

# Perfect Graph Recognition and Coloring

Dni Magistranta

Adrian Siwiec

December 05, 2019

## **"The strong perfect graph theorem"**

*Maria Chudnovsky, Neil Robertson, Paul Seymour, Robin Thomas*

<https://arxiv.org/abs/math/0212070>

## **"A polynomial algorithm for recognizing perfect graphs"**

*Gérard Cornuéjols, Xinming Liu, Kristina Vušković*

<https://ieeexplore.ieee.org/document/1238177>

## **"Recognizing Berge Graphs"**

*Maria Chudnovsky, Gérard Cornuéjols, Xinming Liu, Paul Seymour, Kristina Vušković*

<https://link.springer.com/article/10.1007/s00493-005-0012-8>

## **"Colouring perfect graphs with bounded clique number"**

*Maria Chudnovsky, Aurélie Lagoutte, Paul Seymour, Sophie Spirkl*

<https://arxiv.org/abs/1707.03747>

# Perfect Graphs

## Perfect Graphs

A graph is *perfect* if the chromatic number of every induced subgraph equals the size of its largest clique.

# Perfect Graphs

## Perfect Graphs

A graph is *perfect* if the chromatic number of every induced subgraph equals the size of its largest clique.

## Perfect Graphs are interesting

In all perfect graphs, the *graph coloring problem*, *maximum clique problem*, and *maximum independent set problem* can all be solved in polynomial time. (Grötschel, Lovász, Schrijver 1988)

# Perfect Graphs

## Perfect Graphs

A graph is *perfect* if the chromatic number of every induced subgraph equals the size of its largest clique.

## Perfect Graphs are interesting

In all perfect graphs, the *graph coloring problem*, *maximum clique problem*, and *maximum independent set problem* can all be solved in polynomial time. (Grötschel, Lovász, Schrijver 1988)

## Families of graphs that are perfect

- Bipartite graphs
- Line graphs of bipartite graphs
- Chordal graphs
- Comparability graphs
- ...

# Strong Perfect Graph Theorem

## Perfect Graphs

A graph is *perfect* if the chromatic number of every induced subgraph equals the size of its largest clique.

## Berge Graphs

A graph is *Berge* if no induced subgraph of  $G$  is an odd cycle of length at least five or the complement of one.

# Strong Perfect Graph Theorem

## Perfect Graphs

A graph is *perfect* if the chromatic number of every induced subgraph equals the size of its largest clique.

## Berge Graphs

A graph is *Berge* if no induced subgraph of  $G$  is an odd cycle of length at least five or the complement of one.

## Strong Perfect Graph Theorem

A graph  $G$  is Perfect if and only if it is Berge.

# Recognizing Berge Graphs - an overview of an overview

## An odd hole

An odd hole in  $G$  is an induced subgraph of  $G$  that is a cycle of odd length at least five.

A graph  $G$  is Berge if  $G$  and its complement both have no odd hole.

## Algorithm

The idea of our algorithm is to decompose the input graph  $G$  into a polynomial number of simpler graphs  $G_1, \dots, G_m$  so that:

- $G$  is odd-hole-free if and only if every  $G_i$  is odd-hole-free.
- It is easy to check if  $G_i$  is odd-hole-free.



# Decompositions

## 2-join

A graph  $G$  has a 2-join  $V_1|V_2$  with special sets  $(A_1, A_2, B_1, B_2)$  if  $A_i, B_i \subset V_i$ , every vertex of  $A_1$  is adjacent to every vertex of  $A_2$ , every vertex of  $B_1$  is adjacent to every vertex of  $B_2$  and there are no other adjacencies between  $V_1$  and  $V_2$ .

## Double Star

A set  $S$  of vertices is a double star if  $S$  contains two adjacent vertices  $u$  and  $v$  such that  $S \subseteq N(u) \cup N(v)$ .

Double-star cutsets pose a problem.

## Skew Partitions

A *skew partition* in  $G$  is a partition  $(A, B)$  of  $V(G)$ , such that  $G[A]$  is not connected and  $\overline{G}[B]$  is not connected.

## Decomposition Theorem

Every Berge graph either admits a balanced skew partition, or admits one of two other decompositions, or it belongs to one of five well-understood classes.

## Goals

- Implement Berge Graph recognition algorithm.
  - Compare with existing *Java* implementation.
  - Are there any other implementations to compare to?
- Implement Perfect Graph colouring algorithm.
  - Compare with existing linear programming solutions.