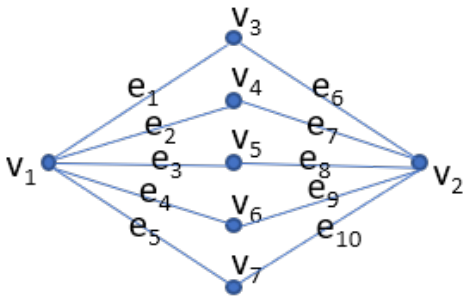
ANSWERS

Answer 1



(a)

(b)

$$
\begin{aligned}
(\mathrm{D}) \quad \max \quad & y_1+y_2+y_3+y_4+y_5+y_6+y_7+y_8+y_9+y_{10} \\
s.t. \quad & y_1+y_2+y_3+y_4+y_5 \le 3 \\
& y_6+y_7+y_8+y_9+y_{10} \le 3 \\
& y_1+y_6 \le 2 \\
& y_2+y_7 \le 2 \\
& y_3+y_8 \le 2 \\
& y_4+y_9 \le 2 \\
& y_5+y_{10} \le 2 \\
& y_i \ge 0 \qquad \text{for } i=1,2,\ldots,10.
\end{aligned}
$$

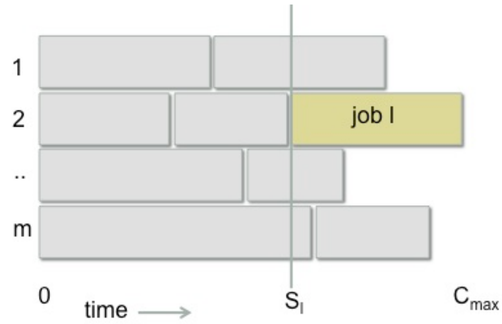
(c)

iter.	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}
0	0	0	0	0	0	0	0	0	0	0
1	2	0	0	0	0	0	0	0	0	0
2	2	1	0	0	0	0	0	0	0	0
3	2	1	0	0	0	0	1	0	0	0
4	2	1	0	0	0	0	1	2	0	0

The dual solution found is $y_1=2, y_2=1, y_7=1, y_8=2$ and all others are zero. The constraints are tight for sets S_1, S_2, S_3, S_4, S_5. The total weight is $3+3+2+2+2=12$.

Answer 2.

Lower Bound 1: OPT $\geq \max_j p_j$
Lower Bound 2: OPT $\geq \sum_{j=1}^{n} p_j/m$



Let l be the job that completes last. See Figure b. Let S_l be the start time of job l in the schedule. Then $C_{\max} = C_l = S_l + p_l$. No machine has a load (length) that is less than S_l since otherwise the list scheduling algorithm could have started l earlier. Hence, $mS_l \leq \sum_{j \neq l} p_j$. Together with the two lower bounds we obtain:

$$
\begin{aligned}
C_{\max} &= S_l + p_l \\
&\leqslant \frac{1}{m}\sum_{j \neq l} p_j + p_l \\
&= \frac{1}{m}\sum_{j} p_j + (1 - \frac{1}{m})p_l \\
&\overset{(\text{LB1})}{\leqslant} \frac{1}{m}\sum_{j} p_j + (1 - \frac{1}{m})\text{OPT} \\
&\overset{(\text{LB2})}{\leqslant} \text{OPT} + (1 - \frac{1}{m})\text{OPT} \\
&= (2 - \frac{1}{m})\text{OPT}.
\end{aligned}
$$

Answer 3

(a) (1) Find a minimum cost spanning tree T.

(2) Find a minimum cost matching M on the veritces of odd degree in T

(3) Add the matching M to the tree T

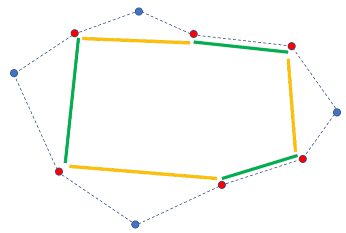
(4) Find an Euler tour in this graph $T+M$.

(5) Cut short.

(b) Removing any edge from the optimal tour gives a tree. So, $\text{cost}(T) \leq \text{OPT}$.

Let V' be the set of vertices of odd degree in T. When we cut the optimal tour short on V' we get a tour on V' of length at most OPT. This tour is composed of 2 matchings on V', say M_1 and M_2. Hence, $\text{OPT} \geq \text{cost}(M_1) + \text{cost}(M_2)$. Since the algorithm computes a minimum cost matching we get: $\text{cost}(M) \leq \min\{\text{cost}(M_1), \text{cost}(M_2)\} \leq \text{OPT}/2$.

Shortcutting does not increase the length since we assume that the triangle inequality holds.



Answer 4

(a) $1/4+1/4+1/4+1/4+1/4 = 5/4$

(b) Let Z be the number of clauses satisfied. Then

$E[Z|x_1 = T] = 1/2+1/2+0+1/4+1/4 = 1.5.$
$E[Z|x_1 = F] = 0+0+1/2+1/4+1/4 = 1.$
$\Rightarrow x_1 =$True.
$E[Z|x_1 = T, x_2 = T] = 1+1/2+0+0+1/2 = 2.$
$E[Z|x_1 = T, x_2 = F] = 0+1/2+0+1/2+0 = 1.$
$\Rightarrow x_2 =$True.
$E[Z|x_1 = T, x_2 = T, x_3 = T] = 1+0+0+0+0 = 1.$
$E[Z|x_1 = T, x_2 = T, x_3 = F] = 1+1+0+0+1 = 3.$
$\Rightarrow x_3 =$False.
3 clauses are satisfied (C_1, C_2 and C_5)

Answer 5

(a) Client j can only be connected to facility i if i is opened.

(b) If $x_{ij} \geq 1/3$ then round it to 1 ($\hat{x}_{ij} = 1$) and also round y_i to 1 ($\hat{y}_i = 1$). Connect each client to some i with $x_{ij} \geq 1/3$.

The solution is feasible since for each j there is at least one i with $x_{ij} \geq 1/3$. This follows from the equality constraint and it is given that there are at most 3 values i with $x_{ij} > 0$.

If $x_{ij} \geq 1/3$ then also $y_i \geq x_{ij} \geq 1/3$. Hence, by this rounding we have $\hat{x}_{ij} \leq 3x_{ij}$ and $\hat{y}_i \leq 3y_i$ for all variables x_{ij}, y_i. The value of the rounded solution is at most 3 times the optimal LP-value, which is at most 3 time the optimal (ILP) value.

Answer 6

(a)

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

Let the 4 vectors correspond to 4 colors: 1,2,3,4. Given a 4-coloring of the graph we give the vector for vertex i the value $u_i = v_k$ if the color of vertex i is k (with $k \in \{1,2,3,4\}$). Then for any edge (i, j) we have $u_1 \cdot u_j = \cos(\alpha) = -1/3$. For this solution, the value of the VP is $-1/3$ so the optimal value is at most $-1/3$.

(b) Let $u_1, u_2, \cdot, u_n$ be an optimal solution. Apply the rounding just as in the Goemans and Williamson algorithm for max cut:

Take a unit length vector $r \in \mathbb{R}^n$ uniformly at random. Partition the vertices depending on $u_i \cdot r \geq 0$ or $u_i \cdot r < 0$. Then:

$$\Pr((i, j) \text{ in cut}) \geq \alpha/\pi > 0.6$$

Thus, the cut contains more than 60 percent of the edges in expectation.