CS 6200: Algorithmics II	Fall 2020
Instructor: Avah Banerjee, S&T.	
Lectures: 23	Dates: 10/19

1 NP-complete problems

Theorem 1.1 *3-conjunctive normal form satisfiability is NP-complete*

Proof: We start the proof from the point where we know that SAT problem is NP-complete.From previous Lemma, to show that 3-SAT problem is NP complete,it is enough to show that SAT $\leq_p$ 3-SAT ie., the SAT problem can be reduced to 3-SAT problem in polynomial time.
Let, ϕ be the input formula to the SAT and $C_i = t_1 V t_2 V ... t_k$ be a clause.A SAT (specifically k-SAT problem contains k literals) can be reduced to a 3-SAT problem by considering the following 2 cases.
Case 1: if k$\leq$ 2
We have, C_i= $t_1 \bigvee t_2$
this can be converted to 3 literals by duplicating any of the literal
$D_i = t_1 \bigvee t_2 \bigvee t_2$.
Hence,it is reduced to a 3-SAT problem.
Case 2: if k>3, we can have several clauses with different combination of k literals.Let us consider one such clause
$C_i = t_1 \bigvee t_2 \bigvee t_3 ......t_k$
We can convert C_i into k-2 sets each containing 3 literals by adding additional variables in the following way
$C_i \iff (t_1 \bigvee t_2 \bigvee S_1) \bigwedge(\neg S_1 \bigvee t_3 \bigvee S_2) \bigwedge ....(\neg S_{k-3} \bigvee t_{k-1} \bigvee t_k)$

- Let C_i be satisfialbe that means any one of the literals $t_1 to t_n$ can be true .Let t_3 be true. From second equation we can see that additional variable $\neg s_1 and_2$ can be false and hence their negations in the other clauses will be true that implies that the truth assignments are propagating along both directions in other clauses making the entire equation satisfiable.
- Let all the clauses in the second equation be true.If we closely observe the additional variables it cannot make every clause true (because if the additional variable in one clause is true then its negation in other clauses will be false.This implies that to make every clause true there must be atleast one main literal that is true thus making the first equation satisfiable.

Thus, we prove that 3-SAT problem is NP-complete. ■

Theorem 1.2 *INDEPENDENT SET $\in$ NP-complete*

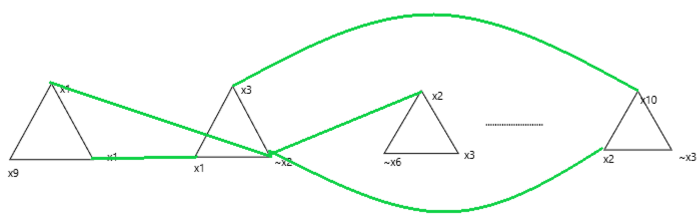
Proof: We prove this by proving 3-SAT $\leq_p$ INDEPENDENT-SET

Definition 1.1 *Independent Set or Stable set is a set of vertices in a graph, no two of which are adjacent. That is, it is a set S of vertices such that for every two vertices in S, there is no edge connecting the two.*

The independent set problem can be formulated into a graph G=(V,E) ,where independent set is the set of vertices that do not have edges between them.There is possibility of having independent sets of size k in graph G. In the compliment graph of G ,we will then have a clique of size k.Therefore, to have an independent set it is enough to to show a clique of size k in the compliment graph.

To prove this we construct the following graph in the following setting

- Every triangle contains vertices that are part of a clause.
- We draw an edge between every literal and its negation ,thus we guarantee that the any independent set contains either a literal or its negation but not both.



We need to prove ϕ is satisfiable $\iff$ each clause has true literal.

- Case 1 : Choose from each clause a true literal .Let the set of all such true literals is $< V_1, V_2, V_3 .....V_m >$.The graph corresponding to these vertices is independent because our initial setting for the graph has edges only between a literal and its negation,So there is no chance for having a edge between two true literals.
- Case 2: If every clause has a true literal,then the outcome of each literal is always true and hence ϕ is satisfiable.

■

Definition 1.2 *A 3D matching is a generalization of bipartite matching to tri partite hyper graphs.If X, Y, Z are three disjoint sets such that $|X| = |Y| = |Z| = n$.A 3D matching is defined as 3DM=$\{< G, K > | G$ has a 3D matching of size $\geq k$ $\}$*

Theorem 1.3 *3D Matching is NP complete*

Proof: 3D matching is NP because given a collection of n edges we can verify that it covers every vertex exactly once can be verified using a polynomial time algorithm.
We take 3 SAT problem and reduce it to 3D matching in polynomial time 3-SAT $\leq_p$ 3DM
3-SAT instance : Given a ϕ we create a 3D Matching.If ϕ has n variables and m clauses.Consider the hyper graph construction H_{x_i}.It has x_i variable gadgets and $2k_i$ triangles. For any matching

We can only choose either the even triangles or the odd triangles of the gadget.The vertices in every gadget that is not shared is called as tip.Let the vertices in even triangles be true literals and those in odd triangles be false literals. Using this gadgets we now create a clause gadget as follows.

- $C_j = x_i \bigvee x_2 \bigvee \neg x_3$ where the three individual literals are the vertices of 3 different gadgets.
- Even if we satisfy all the clauses we may have extra tips left over.In order to cover them too we create $2m$ additional pair of vertices and connect each pair with every tip to create a triangle.

Thus complete Hyper graph H contains the following edges
H=$\{\bigcup H_{x_i} \bigcup H_{C_i} \bigcup H_s\}$

- All the edges in the gadgets H_{x_i}
- Edges for forming clause gadget H_{C_i}
- Additional triangles H_s

We need to show that ϕ is satisfiable $\iff$ H has a perfect 3D matching.

- Case 1: Suppose ϕ is satisfiable, we choose either even or odd gadget corresponding each variable gadget basing on opposite of the assignment of x_i.Then at least one free tip would be available which can be covered using clause gadget.We then use the additional edges to cover all the rest of tips.Hence there is a 3D matching.
- Case 2: Suppose there is a 3D matching
  If variable gadget H_{x_i} has a even triangle in its matching then the clauses containing the even tips should have the same literal that of x_i.If the even triangle corresponds to x_i make it false then any clause containing $\neg x_i$ will have a odd triangle and the clause would be satisfiable.

■

Theorem 1.4 *Let S=$\{s_1, s_2, ...\}$ be a set of integers Is there any subset T of S such that $\sum_{s_i \in T} = k$.Prove this problem is NP complete.*

Proof: Given a set T we can simply prove that it is a valid subset by adding up all the elements in the set.This can be easily done in polynomial time.Hence the problem belongs to NP class.
To show that the problem is NP complete we reduce the 3-SAT problem to sub set problem by a polynomial reduction that is we show 3-SAT $\leq_p$ Subset Sum.
For a given formula ϕ of a 3-SAT instance with n variables and m clauses we create a sequence of 2m+2n numbers each with n+m digits.
We create 2n variables 2 for each (one for variable and another for its negation).The first n digits of these variables will have 1 at the variable number position and rest of all as zeroes. Example : the variables $x_3 and \neg x_3$ will have 1 at 3rd position and zeroes in all other n-1 places.In this way

the variable assignment is enforced because we can select the variables in such a way that it adds to a final number with all 1's in the first n digits.
The next m digits are assigned as following,It contains 1 in the clause number position and rest of all other places are zeroes.Suppose the clause $c_1 = (\neg x_1 \bigvee x_2 \bigvee x_3)$ then the variables $x_1, x_2, \neg x_1$ will have 1 at the first place and zeroes at all other places.

The goal is to select the numbers in such a way that when we add all the n digits it should have 1's at all places.In such case if we select a literal and its negation together then the corresponding digit in the output would be 2 which violates our assumption. We need to prove ϕ is satisfiable $\iff$ we have valid subset T

- Case 1: If ϕ is satisfiable,we choose the numbers in the truth assignment such that when we add them all the digits are 1's.

- If there exists a T ,we know that both number and its negation cannot be together in one clause from which we get a truth assignment and each clause would be satisfiable.

■