Approximation Algorithms

- Approximation Algorithms for Optimization Problems: Types
- Absolute Approximation Algorithms
- Inapproximability by Absolute Approximate Algorithms
- Relative Approximation Algorithm
- InApproximability by Relative Approximate Algorithms
- Polynomial Time Approximation Schemes
- Fully Polynomial Time Approximation Schemes

Imdad Ullah Khan

Quality of Approximation: Types

Absolute Approximation Algorithms

Given an optimization problem P with value function f on solution space

An algorithm A is called **absolute approximation** algorithm if there is a constant k such that for any instance I

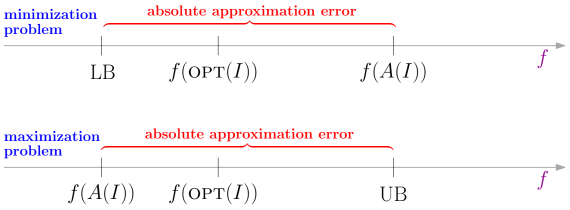
$$\left| f\big(A(I)\big) - f\big(\mathrm{OPT}(I)\big) \right| \leq k$$

- For a minimization problem this means $f\left(A(I)\right) \leq f\left(\mathrm{OPT}(I)\right) + k$
- For a maximization problem this means $f\left(A(I)\right) \geq f\left(\mathrm{OPT}(I)\right) - k$

How to Design an Approximation Algorithm?

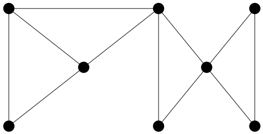
Any approximation algorithm involves 2 main steps:

1. Design a good algorithm A
   - For approximation guarantee on $A(I)$ we need the value of the optimal solution $f(\text{OPT}(I))$
   - How to find it? almost equally difficult (version of the problem)
2. Find a good lower or upper bound on $f(\text{OPT}(I))$
   - Compare $f(A(I))$ with the bound on $f(\text{OPT}(I))$

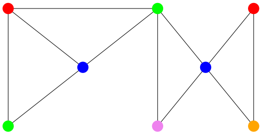


Graph Coloring

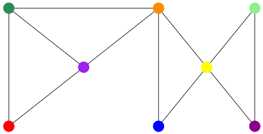
A graph (vertex) coloring is to assign a color to each vertex such that no two adjacent vertices get the same color



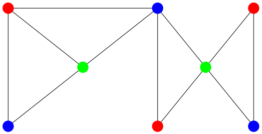
A graph G on 8 vertices



A coloring with 6 colors



A coloring with 8 colors



A coloring with (optimal) 3 colors

[Coloring(G) problem:] Color G with minimum number of colors, $\chi(G)$

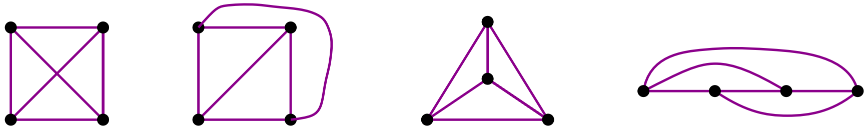
Graph Coloring (Optimization) Problem

The graph coloring problem

- $\mathcal{I}$: Graphs
- $S(G)$: An assignment of colors to vertices of input graph, such that no two adjacent vertices have the same color (feasibility)
- $f : S(G) \to \mathbb{Z}^+$
  - For $s \in S(G)$, $f(s)$ is number of colors used in the coloring s
- $\chi(G)$: the minimum number of colors needed to color G
  - $\chi(G) = f(\textsc{opt}(G))$

Planar Graphs

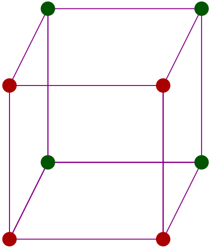
A graph is planar if it can be drawn in the plane without any edge crossing

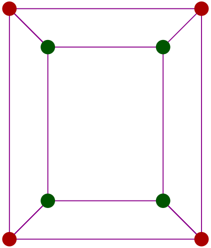


Planar Graphs

A graph is planar if it can be drawn in the plane without any edge crossing

Just because in a drawing of G edges are crossing doesn't mean G is not planar

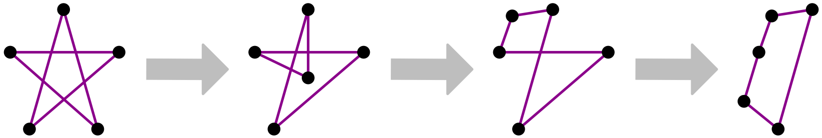




Planar Graphs

A graph is planar if it can be drawn in the plane without any edge crossing

To prove planarity, move the vertices around, redraw the edges without crossing (sometimes in a very indirect faction)



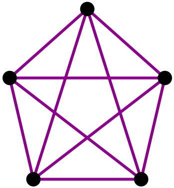
Planar Graphs

Planar Graphs

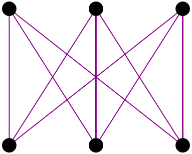
A graph is planar if it can be drawn in the plane without any edge crossing

To prove planarity, move the vertices around, redraw the edges without crossing (sometimes in a very indirect faction)

Even harder to prove non-planarity



K_5

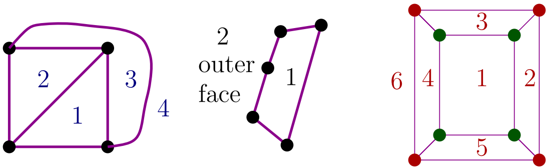


$K_{3,3}$

Planar Graphs: A characterization

A plane drawing of a planar graphs divides plane into regions or faces, one of them the outer face

A region (or face) is a part of the plane disconnected from other parts by the edges

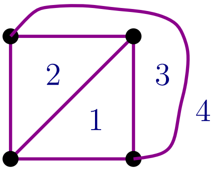


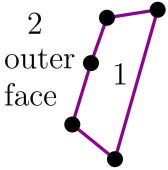
Planar Graphs: A characterization

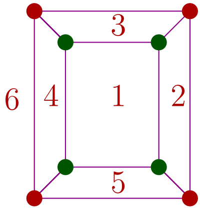
Euler's Formula: The number of faces of a connected planar graph is invariant of its drawing and is given by

$$f = e - v + 2 \qquad f = |faces|, \; e = |E|, \; v = |V|$$

Verify it for the following planar graphs







Planar Graphs: A characterization

From Euler's formula and the *Face-Edge Handshaking Lemma* we get

If G is a connected planar graph with $v \geq 3$, then

$$e \leq 3v - 6$$

An immediate corollary from this using the Handshaking Lemma is

Every planar graph has a vertex with degree at most 5

Coloring Planar Graphs

The PLANAR-GRAPH-COLORING(G) problems: Given a planar graph G, color it with minimum colors

6-Coloring Planar Graphs

Every planar graph has a vertex with degree at most 5

- Using the lemma we give a recursive 6-coloring algorithm
- Can apply the algorithm to components of disconnected graphs

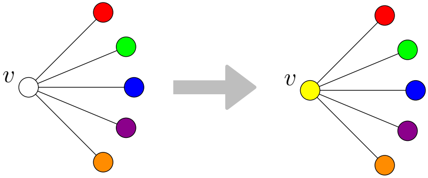
Algorithm 6-COLOR($G, C = \{c_1, \dots, c_6\}$)

Let $v \in V$ such that $deg(v) \leq 5$

6-COLOR($G - v, C = \{c_1, \dots, c_6\}$)

Let $C' \subset C$ be the set of colors used for $N(v)$ ▷ $|C'| \leq 5$

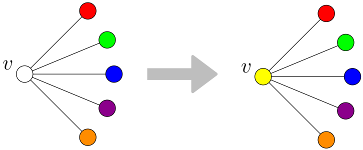
Color v with a color in $C \setminus C'$



6-Coloring Planar Graphs

Every planar graph has a vertex with degree at most 5

- Using the lemma we give a recursive 6-coloring algorithm
- Can apply the algorithm to components of disconnected graphs
- Can be implemented with no recursion or modification to adjacency list
- Order vertices so no vertex has more than 5 neighbors preceding it
- Greedily color vertices from left to right using the scheme in figure
- Clearly polynomial time



3-absolute approximation for Planar Graph Coloring

Theorem: The decision problem of planar graph 3-Coloring is NP-COMPLETE

Use the fact that: If G is bipartite, then it is 2-colorable

Algorithm APPROX-PLANAR-COLOR(G)

if G is bipartite **then** ▷ Easy to check with a BFS

  Color G with the obvious 2-coloring

else

  6-COLOR($G, C = \{c_1, \ldots, c_6\}$)

APPROX-PLANAR-COLOR is a 3-absolute approximate algorithm

- Non-bipartite graphs require ≥ 3 colors ($f(\text{OPT}(G)) \geq 3$) ▷ **(LB)**
- We use at most 6 colors ($f(\text{APPROX-PLANAR-COLOR}(G)) \leq 6$)
- The statement follows

2 and 1-absolute approximation for Planar Graph Coloring

A slightly complicated Kempe algorithm colors planar graphs with 5 colors

▷ A 2-absolute approximate algorithm

A much more complicated proof that planar graphs can be colored with 4 colors, yields a 1-absolute approximate algorithm

- The proof is due to Appel and Haken (1976), is very complicated and involves a huge number of cases tested with a computer program
- After significant simplifications it still is very involved

Figure: UIUC stamp in honor of the 4-Color theorem