The 1-2-3 Conjecture and its relatives on graphs and hypergraphs

Florian Pfender

UC Denver

joint work with M. Kalkowski and M. Karoński

October 15, 2013

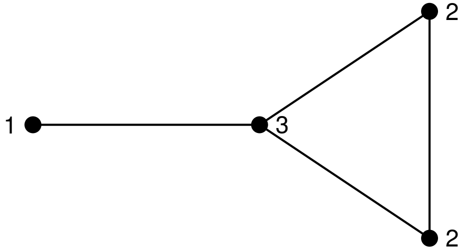
Irregularity strength

The 1-2-3 Conjecture

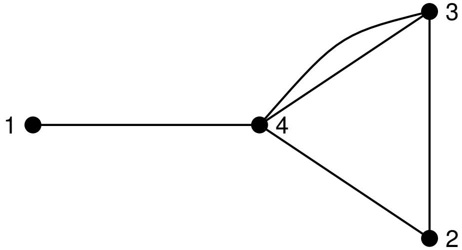
Related questions on graphs

Hypergraphs

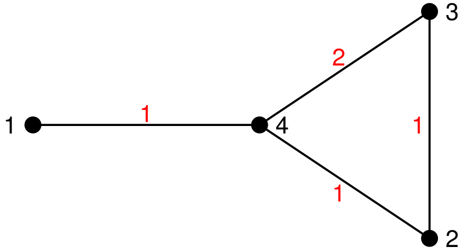
Irregular assignments



Irregular assignments



Irregular assignments



Irregular assignments

Given a simple graph G, and an assignment

$$w : E(G) \to \{1, ..., k\} = [k],$$

define the weighted degree

$$w(v) = \sum_{u \in N(v)} w(uv).$$

Irregular assignments

Given a simple graph G, and an assignment

$$w : E(G) \to \{1, ..., k\} = [k],$$

define the weighted degree

$$w(v) = \sum_{u \in N(v)} w(uv).$$

An assignment is irregular if all the weighted vertex degrees are different.

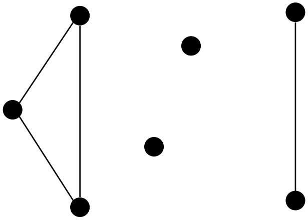
Irregularity strength

The irregularity strength $s(G)$, is the maximal weight k, minimized over all irregular weight assignments, and is set to ∞ if no such assignment is possible.

Irregularity strength

The irregularity strength $s(G)$, is the maximal weight k, minimized over all irregular weight assignments, and is set to ∞ if no such assignment is possible.

- Clearly, $s(G) < \infty$ if and only if G contains no isolated edges and at most one isolated vertex, i.e., when G is “nice”.



Lower bounds on $s(G)$

$$s(G) \geq \lambda(G) := \max_j \left(\frac{1}{j} \sum_{i=1}^{j} n_i \right),$$

where n_i is the number of vertices of degree i

Lower bounds on $s(G)$

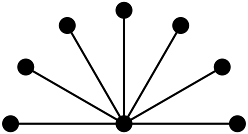
$$s(G) \geq \lambda(G) := \max_j \left(\frac{1}{j} \sum_{i=1}^{j} n_i \right),$$

where n_i is the number of vertices of degree i

d-regular graphs: $\lambda(G) = \frac{n+d-1}{d}$

Upper bounds on $s(G)$

$s(G) \leq n - 1$ for all (nice) graphs *Nierhoff (2000)*



Upper bounds on $s(G)$ involving δ

$s(G) \leq 60n/\delta$ for graphs with maximum degree $\Delta \leq \sqrt{n}$,
$s(G) \leq (c \log n)n/\delta$ *Frieze, Gould, Karoński and P. (2002)*

$s(G) \leq 48n/\delta$ for large δ *Cuckler and Lazebnik (2008)*

Upper bounds on $s(G)$ involving δ

$s(G) \leq 60n/\delta$ for graphs with maximum degree $\Delta \leq \sqrt{n}$,
$s(G) \leq (c \log n)n/\delta$ *Frieze, Gould, Karoński and P. (2002)*

$s(G) \leq 48n/\delta$ for large δ *Cuckler and Lazebnik (2008)*

$s(G) \leq 112n/\delta$,
$s(G) \leq 16n/d$ for d-regular graphs *Przybyło (2008)*

Upper bounds on $s(G)$ involving δ

$s(G) \leq 60n/\delta$ for graphs with maximum degree $\Delta \leq \sqrt{n}$,
$s(G) \leq (c \log n)n/\delta$ *Frieze, Gould, Karoński and P. (2002)*

$s(G) \leq 48n/\delta$ for large δ *Cuckler and Lazebnik (2008)*

$s(G) \leq 112n/\delta$,
$s(G) \leq 16n/d$ for d-regular graphs *Przybyło (2008)*

$s(G) \leq 6\lceil n/\delta \rceil$ *Kalkowski, Karoński and P. (2010)*

Neighbor-distinguishing assignments

- Given a simple nice graph G of order n, and an assignment $w : E(G) \to [k]$ of positive integers weights to the edges of G.

Neighbor-distinguishing assignments

- Given a simple nice graph G of order n, and an assignment $w : E(G) \to [k]$ of positive integers weights to the edges of G.
- An assignment is neighbor-distinguishing if the weighted degrees of adjacent vertices are different.

Neighbor-distinguishing assignments

- Given a simple nice graph G of order n, and an assignment $w : E(G) \to [k]$ of positive integers weights to the edges of G.
- An assignment is neighbor-distinguishing if the weighted degrees of adjacent vertices are different.

neighbor-distinguishing $\Longleftrightarrow$ **proper vertex coloring**

Neighbor-distinguishing assignments

- Given a simple nice graph G of order n, and an assignment $w : E(G) \to [k]$ of positive integers weights to the edges of G.
- An assignment is neighbor-distinguishing if the weighted degrees of adjacent vertices are different.

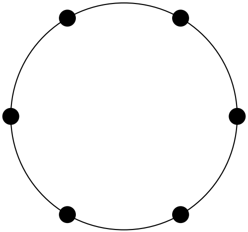
neighbor-distinguishing $\Longleftrightarrow$ **proper vertex coloring**

Conjecture (Karoński, Łuczak and Thomason, 2002)

It is possible to weight the edges of any (nice) graph with the integers $1, 2, 3$ such that the resultant vertex weighting is a proper coloring.

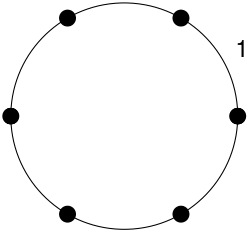
$\{1, 2\}$ is not enough

Cycles of length not divisible by 4



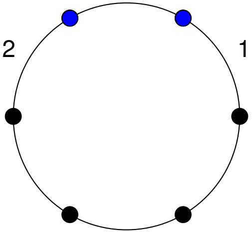
$\{1, 2\}$ is not enough

Cycles of length not divisible by 4



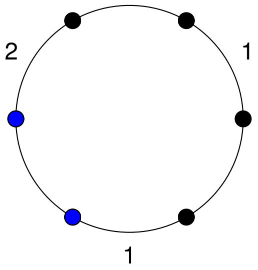
$\{1, 2\}$ is not enough

Cycles of length not divisible by 4



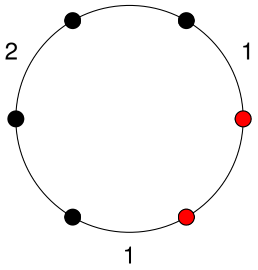
$\{1, 2\}$ is not enough

Cycles of length not divisible by 4



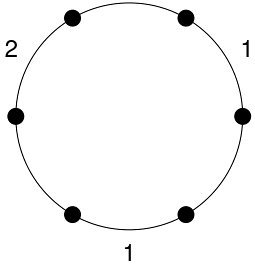
$\{1, 2\}$ is not enough

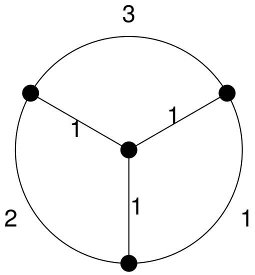
Cycles of length not divisible by 4



$\{1, 2\}$ is not enough

Cycles of length not divisible by 4





Complete graphs

The 1,2,3 - Conjecture

The 1,2,3 - Conjecture is true

- for all non-trivial graphs on $n \leq 11$ vertices
- for all 3-colorable graphs

Theorem (Łuczak, Thomason, Karoński, 2002)

Let Γ be a finite abelian group of odd order and let G be a non-trivial $|\Gamma|$-colorable graph. Then there is a weighting of the edges of G with the elements of Γ such that the resultant vertex weighting is a proper coloring.

The 1,2,3 - Conjecture

The 1,2,3 - Conjecture is true

- for all non-trivial graphs on $n \leq 11$ vertices
- for all 3-colorable graphs
- for random graphs

Theorem (Addario-Berry, Dalal, Reed, 2006)

For fixed $p \in (0, 1)$, asymptotically almost surely, $G \in G_{n,p}$ admits a vertex coloring 1-2-weighting of the edges.

Progress on the 1,2,3 - Conjecture

Theorem (A.-B., D., McDiarmid, R. and Thomason, *2007*)

Every (nice) graph permits a vertex-coloring 30-edge weighting.

Theorem (Addario-Berry, Dalal, and Reed, *2008*)

Every (nice) graph permits a vertex-coloring 16-edge weighting.

Theorem (Wang and Yu, *2008*)

Every (nice) graph permits a vertex-coloring 13-edge weighting.

Progress on the 1,2,3 - Conjecture

Theorem (A.-B., D., McDiarmid, R. and Thomason, *2007*)

Every (nice) graph permits a vertex-coloring 30-edge weighting.

Theorem (Addario-Berry, Dalal, and Reed, *2008*)

Every (nice) graph permits a vertex-coloring 16-edge weighting.

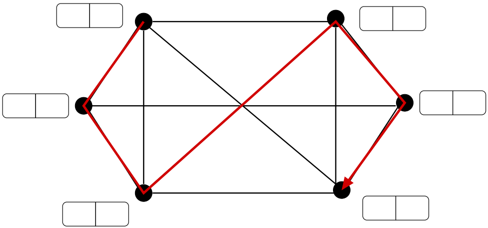
Theorem (Wang and Yu, *2008*)

Every (nice) graph permits a vertex-coloring 13-edge weighting.

Theorem (Kalkowski, Karoński, and P, *2009*)

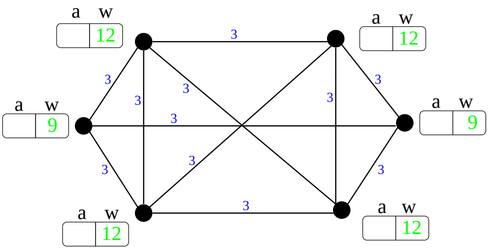
Every (nice) graph permits a vertex-coloring 5-edge weighting.

Proof sketch



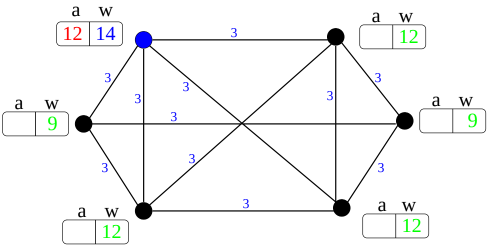
- order the vertices so that the last vertex has degree at least 2, and every other vertex has a forward edge

Proof sketch



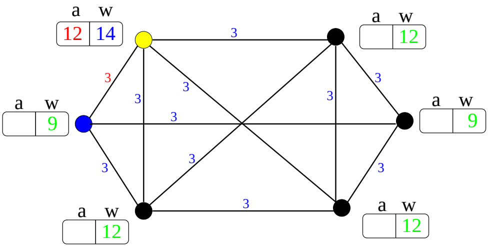
- order the vertices so that the last vertex has degree at least 2, and every other vertex has a forward edge
- start with all edge-weights of a graph set to 3

Proof sketch



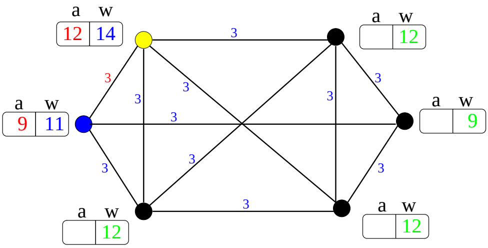
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



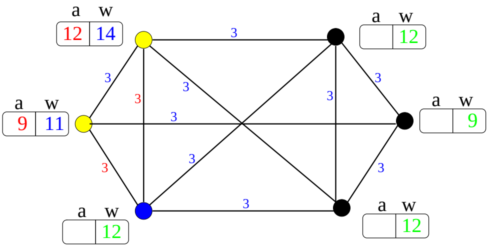
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



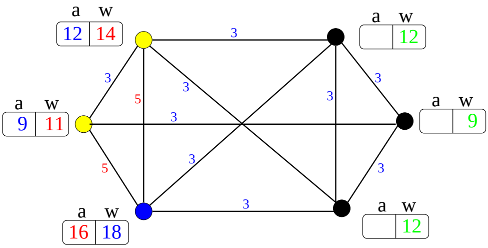
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



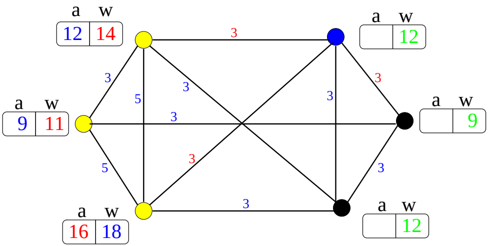
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



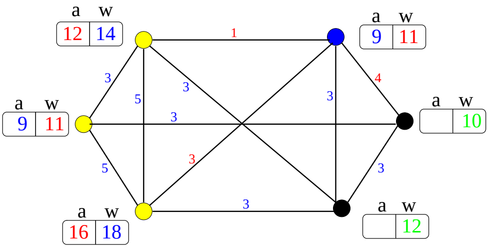
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



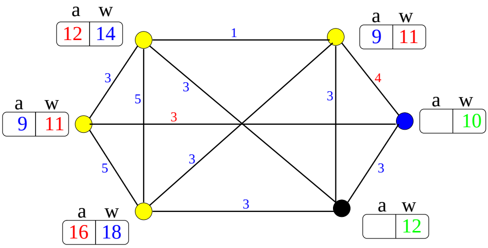
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



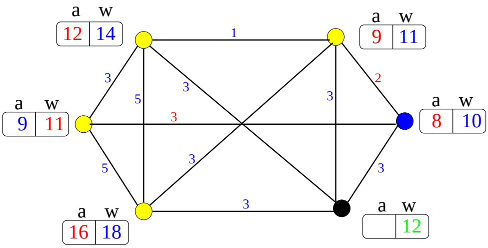
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



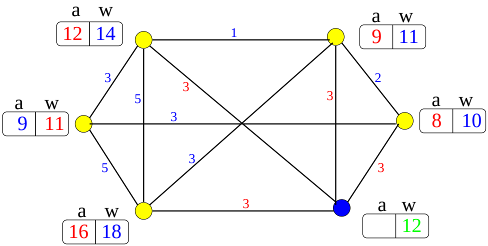
- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a + 2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

Proof sketch



- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

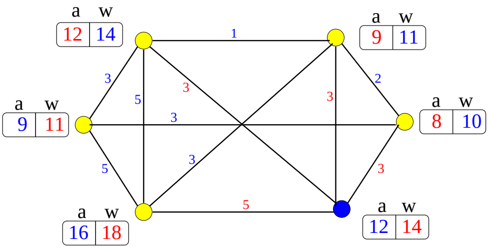
Proof sketch



- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

- fit in the last vertex

Proof sketch



- change backwards weights by 2, and the first forward weight by 1
- assign 2 weights $\{a, a+2\}$ to each vertex
- never allow the weighted degree of a vertex to leave its set

- fit in the last vertex

Vertex coloring edge weightings—a polynomial view

$$p(G) := \prod_{xy \in E} (w(x) - w(y)) = \prod_{xy \in E} \left(\sum_{u \in N(x)} w(ux) - \sum_{v \in N(y)} w(vy) \right)$$

Theorem (Combinatorial Nullstellensatz, Alon '99)

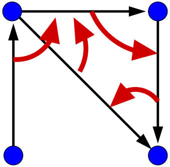
Let F be an arbitrary field, and let $f = f(x_1, ..., x_n)$ be a polynomial in $F[x_1, ..., x_n]$. Suppose the degree $deg(f)$ of f is $n = \sum t_i$, where each t_i is a nonnegative integer, and suppose the coefficient of $\prod x_i^{t_i}$ in f is nonzero.
Then, if $S_1, ..., S_n$ are subsets of F with $|S_i| > t_i$, there are $s_1 \in S_1, s_2 \in S_2, ..., s_n \in S_n$ so that $f(s_1, ..., s_n) \neq 0$.

Vertex coloring edge weightings—a polynomial view

$$p(G) := \prod_{xy \in E} (w(x) - w(y)) = \prod_{xy \in E} \left(\sum_{u \in N(x)} w(ux) - \sum_{v \in N(y)} w(vy) \right)$$

Vertex coloring edge weightings—a polynomial view

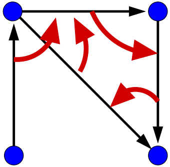
$$p(G) := \prod_{xy \in E} (w(x) - w(y)) = \prod_{xy \in E} \left(\sum_{u \in N(x)} w(ux) - \sum_{v \in N(y)} w(vy) \right)$$



$$A = \begin{pmatrix} 0 & \boxed{+} & 0 & + \\ - & 0 & \boxed{+} & - \\ 0 & - & 0 & \boxed{+} \\ - & \boxed{-} & + & 0 \end{pmatrix}$$

Vertex coloring edge weightings—a polynomial view

$$p(G) := \prod_{xy \in E} (w(x) - w(y)) = \prod_{xy \in E} \left(\sum_{u \in N(x)} w(ux) - \sum_{v \in N(y)} w(vy) \right)$$



$$A = \begin{pmatrix} 0 & \boxed{+} & 0 & + \\ - & 0 & \boxed{+} & - \\ 0 & - & 0 & \boxed{+} \\ - & \boxed{-} & + & 0 \end{pmatrix}$$

Consider $Per(A)$, the coefficient of $\prod_{xy \in E} w(xy)$.

Vertex coloring edge weightings from lists

Lemma

If $Per(A) \neq 0$, *then G admits a vertex coloring edge weighting from lists of size* 2.

$$A = \begin{pmatrix} 0 & + & 0 & + \\ - & 0 & + & - \\ 0 & - & 0 & + \\ - & - & + & 0 \end{pmatrix}$$

Vertex coloring edge weightings from lists

Lemma

If $Per(A) \neq 0$, then G admits a vertex coloring edge weighting from lists of size 2.

Lemma

If $B \subset (A \mid A)$ is an $|E| \times |E|$ matrix and if $Per(B) \neq 0$, then G admits a vertex coloring edge weighting from lists of size 3.

$$(A|A) = \begin{pmatrix} 0 & + & 0 & + & 0 & + & 0 & + \\ - & 0 & + & - & - & 0 & + & - \\ 0 & - & 0 & + & 0 & - & 0 & + \\ - & - & + & 0 & - & - & + & 0 \end{pmatrix}$$

Neighbor-distinguishing total weightings

A *k-total weighting* of a simple graph G is an assignment of an integer weight $w(e), w(v) \in [k]$ to each edge e and each vertex of G.

Neighbor-distinguishing total weightings

A k*-total weighting* of a simple graph G is an assignment of an integer weight $w(e), w(v) \in [k]$ to each edge e and each vertex of G.

A k-total weighting is *neighbor-distinguishing* if for every edge uv

$$w(u) + \sum_{e:u\in e} w(e) \neq w(v) + \sum_{e:v\in e} w(e).$$

The 1,2-conjecture

Conjecture (Przybyło and Woźniak, 2007, possibly also Hulgan, Lehel and Yoshimoto)

Every simple graph permits a neighbor-distinguishing 2-total-weighting.

The 1,2-conjecture

Conjecture (Przybyło and Woźniak, 2007, possibly also Hulgan, Lehel and Yoshimoto)

Every simple graph permits a neighbor-distinguishing 2-total-weighting.

The conjecture is true for bipartite graphs, complete graphs, cycles, cubic graphs, 4-regular graphs, graphs with maximum degree $\Delta \leq 3$.

The 1,2-conjecture

Conjecture (Przybyło and Woźniak, 2007, possibly also Hulgan, Lehel and Yoshimoto)

Every simple graph permits a neighbor-distinguishing 2-total-weighting.

The conjecture is true for bipartite graphs, complete graphs, cycles, cubic graphs, 4-regular graphs, graphs with maximum degree $\Delta \leq 3$.

Theorem (Kalkowski, 2008)

Every simple graph permits a neighbor-distinguishing 3-total-weighting. Moreover, we may refrain from assigning weight 3 to the vertices.

The 1,2-conjecture

Conjecture (Przybyło and Woźniak, 2007, possibly also Hulgan, Lehel and Yoshimoto)

Every simple graph permits a neighbor-distinguishing 2-total-weighting.

The conjecture is true for bipartite graphs, complete graphs, cycles, cubic graphs, 4-regular graphs, graphs with maximum degree $\Delta \leq 3$.

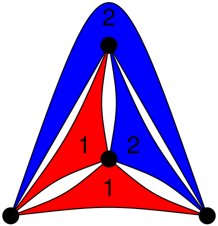
Theorem (Kalkowski, 2008)

Every simple graph permits a neighbor-distinguishing 3-total-weighting. Moreover, we may refrain from assigning weight 3 to the vertices.

Theorem (Zhu)

Kalkowski's theorem is true in the list version.

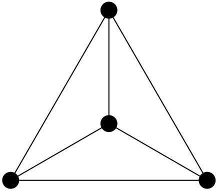
What about hypergraphs?



Proper Coloring: no monochromatic edge

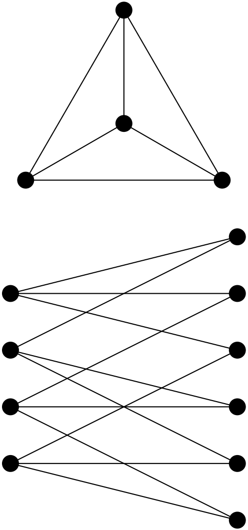
Lower bounds

- Start with any hypergraph F



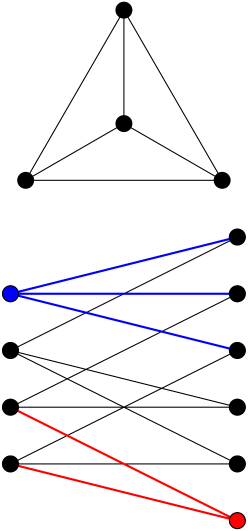
Lower bounds

- Start with any hypergraph F
- Look at the vertex-edge incidence graph G

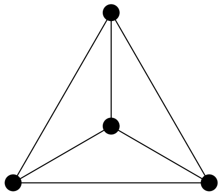


Lower bounds

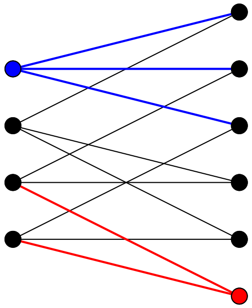
- Start with any hypergraph F
- Look at the vertex-edge incidence graph G
- Construct a new hypergraph H: $V(H) = E(G)$, $E(H) = V(G)$ (the dual hypergraph of G)



Lower bounds

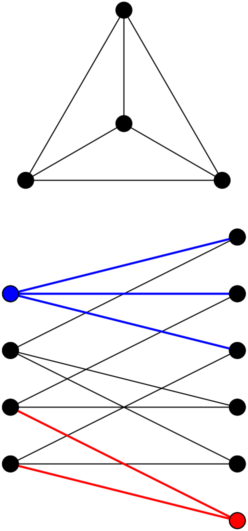


- Start with any hypergraph F
- Look at the vertex-edge incidence graph G
- Construct a new hypergraph H: $V(H) = E(G)$, $E(H) = V(G)$ (the dual hypergraph of G)
- H is bipartite and simple



Lower bounds

- Start with any hypergraph F
- Look at the vertex-edge incidence graph G
- Construct a new hypergraph H:
  $V(H) = E(G)$, $E(H) = V(G)$
  (the dual hypergraph of G)
- H is bipartite and simple
- We need at least $\{1, 2, \ldots, \kappa(F)\}$



Upper bounds

Theorem

For every nice simple hypergraph H with all edges of order between 2 and $r \geq 2$, there is a weighting $\omega : E(H) \to \{1, 2, \ldots, \max\{5, r+1\}\}$, such that the induced vertex weights properly color $V(H)$.

Upper bounds

Theorem

For every nice simple hypergraph H with all edges of order between 2 and $r \geq 2$, there is a weighting $\omega : E(H) \to \{1, 2, \ldots, \max\{5, r+1\}\}$, such that the induced vertex weights properly color $V(H)$.

Theorem

For every nice hypergraph H with all edges of order between 2 and 3, there is a weighting $\omega : E(H) \to \{1, 2, \ldots, 5\}$, such that the induced vertex weights properly color $V(H)$.

Upper bounds

Theorem

For every nice simple hypergraph H with all edges of order between 2 and $r \geq 2$, there is a weighting $\omega : E(H) \to \{1, 2, \dots, \max\{5, r+1\}\}$, such that the induced vertex weights properly color $V(H)$.

Theorem

For every nice hypergraph H with all edges of order between 2 and 3, there is a weighting $\omega : E(H) \to \{1, 2, \dots, 5\}$, such that the induced vertex weights properly color $V(H)$.

Theorem

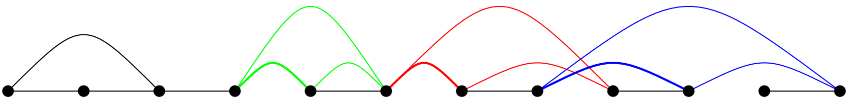
For every nice hypergraph H with all edges of order between 2 and r, there is a weighting $\omega : E(H) \to \{1, 2, \dots, 5r-5\}$, such that the induced vertex weights properly color $V(H)$.

Proof sketch

- Delete vertices until every vertex is in a 2-edge

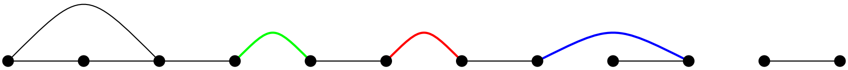
Proof sketch

- Delete vertices until every vertex is in a 2-edge
- Sort the vertices in a smart order, and consider the graph G whose edges are the first two vertices of every edge in H



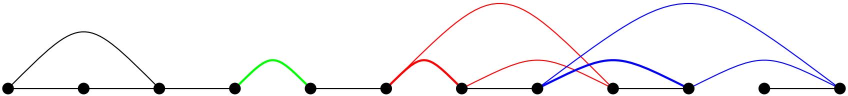
Proof sketch

- Delete vertices until every vertex is in a 2-edge
- Sort the vertices in a smart order, and consider the graph G whose edges are the first two vertices of every edge in H
- Use the graph algorithm on the large components of G



Proof sketch

- Delete vertices until every vertex is in a 2-edge
- Sort the vertices in a smart order, and consider the graph *G* whose edges are the first two vertices of every edge in *H*
- Use the graph algorithm on the large components of *G*
- Deal with the components of size 2



Uniform hypergraphs

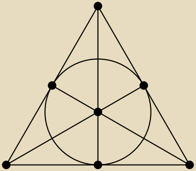
Uniform hypergraphs

If F is r-regular, r-uniform and not bipartite, our construction gives us an r-uniform hypergraph which needs at least $\{1, 2, 3\}$.

Uniform hypergraphs

If F is r-regular, r-uniform and not bipartite, our construction gives us an r-uniform hypergraph which needs at least $\{1, 2, 3\}$.

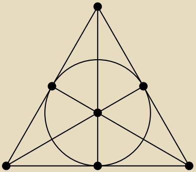
For $r = 3$, there are infinite families.



Uniform hypergraphs

If F is r-regular, r-uniform and not bipartite, our construction gives us an r-uniform hypergraph which needs at least $\{1, 2, 3\}$.

For $r = 3$, there are infinite families.



Theorem (Thomassen '92)

For $r \geq 4$, there are none.

Uniform hypergraphs

Conjecture

For every 3*-uniform hypergraph* H *without an isolated edge, there is a weighting* $\omega : E(H) \to \{1, 2, 3\}$*, such that the induced vertex weights properly color* $V(H)$*.*

Uniform hypergraphs

Conjecture

For every 3*-uniform hypergraph* H *without an isolated edge, there is a weighting* $\omega : E(H) \to \{1, 2, 3\}$*, such that the induced vertex weights properly color* $V(H)$*.*

Conjecture

For $r \geq 4$*, for every* r*-uniform hypergraph* H *without an isolated edge, there is a weighting* $\omega : E(H) \to \{1, 2\}$*, such that the induced vertex weights properly color* $V(H)$*.*