Homework 1

Wenting LI liw14@rpi.edu

September 9, 2017

1.

a.

Given a $(\Delta + 1)$-colorable graph, where Δ is the maximum degree of any vertex in $G = (V, E)$,

1. Find the vertex v in G with the maximum degree Δ and its neighbor $N(v) = \{w|(v, w) \in E, i = 1, \ldots, \Delta\}$;
2. First color v with one color, then color its neighbors $w_i \in N(v)$ one by one. If w_i is also connected to $w_j, i \neq j$, then color them with different colors. In this way, we need at most $\Delta + 1$ colors to color v and its neighbor $N(v)$;
3. Then find another v' not colored yet and its neighbor $N(v') = \{w|(v', w) \in E, i \leq \Delta\}$. Color them in the same way as step (2) using the $\Delta + 1$ colors;
4. Continue to color all other vertexes with less than $\Delta + 1$ colors until the graph G is completely colored.

As we can color G in poly-time with at most $(\Delta + 1)$ colors, then G is $(\Delta + 1)$ colorable.

b.

Bipartite graph is an undirected graph with nodes partitioned into group X and Y. For each line, its one ending node is in group X and the other ending node is in Y.

Then we can color all the nodes in group X with one color and the nodes in group Y with the other color. Therefore, bipartite graphs are 2-colorable.

c.

Given undirected 3-colorable graph $G = (V, E)$, $|V| = n$. For each $v \in V$ vertext, let $d(v)$ be the degree of v.

If $d(v) \geq \sqrt{(n)}$, then remove v and its neighbors $N(v) = \{w_i|(v, w_i) \in E, i \geq \sqrt{(n)}\}$, and color v with the first color and then color its neighbor w_1 with the second color, and then color all other neighbors one by one. If two of its neighbors w_i, w_j are connected, then color them with the second and the third color, otherwise color the neighbors with the second color. We can color the subgraph with a poly-time $O(n)$.

As graph G is a 3 colorable graph and thus its subgraph is also a 3-colorable graph, thus we need at most 3 colors to color each subgraph.

Each time we remove at most $\sqrt{n}+1$ vetex from G, which has n vertex in total, thus we need to remove at most $\sqrt{(n)}$. As we need color each subgraph with at most 3 colors, then in order to color all these vertex and its neighbors, we need at most $O(\sqrt{n})$ colors.

Until all the vertex in G have degree less than $\sqrt{(n)}$, we can apply the algorithm in (a) to color them with $O(\sqrt{(n)})$ colors with a poly-time.

In summary, we need $O(\sqrt{n})$ colors.

d.

Given undirected 3-colorable graph $G = (V, E)$, $|V| = n$. For each $v \in V$ vertext, let $d(v)$ be the degree of v.
If $d(v) \geq n^{\frac{2}{3}}$, then remove v and its neighbors $N(v) = \{w_i|(v, w_i) \in E, i \geq \sqrt{(n)}\}$, and color v with the first color and then color its neighbor w_1 with the second color, and then color all other neighbors one by one. If two of its neighbors w_i, w_j are connected, then color them with the second and the third color, otherwise color the neighbors with the second color. We can color the subgraph with a poly-time $O(n)$.

As graph G is a 4 colorable graph and thus its subgraph is also a 3-colorable graph, thus we need at most 4 colors to color each subgraph.

Each time we remove at most $(n^{\frac{2}{3}} + 1)$ vetex from G, which has n vertex in total, thus we need to remove at most $n^{\frac{1}{3}}$. As we need color each subgraph with at most 4 colors, then in order to color all these vertex and its neighbors, we need at most $O(n^{\frac{1}{3}})$ colors.

Until all the vertex in G have degree less than $O(n^{\frac{1}{3}}) + O(n^{\frac{2}{3}})$, we can apply the algorithm in (a) to color them with $O(n^{\frac{2}{3}})$ colors with a poly-time.

In summary, we need $O(n^{\frac{2}{3}})$ colors.

e.

Claim: Suppose VERTEX COLORING is approximable to a factor better than $\frac{4}{3}k$ for a k-colorable graph when $P \neq NP$, then determining if a graph is 3-colorable is solvable in poly-time.

Proof: $\Rightarrow$

If VERTEX COLORING is approximable to a factor better than $\frac{4}{3}k$ when $P \neq NP$, then we can find a poly-time algorithm that using less $\frac{4}{3}k$ colors for a k-colorable graph.

When $k = 3$ and $\frac{4}{3}k = 4$, as $3 < \frac{4}{3}k$, then we can find a poly-time algorithm to determine if a graph is 3-colorable.

This is contradict to the fact that determining if a graph is 3-colorable is NP-Complete. Therefore, VERTEX COLORING is not approximable to a factor better than $\frac{4}{3}$ unless $P = NP$.

1.5

(a)

As an extreme point is a feasible solution of the linear program, we will show that if any extreme point is not in the set of $\{0, \frac{1}{2}, 1\}$, then it is not a feasible solution of the linear program by contradiction.

Proof: $\Rightarrow$

Suppose there is one extreme point $x_{i'}$ of the linear program that has $x_{i'} \notin \{0, \frac{1}{2}, 1\}$.

$\Rightarrow$ the optimal solution f^* of the linear program is

$$f^* = x_{i'}w_{i'} + \Sigma_{i \in V, i \neq i'} w_i x_i \tag{1}$$

where x_i are all the extreme points.

As $\forall (i,j) \in E, x_i + x_j \geq 1$ and there is at least one point x'_j that $(i', j') \in E$, we have $x_{i'} \geq 1 - x_{j'}$.
$\Rightarrow$ (1) becomes:

$$
\begin{aligned}
f^* &= x_{i'}w_{i'} + x_{j'}w_{j'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i && (2)\\
&\geq (1 - x_{j'})w_{i'} + x_{j'}w_{j'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i && (3)\\
&= w_{i'} + x_{j'}(w_{j'} - w_{i'}) + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i && (4)
\end{aligned}
$$

However, we can find other solutions when $x_{i'} \in \{0, \frac{1}{2}, 1\}$,

1. If $x_{i'} = 0$ and $x_{j'} = 1$, then the solution $f_1 = w_{j'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i$
2. If $x_{i'} = \frac{1}{2}$ and $x_{j'} = \frac{1}{2}$, then the solution $f_2 = \frac{1}{2}(w_{i'} + w_{j'}) + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i$
3. If $x_{i'} = 1$ and $x_{j'} = 0$, then the solution $f_3 = w_{i'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i$

Then we will show the contradiction that f^* is not the minimum.

(1) . If $w_{i'} > w_{j'}$, then from (2), we have

$$
\begin{aligned}
f^* &\geq x_{i'}w_{j'} + x_{j'}w_{j'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i && (5)\\
&\geq w_{j'} + \Sigma_{i \in V, i \neq i', i \neq j'} w_i x_i && (6)\\
&= f_1 && (7)
\end{aligned}
$$

(2) . If $w_{i'} = w_{j'}$, then from (4), we have $(w_{j'} - w_{i'}) = 0$ and $f^* \geq f_3$;

(3) . If $w_{i'} < w_{j'}$, then from (4), we have $(w_{j'} - w_{i'}) > 0$ and $f^* > f_3$.

Therefore, f^* is not the minimum and this is contradict to our assumption. Hence, we have demonstrated that any the extreme point of the linear program has the property that $x_i \in \{0, \frac{1}{2}, 1\}$ for all $i \in V$.

(b)

I first give an algorithm for the vertex cover problem when the input graph is planar, then I will show that it is a $\dfrac{3}{2}$-approximation algorithm.

My algorithm is as following:

1. Input the planar graph and run the 4-color for it (assign each vertex one of four colors such that for any edge $(i, j) \in E$, vertices i and j have been assigned different colors);
2. Find the color $*$ that is assigned to the maximum number of points;
3. Take all the points that are assigned by the other three colors rather than the color $*$. Then these points form the vertex cover of the graph.

Then I will demonstrate that this algorithm is $\dfrac{3}{2}$-approximation.
Proof: $\Rightarrow$

1. As running the 4-color algorithm and finding the color with the maximum number can be finished in poly-time, this algorithm is a poly-time method;

2. As each line is assigned with two distinct colors and at least one of them is not the color $*$, at least one of the points of each line is selected by this algorithm. Thus our solution is a vertex cover;

3. We employ the optimal solution of linear program LP-OPT as the lower bound of the vertex cover problem.

   When there are n points in the graph, then the LP-OPT$=\frac{n}{2}$ and the optimal solution of vertex cover LP-OPT $\leq$ OPT.

   As all the n points of the graph is assigned with one of the four colors, the maximal number of points $N(*)$ assigned by the color $*$ is larger than $\frac{n}{4}$, otherwise the total number of all these four colors is less than n. Thus the number of points selected by my algorithm S satisfies

$$
\begin{aligned}
S = n - N(*) &\leq n - \frac{n}{4} && (8)\\
&= \frac{3n}{4} && (9)\\
&= \frac{3}{2}\text{LP-OPT} && (10)\\
&\leq \frac{3}{2}\text{OPT} && (11)
\end{aligned}
$$

   Therefore, this algorithm is a $\frac{3}{2}$-approximation algorithm.

4. There is one example in Figure 1 that shows when the equal sign is reached. For different colors are assigned to these 4 points and our algorithm select 3 of them. The number of vertex cover is 2. Therefore, our algorithm is $\frac{3}{2}$ times of the OPT.

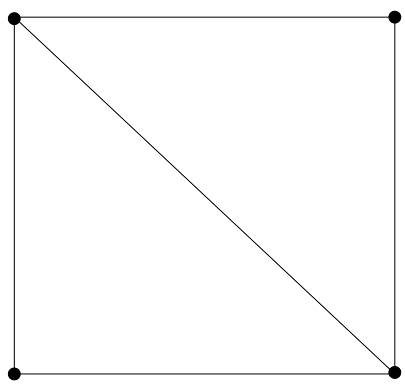

Figure 1: One example