CS 830: Randomness and Complexity 3/17/05

Lecture 18: Nisan's Generator for Space Bounded Computation (Part 2)

instructor: Jin-Yi Cai Scribe: Bess Berg, Michael Kowalczyk

1 An Aside on Matrix Norms

Recall that in our discussion we are using a matrix notation where M_{ji} is the probability of going from state i to state j, rather than the more common Markov chain notation, which is reversed. Also recall that the matrix 1-norm is defined by

$$\|M\|_1 = \max_{\|x\|_1=1} \|Mx\|_1 = \max_j \sum_i |m_{ij}|$$

One mnemonic for this is that the "1" is reminiscent of a vertical column in the matrix, so we are looking at the maximum column. In the case of $\|M\|_\infty$,

$$\|M\|_\infty = \max_i \sum_j |m_{ij}|,$$

the "∞" is drawn more horizontally and this reminds us to take the maximum row of the matrix. Of course, $\|M^T\|_1 = \|M\|_\infty$.

Not surprisingly, there is also a notion for the p-norm of a matrix for any $p, 1 \leq p < \infty$.

$$\|M\|_p = \max_{\|x\|_p=1} \|Mx\|_p$$

In fact, it can be shown that for p, q such that $\frac{1}{p} + \frac{1}{q} = 1$, $|\langle x, y\rangle| \leq \|x\|_p \|y\|_q$. Note that this holds for the case where q limits to ∞, which will be useful later on.

$$|\langle x, y\rangle| \leq \|x\|_1 \|y\|_\infty \tag{1}$$

2 Back to Nisan's

We want to prove the following technical lemma from last lecture.

Lemma 2.1 $\Pr_{h_1,\ldots,h_k \in H}(\|Q(G_k(*, h_1, \ldots, h_k)) - Q(U_{t\cdot 2^k})\| > (2^k - 1)\cdot \epsilon) \leq \frac{2^{5S}\cdot k}{\epsilon^2 \cdot 2^t}$

Proof.

We already showed that Claim 1 is true, but we still need to show that Claim 2 holds in order to complete the proof by induction.

Claim 1

$$\Pr[\neg E1 \vee \neg E2] \leq \frac{2^{5S} k}{\epsilon^2 2^t}$$

Claim 2

$$(E1 \wedge E2) \Rightarrow \|Q(G_k(*, h_1, \ldots, h_k)) - Q(U_{t\cdot 2^k})\| \leq (2^k - 1) \cdot \epsilon$$

$$\begin{aligned}
\text{Where } E1 &: \|Q(G_{k-1}(*, h_1, \ldots, h_{k-1})) - Q(U_{t\cdot 2^{k-1}})\| \leq (2^{k-1} - 1) \cdot \epsilon \\
E2 &: \text{For all } i, j, \ell \in \{1, \ldots, 2^S\}, h_k \text{ is } (2^{-2S}\epsilon)\text{-random for } (B_{j,\ell}^{h_1,\ldots,h_{k-1}}, B_{\ell,i}^{h_1,\ldots,h_{k-1}}) \\
B_{i,j}^{h_1,\ldots,h_{k-1}} &= \{x : Q(G_{k-1}(x, h_1, \ldots, h_{k-1})) \text{ goes from } i \text{ to } j\}
\end{aligned}$$

Note that we are really considering a transition from state j to state i, but we analyze this in the context of an intermediate state. To make the notation more manageable, we first denote

$$\begin{aligned}
M &= Q(G_k(*, h_1, \ldots, h_k)) \\
N &= Q(U_{t\cdot 2^k}) \\
L &= Q(G_{k-1}(*, h_1, \ldots, h_{k-1}))
\end{aligned}$$

Note that L^2 intuitively interpolates between the "extremes" M and N. On a high level, this is the idea that guides the analysis. Note that

$$\begin{aligned}
M_{j,i} &= \sum_{\ell=1}^{2^S} \Pr_x[x \in B_{\ell,i}^{h_1,\ldots,h_{k-1}} \wedge h_k(x) \in B_{j,\ell}^{h_1,\ldots,h_{k-1}}] \\
(L^2)_{j,i} &= \sum_{\ell=1}^{2^S} \Pr_{x,y}[x \in B_{\ell,i}^{h_1,\ldots,h_{k-1}} \wedge y \in B_{j,\ell}^{h_1,\ldots,h_{k-1}}] \\
&= \sum_{\ell=1}^{2^S} \Pr_x[x \in B_{\ell,i}^{h_1,\ldots,h_{k-1}}] \cdot \Pr_y[y \in B_{j,\ell}^{h_1,\ldots,h_{k-1}}]
\end{aligned}$$

Now it follows from E2 that

$$|M_{j,i} - (L^2)_{j,i}| \leq \sum_{\ell=1}^{2^S} 2^{-2S} \cdot \epsilon = 2^{-S} \cdot \epsilon$$

Remembering that the norm can be characterized as the maximum column sum, we get

$$\|M - L^2\| \leq \epsilon.$$

Now we need to handle $\|L^2 - N\|$. Clearly, $N = (N_1)^2$ where $N_1 = Q(U_{t\cdot 2^{k-1}})$. So, by using E1, we get

$$\begin{aligned}
\|M - N\| &\leq \|M - L^2\| + \|L^2 - N_1^2\| \\
&\leq \epsilon + \|L^2 - LN_1 + LN_1 - N_1^2\| \\
&\leq \epsilon + \|L^2 - LN_1\| + \|LN_1 - N_1^2\| \\
&\leq \epsilon + \|L\| \cdot \|L - N_1\| + \|L - N_1\| \cdot \|N_1\| \\
&= \epsilon + 2 \cdot \|L - N_1\| \\
&= \epsilon + (2^k - 2) \cdot \epsilon \\
&= (2^k - 1) \cdot \epsilon
\end{aligned}$$

With both claims secured, the lemma is complete. ♣

Now we can finally address the question of how the probability of the automaton Q accepting is effected by using pseudorandomness instead of pure randomness, that is, the probability of Q accepting under M versus N.

Let i_0 be the starting state of Q, and J be the set of accepting states. Also, let e_{i_0} be the column vector with 1 in the i_0 place, and 0 elsewhere. Finally, let e_J be the column vector with 1s in all positions $\in J$, and 0 elsewhere. Then using equation (1) we can write for fixed $h_1, \ldots, h_k$,

$$\begin{aligned}
\Pr[Q \text{ accepts under } M] &= e_J^T \cdot M e_{i_0} \\
\Pr[Q \text{ accepts under } N] &= e_J^T \cdot N e_{i_0} \\
\Pr[Q \text{ accepts under } M] - \Pr[Q \text{ accepts under } N] &= |\langle e_J, (M-N)e_{i_0} \rangle| \\
&\leq \|e_J\|_\infty \cdot \|(M-N) \cdot e_{i_0}\|_1 \\
&\leq \|M-N\|_1
\end{aligned}$$

Although not essential for our argument, the stronger statement

$$\Pr[Q \text{ accepts under } M] - \Pr[Q \text{ accepts under } N] \leq \frac{1}{2}\|M-N\|_1$$

is also true, and this is left as an exercise.

Now we need to figure out the probability when $h_1, \ldots, h_k$ vary. Let A be the event that Q accepts under G_k, B be the event that Q accepts under N, and C be the event that $\|A - B\| > (2^k - 1) \cdot \epsilon$. Then

$$\begin{aligned}
\Pr_{h_1,\ldots,h_k,x}[A] - \Pr_x[B] &= P(A) - P(B) \\
&= P(C) \cdot P(A|C) + P(\neg C) \cdot P(A|\neg C) - [P(C) \cdot P(B) + P(\neg C) \cdot P(B)] \\
&= P(C) \cdot (P(A|C) - P(B)) + P(\neg C) \cdot (P(A|\neg C) - P(B)).
\end{aligned}$$

Picking $\epsilon = \frac{2^{-2S}}{2^k-1}$, and $t = 13S$ we can write

$$\begin{aligned}
\left| \Pr_{h_1,\ldots,h_k,x}[A] - \Pr_x[B] \right| &\leq P(C) \cdot |P(A|C) - P(B)| + P(\neg C) \cdot |P(A|\neg C) - P(B)| \\
&\leq \frac{2^{5S} \cdot k}{\epsilon^2 \cdot 2^t} + (2^k - 1) \cdot \epsilon \\
&= \frac{2^{9S} \cdot k \cdot (2^k-1)^2}{2^{13S}} + (2^k - 1) \cdot \frac{2^{-2S}}{2^k - 1} \\
&\leq 2^{-2S} \cdot S + 2^{-2S} \\
&< 2^{-S}.
\end{aligned}$$

This concludes the proof of

THEOREM 2.2 (NISAN) *If L is accepted by a probabilistic algorithm A with 2 sided (or 1 sided, respectively) error using space S, time T, and R random bits, then L is also accepted by a probabilistic algorithm B with 2 sided (1 sided) error using $O(S \log \frac{R}{S}) = O(S^2)$ space, $O(ST \log \frac{R}{S}) = O(S^2 \cdot T)$ time, and $O(S \cdot \log \frac{R}{S}) = O(S^2)$ random bits.*