Flagmatic and stability

Jakub Sliačan
joint work with Oleg Pikhurko and Konstantinos Tyros

25th British Combinatorial Conference, University of Warwick

July 6, 2015

Flagmatic

Home 3-Graphs Graphs Oriented Graphs

Flagmatic

Queen Mary University of London

A tool for researchers in extremal graph theory.

Download Flagmatic

View Project on GitHub

Flagmatic is an application that uses flag algebras and semi-definite programming to find bounds on Turán density and related problems, using the method of Razborov.

Flagmatic is designed in a way that means it can be adapted to handle different kinds of problems. Currently, it can solve graph, oriented graph and 3-graph problems. In fact, it was originally created to solve 3-graph problems.

Flagmatic 2.0 is a reinvention of Flagmatic as a 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.)

For Flagmatic 1.5, see the old website.

© E. R. Vaughan 2013

https://github.com/jsliacan/flagmatic-dev.git

(tested with Sage 6.4)

Problem type

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  *in a* *-free G.*

Definitions

density

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

asymptotic density

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

Stability on an example

Theorem (Mantel, 1907)

$$\lambda_{\triangle}(|) = 1/2$$

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 $\triangle$-free graph G on $n > n_0$ vertices that satisfies $p(|; G) \geq 1/2 - \delta$ also satisfies $d_{\text{edit}}(G, K_2(\frac{1}{2}, \frac{1}{2})) < \epsilon$.

Inducibility of K_4^-

Theorem (Hirst, 2013)

λ() $= \frac{72}{125}$

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 $n \geq n_0$ vertices with

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

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

Stability for inducibility of $K_{1,1,2}$

Theorem (Hirst, 2013)

$\lambda($$) = \frac{3}{8}$

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

Theorem

The inducibility of  *is a stable problem.*

Stability of Turán problem for K_5

Theorem (Hirst, 2013)

$$\lambda_{K_5}(\bullet\!\!-\!\!\bullet) = \frac{3}{4}$$

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_5s in a K_3-free graph

Theorem (Grzesik, 2012)

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

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 *-free graph is 1/2.*

Simple bound

Do not know $p(\overset{\bullet}{\underset{\bullet}{|}}; G)$, so

$$
\begin{aligned}
p(\overset{\bullet}{\underset{\bullet}{|}}; G) &= \sum_{|F|=k} p(\overset{\bullet}{\underset{\bullet}{|}}; F)p(F; G) \\
&\leq \max_{|F|=k} p(\overset{\bullet}{\underset{\bullet}{|}}; F) \\
&= \lambda(\overset{\bullet}{\underset{\bullet}{|}}, k).
\end{aligned}
$$

Need a better bound

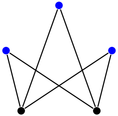
A simple max bound is rarely sharp.

Example

$p($$; G) \leq \max_{|F|=3} p($$; F)$

$= p($$;$ $) = 2/3$

Only sharp if every subgraph of G on 3 vertices is a *.*
Impossible for G with ≥ 5 vertices:



Account for subgraph overlaps (via example)

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

$$p(\overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}}) = \frac{\deg(\textcolor{red}{\bullet})}{|G|-1}$$

1. $p(\overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}})p(\overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}})$ choosing two neighbours of $\textcolor{red}{\bullet}$ (with repetition)
2. $p(\overset{\bullet}{\textcolor{red}{\bullet}}, \overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}}) = p({\bullet}\overset{\textcolor{red}{\bullet}}{\wedge}{\bullet}; G_{\textcolor{red}{\bullet}}) + p({\bullet}\overset{\textcolor{red}{\bullet}}{\triangle}{\bullet}; G_{\textcolor{red}{\bullet}})$ choosing two neighbours of $\textcolor{red}{\bullet}$ (without repetition)

Negligible difference when G big. $\implies$ start with 1., switch to 2., uncolor (average over all choices of $\textcolor{red}{\bullet}$ in G). Left with $\alpha p({\bullet}\overset{\bullet}{\wedge}{\bullet}; G)$.

$$[\![p(\overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}})p(\overset{\bullet}{\textcolor{red}{\bullet}}; G_{\textcolor{red}{\bullet}})]\!]_{\textcolor{red}{\bullet}} \sim \frac{1}{3} p({\bullet}\overset{\bullet}{\wedge}{\bullet}; G)$$

Manipulation (via example)

Vector $v = [p(\overset{\bullet}{\color{red}\bullet}; G_{\color{red}\bullet}), p(\overset{\bullet}{\overset{|}{\color{red}\bullet}}; G_{\color{red}\bullet})]$.

$$\left[\!\left[vv^T \right]\!\right]_{\color{red}\bullet} \geq 0$$

Similarly, for every $A \succeq 0$,

$$\left[\!\left[vAv^T \right]\!\right]_{\color{red}\bullet} \geq 0$$

$$\begin{aligned}
p(\overset{\bullet}{\overset{|}{\bullet}}; G) &= \sum_{|F|=3} p(\overset{\bullet}{\overset{|}{\bullet}}; F) p(F; G) \\
&\leq \sum_{|F|=3} p(\overset{\bullet}{\overset{|}{\bullet}}; F) p(F; G) + \left[\!\left[vAv^T \right]\!\right] \qquad \text{with } A \succeq 0 \\
&= \sum_{|F|=3} \left(p(\overset{\bullet}{\overset{|}{\bullet}}; F) + c_F \right) p(F; G) \\
&\leq \max_{|F|=3} p(\overset{\bullet}{\overset{|}{\bullet}}; F) + c_F
\end{aligned}$$

Delegating tasks to the PC

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

Optimization:

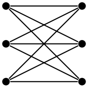
$$\min \gamma :$$

$p(\overset{\bullet}{\underset{\bullet}{|}}; F) + c_F \leq \gamma$,for all F

$$A \succeq 0$$

Mantel in Flagmatic

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

e.g. 

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.
Generating 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)
```