Solutions

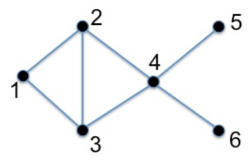
1a For example, $1,2,4$. (Other possibilities: $2,3,4$ and $1,3,4$). Optimal value is 3.

1b

$$\begin{array}{lll} \min & x_1+x_2+x_3+x_4+x_5+x_6 & \\ s.t. & x_1+x_2 \geqslant 1 & \\ & x_2+x_3 \geqslant 1 & \\ & x_1+x_3 \geqslant 1 & \\ & x_2+x_4 \geqslant 1 & \\ & x_3+x_4 \geqslant 1 & \\ & x_4+x_5 \geqslant 1 & \\ & x_4+x_6 \geqslant 1 & \\ & x_i \in \{0,1\} & \text{i=1,2,..., 6.} \end{array}$$

1c For example, $(x_1,x_2,x_3,x_4,x_5,x_6) = (\frac{1}{2},\frac{1}{2},\frac{1}{2},1,0,0)$ with value 2.5.

1d



2a

$$\begin{array}{lll} \text{(D)}\ \max & Z = \sum_{i=1}^{n} y_i & \\ s.t. & \sum_{i:e_i \in S_j} y_i \leqslant w_j & \text{for all } j=1,\dots,m \\ & y_i \geqslant 0 & \text{for all } i=1,\dots,n. \end{array}$$

2b Assume that e_i is not covered. Then, none of the constraints j with $e_i \in S_j$ is tight. But then we can increase y_i^* by a small positive value and obtain a feasible solution with higher value. This contradicts that y^* is optimal.

2c The value of the solution found is

$$\sum_{j\in I} w_j = \sum_{j\in I} \sum_{i:e_i\in S_j} y_i^* \leqslant f \sum_{i=1}^{n} y_i^* = f Z_D^* \leqslant f Z_{LP}^* \leqslant f\,\text{OPT}.$$

The first *equality* above follows since only tight sets S_j were picked. The first *inequality* follows from the fact that each of the y_i^*'s appears at most f times in the summation. The second inequality follows from weak duality. (Also, the algorithm runs in polynomial time and in (b) we already showed that any solution is feasible.)

3a Follows by a reduction from the Hamiltonian Cycle problem. Assume we have an α-approximation algorithm ALG. Given an instance $G=(V,E)$ define an instance of TSP by taking

$$c_{ij} = 1 \text{ if } (i,j) \in E \text{ and } c_{ij} = M \text{ if } (i,j) \notin E,$$

where M is a large number. Let OPT and ALG denote the optimal value and algorithm's value for the TSP instance. Then, the following implications hold.

$$\begin{array}{llllll} \text{G has a Hamiltonian cycle} & \Rightarrow & OPT = n & \Rightarrow & ALG \leqslant \alpha n. \\ \text{G has no Hamiltonian cycle} & \Rightarrow & OPT \geqslant n-1+M & \Rightarrow & ALG \geqslant n-1+M. \end{array}$$

Choose M such that $\alpha n < n-1+M$. For example, $M = \alpha n$.

3b This is done by Christofides' algorithm: (1) Construct a minimum spanning tree T. (2) Find a mincost perfect matching M of the odd-degree vertices of T. (3) Find an Euler tour in the graph with edges $T \cup M$. (4) Shortcut the tour.
Claim 1: length of $T \leqslant$ OPT: If we delete an edge from the optimal tour then we get a path connecting all vertices. Since this is also a tree, the minimum spanning tree has length at most OPT.
Claim 2: length of $M \leqslant$ OPT$/2$: Let O be the odd degree vertices in T. Shortcut the optimal tour on O. This tour consists of exactly two perfect matchings on O. Hence, the length (cost) of M is at most OPT$/2$.
Claim 1+2 $\Rightarrow$ ALG $\leqslant$ OPT + OPT$/2$.

4a The number of constraints is not polynomially bounded. There may be exponentially many simple s,t paths.

4b Given en LP-solution x, a separation oracle either states (correctly) that x is feasible or it gives us a violated constraint. For the given LP-relaxation, a separation oracle should tell wether or not there is a simple s,t path P for which $\sum_{(u,v)\in P} x_{uv} < 1$. This can be done by computing the shortest path from s to t using x for the distances of the edges. If the shortest path has length at least 1 then the solution is feasible and otherwise the shortest path P will be a violated constraint.

4c Consider an edge (u,v) and assume $L(u) \leqslant L(v)$. Then,

$$\Pr(\text{ edge } (u,v) \text{ in cut }) = \Pr(L(u) \leqslant \gamma < L(v)) \leqslant L(v) - L(u) \leqslant x^*_{uv}.$$

The last inequality follows since $L(v)$ is at most the length of the path from s to v via u: $L(v) \leqslant L(u) + x^*_{uv}$. Hence,

$$\mathbb{E}[|W|] = \sum_{(u,v)\in E} \Pr(\text{ edge } (u,v) \text{ in cut }) \leqslant \sum_{(u,v)\in E} x^*_{uv} = Z^*.$$

Although not asked for, you also get points if you showed that W is indeed a feasible cut. Since $L(s) = 0$ and $L(t) \geq 1$ it follows from the definition of S_γ that $s \in S_\gamma$ and $t \notin S_\gamma$ for any $\gamma \in [0,1[$. So W is a feasible cut.

4d Since Z^* is the optimal value of the relaxation we have $Z^* \leqslant$ OPT. With question **4c** this implies $\mathbb{E}[|W|] \leqslant$ OPT. Since W is always a feasible cut we have $|W| \geq$ OPT for any choice of γ. Together with $\mathbb{E}[|W|] \leqslant$ OPT this implies that $W =$ OPT for any choice of γ. Therefore, the derandomized algorithm can fix any value of γ. For example, $\gamma = 0$.

N.B. *It is also fine if you answered here that the derandomized algorithm simply tries many different values of γ and then takes the best solution. But do note here that it is enough to try only the values $L(v)$ for all $v \in V$, which are at most n different values. It is not OK if you answered 'by the method of conditional expectations' without any further explanation.*

5a

$$\begin{array}{lll} \min & Z & \\ s.t. & \sum_{i\in C_e} x_i + \sum_{i\notin C_e} (1-x_i) \leqslant Z & \text{for all edges } e \\ & x_i \in \{0,1\} & \text{for all calls } c_i \\ & Z \geqslant 0 \text{ (not really needed)} & \end{array}$$

5b Algorithm:
(1) Solve the LP-relaxation in which $x_i \in \{0,1\}$ is replaced by $0 \leqslant x_i \leqslant 1$.
(2) Route c_i clockwise if $x_i^* \geqslant 1/2$ and route it counter clockwise otherwise.
For the proof it is convenient to define the value $y_i = 1$ if $x_i^* \geqslant 1/2$ and $y_i = 0$ other wise. Then, $y_i \leqslant 2x_i^*$ and $1 - y_i \leqslant 2(1 - x_i^*)$. The load on an edge e is

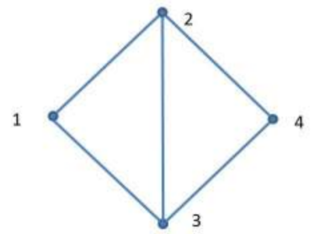
$$\sum_{i \in C_e} y_i + \sum_{i \notin C_e} (1 - y_i) \leqslant \sum_{i \in C_e} 2x_i^* + \sum_{i \notin C_e} 2(1 - x_i^*) = 2Z^* \leqslant 2\text{OPT}.$$

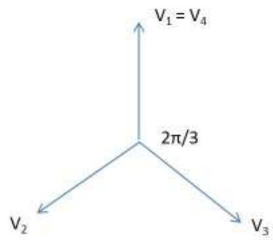
Other algorithms are possible. For example, always choosing the shortest of the two directions is also a 2-approximation.

6a For graph $G = (V, E)$ with $|V| = n$, the relaxation is

$$\begin{array}{lll} \min & \lambda & \\ s.t. & v_i \cdot v_j \leqslant \lambda & \text{for all } (i,j) \in E \\ & v_i \cdot v_i = 1 & \text{for all } i \in V \\ & v_i \in \mathbb{R}^n & \text{for all } i \in V. \end{array}$$

6b For example the graph left. (Actually, any graph with 5 edges is OK here.) The solution (right) has value -0.5. Another example (C_5) is given in the lecture notes.





6c For any edge (i,j) and <u>one</u> random hyperplane:

$$\Pr(v_i \text{ and } v_j \text{ are not separated }) \leqslant \frac{\pi/3}{\pi} = \tfrac{1}{3}.$$

Thus,

$$\begin{aligned} & \Pr(i \text{ and } j \text{ get the same color}) \\ = \;& \Pr(v_i \text{ and } v_j \text{ not separated be either hyperplane }) \leqslant \tfrac{1}{3}\tfrac{1}{3} = \tfrac{1}{9}. \end{aligned}$$

$$\Rightarrow \Pr(\text{endpoints of } \underline{\text{some}} \text{ edge get the same color }) \leqslant 5 \cdot \tfrac{1}{9} = \tfrac{5}{9}.$$

$$\Rightarrow \Pr(\text{coloring is feasible }) \geqslant 1 - \tfrac{5}{9} = \tfrac{4}{9}.$$

– –