Decomposing Cubic Graphs into Connected Subgraphs of Size Three

Laurent Bulteau Guillaume Fertin **Anthony Labarre**
Romeo Rizzi Irena Rusu

International Computing and Combinatorics Conference (COCOON)

August 3rd, 2016

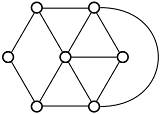
The graph decomposition problem

Given a set S of graphs, an S-**decomposition** of a graph G is a partition of $E(G)$ into subgraphs, all of which are isomorphic to a graph in S.

The graph decomposition problem

Given a set S of graphs, an S-**decomposition** of a graph G is a partition of $E(G)$ into subgraphs, all of which are isomorphic to a graph in S.

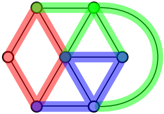
Example (S = connected graphs on four edges)



The graph decomposition problem

Given a set S of graphs, an S-**decomposition** of a graph G is a partition of $E(G)$ into subgraphs, all of which are isomorphic to a graph in S.

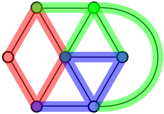
Example ($S =$ connected graphs on four edges)



The graph decomposition problem

Given a set S of graphs, an S-**decomposition** of a graph G is a partition of $E(G)$ into subgraphs, all of which are isomorphic to a graph in S.

Example (S = connected graphs on four edges)



S-DECOMPOSITION

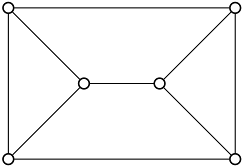
Input: a graph $G = (V, E)$, a set S of graphs.
Question: does G admit an S-decomposition?

S-DECOMPOSITION is NP-complete, even when S contains a single connected graph with at least three edges [Dor and Tarsi, 1997].

Graph decompositions for cubic graphs

We study the S-decomposition problem in the case where G is cubic and S is the set of all connected graphs on three edges.

Example

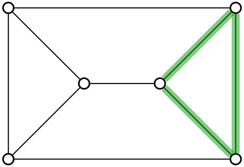


$\overline{C_6} =$

Graph decompositions for cubic graphs

We study the S-decomposition problem in the case where G is cubic and S is the set of all connected graphs on three edges.

Example

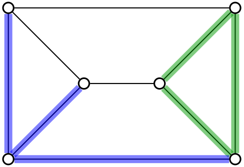


$\overline{C_6} = K_3 +$

Graph decompositions for cubic graphs

We study the S-decomposition problem in the case where G is cubic and S is the set of all connected graphs on three edges.

Example

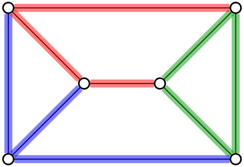


$\overline{C_6} = K_3 + K_{1,3} +$

Graph decompositions for cubic graphs

We study the S-decomposition problem in the case where G is cubic and S is the set of all connected graphs on three edges.

Example

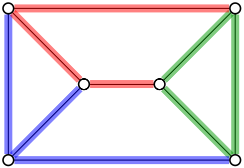


$$\overline{C_6} = K_3 + K_{1,3} + P_4$$

Graph decompositions for cubic graphs

We study the S-decomposition problem in the case where G is cubic and S is the set of all connected graphs on three edges.

Example

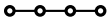


$$\overline{C_6} = K_3 + K_{1,3} + P_4$$

S'-DECOMPOSITION

Input: a cubic graph $G = (V, E)$, a non-empty set $S' \subseteq S$.
Question: does G admit a S'-decomposition?

Our contributions

Here is a summary of what is known about decomposing graphs using subsets of {  ,  ,  }:

Allowed subgraphs			Complexity according to graph class	
			cubic	arbitrary
✓				NP-complete [Dyer and Frieze, 1985]
	✓		$O(1)$ (impossible)	NP-complete [Holyer, 1981]
		✓	in P [Kotzig, 1957]	NP-complete [Dyer and Frieze, 1985]
✓	✓			NP-complete [Dyer and Frieze, 1985]
✓		✓		NP-complete [Dyer and Frieze, 1985]
	✓	✓		NP-complete [Dyer and Frieze, 1985]
✓	✓	✓		NP-complete [Dyer and Frieze, 1985]

Our contributions

Here is a summary of what is known about decomposing graphs using subsets of {  ,  ,  }:

Allowed subgraphs			Complexity according to graph class	
			cubic	arbitrary
✓			in P	NP-complete [Dyer and Frieze, 1985]
	✓		$O(1)$ (impossible)	NP-complete [Holyer, 1981]
		✓	in P [Kotzig, 1957]	NP-complete [Dyer and Frieze, 1985]
✓	✓		in P	NP-complete [Dyer and Frieze, 1985]
✓		✓	NP-complete	NP-complete [Dyer and Frieze, 1985]
	✓	✓	in P	NP-complete [Dyer and Frieze, 1985]
✓	✓	✓	NP-complete	NP-complete [Dyer and Frieze, 1985]

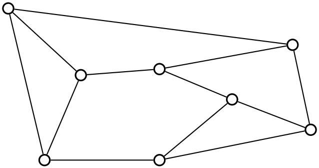
our contributions

Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

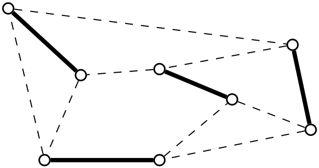


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

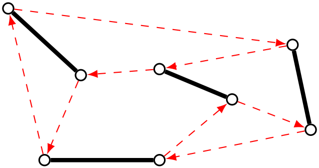


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

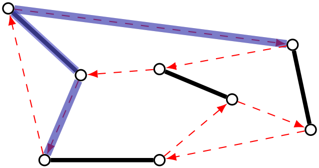


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

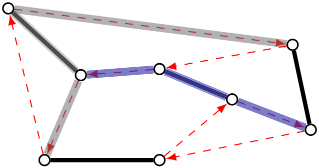


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

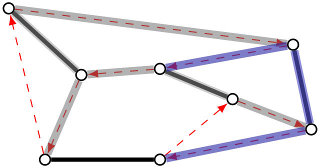


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

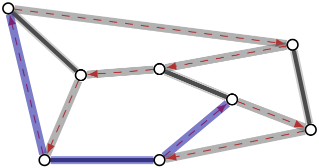


Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.



Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

We strengthen this result as follows:

Proposition

A cubic graph admits a $\{K_3, P_4\}$-decomposition if and only if it has a perfect matching.

Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

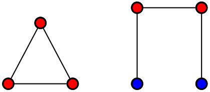
A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

We strengthen this result as follows:

Proposition

A cubic graph admits a $\{K_3, P_4\}$-decomposition if and only if it has a perfect matching.

Degree constraint:



A red vertex (degree 2) in some subgraph of the decomposition must be blue (degree 1) in another.

Decomposing cubic graphs without $K_{1,3}$'s

We need the following result:

Proposition ([Kotzig, 1957])

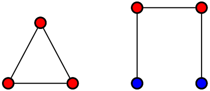
A cubic graph admits a P_4-decomposition if and only if it has a perfect matching.

We strengthen this result as follows:

Proposition

A cubic graph admits a $\{K_3, P_4\}$-decomposition if and only if it has a perfect matching.

Degree constraint:



A red vertex (degree 2) in some subgraph of the decomposition must be blue (degree 1) in another.

Use counting argument $\Rightarrow$ no K_3 can be used.

Decomposing cubic graphs without P_4's

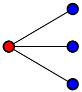
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

Proof.

$\boxed{\Rightarrow}$



A center (red) belongs to only one subgraph
$\Rightarrow$ Bipartition: centers – leaves
(each edge connects a center and a leaf)

Decomposing cubic graphs without P_4's

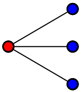
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

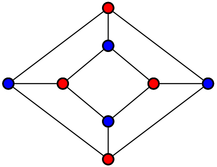
Proof.

$\Rightarrow$



A center (red) belongs to only one subgraph
$\Rightarrow$ Bipartition: centers – leaves
(each edge connects a center and a leaf)

$\Leftarrow$



Use one part for centers, the other for leaves

□

Decomposing cubic graphs without P_4's

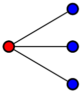
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

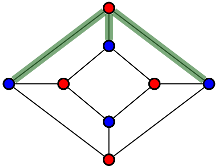
Proof.

$\Rightarrow$



A center (red) belongs to only one subgraph
$\Rightarrow$ Bipartition: centers – leaves
(each edge connects a center and a leaf)

$\Leftarrow$



Use one part for centers, the other for leaves

□

Decomposing cubic graphs without P_4's

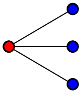
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

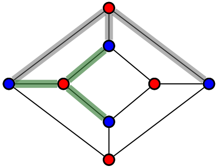
Proof.

$\Rightarrow$



A center (red) belongs to only one subgraph
$\Rightarrow$ Bipartition: centers – leaves
(each edge connects a center and a leaf)

$\Leftarrow$



Use one part for centers, the other for leaves

□

Decomposing cubic graphs without P_4's

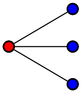
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

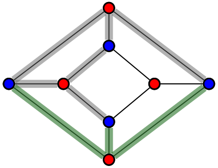
Proof.

$\Rightarrow$



A center (red) belongs to only one subgraph
$\Rightarrow$ Bipartition: centers – leaves
(each edge connects a center and a leaf)

$\Leftarrow$



Use one part for centers, the other for leaves

□

Decomposing cubic graphs without P_4's

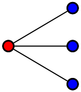
Let us start with $K_{1,3}$-decompositions:

Proposition

A cubic graph admits a $K_{1,3}$-decomposition if and only if it is bipartite.

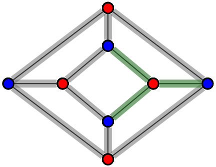
Proof.

⇒



A center (red) belongs to only one subgraph
⇒ Bipartition: centers – leaves
(each edge connects a center and a leaf)

⇐

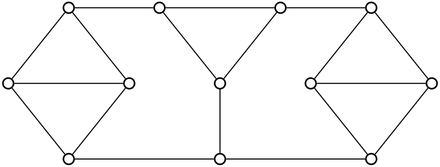


Use one part for centers, the other for leaves

□

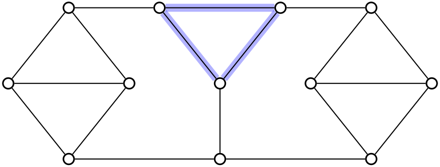
Decomposing cubic graphs without P_4's

What if we also allow K_3's?



Decomposing cubic graphs without P_4's

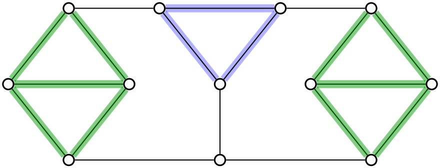
What if we also allow K_3's?



We distinguish between *isolated* and *nonisolated* triangles:

Decomposing cubic graphs without P_4's

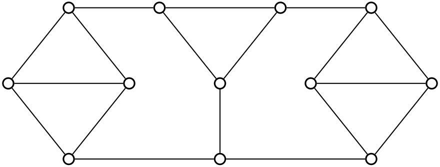
What if we also allow K_3's?



We distinguish between *isolated* and *nonisolated* triangles:

Decomposing cubic graphs without P_4's

What if we also allow K_3's?



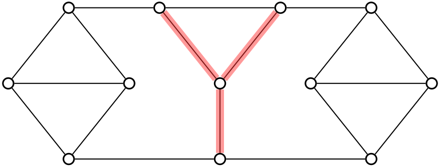
We distinguish between *isolated* and *nonisolated* triangles:

Lemma

If a cubic graph G admits a $\{K_{1,3}, K_3\}$-decomposition D, then every isolated K_3 in G belongs to D.

Decomposing cubic graphs without P_4's

What if we also allow K_3's?



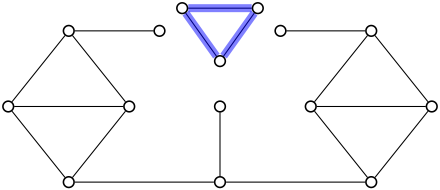
We distinguish between *isolated* and *nonisolated* triangles:

Lemma

If a cubic graph G admits a $\{K_{1,3}, K_3\}$-decomposition D, then every isolated K_3 in G belongs to D.

Decomposing cubic graphs without P_4's

What if we also allow K_3's?



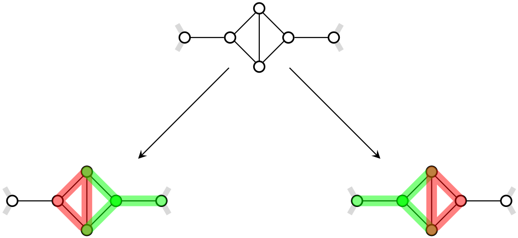
We distinguish between *isolated* and *nonisolated* triangles:

Lemma

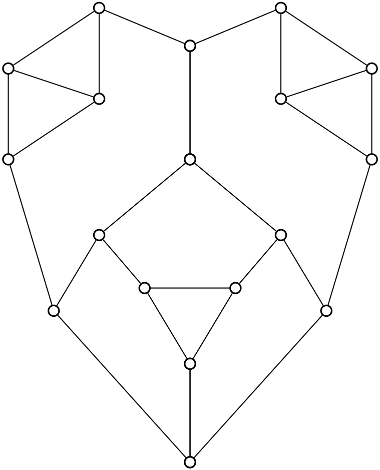
If a cubic graph G admits a $\{K_{1,3}, K_3\}$-decomposition D, then every isolated K_3 in G belongs to D.

Decomposing cubic graphs without P_4's

If G also contains nonisolated K_3's, then we only have two choices to try:

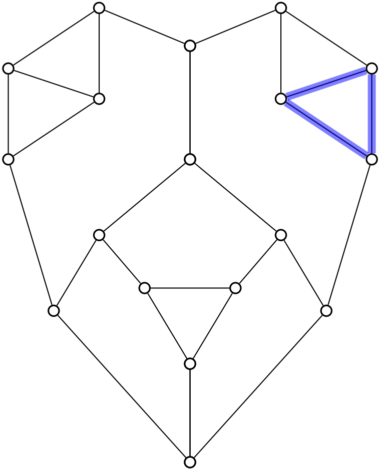


Summary of algorithm



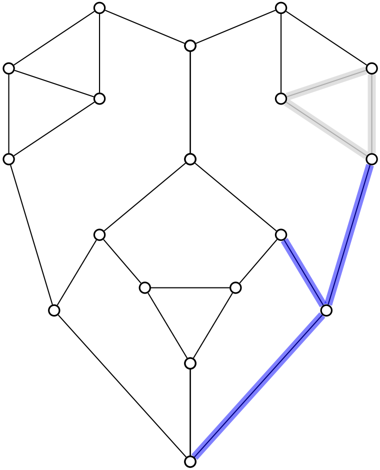
- Select a diamond, pick one K_3

Summary of algorithm



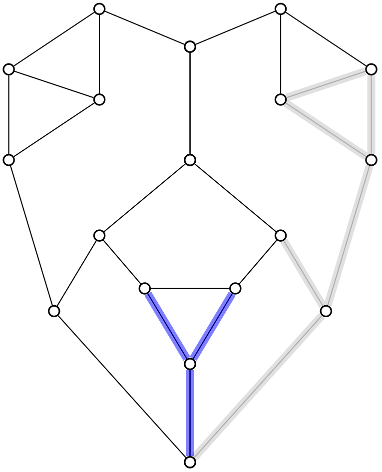
- Select a diamond, pick one K_3

Summary of algorithm



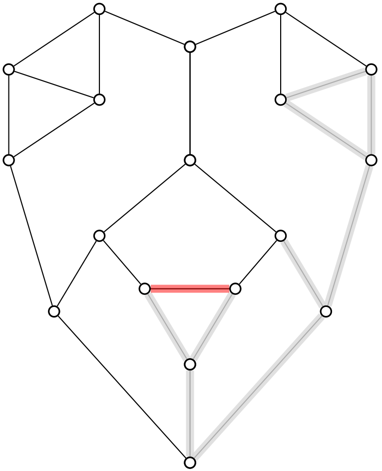
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$

Summary of algorithm



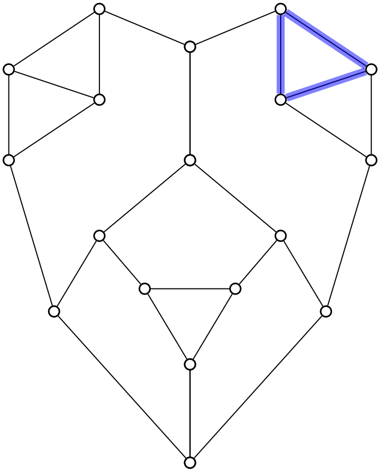
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3

Summary of algorithm



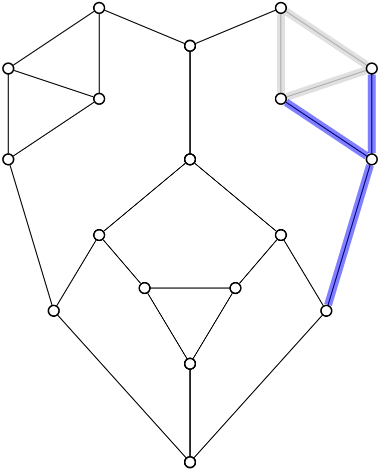
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3

Summary of algorithm



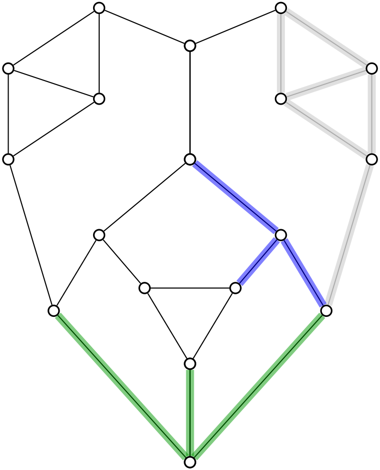
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3

Summary of algorithm



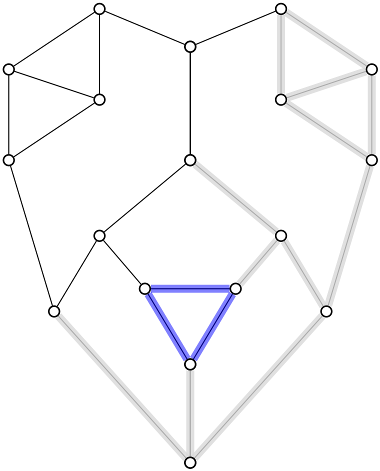
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3

Summary of algorithm



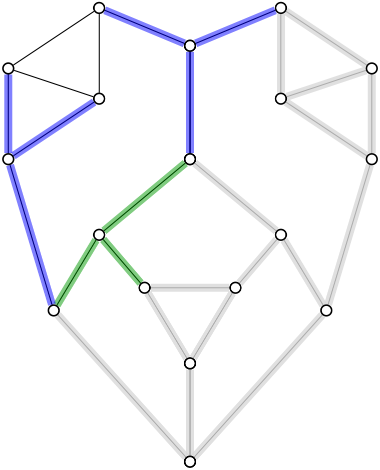
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3

Summary of algorithm



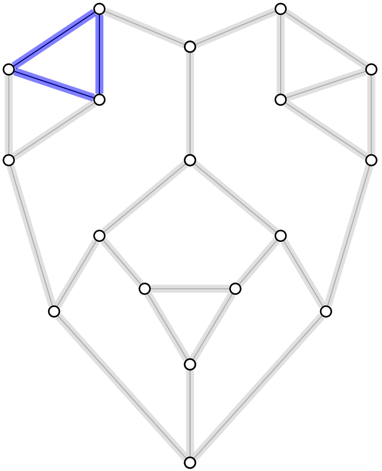
- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3

Summary of algorithm



- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3

Summary of algorithm



- Select a diamond, pick one K_3
- Follow degree-1,2 nodes:
  - Degree 1: pick as leaf of $K_{1,3}$
  - Degree 2 outside any K_3: pick as leaf of $K_{1,3}$
  - Degree 2 inside a K_3: pick the K_3
- If it fails, try the other starting K_3
- Only one branching $\Rightarrow$ polynomial time algorithm

Hardness results

We now show that $\{K_{1,3},\ P_4\}$-DECOMPOSITION is NP-complete, using three reductions:

CUBIC PLANAR MONOTONE 1-IN-3 SATISFIABILITY

$\leq_P$ DEGREE-2,3 $\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION WITH MARKED EDGES

$\leq_P$ $\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION WITH MARKED EDGES

$\leq_P$ $\{K_{1,3},\ P_4\}$-DECOMPOSITION

Hardness results

We now show that $\{K_{1,3},\ P_4\}$-DECOMPOSITION is NP-complete, using three reductions:

 CUBIC PLANAR MONOTONE 1-IN-3 SATISFIABILITY

$\leq_P$ DEGREE-2,3 $\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION WITH MARKED EDGES

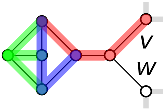
$\leq_P$ $\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION WITH MARKED EDGES

$\leq_P$ $\{K_{1,3},\ P_4\}$-DECOMPOSITION

A similar approach can be used to show the NP-completeness of $\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION.

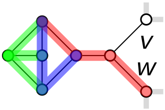
Hardness results 1/3: marked edges

The co-fish gadget



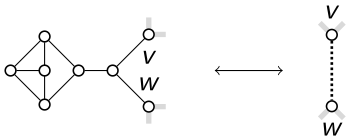
Hardness results 1/3: marked edges

The co-fish gadget



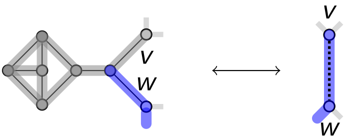
Hardness results 1/3: marked edges

The co-fish gadget



This gadget is equivalent to an edge . . . .

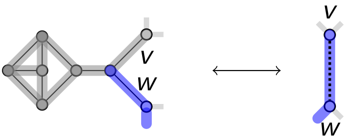
The co-fish gadget



This gadget is equivalent to an edge that cannot be in the middle of a $P_4 \Rightarrow$ **marked** *edges*.

Hardness results 1/3: marked edges

The co-fish gadget



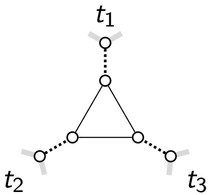
This gadget is equivalent to an edge that cannot be in the middle of a P_4 ⇒ **marked** *edges*.

$\{K_{1,3},\ K_3,\ P_4\}$-DECOMPOSITION WITH MARKED EDGES

Input: a cubic graph $G = (V, E)$ and a subset $M \subseteq E$ of edges.
Question: does G admit a $\{K_{1,3},\ K_3,\ P_4\}$-decomposition D such that no edge in M is the middle edge of a P_4 in D and such that every K_3 in D has either one or two edges in M?

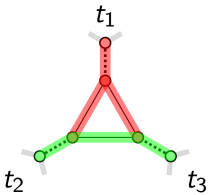
Hardness results 2/3: leafless subcubic graphs

The net gadget



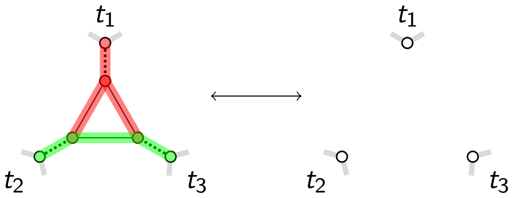
Hardness results 2/3: leafless subcubic graphs

The net gadget



Hardness results 2/3: leafless subcubic graphs

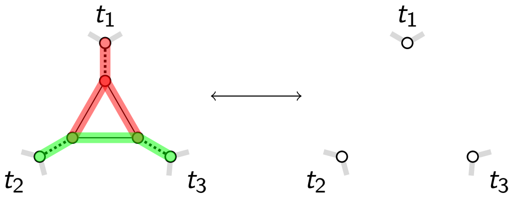
The net gadget



The net gadget is equivalent to 3 degree-2 nodes

Hardness results 2/3: leafless subcubic graphs

The net gadget



The net gadget is equivalent to 3 degree-2 nodes

We can restrict our attention to DEGREE-2,3 $\{K_{1,3}, K_3, P_4\}$-DECOMPOSITION WITH MARKED EDGES, a variant where the input graph contains vertices with degree 2 or 3.

Hardness results 3/3: satisfiability

CUBIC (PLANAR) MONOTONE 1-IN-3 SATISFIABILITY

Input: a Boolean formula $\phi = C_1 \wedge C_2 \wedge \cdots$ without negations; $|C_i| = 3$ for each i and each literal appears in exactly three clauses;

Question: is there an assignment of truth values $f : \Sigma \to \{\text{TRUE}, \text{FALSE}\}$ such that each clause of ϕ contains exactly one TRUE literal?

CUBIC PLANAR MONOTONE 1-IN-3 SATISFIABILITY

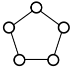
$\leq_P$ DEGREE-2,3 $\{K_{1,3}, K_3, P_4\}$-DECOMPOSITION WITH MARKED EDGES

$\leq_P$ $\{K_{1,3}, K_3, P_4\}$-DECOMPOSITION WITH MARKED EDGES

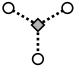
$\leq_P$ $\{K_{1,3}, P_4\}$-DECOMPOSITION

The reduction from CUBIC MONO-1-IN-3-SAT

Clause



Variable

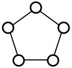


The reduction

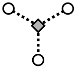
- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.

The reduction from CUBIC MONO-1-IN-3-SAT

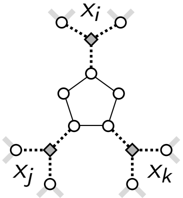
Clause



Variable



$C = x_i \vee x_j \vee x_k$

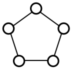


The reduction

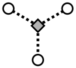
- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.

The reduction from CUBIC MONO-1-IN-3-SAT

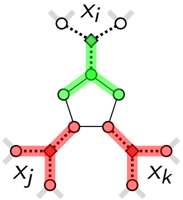
Clause



Variable



$C = x_i \vee x_j \vee x_k$

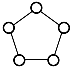


The reduction

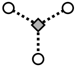
- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.
- From assignments to decompositions: variables set to FALSE yield red $K_{1,3}$'s, those set to TRUE yield green $K_{1,3}$'s.

The reduction from CUBIC MONO-1-IN-3-SAT

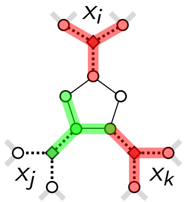
Clause



Variable



$C = x_i \vee x_j \vee x_k$

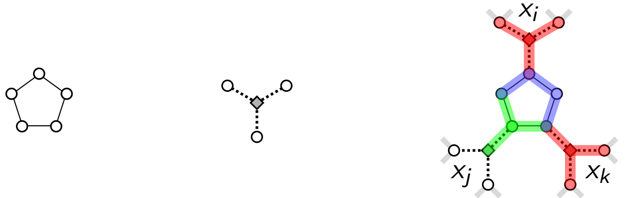


The reduction

- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.
- From assignments to decompositions: variables set to FALSE yield red $K_{1,3}$'s, those set to TRUE yield green $K_{1,3}$'s.

The reduction from CUBIC MONO-1-IN-3-SAT

Clause   Variable $C = x_i \vee x_j \vee x_k$

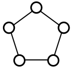


The reduction

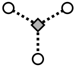
- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.
- From assignments to decompositions: variables set to FALSE yield red $K_{1,3}$'s, those set to TRUE yield green $K_{1,3}$'s.

The reduction from CUBIC MONO-1-IN-3-SAT

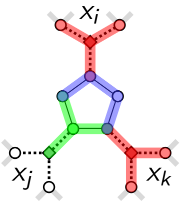
Clause



Variable



$C = x_i \vee x_j \vee x_k$



The reduction

- Map clauses onto C_5's and variables onto marked $K_{1,3}$'s.
- From assignments to decompositions: variables set to FALSE yield red $K_{1,3}$'s, those set to TRUE yield green $K_{1,3}$'s.
- From decompositions to assignments: show that a decomposable graph **must** conform to the above configuration $\Rightarrow$ truth assignment

Conclusions

- Future work:
  - hardness for *planar* cubic graphs?
  - complexity of those problems for subcubic graphs?
  - generalise positive results to k-regular graphs for $k > 3$;

Thank you!

References

Dor, D. and Tarsi, M. (1997).
Graph decomposition is NP-complete: A complete proof of Holyer's conjecture.
SIAM J. Comput., 26:1166–1187.

Dyer, M. E. and Frieze, A. M. (1985).
On the complexity of partitioning graphs into connected subgraphs.
Discrete Appl. Math., 10(2):139–153.

Holyer, I. (1981).
The NP-completeness of some edge-partition problems.
SIAM J. Comput., 10(4):713–717.

Kotzig, A. (1957).
Z teorie konečných pravidelných grafov tretieho a štvrtého stupňa.
Časopis pro pěstování matematiky, pages 76–92.