Two corollaries follow from forward direction of the main theorem from Lecture 21 and the standard use of pseudo random generator for derandomization of a randomized algorithms $A$.

**Corollary 1** If $\text{EXPTIME}$ cannot be approximated by polynomial size circuits, then $\text{BPP} \subseteq \bigcap_{n>0} \text{DTIME}(2^{n^e})$.

**Exercise 1** Strengthen the antecedent of the above corollary enough so that consequence is "$\text{BPP} \subseteq P$".

**Corollary 2** $\text{RAC}^0 \subseteq \bigcup_c \text{DSPACE}((\log n)^c) \subseteq \bigcup_c \text{DTIME}(2^{c \log^e n})$.

**Proof.** ($\iff$ is left out.) $(1) \implies (2)$ is easy using Yao’s XOR lemma.

**Exercise 2** Yao’s XOR lemma implies

$(1) \implies (2)$.

Let $s(t)$ be any function such that $l \leq s(l) \leq 2t$. If $\exists$ a function in $\text{EXPTIME}$ that cannot be approximated by circuits of size $s(l)$ (in the weak sense), then for some $c > 0$, $\exists$ another function in $\text{EXPTIME}$ whose hardness $H_{f^c}(l) \geq s(l^c)$.

(This is strong non-approximation ability.)

$(2) \implies (3)$.

There are two subparts of the proof here (two lemmas). The first subpart is the obstruction of pseudo random generator $G : l \rightarrow s(l^c)$, where $l^c = n$, using a hard function $f$, by first constructing a $(\log n, l^{1/2})$ design on $\{1, \ldots, l\}$, (Here $m = l^{1/2}$ and $k = \log n$) and show the pseudo random generator satisfies condition $cl$. The second subpart is to show such a design exists.

For $x \in \{0, 1\}^l$,

$G(x) = f(z_1), \ldots, f(z_n)$,

where $z_i$ is the restriction of $x$ to the indices in $S_i$.

**Example 1** $x = 01110101$

$l = 8, S_i = \{1, 3, 5\}$

$z_i = 010$.

**Subpart 1** If $G$ is based on a $(\log n, l^{1/2})$ design, then it is a pseudo random generator satisfying the bias requirement in the statement of the theorem. This is Lemma 3 stated in Lecture 21.
Subpart 2 \((\log n, t^{1/2})\) design exists and can be constructed in \(DTIME(2^t)\), where \(n = s(t^e)\). This will be Lemma 4 stated in the next lecture.

Aside Sitharam'93 has a 2 line proof of subpart 1 using Fourier transforms of Abelian groups showing relations between pseudorandom generators and learning algorithms of sampling.

Proof of Subpart 1 Proof by contradiction. Assume such a pseudorandom generator \(G\) based on the \((\log n, t^{1/2})\) design does not satisfy condition cl.

......