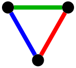
Rainbow triangles in 3-edge-colored graphs

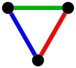
József Balogh, Ping Hu, Bernard Lidický, Florian Pfender,
Jan Volec, Michael Young

Guangdong Combinatorics and Graphs Conference
8 July 2018

Problem

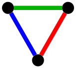
Find a 3-edge-coloring of a complete graph K_n maximizing the number of copies of rainbow colored triangles  *.*

Problem

Find a 3-edge-coloring of a complete graph K_n maximizing the number of copies of rainbow colored triangles  *.*

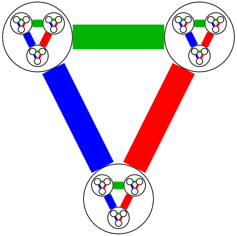
Color edges randomly, expected density $\frac{2}{9}$.

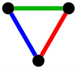
Problem

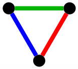
Find a 3-edge-coloring of a complete graph K_n maximizing the number of copies of rainbow colored triangles  *.*

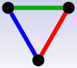
Color edges randomly, expected density $\frac{2}{9}$.

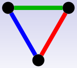
Iterated blow-up of triangle



$\frac{1}{4} =$  .

 denotes graph and/or its density

$F(n) =$ max # of  over all 3-edge-colorings of K_n

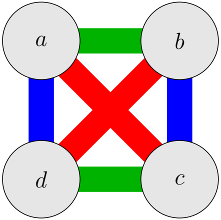
$F(n) = \max \#$ of  over all 3-edge-colorings of K_n

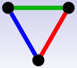
Conjecture (Erdős and Sós; '72$^-$)

For all $n > 0$,

$$F(n) = F(a) + F(b) + F(c) + F(d) + abc + abd + acd + bcd,$$

where $a + b + c + d = n$; a, b, c, d are as equal as possible, and $F(0) = 0$.



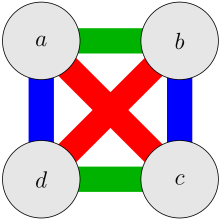
$F(n) = \max \#$ of  over all 3-edge-colorings of K_n

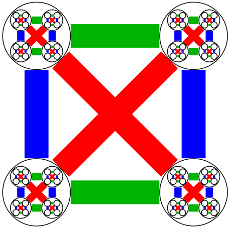
Conjecture (Erdős and Sós; '72⁻)

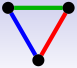
For all $n > 0$,

$$F(n) = F(a) + F(b) + F(c) + F(d) + abc + abd + acd + bcd,$$

where $a + b + c + d = n$; a, b, c, d are as equal as possible, and $F(0) = 0$.





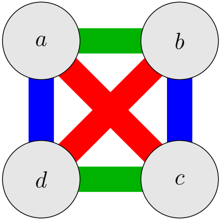
$F(n) = \max \#$ of  over all 3-edge-colorings of K_n

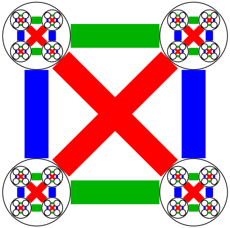
Conjecture (Erdős and Sós; '72⁻)

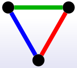
For all $n > 0$,

$$F(n) = F(a) + F(b) + F(c) + F(d) + abc + abd + acd + bcd,$$

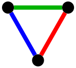
where $a + b + c + d = n$; a, b, c, d are as equal as possible, and $F(0) = 0$.

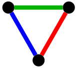




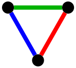
$0.4 =$ 

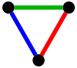
Flag algebras application

Construction: $0.4 \leq$ 

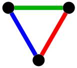
- get a matching upper bound  ≈ 0.4
- round the result
- get subgraphs with 0 density
- get extremal construction (stability)

Flag algebras application

Construction: $0.4 \leq$ 

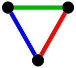
- get a matching upper bound  ≈ 0.4
- round the result
- get subgraphs with 0 density
- get extremal construction (stability)

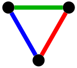
Flag algebras (on 6 vertices) give only

 $\leq 0.4006,$

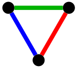
not enough for rounding.

Flag algebras application

Construction: $0.4 \leq$ 

- get a matching upper bound  ≈ 0.4
- round the result
- get subgraphs with 0 density
- get extremal construction (stability)

Flag algebras (on 6 vertices) give only

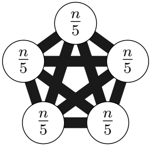
 ≤ 0.4006,

not enough for rounding.
The iterative extremal construction is causing troubles....

Not iterated extremal constructions

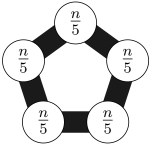
Theorem (Turán)

# of edges over K_l-free graphs is maximized by



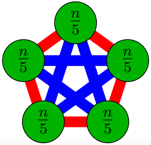
Theorem (Hatami, Hladký, Kráľ, Norine, Razborov)

# of C_5s over triangle-free graphs is maximized by



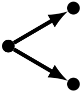
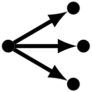
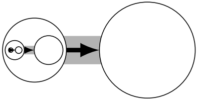
Theorem (Cummings, Kráľ, Pfender, Sperfeld, Treglown, Young)

# of monochromatic triangles over 3-edge-colored graphs is minimized by

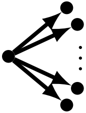
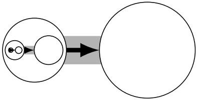


Iterated extremal constructions

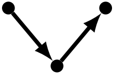
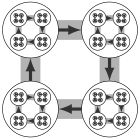
Theorem (Falgas-Ravry, Vaughan)

 # of  *and*  *is maximized by* 

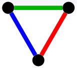
Theorem (Huang)

 # of  *is maximized by* 

Theorem (Hladký, Král', Norine)

 # of  *is maximized by* 

Our main result

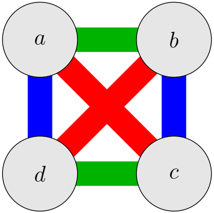
$F(n) =$ max # of  over all coloring of K_n

Theorem (Balogh, H., Lidický, Pfender, Volec, Young 2017)

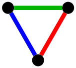
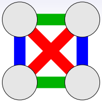
For all $n > n_0$,

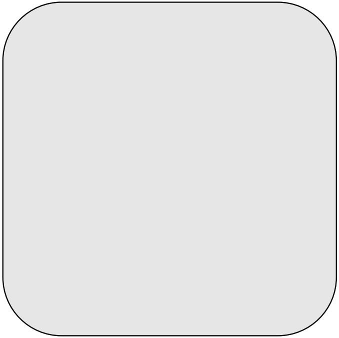
$$F(n) = F(a) + F(b) + F(c) + F(d) + abc + abd + acd + bcd,$$

where $a + b + c + d = n$; a, b, c, d are as equal as possible.

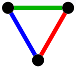
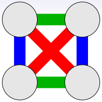


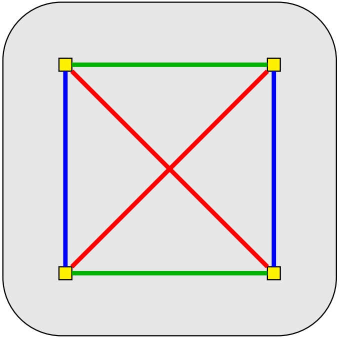
Sketch of proof

Goal: maximizing  gives edge-coloring like 



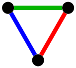
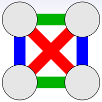
Sketch of proof

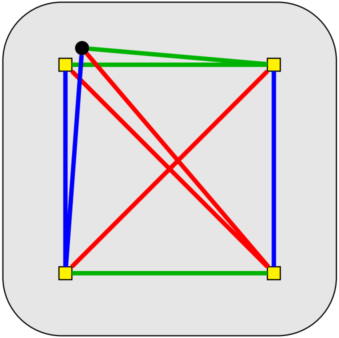
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4

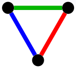
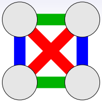
Sketch of proof

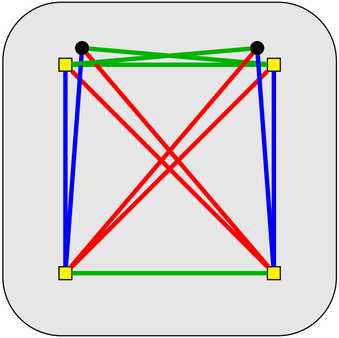
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4

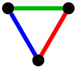
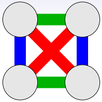
Sketch of proof

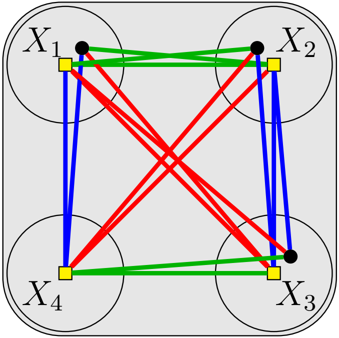
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4

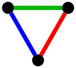
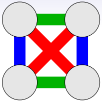
Sketch of proof

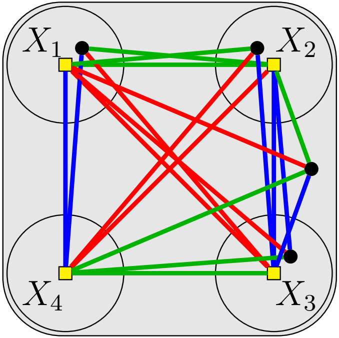
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest

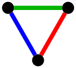
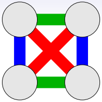
Sketch of proof

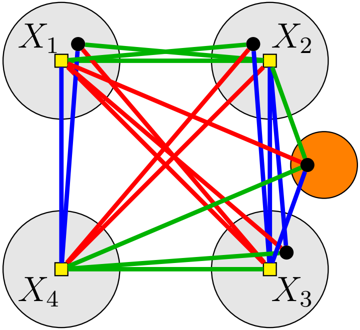
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest

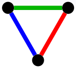
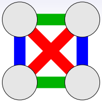
Sketch of proof

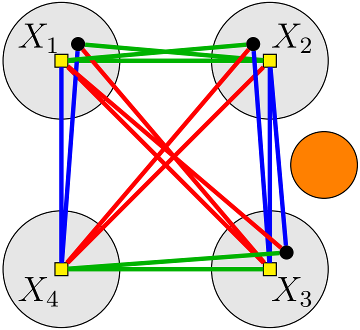
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest

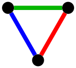
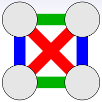
Sketch of proof

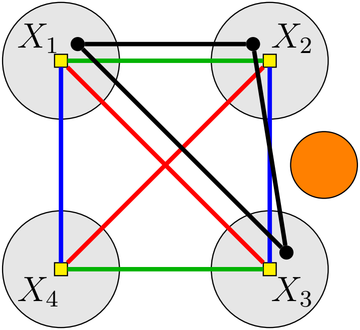
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest

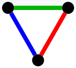
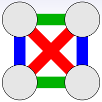
Sketch of proof

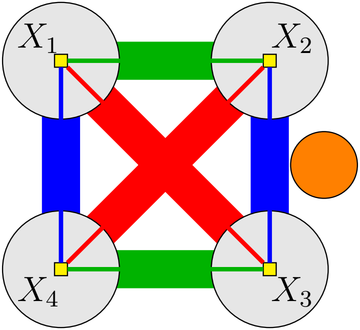
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest
- correct edges between X_is

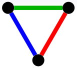
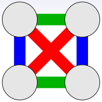
Sketch of proof

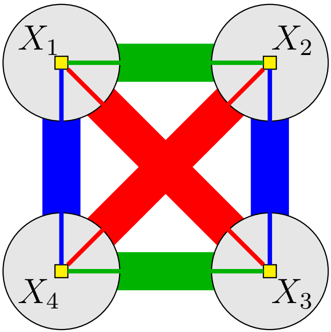
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest
- correct edges between X_is
- no orange trash

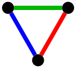
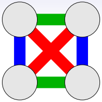
Sketch of proof

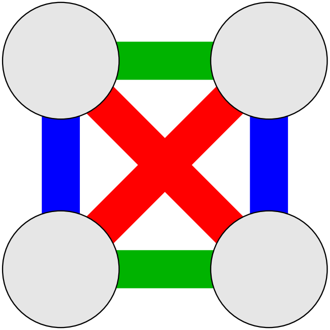
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest
- correct edges between X_is
- no orange trash
- balance sizes of X_is

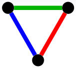
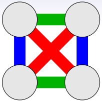
Sketch of proof

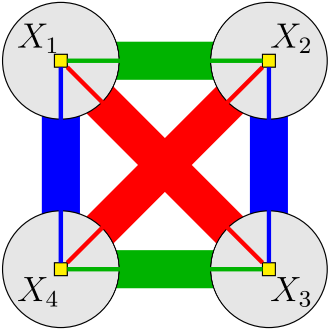
Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest
- correct edges between X_is
- no orange trash
- balance sizes of X_is

Sketch of proof

Goal: maximizing  gives edge-coloring like 



- pick a properly 3-edge-colored K_4
- partition the rest
- correct edges between X_is
- no orange trash
- balance sizes of X_is

How to pick the properly 3-edge-colored K_4?

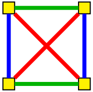
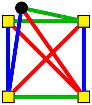
($|X_i|$s close to $0.25n$, few wrongly colored edges, small trash)

How to pick K_4?

Use Flag Algebras!

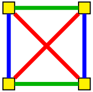
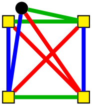
How to pick K_4?

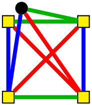
Use Flag Algebras!

Try 1: Pick  maximizing 

How to pick K_4?

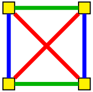
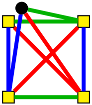
Use Flag Algebras!

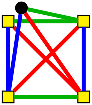
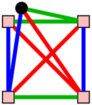
Try 1: Pick  maximizing 

 $(n-4) \geq$

How to pick K_4?

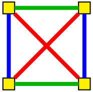
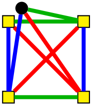
Use Flag Algebras!

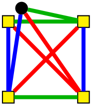
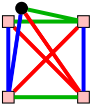
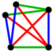
Try 1: Pick  maximizing 

 $(n-4) \geq \frac{1}{\square \binom{n}{4}}$ $\sum$   $(n-4)$

How to pick K_4?

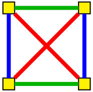
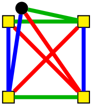
Use Flag Algebras!

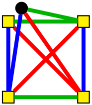
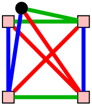
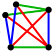
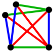
Try 1: Pick  maximizing 

 $(n-4) \geq \frac{1}{}$  $\binom{n}{4}$ $\sum$   $(n-4) = 2$  $\binom{n}{5}$ /  $\binom{n}{4}$

How to pick K_4?

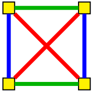
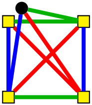
Use Flag Algebras!

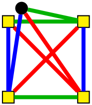
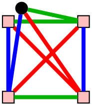
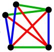
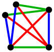
Try 1: Pick  maximizing 

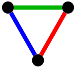
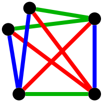
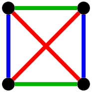
 $(n-4) \geq \frac{1}{\text{[graph]}\binom{n}{4}}$  $\sum$ (over )  $(n-4) =$ $\frac{2\,\text{[graph]}\binom{n}{5}}{\text{[graph]}\binom{n}{4}}$   $= \frac{2\,\text{[graph]}}{5\,\text{[graph]}}$   $(n-5)$

HOW TO PICK K_4?

Use Flag Algebras!

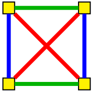
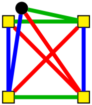
Try 1: Pick  maximizing 

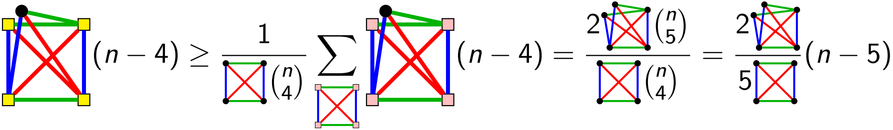
 $(n-4) \geq \frac{1}{\text{[}}$$\binom{n}{4}\text{]}$ $\sum$ (over )  $(n-4) = \frac{2 \text{ [}}{}$$\binom{n}{5}\text{] / [}$$\binom{n}{4}\text{]} = \frac{2\text{ [}}{}$$\text{] / [}5$ $\text{]}(n-5)$

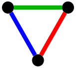
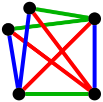
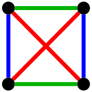
FA:  ≥ 0.4 then  > 0.23516,  < 0.0952

How to pick K_4?

Use Flag Algebras!

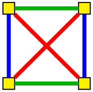
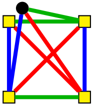
Try 1: Pick  maximizing  > 0.988

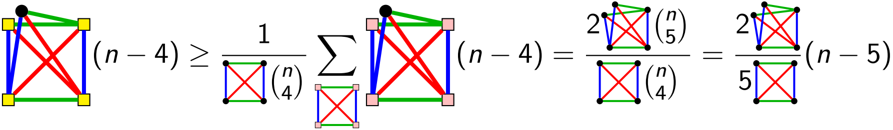


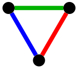
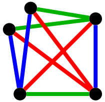
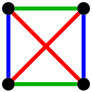
FA:  ≥ 0.4 then  > 0.23516,  < 0.0952

How to pick K_4?

Use Flag Algebras!

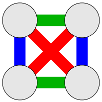
Try 1: Pick  maximizing  > 0.988



FA:  ≥ 0.4 then  > 0.23516,  < 0.0952

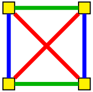
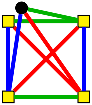
Result for K_n:

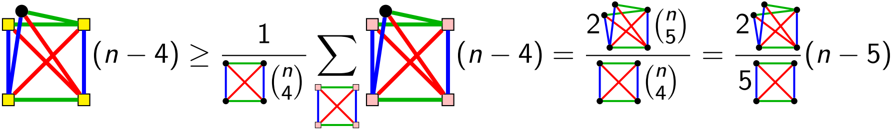
$$|X_1| + |X_2| + |X_3| + |X_4| > 0.988(n-4)$$

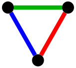
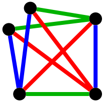
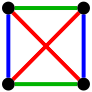


How to pick K_4?

Use Flag Algebras!

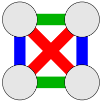
Try 1: Pick  maximizing  > 0.988



FA:  ≥ 0.4 then  > 0.23516,  < 0.0952

Result for K_n:

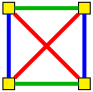
$$|X_1| + |X_2| + |X_3| + |X_4| > 0.988(n-4)$$

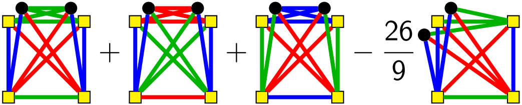


Balancing needed...

How to pick K_4?

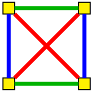
Use Flag Algebras!

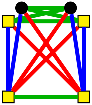
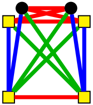
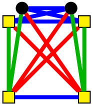
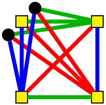
Try 2: Pick  maximizing

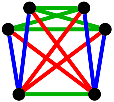
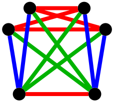
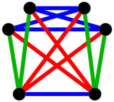
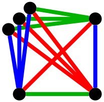


How to pick K_4?

Use Flag Algebras!

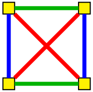
Try 2: Pick  maximizing

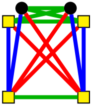
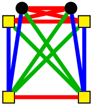
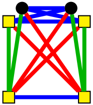
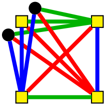
 $+$  $+$  $-\frac{26}{9}$ 

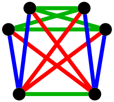
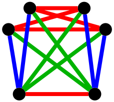
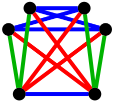
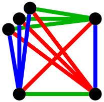
FA: $\frac{4}{15}\Big($  $+$  $+$  $\Big) - \frac{26}{45}$  > 0.002629

How to pick K_4?

Use Flag Algebras!

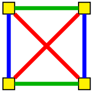
Try 2: Pick  maximizing

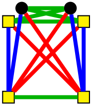
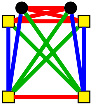
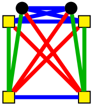
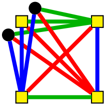
 $+$  $+$  $- \frac{26}{9}$  > 0.0276

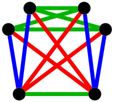
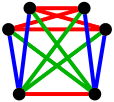
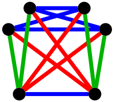
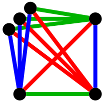
FA: $\frac{4}{15} \Bigg($  $+$  $+$  $\Bigg) - \frac{26}{45}$  > 0.002629

How to pick K_4?

Use Flag Algebras!

Try 2: Pick  maximizing

 $+$  $+$  $-\frac{26}{9}$  > 0.0276

FA: $\frac{4}{15}\Big($  $+$  $+$  $\Big) - \frac{26}{45}$  > 0.002629

Final equation:

$$2\sum_{1\le i<j\le 4}|X_i||X_j| - |F| - \tfrac{26}{9}\sum_{1\le i\le 4}|X_i|^2 > 0.0276n^2$$

F = wrongly colored edges.

How the first step worked

Quadratic programming

$$2 \sum_{1 \le i < j \le 4} |X_i||X_j| - |F| - \frac{26}{9} \sum_{1 \le i \le 4} |X_i|^2 > 0.0276n^2$$

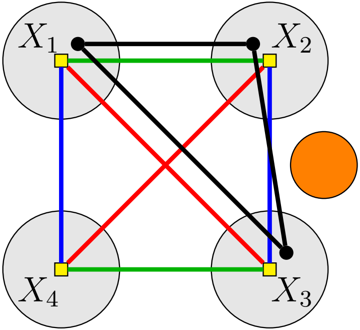
Use Lagrange multipliers:

$$0.244n < |X_i| < 0.256n$$

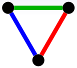
$$|\mathit{Trash}| < 0.006n$$

$$|F| < 0.00008\binom{n}{2}$$

F = wrongly colored edges.



Main Result

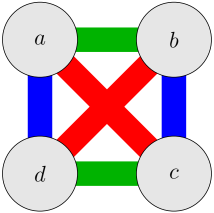
$F(n) =$ max # of  over all coloring of K_n

Theorem (Balogh, H., Lidický, Pfender, Volec, Young 2017)

For all $n > n_0$,

$$F(n) = F(a) + F(b) + F(c) + F(d) + abc + abd + acd + bcd,$$

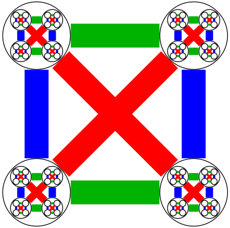
where $a + b + c + d = n$; a, b, c, d are as equal as possible.



More results

Theorem

# of rainbow K_3s is maximized by

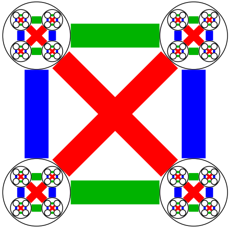


if on 4^k vertices.

More results

Theorem

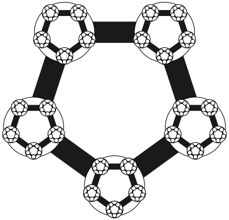
# of rainbow K_3s is maximized by



if on 4^k vertices.

Theorem

# of induced C_5s is maximized by

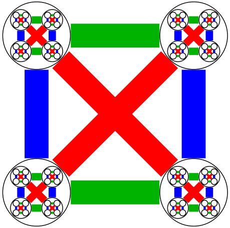


if on 5^k vertices.

More results

Theorem

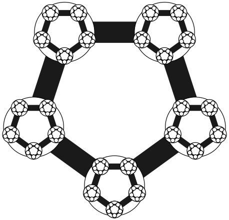
# of rainbow K_3s is maximized by



if on 4^k vertices.

Theorem

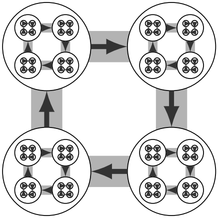
# of induced C_5s is maximized by



if on 5^k vertices.

Theorem

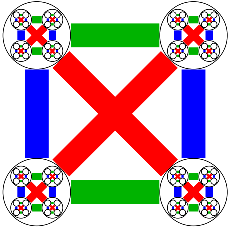
# of induced oriented C_4s is maximized by



if on 4^k vertices.

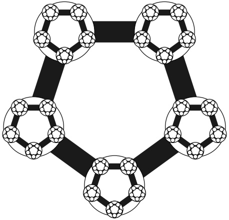
More results

Theorem
# of rainbow K_3s is maximized by



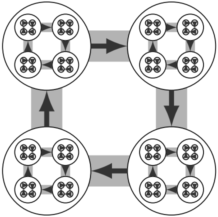
if on 4^k vertices.

Theorem
# of induced C_5s is maximized by



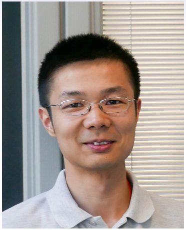
if on 5^k vertices.

Theorem
# of induced oriented C_4s is maximized by



if on 4^k vertices.

(for all k)


Thank you for listening!