An Empty Hexagon in Every Set of 30 Points

Marijn J.H. Heule

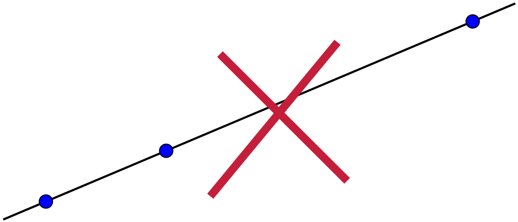
joint work with Manfred Scheucher

30th International Conference on Tools and Algorithms
for the Construction and Analysis of Systems

Luxembourg April 8, 2024

Points in General Position

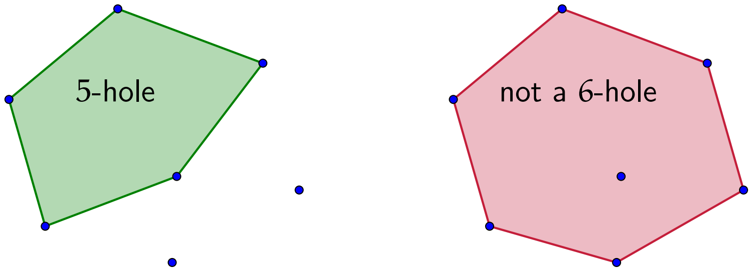
A finite point set S in the plane is in general position if no three points in S are on a line.



Throughout this talk, every set is in general position

k-Holes

A k-hole (in S) is a convex k-gon containing no other points of S



$h(k)$: the smallest number of points that contain a k-hole

For k fixed, does every sufficiently large point set in general position contain k-holes?

k-Holes Overview

For k fixed, does every sufficiently large point set in general position contain k-holes?

- 3 points $\Rightarrow \exists$ 3-hole (trivial)
- 5 points $\Rightarrow \exists$ 4-hole [Klein '32]
- 10 points $\Rightarrow \exists$ 5-hole [Harborth '78]
- Arbitrarily large point sets with no 7-hole [Horton '83]

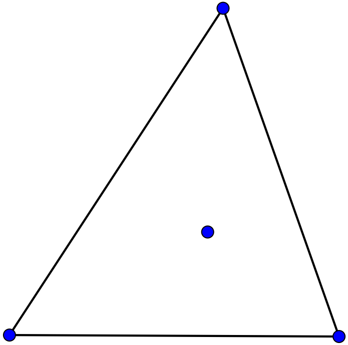
Main open question: what about 6-hole?

- Sufficiently large point sets contain a 6-hole [Gerken '08 and Nicolás '07, independently]
- Conjecture: $h(6) = 30$ (proved in TACAS'24 paper)

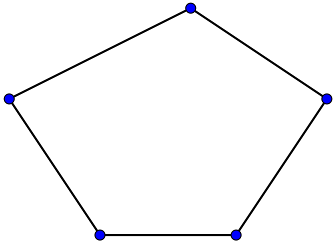
Lowerbound for 4-Hole: $h(4) > 4$

Clearly, any 3-point set in general position has a 3-hole

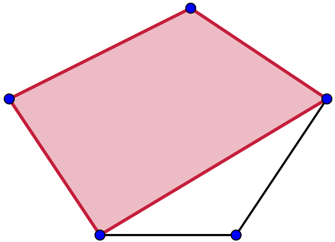
Some sets with four points have no 4-hole, so $h(4) > 4$:



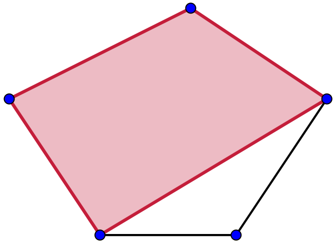
Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]

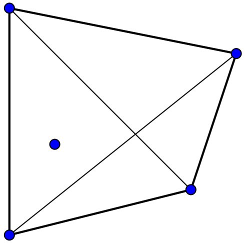


Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]

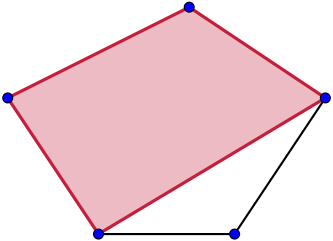


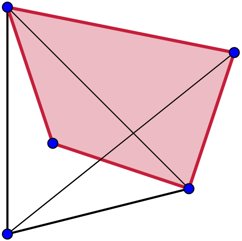
Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]



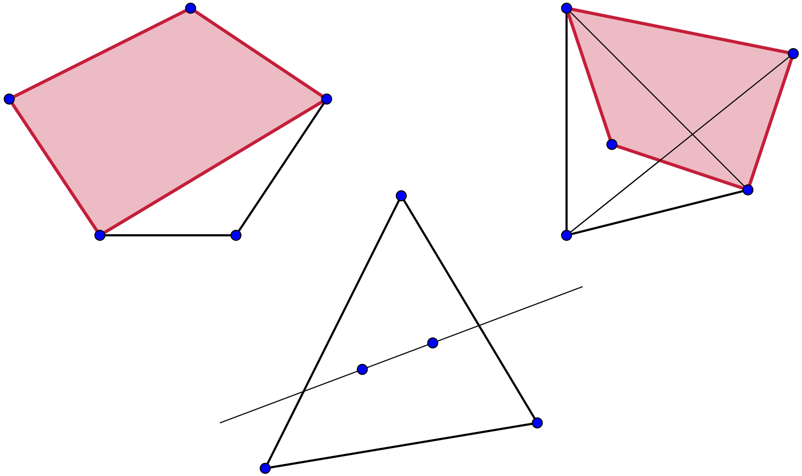


Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]

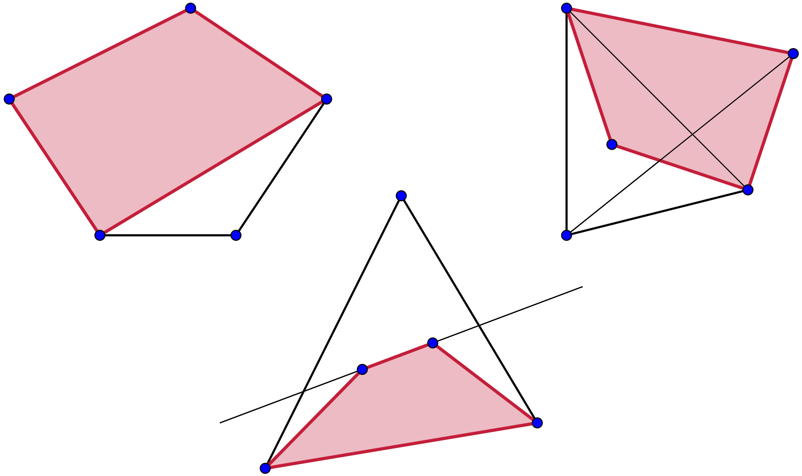




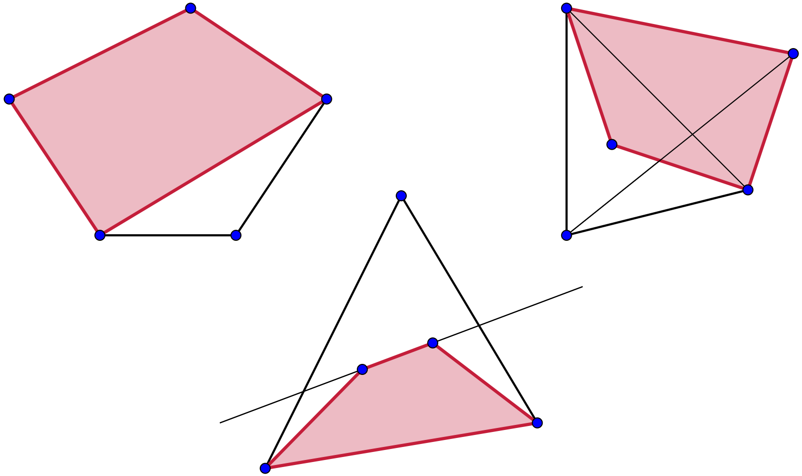
Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]



Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]

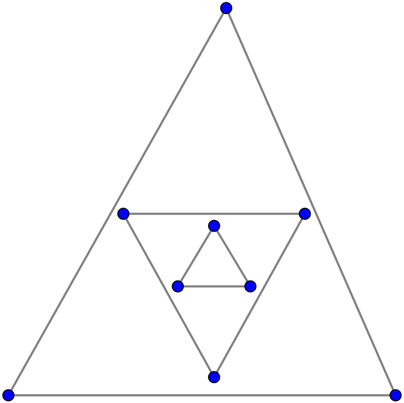


Upperbound for 4-Hole: $h(4) = 5$ [Klein, 1930s]



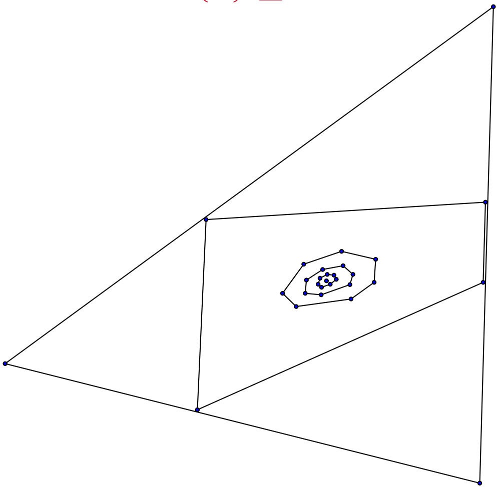
Happy ending problem

Lowerbound for 5-Hole: $h(5) \geq 10$



All 5-gons in these 9 points have an inner point: $h(5) = 10$

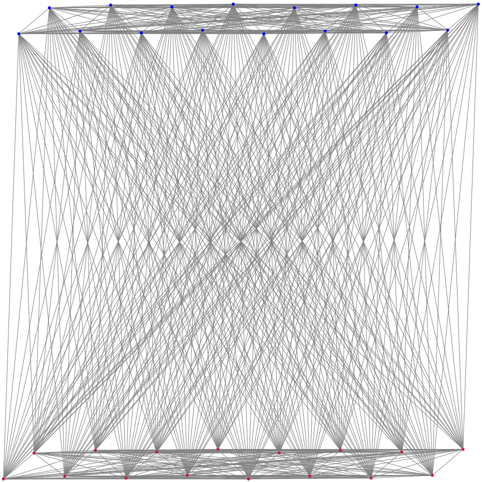
Lowerbound for 6-Hole: $h(6) \geq 30$



29 points, no 6-hole [Overmars '02]

- Found using simulated annealing... is now easy using SAT
- This contains 7-gons. Each 9-gon contains a 6-hole

No Lowerbound for 7-Hole: Horton's Construction

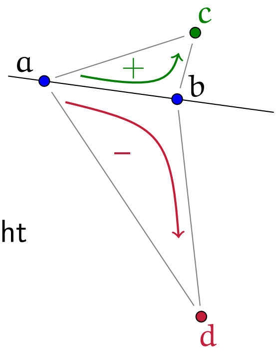


2^5 points, no 7-hole

Orientation Variables

No explicit coordinates of points

Instead for every triple $a < b < c$,
one orientation variable $O_{a,b,c}$ to denote whether point c is above the line ab



Triple orientations are enough to express k-gons and k-holes

WLOG points are sorted from left to right

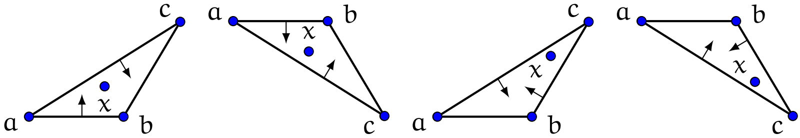
Not all assignments are realizable

- Realizability is hard [Mnëv '88]
- Additional clauses eliminate many unrealizable assignments

Inside Variables

We introduce inside variables $I_{x;a,b,c}$ which are true if and only if point x is in the triangle abc with $a < x < b$ or $b < x < c$.

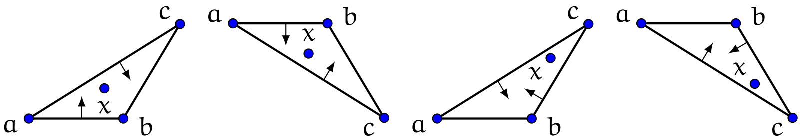
Four possible cases:



Inside Variables

We introduce inside variables $I_{x;a,b,c}$ which are true if and only if point x is in the triangle abc with $a < x < b$ or $b < x < c$.

Four possible cases:



The left two cases with $a < x < b$:

$$I_{x;abc} \leftrightarrow \Big(\big(O_{abc} \rightarrow (\overline{O_{axb}} \wedge O_{axc})\big) \wedge \big(\overline{O_{abc}} \rightarrow (O_{axb} \wedge \overline{O_{axc}})\big) \Big)$$

The right two cases with $b < x < c$:

$$I_{x;abc} \leftrightarrow \Big(\big(O_{abc} \rightarrow (O_{axc} \wedge \overline{O_{bxc}})\big) \wedge \big(\overline{O_{abc}} \rightarrow (\overline{O_{axc}} \wedge O_{bxc})\big) \Big)$$

Hole Variables

We introduce hole variables H_{abc} which are true if and only if no points occur with the triangle abc with $a < b < c$.

$$\bigwedge_{a<x<c} \overline{I_{x;abc}} \rightarrow H_{abc}$$

Simple 6-hole encoding:

$$\bigvee_{a,b,c \in X} \overline{H_{abc}} \quad \forall\, X \subset S \text{ with } |X| = 6$$

6-Hole Encoding: One Triangle-is-Empty Check Required

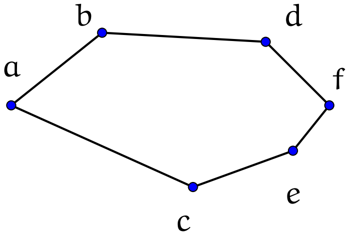
Trusted 6-hole encoding uses $O(n^6)$ clauses with 20 literals:

$$\bigvee_{a,b,c \in X} \overline{H_{abc}} \quad \forall\, X \subset S \text{ with } |X| = 6$$

Example

Consider an assignment with

- $O_{abd} = 0$ and $O_{bdf} = 0$
- $O_{ace} = 1$ and $O_{cef} = 1$



6-Hole Encoding: One Triangle-is-Empty Check Required

Trusted 6-hole encoding uses $O(n^6)$ clauses with 20 literals:

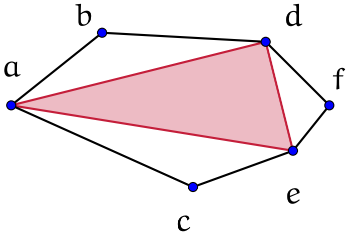
$$\bigvee_{a,b,c \in X} \overline{H_{abc}} \quad \forall\, X \subset S \text{ with } |X| = 6$$

Example

Consider an assignment with

- $O_{abd} = 0$ and $O_{bdf} = 0$
- $O_{ace} = 1$ and $O_{cef} = 1$
- $H_{ade} = 1$

This implies the existence of a 6-hole!



6-Hole Encoding: One Triangle-is-Empty Check Required

Trusted 6-hole encoding uses $O(n^6)$ clauses with 20 literals:

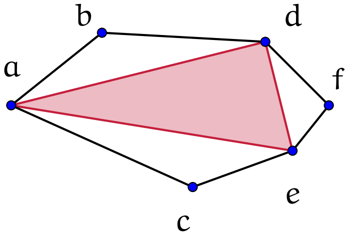
$$\bigvee_{a,b,c \in X} \overline{H_{abc}} \quad \forall\, X \subset S \text{ with } |X| = 6$$

Example

Consider an assignment with

- $O_{abd} = 0$ and $O_{bdf} = 0$
- $O_{ace} = 1$ and $O_{cef} = 1$
- $H_{ade} = 1$

This implies the existence of a 6-hole!



Clause to prevent this: $O_{abd} \vee O_{bdf} \vee \overline{O_{ace}} \vee \overline{O_{cef}} \vee \overline{H_{ade}}$

6-Hole Encoding: One Triangle-is-Empty Check Required

Trusted 6-hole encoding uses $O(n^6)$ clauses with 20 literals:

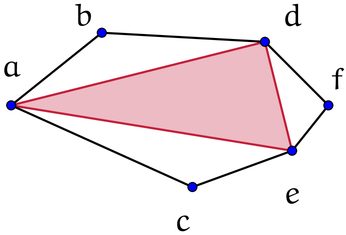
$$\bigvee_{a,b,c \in X} \overline{H_{abc}} \quad \forall\, X \subset S \text{ with } |X| = 6$$

Example

Consider an assignment with

- $O_{abd} = 0$ and $O_{bdf} = 0$
- $O_{ace} = 1$ and $O_{cef} = 1$
- $H_{ade} = 1$

This implies the existence of a 6-hole!



Clause to prevent this: $O_{abd} \vee O_{bdf} \vee \overline{O_{ace}} \vee \overline{O_{cef}} \vee \overline{H_{ade}}$

This encoding is 5 times larger, but much easier to solve

k-Hole Encoding Using $O(n^4)$ Clauses

Shorter clauses, thus more propagation, but still $O(n^6)$

Example

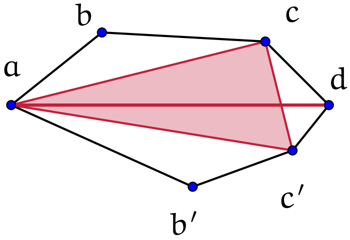
Introduce $O(n^3)$ auxiliary variables:

- A_{acd}: a 4-gon above the line ad

  $\overline{O_{abc}} \wedge \overline{O_{bcd}} \rightarrow A_{acd}$
- $B_{ac'd}$: a 4-gon below the line ad

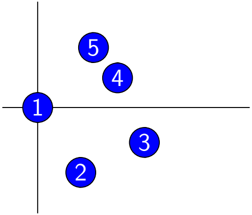
  $O_{ab'c'} \wedge O_{b'c'd} \rightarrow B_{ac'd}$
- Combine them to block 6-holes

  $\overline{A_{acd}} \vee \overline{B_{ac'd}} \vee \overline{H_{acc'}}$

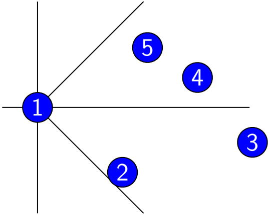


This reduces the size of the encoding to $O(n^4)$ clauses

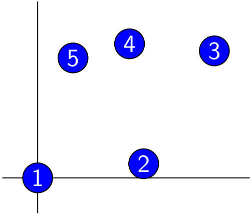
Symmetry Breaking: Sorted & Rotated Around Point 1



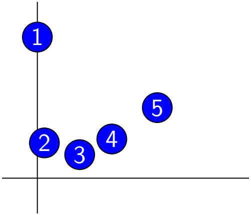
place leftmost point at origin



stretch points to the right to be within $y = x$ and $y = -x$



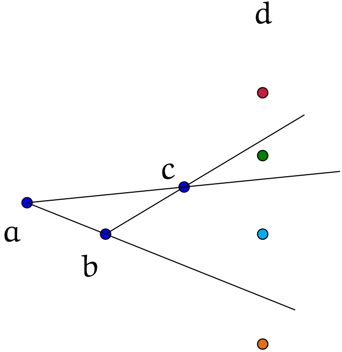
rotate by 45 degrees



projective transformation $(x, y) \mapsto (y/(x + \epsilon), 1/(x + \epsilon))$

Realizability Constraints

Under the assumption that points are sorted from left to right



O_{abc}	O_{abd}	O_{acd}	O_{bcd}
+	+	+	+
+	+	+	−
+	+	−	−
+	−	−	−
−	−	−	−
−	−	−	+
−	−	+	+
−	+	+	+

Block multiple sign changes with $\Theta(n^4)$ (ternary) clauses
[Felsner & Weil '01]

Impact of the Encoding

Four different encodings of a random subproblem

- T: the trusted encoding
- O_1: the explicit encoding with a single empty triangle
- O_2: reduce the size of O_1 with auxiliary variables to $O(n^4)$
- O_3: O_2 without redundant clauses

Γ	#var	#clause	#conflict	#propagation	time (s)
T	62 930	1 171 942	1 082 569	1 338 662 627	243.07
O_1	62 930	5 823 078	228 838	282 774 472	136.20
O_2	75 110	667 005	211 272	343 388 591	45.49
O_3	75 110	436 047	234 755	340 387 692	39.46

Problem Partitioning

Partitioning to split the problem into easier subproblems

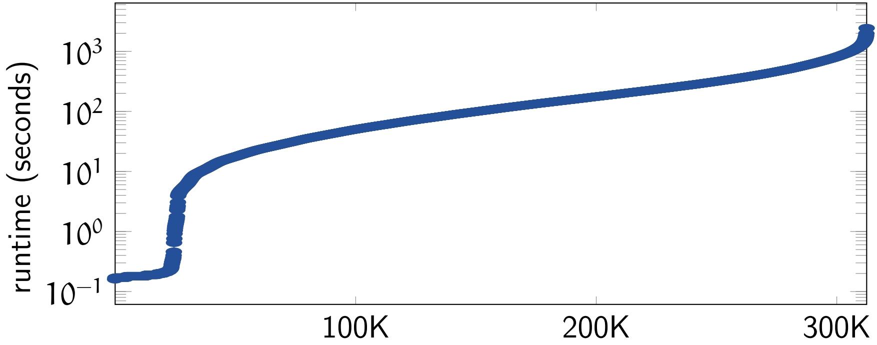
- Original problem UNSAT iff all subproblems UNSAT
- Split on variables $O_{a,a+1,a+2}$ starting from the middle
- One parameter: the length ℓ, roughly 1.83^{ℓ} cubes
- Tested on: 24 points contain 6-hole or 7-gon

ℓ	#cubes	avg time (s)	max time (s)	total (h)
21	312 418	6.99	66.86	606.55
19	89 384	13.61	123.70	337.96
17	25 663	34.29	293.10	244.50
15	7393	112.61	949.50	**231.27**
13	2149	431.26	3 347.59	257.44
11	629	1 847.46	11 844.05	322.79
9	188	7 745.14	32 329.05	404.47
7	57	32 905.90	105 937.76	521.01

Empty Hexagon Theorem Summary

Theorem: $h(6) = 30$

- Partitioned problem using 312 418 cubes ($\ell = 21$)
- Total runtime: 17 000 CPU hours on AWS
- Linear speedups using 1 000 machines
- Proof: 180 terabytes in unprocessed LRAT format
- Validated with formally-verified checker



Verification

The optimization steps are validated or part of the proof

Concurrent solving and proof checking for the first time

- The solver pipes the proof to a verified checker
- This avoids storing/writing/reading huge files
- Verified checker can easily catch up with the solver

CMU students have formalized and verified all parts in Lean

- Paper submitted to ITP '24

Conclusions

Theorem
$h(6) = 30$

SAT appears to be the most effective technology to solve a range of problems in computational geometry

Many interesting open problems:

- Minimum number of 4-gons / 5-gons / 6-gons
- Determine whether $g(7) = 33$
- Unbalanced configurations (points can be collinear)