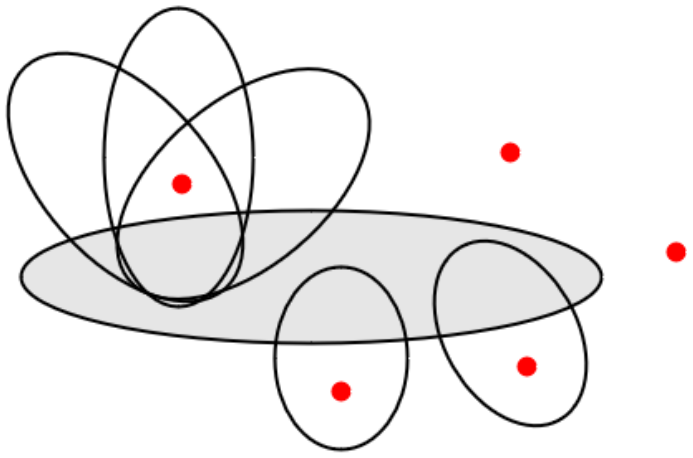
Structure we need to analyze is a lot more complicated.



We only need to consider heavy sunflowers with cores that don't contain cores of other large sunflowers.

# Minimum Hitting Set: Shallow Edges

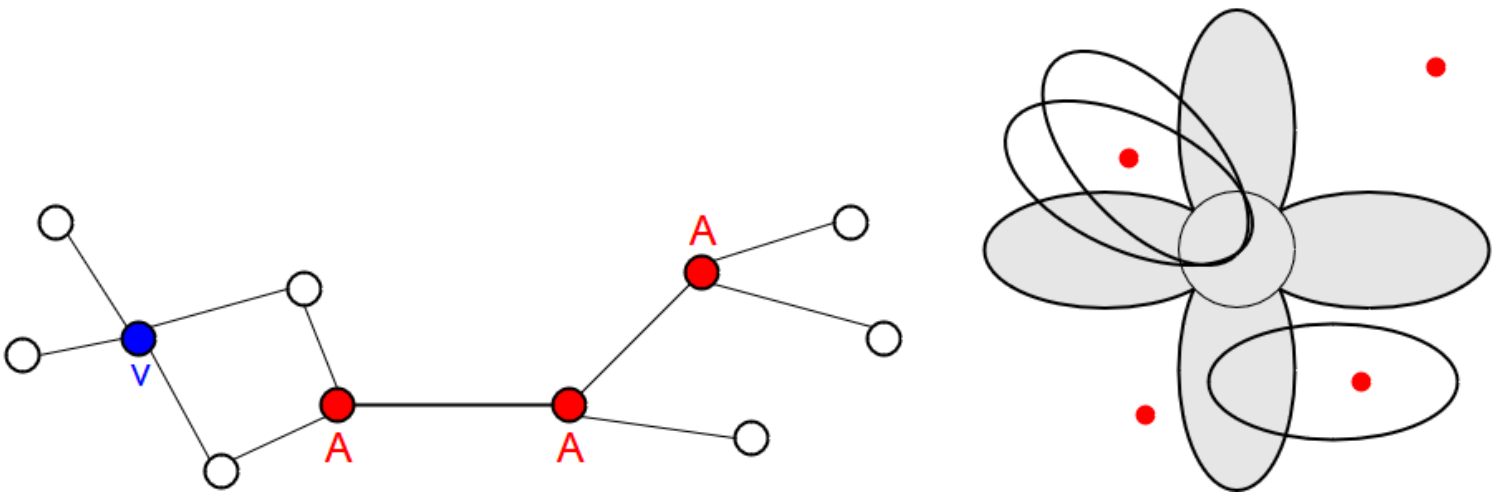


For every shallow hyperedge $e$, want to find a set of vertices $A$ s.t. all edges except $e$ contain a vertex from $A$.

Can show that there exists such a set of size $O(k)$.

# Minimum Hitting Set: Heavy Cores




For every heavy core $C$, want to find a set $A$ that does not intersect $C$ s.t. all edges not containing $C$ contain a vertex from $A$.

Can show that there exists such a set of size $O(k)$.

# Maximum Hyper-matching

Let $|\mathsf{match}(G)| \leq k$. Then $|\mathsf{hs}(G)| \leq dk$.

- Shallow matching edges are sampled.
- A matching edge on a heavy core might not be sampled, but we can always pick a "replacement" edge.

Thus, preserving the size of hyper-matching.

# Approximating Large Maximum Matchings

We present an $\tilde{O}(\frac{n^2}{\alpha^3})$-space algorithm that returns an $\alpha$-approximation for matchings of arbitrary size.

This implies an $n^{1/3}$-approximation using $\tilde{O}(n)$ space.

Instead of sampling an edge for every color pair, we sample edges with **endpoints of the same color**.

# Approximating Large Maximum Matchings

- Color the graph using $\Theta(\frac{k}{\alpha})$ colors
- For each color sample an edge with endpoints of that color
- Repeat $\tilde{O}(\frac{k}{\alpha^2})$ times and take the union of samples (call $G'$)

**Lemma**

*If* $|\text{match}(G)| \geq k$, *then* $|\text{match}(G')| \geq \frac{k}{\alpha}$ *whp.*

By running $\log(n)$ copies in parallel for $k = 1, 2, 4, 8, ..., n$ we obtain an $\alpha$-approximation using $\tilde{O}(\frac{n^2}{\alpha^3})$ space.

# Lemma Proof Idea

Construct matching greedily: in each sampling round pick sampled edges that can be added to the current matching.

We can show that with constant probability, if current matching is smaller than $\frac{k}{\alpha}$, we can add $\Omega(\alpha)$ edges to it in the next round.

Thus, after $\tilde{O}(\frac{k}{\alpha^2})$ rounds, we have a matching of size $\frac{k}{\alpha}$ whp.

# Why you should read our paper

What did not make it into this talk:

- Detailed proofs of the parameterized results for hypergraphs
- Extension of the matching results to weighted case
- Matchings in graphs with low arboricity
- General family of parameterized problems
- Lower bounds for problems outside of that family
- Implementation details for dynamic graph streams

# Conclusion

**General sampling method**:

- color vertices
- sample edges with particular color combinations

Can be used for solving a variety of graph problems. For problems with unbounded solution size provides approximate solutions.

Can be easily implemented in a variety of computational models including dynamic graph streams and distributed systems such as MapReduce.

Thank you for your attention!

# Bibliography

S. Assadi, S. Khanna, Y. Li, and G. Yaroslavtsev.
*Maximum Matchings in Dynamic Graph Streams and the Simultaneous Communication Model.*
SODA 2016.

R. H. Chitnis, G. Cormode, M. T. Hajiaghayi, and M. Monemizadeh.
*Parameterized Streaming: Maximal Matching and Vertex Cover.*
SODA 2015.