CS3100

More reductions
Announcement

- HW due Wednesday

For HW: Show A is NP-Complete

1. in NP - "yes" solution can be checked in poly time
2. NP-Hard if reduce known NP-Hard problem to A

In general: may use any listed problems in lecture notes
Graph reduction:
- Ind. Set
- Clique
- Vertex Cover

In lecture notes:
- Hamiltonian cycle
- Traveling salesman

Given a weighted graph, design a tour that visits every vertex of minimum length
Subset Sum:

Given a set of numbers
\[ X = \{ x_1, x_2, x_3, \ldots, x_n \} \]
and a target \( t \), does some subset of \( X \) sum to \( t \)?

Ex: (actually did this one! see lecture on Sept 8)

Runtime: exponential

in \( N \)
Subset Sum is NP-Hard.

Reduction: Vertex Cover

Input: Graph $G$ of size $k$.

Construct a set of numbers: $X$

Label each edge $0, \ldots, m-1$

Add a number to $X$ for each $e_i$: $4^i = b_i$

Add a number to $X$ for each vertex $v$: $a_v: 4^m + \sum_{e_i \text{ adjacent to } v} 4^i$

View these as base 4:

edges $e_i$: $0 \ 0 \ 0 \ 0 \ 1 \ 0 \ \ldots \ 0$

$v: \frac{1}{m}$ of edges
Nice feature: for $i < m$, only 3 1s anywhere in $X$  
$\Rightarrow$ no carrying for any subset of these!

Set $t = k \cdot 4^m + \sum_{i=0}^{m-1} 2 \cdot 4^i$

Poly time conversion:

$n + m$ numbers
if cover of size \( k \) exists, \( \Rightarrow \) subset of value \( t \)

Take a vertex cover in \( G \).

For each \( v \), choose \( a_v \)

is subset of \( X \).

Have \( \geq k \cdot 4^m + \sum_{i=0}^{m-1} 1 \cdot 4^i \)

For every edge \( w \) with only 1 endpoint in cover, also add \( b_w \) to subset.

\( \Rightarrow \) Sum is exactly \( k \cdot 4^m + \sum_{i=0}^{m-1} 2 \cdot 4^i \)

Let \( \sum \) spps subset \( X \subseteq X \)

sums to \( t = k \cdot 4^m + \sum_{i=0}^{m-1} 2 \cdot 4^i \)

Know choose exactly \( t \) of \( a_v \)s

since lower terms can't carry.

These are a cover, since including \( b_i \) isn't enough to hit \( 2 \cdot 4^i \)
Another: Partition

Given a set of $n$ numbers, can you partition into 2 sets $X + Y$ so that

$$\sum_{x \in X} x = \sum_{y \in Y} y.$$ 

Easy reduction:

On worksheet...
Set Cover:
Given a set $U$ of $n$ elements, a collection $S_1, S_2, \ldots, S_m$ of subsets of $U$, and a number $k$, is there a collection of $k$ of the $S_i$'s whose union is all of $U$?

Ex:
Elements in $U$:
Subsets $S_1, \ldots, S_7$ and $k = 3$.

Answer? Yes!
Set Cover is NP-Hard: Reduction from vertex cover, so input is $G$ and $k$.

Construct:

$$U = \text{edges} = \{e_1, \ldots, e_m\}$$

$$S_i = \{\text{edges adjacent to vertex } e_i\}$$

$$k = k \text{ (same)}$$

Diagram:

- $V_1, V_2, V_3$
- $e_1, e_2, e_3, e_4, e_5, e_6$

$$U = \{e_1, \ldots, e_6\}$$

$$S_1 = \{e_1, e_2\}$$

$$S_2 = \{e_1, e_3\}$$
Vertex cover of size $k$ \( \iff \) Set cover of size $k$
Some fun examples

**Classical Nintendo Games are (Computationally) Hard**

Greg Aloupis, Erik D. Demaine, Alan Guo, Giovanni Viglietta

(Submitted on 8 Mar 2012 (v1), last revised 8 Feb 2015 (this version, v3))

We prove NP-hardness results for five of Nintendo’s largest video game franchises: Mario, Donkey Kong, Legend of Zelda, Metroid, and Pokemon. Our results apply to generalized versions of Super Mario Bros. 1-3, The Lost Levels, and Super Mario World; Donkey Kong Country 1-3; all Legend of Zelda games; all Metroid games; and all Pokemon role-playing games. In addition, we prove PSPACE-completeness of the Donkey Kong Country games and several Legend of Zelda games.

**Submission history**

From: Alan Guo [view email]

[v2] Thu, 6 Feb 2014 18:24:15 GMT (3309kb,D)
[v3] Sun, 8 Feb 2015 19:45:26 GMT (3425kb,D)

Which authors of this paper are endorsers? I Double MathJax (What is MathJax?)

Link back to: arXiv, form interface, contact.

Left: Start gadget for Super Mario Bros. Right: The item block contains a

Figure 9: Finish gadget for Super Mario Bros.

Figure 10: Variable gadget for Super Mario Bros.

Figure 11: Clause gadget for Super Mario Bros.

Figure 12: Crossover gadget for Super Mario Bros.
Another: Tetris

NP-Hard: Reduce 3-partition

Fig. 2. The initial gameboard for a Tetris game mapped from an instance of 3-Partition.

Fig. 3. A valid sequence of moves within a bucket.
Next time: