CS173 Lecture B, October 29, 2015

Tandy Warnow

October 28, 2015

CS 173, Lecture B
October 29, 2015
Tandy Warnow

Karp Reduction

Let π and π' be two decision problems.
A Karp reduction from π to π' in a function f that satisfies the following requirements:

- The domain of f is the set of valid inputs to π.
- The co-domain of f is the set of valid inputs to π'.
- The function f can be computed in polynomial time (in the size of the input).
- The size of $f(I)$ is no more than a polynomial in the size of I, for any I in the domain of f.
- For any valid input I to problem π, I is a YES-instances to π if and only if $f(I)$ is a YES-instances to π'.

Karp Reduction

Recall the Karp reduction from 2-colorability to 3-colorability. The valid inputs to each problem are just graphs (any graphs). The function f has to satisfy:

1. f maps graphs to graphs
2. We can compute $f(G)$ is polynomial time (so polynomial in the number of vertices of G)
3. The size of $f(G)$ is no more than a polynomial in the size of G (equivalently, the number of vertices in $f(G)$ is at most a polynomial in n, where n is the number of vertices in G)
4. G can be 2-colored if and only if $f(G)$ can be 3-colored

We defined the Karp reduction f by: $f(G)$ is the graph obtained by adding a new vertex to G and making it adjacent to every vertex in G.

Karp reduction

We defined the Karp reduction f by: $f(G)$ is the graph obtained by adding a new vertex v^* to G and making it adjacent to every vertex in G.

It is easy to see that the first three properties hold. For the fourth, we need to show

- If G can be 2-colored, then $f(G)$ can be 3-colored.
- If $f(G)$ can be 3-colored, then G can be 2-colored

Karp reduction

$f(G)$ is the graph obtained by adding a new vertex to G and making it adjacent to every vertex in G. We need to show

- If G can be 2-colored, then $f(G)$ can be 3-colored.
- If $f(G)$ can be 3-colored, then G can be 2-colored

Here are the proofs.

- Suppose G can be 2-colored. To color $f(G)$, just use the 2-coloring for G and then use the third color for the new vertex in G'. Voilá!
- Suppose $f(G)$ can be 3-colored. We want to show that G can be 2-colored. Let c be a proper 3-coloring for $f(G)$. Then c is a proper coloring for G, too (since G is an induced subgraph of $f(G)$). I claim c only uses two colors on G. To see this, note that the new vertex v^* must get a color that is different from every other vertex in $f(G)$. Hence, there are only two colors used in G. Thus, G is 2-colorable.

Formulating real world problems as graph problems

Consider the following problems, all based on the set of people in this class, and under the assumption where you know who likes who in the class (and you assume this is mutual).

- You want to find a set of people in this class so that between them they know everyone in the class, and the set is as small as possible.
- You want to partition the set of people into subsets so that every two people in any subset both like each other, and make the number of subsets as small as possible.
- You want to pair people off in the class so that they will study together. You want to have as large a number of people able to be in study groups, but you have the following rules: study groups must have pairs of people who like each other, and no person can be in two study groups.
- You want to have a party and invite as many people as you can to it, subject to (a) they all like you, and (b) they all like each other.

Formulating real world problems as graph problems

You want to find a set of people in this class so that between them they know everyone in the class, and the set is as small as possible. Assume you know who likes who.
Challenge: describe this as a graph problem.

Solution:
The graph $G = (V, E)$ is defined by:

- Let V denote all the people in the class.
- Put an edge between v and w if v and w know each other.

We are looking for the smallest $V_0 \subseteq V$ such that $\forall v \in V - V_0$, $\exists w \in V_0$ such that $(v, w) \in E$.

- Does a solution always exist? YES!
- Does it have to be unique? NO
- What graph problem does this look like? DOMINATING SET

Describing a real world problem

You want to partition the set of people into subsets so that every two people in any subset both like each other, and make the number of subsets as small as possible.

Solution: The graph $G = (V, E)$ is defined by

- V is the set of people in the class
- E contains (v, w) if and only if v and w like each other

We are looking for a partition of V into a small number of sets so that every one of the sets is a clique.
In other words, we want to write $V = V_1 \cup V_2 \cup \ldots \cup V_k$, where V_i is a clique in G and where k is minimized.

- Does a solution always exist?
- Does it have to be unique?
- What graph problem does this look like?

A different solution

You want to partition the set of people into subsets so that every two people in any subset both like each other, and make the number of subsets as small as possible.

Solution: The graph $G = (V, E)$ is defined by

- V is the set of people in the class
- E contains (v, w) if and only if v doesn't like w or w doesn't like v (i.e., they don't like each other)

We are looking for a partition of V into a small number of sets so that every one of the sets is an independent set.
Note - this is the same thing as finding a minimum **vertex coloring**.

Things to note

When you formulate a real world problem as a graph problem, you have to:

- Describe the graph precisely. What are the vertices? What are the edges? Do the edges have weights? Use correct terminology (don't be sloppy about language).
- Once the graph is defined, the problem (whether a decision problem, optimization problem, or construction problem) is then defined only in terms of the graph and not in terms of the original problem.

The power in making this formulation is that there are many algorithms (and software) for most natural graph problems, and so you can use those programs to solve your problem.

Many graph problems are NP-hard, but sometimes you have extra structure in your problem that allows you to solve the problem in polynomial time. (For example, MIN VERTEX COLORING is NP-hard, but solvable in polynomial time on trees.)

Toy Example

You work for some spy agency, and you want to listen in on all the phone calls happening in Urbana. You can put bugs in cellphones, and the bugs will let you listen to any conversation that takes place using that cellphone. You know which people call which people, and so you can try to use that information to reduce the number of bugs you need to buy and install. You assume everyone has exactly one cellphone and all calls are made using cellphones to cellphones. Formulate this as a graph problem.

- What are the vertices? (Answer: Cellphones)
- What are the edges? (Answer: pairs of cellphones where their owners are known to call each other.)
- What are you looking for?
  - Answer: the smallest number of cellphones so that all phone calls involve at least one cellphone in the set.
  - Better answer: the smallest set $V_0 \subseteq V$ of vertices so that every edge in E has at least one endpoint in V_0.

Do you recognize this problem?