Reduction of IND to Subset_Sum

We define an instance of IND, the independent set problem, to be a string of the form $\langle G\rangle\langle k\rangle$ where G is a graph and k a positive integer. That string is a member of the language IND if there is some set I vertices of G such that $|I| = k$ and no two members of I form an edge of G, a *k-independent set* of G.

We define an instance of *Subset_Sum*, the *subset sum problem* to be a string $\langle X\rangle\langle K\rangle$ where X is a list of positive numbers and K is a number. [1] We say $\langle X\rangle\langle K\rangle \in$Subset_Sum if there is some sublist of X whose sum is K. We prove that Subset_Sum is $\mathcal{NP}$, using the certificate method. The sublist whose total equals K is the certificate, which can be (trivially) verified in polynomial time.

We now define a $\mathcal{P}$-TIME reduction R of IND to Subset_Sum, where IND is the independent set problem. We assume that all of our languages (problems) are over an alphabet Σ. Without loss of generality, Σ is the binary alphabet. Each reduction must be a function

$$R : \Sigma^* \to \Sigma^*$$

When we define $R(w)$ below, we will assume that w is an instance of the independent set problem. If w is any other string, we define $R(w) = \lambda$, the empty string. No further discussion of this case is necessary.

Let $\langle G\rangle\langle k\rangle$ be an instance of IND, where $G = (V, E)$. Write $V = \{v_1, v_2, \ldots v_n\}$, the vertices of G, and and $E = \{e_1, e_2, \ldots e_m\}$, the edges of G. We say e_j *meets* v_i, and write $e_j \perp v_i$, if v_i is one of the two end points of e_j.

We now define $R(\langle G\rangle\langle k\rangle) = \langle X\rangle\langle K\rangle$, an instance of the subset sum problem. Define $weight(v_i) = 10^{m+1} + \sum_{j\,:\,e_j\perp v_i} 10^j$ and $weight(e_j) = 10^j$. Let $X = weight(v_1)\ldots weight(v_n), weight(e_1)\ldots weight(e_m)$, and let $K = k \cdot 10^{m+1} + \sum_{j=1}^{m} 10^j$.

By the following two lemmas, R is a reduction of IND to Subset_Sum.

Lemma 1 *If G has an independent set of size k, then $\langle X\rangle\langle K\rangle \in$ Subset_Sum.*

Proof:

Let $\mathcal{I}$ be a set of k independent vertices of G. Let $\mathcal{J}$ be the set of edges which do not meet any of the vertices in $\mathcal{I}$. Let S be the sum of the weights of vertices in $\mathcal{I}$ and the edges in $\mathcal{J}$, that is, S is the sum of a subsequence of X. Write $S = \sum_| \ell = 1^{m+1} \alpha_\ell 10^\ell$.

Claim: $S = K$.

Proof of Claim: Since I has cardinality k, we have $\ell_{m+1} = k$ since I has cardinality k. For any $1 \leq \ell \leq m$, If e_ℓ does not meet any member of I, y_ℓ contributes 10^ℓ to S, while if e_ℓ meets v_i, then x_i contributes 10^ℓ to S. Since I is independent, e_ℓ does not meet any other vertex, there is no additional contribution of 10^ℓ to S. That is, $\alpha_\ell = 1$ in either case. Thus, $S = K$. ∎

Lemma 2 *If $\langle X\rangle\langle K\rangle \in$ Subset_Sum, then G has an independent set of size k.*

[1] Note that the *size* of an instance $\langle X\rangle\langle K\rangle$ is the number of bits in that string, not the number of numbers encoded. Similarly, the size of an instance of IND is the number of bits in the string $\langle G\rangle\langle k\rangle$.

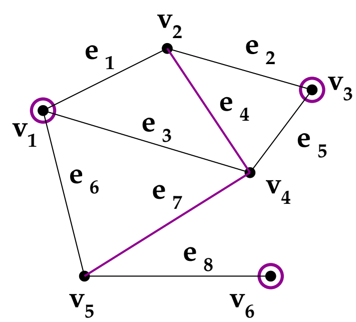
Proof: Suppose K is the sum of a sublist of X. Then there are sets $I \subseteq V$ and $J \subseteq E$ such that K is the sum of the weights of a $I \subseteq V$ and $J \subseteq E$. Write $K = \sum_{|} \ell = 1^{m+1} \alpha_\ell 10^\ell$. Since $\alpha_{m+1} = k$, the cardinality of I is k. No two members of J span an edge, since otherwise $\alpha_\ell = 2$ for some ℓ. Thus, I is a k-independent set of G. ∎

Immdiately from Lemmas 1 and 2:

Theorem 1 *If* IND *is* $\mathcal{NP}$–COMPLETE *then* Subset Sum *is* $\mathcal{NP}$*-complete.*

Example

Let G be the graph illustrated below, where $n = 6$ and $m = 8$. Let $k = 3$. The set $\mathcal{I} = \{v_1, v_3, v_6\}$ is an independent set of vertices of G of size k. In our reduction, $\mathcal{J} = \{e_4, e_7\}$. We write k and all the weights in base 10. The first array shows the weights of all items, while the second array shows that the weights of the selected items sum to k.



$$
\begin{array}{lcllllllllll}
K & = & 3 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
y_1 & = & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
y_2 & = & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
y_3 & = & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
y_4 & = & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
y_5 & = & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
y_6 & = & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
z_1 & = & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
z_2 & = & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
z_3 & = & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
z_4 & = & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
z_5 & = & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
z_6 & = & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
z_7 & = & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
z_8 & = & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

$$
\begin{array}{lcllllllllll}
y_1 & = & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
y_3 & = & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
y_6 & = & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
z_4 & = & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
z_7 & = & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
K & = & 3 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}
$$