New insights into Probabilistically Checkable Proofs (PCPs)

Eli Ben-Sasson
Computer Science Department
Technion

WoLLIC `06, Stanford, July 2006

Talk outline

- Probabilistically checkable proofs (PCPs)
  - Definition and statement of results
  - Applications

- PCP building blocks
  - Sublinear coding theory
  - PCPs of proximity
  - Soundness preservation/amplification

NP – Efficient proof verification

x y

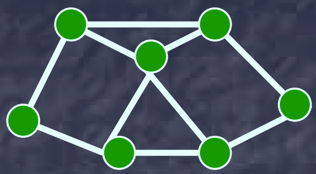
M_L $x \in L$ iff $\exists y M_L(x, y) =$accept

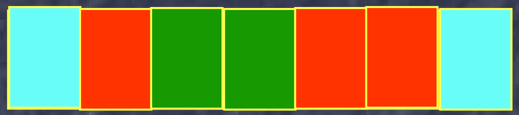
Efficiency: M_L runs in deterministic polynomial time in $|x|$

Completeness: $x \in L \Rightarrow \exists y, \quad M_L(x, y) = \text{accept}$

Soundness: $x \notin L \Rightarrow \forall y, \quad M_L(x, y) = \text{reject}$

NP – Efficient proof verification





$x \in L$ iff $\exists y M_L(x,y) =$accept

Efficiency: M_L runs in deterministic polynomial time in $|x|$

Completeness: $x \in L \Rightarrow \exists y, \quad M_L(x,y) = \text{accept}$

Soundness: $x \notin L \Rightarrow \forall y, \quad M_L(x,y) = \text{reject}$

NP – Efficient proof verification

$(x \vee y \vee \bar{z}) \wedge$

$\vdots$

$\wedge (\bar{x} \vee y \vee z)$

0 1 1 0 1 1 1

M_L

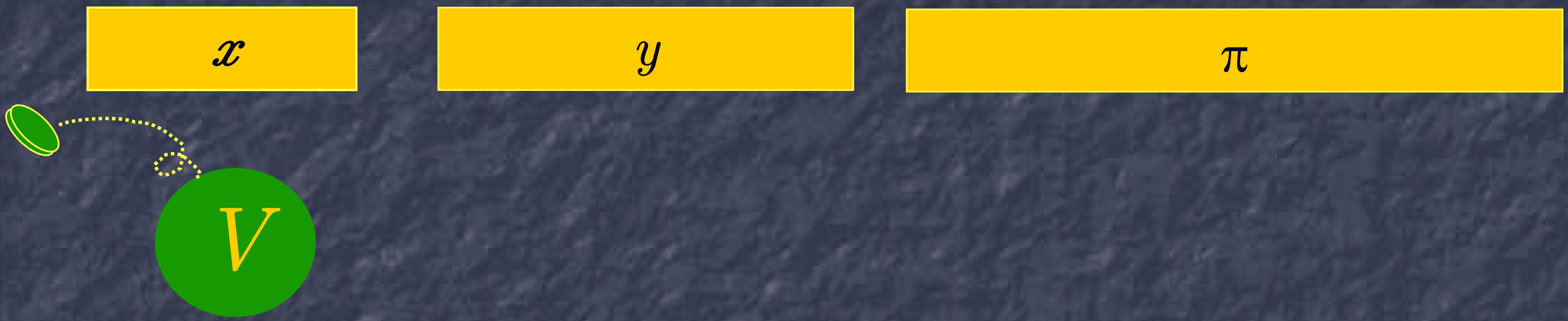
$x \in L$ iff $\exists y M_L(x, y) =$accept

Efficiency: M_L runs in deterministic polynomial time in $|x|$

Completeness: $x \in L \Rightarrow \exists y, \quad M_L(x, y) = \text{accept}$

Soundness: $x \notin L \Rightarrow \forall y, \quad M_L(x, y) = \text{reject}$

PCP – Super-Efficient Proof Verification

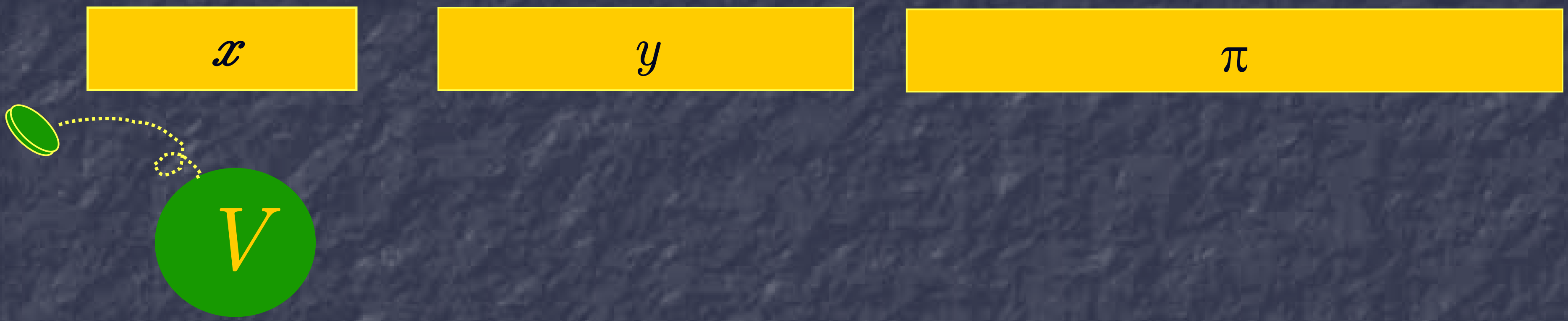


Efficiency: V runs in randomized polynomial time in $|x|$

Completeness: $x \in L \Rightarrow \exists\pi,\ \Pr[V^{\pi}(x) = \text{accept}] = 1$

Soundness: $x \notin L \Rightarrow \forall\pi,\ \Pr[V^{\pi}(x) = \text{reject}] \geq 1/2$

PCP – Super-Efficient Proof Verification



Pros

- Few queries into proof π
- Running time polylog(π)

Cons

- Errors possible
- Proofs longer

Efficiency: V runs in randomized polynomial time in $|x|$

Completeness: $x \in L \Rightarrow \exists \pi,\ \Pr[V^{\pi}(x) = \text{accept}] = 1$

Soundness: $x \notin L \Rightarrow \forall \pi,\ \Pr[V^{\pi}(x) = \text{reject}] \geq 1/2$

Definition: PCP language class

We say $L \in \mathrm{PCP}\left[\begin{array}{ll|ll} \text{time} & \leq t(n) & & \\ \text{length} & \leq l(n) & \text{comp.} & \geq c(n) \\ & & \text{sound.} & \geq s(n) \\ \text{query} & \leq q(n) & & \end{array}\right]$

If there exists verifier $V = V_L$ that on input x, $|x|=n$, runs in time $t(n)$, makes $q(n)$ quries to a proof of length $l(n)$, such that:

Completeness: $x \in L \Rightarrow \exists \pi,\ \Pr[V^{\pi}(x) = \text{accept}] \geq c(n)$

Soundness: $x \notin L \Rightarrow \forall \pi,\ \Pr[V^{\pi}(x) = \text{reject}] \geq s(n)$

PCP Theorems

Thm: $\mathbf{NP} \subseteq \mathrm{PCP}\left[\begin{array}{lll} \text{time} & \leq & n^{O(1)} \\ \text{length} & \leq & n^{O(1)} \\ \text{query} & \leq & O(1) \end{array} \middle| \begin{array}{lll} \text{comp.} & \geq & 1 \\ \text{sound.} & \geq & 1/2 \end{array}\right]$

Two settings, two applications:

- Hardness of approximation [FGL+91]

PCP Theorems

Thm: $\mathbf{NP} \subseteq \mathrm{PCP}\left[\begin{array}{ll} \text{time} & \leq \textit{polylog } n \\ \text{length} & \leq n^{O(1)} \\ \text{query} & \leq t(n) \end{array} \quad \begin{array}{ll} \text{comp.} & \geq 1 \\ \text{sound.} & \geq 1/2 \end{array}\right]$

Two settings, two applications:

- Hardness of approximation [FGL+91]
- Super-efficient proof/computation verification [BFL+91]

PCPs and Hardness of approximation [FGL+91]

Example [Hås97]:

Thm: $\mathbf{NP} \subseteq \mathrm{PCP}$ $\left[\begin{array}{lll} \text{time} & \le & n^{O(1)} \\ \text{length} & \le & n^{O(1)} \\ \text{query} & \le & 3 \text{ bits} \end{array} \middle| \begin{array}{lll} \text{comp.} & \ge & 1\text{-}\varepsilon \\ \text{sound.} & \ge & 1/2\text{-}\varepsilon \end{array} \right]$

V computes XOR of 3 answer bits

List all possible verifier tests:

$$y_1 \oplus y_2 \oplus y_3 = 1$$

$$y_3 \oplus y_5 \oplus y_{20} = 0$$

$$\vdots$$

Completeness: $x \in L$: Exists y satisfying $1\text{-}\varepsilon$ fraction of constraints

Soundness: $x \notin L$: Every y satisfies $\le 1/2\text{-}\varepsilon$ frac. of constraints

Corollary: NP-hard to 2-approximate MAX3LIN.

NP-hard to 8/7-approximate MAX3SAT.

PCPs and Hardness of approximation [FGL+91]

Thm: $$\mathbf{NP} \subseteq \mathrm{PCP}\left[\begin{array}{lll} \text{time} & \leq & n^{O(1)} \\ \text{length} & \leq & n^{O(1)} \\ \text{query} & \leq & O(1) \end{array} \middle| \begin{array}{lll} \text{comp.} & \geq & 1 \\ \text{sound.} & \geq & 1/2 \end{array}\right]$$

- Many hardness of approximation results
  - [Hås96] Clique $n^{1-\varepsilon}$
  - [Hås97] MAX3SAT $8/7 - \varepsilon$
  - [Hås97] MAXCUT 17/16
  - [Fei98] Set Cover $(1- \varepsilon) \ln n$
  - [DR02] Vertex cover 1.36
  - …

PCPs and super-efficient verification [BFL+91]

Thm [BS05; BGH+05]: **NTIME**($f(n)$) $\subseteq$

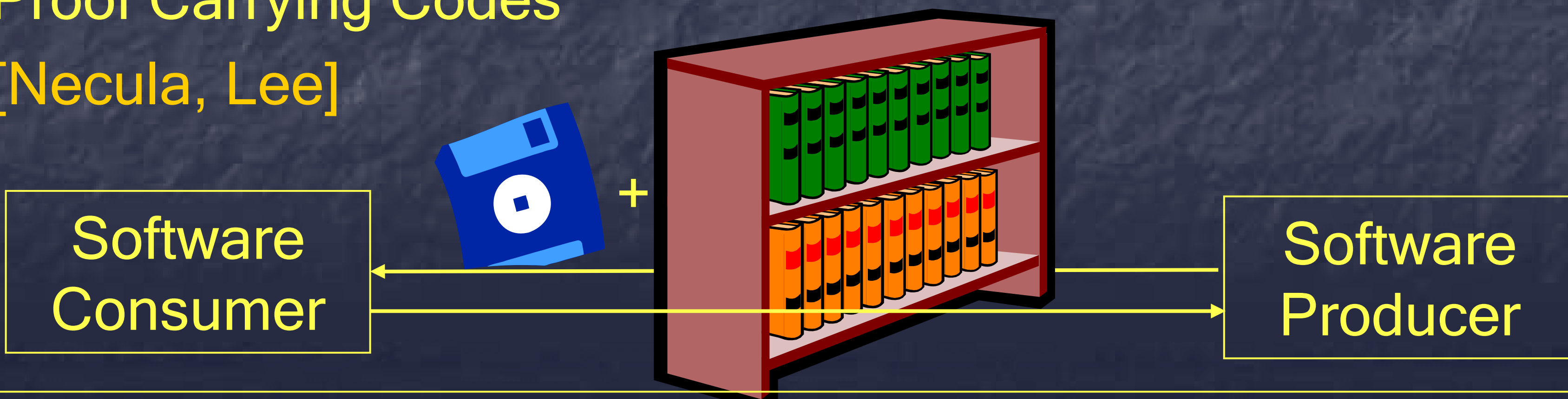
$$\mathrm{PCP}\left[\begin{array}{lll} \text{time} & \leq & f^{O(1)}(n) \\ \text{length} & \leq & f(n)\cdot \mathrm{polylog}\, f(n) \\ \text{query} & \leq & \mathrm{polylog}\, f(n) \end{array} \;\middle|\; \begin{array}{ll} \text{comp.} & \geq 1 \\ \text{sound.} & \geq 1/2 \end{array}\right]$$

- Not enough time to read input x (!)
- Settle for approximate soundness:

  If input x is ~~not in~~ far (in Hamming distance) from L, then V rejects.

Proof Carrying Codes

[Necula, Lee]



PCPs and super-efficient verification [BFL+91]

Thm [BS05; BGH+05]: **NTIME**($f(n)$) $\subseteq$

$$\text{PCP}\left[\begin{array}{lll} \text{time} & \leq & f^{O(1)}(n) \\ \text{length} & \leq & f(n) \cdot \text{polylog} f(n) \\ \text{query} & \leq & \text{polylog} f(n) \end{array} \;\middle|\; \begin{array}{ll} \text{comp.} & \geq 1 \\ \text{sound.} & \geq 1/2 \end{array}\right]$$

- Not enough time to read input x (!)
- Settle for approximate soundness:

  If input x is ~~not in~~ far (in Hamming distance) from L, then V rejects.

Proof Carrying Codes

[Necula, Lee]

\+

Software Consumer

Software Producer

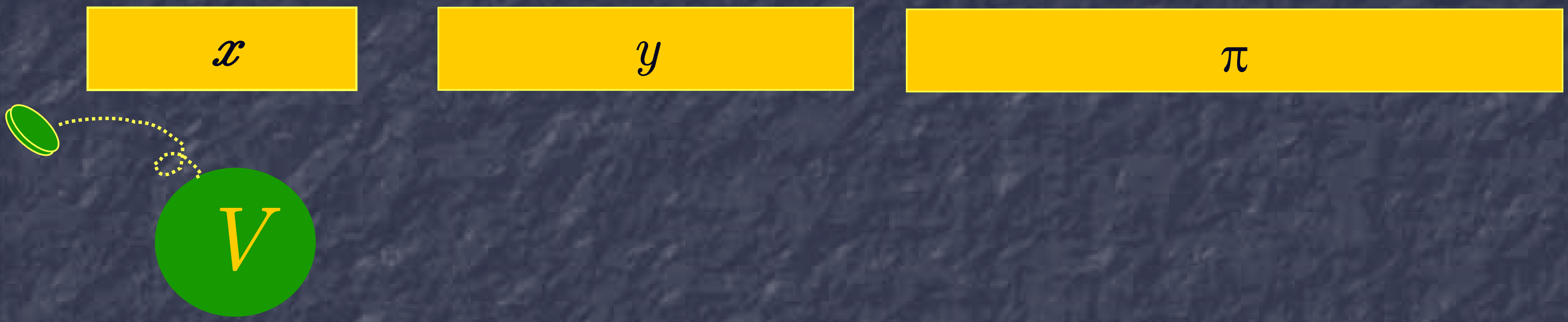
[Kil92], [Mic94]

Talk outline

- ✓ Probabilistically checkable proofs (PCPs)
  - ✓ Definition and statement of results
  - ✓ Applications

- PCP building blocks
  - Sublinear coding theory
  - PCPs of proximity
  - Soundness preservation/amplification

PCP Blueprint



- Want to verify that y witnesses x is in L
- Encode y, "spreading" its information. Minimal requirements from code:
  - Locally testable
  - Locally decodable
- Problem: Too many queries/too little soundness
- Solution: Proof composition

Error Correcting Codes

Encoding: E:{0,1}$^k \to$ {0,1}^{n}, C={E(m): m in {0,1}^{k}}

Rate = k/n, blowup = 1/rate

Code space={$0,1$}^{n}

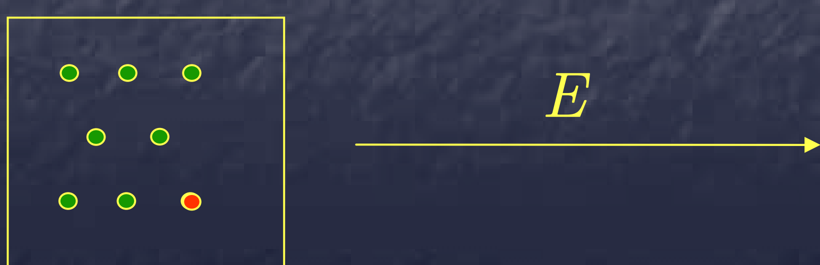
Distance:

$$\delta(x,y) = \Pr_{i\in[n]}[x_i \neq y_i]$$

$$\delta(C) = \min_{x\neq y\in C}\{\delta(x,y)\}$$

$$\delta_C(w) = \min_{x\in C}\{\delta(w,x)\}$$

Message space={$0,1$}^{k}



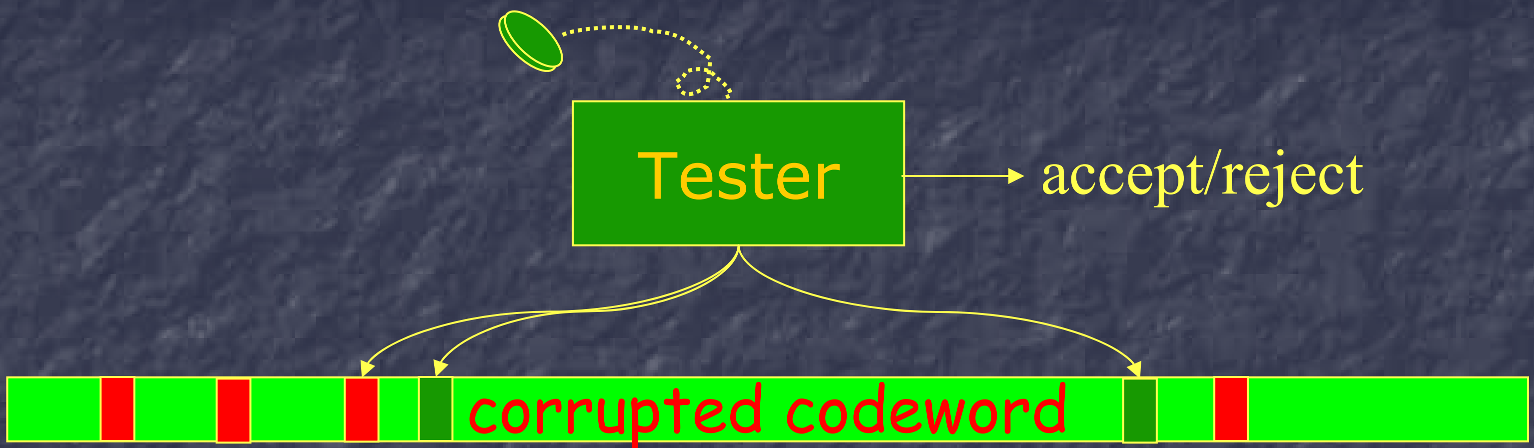
Sub-linear coding algorithms

Running time $= o(n)$, typically $\mathrm{poly}(\log n)$

n bits

- Want "good" code (large rate and distance) s.t.
  - ~~Sub-linear time for encoding~~ i^{th} bit
  - Sub-linear distance estimation
    locally testable code (LTC)
  - Sub-linear decoding of one message-bit
    locally decodable code (LDC)

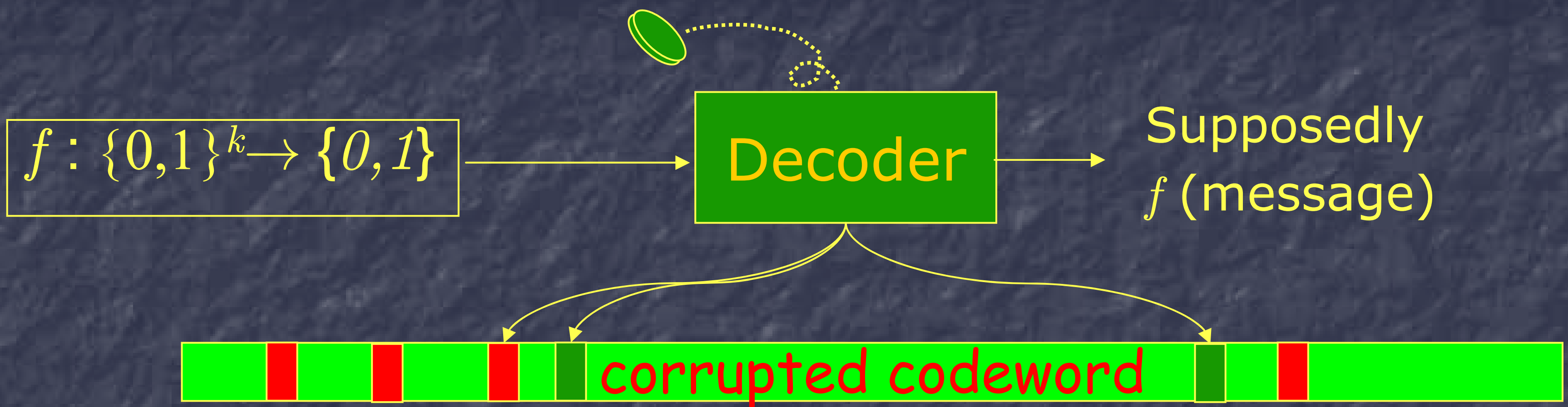
Locally Testable Code



- $t(n)=o(n)$, think of polylog n
- $q(n)=o(n)$, think of $O(1)$
- Comp. : $w \in C \Rightarrow$ Pr[Tester^w=accept] = 1
- Sound.: $\delta_C(w) > \delta_0 \Rightarrow$ Pr[Tester^w=reject] > .99

Def: Implicit in [BFL+91], explicit in [Aro94; Spi95; FS95]

Locally Decodable Code



- Let F be family of Boolean functions on k bits
- F is loc. dec. from E if $t(n)$, $q(n)=o(n)$ and for all f in F,
  Comp.: $\delta(w,E(m))<\delta_0 \Rightarrow \Pr[\text{Dec.}^w(f)=f(m)] \geq .99$
- Remark: No soundness requirement

Def: Implicit in [BFL+91; Sud92], explicit in [KT00]

LTCs and LDCs – brief comparison

- Applications (other than PCPs and coding theory)
  - LTCs: Property testing
  - LDCs: Derandomization, Cryptography, Private Information Retrieval

- Rate comparison for $q=O(1)$
  - LTCs: $n = k \cdot \text{polylog } k$ [BS05;Din06]
  - LDCs: $n = \exp(k^{\varepsilon})$ [BIK+02]

LTCs – results

- Positive (constructions)
  - Hadamard codes [BLR90; BCH+96]
  - Reed-Muller codes [BFL+91; ALM+92; AS97; RS97 …]
  - Derandomized Hadamard/Reed-Muller testers [GS02; BSV+03; BGH+04; SW04; BS05; RM06]
  - Tensor codes [BS04; DSW06]
- Negative (lower bounds)
  - $q=2$ [BGS03]
  - LDPC expander codes [BHR03]
  - Cyclic codes [BSS05]
  - Two-wise tensor [Val05; CR05]
  - Very little known…

LDCs - results

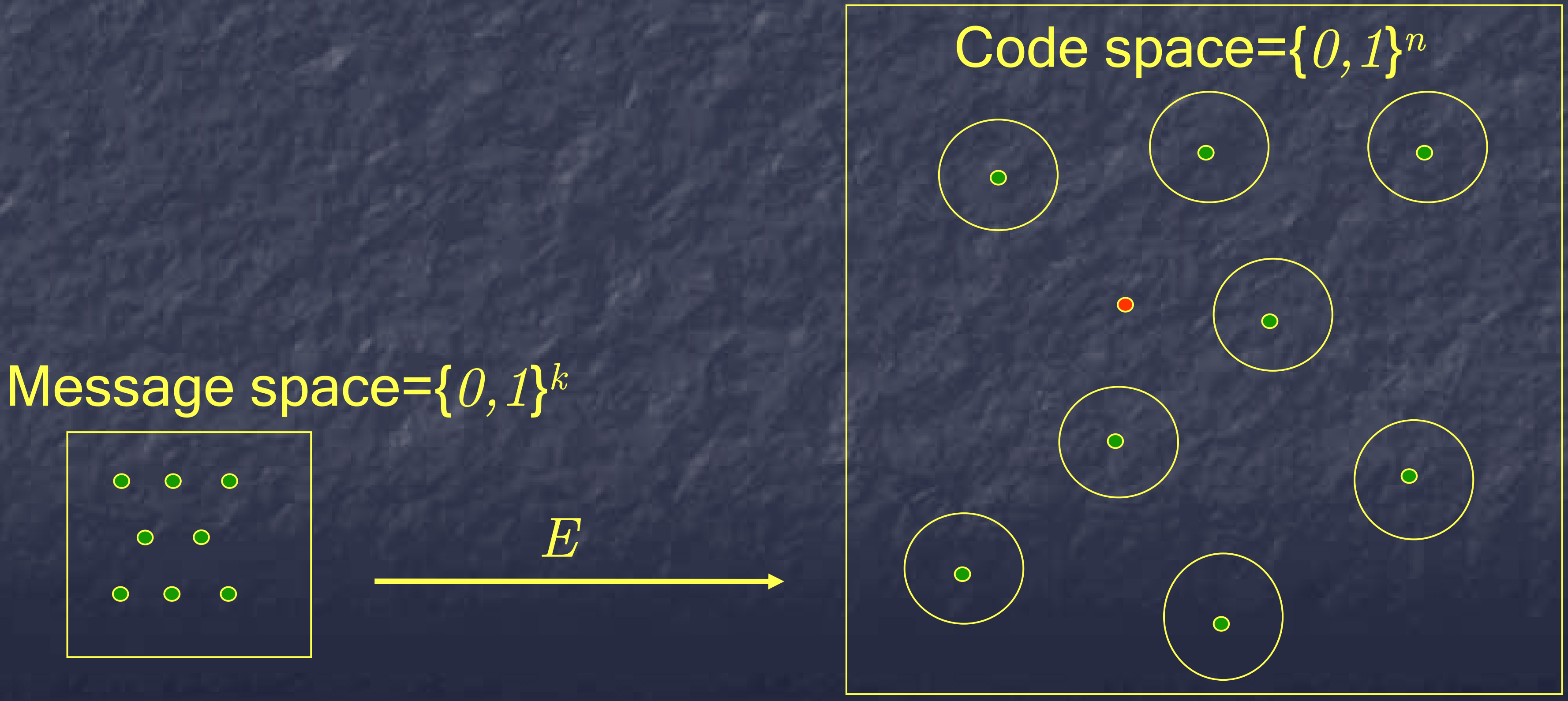
- Positive (lower bounds)
  - Hadamard codes [BLR90]
  - Reed-Muller codes [BF90]
  - Improvements [Amb97; IK99; BI01; BIKR02]
- Negative (lower bounds)
  - [Man98; KT00; GKS+02; Oba02]
  - Exponential lower bounds for $q=2$ [KdW03]
  - Very little known …

LTCs, LDCs and PCP Blueprint

Given x as input, request $E(y)$, where

- E is Locally testable
- "Interesting" F is locally decodable from E

Use F to locally test that y witnesses x is in L



Example: Hadamard-Walsh based PCP

Given x as input, request $E(y)$, where

- E is Locally testable
- "Interesting" F is locally decodable from E

Use F to locally test that y witnesses x is in L

E is a LTC, with 3 queries [BLR90]

Every linear function is Loc. Dec. from E, with 2 queries

Verifying x is in L can be reduced to decoding a constant number of linear functions [ALM+91]

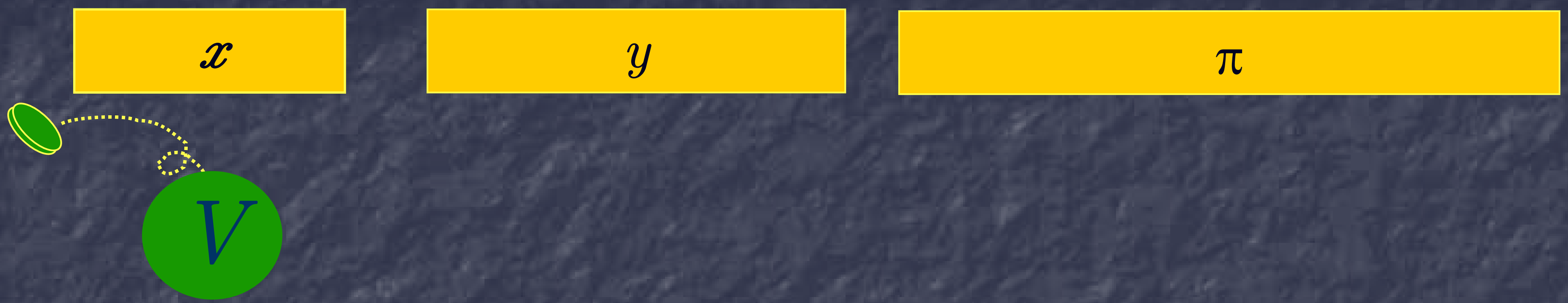
Problem: rate... $E : \{0,1\}^k \to \{0,1\}^{2^k}$

$$\underbrace{\begin{pmatrix}0\\1\\0\\1\\1\\0\\1\\0\\1\\0\\1\\0\\0\\1\\0\\1\end{pmatrix}}_{E(m)} = \begin{pmatrix}0&0&0&0\\0&0&0&1\\0&0&1&0\\0&0&1&1\\0&1&0&0\\0&1&0&1\\0&1&1&0\\0&1&1&1\\1&0&0&0\\1&0&0&1\\1&0&1&0\\1&0&1&1\\1&1&0&0\\1&1&0&1\\1&1&1&0\\1&1&1&1\end{pmatrix} \cdot \underbrace{\begin{pmatrix}1\\1\\0\\1\end{pmatrix}}_{m}$$

Talk outline

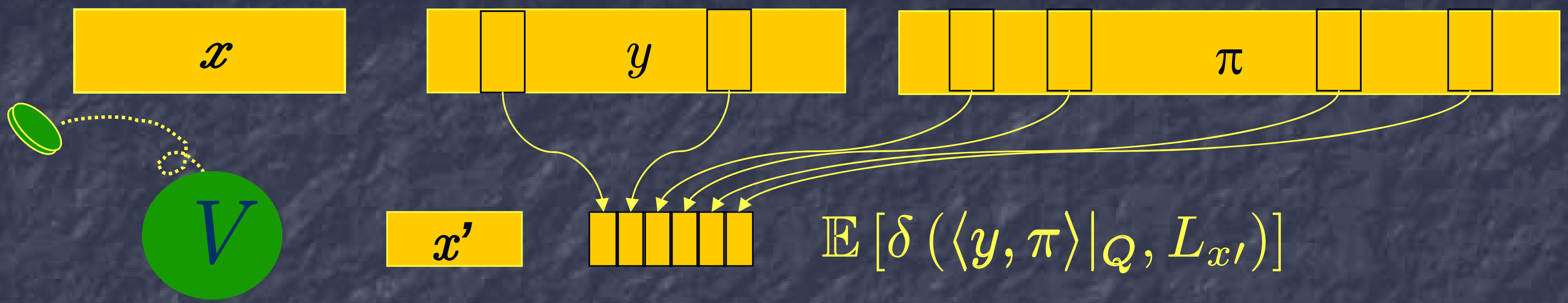
- ✓ Probabilistically checkable proofs (PCPs)
  - ✓ Definition and statement of results
  - ✓ Applications
- PCP building blocks
  - ✓ Sublinear coding theory
  - PCPs of proximity
  - Soundness preservation/amplification

Proof Composition [AS91]



- Problems
  - If $q(n) = O(1)$, $s(n)=1/2$, then $l(n)=\exp(n^2)$
  - If $l(n)=\text{poly}(n)$, $q(n)=O(1)$, then $s(n)=1/n$
  - If $l(n)=\text{poly}(n)$, $s(n)=1/2$, then $q(n)=\text{polylog}(n)$
- Solution
  - Proof composition

PCPs of Proximity / Assignment testers [BGH+05; DR05]

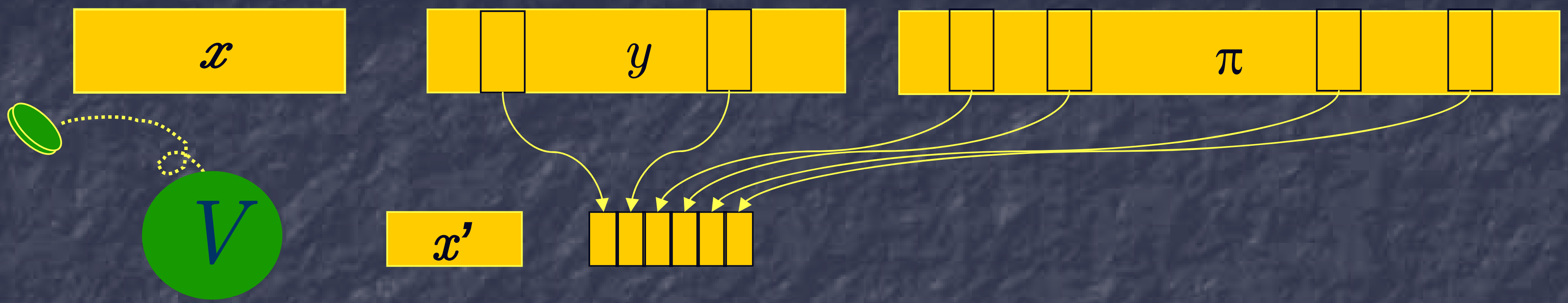


Let $L_2 = \{(x,y) : M_L(x,y) = \text{accept}\}$

Let $L_x = \{y : M_L(x,y) = \text{accept}\}$

A PCPP-verifier V verifies that y is close to L_x

PCPs of Proximity / Assignment testers [BGH+05; DR05]



Definition:

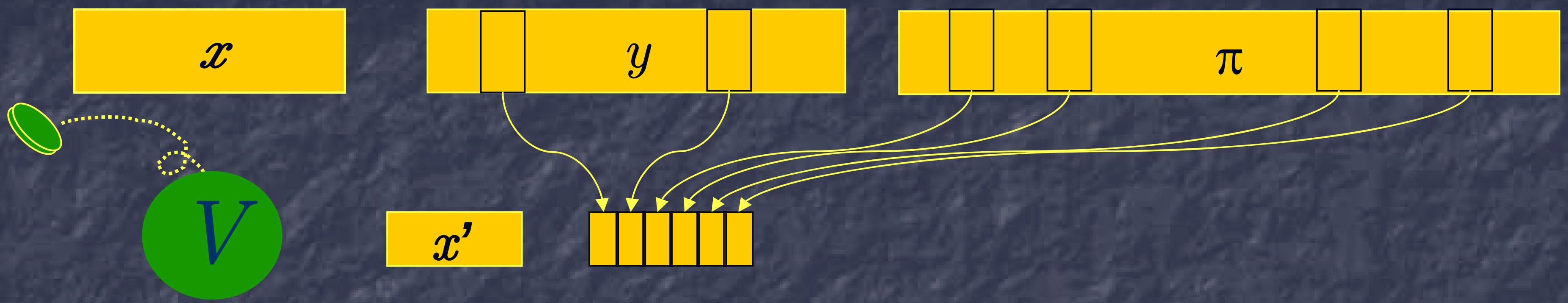
We say $L_2 \in \mathrm{PCPP}\left[\begin{array}{ll|ll} \text{time} & \leq t(n) & & \\ \text{length} & \leq l(n) & \text{comp.} & = 1 \\ \text{query} & \leq q(n) & \text{sound.} & \geq .99 \end{array}\right]$

If there exists a nonadaptive PCPP verifier V running in time $t(n)$, making $q(n)$ quries to a proof of length $l(n)$, such that:

Completeness: $y \in L_x \Rightarrow \exists \pi \; \mathbb{E}\left[\delta\left(\langle y, \pi\rangle|_Q, L_{x'}\right)\right] = 0$

Robust Soundness: $\forall \pi \; \mathbb{E}\left[\delta\left(\langle y, \pi\rangle|_Q, L_{x'}\right)\right] \geq 0.99 \cdot \delta(y, L_x)$

PCPs of Proximity / Assignment testers [BGH+05; DR05]



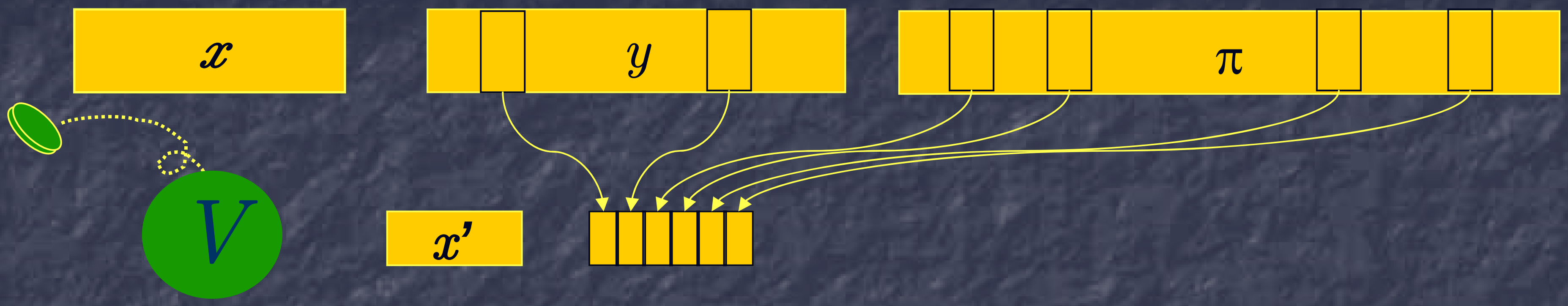
Theorem [BS05; Din06]: If $L \in \mathbf{NTIME}(f(n))$, then

$$L_2 \in \mathrm{PCPP}\left[\begin{array}{ll}\text{time} & \leq f^{O(1)}(n) \\ \text{length} & \leq f(n)\cdot \mathrm{polylog}\, f(n) \\ \text{query} & \leq O(1)\end{array}\;\middle|\; \begin{array}{ll}\text{comp.} & = 1 \\ \text{sound.} & \geq .99\end{array}\right]$$

Completeness: $y \in L_x \Rightarrow \exists \pi\ \mathbb{E}\left[\delta\left(\langle y, \pi\rangle|_Q, L_{x'}\right)\right] = 0$

Robust Soundness: $\forall \pi\ \mathbb{E}\left[\delta\left(\langle y, \pi\rangle|_Q, L_{x'}\right)\right] \geq 0.99 \cdot \delta(y, L_x)$

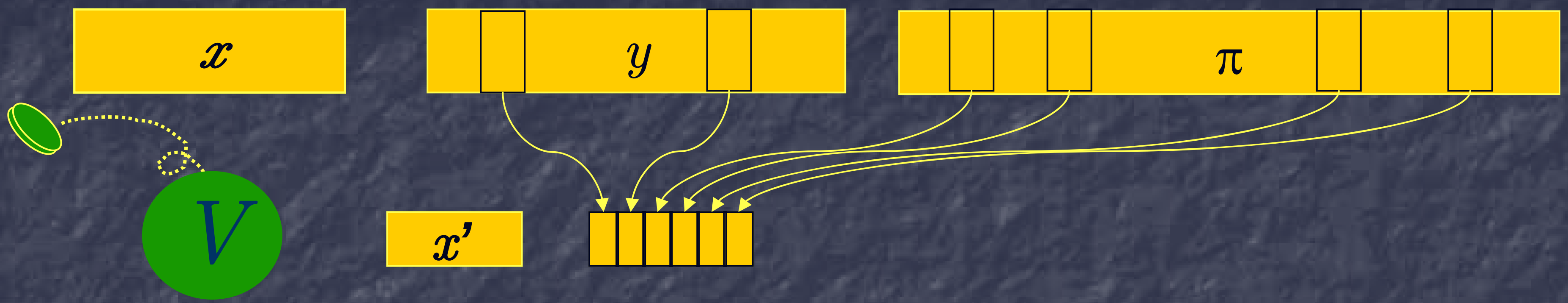
PCPs of Proximity / Assignment testers [BGH+05; DR05]



PCPPs - History

- Holographic proofs - PCPPs where assignment y is encoded. [BFL+91]
- PCPP - implicit in low-degree tests [RS92; ALM+91]
- PCPPs - special case of “PCP Spot Checkers” [EKR99]
- PCPP – extension of Property Testing [RS92; GGR96]

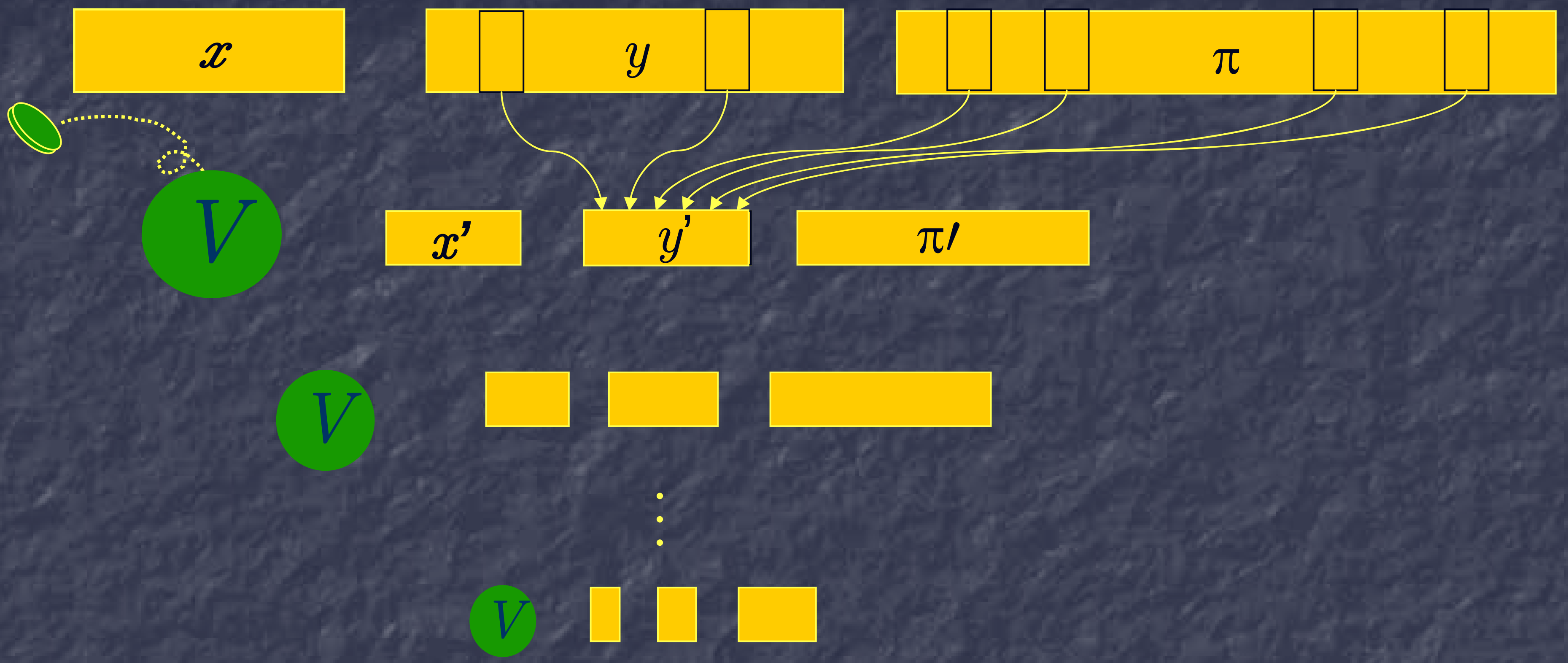
PCPs of Proximity / Assignment testers [BGH+05; DR05]



Applications of PCPPs

- PCPPs yield PCPs
- Simpler proof composition, essential in
  - Shorter PCPs [BGH+05; BS05; BGH+06]
  - PCPs via gap amplification [Din06]
- Coding
  - Locally Testable Codes [GS02;BSV+03;BGH+05…]
  - Relaxed Locally Decodable Codes [BGH05+]
- Property testing
  - Every property is locally testable (with a little help)
  - Lower bounds for tolerant testing [FF05]

PCPP Composition



Completeness: $y \in L_x \Rightarrow \exists \pi \ \mathbb{E}\left[\delta\left(\langle y, \pi \rangle|_Q, L_{x'}\right)\right] = 0$

Soundness: $\forall \pi \ \mathbb{E}\left[\delta\left(\langle y, \pi \rangle|_Q, L_{x'}\right)\right] \geq 0.99 \cdot \delta(y, L_x)$

Talk outline

- ✓ Probabilistically checkable proofs (PCPs)
  - ✓ Definition and statement of results
  - ✓ Applications

- PCP building blocks
  - ✓ Sublinear coding theory
  - ✓ PCPs of proximity
  - Soundness preservation/amplification

Putting it all together

- Algebraic approach
  - Encode using LTCs/LDCs based on polynomials, specifically, Reed-Solomon and Reed-Muller codes
  - Large q, large s
  - PCPP Composition to reduce q, while preserving s

- Expander-based approach [Din06]
  - Constant q, small s
  - Randomness-efficient repetition to boost s (but q also increases)
  - Encode using simple, rate-inefficient LTCs/LDCs
  - PCPP Composition to reduce q, while preserving s

PCP via gap amplification [Din06]

Gap amplification: There exists $c>0$ s.t. for $s(n)<c$,

$$\mathrm{PCP}\left[\begin{array}{ll|ll} \text{time} & \le t(n) & \text{comp.} & \ge 1 \\ \text{length} & \le l(n) & \text{sound.} & \ge s(n) \\ \text{query} & \le 2 & & \end{array}\right] \subseteq$$

$$\mathrm{PCP}\left[\begin{array}{ll|ll} \text{time} & \le O(t(n)) & \text{comp.} & \ge 1 \\ \text{length} & \le O(l(n)) & \text{sound.} & \ge 2 \cdot s(n) \\ \text{query} & \le 2 & & \end{array}\right]$$

Proof of PCP Theorem:

$$\mathbf{NP} \subseteq \mathrm{PCP}\left[\begin{array}{ll|ll} \text{time} & \le n^{O(1)} & \text{comp.} & \ge 1 \\ \text{length} & \le n^{O(1)} & \text{sound.} & \ge 1/n \\ \text{query} & \le 2 & & \end{array}\right]$$

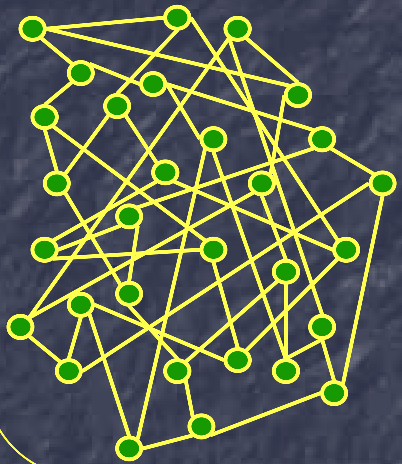
Apply gap amplification log n times...

$$\subseteq \mathrm{PCP}\left[\begin{array}{ll|ll} \text{time} & \le n^{O(1)} & \text{comp.} & \ge 1 \\ \text{length} & \le n^{O(1)} & \text{sound.} & \ge c \\ \text{query} & \le 2 & & \end{array}\right]$$

QED

PCP via gap amplification [Din06]

Gap amplification: There exists $c > 0$ s.t. for $s(n) < c$,



$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \leq t(n) \\ \text{length} & \leq l(n) \\ \text{query} & \leq 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \geq 1 \\ \text{sound.} & \geq s(n) \end{array}\right] \subseteq$$

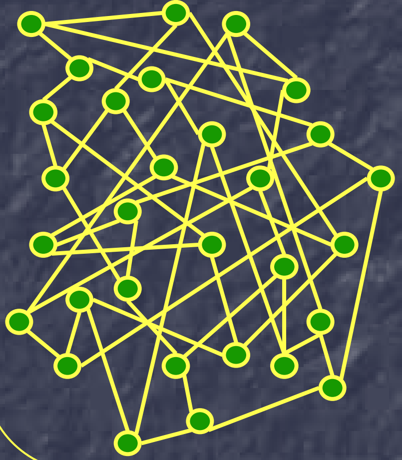
$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \leq O(t(n)) \\ \text{length} & \leq O(l(n)) \\ \text{query} & \leq 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \geq 1 \\ \text{sound.} & \geq 2 \cdot s(n) \end{array}\right]$$

Constraint graph

- Vertices: Proof symbols
- Edges: constraints over pair of queries
- $x \in L \Rightarrow$ All constraints can be satisfied
- $x \notin L \Rightarrow$ At least $s(n)$ frac. of constraints reject

PCP via gap amplification [Din06]

Gap amplification: There exists $c>0$ s.t. for $s(n)<c$,



$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \le t(n) \\ \text{length} & \le l(n) \\ \text{query} & \le 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \ge 1 \\ \text{sound.} & \ge s(n) \end{array}\right] \subseteq$$

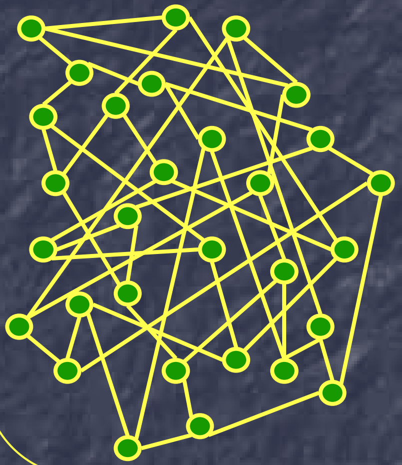
$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \le O(t(n)) \\ \text{length} & \le O(l(n)) \\ \text{query} & \le 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \ge 1 \\ \text{sound.} & \ge 2 \cdot s(n) \end{array}\right]$$

Boosting soundness – 1st attempt

- Query 100 edges (sequential repetition)
- $x \in L \Rightarrow$ all constraints can be satisfied
- $x \notin L \Rightarrow$ at least $10s(n)$ frac. of constraints reject
- Problem: q is large

PCP via gap amplification [Din06]

Gap amplification: There exists $c>0$ s.t. for $s(n)<c$,



$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \le t(n) \\ \text{length} & \le l(n) \\ \text{query} & \le 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \ge 1 \\ \text{sound.} & \ge s(n) \end{array}\right] \subseteq$$

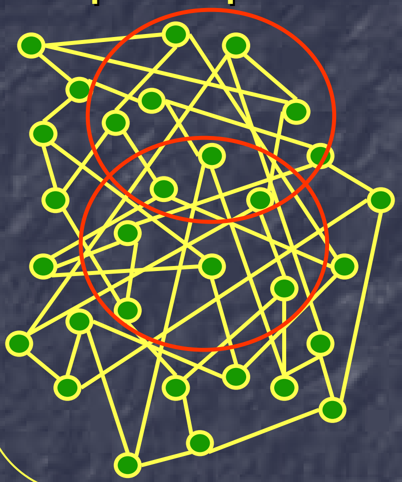
$$\mathrm{PCP}\left[\begin{array}{ll} \text{time} & \le O(t(n)) \\ \text{length} & \le O(l(n)) \\ \text{query} & \le 2 \end{array}\middle|\begin{array}{ll} \text{comp.} & \ge 1 \\ \text{sound.} & \ge 2 \cdot s(n) \end{array}\right]$$

Boosting soundness – 2nd attempt

- Encode ass. to every 100-tuple of vertices using LDC/LTC
- Pick 100 edges, make 2 queries to get ass. to endpoints
- Use PCPPs to prove codewords satisfy all constraints
- $q=2$, $c=1$, sound. > $9s(n)$
- Problems: (1) $l=n^{100}$, (2) consistency

PCP via gap amplification [Din06]

Gap amplification: There exists $c>0$ s.t. for $s(n)<c$,



$$\mathrm{PCP}\left[\begin{array}{ll}\text{time} & \leq t(n)\\ \text{length} & \leq l(n)\\ \text{query} & \leq 2\end{array}\middle|\begin{array}{ll}\text{comp.} & \geq 1\\ \text{sound.} & \geq s(n)\end{array}\right] \subseteq$$

$$\mathrm{PCP}\left[\begin{array}{ll}\text{time} & \leq O(t(n))\\ \text{length} & \leq O(l(n))\\ \text{query} & \leq 2\end{array}\middle|\begin{array}{ll}\text{comp.} & \geq 1\\ \text{sound.} & \geq 2\cdot s(n)\end{array}\right]$$

Boosting soundness – 3rd (final) attempt

- W.l.o.g. G is constant degree regular expander graph
- Encode assignment to ball of radius 100 around every v using LDC/LTC
- Pick u,v at distance 150, query balls around u,v
- Use PCPPs to prove balls agree and satisfy intersection
- $q=2$, $c=1$, sound. > $4s(n)$, $l=O(n)$ (deg(G)=$O(1)$)
- Problem: consistency. Solution: G is an expander… QED

Summing up

- PCPs are fundamental computational objects used in:
  - Hardness of approximation
  - Super-efficient verification of proofs
- Main building blocks:
  - Locally testable and decodable codes
  - PCPP composition
  - Soundness amplification/preservation
- Open question:

$$NP \overset{?}{\subseteq} \mathrm{PCP}\left[\begin{array}{ll} \text{time} & \leq n^{O(1)} \\ \text{length} & \leq n \log^{O(1)} n \\ \text{query} & \leq 3 \text{ bits} \end{array} \middle| \begin{array}{ll} \text{comp.} & \geq 1-\epsilon \\ \text{sound.} & \geq 1/2-\epsilon \end{array}\right]$$

New insights into Probabilistically Checkable Proofs (PCPs)

Eli Ben-Sasson
Computer Science Department
Technion

Thank you

WoLLIC `06, Stanford, July 2006