CSC 505, Fall 2000: Week 12

Proving the $\mathcal{NP}$-completeness of a decision problem A:

1. Prove that A is in $\mathcal{NP}$ — give a simple guess and check algorithm (the certificate being guessed should be something requiring very little computation).
2. Pick a known $\mathcal{NP}$-complete problem C and prove that $C \leq_P A$:
   (a) Give a recipe for transforming an instance x of C into an instance $f(x)$ of A. That f can be computed in polynomial time should be obvious.
   (b) Give a recipe for transforming a certificate for x into one for $f(x)$ (usually easy).
   (c) Give a recipe for transforming a certificate for $f(x)$ into one for x (sometimes involves showing that a certificate for $f(x)$ must have a special form).

NP-complete problems related to graph partitioning

MAX-CUT: Given an undirected graph $G = (V, E)$ and an integer K, does there exist a cut $(S, V - S)$ whose cutset $E^* = \{uv \in E \mid u \in S, v \in V - S\}$ has cardinality $\geq K$.

GP (graph partition): Given an undirected graph $G = (V, E)$ and an integer K, does there exist a cut $(S, V - S)$ with $|S| = |V|/2$ and whose cutset has cardinality $\leq K$.

Proof that each of these is in $\mathcal{NP}$ is easy (choose S and verify the appropriate conditions). First we'll reduce **IS** (independent set) to **MAX-CUT**.

Recall **IS:** Given an undirected graph $G = (V, E)$ and an integer K, does there exist $S \subseteq V$ with $|S| \geq K$ and $u, v \in S$ implies $uv \notin E$.

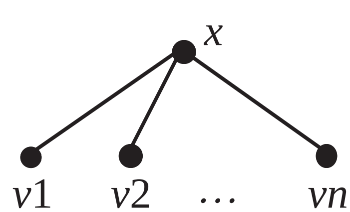
Proof overview

It is easy to show that both **MAX-CUT** and **GP** are in $\mathcal{NP}$: guess the cut (or the set S) and count the number of edges that cross the cut (in the case of **GP**, also make sure that $|S| = |V|/2$).

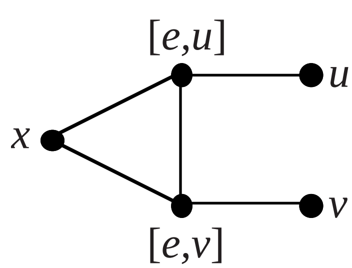
We now prove that **IS** $\leq_P$ **MAX-CUT**
and **MAX-CUT** $\leq_P$ **GP**.

Design of reduction from IS to MAX-CUT

V = {$v1$,$v2$,…,vn} (set of vertices for IS)
maps to



Each edge $e = uv$ maps to a "gadget" that looks like



If u and v are both on the same side of the cut as x, put [e,u] and [e,v] on the opposite side (4 edges cross the cut).
If only u is with x, put [e,u] on the opposite side (same with only v).
If neither u nor v is with x, at most 3 edges will cross the cut.

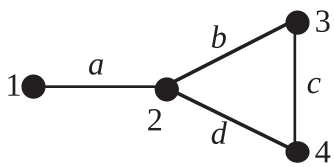
Details of the transformation from IS instance to MAX-CUT instance

Let the instance of **IS** be described by $G = (V, E)$ and k. Then the **MAX-CUT** instance is $G' = (V', E')$ and k', where

$$
\begin{aligned}
V' &= V \cup \{x\} \cup V'' \\
V'' &= \{[e,u], [e,v] \mid e = uv \in E\} \\
E' &= \{xv \mid v \in V\} \cup E'' \\
E'' &= \{x[e,u], x[e,v], [e,u][e,v], [e,u]u, [e,v]v \mid e = uv \in E\} \\
k' &= k + c|E| \qquad \text{where } c = 4
\end{aligned}
$$

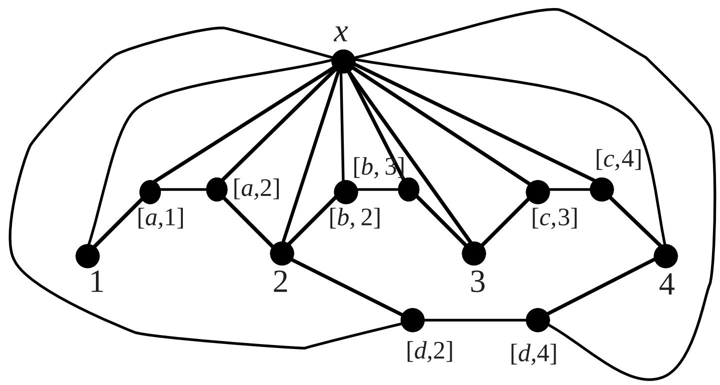
V'' is the set of extra gadget vertices; E'' the gadget edges.

The transformation on an example



G

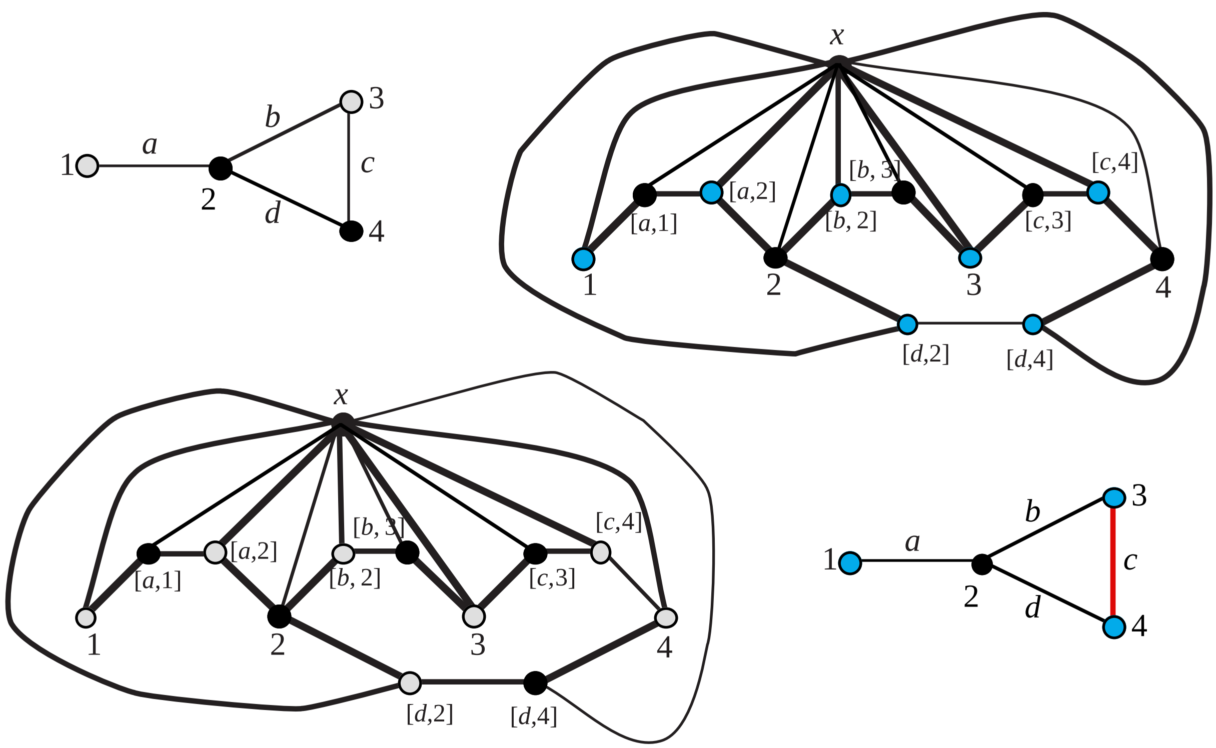
$k = 2$



G'

$k' = 2 + 4 \times 4 = 18$

Mapping of certificates on the example



Proof that IS certificate implies MAX-CUT certificate

Gadget property (for a gadget based on $e = uv$): If at least one of u or v is on the same side of the cut as x, the gadget vertices can be assigned so that c $(= 4)$ gadget edges (but no more) cross the cut. Otherwise at most $c-1$ gadget edges cross the cut.

Let $S \subseteq V$ be the certificate for the **IS** instance. Let $S' = S \cup S''$, where S'' is a set of gadget vertices that will cause c edges from each edge gadget to cross the cut (S'' must exist by the gadget property because each edge has at most one endpoint from S). The total number of edges crossing the cut $(S', V - S')$ in G' will be k (each vertex of S is on the side opposite x) plus $c \cdot |E|$, which adds up to k'. Therefore, S' is a certificate for the **MAX-CUT** instance.

Proof that MAX-CUT certificate implies IS certificate

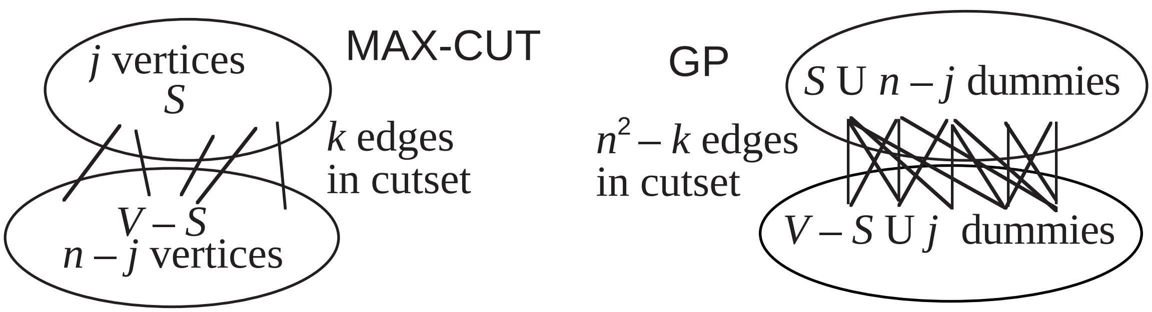
- Let $(S', V - S')$ be the cut that certifies the **MAX-CUT** instance and let E_c be the set of edges crossing the cut. We know that $|E_c| \geq k' = k + 4|E|$.
- Each edge gadget has at most $c = 4$ edges in E_c implies at least k edges of E_c are not gadget edges.
- Non-gadget edges are of the form xv for some $v \in V$. Let S_c be $\{v \in V \mid xv \in E_c\}$ and let $|S_c| = k + \ell$.
- There are $\leq \ell$ edges of G whose gadgets have fewer than 4 E_c (cut) edges. Call these edges $E_{\mathbf{v}}$ (for *violations*).
- For each $e \in E_{\mathbf{v}}$ choose one endpoint and remove it from S_c — call the remaining vertices S.
- S is an independent set — no edge has more than one endpoint in S — and $|S| \geq k$ — $|S_c| \geq k + \ell$ and $\leq \ell$ vertices were removed to form S — so S is the desired certificate.

Idea behind reduction from MAX-CUT to GP

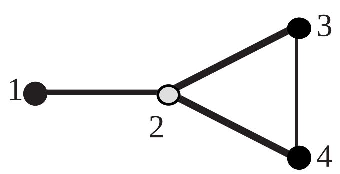
The **MAX-CUT** instance is defined by $G = (V, E)$ and k.

Try using $\overline{G}$ and $k' = \binom{n}{2}$ for the **GP** instance. Large cutset in G corresponds to small cutset in $\overline{G}$.

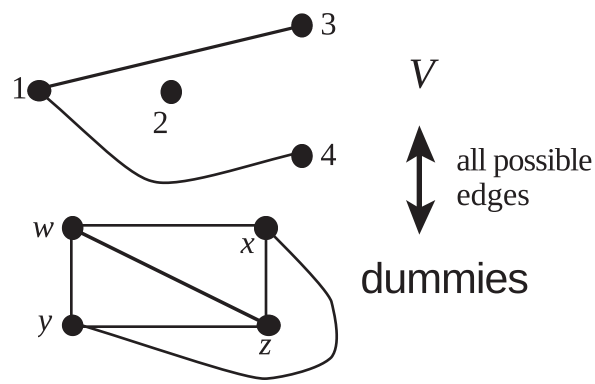
Problem: the converse is not true, a small cutset in $\overline{G}$ does not necessarily translate to a large one in G — for example, a cut that puts a single vertex on one side and everything else on the other will have a small cutset in both G and $\overline{G}$.



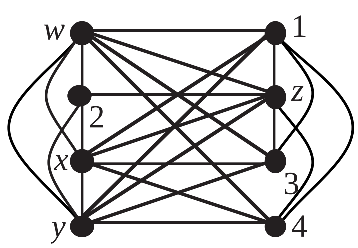
An example to illustrate the reduction



MAX-CUT instance



GP instance



GP instance, showing minimum cut

Details of proof

(a) Let $G = (V, E)$ and k define an instance of **MAX-CUT**. Then $G' = (V', E')$ and k' define an instance of **GP**, where $V' = \underline{V} \cup \{x_1, \ldots, x_n\}$ $(n = |V|)$,
$E' = \overline{E} \cup \{x_i x_j \mid i \neq j\} \cup \{x_i u \mid u \in V\}$, and $k' = n^2 - k$.

(b) Let $(S, V - S)$ be a certificate for the **MAX-CUT** instance and let $\ell = |S|$. Then
$(S \cup \{x_1, \ldots, x_{n-\ell}\}, V - S \cup \{x_{n-\ell+1}, \ldots, x_n\})$ is a certificate for the **GP** instance. Why? At least k of the potential n^2 edges between the two sides of the cut are not present (not in $\overline{E}$).

(c) Let $(S', V' - S')$ be a certificate for the **GP** instance. Then $(S' \cap V, (V - S') \cap V)$ is a certificate for the **MAX-CUT** instance. The edges of G' that cross the cut $(S', V' - S')$ include all possibilities except for at least k missing edges. Because G' includes every possible edge with an x_i (dummy) vertex as an endpoint, the missing edges could only involve two vertices of V and must therefore be edges from E (edges not in $\overline{E}$).

A completely different problem: Subset Sum

Subset Sum (SUM): Given a set S of integers and a target integer t, does there exist $S' \subseteq S$ with $\sum_{x \in S'} x = t$?

SUM is a special case of the famous *knapsack problem*: Given n objects with weights w_i, $1 \leq i \leq n$, and values v_i, $1 \leq i \leq n$, a capacity c, and a target value t, does there exist a *packing*, a subset P of $\{1, \ldots, n\}$ (the objects) such that

$$\sum_{i \in P} w_i \leq c \text{ and } \sum_{i \in P} v_i \geq t$$

To turn an instance of **SUM** into a knapsack instance, let S, in the **SUM** instance, be $\{x_1, \ldots, x_n\}$. To get the knapsack instance, let $v_i = w_i = x_i$ for $1 \leq i \leq n$ and let $c = t$ (use the same target t in knapsack as in **SUM**).

Subset sum is NP-complete

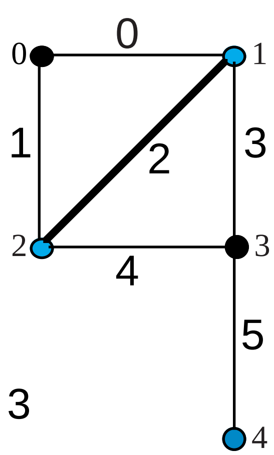
SUM is in $\mathcal{NP}$: guess the subset S' and see if the numbers add up to t.

We can show that **VC** $\leq_P$ **SUM** (see CLR, pp. 951–953).

(a) Let $G = (V, E)$ and k define an instance of **VC**. Number the vertices from 0 to $n-1$ and the edges from 0 to $m-1$. Let $S = \{x_0, \ldots, x_{n-1}\} \cup \{y_0, \ldots, y_{m-1}\}$. Each x_i consists of $m+1$ digits (in base 10 to make it easy, but any base ≥ 4 will work) and can be written as $x_{i,m}x_{i,m-1}\ldots x_{i,0}$. The digit $x_{i,m}$ is always 1. Each remaining $x_{i,j}$ is 1 if vertex i is an endpoint of edge j, 0 otherwise. Each y_i has $i+1$ digits: a 1 followed by i 0's. Finally, let t be the (base 10 representation of the) integer k followed by m 2's.

The reduction on an example

Vertex Cover instance



k = 3

Subset Sum instance

$$x_0 = 1000011$$
$$x_1 = 1001101$$
$$x_2 = 1010110$$
$$x_3 = 1111000$$
$$x_4 = 1100000$$
$$y_0 = 1$$
$$y_1 = 10$$
$$y_2 = 100$$
$$y_3 = 1000$$
$$y_4 = 10000$$
$$y_5 = 100000$$
$$t = 3222222$$

k = 3

Details of certificate mappings

(b) Let V' be the **VC** certificate. Then let
$S' = \{x_i \mid \text{ vertex } i \in V' \}$
$\quad \cup \{y_i \mid \text{ only one endpoint of edge } i \text{ is in } S' \}$.
Since there are three 1's in digit positions 0 through $m-1$, there will be no carries from those positions. The choice of S' items guarantees that each of these digit positions has a sum of 2, as required by t. Since $|V'| = k$, the x_i's in S' will contribute exactly k 1's in position m for a total of k. Thus S' is a certificate for the **SUM** instance.

(c) Let S' be the **SUM** certificate. Then let V' be the set of all vertices numbered i for which $x_i \in S'$. Because there are no carries in the lowest m digits, there must be exactly k vertices in V' (to get t to start with k) and each edge must have at least one endpoint in V' (if edge i has no endpoints in V' then S' has only a single 1 among all the i-th digits and the sum of S' cannot have a 2 in that position).