Anthony Rusignuolo

CS 600WS – Advanced Algorithms

Reza Peyrovian

Homework 9

I pledge my honor that I have abided by the Stevens Honor System.

1. R-17.3 Show that the problem SAT, which takes an arbitrary Boolean formula S as input and asks whether S is satisﬁable, is NP-complete.
   1. A Boolean formula can always be reduced to a Boolean circuit, thus making SAT a CIRCUIT-SAT problem which is confirmed to be NP-Complete.
2. R-17.7 Show that the CLIQUE problem is in NP.
   1. Lemma 17.2 holds, as there is a non-deterministic algorithm for accepting an instance of a clique, C, that uses the choose method to “guess” whether there is an edge between every pair of vertices in C, and then checks whether those “guesses” are true in polynomial time.
3. C-17.10 Deﬁne INDEPENDENT-SET as the problem that takes a graph G and an integer k and asks whether G contains an independent set of vertices of size k. That is, G contains a set I of vertices of size k such that, for any v and w in I, there is no edge (v, w) in G. Show that INDEPENDENT-SET is NP-complete.
   1. In order to prove Independent-set is NP-Complete, Independent-set must be proven to be an NP and NP-Hard problem. Using lemma 17.2 proves that the problem is within NP. Reducing the Vertex-cover problem to Independent-set proves that Independent-set is NP-Hard. Therefore, the problem is NP-Complete.
4. A-17.5 Suppose you are computer security expert working for a major company, CableClock, any you have just discovered that many of the computers at CableClock are infected with malware that must have come from users visiting unsafe web-sites. For each infected computer, you are given a log ﬁle that lists all websites it has visited since the last time it was scanned for malware. Unfortunately, as you look over these log ﬁles, you notice that there isn’t a single website that they all visited. You conclude, therefore, that there must be a number of websites that are able to inject this malware, and the most likely candidates would be in a smallest collection that is visited by all the infected computers. Show that the decision version of the problem of determining such a collection is NP-complete.
   1. This is a set-cover problem, which is known to be NP-Complete.
5. R-18.11 Suppose we are given the following collection of sets:

S1 = {1, 2, 3, 4, 5, 6},  
S2 = {5, 6, 8, 9},  
S3 = {1, 4, 7, 10},

S4 = {2, 5, 7, 8, 11},  
S5 = {3, 6, 9, 12},  
S6 = {10, 11}.  
What is the optimal solution to this instance of the SET-COVER problem and what is the solution produced by the greedy algorithm?

* 1. Optimal: S3, S4, S5  
     Greedy: S1, S4, S5, S3

1. C-18.1 Consider the general optimization version of the TSP problem, where the underlying graph need not satisfy the triangle inequality. Show that, for any ﬁxed value δ ≥ 1, there is no polynomial-time δ-approximation algorithm for the general TSP problem unless P = NP.  
   Hint: Reduce HAMILTONIAN-CYCLE to this problem by deﬁning a cost function for a complete graph H for the n-vertex input graph G so that edges of H also in G have cost 1 but edges of H not in G have cost δn more than 1.
   1. Using the hint above, if all the edges of H are in G the total cost would be n. Otherwise, the total cost would be n\*(1+ δn). With that in mind, yes instances of Hamiltonian Cycle correspond to instances of TSP of at most nk, for some k. However, no instances of Hamiltonian cycle correspond to instance of TSP with at least δnk. Therefore, there’s no polynomial approximation algorithm for TSP unless P = NP.
2. A-18.3 Suppose you work for a major package shipping company, FedUP, and it is your job to ship a set of n boxes from Rhode Island to California using a given collection of trucks. You know that these trucks will be weighed at various points along this route and FedUP will have to pay a penalty if any of these trucks are overweight. Thus, you would like to minimize the weight of the most heavily loaded truck. Assuming you know the integer weight of each of the n boxes, describe a simple greedy algorithm for assigning boxes to trucks and show that this algorithm has an approximation ratio of at most 2 for the problem of minimizing the weight of the most heavily loaded truck.
   1. The Greedy algorithm would be to place any new packages in the least weighted truck. A traditional algorithm would place all the packages in a truck until it is full therefore, the ratio of traditional to Greedy would be 2/1.