Advanced Algorithms and Complexity : Lecture 5 NP-Completeness of 3SAT, 0/1 Integer Programming and Independent Set

August 13, 2018

3SAT is NP-Complete: Proof of $3SAT \in NP$ is similar to the proof of $SAT \in NP$. To proove that $3SAT$ is NP-Hard, it is sufficient to show that $SAT \leq_p 3SAT$. We know that SAT is NP-Complete: $\forall L \in NP$, $L \leq_p SAT$. From transitivity of polynomial-time reductions:
$\forall L \in NP$, $L \leq_p 3SAT \implies 3SAT$ is NP-Hard. To prove that $SAT \leq_p 3SAT$, we have to convert a given CNF formula Φ into a 3CNF formula Φ_3 in polynomial-time so that $\Phi \in SAT \iff \phi_3 \in 3SAT$. For each clause of Φ, we introduce new variables and break the original clause into smaller clauses as follows: for an example, suppose the clause is $C = u_1 \vee \overline{u_2} \vee \overline{u_3} \vee u_4$. We add a new variable z to Φ_3 and replace C with the pair of clauses $C_1 = u_1 \vee \overline{u_2} \vee z$ and $C_2 = \overline{u_3} \vee u_4 \vee z$. C is replaced with $C_1 \wedge C_2$. Suppose C is True $\implies$ either $(u_1 \vee \overline{u_2})$ is true or $(\overline{u_3} \vee u_4)$ is true. For first case we can take $z = 0$, and for the second case we can take $z = 1$ so that $C_1 \wedge C_2$ is true for both cases.
$C =$ True $\implies C_1 \wedge C_2$ is True.
Suppose C is False $\implies$ both $(u_1 \vee \overline{u_2})$ and $(\overline{u_3} \vee u_4)$ are False $\implies$ for either value of z (0 or 1) at least one of C_1 and C_2 will be false $\implies C_1 \wedge C_2$ is false. This proves that $\Phi \in SAT \iff \Phi_3 \in 3SAT$ and so $SAT \leq_p 3SAT$.

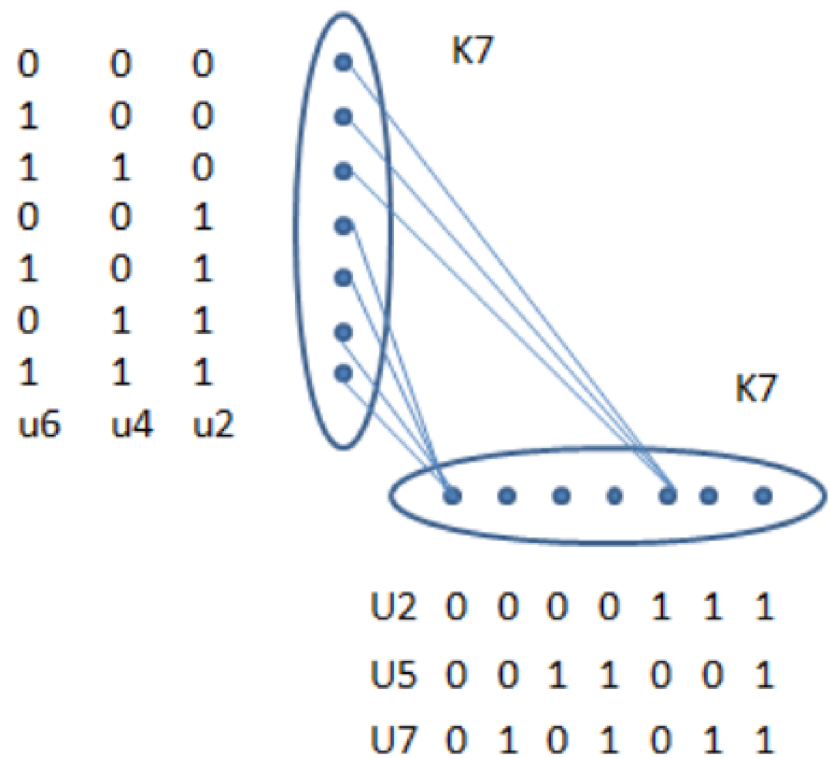
01 **Integer Programming:** Given a list of m linear inequalities with rational coefficients over n variables $u_1, ..., u_n$, find out if there is an assignment of 0's and 1's to $u_1, ...u_n$ satisfying all the inequalities.

$01IPROG$ **is NP-Complete:** To prove that $01IPROG \in NP$, the NTM N will non-deterministically guess an assignment (0 or 1) to the variables, and then it will evaluate and verify all the inequalities in polynomial-time.

$01IPROG$ **is NP-Hard:** We will prove this by proving that $SAT \leq_p 01IPROG$ as follows: for the given CNF formula $\Phi = C_1 \wedge C_2 \wedge ... \wedge C_m$, corresponding to each clause $C_i (1 \leq i \leq m)$ there will be an inequality whose solution will be in 0 or 1. For example, suppose the clause is $u_1 \vee \overline{u_2} \vee \overline{u_3}$, then the corresponding linear inequality will be $u_1 + (1 - u_2) + (1 - u_3) \geq 1$. Suppose $u_1 \vee \overline{u_2} \vee \overline{u_3}$ is True $\implies$ at least one of of $u_1, \overline{u_2}$ or $\overline{u_3}$ is True $\implies$ at least one of $u_1, (1 - u_2)$ or $(1 - u_3)$ is 1
$\implies u_1 + (1 - u_2) + (1 - u_3) \geq 1$.
Suppose $u_1 \vee \overline{u_2} \vee \overline{u_3}$ is False $\implies$ all of $u_1, \overline{u_2}$ and $\overline{u_3}$ are False $\implies$ all of $u_1, (1 - u_2)$ and $(1 - u_3)$ are 0
$\implies u_1 + (1 - u_2) + (1 - u_3) < 1$
$\implies \Phi \in SAT$ will imply that the equivalent 01 integer program $\in 01IPROG$. Also the transformation can be done in polynomial time $\implies SAT \leq_p 01IPROG \implies 01IPROG$ is NP-Hard.

INDSET $\in$ NP-Complete: We have proved that $INDSET \in NP$. We will prove that $INDSET \in$ NP-Hard by showing that $3SAT \leq_p INDSET$. On input an $m-$clause 3CNF formula $\Phi = C_1 \wedge C_2 \wedge ... \wedge C_m$, we will convert it into a $7m-$vertex graph G in polynomial-time such that Φ is satisfiable $\iff G$ has an independent set of size atleast m. We associate a cluster of 7 vertices in G with each clause of Φ. The vertices in a cluster associated with a clause correspond to the seven possible satisfying partial assignments out of $(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1)$. We put an edge between two vertices of G if they correspond to inconsistent partial assignments. Two partial assignments are consistent if they give the same value to all the variables they share. For example, the assignment $u_2 =$

$0, u_4 = 1, u_6 = 1$ is inconsistent with the assignment $u_2 = 1, u_5 = 0, u_7 = 1$ because they share a variable u_2 to which they give a different value. In addition, we put edges between every two vertices that are in the same cluster. Suppose we have $C_1 = \overline{u_2} \vee \overline{u_5} \vee u_7$ and $C_2 = u_2 \vee \overline{u_4} \vee u_6$:



The output graph has $7m$ vertices. It can have at most $\frac{7m(7m-1)}{2}$ edges so that the output graph can be produced in polynomial-time. Suppose Φ is satisfiable $\implies$ all clauses $C_1, ...C_m$ are true for some assignment of variables. Corresponding to this assignment of variables, in each clause we select the vertex corresponding to the partial assignment. There are m selected vertices. There cannot be an edge between these m selected vertices because the partial

assignments are consistent $\implies (G, m) \in INDSET$.
Now suppose that Φ is not satisfiable. Now if there exists an independent set of size m, then it will have one vertex from each cluster. We cannot have two vertices belonging to the same cluster because each cluster is a complete graph K_7. Now if the selected m vertices make an independent set $\implies$ the corresponding partial assignment of variables are consistent $\implies$ the corresponding assignment to Φ will make it True which is a contradiction to our assumption.
$\implies (G, m) \in INDSET$. From the above two results:
$\Phi \in 3SAT \iff (G, m) \in INDSET$
$3SAT \leq_p INDSET$
$INDSET$ is NP-Hard