CMPT 881 - Pseudorandomness: Problem Set 2
Due: November 25 (at the beginning of the class)

Valentine Kabanets

November 2, 2004

Reminder: you are encouraged to work in groups of two or three; however, you must turn in your own write-up and note with whom you worked. You may consult the course notes and optional texts. Please attempt all problems.

1. **Extractors**

   (a) In this question you are asked to show that randomness extraction is possible only from sources that are statistically close to sources with high min-entropy; thus, high min-entropy is both sufficient and necessary for randomness extraction. More formally, let X be any distribution over $\{0,1\}^n$ and let $Ext : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$. Suppose that $Ext(X, U_d)$ is ϵ-close to the uniform distribution U_m. Prove that X is $O(\epsilon)$-close to some k-source X' where $k \geq m - d - 1$. (Hint: Consider the set A of strings $z \in \{0,1\}^m$ that get assigned probability more than $2 * 2^{-m}$ by $Ext(X, U_d)$. Argue that the set A gets the probability at most 2ϵ with respect to the distribution $Ext(X, U_d)$. Fix the seed $y \in \{0,1\}^d$ so that $\mathbf{Pr}[Ext(X, y) \in A] \leq 2\epsilon$. Argue that for every x such that $Ext(x, y) \notin A$, we have $\mathbf{Pr}[X = x] \leq 2^{-(m-d-1)}$. Conclude that there exists a k-source X' such that $\Delta(X, X') \leq 2\epsilon$.)

   (b) Show that every k-source X over $\{0,1\}^n$, for large k, can be viewed as a block source $X = YZ$. More precisely, let $X = YZ$ be an $(n - \Delta)$-source, for some Δ, where Y is a distribution over ℓ-bit strings for any $\ell \leq n$, and Z is a distribution over strings of length $m = n - \ell$. Prove that Y is a $(\ell - \Delta)$-source. Prove also that, for every $\epsilon > 0$, with probability at least $(1 - \epsilon)$ over the choice of y according to the distribution Y, the conditional distribution $Z|_{Y=y}$ is an $(m - \Delta - \log(1/\epsilon))$-source.

2. **Error reduction in BPP algorithms, using extractors** Let A be any BPP algorithm that on input of length ℓ uses m random bits, and has some constant error probability (say, 1/4). Using a (k, ϵ) extractor $Ext : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ for appropriate parameters n, k, d, show how to reduce the error probability in the algorithm A to less than 2^{-t}, for any $t = \text{poly}(\ell)$, while using at most $m + t$ random bits. Your new randomized algorithm should still run in polytime. Conclude that that every BPP algorithm A has an equivalent BPP algorithm A' using r random bits such that A' errs on at most $2^{\sqrt{r}}$ of all r-bit random strings. (Hint: Your algorithm A' will pick a string $z \in \{0,1\}^n$ uniformly at random, and output the majority decision of A when A uses the strings $Ext(z, s)$ instead of its random strings, over all seeds $s \in \{0,1\}^d$. Analyze the error probability of this algorithm A', and pick the extractor parameters n, k, d, ϵ appropriately.)

3. **Error-correcting codes based on expanders**
Let $G = (L, R, E)$ be a bipartite $(\alpha n, (1-\epsilon)d)$-expander on (n, m) vertices for $m < n$, with left degree d, for a constant d; that is, every set $S \subseteq L$ of size at most αn is expanded by a factor $(1-\epsilon)d$. (Such "lossless" expanders can be constructed explicitly, using the "extractor technology".) Assume that $\epsilon < 1/12$. The graph G defines a binary error-correcting code $\mathcal{C} \subset \{0,1\}^n$ as follows: A string $c \in \{0,1\}^n$ is a codeword if, for every node $i \in R$ with neighbours $j_1, \ldots, j_k \in L$, we have $c_{j_1} \oplus \cdots \oplus c_{j_k} = 0$, where $\oplus$ is addition modulo 2. That is, we view the nodes in L as positions in an n-bit string c, and nodes in R as parity-check constraints, where the constraint corresponding to vertex $i \in R$ checks c in the positions determined by the neighbours of i in L; a string c is a codeword if all m parity check constraints are satisfied.

   (a) Consider a codeword $c \in \mathcal{C}$ of minimum Hamming weight (i.e., with minimum number of 1's). Show that the Hamming weight of this codeword c is greater than αn (and hence, the minimum relative distance of the code $\mathcal{C}$ is greater than α). (Hint: Prove and then use the following fact: every set $S \subseteq L$ of size at most αn has at least $(1-2\epsilon)d|S|$ *unique* neighbours, where a vertex $v \in R$ is a unique neighbour for S if v is connected by an edge to exactly one node in S.)

   (b) Prove that the following decoding algorithm for $\mathcal{C}$ corrects $(1-3\epsilon)\alpha n$ errors in $O(n \log n)$ time.
   Let $m \in \{0,1\}^n$ be a received message. Label the nodes in L with the bits of the string m (so that node $i \in L$ gets the label m_i). Until all the parity checks are satisfied, repeat the following: in parallel, each node $i \in L$ flips its value if the number of unsatisfied parity checks among its d neighbours is at least $2d/3$; otherwise, the node i retains its old value.
   You should fill in the details in the proof outline below.

      i. Let $S \subseteq L$ be a set of error positions at the beginning of a parallel round. Let $N(S) \subseteq R$ be the set of neighbours of S. If $|S| \leq \alpha(1-3\epsilon)n$, then S has at least $(1-2\epsilon)d|S|$ unique neighbours in R (by the previous question). By an averaging argument, show that at least $(1-6\epsilon)$ fraction of nodes in S will have at least $2d/3$ unique neighbours in R. Conclude that at least $(1-6\epsilon)|S| \geq |S|/2$ nodes in S will correct their labels.

      ii. Let $T \subseteq L \setminus S$ be the set of positions outside S that have correct labels before the parallel round, but then incorrectly flip their values during the round. Prove that each node in T has at least $2d/3$ of its neighbours inside the set $N(S)$.

      iii. Using the result of the previous item, show that $|T| \leq \frac{3\epsilon|S|}{1-3\epsilon} \leq |S|/3$. (Hint: The idea is that if T is large, then $T \cup S$ should expand significantly. But, since a lot of neighbours of T are already in $N(S)$, no large expansion of $T \cup S$ is possible.)

      iv. Conclude that each parallel round increases the number of correct positions of the message m by at least $|S|/6$, and so after $O(\log n)$ steps all incorrect positions will be eliminated.