1 The homework is back!

Exercise 1: A many to one reduction
Assuming that BestClique is NP-hard, show that BestClique is \( \nabla \text{NP} \)-complete.

Solution:
BestClique is in \( \text{DP} \) because it asks whether \((G, k)\) is in Clique and \((G, k + 1)\) is not. In other words, we may write BestClique = \( L_1 \setminus L_2 \) for \( L_1 = \text{Clique} \) and \( L_2 = \{(G, k) \mid (G, k + 1) \in \text{Clique}\} \) and observe that \( L_2 \in \text{NP} \).

To show \( \text{DP} \)-hardness, we exhibit a reduction \( L \leq L \text{BestClique} \) for \( L = L_1 \setminus L_2 \) where \( L_1, L_2 \) are any two languages in \( \text{NP} \). The assumption that BestClique is NP-hard means that there exist a logspace reductions \( x \mapsto (G, k) \) such that \( x \in L_1 \iff k \) is the size of a largest clique in \( G \). It is also known that Clique is NP-hard, so we also have a logspace reductions \( x \mapsto (G', k') \) such that \( x \in L_2 \iff k' \) is the size of a clique in \( G' \). We modify the second reduction by adding a clique of size \( k' - 1 \) to \( H \), ensuring that the minimum size of a clique in \( H \) is \( k' \) if \( x \in L_2 \), \( k' - 1 \) otherwise. We also make sure that \( k \neq k' \) by adding some nodes and edges to one graph if necessary. If we now take the product graph \( G \times G' \), its largest clique has size \( k \times (k' - 1) \) iff \( (G, k) \in \text{BestClique} \) and \((G', k') \notin \text{Clique} \) iff \( x \in L_1 \) and \( x \notin L_2 \).

Exercise 2: Another many to one
Assuming \( \text{NP} \neq \text{coNP} \), show that BestClique \( \notin \text{NP} \).

Solution:
We reason contrapositively and prove that BestClique \( \in \text{NP} \) implies \( \text{NP} = \text{coNP} \). For this it is enough to show that Clique reduces to BestClique since Clique is well-known to be coNP-complete.

Now \((G, k) \in \text{Clique} \) iff the largest clique in \( G \) has size \( < k \), iff \( G \) extended by a clique of size \( k - 1 \) has its largest clique of size \( k - 1 \). This provides the required reduction.

Exercise 3: Not a many to one
Assuming \( \text{NP} \neq \text{coNP} \), show that BestClique \( \notin \text{coNP} \).

Solution:
Let us assume that BestClique \( \in \text{NP} \). We are given a graph \( G \) and a parameter \( k \). \((G, k) \in \text{Clique} \) means that there is no clique of size \( k \) or greater in \( G \). Thus, we have :

\[(G, k) \in \text{Clique} \iff \bigwedge_{k \leq k' \leq |G|} (G, k') \in \text{BestClique} \]

Thus, given a NTM \( M \) which accepts BestClique, we build \( M' \) which :
— takes input $\langle G, k \rangle$
— simulates (by guessing runs) $M$ on $\langle G, k' \rangle$, for $k \leq k' \leq |G|$.
— Accepts iff all runs where accepting.

With the previous equivalence, $M'$ recognizes Clique, and is a polynomial NTM. So, Clique $\in$ NP, thus, Clique $\in$ coNP, which implies NP = coNP as Clique is NP-complete.

Finally, if NP $\neq$ coNP, we have a contradiction and BestClique $\notin$ coNP.

## 2 The truth about the homework

**Definition 1** A language $L$ is in DP if and only if there are two language $L_1 \in$ NP and $L_2 \in$ coNP such that $L = L_1 \cap L_2$

The class DP (the D stands for differential) was introduced by Papadimitriou and Yannakis in [2]. Some problems were then proved to be complete for this class in [3], and the strictness properties where studied in [1].

**Definition 2** The TSP problem is defined as :

— INPUT : A distance matrix and an integer $B$.
— QUESTION : Is there a tour of length $B$ or less ?

The ExactTSP problem is defined as :

— INPUT : A distance matrix and an integer $B$.
— QUESTION : Is the length of the shortest tour $B$ ?

### Exercise 4: Money Money Money

1. Show that TSP is NP-complete.
   **Hint :** HAMILTONPATH is NP-complete
2. Why it is unlikely that ExactTSP $\in$ NP or ExactTSP $\in$ coNP ?
3. Prove that ExactTSP $\in$ DP.
4. Modify the reduction from 3SAT to HAMILTONPATH, so that, wheter or not the expressions are satisfiable, the produced graph will always contain a broken hamiltonian path, that is, two node-disjoint paths that cover all nodes.
   You do not need to know the reduction of 3SAT to HAMILTONPATH, you just need to know that it produces a graph with a "starting" node and an "ending" node, and if there is only one not satisfied clause, then there is a broken hamiltonian path in the graph.
5. Prove that ExactTSP is DP-complete.

### Solution:

1. It is in NP, as one can guess the tour and computes its length. Given a graph $G$ with $n$ nodes, we construct a matrix $M$ of size $n \times n$ (there are n cities), and the distance between two cities $i$ and $j$ is 1 if they are connected in $G$ 2 otherwise. There is a tour of length $n + 1$ or less in $M$ if and only if $G$ has an hamiltonian path.
2. It seems hard to certify shortly that the optimum cost is indeed $B$, because you need to certify that there is no shortest tour. To certify that the optimum cost is not $B$, you can provide a tour of length shorter than $B$, but if it does not exists, you need to certify that there is no tour of length $B$.
3. compTSP asks whether the optimum cost is at least $B$, and is trivially in coNP.
   ExactTSP = TSP $\cap$ compTSP.
4. Before the reduction, we simply have to add a new litteral $z$ to every clause, and also add the clause $\neg z$. We then do the classical reductions from SAT to 3SAT, and then to HAMILTONPATH.
5. We are given $(\phi, \phi')$ an instance of SATUNSAT, with the previous reduction we obtain two graphs $G$ and $G'$, both with a starting node and an ending node. We then produce a graph $R$ by joining $G$ and $G'$ in a cycle, connecting the starting node of one to the ending node of the other. Let $n$ be the number of nodes in $R$. Then, for nodes $i$ and $j$, if there is an edge between them, the distance is 1, else if they are both from $G$, the distance is 2, else, it is 4. We then have several cases:

- Both formulas are satisfiable. Then, we have an Hamiltonian path in $R$, and the optimal cost is $n$.
- Both formulas are satisfiable. Then, the optimal tour combine two broken Hamiltonian path, a non edge of $G$ (distance 2) and a non edge of $G'$ (distance 4) needs to be used, the optimum cost is then $n - 2 + 2 + 4 = n + 4$.
- $\phi$ is sat and $\phi'$ is not. Then, we combine the Hamiltonian path of $G$ with the broken one of $G'$ and add a non edge in $G'$, the optimal cost is $n - 1 + 4 = n + 3$.
- $\phi'$ is sat and $\phi$ is not, then the optimal cost is $n - 1 + 2 = n + 1$.

Finally,

$$(\phi, \phi') \in \text{SATUNSAT} \iff (R, n + 3) \in \text{ExactTSP}$$

Références

