Hitting Set Generators

Lauri Tarkkala

November 25, 2004

Introduction

Recall the card game between Alice and Bob. There is a deck of N cards. Alice randomly marks $(1-\epsilon)N$ cards with $0 < \epsilon < 1$. Bob must select at most d cards. If Bob picks a card marked by Alice then Bob wins, otherwise Alice wins.

This idea was the foundation for using dispersers (incl. extractors) to deal with insufficient randomness.

The model is too general for derandomization however. The idea behind hitting set generators is to limit Alices choices to a restricted set. Bob must now try to produce picks that will always cover at least one card in any of Alices choices.

Hitting Sets

Definition Let $\mathcal{C}$ be a set system over a finite universe. A $(1-\epsilon)-$ *hitting set* for $\mathcal{C}$ is a set $\mathcal{H}$ s.t. $\forall S \in \mathcal{C}, |S|/|C| \geq 1-\epsilon$: $\mathcal{H} \ \cap S \neq \emptyset$.

Definition A $(1-\epsilon)-$ *hitting set generator* for $\mathcal{C}$ is a deterministic algorithm which on input r outputs $(1-\epsilon)-$ *hitting set* $H_r \subseteq \{0,1\}^r$ for $\mathcal{C}$.

ONE-SIDED CIRCUIT-ACCEPTANCE PROBABILITY ESTIMATION

Let $C : \{0,1\}^r \rightarrow \{0,1\}$ be a Boolean circuit of size r that is guaranteed ("promised") that either $\Pr[C(x) = 1] = 0$ or $\Pr[C(x) = 1] \geq 1/2$ when $x \in \{0,1\}^r$ is random.

The CIRCUIT-ACCEPTANCE PROBABILITY ESTIMATION problem is to determine if $\Pr[C(x) = 1] \geq 1/2$.

This problem is **pRP**-complete. Essentially a circuit accepting a language using randomness can be transformed into a circuit like C described above.

Derandomization

Lemma If there exists a 1/2-hitting set generator for the sets decided by circuits of size r running in time $t(r)$, then $\mathbf{RP} \subseteq \mathbf{TIME}(t(n^{\mathcal{O}(1)})n^{\mathcal{O}(1)})$.

Let us consider circuits $C : \{0,1\}^r \rightarrow \{0,1\}$. Let $\mathcal{C}$ be the set of subsets of $\{0,1\}^r$ recognized by C.

A derandomization of **RP** can be performed if one can construct for arbitrary C a 1/2-hitting set over $\mathcal{C}$. Note that because we are considering **RP** then it is sufficient to accept with a 1/2-probability of error (hence the 1/2-hitting set).

STRONG ONE-SIDED CIRCUIT-ACCEPTANCE PROBABILITY ESTIMATION

Let $\epsilon > 0$. Let $C : \{0,1\}^n \rightarrow \{0,1\}$ be a Boolean circuit of size n^q. Let it be guaranteed that either $\Pr[C(x) = 1] = 0$ or $\Pr[C(x) = 1] \geq 1 - 2^{-n+n^e}$ for a random $x \in \{0,1\}^n$.

The (q, ϵ)- STRONG ONE-SIDED CIRCUIT ACCEPTANCE PROBABILITY ESTIMATION problem is to decide if $\Pr[C(x) = 1] \geq 1 - 2^{-n+n^e}$ for a random $x \in \{0,1\}^n$.

Lemma For all $\epsilon > 0$ there exists a constant q such that (q, ϵ)-STRONG ONE-SIDED CIRCUIT ACCEPTANCE PROBABILITY ESTIMATION (SOSCAPE) is complete for **pRP**.

SOSCAPE is in **pRP** because the ONE-SIDED CIRCUIT-ACCEPTANCE PROBABILITY ESTIMATION problem is. This is due to the behaviour of the inequality $-n + n^e$ when n grows.

Let M be any randomized Turing-machine in **pRP**.

Construct a machine M' that requires $R(n) > n$ random bits and has error probability $2^{-R(n)+R(n)^\epsilon}$ using earlier results.

Convert M' to a circuit C that accepts as input x and random bits $R(|x|)$. Convert this circuit into a circuit C_x that builds x into the circuit and takes only the randomness as input.

This circuit has size $R(|x|)^q$ for some constant q. The size of q depends on the size of M which depends on ϵ.

Lemma For all $\epsilon > 0$ there exists $q \geq 1$ s.t. if a $(1 - 2^{-r+r^{\epsilon}})$ – hitting set generator for the sets decided by circuits $\mathcal{C} : \{0,1\}^r \rightarrow \{0,1\}$ of size r^q exists, running in time $t(r)$, then $\mathbf{RP} \subseteq \mathbf{TIME}(t(n^{\mathcal{O}(1)})n^{\mathcal{O}(1)})$.

This Lemma follows from the two previous Lemmas.

This Lemma shows that a $(1 - 2^{-r+r^{\epsilon}})$-hitting set generator is sufficient to derandomize **RP**.

Note that intuitively it seems at least that it should be easier to generate a $(1 - \epsilon)$-hitting set generator than a $(1 - \epsilon')$ generator when $\epsilon < \epsilon'$ because we have to hit “only larger sets”.

Lemma For all $\epsilon > 0$ there exists $q \geq 1$ and $\delta > 0$ s.t. there is a polynomial time computable function that when input a $(1 - 2^{-r+r^{\epsilon}})$-hitting set H in $\{0,1\}^r$ for the sets decided by circuits of size r^q outputs a $1/2$-hitting set in $\{0,1\}^{r'}$ for the sets decided by circuits of size r' s.t. $r' = r^{\delta}$.

This is the "randomness-amplification" property and it follows from taking an explicit extractor to the set H.

Notes

If a hitting set generator running in polynomial time exists then $\mathbf{BPP} = \mathbf{P}$ (recall that $\mathbf{BPP} = \mathbf{pRP}[\mathbf{pRP}]$).

A way to describe a hitting set is that it is a “derandomized” disperser.

A disperser ($E \subseteq U \times V$) wants to hit every subset of V of density at least $1 - \epsilon$ with high probability (if we hit more than $\epsilon|V|$ elements, we immediately hit every set of density $1 - \epsilon$).

A hitting set hits every subset V of density at least $1 - \epsilon$.

It is not known how to create efficient hitting set generators unconditionally.

Pseudorandom Generators

A hitting set generator is not sufficient to derandomize algorithms with two-sided error. A concept analoguous to an extractor is required.

Definition Let $\mathcal{C}$ be a set system over $\{0,1\}^r$. A pseudorandom set with error ϵ for $\mathcal{C}$ is a (multi-)set $P \subseteq \{0,1\}^r$ s.t. for all $S \in \mathcal{C}$ $|\Pr_{x \in P}[x \in S] - \Pr_{x \in \{0,1\}^r}[x \in S]| \leq \epsilon/2$.

A pseudorandom set induces a distribution that is ϵ-close to the one induced by the uniform distribution.

Definition A pseudorandom generator with error ϵ for circuits of size r is a deterministic algorithm that on input r outputs a pseudorandom set in $\{0,1\}^r$ for the sets decided by circuits of size r.

It is not known how to create efficient pseudorandom generators unconditionally.

Pseudorandom Generators

Lemma If there exists a pseudorandom generator running in time $t(r)$ with error less than 1/3 for the sets decided by circuits of size r then $\mathbf{BPP} \subseteq \mathbf{TIME}(t(n^{\mathcal{O}(1)})n^{\mathcal{O}(1)})$.

This flows from the $pBPP completeness of CIRCUIT ACCEPTANCE PROBABILITY ESTIMATION.

Pseudorandom Generators

Pseudorandom Generators were developed for the purposes of cryptography. Derandomization is “easier” than cryptography.

Cryptographic pseudorandom generators must be able to sample the pseudorandom set efficiently, e.g. in time polynomial to the length of an index to the pseudorandom set.

Cryptographic pseudorandom generators must be pseudorandom for all polynomially sized circuis (r^k for all constants k).

Derandomization requires only that we are able to generate the pseudorandom set in polynomial time, as we simulate the algorithm over the full set. Derandomization also requires that the set is pseudorandom for circuits of size r.

Simplifying, derandomizing pseudorandom generators can use as much time to generate the set as it takes to “break it”.

Cryptographic Pseudorandom Generators

Cryptographic pseudorandom generators have been constructed based on average-case hardness assumptions (e.g. intractability of integer factorization and discrete logarithms).

Impagliazzo et al showed in 1989 that cryptographic pseudorandom generators can be constructed under the assumption that cryptographic one-way functions exist. A cryptographic one-way function is a function $f(x)$ s.t. for all polynomial time computable algorithms g and a uniformly random x the probability that $g(f(x)) = x$ is negligible.

Theorem If a cryptographic one-way function exists then $\mathbf{BPP} \subseteq \mathbf{SUBEXP}$.

Theorem If a cryptographic one-way function exists then $\mathbf{P} \neq \mathbf{NP}$.

Sipser's Hitting Set Generator

Theorem There exists a $\delta > 0$ s.t. if for some $c \geq 1$ there is a language $L \in \mathbf{TIME}(2^{cn})$ s.t. any Turing machine for L use space at least $2^{(c-\delta)n}$ on all sufficiently large input lengths n then $\mathbf{pP} = \mathbf{pRP}$ and $\mathbf{P} = \mathbf{BPP}$.

According to a previous Lemma it is sufficient to show that under the assumptions in the theorem there exists a $(1 - 2^{-r+r^{1/2}})$-hitting set generator for the sets accepted by circuits of size r^q with polynomial (in r) running time for some constant $q \geq 1$.

The Miltersen survey shows the construction for $\delta = (5q)^{-1}$

Sipser's Hitting Set Generator Construction (1/4)

Let r be the input parameter.

Let $L = \{0,1\}^*$ be a language in $\mathbf{TIME}(2^{cn})$ and let M be a Turing machine accepting this language. Assume M works using the alphabet $\{0,1\}$. Assume that the working tapes have a maximum length of 2^{cn}.

Split each work tape into $2^{cn}/r$ blocks of length r bits.

For an input x to M let H_x be the union of all the configurations the blocks take on during the computation. Let $H_{n,r}$ be the union of H_x for all $x \in \{0,1\}^n$.

Let H_r be $H_{2q \log r, r}$. This set can be construtced in polynomial time (in r).

The claim is that H_r is a $(1 - 2^{-r+r^{1/2}})$-hitting set generator for circuits of size r^q. Proof is by contradiction.

Sipser's Hitting Set Generator Construction (2/4)

If H_r is not a hitting set generator then for arbitrary values r there must exist circuits $C_r : \{0,1\}^r \rightarrow \{0,1\}$ of size r^q s.t. $|Z(C_r)| \leq 2^{r^{1/2}}$ and $H_r \subseteq Z(C_r)$. $Z(C_r) = \{x|C_r(x) = 0\}$. Note that $((1 - 2^{-r+r^{1/2}})(2^r) = 2^r - 2^{r^{1/2}})$.

Define a compression function $c(x)$ and its inverse $d(x)$ for H_r. $c(x)$ takes as input an element from H_r and outputs an integer in $\{0,1\}^{r^{1/2}}$ that corresponds to the rank in $Z(C_r)$.

Sipser's Hitting Set Generator Construction (3/4)

The simulation of M on inputs of length $n = 2q \log r$ using space $2^{(c-\delta)n}$ can be done as follows using machine M'.

M' is given C_r (the "program") on a separate special tape.

M' stores on all its work tapes $c(y)$ for each block y on each work tape of M. The blocks that are under a tapehead of M are not stored in compressed form. M' uncompressing/compressing configurations as it moves tapeheads.

The space required is $\mathcal{O}(r^q)$ for the compression/decompression operations, $\mathcal{O}(r)$ for the uncompressed blocks and $\mathcal{O}(r^{1/2}2^{cn}/r)$ for the compressed ones. Total is $\mathcal{O}(r^q + 2^{cn}/r^{1/2}) \leq 2^{(c-\delta)n}$.

Sipser's Hitting Set Generator Construction (4/4)

The simulation can be performed using the "not a hitting set" for any circuit $C : \{0,1\}^r \rightarrow \{0,1\}$ if $|Z(C)| \leq 2^{r^{1/2}}$ and all blocks are members of $Z(C)$. This can be tested on the fly.

If H_r is not a hitting set then a circuit (possibly C_r) will be found and we have a machine that accepts the language $x \in L$ by finding a circuit encoding the x into it accepting the randomness as input.

This machine will use less space than $2^{(c-\delta)n}$ and we have a contradiction.