Instituto de Computação
UNIVERSIDADE ESTADUAL DE CAMPINAS

UNICAMP

Realizing Graphs with Cut Constraints[a]

Lucas de Oliveira Silva[1]
Vítor Gomes Chagas[1]
Samuel Plaça de Paula[1]
Greis Yvet Oropeza Quesquén[1]
Uéverton dos Santos Souza[2,3]

[1] Unicamp, Campinas, Brazil
[2] IMPA, Rio de Janeiro, Brazil
[3] UFF, Niterói, Brazil

30th May 2025

[a] This work was started during the 6th edition of WoPOCA. We thank the organizers and the agencies CNPq (process number 404315/2023-2) and FAEPEX (process number 2422/23).

Classic Problems

Graph Realization Problem

GRAPH REALIZATION

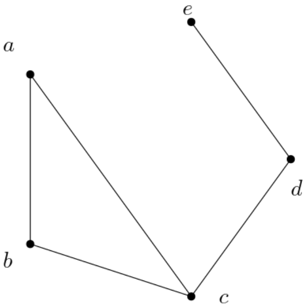
Input:	A non-decreasing sequence $\mathtt{d} = (d_1, \ldots, d_n)$ of natural numbers.
Question:	Is $\mathtt{d}$ a graphic sequence?

Example

Is $\mathtt{d} = (3, 2, 2, 2, 1)$ graphic?

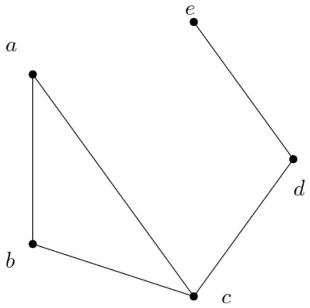
Example

Is $\texttt{d} = (3, 2, 2, 2, 1)$ graphic?

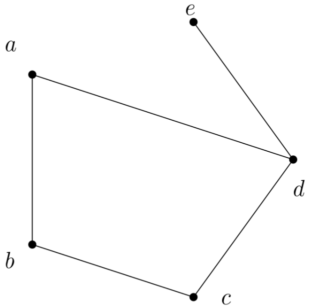


Example

Is $\mathtt{d} = (3, 2, 2, 2, 1)$ graphic?



or



Graphic Sequences Characterization

Theorem (Erdős and Gallai [EG60])

A non-decreasing sequence $\mathtt{d} = (d_1, \ldots, d_n)$ of natural numbers is graphic if and only if

1. *$\sum_{i=1}^{n} d_i$ is even, and*
2. *For every $1 \leq k \leq n$,*

$$\sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min\{d_i, k\}$$

Observations

- Erdős–Gallai gives a simple poly-time criterion;

Observations

- Erdős–Gallai gives a simple poly-time criterion;
- Many variations of the problem have been considered;

Observations

- Erdős–Gallai gives a simple poly-time criterion;
- Many variations of the problem have been considered;
- Vertex degree $=$ size of trivial edge cut;

Observations

- Erdős–Gallai gives a simple poly-time criterion;
- Many variations of the problem have been considered;
- Vertex degree = size of trivial edge cut;
- We generalize this by adding nontrivial cut-size constraints.

k-factors

A *k-factor* of a graph G is a k-regular spanning subgraph of G.

k-factors

A *k-factor* of a graph G is a k-regular spanning subgraph of G.

A matching is a 1-factor.

f-factors

A spanning subgraph H of G is a *f-factor*, where $f\colon V \to \mathbb{N}$, if $d_H(v) = f(v)$ for all $v \in V$.

f-factors

A spanning subgraph H of G is a f*-factor*, where $f\colon V \to \mathbb{N}$, if $d_H(v) = f(v)$ for all $v \in V$.

f-factors can be found in cubic time [Ans85].

Nontrivial Cut Constraints

Preliminaries

Fixed a graph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$

Preliminaries

Fixed a graph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$

- A *cut list* is a list of pairs $\mathcal{L} = \{(S_1, \ell_1), \ldots, (S_m, \ell_m)\}$;

Preliminaries

Fixed a graph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$

- A *cut list* is a list of pairs $\mathcal{L} = \{(S_1, \ell_1), \ldots, (S_m, \ell_m)\}$;
- For each j we have $\emptyset \neq S_j \subsetneq V$ and $\ell_j \in \mathbb{N}$;

Preliminaries

Fixed a graph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$

- A *cut list* is a list of pairs $\mathcal{L} = \{(S_1, \ell_1), \ldots, (S_m, \ell_m)\}$;
- For each j we have $\emptyset \neq S_j \subsetneq V$ and $\ell_j \in \mathbb{N}$;
- *G realizes* $\mathcal{L}$ if $|\partial(S_j)| = \ell_j$ for every $(S_j, \ell_j) \in \mathcal{L}$;

Preliminaries

Fixed a graph $G = (V, E)$ where $V = \{v_1, \dots, v_n\}$

- A *cut list* is a list of pairs $\mathcal{L} = \{(S_1, \ell_1), \dots, (S_m, \ell_m)\}$;
- For each j we have $\emptyset \neq S_j \subsetneq V$ and $\ell_j \in \mathbb{N}$;
- *G realizes* $\mathcal{L}$ if $|\partial(S_j)| = \ell_j$ for every $(S_j, \ell_j) \in \mathcal{L}$;
- By $w(\mathcal{L})$ we denote $\max_j \; |S_j|$.

New Graph Realization Problem

Graph Realization with Cut Constraints (GR-C)

Input: A cut list $\mathcal{L}$ for a set of vertices $V = \{v_1, \ldots, v_n\}$, and a non-decreasing sequence $\mathtt{d} = (d_1, \ldots, d_n)$ of natural numbers.

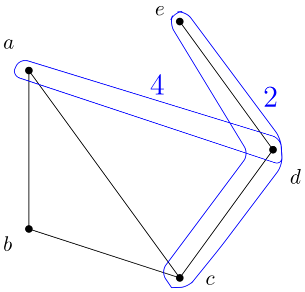
Question: Does there exist a simple graph $G = (V, E)$ such that, for every j, $d(v_j) = d_j$ and G realizes $\mathcal{L}$?

Running Example

Consider $\mathtt{d} = (3, 2, 2, 2, 1)$ and $\mathcal{L} = \{(\{a, d\}, 4), (\{c, d, e\}, 2)\}$:

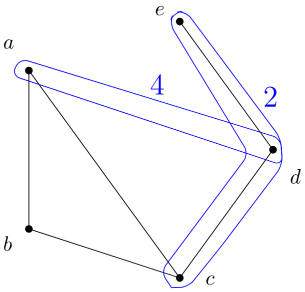
Running Example

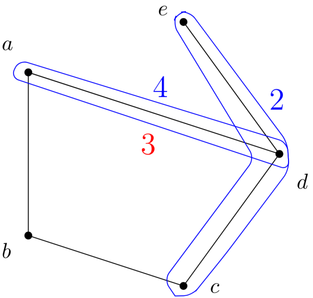
Consider $\mathtt{d} = (3, 2, 2, 2, 1)$ and $\mathcal{L} = \{(\{a, d\}, 4), (\{c, d, e\}, 2)\}$:



Running Example

Consider $\mathtt{d} = (3, 2, 2, 2, 1)$ and $\mathcal{L} = \{(\{a, d\}, 4), (\{c, d, e\}, 2)\}$:





Observations

- If $\mathcal{L} = \emptyset$, we have the classic realization problem;

Observations

- If $\mathcal{L} = \emptyset$, we have the classic realization problem;
- We can see GR-C as a consistency check for cut-queries in learning an unknown graph.

Important Remark

For $S \subseteq V$ let $d(S) = \sum\limits_{u \in S} d(u)$.

Important Remark

For $S \subseteq V$ let $d(S) = \sum_{u \in S} d(u)$.

Remark (1)

An instance $(d, \mathcal{L})$ is true only if, for each $(S, \ell) \in \mathcal{L}$,
$\ell \in \{d(S) - 2k \mid 0 \leq k \leq \binom{|S|}{2}\}$.

Nontrivial Cut Constraints

Small Cuts

Fixed Edges

For a solution G if $(\{u, v\}, d(u) + d(v) - 2) \in \mathcal{L}$ then $uv \in E(G)$.

Forbidden Edges

For a solution G if $(\{u, v\}, d(u) + d(v)) \in \mathcal{L}$ then $uv \notin E(G)$.

Simplification

Replace $(\{u, v\}, d(u) + d(v) - 2)$ by $(\{u, v\}, d(u) + d(v))$ while updating $\mathtt{d}$.

Possibility Graph

Let F be the set of forbidden edges.

Possibility Graph

Let F be the set of forbidden edges.

Then we call $\mathcal{G} = K_n - F$ the *possibility graph*.

Size 2 Cut Constraints

We can reduce GR-C to f-factor!

Size 2 Cut Constraints

We can reduce GR-C to f-factor!

Lemma (1)

An instance $(\mathfrak{d}, \mathcal{L})$ of GR-C can be solved in polynomial time whenever $w(\mathcal{L}) = 2$.

Size 3 Cut Constraints

We can reduce to the previous case!

Size 3 Cut Constraints

We can reduce to the previous case!

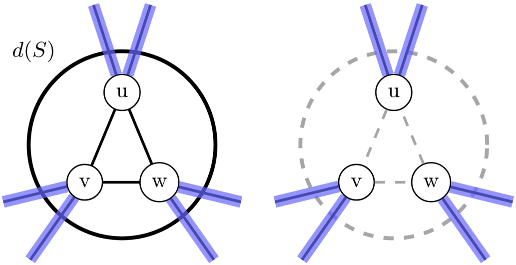
Theorem (1)

An instance $(\mathscr{d}, \mathcal{L})$ of GR-C can be solved in polynomial time whenever $w(\mathcal{L}) = 3$.

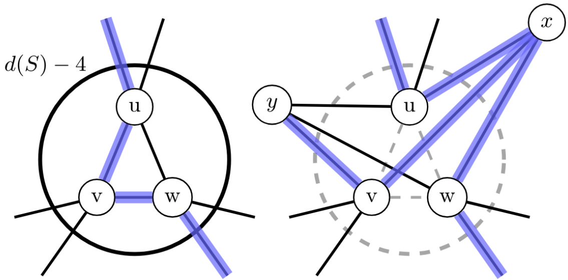
Proof

Consider a cut $(S, \ell) \in \mathcal{L}$ where $S = \{u, v, w\}$.

Case $\ell = d(S)$



Case $\ell = d(S) - 4$

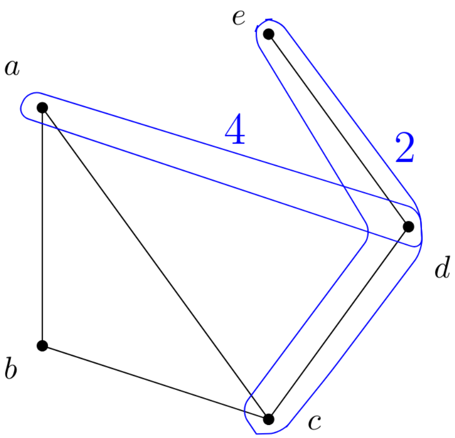


Other Cases

$$\ell = d(S) - 2 \text{ or } \ell = d(S) - 6$$

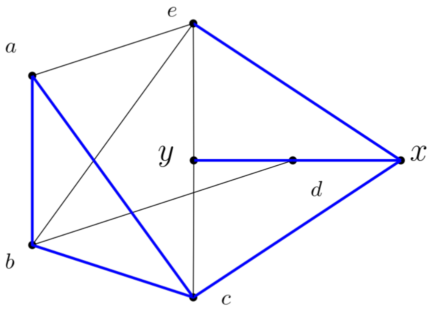
Running Example

$$\mathtt{d} = (3, 2, 2, 2, 1) \text{ and } \mathcal{L} = \{(\{a, d\}, 4), (\{c, d, e\}, 2)\}:$$



Running Example

Equivalent f-factor instance:



Nontrivial Cut Constraints

Large Cuts

Size 4 Cut Constraints

Can we keep doing a case-by-case analysis?

Size 4 Cut Constraints

Can we keep doing a case-by-case analysis?

No, we cannot, as the GR-C becomes hard!

Intuition

For $S \in \binom{V}{3}$, the edges within S determine how degrees change.

Intuition

For $S \in \binom{V}{3}$, the edges within S determine how degrees change.

In contrast, when $S \in \binom{V}{4}$, this claim does not hold anymore.

Hardness

Theorem (2)

The GR-C problem cannot be solved in polynomial time unless $P = NP$ even when $w(\mathcal{L}) = 4$ and all degrees in the degree sequence $\mathtt{d}$ are 1.

Proof

Reduction from k-True 1-in-3-SAT$_{(2,1)}$

1-IN-3-SAT$_{(2,1)}$

Input: A set of variables X and a formula ϕ in conjunctive normal form over X such that:

- each variable of X occurs twice as a positive literal and once as a negative literal;
- each clause of ϕ has two or three literals.

Question: Is there a truth assignment of X such that exactly one literal in every clause of ϕ is true?

1-**in**-3-**SAT**$_{(2,1)}$

Lemma (2)

1-in-3-SAT$_{(2,1)}$ is NP-complete.

k-True 1-in-3-SAT$_{(2,1)}$

k-TRUE 1-IN-3-SAT$_{(2,1)}$

Input: A tuple (X, ϕ, k), where (X, ϕ) is an instance of 1-in-3-SAT$_{(2,1)}$ and k is a nonnegative integer.

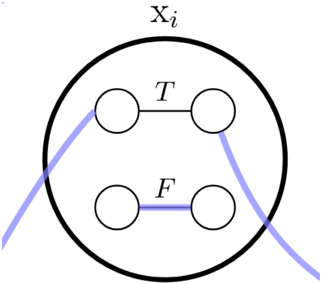
Question: Is there a feasible solution to (X, ϕ) in which exactly k variables are assigned to true?

k-True 1-in-3-SAT$_{(2,1)}$

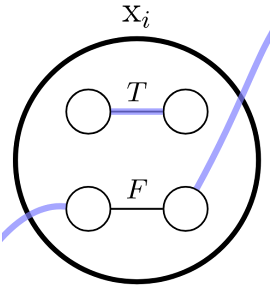
Lemma (3)

k-True 1-in-3-SAT$_{(2,1)}$ cannot be solved in polynomial time unless $P = NP$.

Variable Gadget

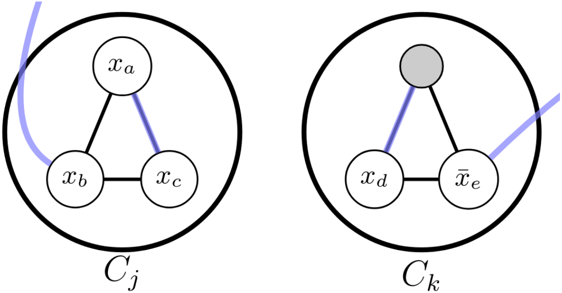


Variable Gadget



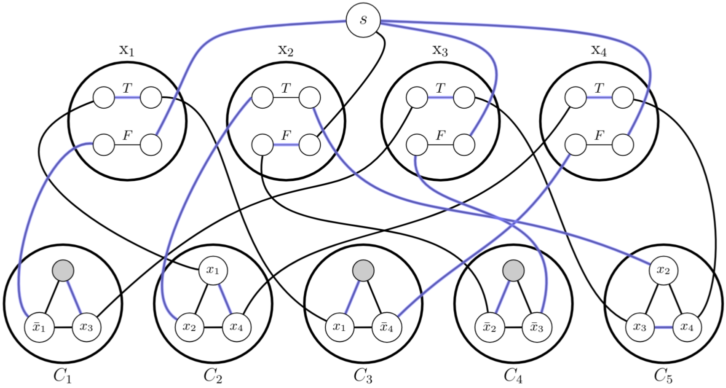
Clause Gadget

$C_j = (x_a + x_b + x_c)$ and $C_j = (x_d + \bar{x}_e)$



Complete Example

$$(\bar{x}_1 + x_3)(x_1 + x_2 + x_4)(x_1 + \bar{x}_4)(\bar{x}_2 + \bar{x}_3)(x_2 + x_3 + x_4) \text{ and } k = 1$$



Conclusion

Tree Possibility Graph

Proposition (1)

Given an instance $(\mathscr{d}, \mathcal{L})$ of GR-C with a tree possibility graph $\mathcal{G}$, we can decide if there is a solution in polynomial time.

Bipartite Possibility Graph

Theorem (3)

The GR-C problem is NP-complete when the possibility graph $\mathcal{G}$ is subcubic and bipartite, even when $w(\mathcal{L}) = 6$ and $\mathbb{d}$ is a sequence of ones.

Open Problems

$\mathcal{G}$ is planar or has bounded treewidth

Open Problems

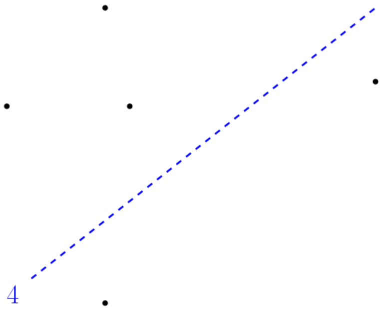
The size of $\mathcal{L}$ is small (like $|\mathcal{L}| = 1$)

Open Problems

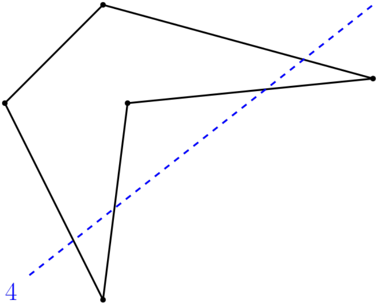
Complexity of 1-in-3 $\mathrm{SAT}_{(2,2)}$

Open Problems

Geometric version of GR-C



Polygon Realization with Cut Constraints



Thank you all for the attention...

The End

Bibliography

[Ans85] R.P Anstee.
An algorithmic proof of tutte's f-factor theorem.
Journal of Algorithms, 6(1):112–131, 1985.

[EG60] Paul Erdős and Tibor Gallai.
Gráfok előírt fokszámú pontokkal.
Matematikai Lapok, 11:264–274, 1960.