Flagmatic and stability

Jakub Sliačan joint work with Oleg Pikhurko and Konstantinos Tyros

25th British Combinatorial Conference, University of Warwick

July 6, 2015

Figuratic is an application that uses flig algebras are control for the problem. Figuratic is designed in a way that means it can be used to reflect the figuration of the problems. Figuratic is designed in a way that means it can be used to reflect the figuration of the problems. Figuratic 20 is a retrivention of Figuratic are like practices. Figuratic 20 is a retrivention of Figuratic are like practices. Figuratic 20 is a retrivention of Figuratic are like practices. To download, please use the link above. Also, you can read the User's Guide. To download, please use the link above. Also, you can read the User's Guide.		vÔ/ Queen Marv
A load for reasonables in extremel graph theory. The other reasonables in extremel graph theory. The other reasonables in the set of the set		Ilarmatic
A load for researching in external graph theory.		liagillaut
Construct Fagment: Vew Project on GifHub Figmatic is an application that uses flag algebras age of the approximation of the bounds on Virial density and related problems, using the memily of of Razborov. Figmatic is designed in a way that means it can be bound to be different kinds of problems. Currently, is can solve graph, ordenied graph and Lengan motions. In fact, it was originally created to obside 3-graph problems. Figmatic 20 is a relevention of Figuratia as: Sange package. To download, please use the link above. Also, you can read the Leng's Guide. You may need to download the Machinary of CSDP (More information about CSDP here.)		A tool for researchers in extremal graph theory.
Controled Flagmatic Controled Flagmatic Control of Flagmatic and Flagmati		
Fligmatic is an application that uses flag algebras and formation to provide the second secon		Download Flagmatic View Project on GitHub
Figmatic is an application that uses flag algebras an earlier common common programming to find bounds on Figmatic is designed in a way that means it can be compared to a factorize. Figmatic is designed in a way that means it can be compared to a single different kinds of problems. Currently, it can solve graph, oriented graph and <u>3-graph problems</u> . In fact, it was originally created to solve 3-graph problems. Flagmatic 20 is a reinvention of Flagmatic as Stage package. To download, please use the link above. Also, you can read the Liser's Guide. You may need to download the Maching of CSDP. (More information about CSDP here.)		
Flagmatic is an application that uses flag algebras application that application that uses flag algebras application that uses flag algebras application that use flag algebras application that use flag algebras application that uses flag algebras applications. Flagmatic 20 is a releviention of Flagmatic as flag application that uses flag algebras applications. To download, please use the link above. Also, you can read the User's Guide. You may need to download the Mac brandy of CSDP (More information about CSDP here.)		
Flagmatic is designed in a way that means it can be observed and the function of problems. Currently, it can observe applic, reinted graph and <u>3-craph</u> problems. In fact, it was originally created to solve <u>3-graph</u> problems. Flagmatic 20 is a reinvention of Flagmatic as lisage package. To download, please use the link above. Also, you can read the Liser's Guide. You may need to download the Machinary of CSDP. (More information about CSDP here.)		
Currently, it can solve grape, mented grappin not engined incoments, in tack, it was organary oreated to solve 3-graph problem. Flagmatic 20 is a reinvention of Flagmatic as a large package. To download, please use the link above. Also, you can read the Liter's Guide. You may need to download the Machinary of CSDP. (More information about CSDP here.)	Flagmatic is an ap Turán density and	oplication that uses flag algebras and semi-demine opramming to find bounds on related problems, using the method of Razborov.
Flagmatic 2.0 is a reinvention of Flagmatic as Sage package. To download, please use the link above, Also, you can read the User's Guide. You may need to download the Mac binary of CSDP, (More information about CSDP here.)	Flagmatic is an ap Turán density and Flagmatic is desig	oplication that uses flag algebras any soft-beams corranning to find bounds on related problems, using the metibol of Razborov. med in a way that means it can be accurate the hear on different kinds of problems.
To download, please use the link above. Also, you can read the User's Guide. You may need to download the Mac binary of CSDP. (More information about CSDP here.)	Flagmatic is an ap Turán density and Flagmatic is desig Currently, it can s solve 3-graph pro	oplication that uses flag algebras any semi-deametrogramming to find bounds on related problems, using the method of Razborov. pred in a way that means it can be applied bounded different kinds of problems. over graph, oriented graph and <u>S-graph</u> problems. In fact, it was originally created to bems.
You may need to download the Mac binary of CSDP. (More information about CSDP here.)	Flagmatic is an ap Turán density and Flagmatic is desig Currently, it can s solve 3-graph pro Flagmatic 2.0 is a	plication that uses flag algebras and form the community of find bounds on related problems, using the method of Razborov, pred in a way that means it can be appreciate and different kinds of problems. Now graph, oriented graph and august problems. In fact, it was originally created to beins.
	Flagmatic is an ap Turán density and Flagmatic is desig Currently, it can s solve 3-graph pro Flagmatic 2.0 is a To download, plea	plication that uses flag algebras and full doministic oranning to find bounds on related problems, using the matural of Razborov. med in a way that means it can be considered within different kinds of problems. over graph, oriented graph and <u>Cargeb</u> problems. In fact, it was originally created to blems. reinvention of Flagmatic are <u>Rage</u> package, see use the link above. Also, you can read the User's Guide.

https://github.com/jsliacan/flagmatic-dev.git
(tested with Sage 6.4)

Maximize induced density of a small H in a big F-free G.

$$p(H;G) = \frac{\# \text{ induced copies of } H \text{ in } G}{\binom{|G|}{|H|}}$$

Example Maximize the density of 1 in a 4-free G.

density

$$\lambda_{\mathcal{F}}(H,n) = \max_{G \in \mathcal{G}_n} p(H;G)$$

asymptotic density

$$\lambda_{\mathcal{F}}(H) = \lim_{n \to \infty} \lambda_{\mathcal{F}}(H, n)$$

Theorem (Mantel, 1907)

$$\lambda_{\bigstar}(\mathbf{I}) = 1/2$$

with the extremal graph being complete balanced bipartite.

Every sufficiently large almost extremal graph is close in edit distance to a complete balanced bipartite graph.

The problem is stable if for every $\epsilon > 0$ there is a $\delta > 0$ and an $n_0 \in \mathbb{N}$ such that every \bigtriangleup -free graph G on $n > n_0$ vertices that satisfies $p(\bullet; G) \ge 1/2 - \delta$ also satisfies $d_{\text{edit}}(G, K_2(\frac{1}{2}, \frac{1}{2})) < \epsilon$.

Inducibility of K_4^-

Theorem (Hirst, 2013)

$$\lambda(\mathbf{k}) = \frac{72}{125}$$

with the extremal graph being a balanced blow-up of K_5 .

Theorem

The inducibility of K_4^- is a stable problem.

For every $\epsilon > 0$ there exists $\delta > 0$ and $n_0 \in \mathbb{N}$ such that for all graphs G on $n \ge n_0$ vertices with

$$p(K_4^-;G) \geq \frac{72}{125} - \delta_2$$

there exists a partition of the vertex set of G, $V(G) = V_1 \cup \ldots \cup V_5$ such that

 $d_{edit}(G(V_1,\ldots,V_5),K_5(n)) < \epsilon.$

Proof of stability

Listing 1: Flagmatic-dev script

```
K5 = "5:12131415232425343545"
p = GraphProblem(7, density="4:1213142334")
p.set_extremal_construction(GraphBlowupConstruction(K5))
p.solve_sdp()
p.make_exact(denominator=1500)
p.verify_stability(K5, K5)
```

certificates given by the script

stability theorem from the previous talk

Theorem (Hirst, 2013)

$$\lambda(\overset{\bullet}{\checkmark}) = \frac{3}{8}$$

with the extremal graph being a balanced blow-up of $F = \mathbf{I} \cup \mathbf{I}$.

Theorem

The inducibility of $\stackrel{\bullet}{\checkmark}$ is a stable problem.

Stability of Turán problem for K_5

Theorem (Hirst, 2013)

$$\lambda_{K_5}(\mathbf{I}) = \frac{3}{4}$$

with the extremal graph being a balanced blow-up of K_4 .

Theorem

The Turán problem for K_5 is stable.

Listing 2: Flagmatic-dev script

```
p = GraphProblem(5, forbid="5:12131415232425343545")
c = GraphBlowupConstruction("4:121314232434")
p.set_extremal_construction(c)
p.solve_sdp()
p.make_exact()
p.verify_stability("3:121323", "4:121314232434")
```

Density of C_5 s in a K_3 -free graph

Theorem (Grzesik, 2012)

$$\lambda_{\mathbf{4}}(C_5) = \frac{5!}{5^5}$$

with the extremal graph being a balanced blow-up of C_5 .

Theorem

The above problem is stable.

Listing 3: Flagmatic-dev script

```
C5 = "5:1223344551"
p = GraphProblem(5, forbid="3:121323", density=C5)
p.set_extremal_construction(GraphBlowupConstruction(C5))
p.solve_sdp()
p.make_exact()
p.verify_stability("3:12", C5)
```

Gentle introduction on Mantel's Theorem:

Theorem (Mantel, 1907)

The maximum edge density of a \triangle -free graph is 1/2.

Do not know $p(\stackrel{\bullet}{:}; G)$, so

$$p(\mathbf{\hat{l}}; G) = \sum_{|F|=k} p(\mathbf{\hat{l}}; F) p(F; G)$$
$$\leq \max_{|F|=k} p(\mathbf{\hat{l}}; F)$$
$$= \lambda(\mathbf{\hat{l}}, k).$$

Need a better bound

A simple max bound is rarely sharp. Example

$$p(\mathbf{I}; G) \leq \max_{|F|=3} p(\mathbf{I}; F)$$
$$= p(\mathbf{I}; \mathbf{A}) = 2/3$$

Only sharp if every subgraph of G on 3 vertices is a \checkmark . Impossible for G with \geq 5 vertices:



Account for subgraph overlaps (via example)

 G_{\bullet} is G with one vertex red (fixed).

$$p(\mathbf{\hat{s}}; G_{\bullet}) = rac{\mathsf{deg}(ullet)}{|G| - 1}$$

p(↓; G_•)p(↓; G_•) choosing two neighbours of • (with repetition)
 p(↓,↓; G_•) = p(↓, ; G_•) + p(↓; G_•) choosing two neighbours of • (without repetition)

Negligible difference when G big. \implies start with 1., switch to 2., uncolor (average over all choices of • in G). Left with $\alpha p(\checkmark; G)$.

$$\llbracket p(\stackrel{\bullet}{}; G_{\bullet})p(\stackrel{\bullet}{}; G_{\bullet}) \rrbracket_{\bullet} \sim \frac{1}{3}p(\stackrel{\bullet}{}; G)$$

Manipulation (via example)

Vector
$$v = [p(\bullet; G_{\bullet}), p(\bullet; G_{\bullet})].$$
$$\begin{bmatrix} vv^T \end{bmatrix}_{\bullet} \ge 0$$
Similarly, for every $A \succeq 0$

Similarly, for every $A \succeq 0$,

$$\left[\!\left[vAv^{T}\right]\!\right]_{\bullet} \ge 0$$

$$p(\mathbf{\hat{l}}; G) = \sum_{|F|=3} p(\mathbf{\hat{l}}; F) p(F; G)$$

$$\leq \sum_{|F|=3} p(\mathbf{\hat{l}}; F) p(F; G) + \left[vAv^T \right] \quad \text{with } A \succeq 0$$

$$= \sum_{|F|=3} \left(p(\mathbf{\hat{l}}; F) + c_F \right) p(F; G)$$

$$\leq \max_{|F|=3} p(\mathbf{\hat{l}}; F) + c_F$$

Clearly, the proces was rather systematic. Need to know: density graphs, forbidden graphs. The rest can be done by the PC.

Optimization:

 $\min \gamma:$ $p(\mathbf{l};F) + c_F \leq \gamma$,for all F $A \succeq 0$

Maximise i in a graph without copies of A. Recall $\lambda(i) \leq 1/2$. Extremal graph is complete balanced bipartite:



In Flagmatic 2.0 [Emil's]

Listing 4: Mantel's theorem.

```
p = GraphProblem(3, forbid="3:121323")
p.set_extremal_construction(GraphBlowupConstruction("2:12"))
p.solve_sdp(solver="csdp")
p.make_exact()
```

Listing 5: Output

```
Forbidding 3:121323 as a subgraph.
Generating graphs...
Generated 3 graphs.
Generated 1 types and flags...
Generated 1 types of order 1, with [2] flags of order 2.
Computing products.
Writing SDP input file...
Running SDP solver...
Returncode is 0. Objective value is 0.50000001.
Checking numerical bound...
Bound of 1/2 attained by:
1/2 : graph 0 (3:)
1/2 : graph 2 (3:1213)
```