CS 512: Independent Sets and Randomized Algorithms

16:198:512

Instructor: Wes Cowan

Name:
RUID:

Consider an undirected graph $G = (V, E)$. An **independent set** is a subset $I \subset V$ such that for any vertices $i, j \in I$, there is no edge between i and j in E. A set i is a **maximal independent set** if no additional vertices of V can be added to I without violating its independence. Note, however, that a maximal independent sent is not necessarily the *largest* independent set in G. Let $\alpha(G)$ denote the size of the largest maximal independent set in G.

1) What is $\alpha(G)$ is G is a complete graph on n vertices?

2) Argue that if G is a cycle, then any maximal independent set is as large as the largest maximal independent set in G.

Consider the following greedy algorithm for generating maximal independent sets: starting with an empty set I, process the vertices in V one at a time, adding v to I is v is not connected to any vertex already in I.

3) Argue that the output I of this algorithm is not only an independent set, it is necessarily a maximal independent set.

One significant weakness of this algorithm is that the result depends heavily on the order the vertices are processed. *Note, the order of vertex processing is vitally important to a number of graph algorithms; Dijkstra, Bellman-Ford, linearization on DAGs for shortest paths - all these can be thought of as trying to construct a good ordering of vertices so that the output of the resulting algorithm is correct.*

4) Construct an example of a graph on n vertices such that running this greedy algorithm on the vertices in one order yields an independent set of size 1, and processing the vertices in a different order yields an independent set of size $n - 1$, which is maximal.

One way of trying to avoid this dependence on ordering is the use of *randomized algorithms*. Essentially, by processing the vertices in a random order, you can potentially avoid (with high probability) any particularly bad orderings. So consider the following randomized algorithm for constructing independent sets:

Step 1) Starting with an empty set I, add each vertex of G to I independently with probability p.

Step 2) For any edges with both vertices in I, delete one of the two vertices from I.

5) Argue that the output of this algorithm is an independent set. Is it a maximal independent set?

6) Argue that the expected number of vertices in I after Step 1 is $p|V|$.

7) Argue that the expected number of *edges* in I after Step 1 is $p^2|E|$.

8) Argue that the expected size of S after Step 2 is $p|V| - p^2|E|$.

9) Argue that with the appropriate choice of p, the expected size of I is $|V|^2/(4|E|)$. What is this in terms of the *average degree* of G?

10) Argue the following general theoretical bound based on applying this algorithm to any graph G:

$$\alpha(G) \geq \frac{|V|^2}{4|E|}. \tag{1}$$

11) How can this algorithm be (efficiently) improved? What is the improved algorithm's complexity?

12) Consider applying the improved algorithm to the example in (4). What is the probability of discovering a largest maximal independent set?