**Comp 4450**

**Assignment 3**

The Australian National University

Submitted by: -

Prateek Arora

U6742441

**Task 2: Computational Complexity**

1. 3SAT is the decision problem of deciding whether a propositional formula in the form: (x\_11 v x\_12 v x\_13)∧(x\_21 v x\_22 v x\_23)∧…∧(x\_n1 v x\_n2 v x\_n3) is satisfiable (x ij is a positive or negative literal) [1]. Here the literals taken by the clauses are x, y and z. The left tunnels are considered as True whereas right one’s are considered as False. So, here from the game we can easily see that the first pig can only be killed by the x or y tunnel. If we have to kill the first pig, then either x has to be true or y has to be true due to which case 000, 001 are discarded. Let’s say that the x=0 and y= 1 which is done so that pigs 1,3 and 4 can be killed easily but when it comes to 2nd pig, that can only be killed when z=0 not z=1 in this case. So, 010 permutation is also rejected. Now, we are left with the 011, 100, 101, 110, 111. 011, 100, 101 and 110 kills all the pigs whereas when 111 permutation is applied then only 1st and 2nd pig will be killed and we will be left with the 3rd and 4th pig.

So, the instance here is

(x v y v z) ∧ (x v y v ~z) ∧ (x v ~y v z) ∧ (~x v ~y v ~z)

(~ represents the negation in the clause for that particular iteral)

1. Here as the condition say that the at least one literal should be false. So, here all literals can occur except for TTT. For this, the clauses should be in a way that the if all are false, then also pig should die. So, we have to make sure that for the possible ways in which all pigs are getting killed, we have to make in a way that if we put that in clause, then it not only satisfies the clause but kills the pigs as well. So, for this, the old clauses in part should be taken for this but complement of all clauses should be done because if the pig dies in the previous instance then, not only instance is true but at least one of the clauses also has all true in it. Putting a negation on the clauses will not just only kill the pig but also will satisfy the condition for at least literal to be false. So, the clauses are:

(~x v~ y v ~z) ∧ (~x v ~y v z) ∧ (~x v y v ~z) ∧ (x v y v z).

So, it’s working can be explained using example.

If we take that

X = True

Y = False

Z = False

So, here, in the clause, ~x would be false, ~y would be true, ~z would be true. So, in the clause the values should be

(F V T V T) ∧ (F V T V F) ∧ (F V F V T) ∧ (T V F V F)

This will satisfy the condition that at least one literal is true for this condition. So, for the condition for which pig should die are 011,100,101,110 for which these clauses evaluate to having at least one literal being false.

1. As given in the question, at least one literal should be false and at least one should be True. So, the clauses shouldn’t be satisfied when all are true or all are false. So, in this 000 and 111 will not violate the rule. So, in this one, same strategy should be used as used in the part b. So, here also negation is taken because for the condition, the pigs die contains at least one literal true and one literal as false (011,100,101,110). So, the clauses would be

(~x v~ y v ~z) ∧ (~x v ~y v z) ∧ (~x v y v ~z) ∧ (x v y v z).

So, in this if any of the value is taken:

Let’s say:

X = False

Y = True

Z= True

So, here, in the clause, ~x would be true, ~y would be true, ~z would be false. So, in the clause the values should be

(T V F V F) ∧ (TV F V T) ∧ (T V T V F) ∧ (F V T V T)

Which not only kills the pigs but also satisfies the condition that at least one literal should be true and one literal should be false.

1. As we can deduce from the question as well as the slides, that the angry bird game is NP hard. The relation between the algorithm can be solved by reduction. In this, if we say that

K reduces to c and c is NP hard. So, here, we say that the if we take instance I1 for k which it will convert into algorithm c with an instance I2. So, if K can be solved in polynomial time and the same algorithm can be used to solve the c, then it will be under the category of the c, which is NP hard. So, here, also, NOT – ALL – EQUAL – 3SAT problem is k and c is the angry bird problem. So, if the NOT – ALL – EQUAL – 3SAT reduces to angry bird problem, then we can say that if angry bird problem is solved then automatically NOT – ALL – EQUAL – 3SAT problem can be solved and vice-versa. So, they are under the same group, i.e., NP Hard. But NOT -ALL – EQUAL-3SAT has 3 variables and has 8 combinations and for n variables , there would 2^n combinations to check which makes the time complexity to 2^n which is polynomial time complexity algorithm due to which it NP and if algorithm is both NP Hard and NP , then the algorithm lies under NP Complete, i.e., complexity for NOT - ALL – EQUAL – 3SAT problem is **NP** **Complete**.

1. Condition for this part says that the pig should be killed when there is only one true. So, in this, left tunnel would be considered as false and right one’s are considered as true. So, the clause for this should be:

(x v y v z)

This because if we consider that then , the pigs will be killed by condition 001,010,100 etc. In this one, if we take x = false, y = false and x = true, then the pig will be killed in the game if left tunnel is considered as false and right as true.

The clause would be:

(F V F V T)

This also satisfies the condition for having only one literal true.

References:

[1] <https://wattlecourses.anu.edu.au/pluginfile.php/1914628/mod_resource/content/2/COMP2550-2019-trm2a.pdf>

(Wattle Lecture slides (Jochen Renz))