Average-Case Fine-Grained Hardness

Marshall Ball   Alon Rosen   Manuel Sabin   Prashant Nalini Vasudevan

Average-Case Fine-Grained Hardness

Average-Case **Fine-Grained** Hardness

Average-Case Fine-Grained Hardness

- 3SUM

Average-Case **Fine-Grained** Hardness

- 3SUM
- APSP

Average-Case **Fine-Grained** Hardness

- 3SUM
- APSP
- Orthogonal Vectors

Average-Case Fine-Grained Hardness

Average-Case Fine-Grained Hardness

Average-Case Fine-Grained Hardness

- Natural object of study

Average-Case Fine-Grained Hardness

- Natural object of study
- Necessary for cryptography

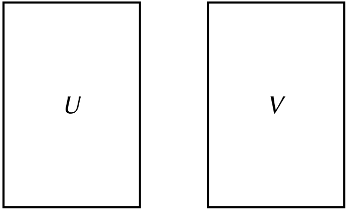
Average-Case Fine-Grained Hardness

- Natural object of study
- Necessary for cryptography
- Potential use in algorithm design

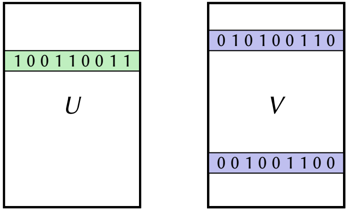
Plan

- Introduce problems
- Present average-case reduction
- Summarise
- Present Proof of Work
- ???
- Profit.

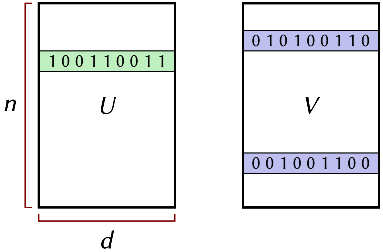
Worst-Case: Orthogonal Vectors



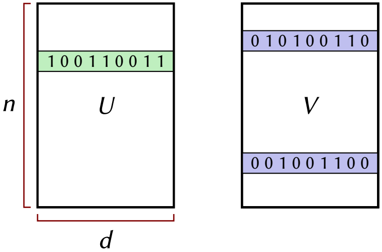
Worst-Case: Orthogonal Vectors



Worst-Case: Orthogonal Vectors

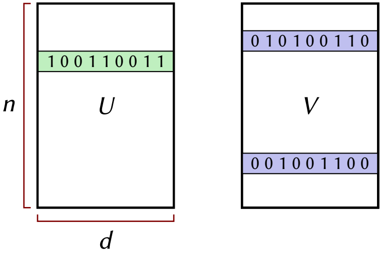


Worst-Case: Orthogonal Vectors



$\exists u \in U, v \in V$: disjoint?

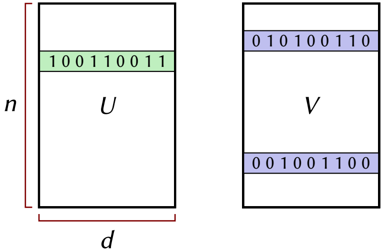
Worst-Case: Orthogonal Vectors



$\exists u \in U, v \in V$: disjoint?

Best known worst-case algorithm [AWY15]: $O(n^{2-1/O(\log(d/\log n))})$

Worst-Case: Orthogonal Vectors



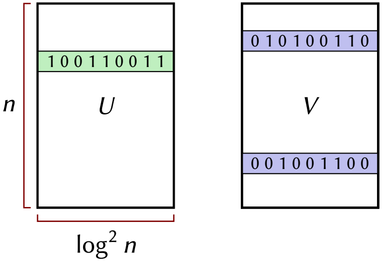
$\exists u \in U, v \in V$: disjoint?

Best known worst-case algorithm [AWY15]: $O(n^{2-1/O(\log(d/\log n))})$

OV Conjecture (implied by SETH [Wil05])

If $d = \omega(\log n)$, OV takes $n^{2-o(1)}$ time.

Worst-Case: Orthogonal Vectors



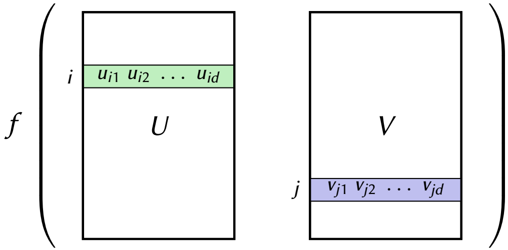
$\exists u \in U, v \in V$: disjoint?

Best known worst-case algorithm [AWY15]: $O\left(n^{2-1/O(\log(d/\log n))}\right)$

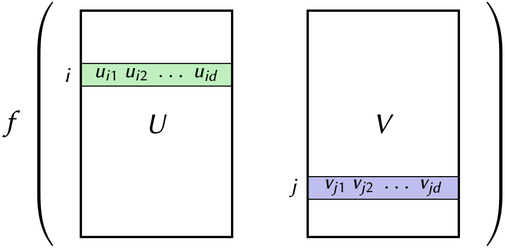
OV Conjecture (implied by SETH [Wil05])

If $d = \omega(\log n)$, OV takes $n^{2-o(1)}$ time.

Average-Case: A Polynomial for OV (independently featured in [Wil16])

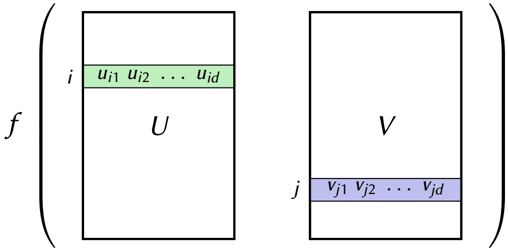


Average-Case: A Polynomial for OV (independently featured in [Wil16])



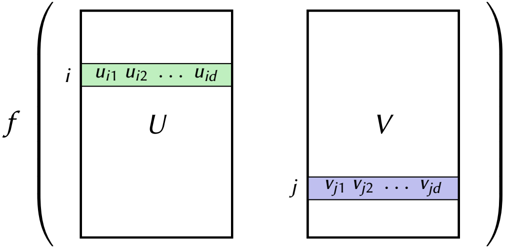
$$(1 - u_{i1}v_{j1})(1 - u_{i2}v_{j2}) \cdots (1 - u_{id}v_{jd})$$

Average-Case: A Polynomial for OV (independently featured in [Wil16])



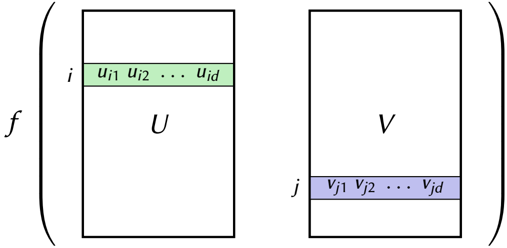
$$\overbrace{(1-u_{i1}v_{j1})(1-u_{i2}v_{j2})\cdots(1-u_{id}v_{jd})}^{1 \Leftrightarrow u_i,\, v_j \text{ disjoint}}$$

Average-Case: A Polynomial for OV (independently featured in [Wil16])



$$= \sum_{i\in[n]} \sum_{j\in[n]} \overbrace{(1 - u_{i1}v_{j1})(1 - u_{i2}v_{j2})\cdots(1 - u_{id}v_{jd})}^{1 \Leftrightarrow u_i,\, v_j \text{ disjoint}}$$

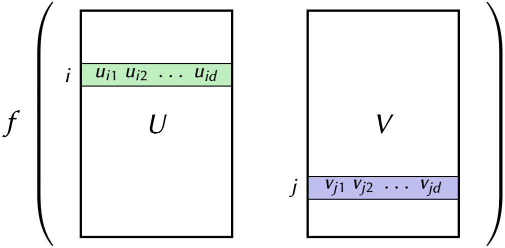
Average-Case: A Polynomial for OV (independently featured in [Wil16])



$$= \sum_{i\in[n]} \sum_{j\in[n]} \overbrace{(1 - u_{i1}v_{j1})(1 - u_{i2}v_{j2})\cdots(1 - u_{id}v_{jd})}^{1 \Leftrightarrow u_i,\ v_j \text{ disjoint}}$$

$$\boxed{\begin{array}{c} p > n^2 \\ f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p \end{array}}$$

Average-Case: A Polynomial for OV (independently featured in [Wil16])



$$= \sum_{i \in [n]} \sum_{j \in [n]} \overbrace{(1 - u_{i1} v_{j1})(1 - u_{i2} v_{j2}) \cdots (1 - u_{id} v_{jd})}^{1 \Leftrightarrow u_i,\, v_j \text{ disjoint}}$$

$$\boxed{\begin{array}{c} p > n^2 \\ f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p \end{array}}$$

$$\boxed{\begin{array}{c} deg(f) = 2d \\ d = \log^2 n \end{array}}$$

Worst-Case to Average-Case

Theorem

$$\exists A \text{ in time } n^{1+\alpha} : \Pr_{x \leftarrow \mathbb{F}_p^{2nd}}[A(x) = f(x)] \geq \frac{1}{n^{o(1)}}$$

$$\Downarrow$$

$$\exists B \text{ in time } n^{1+\alpha+o(1)} \text{ that decides OV}$$

Worst-Case to Average-Case

Theorem

$$\exists A \text{ in time } n^{1+\alpha} : \Pr_{x \leftarrow \mathbb{F}_p^{2nd}}[A(x) = f(x)] \geq \frac{1}{n^{o(1)}}$$
$$\Downarrow$$
$$\exists B \text{ in time } n^{1+\alpha+o(1)} \text{ that decides OV}$$

Corollary

OV takes $n^{2-o(1)} \Rightarrow f$ takes $n^{2-o(1)}$ on *average*

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$

Time: $t = n^{1+\alpha}$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B [B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ Time:

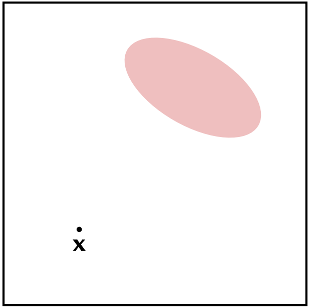
Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B [B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ Time:

$\mathbb{F}_p^{2nd}$



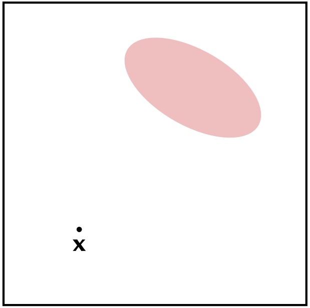
Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B [B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ | Time:

$\mathbb{F}_p^{2nd}$



$$g(t) = f(\mathbf{x} + \mathbf{y}t)$$

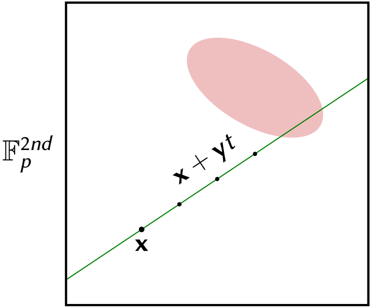
$$g(0) = f(\mathbf{x}),\ deg(g) \leq 2d$$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\qquad$ Time:



$$g(t) = f(\mathbf{x} + \mathbf{y}t)$$

$$g(0) = f(\mathbf{x}),\ deg(g) \leq 2d$$

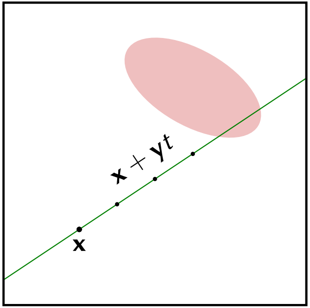
Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B [B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ Time:

$\mathbb{F}_p^{2nd}$



$g(t) = f(\mathbf{x} + \mathbf{y}t)$

$g(0) = f(\mathbf{x}),\ deg(g) \leq 2d$

Error-correct from (noisy) $g(1), g(2), \ldots, g(cd)$

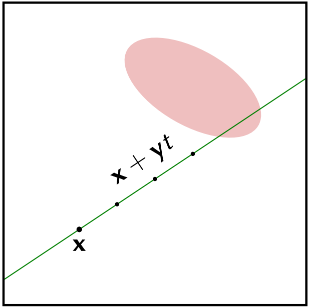
Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\longrightarrow$ Time:

$\mathbb{F}_p^{2nd}$



$g(t) = f(\mathbf{x} + \mathbf{y}t)$

$g(0) = f(\mathbf{x})$, $deg(g) \leq 2d$

Error-correct from (noisy) $g(1), g(2), \ldots, g(cd)$

$\Pr_{\mathbf{y}}[\text{too many } t\text{'s} : A(\mathbf{x} + \mathbf{y}t) \neq g(t)] < \frac{1}{3}$

(Markov Bound)

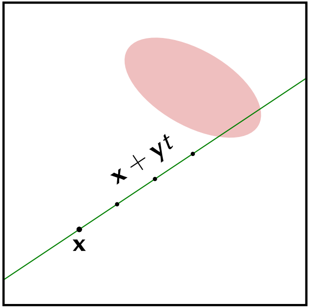
Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\longrightarrow$ Time: $\widetilde{O}(d \cdot nd + d \cdot t + d^3)$

$\mathbb{F}_p^{2nd}$



$g(t) = f(\mathbf{x} + \mathbf{y}t)$

$g(0) = f(\mathbf{x}),\ deg(g) \leq 2d$

Error-correct from (noisy) $g(1), g(2), \ldots, g(cd)$

$\Pr_{\mathbf{y}}[\text{too many } t\text{'s} : A(\mathbf{x} + \mathbf{y}t) \neq g(t)] < \frac{1}{3}$

(Markov Bound)

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\longrightarrow$ Time: $\widetilde{O}(d \cdot nd + d \cdot t + d^3)$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}} [A(\mathbf{x}) = f(\mathbf{x})] \geq 0.9$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B [B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\qquad$ Time: $\widetilde{O}(d \cdot nd + d \cdot t + d^3)$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

$$= \left(\sum_{\substack{i \in [n/2] \\ j \in [n/2]}} + \sum_{\substack{i \in [n/2] \\ j \in (n/2, n]}} + \sum_{\substack{i \in (n/2, n] \\ j \in [n/2]}} + \sum_{\substack{i \in (n/2, n] \\ j \in (n/2, n]}} \right) \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq \frac{1}{n^{o(1)}}$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\qquad$ Time: $\widetilde{O}(d \cdot nd + d \cdot t + d^3)$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

$$= \left(\sum_{\substack{i \in [n/2] \\ j \in [n/2]}} + \sum_{\substack{i \in [n/2] \\ j \in (n/2, n]}} + \sum_{\substack{i \in (n/2, n] \\ j \in [n/2]}} + \sum_{\substack{i \in (n/2, n] \\ j \in (n/2, n]}} \right) \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

Worst-Case to Average-Case (using ideas from [Lip91, GS92, CPS99])

$$f : \mathbb{F}_p^{2nd} \to \mathbb{F}_p,\ deg(f) = 2d$$

$\Pr_{\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}}[A(\mathbf{x}) = f(\mathbf{x})] \geq \frac{1}{n^{o(1)}}$ $\longrightarrow$ $\forall \mathbf{x} : \Pr_B[B(\mathbf{x}) = f(\mathbf{x})] \geq \frac{2}{3}$

Time: $t = n^{1+\alpha}$ $\qquad$ Time: $t^{1+o(1)}$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

$$= \left(\sum_{\substack{i \in [n/2] \\ j \in [n/2]}} + \sum_{\substack{i \in [n/2] \\ j \in (n/2, n]}} + \sum_{\substack{i \in (n/2, n] \\ j \in [n/2]}} + \sum_{\substack{i \in (n/2, n] \\ j \in (n/2, n]}} \right) \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell})$$

Intermediate Summary

We have a worst-to-average case reduction from OV (resp. 3SUM, APSP) to evaluating a polynomial f (other respective polynomials).

Intermediate Summary

We have a worst-to-average case reduction from OV (resp. 3SUM, APSP) to evaluating a polynomial f (other respective polynomials). In addition,

- f has low degree – $\mathrm{polylog}(n)$.
- f is somewhat efficiently computable – $\widetilde{O}(n^2)$.
- f is downward self-reducible.

Intermediate Summary

We have a worst-to-average case reduction from OV (resp. 3SUM, APSP) to evaluating a polynomial f (other respective polynomials). In addition,

- f has low degree – polylog(n).
- f is somewhat efficiently computable – $\widetilde{O}(n^2)$.
- f is downward self-reducible.

Theorem [Wil16]

There is an MA proof system for proving $(f(\mathbf{x}) = y)$ that has:

- perfect completeness and negligible soundness.
- prover complexity $\widetilde{O}(n^2)$.
- verifier complexity $\widetilde{O}(n)$.

Proof of Work

Prover	Verifier

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\mathbf{x}}$	

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\quad \mathbf{x} \quad}$	
Compute $f(\mathbf{x}) = z$ and MA proof π		
	$\xrightarrow{\quad z, \pi \quad}$	

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\quad \mathbf{x} \quad}$	
Compute $f(\mathbf{x}) = z$ and MA proof π		
	$\xrightarrow{\quad z, \pi \quad}$	
		Verify using π that $f(\mathbf{x}) = z$

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\quad \mathbf{x} \quad}$	
Compute $f(\mathbf{x}) = z$ and MA proof π		
	$\xrightarrow{\quad z, \pi \quad}$	
		Verify using π that $f(\mathbf{x}) = z$

$\widetilde{O}(n^2)$ $\qquad\qquad$ $\widetilde{O}(n)$

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\mathbf{x}}$	
Compute $f(\mathbf{x}) = z$ and MA proof π		
	$\xrightarrow{z, \pi}$	
		Verify using π that $f(\mathbf{x}) = z$

$\widetilde{O}(n^2)$ $\qquad$ $\widetilde{O}(n)$

$$\Pr\left[\text{Prover can run in } n^{2-\epsilon} \text{ and convince Verifier}\right] \leq \frac{1}{n^{\epsilon/2}}$$

Proof of Work

Prover		Verifier
		$\mathbf{x} \leftarrow \mathbb{F}_p^{2nd}$
	$\xleftarrow{\quad \mathbf{x} \quad}$	
Compute $f(\mathbf{x}) = z$ and MA proof π		
	$\xrightarrow{\quad z, \pi \quad}$	
		Verify using π that $f(\mathbf{x}) = z$
$\widetilde{O}(n^2)$		$\widetilde{O}(n)$

$$\Pr\left[\text{Prover can run in } n^{2-\epsilon} \text{ and convince Verifier}\right] \leq \frac{1}{n^{\epsilon/2}}$$

(See [DN92] for generic constructions and applications.)

What Next?

What Next?

- Average-case complexity of OV, 3SUM, etc.

What Next?

- Average-case complexity of OV, 3SUM, etc.
- Fine-grained cryptography
  - Some prior work under other assumptions [Mer78, Hås87, BGI08, DVV16, ...].
  - Fine-grained OWFs from SETH?
  - Beat Merkle's key agreement under these assumptions?

What Next?

- Average-case complexity of OV, 3SUM, etc.
- Fine-grained cryptography
  - Some prior work under other assumptions [Mer78, Hås87, BGI08, DVV16, …].
  - Fine-grained OWFs from SETH?
  - Beat Merkle's key agreement under these assumptions?
- Average-case algorithms
  - Design algorithms to evaluate polynomials that work on average.

What Next?

- Average-case complexity of OV, 3SUM, etc.
- Fine-grained cryptography
  - Some prior work under other assumptions [Mer78, Hås87, BGI08, DVV16, . . .].
  - Fine-grained OWFs from SETH?
  - Beat Merkle's key agreement under these assumptions?
- Average-case algorithms
  - Design algorithms to evaluate polynomials that work on average.
- Better reductions
  - Is it actually possible to do better than guessing at random?

To be passed in case of an abundance of time.

k-SAT and SETH

$$(\overbrace{x_1 \vee \overline{x_2} \vee \dots}^{k}) \wedge (\dots \vee x_n \vee \dots) \wedge \quad \cdots \quad \wedge (\dots \vee \dots \vee \dots)$$

k-SAT and SETH

$$(\overbrace{x_1 \vee \overline{x_2} \vee \dots}^{k}) \wedge (\dots \vee x_n \vee \dots) \wedge \quad \cdots \quad \wedge (\dots \vee \dots \vee \dots)$$

Best known worst-case algorithm [PPSZ05]: $\widetilde{O}(2^{(1-c/k)n})$

k-SAT and SETH

$$\overbrace{(\; x_1 \vee \overline{x_2} \vee \ldots \;)}^{k} \wedge (\ldots \vee x_n \vee \ldots) \wedge \quad \cdots \quad \wedge (\ldots \vee \ldots \vee \ldots)$$

Best known worst-case algorithm [PPSZ05]: $\widetilde{O}(2^{(1-c/k)n})$

Strong Exponential Time Hypothesis (SETH) [IPZ98]

$\forall \epsilon \; \exists k$: k-SAT takes $\widetilde{\Omega}(2^{(1-\epsilon)n})$ time.

An Efficient MA Protocol for f [Wil16]

$$(U, V) \in \mathbb{F}_p^{2nd}, z \in \mathbb{F}_p$$

An Efficient MA Protocol for f [Wil16]

$$(U, V) \in \mathbb{F}_p^{2nd}, z \in \mathbb{F}_p$$

$$\phi_1, \ldots, \phi_d : \mathbb{F}_p \to \mathbb{F}_p$$

$$\forall i \in [n] : \phi_\ell(i) = u_{i\ell}$$

$$deg(\phi_\ell) \leq n - 1$$

An Efficient MA Protocol for f [Wil16]

$$(U, V) \in \mathbb{F}_p^{2nd}, z \in \mathbb{F}_p$$

$$\phi_1, \ldots, \phi_d : \mathbb{F}_p \to \mathbb{F}_p$$
$$\forall i \in [n] : \phi_\ell(i) = u_{i\ell}$$
$$deg(\phi_\ell) \leq n - 1$$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell}) = \sum_{i \in [n]} \left[\sum_{j \in [n]} \prod_{\ell \in [d]} (1 - \phi_\ell(i) v_{j\ell}) \right] = \sum_{i \in [n]} r(i)$$

An Efficient MA Protocol for f [Wil16]

$$(U, V) \in \mathbb{F}_p^{2nd}, z \in \mathbb{F}_p$$

$$\phi_1, \ldots, \phi_d : \mathbb{F}_p \to \mathbb{F}_p$$
$$\forall i \in [n] : \phi_\ell(i) = u_{i\ell}$$
$$deg(\phi_\ell) \leq n - 1$$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell}) = \sum_{i \in [n]} \left[\sum_{j \in [n]} \prod_{\ell \in [d]} (1 - \phi_\ell(i) v_{j\ell}) \right] = \sum_{i \in [n]} r(i)$$

- Proof: Coefficients of r. (Interpolation – $\widetilde{O}(n^2)$)

An Efficient MA Protocol for f [Wil16]

$$(U, V) \in \mathbb{F}_p^{2nd}, z \in \mathbb{F}_p$$

$$\phi_1, \ldots, \phi_d : \mathbb{F}_p \to \mathbb{F}_p$$
$$\forall i \in [n] : \phi_\ell(i) = u_{i\ell}$$
$$deg(\phi_\ell) \leq n - 1$$

$$f(U, V) = \sum_{i \in [n]} \sum_{j \in [n]} \prod_{\ell \in [d]} (1 - u_{i\ell} v_{j\ell}) = \sum_{i \in [n]} \left[\sum_{j \in [n]} \prod_{\ell \in [d]} (1 - \phi_\ell(i) v_{j\ell}) \right] = \sum_{i \in [n]} r(i)$$

- Proof: Coefficients of r. (Interpolation – $\widetilde{O}(n^2)$)
- Verification:
  - Check r at random point. (Computation of ϕ and correct value – $\widetilde{O}(n)$)
  - Compute $r(i)$ for $i \in [n]$ and sum to get $f(U, V)$. (Batch evaluation – $\widetilde{O}(n)$)

Amir Abboud, Richard Ryan Williams, and Huacheng Yu.
More applications of the polynomial method to algorithm design.
In Piotr Indyk, editor, *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 218–230. SIAM, 2015.

Eli Biham, Yaron J. Goren, and Yuval Ishai.
Basing weak public-key cryptography on strong one-way functions.
In Ran Canetti, editor, *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, volume 4948 of *Lecture Notes in Computer Science*, pages 55–72. Springer, 2008.

Jin-yi Cai, Aduri Pavan, and D. Sivakumar.
On the hardness of permanent.
In Christoph Meinel and Sophie Tison, editors, *STACS 99, 16th Annual Symposium on Theoretical Aspects of Computer Science, Trier, Germany, March 4-6, 1999, Proceedings*, volume 1563 of *Lecture Notes in Computer Science*, pages 90–99. Springer, 1999.

Cynthia Dwork and Moni Naor.

Pricing via processing or combatting junk mail.
In *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, pages 139–147, 1992.

Akshay Degwekar, Vinod Vaikuntanathan, and Prashant Nalini Vasudevan.
Fine-grained cryptography.
In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part III*, pages 533–562, 2016.

Peter Gemmell and Madhu Sudan.
Highly resilient correctors for polynomials.
Information processing letters, 43(4):169–174, 1992.

Johan Håstad.
One-way permutations in NC^0.
Information Processing Letters, 26(3):153–155, 1987.

Russell Impagliazzo, Ramamohan Paturi, and Francis Zane.

Which problems have strongly exponential complexity?
In *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*, pages 653–663. IEEE Computer Society, 1998.

Richard Lipton.
New directions in testing.
Distributed Computing and Cryptography, 2:191–202, 1991.

Ralph C. Merkle.
Secure communications over insecure channels.
Commun. ACM, 21(4):294–299, 1978.

Ramamohan Paturi, Pavel Pudlák, Michael E. Saks, and Francis Zane.
An improved exponential-time algorithm for k-sat.
J. ACM, 52(3):337–364, May 2005.

Ryan Williams.
A new algorithm for optimal 2-constraint satisfaction and its implications.
Theor. Comput. Sci., 348(2-3):357–365, 2005.

Ryan Williams.

Strong ETH breaks with merlin and arthur: Short non-interactive proofs of batch evaluation.
In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 2:1–2:17, 2016.