Automated Conjecturing in Mathematics - with the CONJECTURING Program

Craig Larson
(joint work with Nico Van Cleemput)

Virginia Commonwealth University
Ghent University

RUTGERS EXPERIMENTAL MATHEMATICS SEMINAR
28 March 2019

Problem 1.

Given an object-type and an invariant, find a *theory* of the invariant.

- Graphs & independence number
- Matrices & determinant
- Integers & number of ways to represent as a sum of 2 primes
- Chomp P-positions & number of cookies
- Intersecting Set Systems & size of the family

Problem 2.

Given an object-type and a property, find a *theory* of the property.

- Graphs & hamiltonicity
- Matrices & total unimodularity
- Integers & primality
- Chomp positions & whether they are P-positions

Purpose of the talk

- To relate some experiments.
- To relate a program and available code that might be useful.
- To suggest that much more is possible.

Two Main Examples

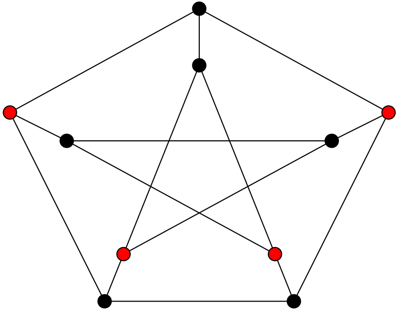
- How can we get better upper and lower bounds for the independence number of a graph?
- How can we get better necessary or sufficient conditions for the property of being Hamiltonian?

What do we do?

- If we want better bounds for the independence number we think about what bounds are known, what graphs are problematic, form conjectures as functions of usually-existing invariants, and check the conjectures against familiar graphs.

- If we want better necessary and/or sufficient conditions for the property of being Hamiltonian we think about what upper and lower bounds are known, what graphs are problematic, form conjectures as functions of usually-existing properties, and check the conjectures against familiar graphs.

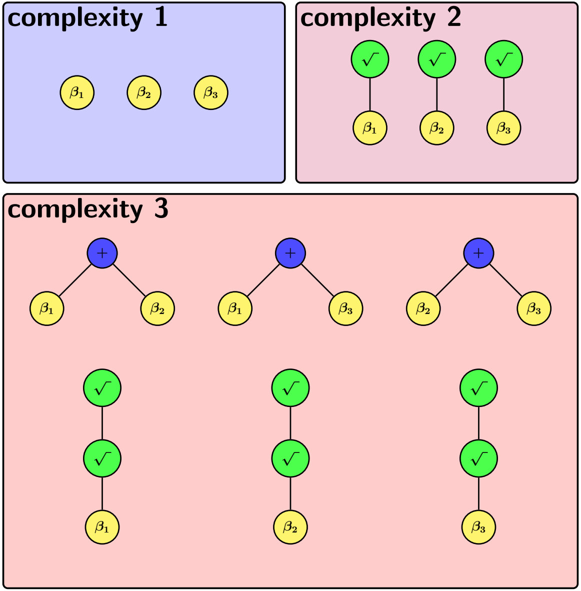
The Independence Number of a Graph



• The independence number α of a graph is the largest number of mutually non-adjacent vertices.

$$\alpha = 4.$$

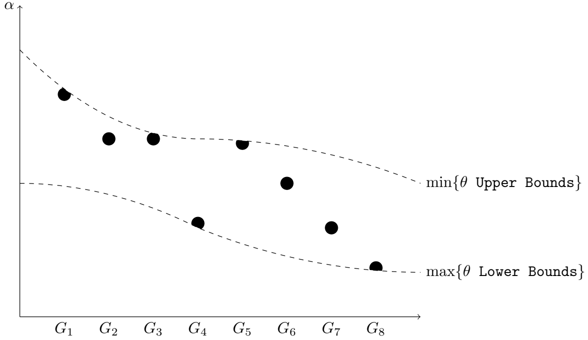
Generating Possible Bounds for an Invariant



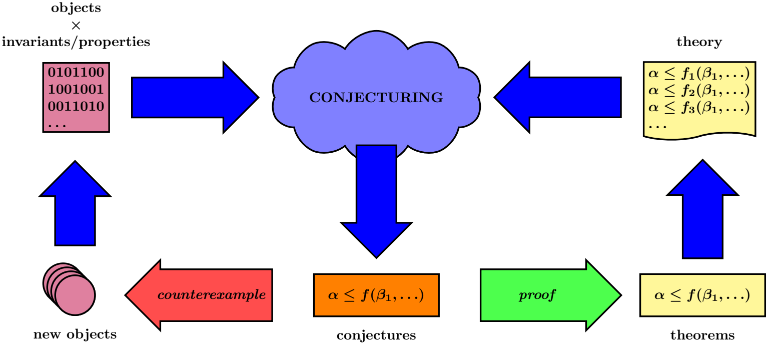
GRAFFITI Heuristics to Find New Bounds for an Invariant

- Generating expressions isn't enough.
- They need to be filtered somehow.
- Truth for examples is one filter.
- Fajtlowicz's Dalmatian heuristic: only store an expression/statement if it gives a better bound for at least one stored object.

GRAFFITI Heuristics to Find New Bounds for an Invariant



The CONJECTURING Process



Graph Theory Coding

- 112 efficiently computable properties, 36 intractable properties.
- 585+ graphs (and various collections: Sloane, DIMACS, pebbling)
- 127 efficiently computable invariants, and 33 intractable invariants.
- Database of values of (most of) these.

The THEORY variable

- Ideally we want conjectures that are not implied by existing theory (theoretical bounds, known bounds),
- that is, conjectures that give a better bound for at least one graph,
- so, for us, at least one graph in our database.
- We call this the `theory` input.

Best Lower Bounds for Independence

- $\alpha \geq$ radius.
- $\alpha \geq$ residue.
- $\alpha \geq$ critical independence number
- $\alpha \geq$ `max_even_minus_even_horizontal`

A Conjectured Lower Bound Theorem

Theorem

For any graph G, $\alpha(G) \geq \Delta(G) - T(G)$.

$\Delta(G) =$ maximum degree, $T(G) =$ number of triangles.

Proof.

Assume the statement is true for graphs with fewer than m edges. Let G be a graph with m edges and v be a vertex of maximum degree. It is easy to see that the conjecture is true in any case where $T(G) = 0$. We can assume there is an edge e not incident to v in some triangle. Let G' be the graph formed by removing edge e (but not its incident vertices). So, by assumption, $\alpha(G') \geq \Delta(G') - T(G')$. We see that $\alpha(G') - 1 \leq \alpha(G)$, $\Delta(G') = \Delta(G)$ and that $T(G') + 1 \leq T(G)$. Then $\alpha(G) \geq \alpha(G') - 1 \geq (\Delta(G') - T(G')) - 1 \geq \Delta(G) - (T(G) - 1) - 1 = \Delta(G) - T(G)$.

□

An Open Lower Bound Conjecture

$\alpha \geq$ `min(girth, floor(lovasz_theta))`

Equivalently, $\alpha \geq$ `girth` or $\alpha =$ `floor(lovasz_theta)`

Best Upper Bounds for Independence

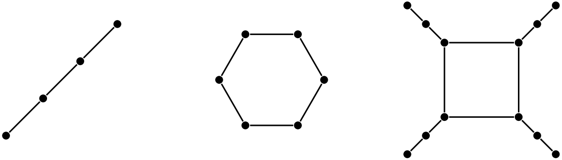
- $\alpha \leq$ annihilation number
- $\alpha \leq$ fractional independence number
- $\alpha \leq$ Lovász number
- $\alpha \leq$ Cvetković bound
- $\alpha \leq$ order - matching number.
- $\alpha \leq$ Hansen-Zheng bound.

(The *Hansen-Zheng bound* is
$\lfloor \frac{1}{2} + \sqrt{\frac{1}{4} + \text{order}^2 - \text{order} - 2 \cdot \text{size}} \rfloor$.)

A Conjectured Upper Bound Theorem

Theorem

For any connected graph, $\alpha \leq$ order $-$ radius.



r-ciliates: $C_{1,1}$, $C_{3,0}$, $C_{2,2}$

A Conjectured Upper Bound Theorem

Theorem

For any connected graph, $\alpha \leq$ *order* $-$ *radius.*

Proof.

Let G be a connected graph with radius r, and r-ciliate $C_{p,q}$ (with $r = p + q$). Note that an r-ciliate is bipartite. It is easy to check that $n(C_{p,q}) = 2p(q+1)$, $\alpha(C_{p,q}) = p(q+1)$, and $\alpha(C_{p,q}) \leq n(C_{p,q}) - r(C_{p,q})$.

Let $V' = V(G) \setminus V(C_{p,q})$, and $n' = |V'|$. Then

$\alpha(G) \leq \alpha(C_{p,q}) + n' \leq (n(C_{p,q}) - r(C_{p,q})) + n' = (n(G) - n') - r(G) + n' = n(G) - r(G)$. □

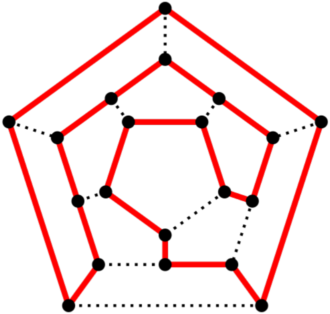
An Open Upper Bound Conjecture

$\alpha \leq$ `(average_distance)^(degree_sum)`

- Tested on all graphs of order ≤ 10.
- Tested on Random Graphs of all orders up to order 120.

Graph Hamiltonicity

A Hamiltonian cycle in a graph is a cycle that covers all of the vertices of the graph.



Necessary Conditions for Hamiltonicity

- If a graph is hamiltonian then it is 2-connected.
- If a graph is hamiltonian then it is *van den heuvel* (Laplacian eigenvalues condition).

A Conjectured Theorem

Thm. `(is_hamiltonian)->((is_cubic)->(is_class1))`

If is a graph is hamiltonian then if it is cubic it is hamiltonian.

If a graph is hamiltonian then either it is not cubic or it is class 1.

If a graph is hamiltonian and cubic then it is class 1.

Sufficient Conditions for Hamiltonicity

(Dirac) If the minimum degree of a graph is at least half the order then the graph is hamiltonian.

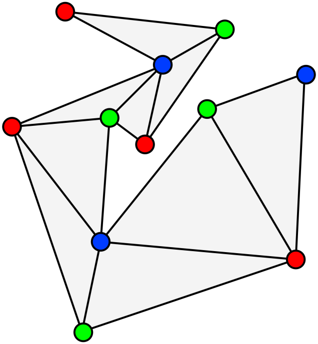
(Note: all graphs are assumed to be connected and have at least 3 vertices.)

(Ore) If the sum of the degrees of any pair of non-adjacent vertices is at least n then the graph is hamiltonian.

(Chvatal-Erdős) If the vertex connectivity of a graph is at least the independence number then the graph is hamiltonian.

Three Conjectured Theorems

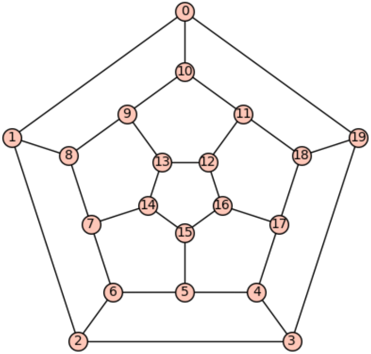
Thm. ((is_two_connected) & (is_circular_planar))->(is_hamiltonian)



Three Conjectured Theorems

Thm. `(is_planar_transitive)->(is_hamiltonian)`

If a graph is planar and vertex-transitive then it is hamiltonian.



Three Conjectured Theorems

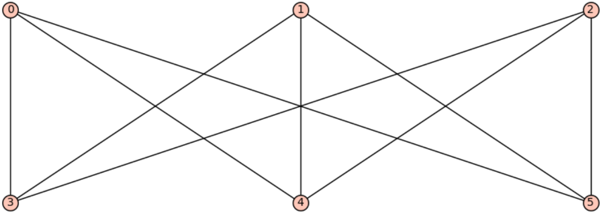
Thm. `(is_planar_transitive)->(is_hamiltonian)`

If a graph is planar and vertex-transitive then it is hamiltonian.

1. Every vertex-transitive graph is regular.
2. (Mader, 1970) If a graph is d-regular vertex-transitive with connectivity κ then $\frac{2(d+1)}{3} \leq \kappa$.
3. (Tutte, 1956) Every 4-connected planar graph is Hamiltonian.
4. (Zelinka, 1977) If a graph is planar, vertex-transitive and 3-regular then it is one of 8 specific graphs or an n-sided prism.
5. Only need to check the prisms!

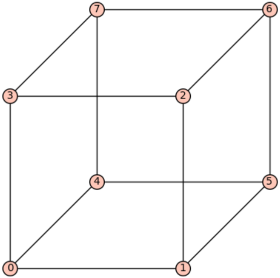
Three Conjectured Theorems

Thm. ((is_bipartite) & (is_strongly_regular))->(is_hamiltonian)



An Open Hamiltonicity Conjecture

Conj. ((is_bipartite) &
(is_distance_regular))->(is_hamiltonian)



CONJECTURING program inputs

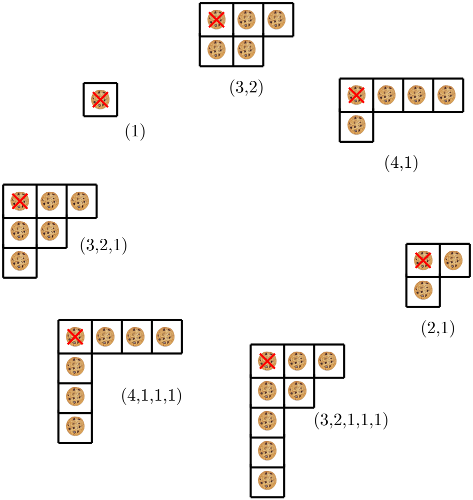
Inputs:

- Examples of objects.
- Definitions of invariants (or properties) for these objects.
- An Invariant (or property) you want bounds for.
- Whether you want upper or lower bounds.
- Any known Theorems (theoretical bounds).

```
#Run 4 of Day 3

current_graph_objects = [k3,pete,c5,k5_5,k3_4,EH,c7_chord,bow_tie,k5,p3,glasses,fish,c5_tail,triangle_with_
current_graph_objects.append(blanusa2)
current_graph_objects.append(frucht)
current_graph_objects.append(heawood)

#properties = [Graph.is_hamiltonian, Graph.is_clique,
Graph.is_regular,Graph.is_cycle,Graph.is_bipartite,Graph.is_chordal,Graph.is_strongly_regular,Graph.is_eule
Graph.is_triangle_free, Graph.is_distance_regular, Graph.is_perfect, Graph.is_planar]

#the properties list used in the conjecturing program will be the on from gt.sage

property_of_interest = properties.index(Graph.is_hamiltonian)

theorem1 = lambda g: g.is_bipartite() and g.is_strongly_regular()

theorems = [Graph.is_cycle, Graph.is_clique, theorem1]

conjs = propertyBasedConjecture(current_graph_objects, properties, property_of_interest, theory = theorems)
for c in conjs:
    print c
```

```
((is_planar)&(is_regular))->(is_hamiltonian)
((is_gallai_tree)^(is_chordal))->(is_hamiltonian)
((is_perfect)&(is_distance_regular))->(is_hamiltonian)
```

Bounds for Chomp invariants



Bounds for Chomp invariants

Conjectured Theorem:

For any position where the *previous-player-to-play* has a winning strategy (a P-position),

the number of cookies on the board $\geq$ 2* the number of (non-empty) columns -1.

Number Theory—Goldbach's Conjecture

For any even integer $x > 3$ let Goldbach(x) be the number of ways x can be written as a sum of two primes.

Goldbach(x) $\geq$ 1/digits10(x)

Goldbach(x) $\geq$ digits10(x) - 1

Matrix Theory—Determinants of Symmetric Matrices

determinant(x) $\leq$ permanent(x)
determinant(x) $\leq$ maximum_eigenvalue(x)*trace(x)
determinant(x) $\leq$ (rank(x) $+$ 1)*spectral_radius(x)

determinant(x) $\geq$ minimum_eigenvalue(x)*separator(x)
determinant(x) $\geq$ minimum(permanent(x), log(nullity(x)))

Integer Sequences

$$\text{input\_sequence} = [1, 3, 4, 7, 11]$$

Integer Sequences

$$\text{input\_sequence} = [1, 3, 4, 7, 11]$$

$$\text{last\_term(x)} \geq \text{average\_difference(x)} + 1$$
$$\text{last\_term(x)} \geq \text{previous\_term(x)} + 1$$
$$\text{last\_term(x)} \geq \text{min(sum\_of\_previous\_two(x), 2*previous\_term(x))}$$

$$\text{last\_term(x)} \leq \text{sum\_of\_previous\_two(x)}$$
$$\text{last\_term(x)} \leq \text{2*previous\_term(x)} + 1$$

Integer Sequences

$$\text{input\_sequence} = [100, 104, 108]$$

Integer Sequences

$$\text{input\_sequence} = [100, 104, 108]$$

$$\text{last\_term}(x) \geq \text{average\_difference}(x) + \text{previous\_term}(x)$$

$$\text{last\_term}(x) \leq \text{average\_difference}(x) + \text{previous\_term}(x)$$

Integer Sequences

$\text{input\_sequence} = [100, 104, 108]$

$\text{last\_term}(x) \geq \text{average\_difference}(x) + \text{previous\_term}(x)$

$\text{last\_term}(x) \leq \text{average\_difference}(x) + \text{previous\_term}(x)$

$[100, 104, 108, 112]$

Integer Sequences

$$\text{input\_sequence} = [1, 3, 9, 27, 81]$$

Integer Sequences

$$\text{input\_sequence} = [1, 3, 9, 27, 81]$$

last_term(x) $\geq$ average_ratio(x)*previous_term(x)
last_term(x) $\geq$ average_ratio(x)

last_term(x) $\leq$ average_ratio(x) ^ previous_term(x)
last_term(x) $\leq$ average_ratio(x)*previous_term(x)

Integer Sequences

$$\text{input\_sequence} = [1, 3, 9, 27, 81]$$

$$\text{last\_term}(x) \geq \text{average\_ratio}(x){*}\text{previous\_term}(x)$$
$$\text{last\_term}(x) \geq \text{average\_ratio}(x)$$

$$\text{last\_term}(x) \leq \text{average\_ratio}(x) \wedge \text{previous\_term}(x)$$
$$\text{last\_term}(x) \leq \text{average\_ratio}(x){*}\text{previous\_term}(x)$$

$$[1, 3, 9, 27, 81, 243]$$

Thank You!

Automated Conjecturing in Sage:
nvcleemp.github.io/conjecturing/

Graph Brain Project:
github.com/math1um/objects-invariants-properties

clarson@vcu.edu