Recent advances on algorithms for geodetic sets

D. C.

Joint work with
Sandip Das, Florent Foucaud, Harmendar Gahlawat, Dimitri Lajou, Bodhayan Roy

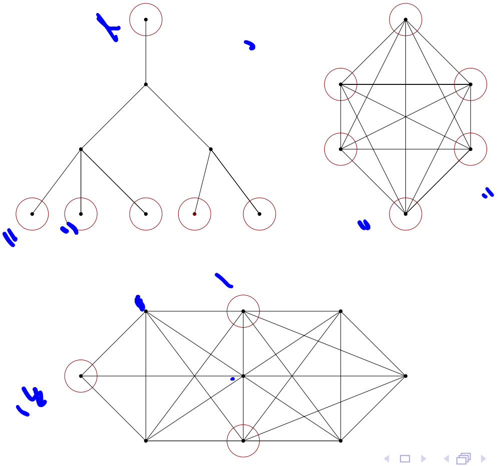
Metric Graph Theory

December 9, 2021

Definition

A set S of vertices is a *geodetic set* if any vertex $v \in V(G)$ lies in some $I(x, y)$ where $\{x, y\} \subseteq S$.

MINIMUM GEODETIC SET (MGS) is to find decide whether a graph G has a geodetic set of size at most k?



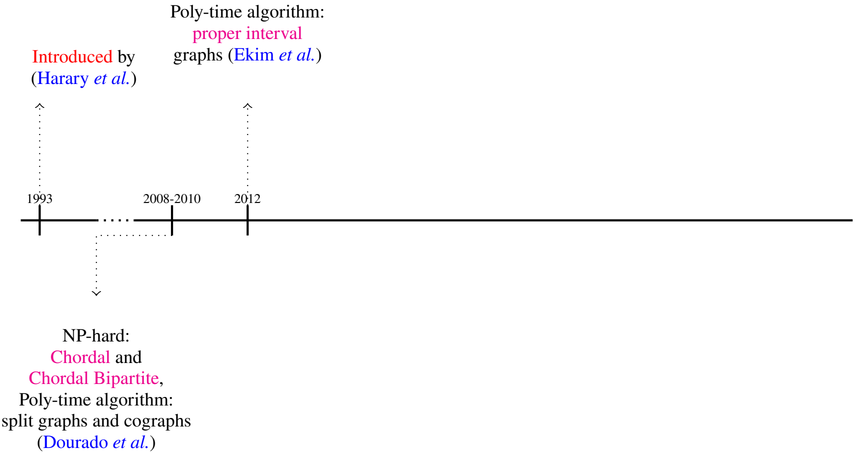
Timeline of algorithmic results

Introduced by
(Harary *et al.*)

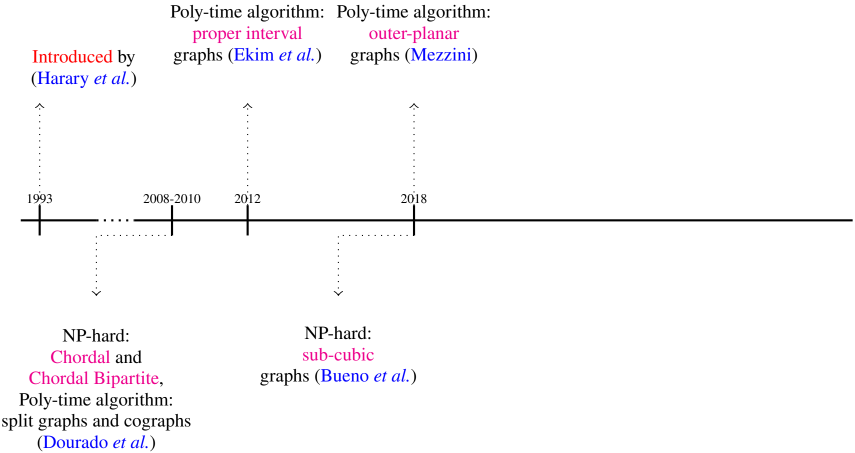
1993

Timeline of algorithmic results

Timeline of algorithmic results



Timeline of algorithmic results



Timeline of algorithmic results

1993: Introduced by (Harary *et al.*)

2008-2010: NP-hard: Chordal and Chordal Bipartite, Poly-time algorithm: split graphs and cographs (Dourado *et al.*)

2012: Poly-time algorithm: proper interval graphs (Ekim *et al.*)

2018: Poly-time algorithm: outer-planar graphs (Mezzini)

2018: NP-hard: sub-cubic graphs (Bueno *et al.*)

2020: W[1] hard: $(fv + pw + k)$ FPT : fe (Kellerhals *et al.*)

2020: NP-hard: line graphs $O(\sqrt[3]{n}\log n)$-approximation (Ch. *et al.*)

Timeline of algorithmic results

1993

Introduced by
(Harary *et al.*)

2008-2010

NP-hard:
Chordal and
Chordal Bipartite,
Poly-time algorithm:
split graphs and cographs
(Dourado *et al.*)

2012

Poly-time algorithm:
proper interval
graphs (Ekim *et al.*)

2018

Poly-time algorithm:
outer-planar
graphs (Mezzini)

NP-hard:
sub-cubic
graphs (Bueno *et al.*)

2020

W[1] hard:
$(f\ v + pw + k)$
FPT : $f\ e$
(Kellerhals *et al.*)

NP-hard: line graphs
$O(\sqrt[3]{n}\log n)$-approximation
(Ch. *et al.*)

Results
of this talk.

Timeline of algorithmic results

Introduced by
(Harary *et al.*)

1993

Poly-time algorithm:
proper interval
graphs (Ekim *et al.*)

2012

Poly-time algorithm:
outer-planar
graphs (Mezzini)

2018

W[1] hard:
$(fv + pw + k)$
~~FPT : fe~~
(Kellerhals *et al.*)

2020

Log-APX hard:
sub-cubic bipartite graphs
(Davot *et al.*)

2021

2008-2010

NP-hard:
Chordal and
Chordal Bipartite,
Poly-time algorithm:
split graphs and cographs
(Dourado *et al.*)

NP-hard:
sub-cubic
graphs (Bueno *et al.*)

NP-hard: line graphs
$O(\sqrt[3]{n}\log n)$-approximation
(Ch. *et al.*)

Results
of this talk.

Results

Theorem

There is a linear time algorithm for MGS on solid grids.

Theorem

MGS is NP-hard on sub-cubic patial grids.

Theorem

MGS is NP-hard on interval graphs with no induced $K_{1,5}$.

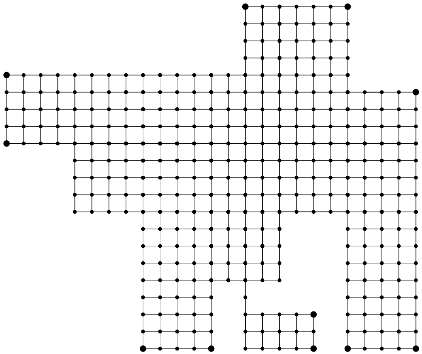
Theorem

MGS can be solved in time $O\left(2^{2^{\omega}} n^{O(1)}\right)$ for chordal graphs and in time $O\left(2^{\omega} n^{O(1)}\right)$ for interval graphs, where n and ω are the order and clique number of the input graph, respectively.

Solid grids

Theorem

There is a linear time algorithm for MGS on solid grids.



A graph is a *solid grid* if it has a grid embedding such that all interior faces have unit area.

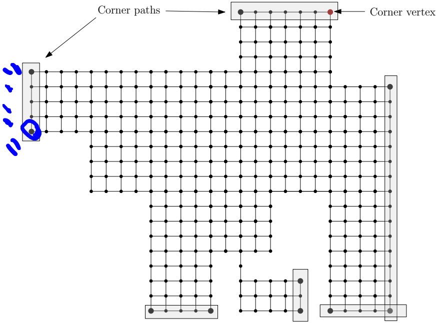
Solid grids (Proof sketch)

A path P of G is a *corner path* if

(i) no vertex of P is a cut-vertex,

(ii) both end-vertices of P have degree 2, and

(iii) all other vertices of P have degree 3.



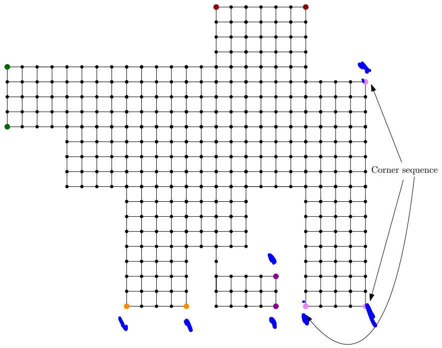
Lemma

Any geodetic set contains at least one vertex from each corner path.

Solid grids (Proof sketch)

Definition

We say that $u_1, u_2, \ldots, u_k$ forms a *corner sequence* if for each $1 \leq i \leq k-1$,
1. there is a corner path with u_i and u_{i+1} as endpoints, and
2. there is no corner vertex in the clockwise traversal of the boundary of the grid embedding from u_i to u_{i+1}.



Solid grids (Proof sketch)

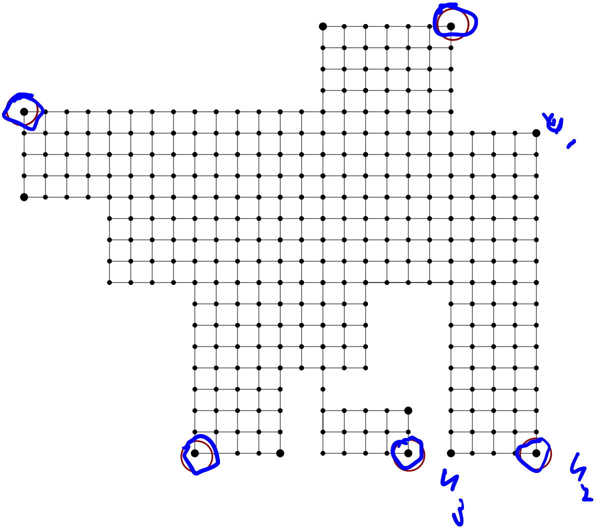
Lemma

Let $\mathcal{S}$ be the set of all maximal corner sequences of a solid grid G, and let t be the number of vertices of G with degree 1. Then, $gn(G) \geq t + \sum_{S \in \mathcal{S}} \lfloor |S|/2 \rfloor$.

Algorithm:

- Choose all vertices with degree 1,
- Traverse the embedding in the clockwise direction, and for each maximal corner sequence, choose vertices alternatively.

Solid grids (Proof sketch)



Lemma

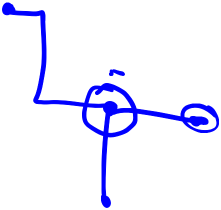
The set of vertices chosen by the above algorithm is a geodetic set.

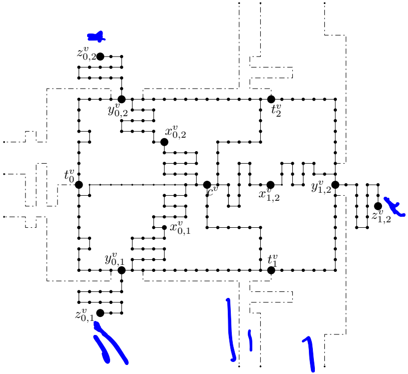
Partial grids

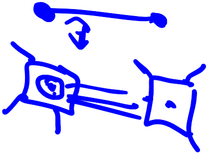
Theorem

MGS is NP-hard on sub-cubic patial grids.

Reduce from MINIMUM VERTEX COVER on cubic planar graphs.







Interval graphs

Theorem

MGS is NP-hard on interval graphs with no induced $K_{1,5}$.

Reduce from 3-SAT.

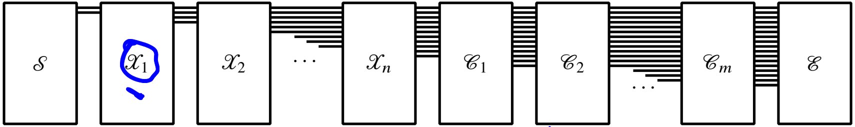


Figure: Overview of the construction. Each box represents one of the main gadget. Lines between such gadgets represent the tracks.

$\boxed{2^{O(n)}}$

Corollary

Assuming ETH, there is no $2^{o(\sqrt{n})}$-time algorithm for MGS on interval graphs of order n.

Chordal graphs

Theorem

MGS can be solved in time $O\left(2^{2^{\omega}} n^{O(1)}\right)$ *for chordal graphs and in time* $O\left(2^{\omega} n^{O(1)}\right)$ *for interval graphs, where* n *and* ω *are the order and clique number of the input graph, respectively.*

- Dynamic programming on the nice tree decomposition T of a chordal graph G.
- Width of T is ω.
- Each *bag* or *node* of T is a clique cut-set.

Chordal graphs (Proof sketch)

- With each bag X_v of T, associate $O(2^{2^{\omega}})$ many “types of partial solutions".

Chordal graphs (Proof sketch)

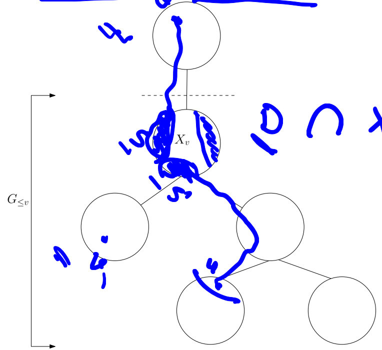
- With each bag X_v of T, associate $O(2^{2^{\omega}})$ many "types of partial solutions".

- A "type of partial solution" τ for a node X_v is a 4-tuple $(\tau^{ext}, \tau^{int}, \tau^{cov}, \tau^{bag})$ where
  - τ^{ext} and τ^{int} are collections of subsets of X_v,
  - τ^{cov}, τ^{bag} are subsets of X_v

Chordal graphs (Proof sketch)

- With each bag X_v of T, associate $O(2^{2^\omega})$ many "types of partial solutions".
- A "type of partial solution" τ for a node X_v is a 4-tuple $(\tau^{ext}, \tau^{int}, \tau^{cov}, \tau^{bag})$ where
  - τ^{ext} and τ^{int} are collections of subsets of X_v,
  - τ^{cov}, τ^{bag} are subsets of X_v

D *is a solution of type* $\tau = (\tau^{ext}, \tau^{int}, \tau^{cov}, \tau^{bag},)$ *if*



$\tau^{ext} = \{ \{\}, \{\}, \{\} \}$

$D \cap X_v = \tau^{bag}$

$\tau^{int} = \{ \{\}, \{\} \}$

S

S

t_0

Summary

Theorem

There is a linear time algorithm for MGS on solid grids.

Theorem

MGS is NP-hard on sub-cubic patial grids.

Theorem

MGS is NP-hard on interval graphs with no induced $K_{1,5}$.

Theorem

MGS can be solved in time $O\left(2^{2^{\omega}} n^{O(1)}\right)$ for chordal graphs and in time $O\left(2^{\omega} n^{O(1)}\right)$ for interval graphs, where n and ω are the order and clique number of the input graph, respectively.

Open problems

- Polynomial time algorithm for Series-parallel graphs.
- Improve the running time for Chordal graphs.
- Constant factor approximation for planar graphs, interval graphs.