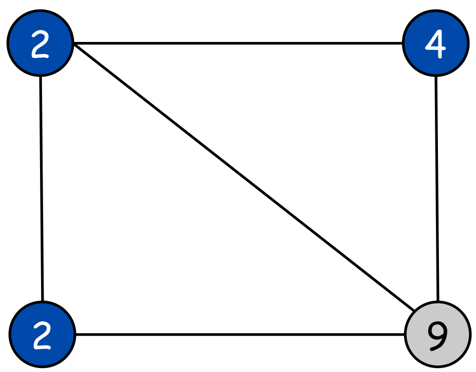
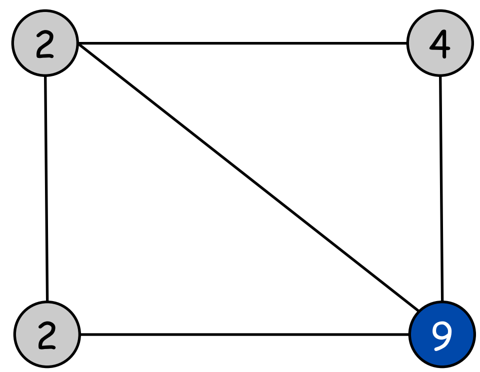
11.4 The Pricing Method: Vertex Cover

Weighted Vertex Cover

Weighted vertex cover. Given a graph G with vertex weights, find a vertex cover of minimum weight.



weight = 2 + 2 + 4



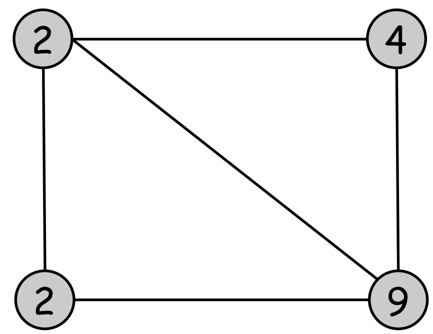
weight = 9

Weighted Vertex Cover

Pricing method. Each edge must be covered by some vertex i. Edge e pays price $p_e \geq 0$ to use vertex i.

Fairness. Edges incident to vertex i should pay $\leq w_i$ in total.

$$\text{for each vertex } i: \sum_{e=(i,j)} p_e \leq w_i$$



Claim. For any vertex cover S and any fair prices p_e: $\sum_e p_e \leq w(S)$.

Proof. ∎

$$\sum_{e \in E} p_e \leq \sum_{i \in S} \sum_{e=(i,j)} p_e \leq \sum_{i \in S} w_i = w(S).$$

each edge e covered by at least one node in S

sum fairness inequalities for each node in S

Pricing Method

Pricing method. Set prices and find vertex cover simultaneously.

$$\sum_{e=(i,j)} p_e = w_i$$

```
Weighted-Vertex-Cover-Approx(G, w) {
   foreach e in E
      pe = 0

   while (∃ edge i-j such that neither i nor j are tight)
      select such an edge e
      increase pe without violating fairness
   }

   S ← set of all tight nodes
   return S
}
```

Pricing Method

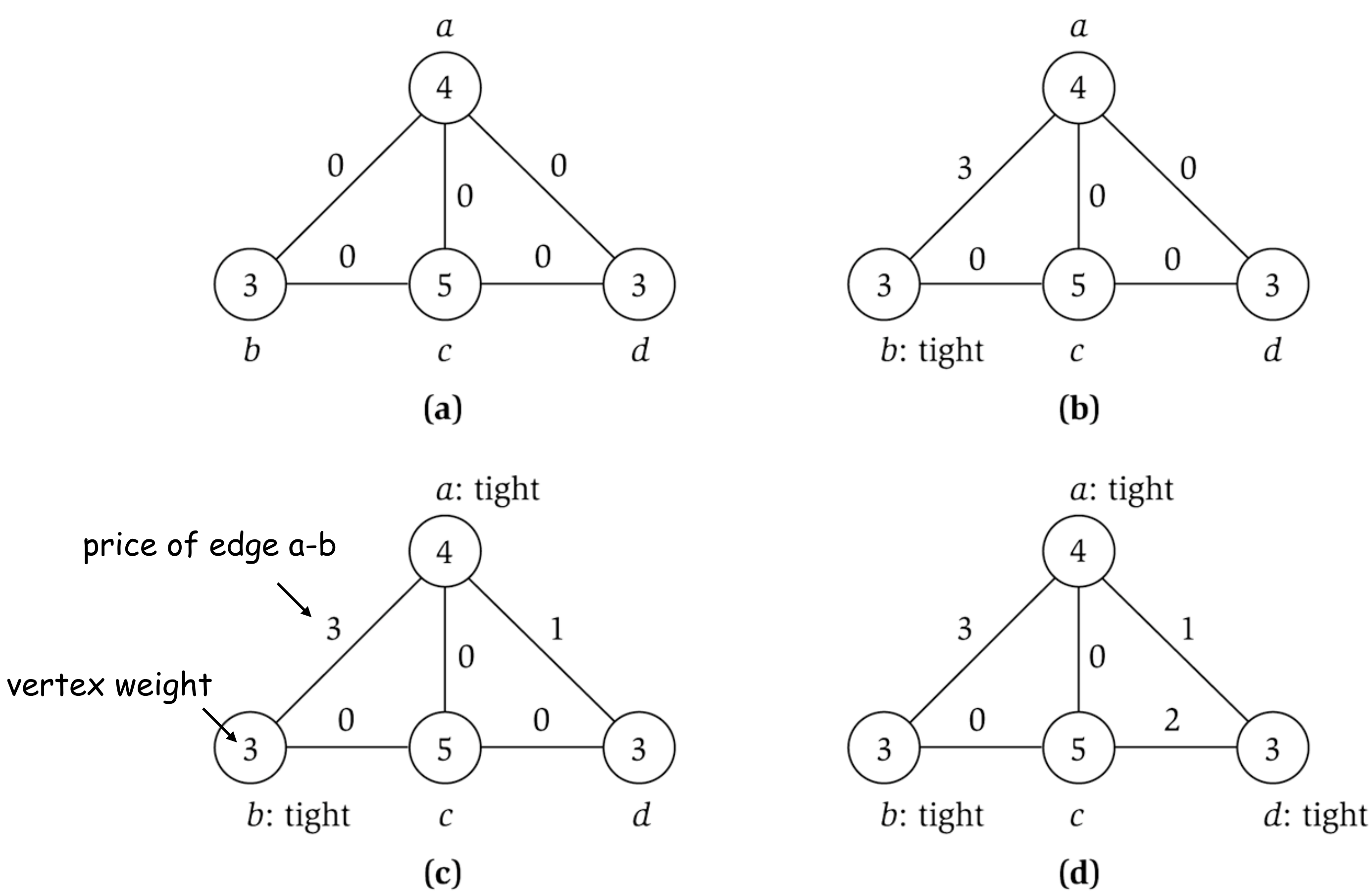


Figure 11.8

Pricing Method: Analysis

Theorem. Pricing method is a 2-approximation.

Pf.

- Algorithm terminates since at least one new node becomes tight after each iteration of while loop.

- Let S = set of all tight nodes upon termination of algorithm. S is a vertex cover: if some edge i-j is uncovered, then neither i nor j is tight. But then while loop would not terminate.

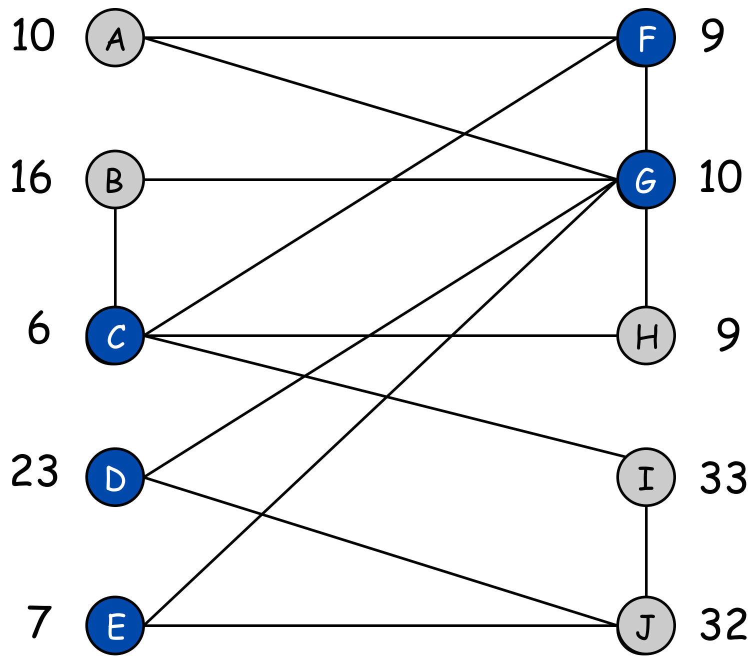
- Let S* be optimal vertex cover. We show w(S) ≤ 2w(S*).

$$w(S) = \underbrace{\sum_{i \in S} w_i =}_{\text{all nodes in S are tight}} \sum_{i \in S} \sum_{e=(i,j)} p_e \underbrace{\leq}_{\substack{S \subseteq V,\\ \text{prices} \geq 0}} \sum_{i \in V} \sum_{e=(i,j)} p_e \underbrace{=}_{\text{each edge counted twice}} 2 \sum_{e \in E} p_e \underbrace{\leq}_{\text{fairness lemma}} 2w(S^*). \quad \blacksquare$$

11.6 LP Rounding: Vertex Cover

Weighted Vertex Cover

Weighted vertex cover. Given an undirected graph $G = (V, E)$ with vertex weights $w_i \geq 0$, find a minimum weight subset of nodes S such that every edge is incident to at least one vertex in S.



total weight = 55

Weighted Vertex Cover: IP Formulation

Weighted vertex cover. Given an undirected graph $G = (V, E)$ with vertex weights $w_i \geq 0$, find a minimum weight subset of nodes S such that every edge is incident to at least one vertex in S.

Integer programming formulation.

- Model inclusion of each vertex i using a 0/1 variable x_i.

$$x_i = \begin{cases} 0 & \text{if vertex } i \text{ is not in vertex cover} \\ 1 & \text{if vertex } i \text{ is in vertex cover} \end{cases}$$

  Vertex covers in 1-1 correspondence with 0/1 assignments: $S = \{i \in V : x_i = 1\}$

- Objective function: maximize $\Sigma_i w_i x_i$.

- Must take either i or j: $x_i + x_j \geq 1$.

Weighted Vertex Cover: IP Formulation

Weighted vertex cover. Integer programming formulation.

$$
\begin{array}{llccl}
(ILP) & \min & \displaystyle\sum_{i \in V} w_i\, x_i & & \\
 & \text{s.t.} & x_i + x_j & \geq & 1 & (i,j) \in E \\
 & & x_i & \in & \{0,1\} & i \in V
\end{array}
$$

Observation. If x^* is optimal solution to (ILP), then $S = \{i \in V : x^*_i = 1\}$ is a min weight vertex cover.

Integer Programming

INTEGER-PROGRAMMING. Given integers a_{ij} and b_i, find integers x_j that satisfy:

$$
\begin{array}{llll}
\max & c^t x & & \\
\text{s.t.} & Ax & \geq & b \\
 & x & & \text{integral}
\end{array}
$$

$$
\begin{array}{rcll}
\sum_{j=1}^{n} a_{ij} x_j & \geq & b_i & 1 \leq i \leq m \\
x_j & \geq & 0 & 1 \leq j \leq n \\
x_j & & \text{integral} & 1 \leq j \leq n
\end{array}
$$

Observation. Vertex cover formulation proves that integer programming is NP-hard search problem.

even if all coefficients are 0/1 and
at most two variables per inequality

Linear Programming

Linear programming. Max/min linear objective function subject to linear inequalities.

- Input: integers c_j, b_i, a_{ij} .
- Output: real numbers x_j.

$$
\begin{array}{llrcl}
(P) & \max & c^t x & & \\
 & \text{s.t.} & Ax & \geq & b \\
 & & x & \geq & 0
\end{array}
$$

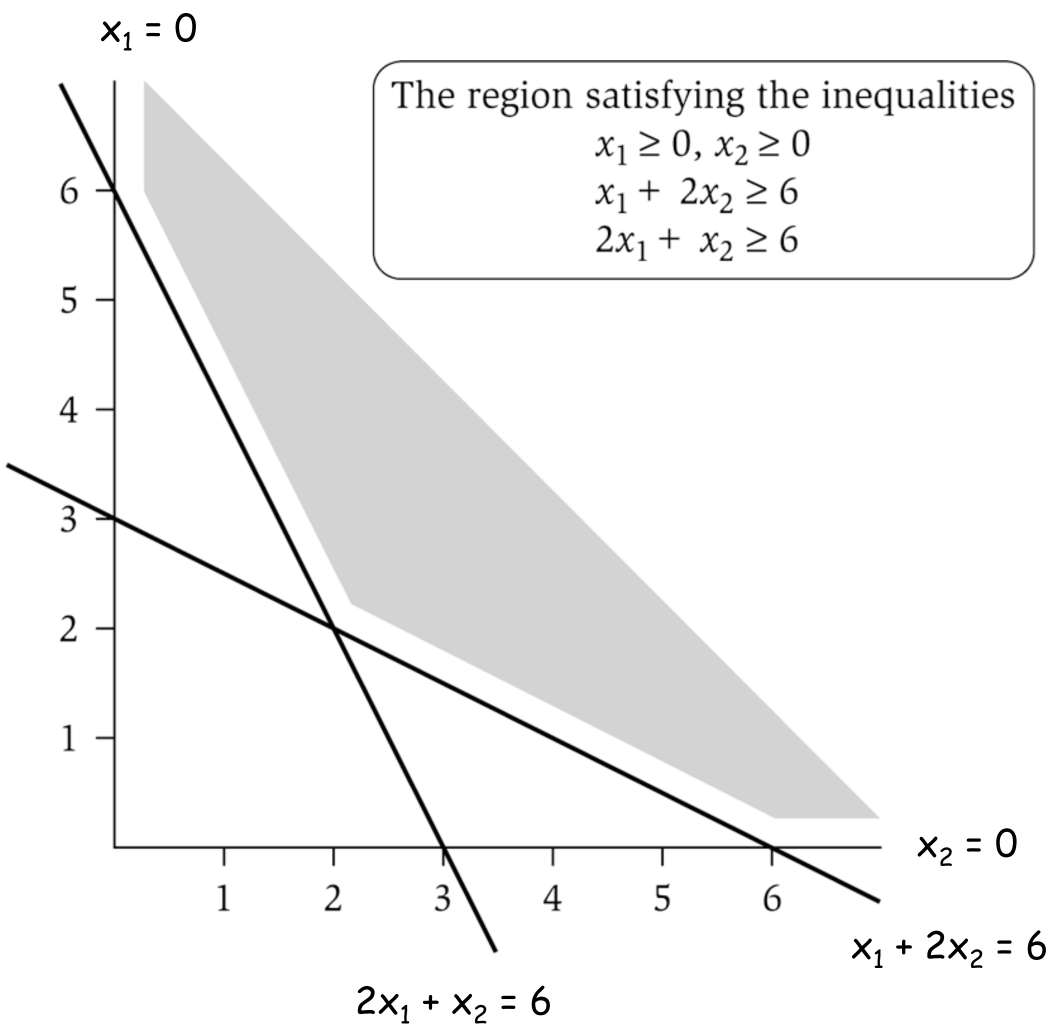
$$
\begin{array}{llrcll}
(P) & \max & \sum_{j=1}^{n} c_j x_j & & & \\
 & \text{s.t.} & \sum_{j=1}^{n} a_{ij} x_j & \geq & b_i & 1 \leq i \leq m \\
 & & x_j & \geq & 0 & 1 \leq j \leq n
\end{array}
$$

Linear. No x^2, xy, $\arccos(x)$, $x(1-x)$, etc.

Simplex algorithm. [Dantzig 1947] Can solve LP in practice.
Ellipsoid algorithm. [Khachian 1979] Can solve LP in poly-time.

LP Feasible Region

LP geometry in 2D.



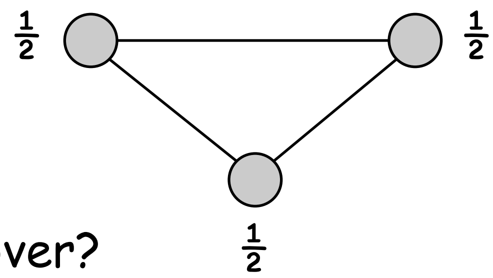
Weighted Vertex Cover: LP Relaxation

Weighted vertex cover. Linear programming formulation.

$$
\begin{array}{llll}
(LP)\ \min & \sum_{i \in V} w_i \, x_i & & \\
\text{s.t.} & x_i + x_j & \geq 1 & (i,j) \in E \\
 & x_i & \geq 0 & i \in V
\end{array}
$$

Observation. Optimal value of (LP) is $\leq$ optimal value of (ILP).
Pf. LP has fewer constraints.

Note. LP is not equivalent to vertex cover.



Q. How can solving LP help us find a small vertex cover?
A. Solve LP and round fractional values.

Weighted Vertex Cover

Theorem. If x^* is optimal solution to (LP), then $S = \{i \in V : x^*_i \geq \frac{1}{2}\}$ is a vertex cover whose weight is at most twice the min possible weight.

Pf. [S is a vertex cover]

- Consider an edge $(i, j) \in E$.
- Since $x^*_i + x^*_j \geq 1$, either $x^*_i \geq \frac{1}{2}$ or $x^*_j \geq \frac{1}{2}$ $\Rightarrow$ (i, j) covered.

Pf. [S has desired cost]

- Let S^* be optimal vertex cover. Then

$$\underbrace{\sum_{i \in S^*} w_i \;\geq\; \sum_{i \in S} w_i x_i^*}_{\text{LP is a relaxation}} \;\geq\; \underbrace{\tfrac{1}{2} \sum_{i \in S} w_i}_{x^*_i \,\geq\, \frac{1}{2}}$$

Weighted Vertex Cover

Theorem. 2-approximation algorithm for weighted vertex cover.

Theorem. [Dinur-Safra 2001] If P $\neq$ NP, then no ρ-approximation for $\rho < 1.3607$, even with unit weights.

$10\sqrt{5} - 21$

Open research problem. Close the gap.