CMSC 641, Design and Analysis of Algorithms, Spring 2010

Sandipan Dey,
Homework Assignment - 7

30/30

April 6, 2010

Half Clique is NP-Complete

✓ 10

Construction

We reduce known problem Clique $\in NPC$ to Half-Clique problem. Let $G(V, E)$ be the graph with $|V| = n$ nodes. Also, let (G, k) be an instance of Clique. We transform it to an instance of Half-Clique $(G\prime, k)$ using the following construction:

- If $k \geq \frac{n}{2}$, add $2k - n$ vertices to G, without adding any edges, to obtain graph $G\prime(V\prime, E)$, with $|V\prime| = 2k$. ✓
- If $k < \frac{n}{2}$, add $n - 2k$ vertices and completely connect them to G and each other, to obtain graph $G\prime(V\prime, E\prime)$, with $|V\prime| = 2(n - k)$. ✓
- Run Half-Clique on this altered graph, $G\prime$. ✓

Reduction Proof

We must show that this transformation is a reduction, i.e., we need to show that G has a clique of size k iff $G\prime$ has a Half-Clique.

- G has a clique of size k.
  - When $k \geq \frac{n}{2}$, since the construction of $G\prime$ does not destroy any edges, $G\prime$ still has a k clique. But, $G\prime$ has size $2k \Rightarrow G\prime$ has a Half Clique, by definition. ✓
  - When $k < \frac{n}{2}$ construction creates another clique of size $n-2k$ vertices in $G\prime$ and every vertex of that clique is connected to the old clique from $G \Rightarrow G\prime$ now has a clique of size $k + n - 2k = n - k$. But, $G\prime$ has $2(n - k)$ vertices, by construction $\Rightarrow G\prime$ has a Half Clique, by definition. ✓
- $G\prime$ has a Half-Clique.

- By construction $|V\prime| = 2k$ or $|V\prime| = 2(n-k)$.
- $|V\prime| = 2k \Rightarrow G\prime$ has a clique of at least size k and since the construction of $G\prime$ does not destroy any edges, G must have the same k clique.
- When $|V\prime| = 2(n-k) \Rightarrow G\prime$ has a clique of at least size $n-k$. Going back to G, the construction only removes all the edges from the $n-2k$ vertices in $G\prime$ and hence even in the worst case G is going to have $n-k-(n-2k) = k$. cliques.

- The reduction is a polynomial time reduction ($O(n^2)$).

Dominating Set

Construction

?

We reduce known problem Vertex Cover $\in NPC$ to Half-Clique problem. Given an undirected graph $G(V, E)$ and a number k, let's construct a graph $G\prime$ so that G has a vertex cover of size at most k iff $G\prime$ has a dominating set of size at most k.

$G\prime$ will have a vertex for each vertex of G, except for any isolated vertices (vertices of degree 0), and a new vertex v_e for each edge e of G. Each edge $e = (u, v) \in E(G)$ is replaced by a triangle of edges in $G\prime : (u, v), (u, v_e)$, and (v, v_e).

10

Reduction Proof

We must show that this transformation is a reduction, i.e., G has a vertex cover of size k iff $G\prime$ has a dominating set of size k.

- X is a vertex cover of G of size k.
  - Every edge of G has a vertex in X incident to it, and so every triangle of vertices in $G\prime$ has at least one member of X in it.
  - Every vertex of $G\prime$ is either in X or adjacent to a vertex in X (Since every vertex of $G\prime$ is in one of the triangles).
  - If G has a vertex cover of size k, then $G\prime$ has a dominating set of size k.
- Y is a dominating set of $G\prime$ of size k.
  - If any vertex in Y is an edge-vertex v_e (added by construction) rather than a vertex of G, we can replace it by one of the G-vertices for the edge e's endpoints and it will still be a dominating set. (v_e only dominated 3 vertices, those in its triangle, and either of the other 2 vertices in this triangle also dominate these 3 vertices).

- Once we replace all edge-vertices in Y by G-vertices, the new Y forms a vertex cover of G, since every edge of G must have at least one of its endpoints in Y (Y being a dominating set of $G\prime$).

- The reduction is a polynomial time reduction ($O(|E|)$).

Clique and Unary Counter

V

10

Part (a)

By condition, if $u, v \in V$ and $u \neq v$ then we have,

$$\begin{aligned} &(\forall u)(\forall v)\,(x_{uv} \leftrightarrow (u,v) \in E) \\ \equiv &\left(\bigwedge_{(u,v)\in E} x_{uv}\right) \wedge \left(\bigwedge_{(u,v)\notin E} \neg x_{uv}\right) \end{aligned}$$

which is conjunction of 2 CNF forms, hence in CNF form.

Hence, we can add clauses $D_1 = \bigwedge_{(u,v)\in E} x_{uv}$, $D_2 = \bigwedge_{(u,v)\notin E} \neg x_{uv}$.

Part (b)

By condition, if $u, v \in V$ and $u \neq v$ then we have,

$$\begin{aligned} &(\forall u)(\forall v)\,((x_u \wedge x_v) \rightarrow x_{uv}) \\ \equiv\ &(\forall u)(\forall v)\,((x_u \wedge x_v) \rightarrow x_{uv}) \\ \equiv\ &(\forall u)(\forall v)\,(\neg x_u \vee \neg x_v \vee x_{uv}) \end{aligned}$$

which is in CNF form.

Hence, we can add clause $E = \bigwedge_{\substack{u,v\in V\\ u\neq v}} (\neg x_u \vee \neg x_v \vee x_{uv})$.

Part (c)

$n(n+1)$ variables to represent the value of the n-bit unary counter through the $n+1$ stages

Let's define the variable

b_i^s = value of the i^{th} bit of the unary counter at stage s.

Now, the unary counter is an n bit counter $\Rightarrow i = 1 \ldots n$ and there are $n+1$ stages of the counter $\Rightarrow s = 0, \ldots n$.

Hence, the total number of variables used to to represent the value of the n-bit unary counter through the $n+1$ stages $= n(n+1)$.

enforce that the counter starts with value 0

By clause $C_1 = \bigwedge_{i=1}^{n} \neg b_i^0$.

enforce that the counter ends with value k

By clause $C_2 = \bigwedge_{i=1}^{k} b_i^n \cdot \bigwedge_{i=k+1}^{n} \neg b_i^n$.

enforce that the counter is incremented (or not) correctly in each stage

$$\begin{cases} b_{s-1}^{i} \Rightarrow b_s^{i}, \ \forall s,\ 1 \le s \le n,\ \forall i,\ 1 \le i \le n \\ \overline{b_{s-1}^{i}} \wedge b_{s-1}^{i-1} \wedge x_i \Leftrightarrow b_s^{i} \end{cases}$$

We notice that

$(b_i^k \wedge x_i) \rightarrow b_{i+1}^k$.

Hence, the corresponding clause will be

$C_3 = \bigwedge_{i=1}^{n} \bigwedge_{k=0}^{n} (\neg b_i^k \vee \neg x_i \vee b_{i+1}^k)$. (By D'Morgan).

Part (d)

Let's construct our boolean formula $B = C_1 \wedge C_2 \wedge C_3 \wedge D_1 \wedge D_2 \wedge E$.

We notice that B is in CNF form and none of the clauses have more than 3 literals, hence B is in 3-CNF form.

B is satisfiable $\Rightarrow$

Any satisfying assignment τ for $C_1 \wedge C_2 \wedge C_3$ picks a set of k vertices (guaranteed by the unary counter), namely selected those u s.t. $\tau(x_u) = true$.

If τ satisfied $D_1 \wedge D_2 \wedge E$ too, then those k vertices must form a clique (since $x_u \wedge x_v \rightarrow x_{uv} \rightarrow (u, v) \in E \Rightarrow$ Graph G has a k clique).

Graph G has a k-clique $\Rightarrow$

If $u_1, u_2, \ldots, u_k$ is a k-clique in G, assign $\tau(x_{u_i}) = true$, $\forall i = 1 \ldots k$ and set $\tau(y) = false$ for all other variables $\Rightarrow D_1, D_2, E$ are true (by clique property).

Now, selecting these k variables at the different stages of the unary counter, C_1, C_2, C_3 are also satisfied $\Rightarrow$ this truth assignment satisfies B.