Expander Graphs and Their Applications (XVIII)

Yijia Chen
Shanghai Jiaotong University

Review of the Previous Lecture

Long-Code Theorem

There exists a *Long-Code Test* $\mathbb{T}$ which is randomized algorithm that has input a function $\psi : \{0,1\}^s \rightarrow \{0,1\}$, and also oracle access to a string $A : L \rightarrow \{0,1\}$ *folded over true and over* ψ. $\mathbb{T}$ reads the input ψ and tosses some random coins. Based on these it computes a three-bit predicate

$$w : \{0,1\}^3 \rightarrow \{0,1\}$$

and three locations $f_1, f_2, f_3 \in L$ in which it queries the string A. It then outputs $w(A(f_1), A(f_2), A(f_3))$. Denote an execution of $\mathbb{T}$ with access to input ψ and string A by $\mathbb{T}^A(\psi)$. Then the following hold.

- (**Perfect Completeness**): If $a \in \{0,1\}^s$ with $\psi(a) = 1$, then

$$\Pr\left[\mathbb{T}^{A_a}(\psi) \text{ accepts}\right] = 1.$$

- (**Strong Soundness**): For every $\delta \in [0,1]$ if $A : L \rightarrow \{0,1\}$ is folded over true and over ψ and at least δ-far from A_a for all $a \in \{0,1\}^s$ with $\psi(a) = 1$, then

$$\Pr\left[\mathbb{T}^{A}(\psi) \text{ rejects}\right] \geq \Omega(\delta).$$

Assignment Tester

Assignment Tester

Assignment Tester

Definition

An Assignment Tester with alphabet Σ_0 and reject probability $\varepsilon > 0$ is an algorithm $\mathbb{P}$ whose input is a circuit Φ over Boolean variables in the set X, and whose output is a constraint graph $G = \langle (V, E), \Sigma_0, \mathscr{C} \rangle$ such that $X \subseteq V$ and such that the following hold.

Assignment Tester

Definition

An Assignment Tester with alphabet Σ_0 and reject probability $\varepsilon > 0$ is an algorithm $\mathbb{P}$ whose input is a circuit Φ over Boolean variables in the set X, and whose output is a constraint graph $G = \langle (V, E), \Sigma_0, \mathscr{C} \rangle$ such that $X \subseteq V$ and such that the following hold.
Let $V' := V \setminus X$ and let $a : X \to \{0, 1\}$ be an assignment.

- **(Completeness)** if a satisfies Φ, then there exists some $b : V' \to \Sigma_0$ such that $\text{unsat}_{a \cup b}(G) = 0$.
- **(Soundness)** if a does not satisfy Φ, then for all $b : V' \to \Sigma_0$

$$\text{unsat}_{a \cup b}(G) \geq \varepsilon \cdot \min \left\{ \text{rdist}(a, s) \mid s : X \to \{0, 1\} \text{ that satisfies } \Phi \right\}.$$

Assignment Tester Theorem

Assignment Tester Theorem

Theorem

There is some $\varepsilon > 0$ *and an explicit construction of an assignment tester* $\mathbb{P}$ *with alphabet* $\Sigma_0 = \{0,1\}^3$ *and rejection probability* ε*.*

The construction

The construction

Recall that there is a Long-Code Test $\mathbb{T}$.

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**
Let $X = \{x_1, \ldots, x_s\}$ be a set of s Boolean variables. Also, let there be a Boolean variable for each $f \in L'_\psi$.

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**
Let $X = \{x_1, \ldots, x_s\}$ be a set of s Boolean variables. Also, let there be a Boolean variable for each $f \in L'_\psi$.
This can be expanded into a an assignment for L which is *folded over true and over* ψ.

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**
Let $X = \{x_1, \ldots, x_s\}$ be a set of s Boolean variables. Also, let there be a Boolean variable for each $f \in L'_{\psi}$.
This can be expanded into a an assignment for L which is *folded over true and over ψ*.

Define a modified test $\mathbb{T}'$:

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**
Let $X = \{x_1, \ldots, x_s\}$ be a set of s Boolean variables. Also, let there be a Boolean variable for each $f \in L'_\psi$.
This can be expanded into a an assignment for L which is *folded over true and over* ψ.

Define a modified test $\mathbb{T}'$: Given an input ψ, oracle access to an assignment $A : L \rightarrow \{0, 1\}$ folded over true and over ψ, and an assignment $\sigma : X \rightarrow \{0, 1\}$, run $\mathbb{T}$ on ψ and A *with probability* $1/2$,

The construction

Recall that there is a Long-Code Test $\mathbb{T}$. We now proceed to construct a system of constraints based on the test $\mathbb{T}$ in *two steps*.

1. **(Modified Test:)**

Let $X = \{x_1, \ldots, x_s\}$ be a set of s Boolean variables. Also, let there be a Boolean variable for each $f \in L'_\psi$.
This can be expanded into a an assignment for L which is *folded over true and over* ψ.

Define a modified test $\mathbb{T}'$: Given an input ψ, oracle access to an assignment $A : L \to \{0, 1\}$ folded over true and over ψ, and an assignment $\sigma : X \to \{0, 1\}$, run $\mathbb{T}$ on ψ and A *with probability* $1/2$, and otherwise choose a random $x_i \in X$ and a random $f \in L$, and test that $\sigma(x_i) = A(f) \oplus A(f \oplus e_i)$.

The construction (cont'd)

The construction (cont'd)

2. **(Creating the Constraints:)**

The construction (cont'd)

2. **(Creating the Constraints:)**
Introduce a new variable z_r *per outcome* r *of the coin tosses of* $\mathbb{T}'$.

The construction (cont'd)

2. **(Creating the Constraints:)**
Introduce a new variable z_r *per outcome* r *of the coin tosses of* $\mathbb{T}'$.
These variables will take values in $\{0, 1\}^3$, supposedly specifying the correct values of all three variables queried by $\mathbb{T}'$ on coin tosses r.

The construction (cont'd)

2. **(Creating the Constraints:)**
Introduce a new variable z_r *per outcome r of the coin tosses of* $\mathbb{T}'$.
These variables will take values in $\{0, 1\}^3$, supposedly specifying the correct values of all three variables queried by $\mathbb{T}'$ on coin tosses r.

We construct the following system of constraints:

The construction (cont'd)

2. **(Creating the Constraints:)**
Introduce a new variable z_r *per outcome r of the coin tosses of* $\mathbb{T}'$.
These variables will take values in $\{0,1\}^3$, supposedly specifying the correct values of all three variables queried by $\mathbb{T}'$ on coin tosses r.

We construct the following system of constraints:
For every $z_r \in Z$ and a variable y of the three bits accessed by $\mathbb{T}'$ on coin toss r, so $y \in X \cup L$.

The construction (cont'd)

2. **(Creating the Constraints:)**
Introduce a new variable z_r *per outcome* r *of the coin tosses of* $\mathbb{T}'$.
These variables will take values in $\{0, 1\}^3$, supposedly specifying the correct values of all three variables queried by $\mathbb{T}'$ on coin tosses r.

We construct the following system of constraints:
For every $z_r \in Z$ and a variable y of the three bits accessed by $\mathbb{T}'$ on coin toss r, so $y \in X \cup L$.
This constraint will check that the assignment for z_r would have satisfied $\mathbb{T}'$, and that it is consistent with the assignment for y.

The construction (cont'd)

The construction (cont'd)

The algorithm $\mathbb{P}$ outputs the constraint graph G:

The construction (cont'd)

The algorithm $\mathbb{P}$ outputs the constraint graph G:

- vertices of G are $X \cup L \cup Z$;
- constraints (and edges) are as specified in (2);
- the alphabet is $\Sigma_0 = \{0, 1\}^3$, where the Boolean variables $X \cup L$ take values only in $\{000, 111\}$ identified with $\{0, 1\}$

The construction (cont'd)

The algorithm $\mathbb{P}$ outputs the constraint graph G:

- vertices of G are $X \cup L \cup Z$;
- constraints (and edges) are as specified in (2);
- the alphabet is $\Sigma_0 = \{0, 1\}^3$, where the Boolean variables $X \cup L$ take values only in $\{000, 111\}$ identified with $\{0, 1\}$ (i.e., a constraint involving $y \in X \cup L$ immediately rejects if the value of y is not in $\{000, 111\}$).

Lemma

$\mathbb{P}$ taking $\psi : \{0,1\}^s \rightarrow \{0,1\}$ to G is an assignment tester, with $\Sigma_0 = \{0,1\}^3$ and constant rejection probability $\varepsilon > 0$.

Lemma

$\mathbb{P}$ *taking* $\psi : \{0,1\}^s \to \{0,1\}$ *to* G *is an assignment tester, with* $\Sigma_0 = \{0,1\}^3$ *and constant rejection probability* $\varepsilon > 0$.

Proof.

Lemma

$\mathbb{P}$ taking $\psi : \{0,1\}^s \to \{0,1\}$ to G is an assignment tester, with $\Sigma_0 = \{0,1\}^3$ and constant rejection probability $\varepsilon > 0$.

Proof.

We need to show:

Lemma

$\mathbb{P}$ *taking* $\psi : \{0,1\}^s \to \{0,1\}$ *to* G *is an assignment tester, with* $\Sigma_0 = \{0,1\}^3$ *and constant rejection probability* $\varepsilon > 0$.

Proof.

We need to show:

- **(Completeness)** If a satisfies ψ, then there exists some $b \in L \cup Z \to \Sigma_0$ such that $\text{unsat}_{a \cup b}(G) = 0$.
- **(Soundness)** If a does not satisfy ψ, then for all $b \in L \cup Z \to \Sigma_0$ we have $\text{unsat}_{a \cup c}(G) \geq \varepsilon \cdot \text{rdist}(a, c)$ for every $c : X \to \{0,1\}$ which satisfies ψ.

Lemma

$\mathbb{P}$ taking $\psi : \{0,1\}^s \to \{0,1\}$ to G is an assignment tester, with $\Sigma_0 = \{0,1\}^3$ and constant rejection probability $\varepsilon > 0$.

Proof.

We need to show:

- **(Completeness)** If a satisfies ψ, then there exists some $b \in L \cup Z \to \Sigma_0$ such that $\text{unsat}_{a \cup b}(G) = 0$.
- **(Soundness)** If a does not satisfy ψ, then for all $b \in L \cup Z \to \Sigma_0$ we have $\text{unsat}_{a \cup c}(G) \geq \varepsilon \cdot \text{rdist}(a, c)$ for every $c : X \to \{0,1\}$ which satisfies ψ.

For the completeness, let the assignment for the variables in L be A_a. It is then easy to assign the variables is Z in a consistent manner.

Proof of the Soundness

Proof of the Soundness

Let $\sigma : X \to \{0, 1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0, 1\}$ which satisfies ψ.

Proof of the Soundness

Let $\sigma : X \to \{0,1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0,1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \upharpoonright L$.

Proof of the Soundness

Let $\sigma : X \to \{0, 1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0, 1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \restriction L$.

Proposition. $\Pr\left[\mathbb{T}'^{A,\sigma}(\psi) \textit{ rejects}\right] = \Omega(\delta)$.

Proof. As in (1), we assume A is folded over true and over ψ.

Proof of the Soundness

Let $\sigma : X \to \{0,1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0,1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \restriction L$.

Proposition. $\Pr\left[\mathbb{T}'^{A,\sigma}(\psi) \text{ rejects}\right] = \Omega(\delta)$.

Proof. As in (1), we assume A is folded over true and over ψ. Assume first that $A : L \to \{0,1\}$ is $\delta/2$-far from A_a for all a satisfying ψ, then by the *Long-Code Theorem*, $\mathbb{T}$ rejects with probability $\Omega(\delta)$, so does $\mathbb{T}'$.

Proof of the Soundness

Let $\sigma : X \to \{0,1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0,1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \restriction L$.

Proposition. $\Pr\left[\mathbb{T}'^{A,\sigma}(\psi) \textit{ rejects}\right] = \Omega(\delta)$.

Proof. As in (1), we assume A is folded over true and over ψ. Assume first that $A : L \to \{0,1\}$ is $\delta/2$-far from A_a for all a satisfying ψ, then by the *Long-Code Theorem*, $\mathbb{T}$ rejects with probability $\Omega(\delta)$, so does $\mathbb{T}'$.

Otherwise, A is $\delta/2$-close to some $A_{a'}$ with a' satisfying ψ.

Proof of the Soundness

Let $\sigma : X \to \{0, 1\}$ and $\delta > 0$ with

$$\text{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0, 1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \restriction L$.

Proposition. $\Pr\left[\mathbb{T}'^{A,\sigma}(\psi) \textit{ rejects}\right] = \Omega(\delta)$.

Proof. As in (1), we assume A is folded over true and over ψ. Assume first that $A : L \to \{0, 1\}$ is $\delta/2$-far from A_a for all a satisfying ψ, then by the *Long-Code Theorem*, $\mathbb{T}$ rejects with probability $\Omega(\delta)$, so does $\mathbb{T}'$.

Otherwise, A is $\delta/2$-close to some $A_{a'}$ with a' satisfying ψ. We now compare a' and δ which are both assignments for the variables of ψ.

Proof of the Soundness

Let $\sigma : X \to \{0,1\}$ and $\delta > 0$ with

$$\mathrm{rdist}(\sigma, c) \geq \delta$$

for every $c : X \to \{0,1\}$ which satisfies ψ.

Fix some $b : L \cup Z \to \Sigma_0$ and let $A := b \upharpoonright L$.

Proposition. $\Pr\left[\mathbb{T}'^{A,\sigma}(\psi) \textit{ rejects}\right] = \Omega(\delta)$.

Proof. As in (1), we assume A is folded over true and over ψ. Assume first that $A : L \to \{0,1\}$ is $\delta/2$-far from A_a for all a satisfying ψ, then by the *Long-Code Theorem*, $\mathbb{T}$ rejects with probability $\Omega(\delta)$, so does $\mathbb{T}'$.

Otherwise, A is $\delta/2$-close to some $A_{a'}$ with a' satisfying ψ. We now compare a' and δ which are both assignments for the variables of ψ. Since a' satisfies ψ

$$\Pr_i\left[\sigma(x_i) \neq a'(x_i)\right] = \mathrm{rdist}(\sigma, a') \geq \delta.$$

Proof of the Soundness (cont'd)

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

A is $\delta/2$-close to $A_{a'}$, for all $i \in [s]$:

$$\begin{aligned}
&\Pr_{f \in L}\left[A(f) \oplus A(f \oplus e_i) = a'(x_i)\right] \\
\geq &\Pr_{f \in L}\left[A(f) = f(a') \text{ and } A(f \oplus e_i) = (f \oplus e_i)(a')\right] \\
\geq &\, 1 - 2 \cdot \delta/2 = 1 - \delta.
\end{aligned}$$

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

A is $\delta/2$-close to $A_{a'}$, for all $i \in [s]$:

$$\begin{aligned} &\Pr_{f \in L}\left[A(f) \oplus A(f \oplus e_i) = a'(x_i)\right] \\ \geq &\Pr_{f \in L}\left[A(f) = f(a') \text{ and } A(f \oplus e_i) = (f \oplus e_i)(a')\right] \\ \geq &\ 1 - 2 \cdot \delta/2 = 1 - \delta. \end{aligned}$$

The check fails if $a'(x_i) \neq \sigma(x_i)$ and yet $A(f) \oplus A(f \oplus e_i) = a'(x_i)$.

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

A is $\delta/2$-close to $A_{a'}$, for all $i \in [s]$:

$$\begin{aligned}
&\Pr_{f \in L}\left[A(f) \oplus A(f \oplus e_i) = a'(x_i)\right] \\
\geq &\Pr_{f \in L}\left[A(f) = f(a') \text{ and } A(f \oplus e_i) = (f \oplus e_i)(a')\right] \\
\geq &1 - 2 \cdot \delta/2 = 1 - \delta.
\end{aligned}$$

The check fails if $a'(x_i) \neq \sigma(x_i)$ and yet $A(f) \oplus A(f \oplus e_i) = a'(x_i)$. Altogether this occurs with probability $(1 - \delta) \cdot \delta \geq \delta/2$.

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

A is $\delta/2$-close to $A_{a'}$, for all $i \in [s]$:

$$\begin{aligned} &\Pr_{f \in L} \left[A(f) \oplus A(f \oplus e_i) = a'(x_i)\right] \\ \geq &\Pr_{f \in L} \left[A(f) = f(a') \text{ and } A(f \oplus e_i) = (f \oplus e_i)(a')\right] \\ \geq &\, 1 - 2 \cdot \delta/2 = 1 - \delta. \end{aligned}$$

The check fails if $a'(x_i) \neq \sigma(x_i)$ and yet $A(f) \oplus A(f \oplus e_i) = a'(x_i)$. Altogether this occurs with probability $(1-\delta) \cdot \delta \geq \delta/2$. The result follows from the fact that $\mathbb{T}'$ runs this test with probability $1/2$. $\dashv$

Proof of the Soundness (cont'd)

Proof of the proposition (cont'd).

Recall: With probability $1/2$, $\mathbb{T}'$ chooses a random i and a random f and checks that $A(f) \oplus A(f \oplus e_i) = \sigma(x_i)$.

A is $\delta/2$-close to $A_{a'}$, for all $i \in [s]$:

$$\begin{aligned} & \Pr_{f \in L}\left[A(f) \oplus A(f \oplus e_i) = a'(x_i)\right] \\ \geq & \Pr_{f \in L}\left[A(f) = f(a') \text{ and } A(f \oplus e_i) = (f \oplus e_i)(a')\right] \\ \geq & 1 - 2 \cdot \delta/2 = 1 - \delta. \end{aligned}$$

The check fails if $a'(x_i) \neq \sigma(x_i)$ and yet $A(f) \oplus A(f \oplus e_i) = a'(x_i)$. Altogether this occurs with probability $(1 - \delta) \cdot \delta \geq \delta/2$. The result follows from the fact that $\mathbb{T}'$ runs this test with probability $1/2$. $\dashv$

Proof of the Soundness (cont'd)

Proof of the Soundness (cont'd)

Consider the assignment $b \upharpoonright Z$.

Proof of the Soundness (cont'd)

Consider the assignment $b \upharpoonright Z$. For every random string that causes $\mathbb{T}'$ to reject (on input A, σ), the associated variable z_r is either assigned consistently with A, σ which means that its value immediately causes the associated constraint to reject;

Proof of the Soundness (cont'd)

Consider the assignment $b \restriction Z$. For every random string that causes $\mathbb{T}'$ to reject (on input A, σ), the associated variable z_r is either assigned consistently with A, σ which means that its value immediately causes the associated constraint to reject; or it is inconsistent with A, σ.

Proof of the Soundness (cont'd)

Consider the assignment $b \upharpoonright Z$. For every random string that causes $\mathbb{T}'$ to reject (on input A, σ), the associated variable z_r is either assigned consistently with A, σ which means that its value immediately causes the associated constraint to reject; or it is inconsistent with A, σ. Each inconsistency will be detected with probability at least $1/3$.
Thus at least $\Omega(\delta)/3$ fraction of the constraints reject.

Proof of the Soundness (cont'd)

Consider the assignment $b \upharpoonright Z$. For every random string that causes $\mathbb{T}'$ to reject (on input A, σ), the associated variable z_r is either assigned consistently with A, σ which means that its value immediately causes the associated constraint to reject; or it is inconsistent with A, σ. Each inconsistency will be detected with probability at least $1/3$.
Thus at least $\Omega(\delta)/3$ fraction of the constraints reject. Hence

$$\mathrm{unsat}_{a \cup b}(G) = \frac{\Omega(\delta)}{3}.$$

$\square$.