Algorithms for

Independent transversals vs. small dominating sets

Penny Haxell

University of Waterloo

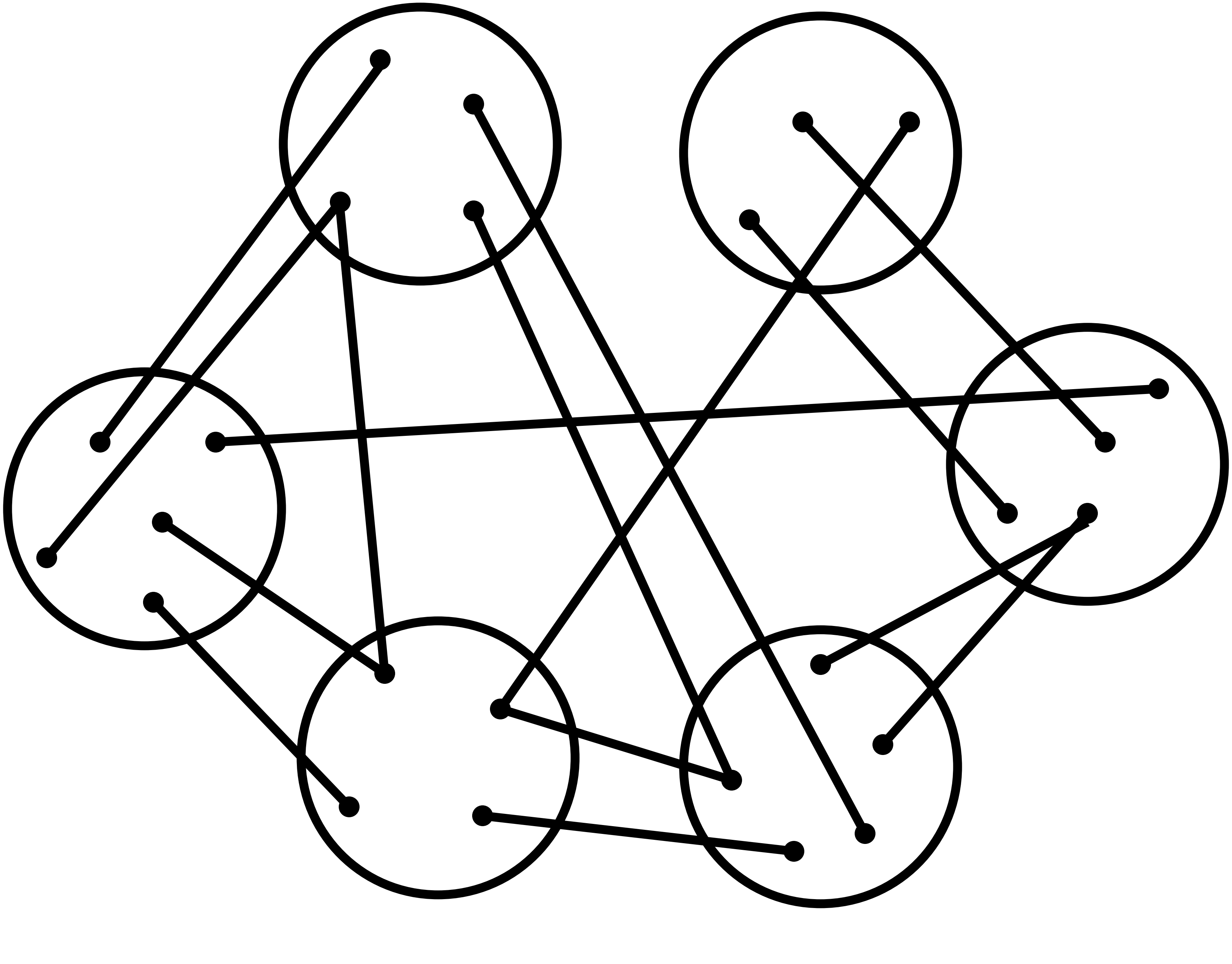
joint work with Alessandra Graf

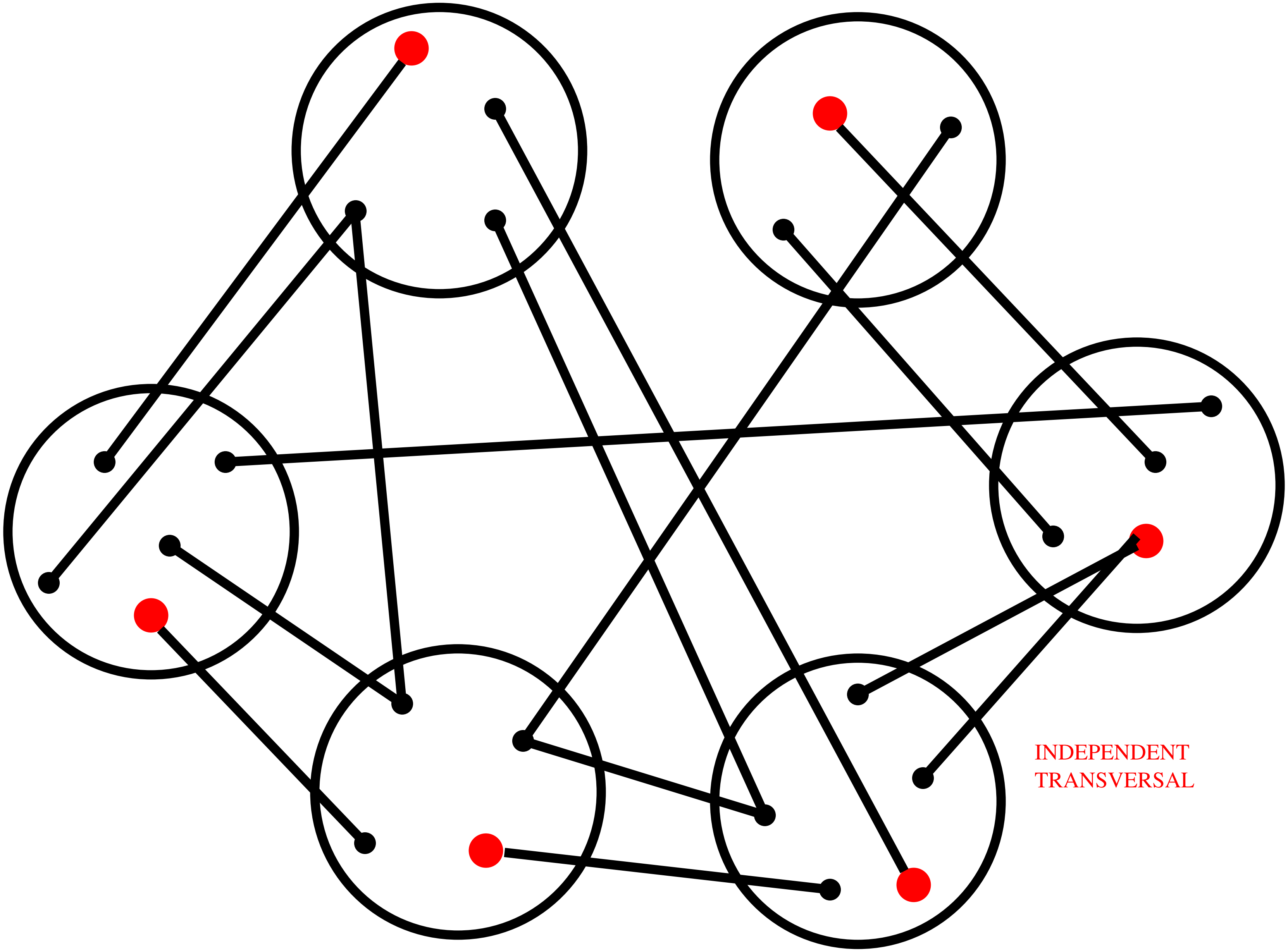
Independent transversals

Let G be a graph with a fixed partition of its vertex set.

An independent transversal in G is a subset M of the vertices such that

- no edge of G joins two vertices of M (independent)
- M contains exactly one vertex from each partition class (transversal)




INDEPENDENT
TRANSVERSAL

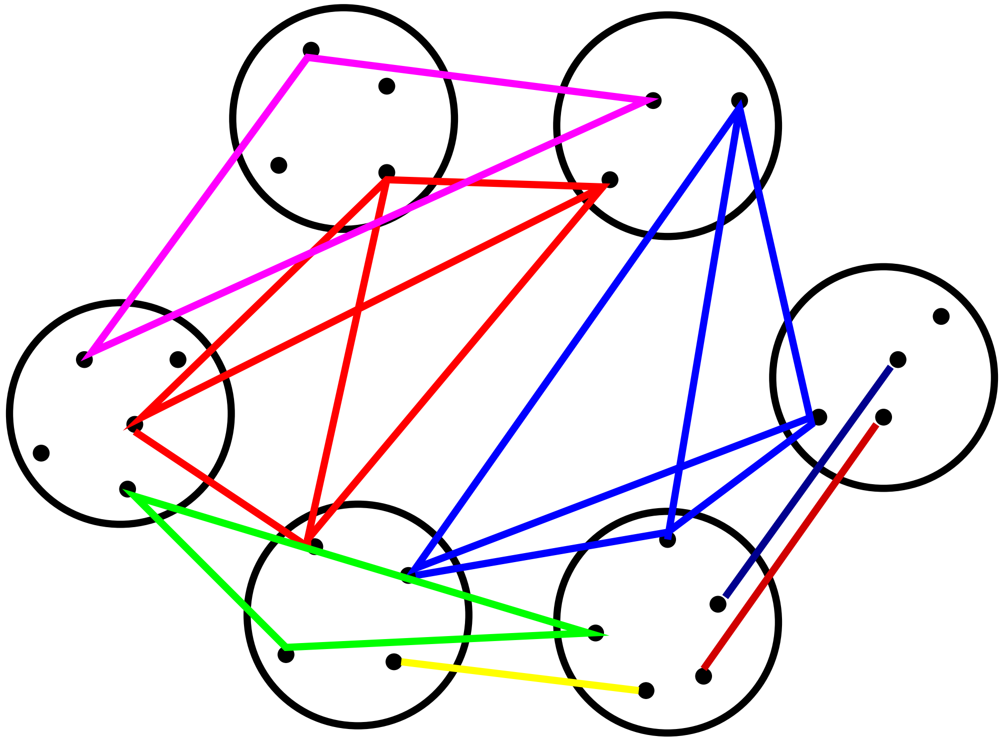
Independent Transversals

Many combinatorial problems can be formulated by asking whether a given graph with a given vertex partition has an independent transversal.

A simple example: matching in bipartite graphs.

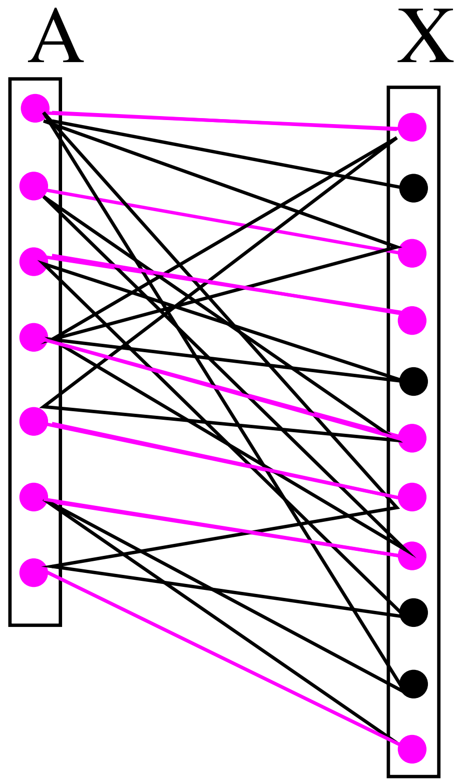
Let B be a bipartite graph with vertex classes A and X. Construct G:

- the set of partition classes is A,
- each edge ax of B is represented by a vertex x_a in the class of a,
- two vertices are adjacent if and only if they represent edges with a common vertex $x \in X$. Thus G is the union of a set of cliques, one for each $x \in X$.



Then an independent transversal corresponds to a set of disjoint edges in B that saturates A.

IN OTHER WORDS: an independent transversal in G corresponds to a complete matching from A to X in B.



Another example: the SAT problem

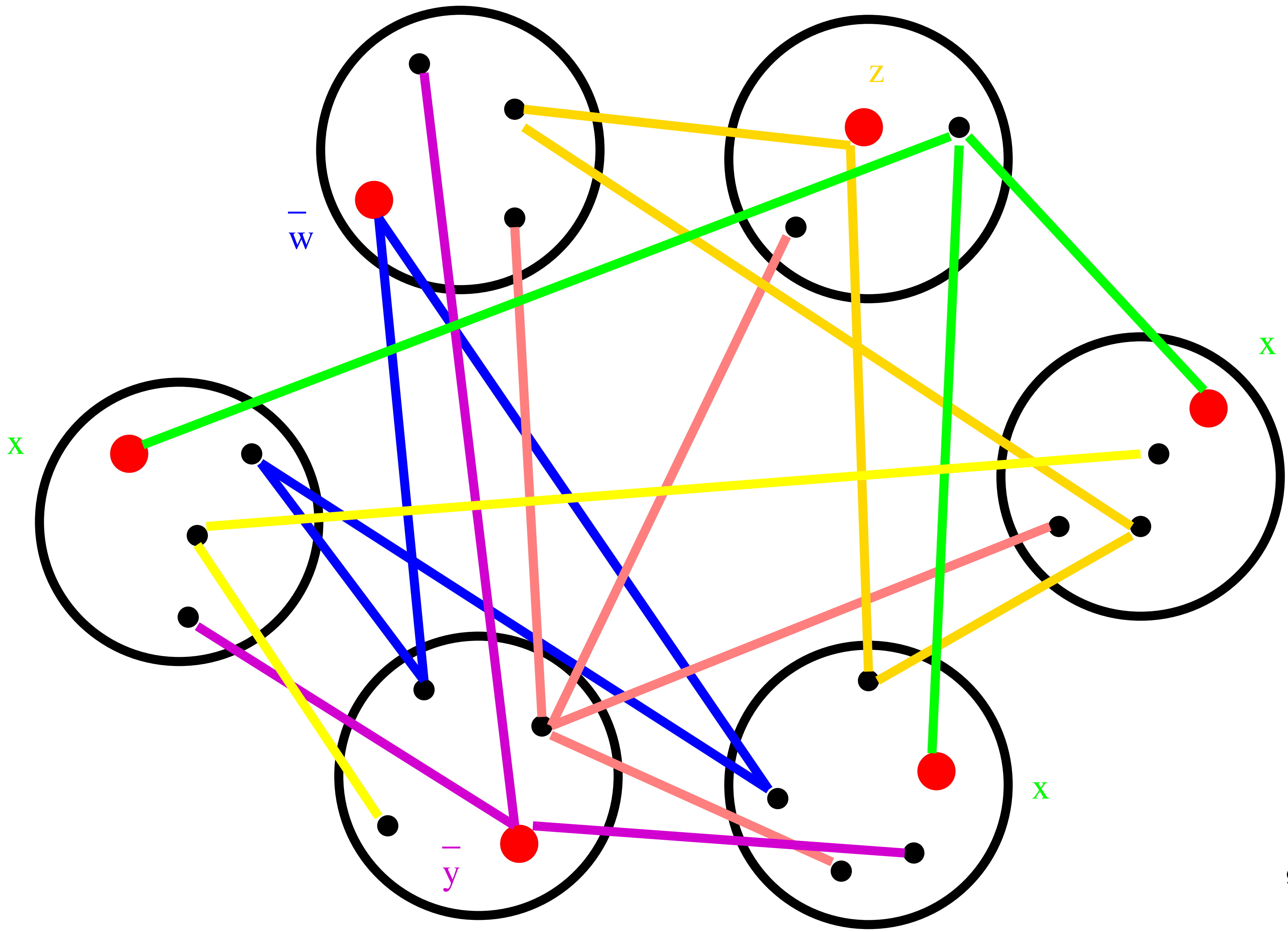
Given a Boolean formula, does it have a satisfying truth assignment?

$$(x_1 \vee \bar{x_4} \vee x_7) \wedge (\bar{x_1} \vee \bar{x_3} \vee \ x_2) \wedge (x_3 \vee \bar{x_2}) \wedge (x_5 \vee x_6 \vee \bar{x_2})$$

Construct G where

- Clauses correspond to partition classes
- Variables correspond to components, each of which is a complete bipartite graph intersecting each class at most once.

Then a satisfying truth assignment corresponds to an independent transversal.


z
$\bar{w}$
x
x
x
$\bar{y}$

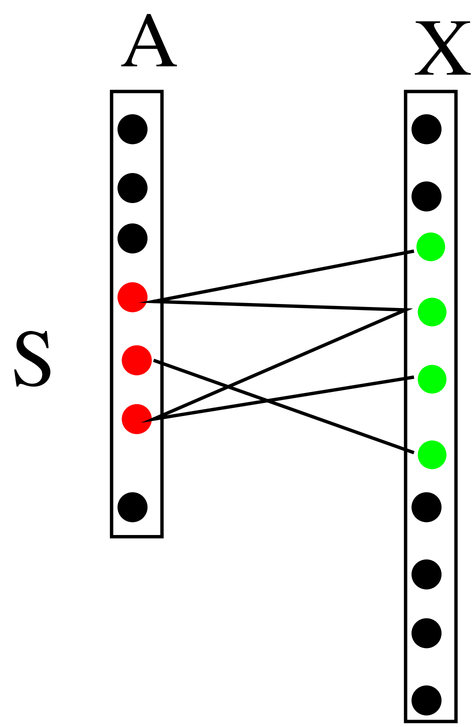
Thus we cannot expect an efficient characterisation of vertex-partitioned graphs with independent transversals.

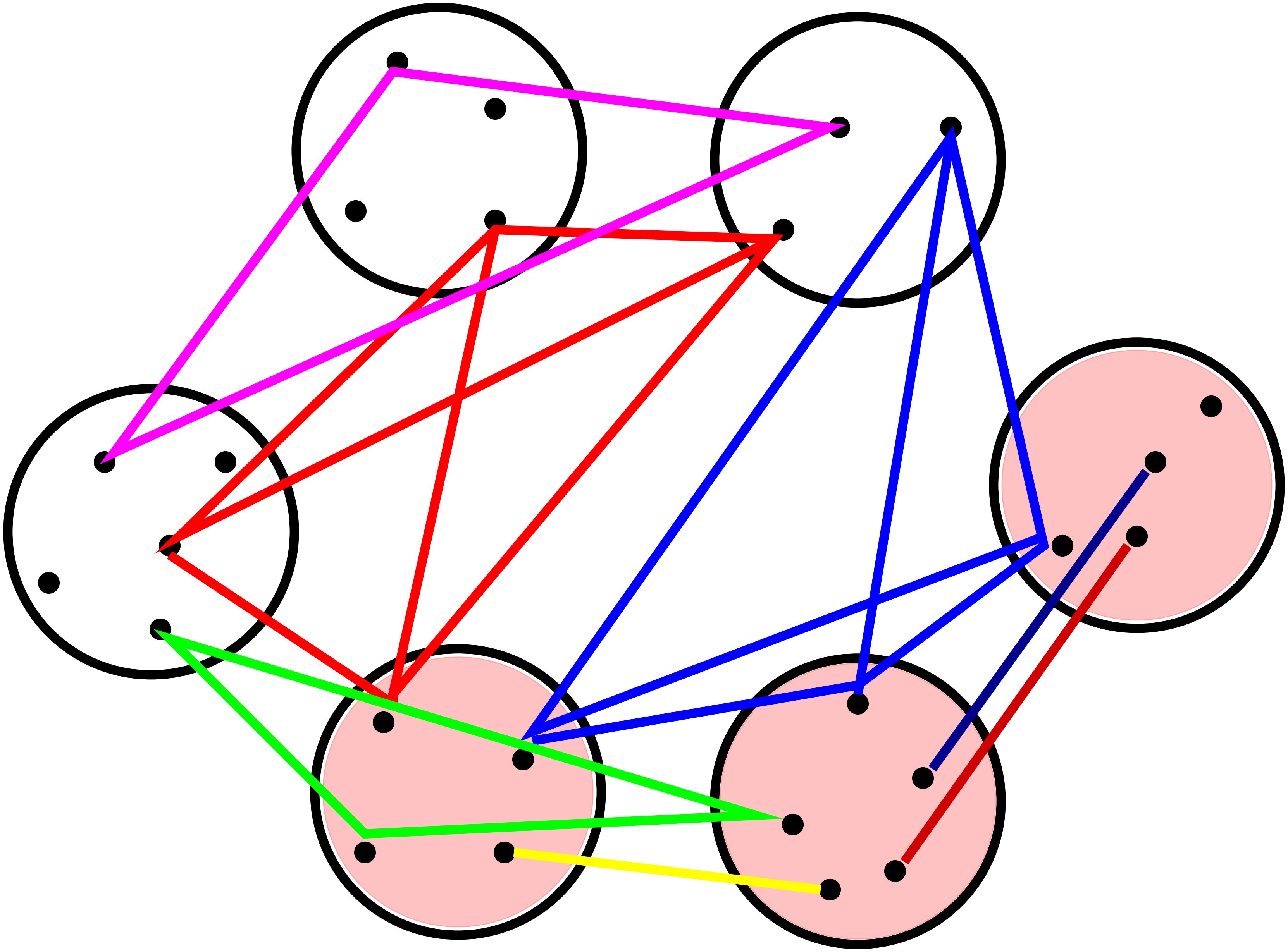
Instead we focus on sufficient conditions that guarantee the existence of an independent transversal.

Bipartite graphs

HALL'S THEOREM: The bipartite graph B has a complete matching if and only if: For every subset $S \subseteq A$, the neighbourhood $\Gamma(S)$ satisfies

$$|\Gamma(S)| \geq |S|.$$





When does an independent transversal exist?

In the case of matchings in bipartite graphs: when every subset S of partition classes contains representatives from at least $|S|$ cliques. (Hall's Theorem).

Moreover an independent transversal can be found efficiently if it exists.

IN OTHER WORDS: if there is NO independent transversal, then for some subset S of classes, the number of components of G_S is too small, i.e. at most $|S| - 1$.

Here G_S denotes the subgraph of G induced by $\cup_{U \in S} U$.

When does an independent transversal exist?

THEOREM 1 (PH): Let G be a vertex-partitioned graph, and suppose G does NOT have an independent transversal.

Then for some subset S of classes, the domination number of G_S is too small, i.e. at most $2(|S|-1)$.

Here the (strong) domination number of a graph H is the smallest size of a set D of vertices of H such that every vertex of H is adjacent to some $u \in D$.

Maximum Degree

In particular, the total number of vertices that a set D can dominate is at most

$$\sum_{u \in D} d(u) \leq \Delta(G)|D|.$$

So $2(|S|-1)$ vertices can dominate at most $2\Delta(G)(|S|-1)$ vertices.

Therefore we get the following.

COROLLARY 1: Let G be a vertex-partitioned graph. If each partition class has size at least $2\Delta(G)$ than G has an independent transversal.

Theorem 1

Theorem 1 has a topological proof (2001) based on the notion of topological connectedness and a combinatorial proof (1995) based on an alternating trees argument.

Corollary 1 is best possible: Szabó and Tardos (2003) gave an example for each d of a graph G with maximum degree d and $2d$ vertex classes, each of size $2d-1$ having NO independent transversal. The graph G itself is a union of disjoint complete bipartite graphs $K_{d,d}$.

Applications of Theorem 1

Corollary 1 answered a question first introduced and studied by Bollobás, Erdös and Szemerédi (1975), with later contributions by various others including Jin (1992), Yuster (1997), Alon (1988, 2002), Fellows (1990), Szabó and Tardos (2003), Bissacot, Fernández, Procacci and Scoppola (2011).

Theorem 1 has been used to solve problems in many areas, including graph theory (e.g. list colouring, strong colouring, delay edge colouring, circular colouring, graph partitioning problems, special independent set problems), hypergraphs (e.g. hypergraph matching, hypergraph packing and covering), group theory (e.g. generators in linear groups), ring theory, theoretical computer science (e.g. job scheduling, other resource allocation problems) etc.

Algorithms

The combinatorial proof of Theorem 1 gives an exponential algorithm for finding either an independent transversal in G, or an easily-dominated subset S, i.e. one where G_S has domination number at most $2(|S|-1)$.

Q: How can the assumptions of Theorem 1 be strengthened in order to give an efficient algorithm for finding either an independent transversal OR an easily-dominated set S of classes in G?

In particular: If G has maximum degree Δ, how big do the vertex classes need to be in terms of Δ in order to guarantee an efficient algorithm for finding an independent transversal?

Previous Work

Most of the work on this problem has been based on algorithmic versions of the Lovász Local Lemma (starting with Beck (1991) and including Moser (2009) and Moser and Tardos (2010) among many others). Algorithmic versions of Corollary 1 for class size linear in terms of Δ were first given by Alon (1988). Harris and Srinivasan (2017) (based on the work of Bissacot, Fernández, Procacci and Scoppola 2011 and Pegden 2014) gave a randomized algorithm that finds an IT in expected time $O(m\Delta)$ in graphs with class size 4Δ. The current best result for polynomial expected time is due to Harris (2016) who improved the bound on the class size to $4\Delta - 1$.

Deterministic algorithms based on derandomizing the Moser-Tardos algorithm have also been found (by e.g. Fischer and Ghaffari 2017, Harris 2018). These require the class size be $C\Delta$ for some large constant C in order to find an IT efficiently.

An Algorithmic Theorem

COROLLARY 1A: Let Δ be given. Then there is an algorithm that takes as input any graph G with maximum degree Δ and vertex classes of size $2\Delta + 1$, and finds an independent transversal in G in time polynomial in $|V(G)|$.

An Algorithmic Theorem

We will say that a vertex-partitioned graph G is r-claw-free with respect to the partition if no vertex of G has r independent neighbours in distinct vertex classes.

THEOREM 1A (PH, A. Graf): Let r and $\epsilon > 0$ be given. Then there is an algorithm that takes as input any vertex-partitioned r-claw-free graph G, and finds in time polynomial in $|V(G)|$ either

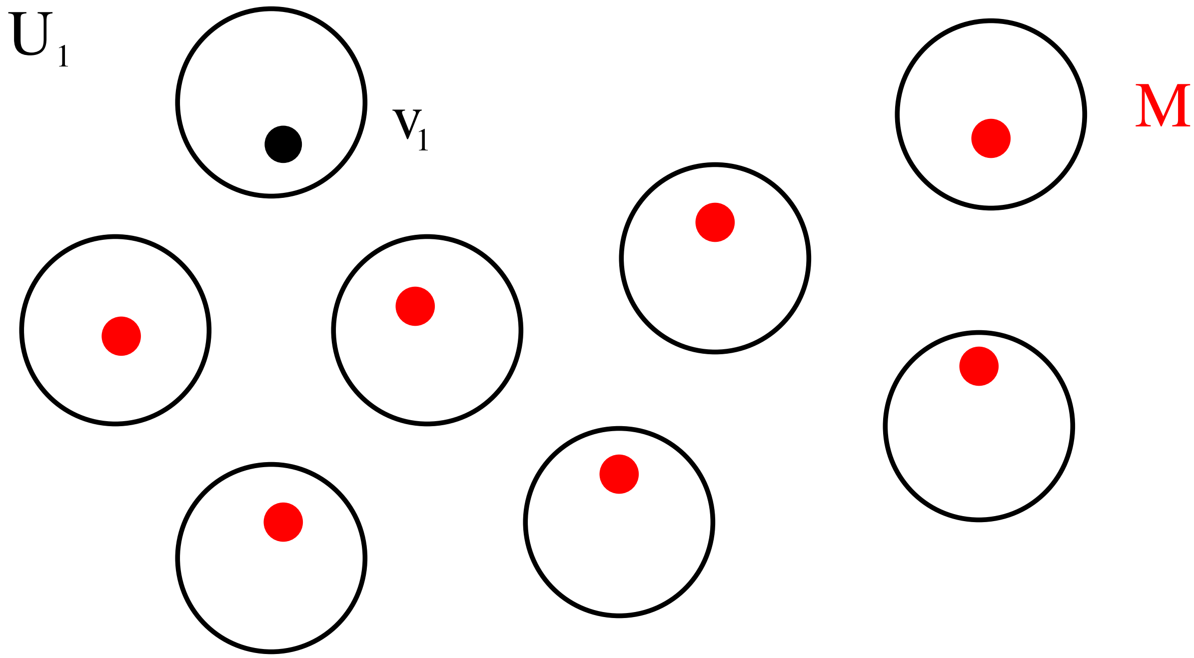
- an independent transversal in G, or
- a subset S of vertex classes, together with a set D that dominates G_S, where

$$|D| < (2+\epsilon)(|S|-1).$$

Taking $r = \Delta + 1$ and $\epsilon = 1/\Delta$ gives Corollary 1A.

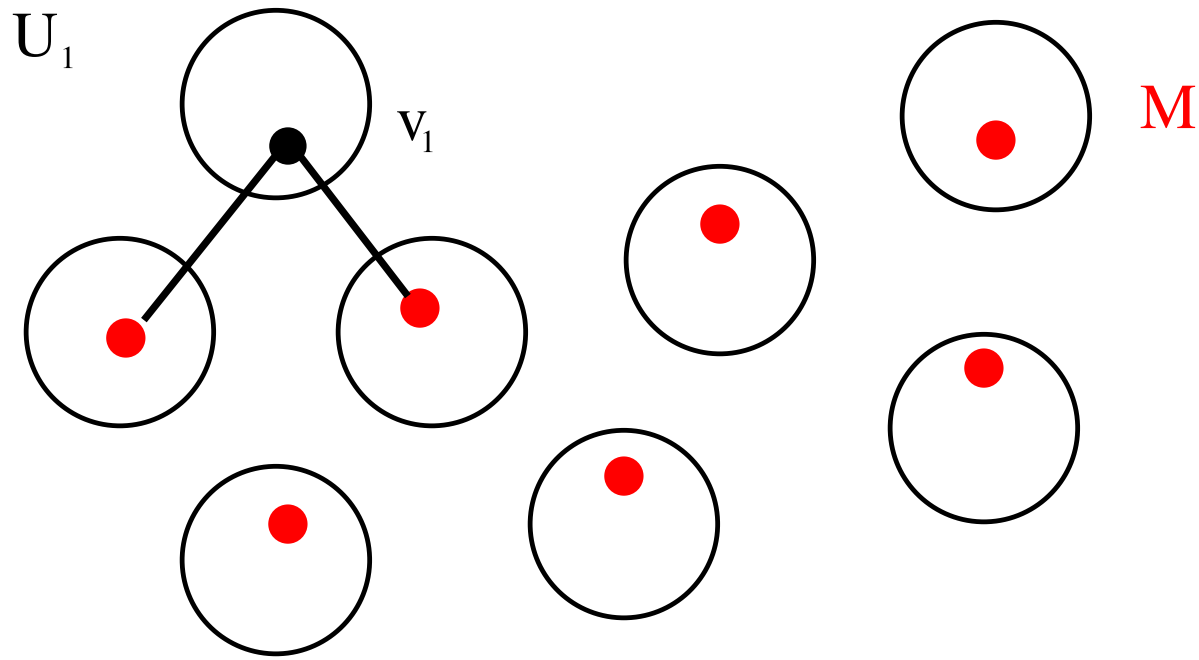
Proof Ideas

Original proof idea of Theorem 1: Start with a partial independent transversal M and a class U_1 such that $U_1 \cap M = \emptyset$.

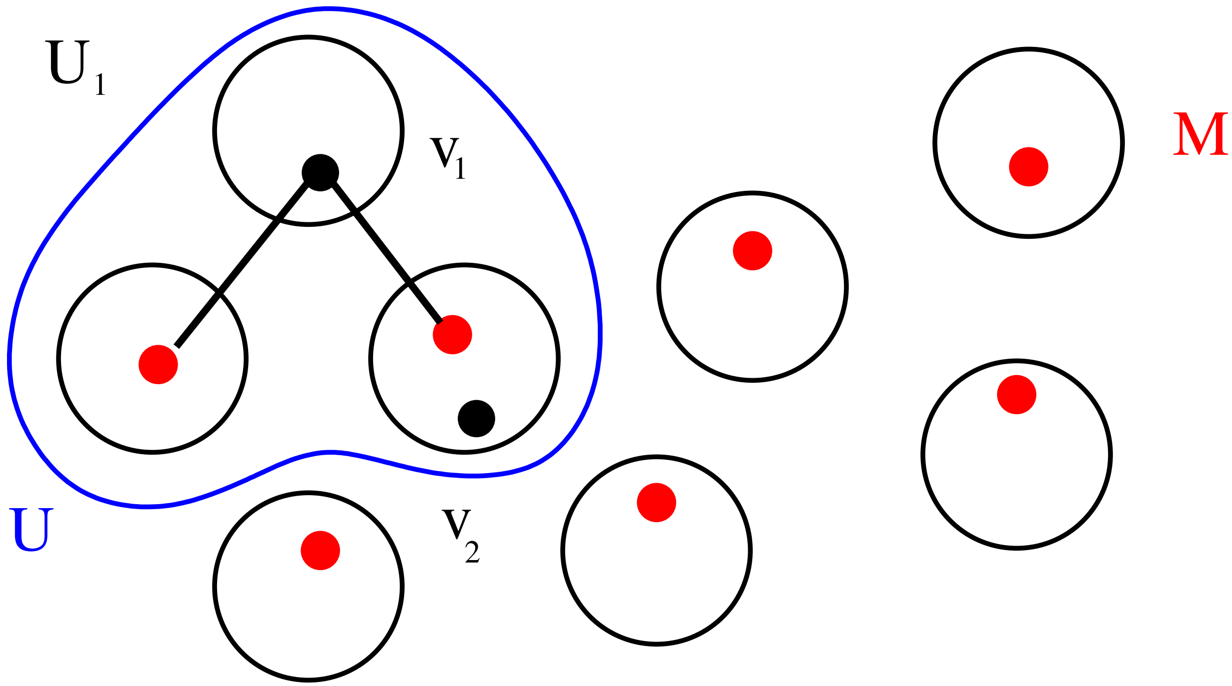


Build a tree T of classes and vertices as follows: Choose $v_1 \in U_1$ and set $T = \{v_1\}$. If $d_M(v_1) = 0$ then improve M by adding v_1.

Otherwise add the M-neighbourhood $\Gamma_M(v_1)$ of v_1 to T.

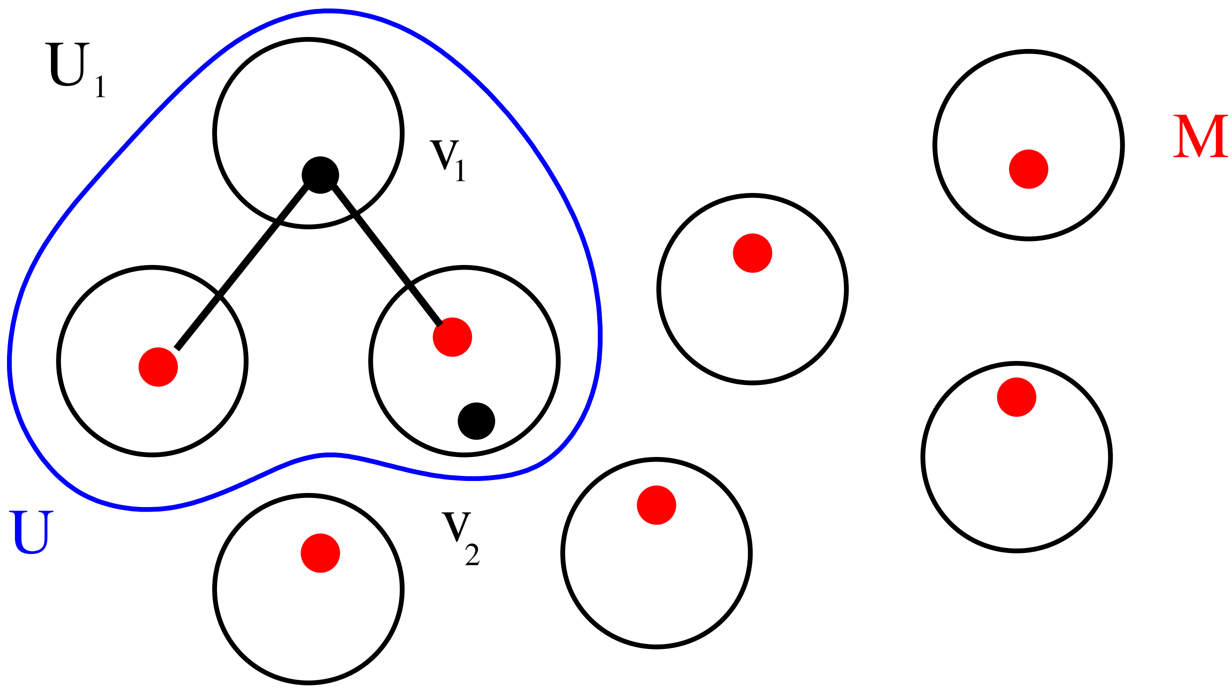


General step: let $U = U(T)$ be the set of all classes containing vertices in T.



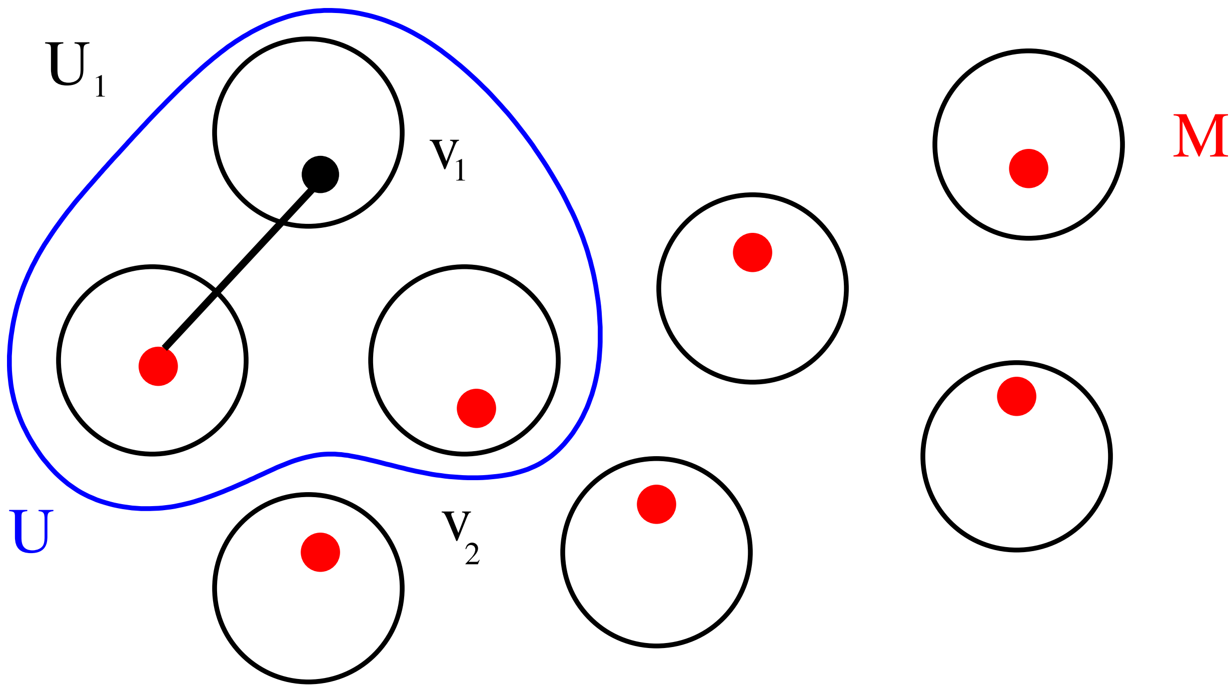
ASK: Does T dominate G_U?

IF YES: then we have found an easily-dominated set $S = U$, since $|T| \leq 2(|U| - 1)$. STOP.



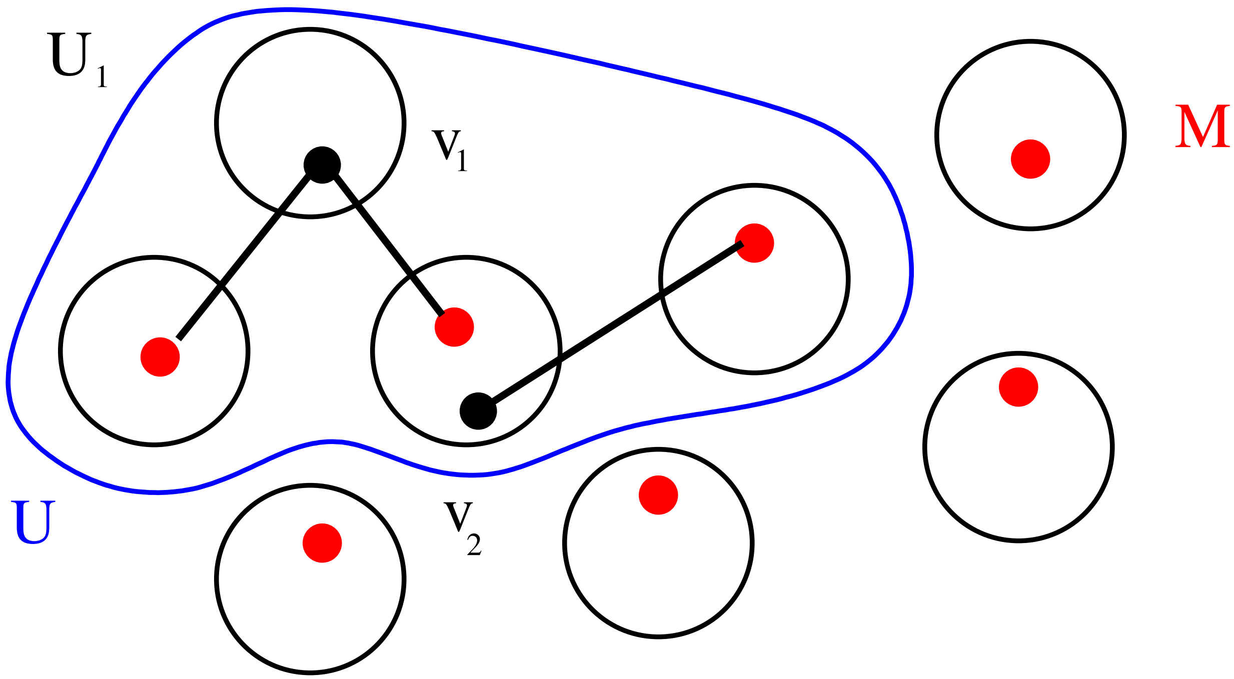
IF NO: then there exists a vertex of G_U that is not adjacent to any vertex in T. Choose such a vertex v_i and add it to T.

If $d_M(v_i) = 0$ then improve M by adding v_i to M and removing the M-vertex in the class of v_i.

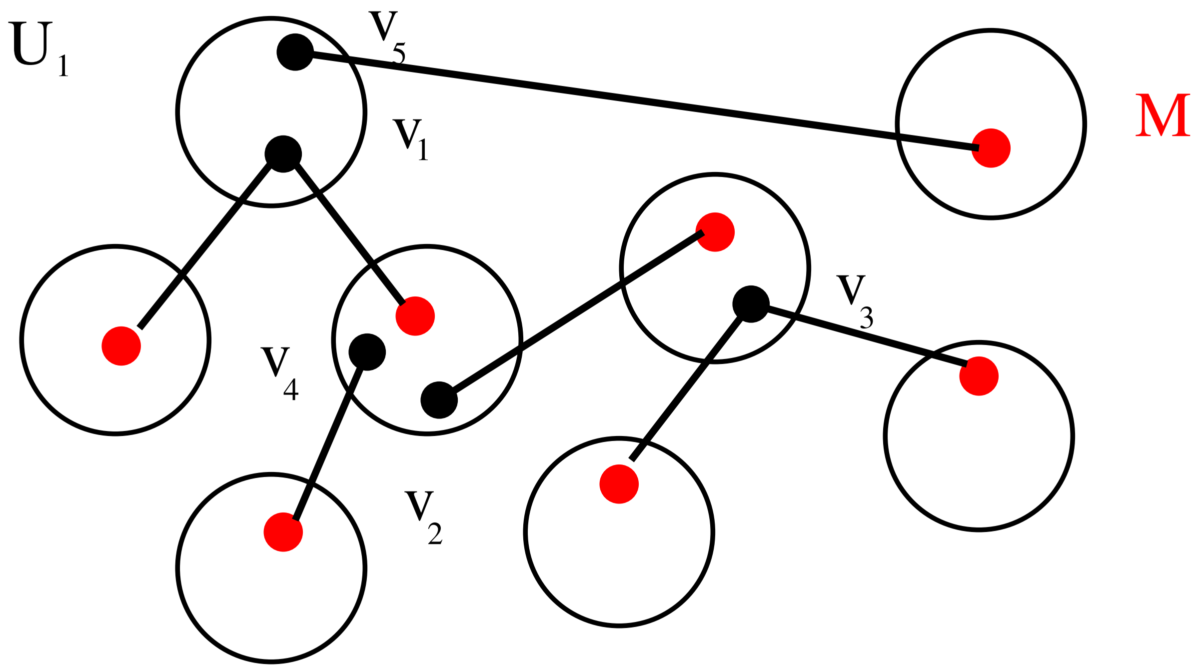


Here improve means lower the degree $d_M(v_j)$ of an earlier vertex v_j.

Otherwise add v_i and $\Gamma_M(v_i)$ to T.



At each step we (a) find a subset S of classes such that T dominates G_S where $|T| \leq 2(|S|-1)$ and STOP, OR (b) grow the tree, OR (c) reduce $d_M(v_i)$ for some i UNTIL M can be altered/extended to include a new vertex.



Analysis

Progress is measured by a signature vector

$$(d_M(v_1), \ldots, d_M(v_t), N, \ldots, N)$$

of length $m-1$ (where N is larger than any degree).

Each step (unless it terminates with an easily-dominated set S) reduces the lexicographic order of the signature vector.

Thus the procedure terminates by finding either an easily-dominated set S or a larger partial independent transversal.

(If G has maximum degree d then the number of steps is $O(d^m)$.)

Proof Ideas 2: Algorithmic version

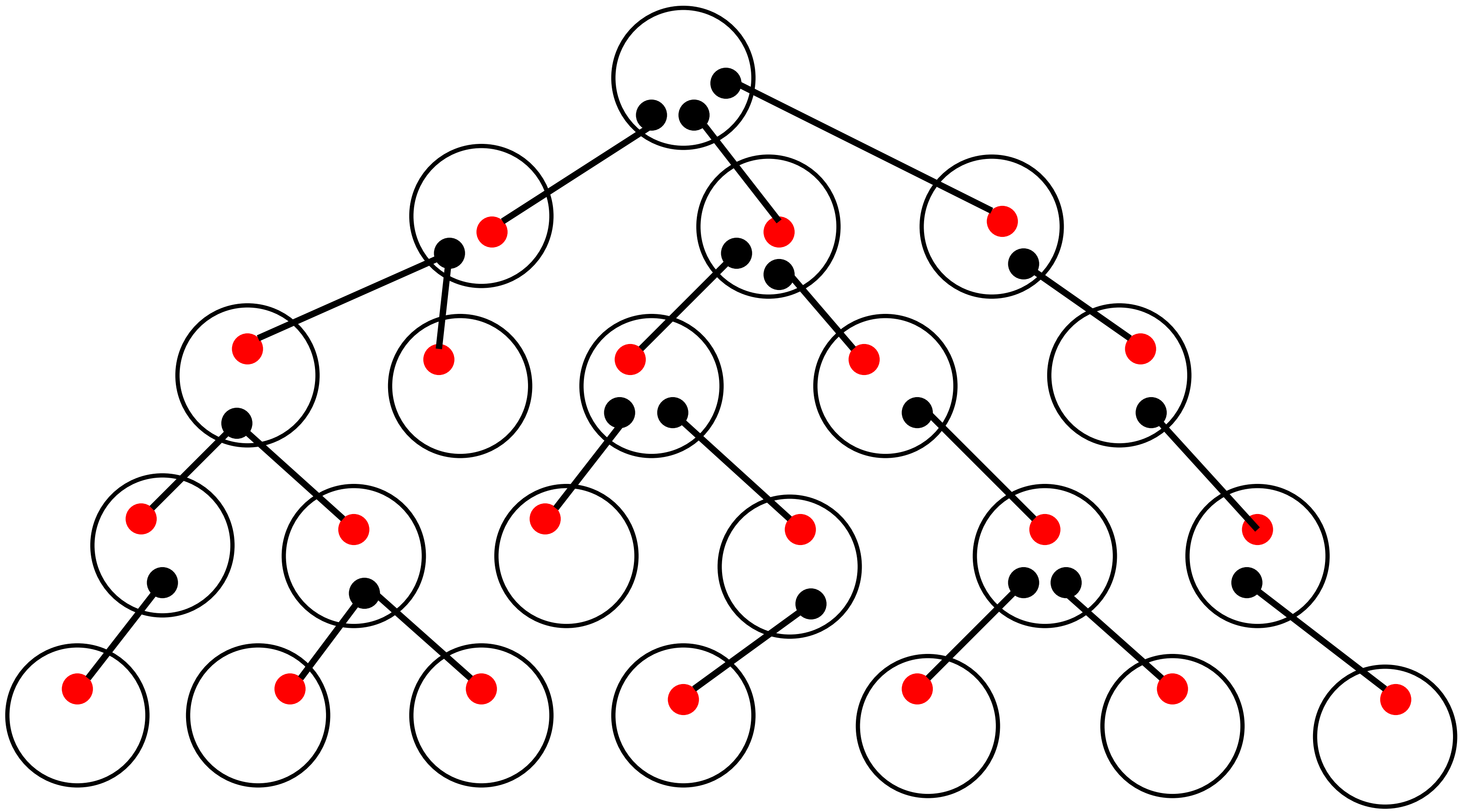
Basic idea similar to combinatorial proof but with modifications. (Several key ideas from C. Annamalai: Matching algorithm for bipartite hypergraphs.)

1. maintain layers

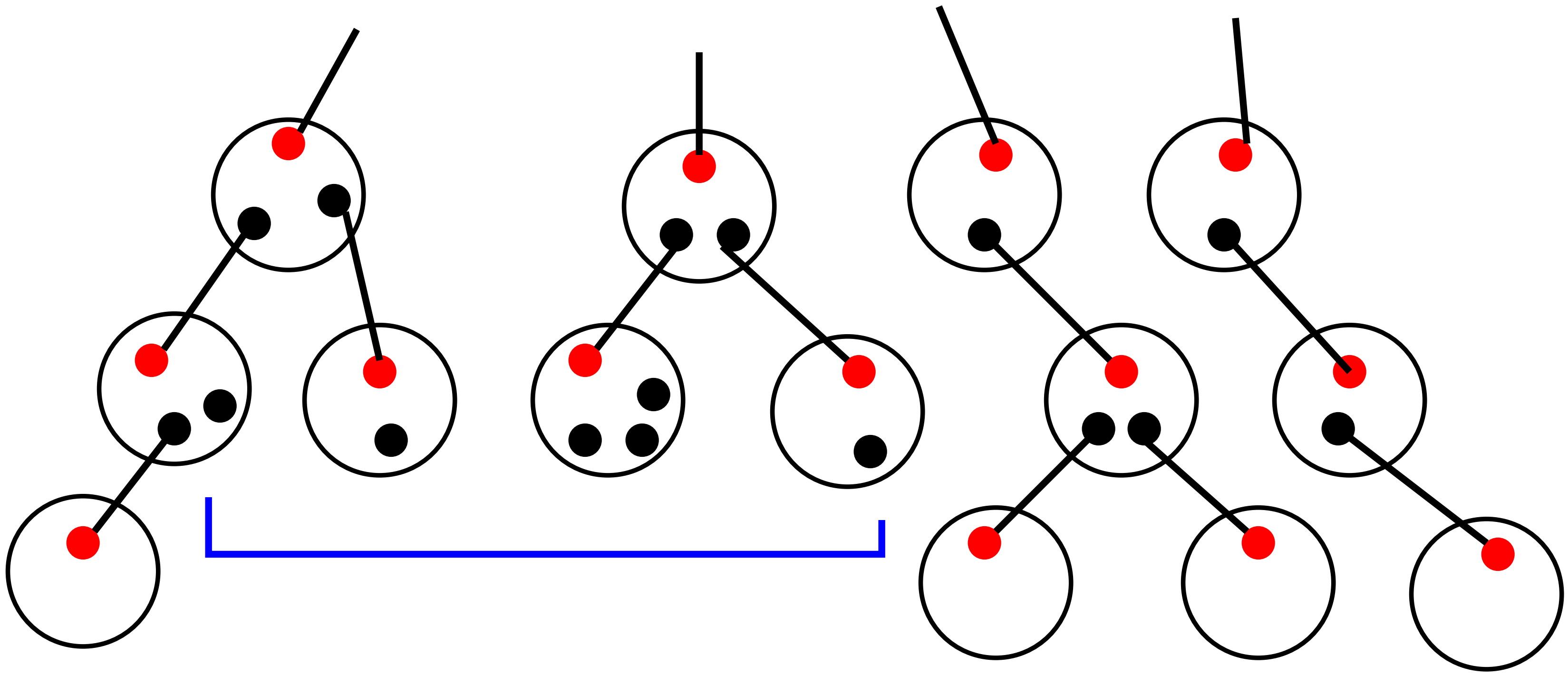
2. update in "clumps": only when at least a positive proportion μ of a layer has $d_M(x) = 0$ (here μ is of order ϵ/r). Discard later layers

3. rebuild layers in "clumps": after an update, rebuild a layer only if it adds a μ proportion of that layer. Discard later layers

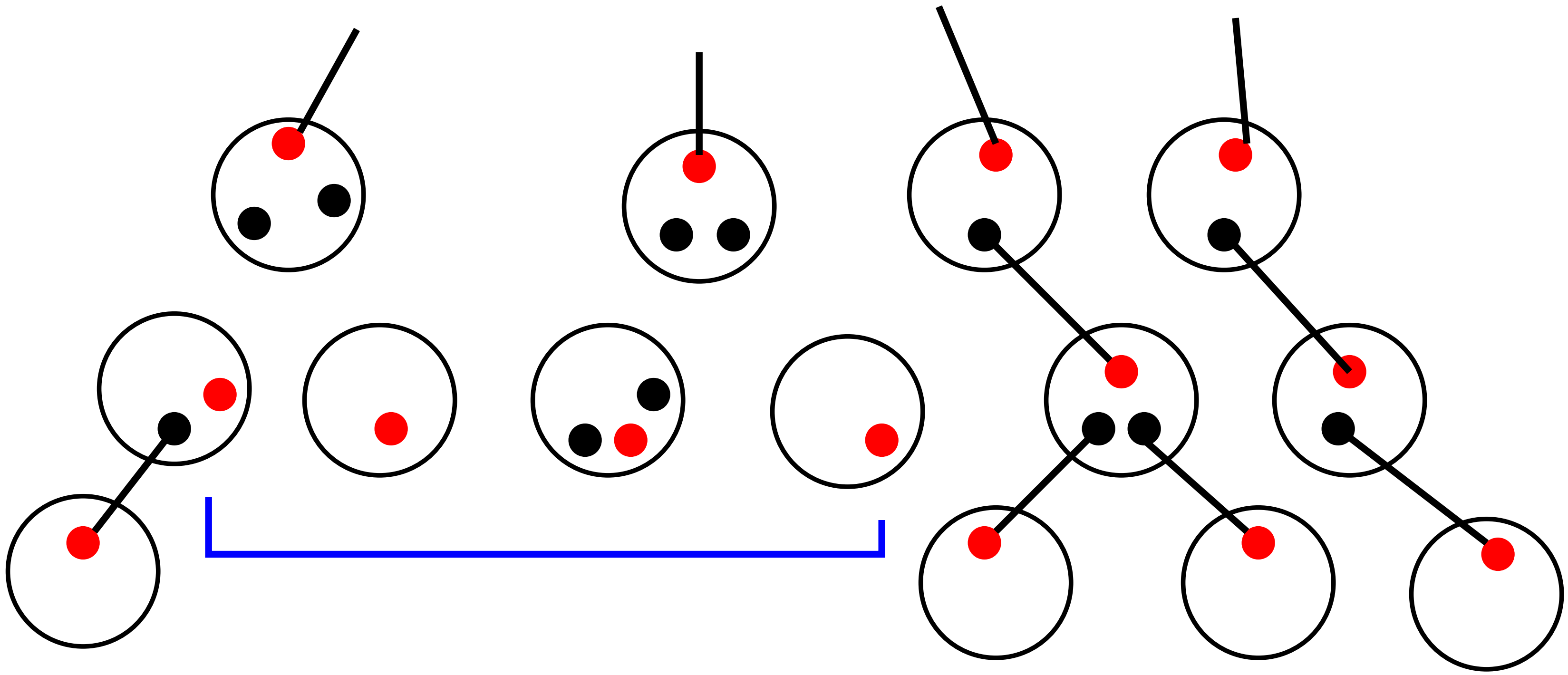
Maintain layers



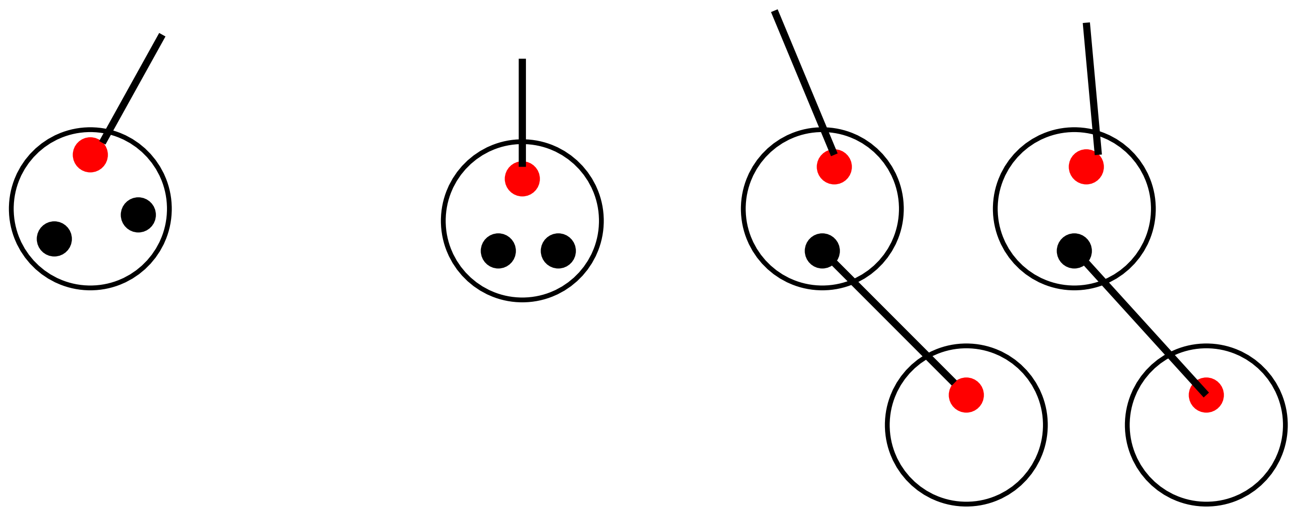
Update in clumps



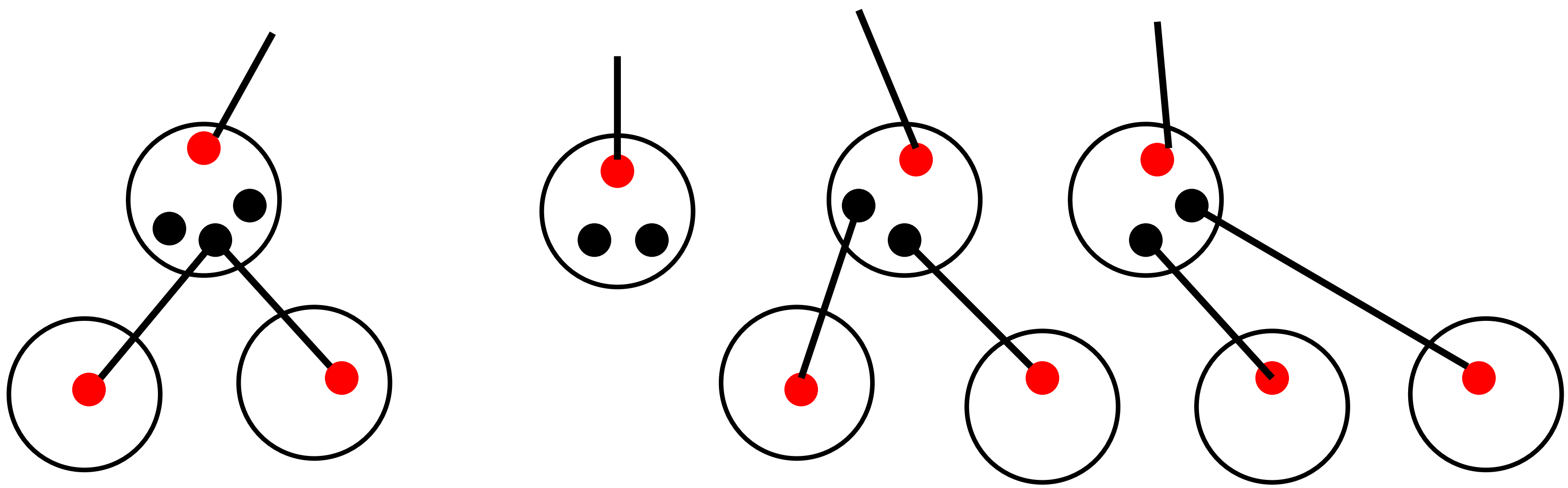
Update in clumps



Update in clumps



Rebuild layers in clumps

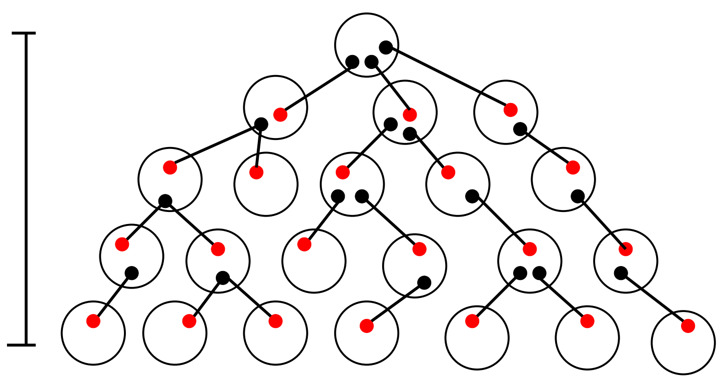


Consequences

Maintaining layers has the effect of "pushing" undominated vertices towards the bottom layer. Unless the algorithm terminates by finding a (quite) easily-dominated set S they constitute a positive proportion of ALL vertices in classes.

If there is no update, then the next layer has size a positive proportion of ALL classes currently in the construction.

This implies that the tree of classes has logarithmic (in m) depth.



Updating and rebuilding layers in clumps allow a signature vector that measures sizes of layers rather than degrees of individual vertices.

Each growth of layer (basic or type (3)) or update of M step either

- increases the number of non-M vertices in the layer substantially, or
- decreases the number of M-vertices substantially.

Here substantially means by a positive proportion. Hence progress can be measured by an integer increase in a log function.

Signature vector

The signature vector has just 2 entries per layer, each a log. One measures number of non-M vertices in the layer, the other the number of M-vertices.

The length of the signature vector is the number of layers (times 2), also a log function.

It can be shown (by an appropriate shift, the coordinates of each signature vector form an increasing sequence. Thus by associating each vector to a subset of a set of size $O(\log m)$, it follows) that the number of possible signature vectors is polynomial in m. (The degree depends on ϵ and r.)

Why r-claw-free?

Rebuilding layers in clumps means that there can exist a few vertices that are not dominated by T i.e. not enough to trigger a rebuilding step.

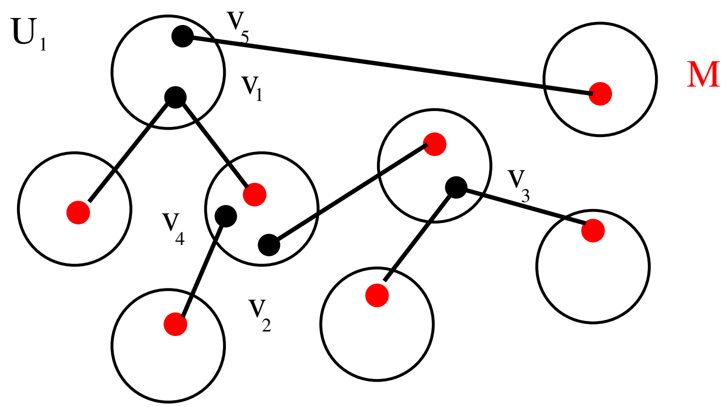
These vertices plus their neighbours in M must be added to the dominating set D.

Their number is small but without the r-claw-free condition their number of M-neighbours could be large.

Form of the dominating set

It follows from the proof of Theorem 1 that if G does not have an IT then G contains a subset S of vertex classes where the set T dominating S has a special form. This is important for some applications.

DEF: Let S be a set of vertex classes in a vertex-partitioned graph G. A constellation for S is an induced subgraph T of G_S, whose components are stars with at least two vertices, each with a centre and a nonempty set of leaves distinct from its centre. The set of all leaves of T forms an IT of $|S|-1$ vertex classes of S. Then $|V(T)| \leq 2(|S|-1)$.



A stronger version

THEOREM 1A: Let r and $\epsilon > 0$ be given. Then there is an algorithm that takes as input any vertex-partitioned r-claw-free graph G, and finds in time polynomial in $|V(G)|$ either

- an independent transversal in G, or
- a subset S of vertex classes, together with a set D that dominates G_S, where
$$|D| < (2+\epsilon)(|S|-1).$$
Moreover D contains $V(T)$ for a constellation T of some $S_0 \supseteq S$, where $|D \setminus V(T)| < \epsilon(|S|-1)$.

Applications

When the maximum degree $\Delta(G)$ and the clique number $\omega(G)$ of a graph G are close enough, the graph contains an independent set meeting all maximum cliques. Finding such a set is important for various colouring problems.

THEOREM (King 2011): Let G be a graph of maximum degree Δ such that $\omega(G) > \frac{2}{3}(\Delta + 1)$. Then G contains an independent set meeting every maximum clique.

This result is best possible as can be seen by replacing each vertex of a 5-cycle by a clique. The proof uses a modification of Theorem 1.

Algorithmic version

THEOREM: Let Δ be a positive integer. There exists an algorithm that finds, in time polynomial in $|V(G)|$, an independent set meeting every maximum clique in any graph G with maximum degree Δ satisfying $\omega(G) > 2(\Delta + 1)/3$.

This is a consequence of Theorem 1A using the additional information about the small dominating sets containing constellations, that is given by the stronger version.

Applications

A circular p/q-edge-colouring of a graph G is a colouring of $E(G)$ with colours $\{0, \ldots, p-1\}$ such that adjacent edges get colours that differ (mod p) by at least q. The circular chromatic index of G is the smallest ratio p/q for which there is a circular p/q-edge-colouring of G.

It is known that this parameter lies between $\chi'(G)-1$ and $\chi'(G)$ for every graph G.

Motivated by the Girth Conjecture for snarks, Kaiser, Král, and Škrekovski proved that if a cubic bridgeless graph has large enough girth then it is "almost" 3-edge-colourable in terms of circular colouring.

Circular edge colourings

THEOREM (Kaiser, Král, Škrekovski 2004): For $p \geq 2$, let G be a cubic bridgeless graph with girth

$$g = \begin{cases} 2(2p)^{2p-2} & \text{if } p \geq 2 \text{ is even} \\ 2(2p)^{2p} & \text{if } p \geq 3 \text{ is odd.} \end{cases}$$

Then G admits a circular $(3p+1)/p$-edge-colouring.

The proof applied Theorem 1 to establish the existence of an independent transversal in a certain auxiliary graph constructed using G, p, and a fixed 1-factor F of G. Then this was used to construct an explicit circular edge colouring.

Algorithmic version

THEOREM: For each $p \geq 2$, there exists an algorithm that takes as input any cubic bridgeless graph G with girth

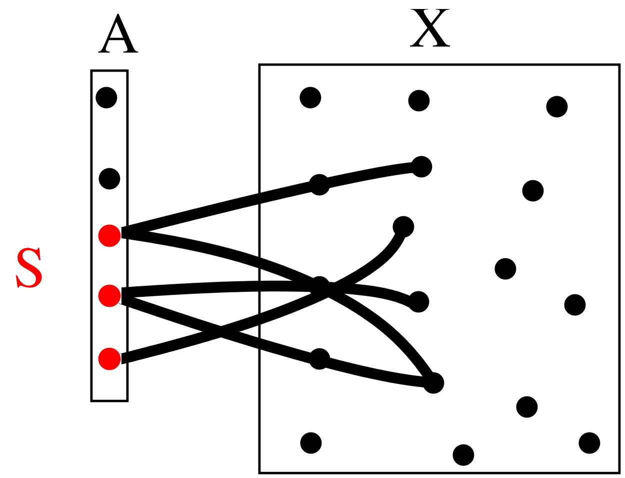
$$g = \begin{cases} 2(2p)^{2p-2} & \text{if } p \geq 2 \text{ is even} \\ 2(2p)^{2p} & \text{if } p \geq 3 \text{ is odd} \end{cases}$$

and finds, in time polynomial in $|V(G)|$, a circular $(3p+1)/p$-edge-colouring of G.

In these two applications, the ϵ error term in Theorem 1A does not introduce any weakening of the original non-algorithmic results. This is for reasons of divisibility.

Applications

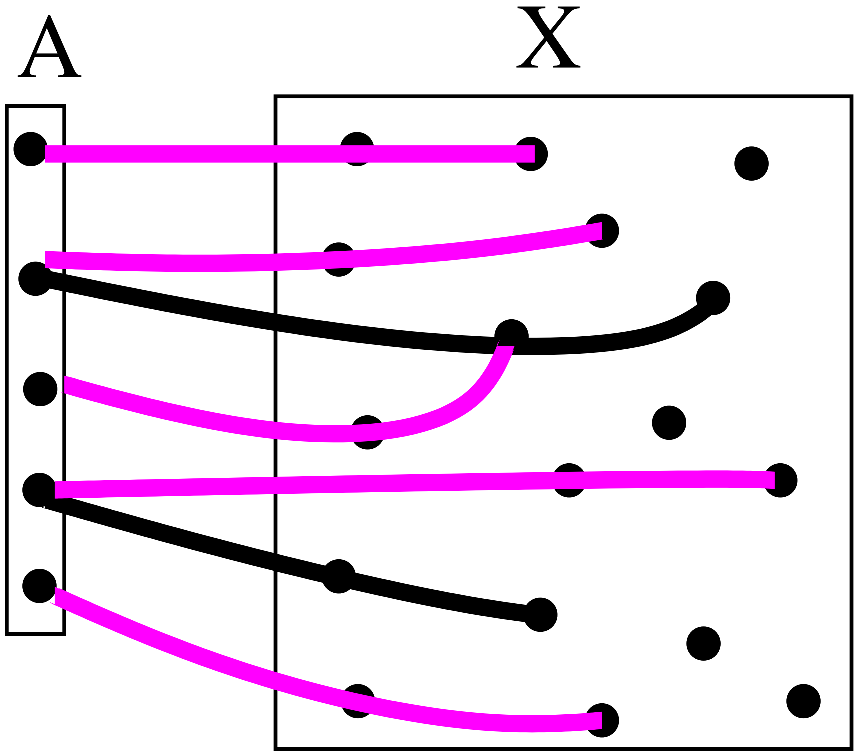
Let H be an r-uniform bipartite hypergraph.



The link of a subset $S \subseteq A$ is an $(r-1)$-uniform hypergraph.

The line graph of the link of A is r-claw-free, and has a natural vertex partition given by the elements of A.

When does H have a complete matching?



This is an independent transversal in the line graph of the link of A.

For a set $S \subseteq A$, write $\tau_X(S)$ for the smallest cardinality of an X-cover of the link of S, that is, a subset $W \subseteq X$ that meets every edge in the link of S.

A dominating set of size at most $2(|S|-1)$ in G_S that is a constellation for S corresponds to an X-cover of the link of S of size at most $(2r-3)(|S|-1)$. Thus Theorem 1 implies the following.

THEOREM (PH 1995): Let H be an r-uniform bipartite hypergraph. If

$$\tau_X(S) > (2r-3)(|S|-1)$$

for every $S \subseteq A$, then H has a complete matching.

This is best possible for every r. Note that when $r = 2$ it is (the nontrivial direction of) Hall's Theorem.

Algorithmic version

Then the stronger version of Theorem 1A implies the following result of Annamalai (2017).

THEOREM: For every $r \geq 2$ and $\epsilon > 0$, there exists an algorithm that finds, in time polynomial in the size of the input, a complete matching in r-uniform bipartite hypergraphs H satisfying

$$\tau_X(S) > (2r - 3 + \epsilon)(|S| - 1)$$

for all $S \subseteq A$.

Remarks and Open Problems

- The exponent of $|V(G)|$ in Theorem 1A depends on $1/\epsilon$ and r. While the dependence on $1/\epsilon$ seems essential, we do not know whether the dependence on r could be substantially weakened or possibly even removed altogether.

- Aharoni, Berger and Ziv proved a version of Theorem 1 for vertex-weighted graphs. Correspondingly there is an algorithmic weighted version of Theorem 1A.

- Theorem 1 also has a topological proof. There are other results giving sufficient conditions for the existence of an IT for which only a topological proof is known. Currently there are no algorithmic versions of these topological results.