CPTH Final Assignment

Kishlaya Jaiswal *(MCS201909)*
Satya Prakash Nayak *(MCS201914)*
Shrisha Rao *(MCS201916)*

May 16, 2020

Solution 1 (Problem 2.19)

We first show that QUADEQ $\in$ NP. For any set of quadratic equations over 0/1 variables $u_1, u_2, \ldots, u_n$, we can non-deterministically guess the values of each u_i and check if the set of equations are satisfiable. Clearly, we can do this in polynomial time, hence QUADEQ $\in$ NP.

Next we show that QUADEQ is NP $-$ complete by showing $3-\text{SAT} \leq_P$ QUADEQ. Given a $3-\text{SAT}$ formula $\phi = C_1 \vee C_2 \vee \cdots C_m$ over variables $x_1, x_2, \ldots, x_n$ (and their negations). We will reduce it to a set of quadratic equations. First let E_0 be the equation $y_0 y_0 = 1$. Now for every clause $C_j = v_1 \vee v_2 \vee v_3$, let E_j be the quadratic equation $a_j P(v_1) + b_j P(v_2) + c_j P(v_3) = 1$, where $P(x_i) = y_i$ and $P(\bar{x_i}) = y_0 + y_i$ for any variable x_i. For example, if $C_j = x_1 \vee \bar{x_2} \vee x_3$ then E_j is the equation $a_j y_1 + b_j(y_0 + y_2) + c_j y_3 = 1$.

Now we claim that the set of equations $E = \{E_0, E_1, \ldots, E_m\}$ is satisfiable iff ϕ is satisfiable. First note that equation E_0 is satisfiable iff $y_0 = 1$, and then taking $y_0 = 1$ gives us $P(\bar{x_i}) = 1 + y_i = \bar{y_i}$, hence each equation $a_j P(v_1) + b_j P(v_2) + c_j P(v_3) = 1$ is satisfiable iff atleast one of $P(v_i)$ is 1 (then we can find suitable values for a_j, b_j, c_j). Now for a satisfiable assignment of ϕ, set $y_0 = 1$ and $y_i = x_i$ for all $1 \leq i \leq n$. Then for any clause $C_j = v_1 \vee v_2 \vee v_3$, atleast one of the v_i's is 1, so the corresponding $P(v_i)$ is also 1, hence E_j is satisfied. Therefore, the set of equations in E are satisfied. Similarly we can find a satisfying assignment for ϕ from a satisfying assignment for the equations in E by setting the values $x_i = y_i$ for all $1 \leq i \leq n$.

Solution 2 (Problem 3.6)

(a)

In the worst-case scenario, while computing $H(n)$, we might have to go over each $i < \log\log n$. But each M_i takes atmost $\sum_{k=1}^{\log n}(ik^i)2^k$ time (to check for each $x \in \{0,1\}^*$ such that $|x| \leq \log n$). Thus the total time taken would be:

$$\begin{aligned}
\sum_{i=1}^{\log\log n}\sum_{k=1}^{\log n} i2^k k^i &\leq \sum_{i=1}^{\log\log n}\sum_{k=1}^{\log n} \log\log n \cdot 2^{\log n} \cdot (\log n)^{\log\log n} && \text{(taking } i = \log\log n \text{ and } k = \log n\text{)} \\
&= \sum_{i=1}^{\log\log n}\sum_{k=1}^{\log n} \log\log n \cdot n \cdot 2^{(\log\log n)^2} \\
&\leq \log\log n \cdot \log n \cdot \left(\log\log n \cdot n \cdot 2^{(\log\log n)^2}\right) \\
&= O\left((\log n)^2 \cdot n \cdot 2^{\log n}\right) && \text{(since } (\log\log n)^2 = O(\log n)\text{)} \\
&= O(n^3)
\end{aligned}$$

(b)

Suppose SAT_H is $NP-complete$. Then we can find a poly-time reduction f from SAT to SAT_H. Let the running time of f be n^{i_0}.

Since $H(n) \xrightarrow{n\to\infty} \infty$, we find N such that $\forall n \geq N$, $H(n) > i_0$ and hence $n^{H(n)} > n^{i_0}$.

Given a formula φ of length $n \geq N$, $f(\varphi) = \psi 01^{m^{H(m)}}$ where $m = |\psi|$, that is f maps length n formula to a length $m + m^{H(m)}$ string. If $m \geq n$ then $m^{H(m)} \geq n^{H(m)} > n^{i_0}$ but since f runs in time n^{i_0}, the output can be no longer than n^{i_0}. Therefore, $m < n$.

This means, we can apply f to reduce our input SAT instance to a smaller size SAT instance (alongwith some padding), as follows:

Algorithm 1: $\mathscr{A}$ - check satisfiability of a given boolean formula

On input φ
if $|\varphi| < N$ **then**
  check by brute-force
  return **True** if there is some satisfying assignment
  return **False** otherwise
else
  compute $f(\varphi) = \psi 01^{m^{H(m)}}$ where $m = |\psi|$
  recursively call $\mathscr{A}$ for the smaller instance ψ
  return output of $\mathscr{A}(\psi)$
end

Running Time: Let $C = 2^{2N}$ (time required for the brute-force case). In worst-case scenario $|\psi| = |\varphi| - 1$ and so it takes $(n - N)$ iterations before reaching the base case and each such iteration requires atmost n^{i_0} time for finding ψ. Hence $\mathscr{A}$ runs in atmost $O(n^{i_0+1})$ time.

Correctness: follows from the fact that

$$\varphi \in SAT \iff \psi 01^{m^{H(m)}} \in SAT_H \iff \psi \in SAT$$

Solution 3 (Problem 5.10)

First we show that for any language A, we have

$$\Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq \mathrm{P}^{\mathrm{A}} \implies \Pi_{\mathrm{i}}^{\mathrm{p,A}} \subseteq \mathrm{P}^{\mathrm{A}}.$$

Assume $\Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq \mathrm{P}^{\mathrm{A}}$. Let $L \in \Pi_{\mathrm{i}}^{\mathrm{p,A}}$, then by definition, there is a polynomial-time Turing machine M and a polynomial q such that

$$x \in L \iff \forall u_1 \in \{0,1\}^{q(|x|)} \exists u_2 \in \{0,1\}^{q(|x|)} \cdots u_i \in \{0,1\}^{q(|x|)} M^A(x, u_1, \ldots, u_i) = 1$$

Complementing both sides gives

$$x \in L^c \iff \exists u_1 \in \{0,1\}^{q(|x|)} \forall u_2 \in \{0,1\}^{q(|x|)} \cdots u_i \in \{0,1\}^{q(|x|)} M^A(x, u_1, \ldots, u_i) = 0$$

Now consider another polynomial-time Turing machine M_c (with oracle access to A) that does the following on input x: Run M^A on x and accept if M^A rejects else reject. Then we have

$$x \in L^c \iff \exists u_1 \in \{0,1\}^{q(|x|)} \forall u_2 \in \{0,1\}^{q(|x|)} \cdots u_i \in \{0,1\}^{q(|x|)} M_c^A(x, u_1, \ldots, u_i) = 1$$

Hence, by definition $L^c \in \Sigma_{\mathrm{i}}^{\mathrm{p,A}} \implies L^c \in P^A$. Then there is a polynomial-time Turing machine N with oracle access to A such that $L(N^A) = L^c$. Now consider another polynomial-time Turing machine N_c (with oracle access to A) that does the following on input x: Run N^A on x and accept if N^A rejects else reject. Then we have $L(N_c^A) = L$ and since N_c is a polynomial time machine, we have $L \in P^A$. Therefore, $\Pi_{\mathrm{i}}^{\mathrm{p,A}} \subseteq \mathrm{P}^{\mathrm{A}}$.

Now suppose $\mathrm{P}^{\mathrm{A}} = \mathrm{NP}^{\mathrm{A}}$ for some language A. We prove by induction on i that $\Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq P^A$. Clearly this is true for $i = 1$ by assumption, since $\Sigma_{\mathrm{i}}^{\mathrm{p,A}} = NP^A \subseteq P^A$. We assume this is true for $i - 1$ and prove that $\Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq P^A$.

Let $L \in \Sigma_{\mathrm{i}}^{\mathrm{p,A}}$, then by definition, there is a polynomial-time Turing machine M and a polynomial q such that

$$x \in L \iff \exists u_1 \in \{0,1\}^{q(|x|)} \forall u_2 \in \{0,1\}^{q(|x|)} \cdots u_i \in \{0,1\}^{q(|x|)} M^A(x, u_1, \ldots, u_i) = 1 \tag{1}$$

Define another language L' as follows:

$$\langle x, u_1 \rangle \in L' \iff \forall u_2 \in \{0,1\}^{q(|x|)} \exists u_3 \in \{0,1\}^{q(|x|)} \cdots u_i \in \{0,1\}^{q(|x|)} M^A(x, u_1, \ldots, u_i) = 1$$

Clearly, $L' \in \Pi_{\mathrm{i}-1}^{\mathrm{p,A}}$. By our assumption, $\Sigma_{\mathrm{i}-1}^{\mathrm{p,A}} \subseteq P^A$ which implies $\Pi_{\mathrm{i}-1}^{\mathrm{p,A}} \subseteq P^A$, hence $L' \in P^A$. Then there is polynomial-time Turing machine N such that $L(N^A) = L'$, which means

$$\langle x, u_1 \rangle \in L' \iff N^A(x, u_1) = 1$$

Then (1) gives us

$$x \in L \iff \exists u_1 \in \{0,1\}^{q(|x|)} N^A(x, u_1) = 1$$

Then this means $L \in \mathrm{NP}^{\mathrm{A}}$ which implies $L \in \mathrm{P}^{\mathrm{A}}$ since $\mathrm{P}^{\mathrm{A}} = \mathrm{NP}^{\mathrm{A}}$. Therefore, by induction $\Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq P^A$ for all i. Then we have

$$\mathrm{PH}^{\mathrm{A}} = \bigcup_i \Sigma_{\mathrm{i}}^{\mathrm{p,A}} \subseteq P^A.$$

Solution 4 (Problem 7.11)

(a)

Let X_0 denote the distribution p which we start with. And X_k denote the distribution after k-random steps.

Given that G is a d-regular graph, we want to show that uniform distribution is stationary. So let the distribution of X_0 be uniform, that is $P(X_0 = u) = \frac{1}{n}$, then, for any vertex v:

$$\begin{aligned}
P(X_1 = v) &= \sum_u P(X_1 = v \mid X_0 = u) P(X_0 = u) \\
&= \left(\sum_{u : u \in N(v)} \frac{1}{d+1} \times \frac{1}{n} \right) + P(X_1 = v \mid X_0 = v) P(X_0 = v) \\
&= \left(\frac{1}{d+1} \times \frac{1}{n} \sum_{u : u \in N(v)} 1 \right) + \left(\frac{1}{d+1} \times \frac{1}{n} \right) \\
&= \left(\frac{1}{d+1} \times \frac{1}{n} \times d \right) + \left(\frac{1}{d+1} \times \frac{1}{n} \right) \\
&= \frac{1}{n}
\end{aligned}$$

(b)

Let π be the uniform distribution over the vertices in G, that is also stationary, i.e. $\pi^k = \pi$. We will show that for any distribution $\mathbf{p}$, as $k \to \infty$ we have $\mathbf{p}^k \to \pi$, which then implies $\Delta(\mathbf{p}^k) \to 0$, hence there exists k such that $\Delta(\mathbf{p}^k) \le (1 - n^{-10n}) \Delta(\mathbf{p})$.

Let X_0 be a vertex of G picked by a distribution $\mathbf{p}$ and let Y_0 be a vertex picked by π. And X_k and Y_k be vertices we reached by taking k-random steps from X_0 and Y_0 respectively. Let $T = \inf\{k : X_k = Y_k\}$.

It is easy to see that $T < \infty$, since G is connected, there exists some l such that $p^l_{uv} > 0$ for all vertices u, v, where p^k_{uv} is the probability of reaching the vertex v from u by k-random steps. Hence, the probability of $Y_k = X_k$ is $\sum_v p^k_{X_0 v} p^k_{Y_0 v} > 0$ for all $k \ge l$. Hence, expected value of minimum k for which $X_k = Y_k$ is finite, hence $T < \infty$.

Now for any $a \le b$, we have

$$\begin{aligned}
P(X_b = u, T = a) &= \sum_{v \in V(G)} P(X_b = u | X_a = v) P(X_a = v, T = a) \\
&= \sum_{v \in V(G)} P(Y_b = u | y_a = v) P(Y_a = v, T = a) \\
&= P(Y_b = u, T = a).
\end{aligned}$$

Then we have

$$\begin{aligned}
P(X_b = u) &= \sum_{a=0}^{\infty} P(X_b = u, T = a) \\
&= \sum_{a=0}^{b} P(Y_b = u, T = a) + P(X_b = u, T > b)
\end{aligned}$$

Now we have

$$\begin{aligned}\sum_{u\in V(G)} |P(X_b = u) - P(Y_b = u)| &= \sum_{u\in V(G)} |P(X_b = u, T > b) - P(Y_b = u, T > b)| \\ &\le \sum_{u\in V(G)} P(X_b = u, T > b) + \sum_{u\in V(G)} P(Y_b = u, T > b) \\ &= 2P(T > b)\end{aligned}$$

Since $P(T > b) \to 0$ as $b \to \infty$, for any vertex u, we have $P(X_b = u) \to P(Y_b = u) \implies \mathbf{p}^k \to \pi$.

(a) Observe that if p is stationary then $\Delta(\mathbf{p}) = \Delta(\mathbf{p^k})$, $\forall k \ge 1$.

Therefore, if p is stationary, then $\Delta(\mathbf{p}) = \Delta(\mathbf{p^k}) \le (1 - n^{-10n})\Delta(\mathbf{p})$ which implies $\Delta(\mathbf{p}) = 0 \implies \mathbf{p}_u \le \frac{1}{n}$, $\forall u$. But if for some u, $\mathbf{p}_u < \frac{1}{n}$ then $\sum_u \mathbf{p}_u < \sum_u \frac{1}{n} \implies 1 < 1$. Thus $\mathbf{p}_u = \frac{1}{n}$ and so uniform distribution is the only stationary distribution.

(b) Let u be a vertex of G. Let the initial distribution $\mathbf{p}$ be such that $\mathbf{p}_u = 1$. Then $X_0 = u$, hence $(X_0, X_1, \ldots)$ is a random walk from the vertex u.

Let v be a vertex of G (we may have $v = u$). Let N_m be the fraction of times we hit the vertex v in a m-step random walk from u. Then we have

$$\begin{aligned}\lim_{m\to\infty} E[N_m] &= \lim_{m\to\infty} \frac{1}{m} \sum_{k=0}^{m} P(X_k = v) \\ &= \lim_{m\to\infty} \frac{1}{m} \sum_{k=0}^{m} \mathbf{p}_v^k = \frac{1}{n} \qquad (\text{since } \mathbf{p}^k \to \pi)\end{aligned}$$

Furthermore, we have that $\lim_{m\to\infty} E[N_m] = \frac{1}{n} \implies E[N_m] > \frac{1}{2n}$ for sufficiently large m, that is, expected number of times a m-step random walk from u hits v will be $E[mN_m] > \frac{m}{2n}$ as required.

(c)

Assume that G is connected. We shall show that $E_u \le 2n$. For the sake of contradiction, suppose not. That is $E_u > 2n$ which means, expected number of times any $2n$-step random walk from u hits u is ≤ 1. So for sufficiently large m, expected number of times any m-step random walk from u hits u is $\le \frac{m}{2n}$. But in previous part we showed that this value is strictly greater than $\frac{m}{2n}$, leading us to a contradiction. Hence $E_u \le 2n$.

Now if G was not connected then let G_u be the connected component containing u. Since G_u satisfies all the above properties, we get $E_u \le 2|V(G_u)| \le 2n$.

(d)

Let $E_{u,v}$ denote expected number of steps to hit v starting from u. Observe that, we can express *time to hit u starting from u* as *starting from u, go to a random neighbour v and then return to u from v*, using conditional expectation. This gives:

$$\begin{aligned}E_u &= \sum_{v:v\in N(u)} \frac{1 + E_{v,u}}{d+1} + \frac{1}{d+1} \\ \implies (E_u - 1)(d+1) &= \sum_{v:v\in N(u)} E_{v,u}\end{aligned}$$

By part (c), we know that $E_u \le 2n$ for any vertex u. Therefore, $E_{v,u} < (d+1)E_u \le 2n(d+1) \le 2n^2 \le 100n^2$ for every edge (v, u).

Now for any two vertices u, v such that there is path of length atmost k: $u - u_1 - u_2 - \cdots v$, we get that $E_{u,v} \le E_{u,u_1} + E_{u_1,u_2} + \cdots < 2kn^2 \le 100kn^2$.

Now if s and t are connected in a graph then they can be atmost distance n apart. Therefore, using Markov's inequality:

$$P(s \text{ hits } t \text{ in } > 1000n^3 \text{ steps}) \le \frac{E_{s,t}}{1000n^3} \le \frac{100.n.n^2}{1000n^3} = \frac{1}{10}$$

(e)

Let X be the number of steps required to hit a vertex t from a vertex s in the graph G' and let s and t are l-steps apart ($l \geq 0$). Now let a k-step random walk, in G, from a vertex s hits the vertex t with probability atleast 0.9, then $k \geq l$.

If $l = 0$ (that is $s{=}t$), then using part (c), we can show that $E(X) = E_s \leq 2n$. Hence, by Markov's inequality we have

$$P(X > 10n^2k) \leq \frac{E(X)}{10n^2k} \leq \frac{2n}{10n^2k} = \frac{1}{5nk} < \frac{1}{2}$$

Now suppose $l > 0$, then using part (d), we can show that $E(X) = E_{st} \leq 2ln^2$. Since $k \geq l$, using Markov's inequality we have

$$P(X > 10n^2k) \leq \frac{E(X)}{10n^2k} \leq \frac{2ln^2}{10n^2k} = \frac{l}{5k} \leq \frac{1}{5} < \frac{1}{2}$$

Hence, in any case, a random walk from s hits t in $10n^2k$ steps with probability atleast $\frac{1}{2}$ in the graph G'.