We show more reductions from 3SAT to several problems.

# 1 Variations of Independent Set

## 1.1 Maximum Clique

Given a (undirected non-weighted) graph $G = (V, E)$, a *clique* $K$ is a set of vertices $K \subseteq V$ such that *any two* vertices in $K$ are adjacent. In the MAXIMUM CLIQUE problem, given a graph $G$ we want to find a largest clique.

In the decision version, given $G$ and a parameter $k$, we want to know whether or not $G$ contains a clique of size at least $k$. It should be clear that the problem is in NP.

We can prove that Maximum Clique is NP-hard by reduction from Maximum Independent Set. Take a graph $G$ and a parameter $k$, and consider the graph $G'$, such that two vertices in $G'$ are connected by an edge if and only if they are not connected by an edge in $G$. We can observe that every independent set in $G$ is a clique in $G'$, and every clique in $G'$ is an independent set in $G$. Therefore, $G$ has an independent set of size at least $k$ if and only if $G'$ has a clique of size at least $k$.

## 1.2 Minimum Vertex Cover

Given a (undirected non-weighted) graph $G = (V, E)$, a *vertex cover* $C$ is a set of vertices $C \subseteq V$ such that for every edge $(u, v) \in E$, either $u \in C$ or $v \in C$ (or, possibly, both). In the MINIMUM VERTEX COVER problem, given a graph $G$ we want to find a smallest vertex cover.

In the decision version, given $G$ and a parameter $k$, we want to know whether or not $G$ contains a vertex cover of size at most $k$. It should be clear that the problem is in NP.

We can prove that Minimum Vertex Cover is NP-hard by reduction from Maximum Independent Set. The reduction is based on the following observation:

**Lemma 1** *If $I$ is an independent set in a graph $G = (V, E)$, then the set of vertices $C = V - I$ that are not in $I$ is a vertex cover in $G$. Furthermore, if $C$ is a vertex cover in $G$, then $I = V - C$ is an independent set in $G$.*

Proof: Suppose $C$ is not a vertex cover: then there is some edge $(u, v)$ neither of whose endpoints is in $C$. This means both the endpoints are in $I$ and so $I$ is not an independent set, which is a contradiction. For the "furthermore" part, suppose $I$ is not an independent set: then there is some edge $(u, v) \in E$ such that $u \in I$ and $v \in I$, but then we have an edge in $E$ neither of whose endpoints are in $C$, and so $C$ is not a vertex cover, which is a contradiction. $\square$

Now the reduction is very easy: starting from an instance $(G, k)$ of Maximum Independent set we produce an instance $(G, n - k)$ of Minimum Vertex Cover.

# 2 Set Cover

In the Set Cover problem, we are given a collection $S_1, \ldots, S_m$ of subsets of a set $X$, such that $\bigcup_{i=1}^m S_i = X$, and we want to find a minimal subcollection of sets whose union is still $X$. To formulate it as an NP search problem, we say that given sets $S_1, \ldots, S_m$ and an integer $k$ we ask whether there is a subset $I \subseteq \{1, \ldots, m\}$ such that $|I| = k$ and $\bigcup_{i \in I} S_i = X$.

We reduce Vertex Cover to Set Cover. Given a graph $G = (V, E)$ with $n$ vertices and $m$ edges, we create a set system where $X = E$, and there is a set $S_v$ for every vertex $v \in V$ of the graph, defined as

$$S_v := \{(u, v) : u \in V mbox and (u, v) \in E\}$$

that is, as the set of edges that touch the vertex $v$. It is easy to see that there is a vertex cover with $k$ vertices in $G$ if and only if there is a set cover with $k$ sets in the instance of set cover described above.

# 3 Partition

The **Partition** problem is defined as follows:

- Given a sequence of integers $a_1, \ldots, a_n$.
- Determine whether there is a partition of the integers into two subsets such the sum of the elements in one subset is equal to the sum of the elements in the other.

  Formally, determine whether there exists $I \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in I} a_i = (\sum_{i=1}^n a_i)/2$.

Clearly, Partition is a special case of Subset Sum. We will prove that Partition is is NP-hard by reduction from Subset Sum.[1]

Given an instance of Subset Sum we have to construct an instance of Partition. Let the instance of Subset Sum have items of size $a_1, \ldots, a_n$ and a parameter $k$, and let $A = \sum_{i=1}^{n} a_i$.

Consider the instance of Partition $a_1, \ldots, a_n, b, c$ where $b = 2A - k$ and $c = A + k$.

Then the total size of the items of the Partition instance is $4A$ and we are looking for the existence of a subset of $a_1, \ldots, a_n, b, c$ that sums to $2A$.

It is easy to prove that the partition exists if and only if there exists $I \subseteq \{1, \ldots, n\}$ such that $\sum_i a_i = k$.

# 4 Bin Packing

The **Bin Packing** problem is one of the most studied optimization problems in Computer Science and Operation Research, possibly the second most studied after TSP. It is defined as follows:

- Given items of size $a_1, \ldots, a_n$, and given unlimited supply of bins of size $B$, we want to pack the items into the bins so as to use the minimum possible number of bins.

You can think of bins/items as being CDs and MP3 files; breaks and commercials; bandwidth and packets, and so on.

The decision version of the problem is:

- Given items of size $a_1, \ldots, a_n$, given bin size $B$, and parameter $k$,
- Determine whether it is possible to pack all the items in $k$ bins of size $B$.

Clearly the problem is in NP. We prove that it is NP-hard by reduction from Partition.

Given items of size $a_1, \ldots, a_n$, make an instance of Bin Packing with items of the same size and bins of size $(\sum_i a_i)/2$. Let $k = 2$.

There is a solution for Bin Packing that uses 2 bins if and only if there is a solution for the Partition problem.

[1] The reduction goes in the non-trivial direction!