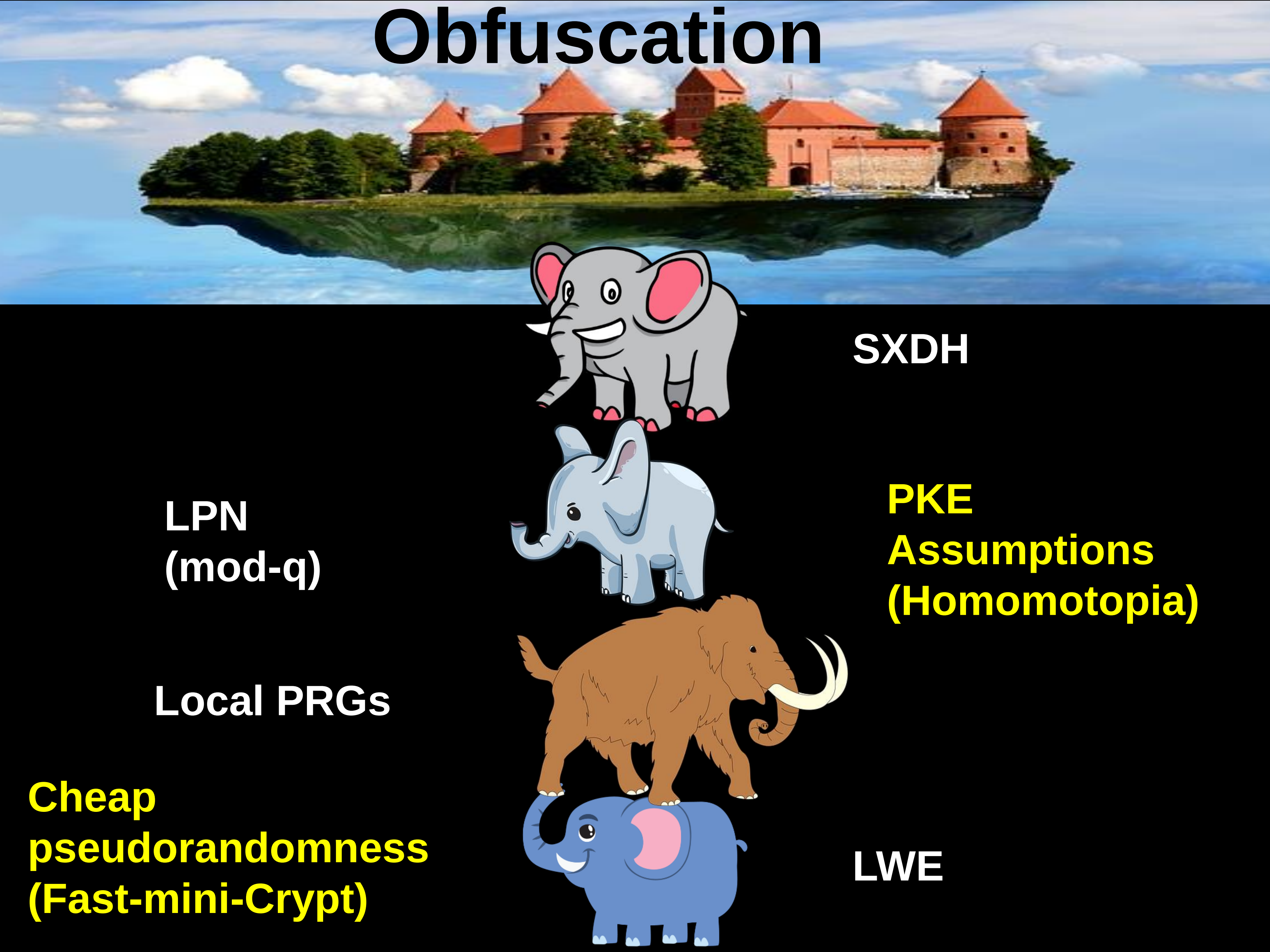
On the **Assumptions** used for **Obfuscation**

Benny Applebaum

Tel Aviv University

New Developments in Obfuscation

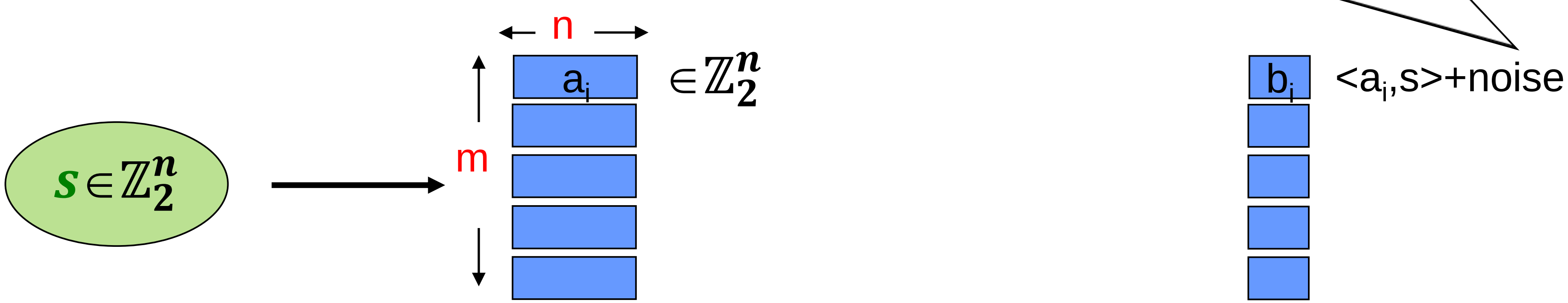
Simons Institute, December 2020


Obfuscation
SXDH
LPN
(mod-q)
PKE
Assumptions
(Homomotopia)
Local PRGs
Cheap
pseudorandomness
(Fast-mini-Crypt)
LWE

Learning Parity with Noise [BFKL94]

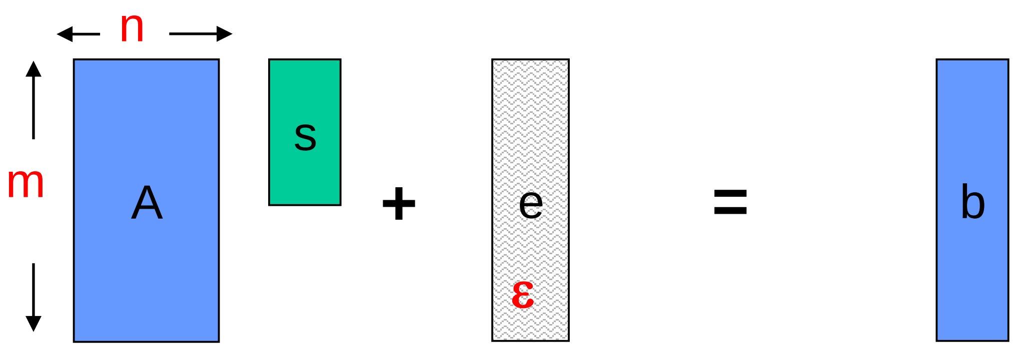
Problem: find s

iid noise:
each bit is 1 w/prob. $\varepsilon < 0.5$



Decoding Random Linear Code [GKL88]

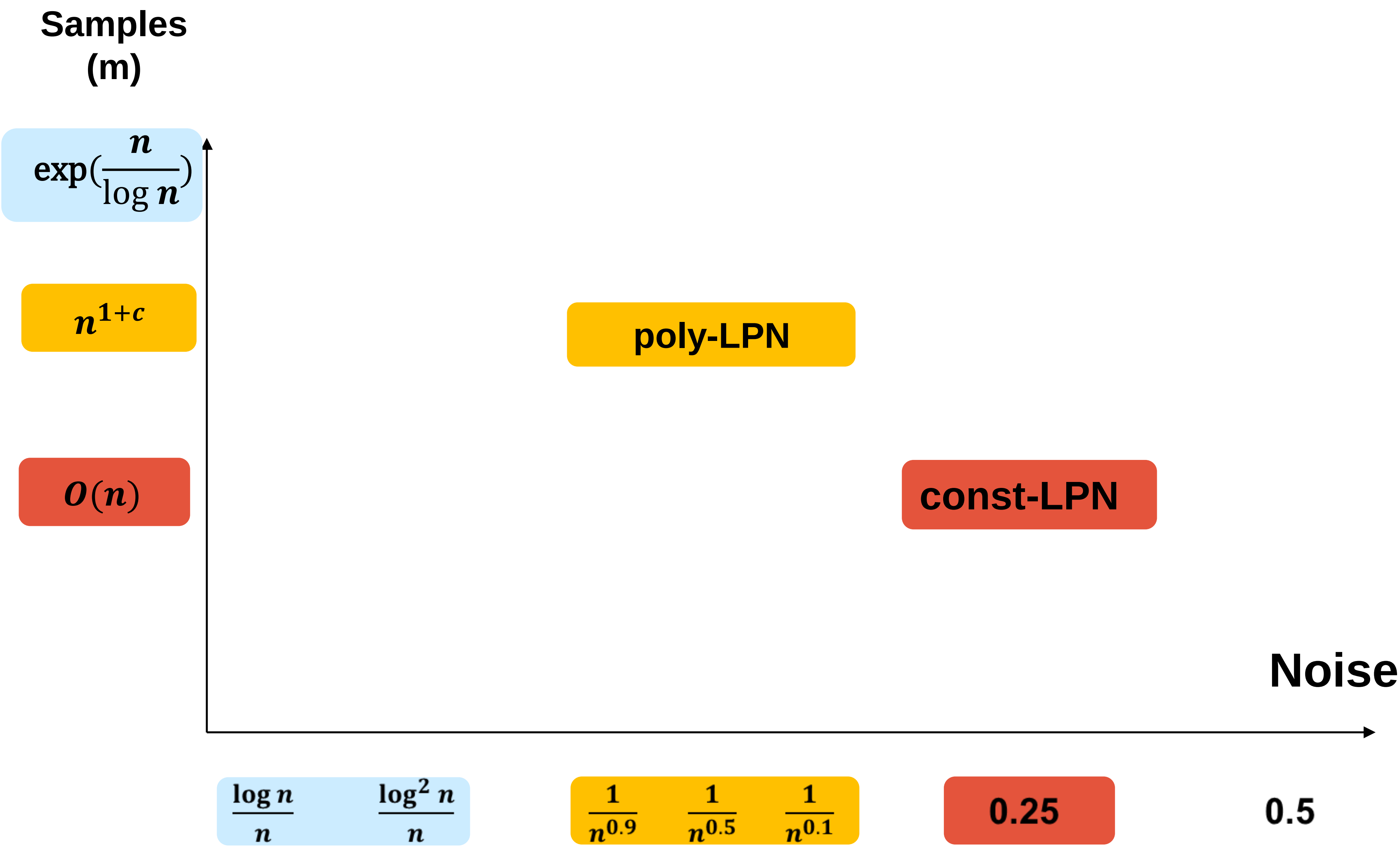
Problem: find s



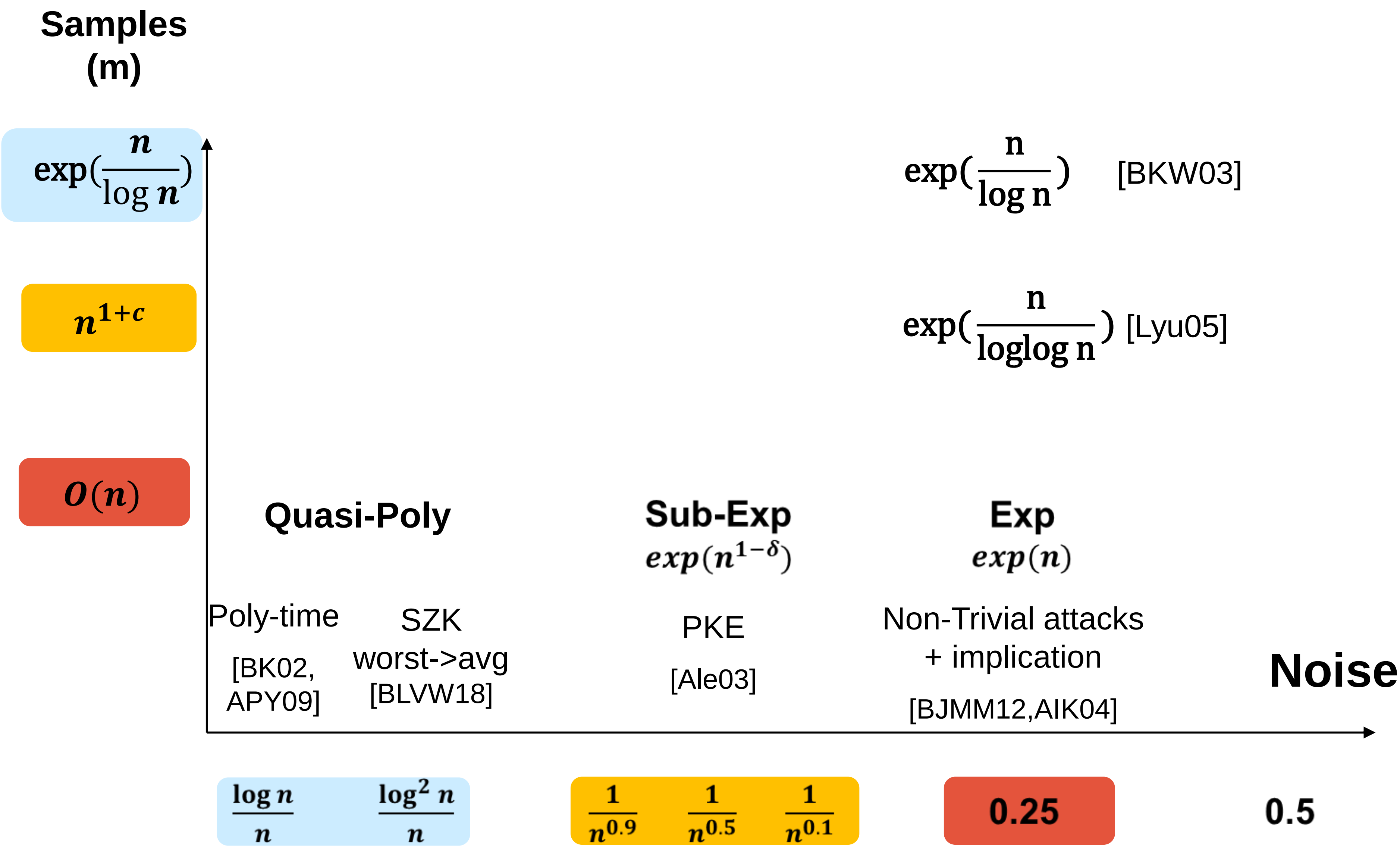
iid noise vector of rate ε

- Information theoretic solvable when $m > n/(1 - H(\epsilon))$
- Gets "easier" when m grows and ϵ decreases
    - Solving LPN(m, ϵ) => Solving LPN($m + m', \epsilon - \epsilon'$)
- Trivially solvable in time $2^{H(\epsilon)n}$
- Trivially solvable w/p $(1 - \epsilon)^n < 1 - \epsilon n$

Known Attacks

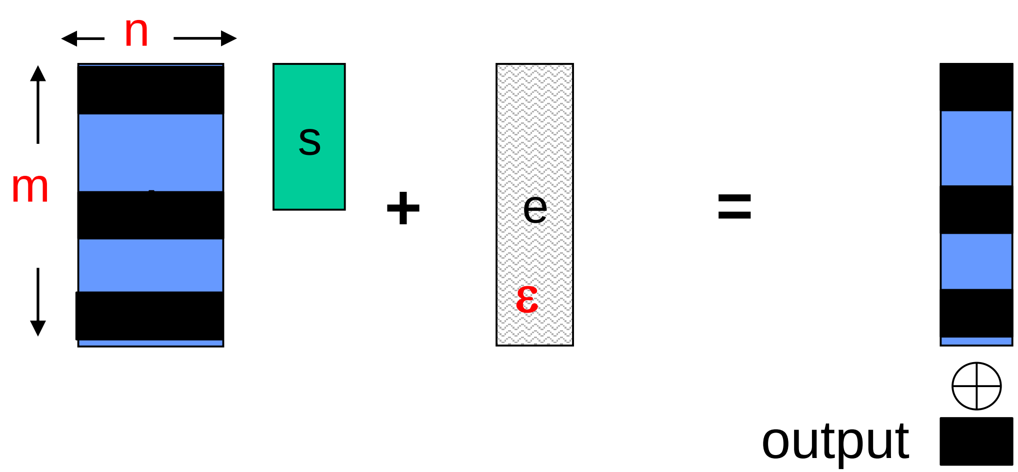


Known Attacks



Simple Distinguishing Attack

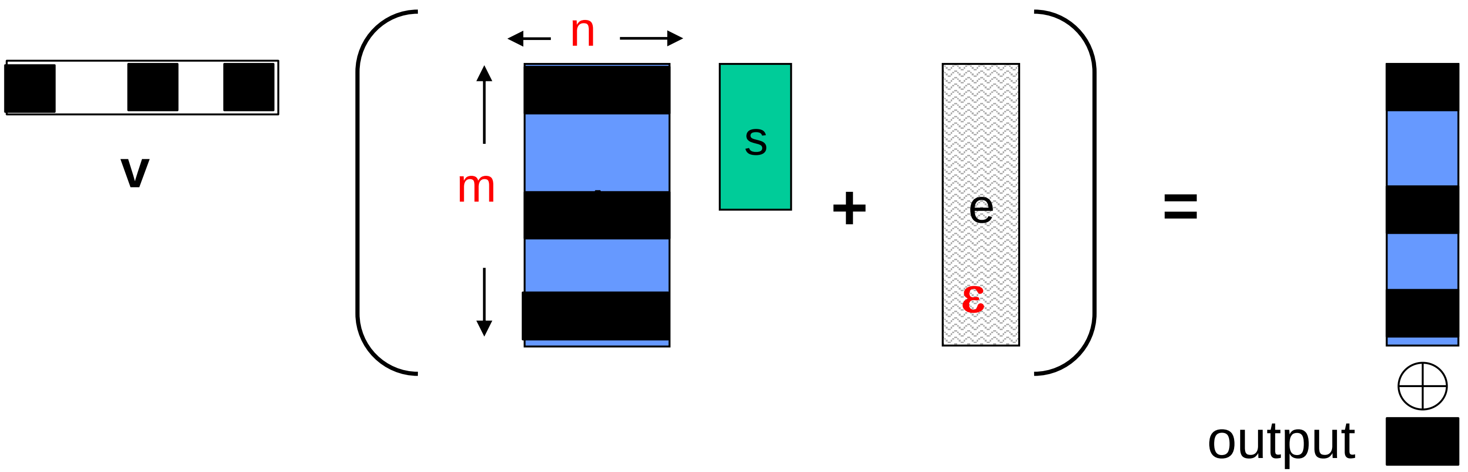
Goal: Distinguish (A,b) from (A, uniform)



1. Find “small” set of linearly dependent rows in A

Simple Distinguishing Attack

Goal: Distinguish (A,b) from (A, uniform)



1. Find “small” set of linearly dependent rows in A

   Δ-weight vector **v** in co-Kernel(A)

2. Output $\langle v, b \rangle = \langle v, e \rangle$

Distinguishing advantage $(0.5-\epsilon)^{\Delta} = \exp(-\Delta/\epsilon)$

How small is $\Delta = \Delta(n,m)$? $\tilde{O}(\frac{n}{\epsilon \log m})$

Ignoring complexity of finding **v** ⇒ overall complexity exp in $\tilde{O}(\frac{n}{\epsilon \log m})$

Pseudorandomness

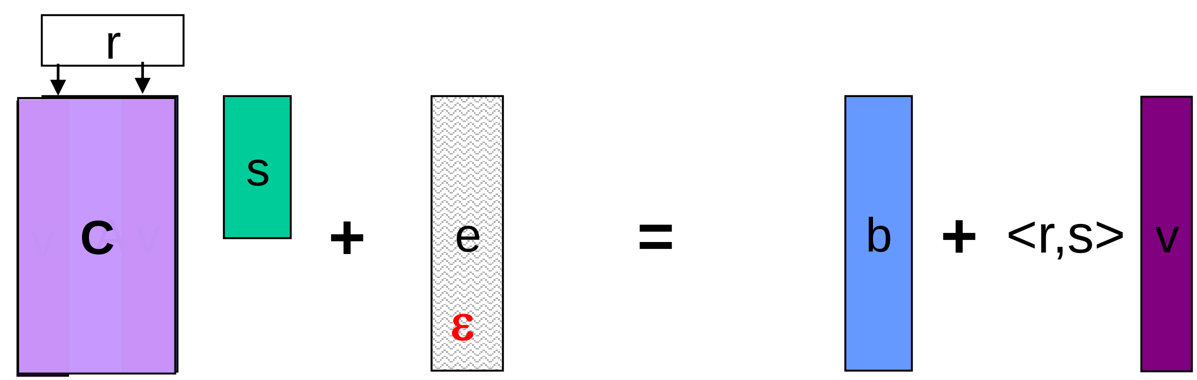
Thm.[BFKL94] LPN ⇒ pseudorandomness (A,As+e) ≈ (A,U_m)

Proof: [AIK07]

- Assume LPN ⇒ By [GL89] can't approximate **<s,r>** for a random **r**
- Use distinguisher **D** to compute hardcore bit **<s,r>** given a random **r**

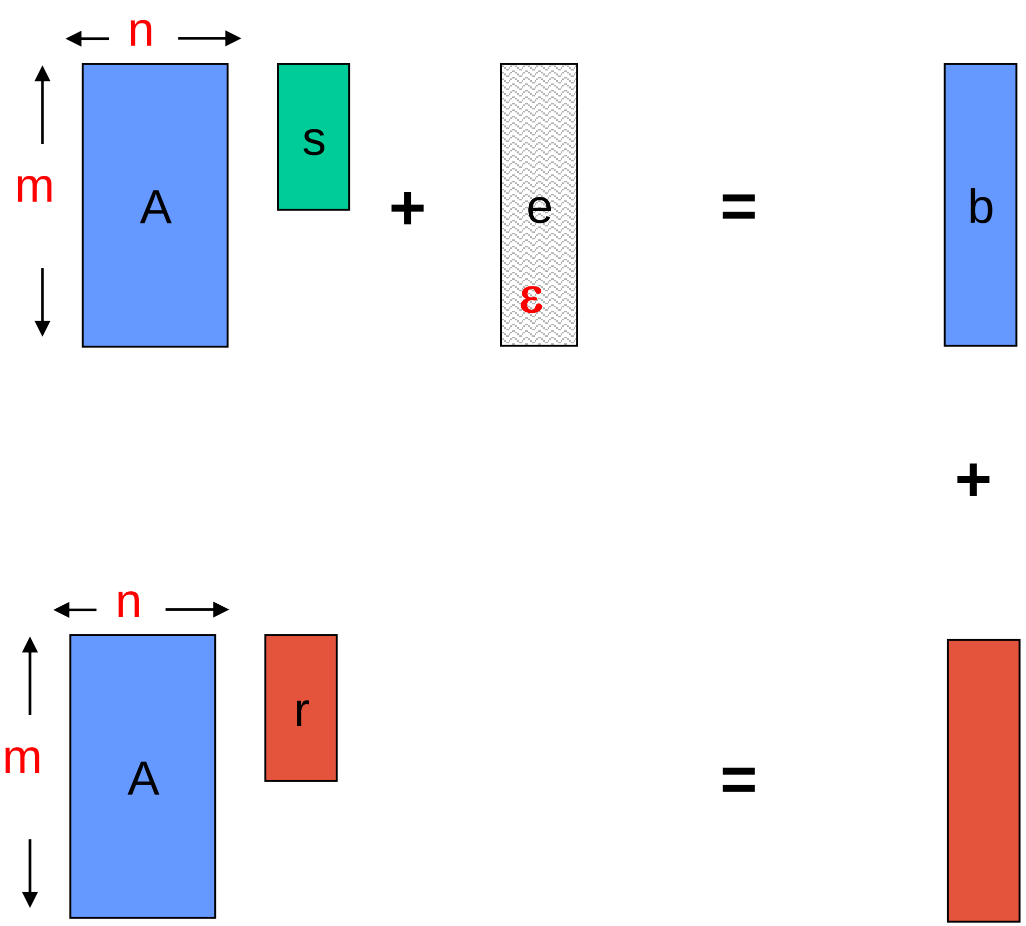
  -Given (A,b=As+e) and **r**$\in\{0,1\}^n$ define **C**=re-random(A) s.t:

C is random and **b** = { Uniform if <r,s>=1 ; Cs+e if <r,s>=0 }



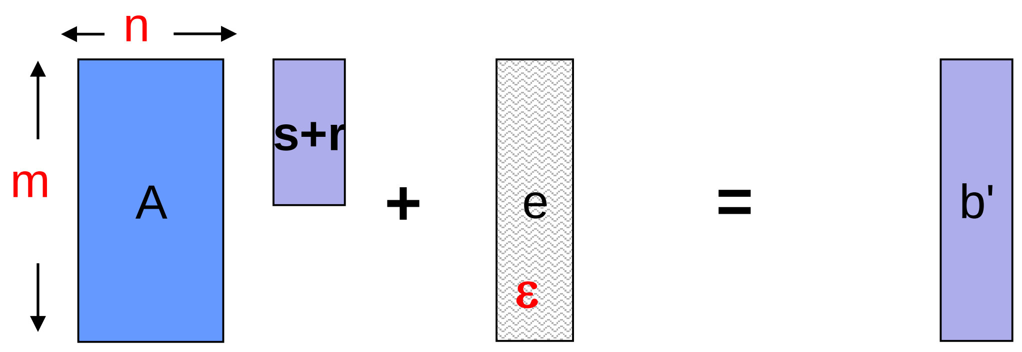
Random Self-Reducibility

Problem: find s



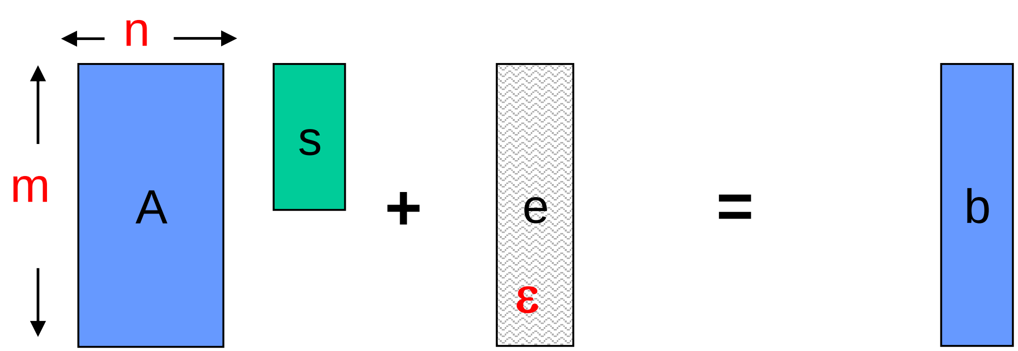
Random Self-Reducibility

Problem: find s



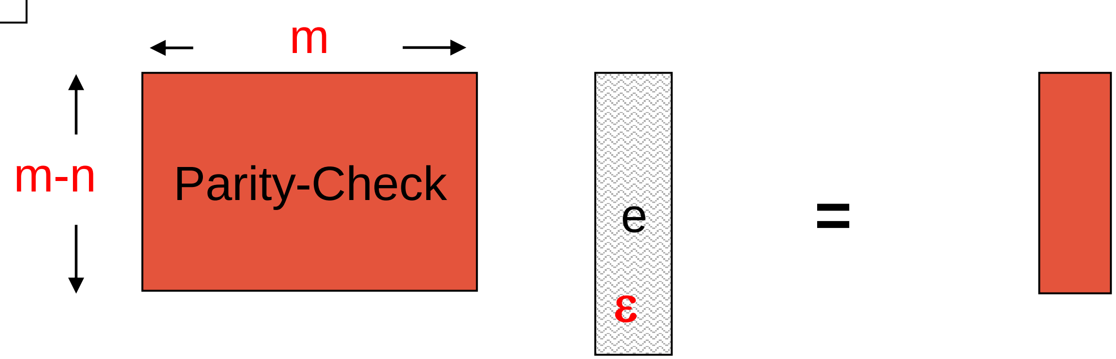
Dual Version: Syndrome Decoding

Problem: find s



iid noise vector of rate ε

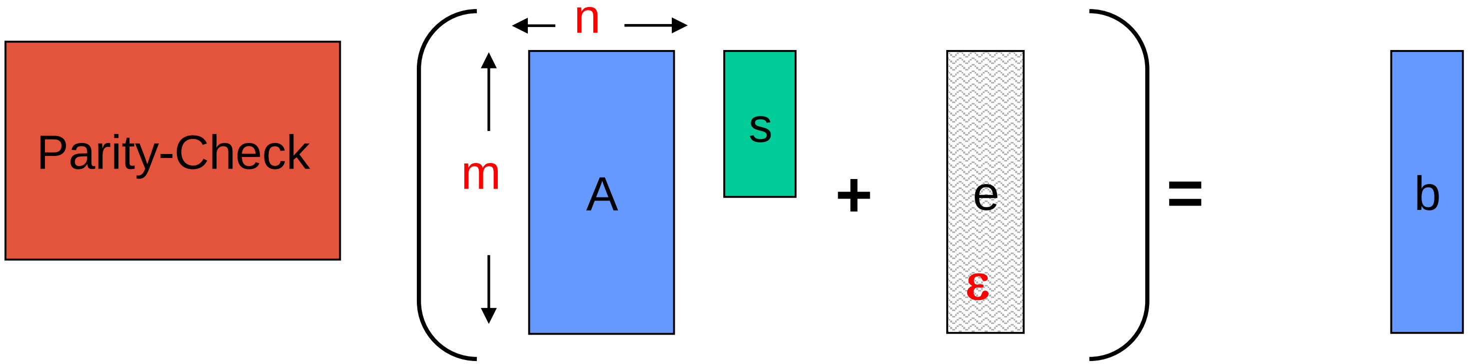
Problem: find e



iid noise vector of rate ε

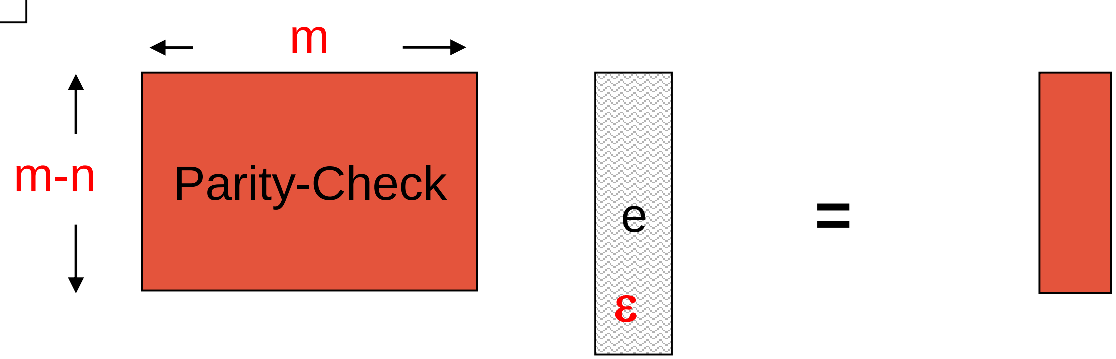
Dual Version: Syndrome Decoding

Problem: find s



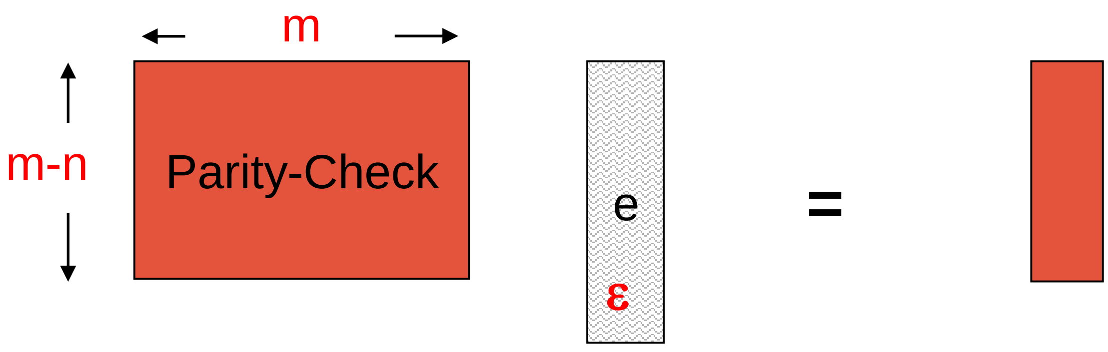
iid noise vector of rate ε

Problem: find x



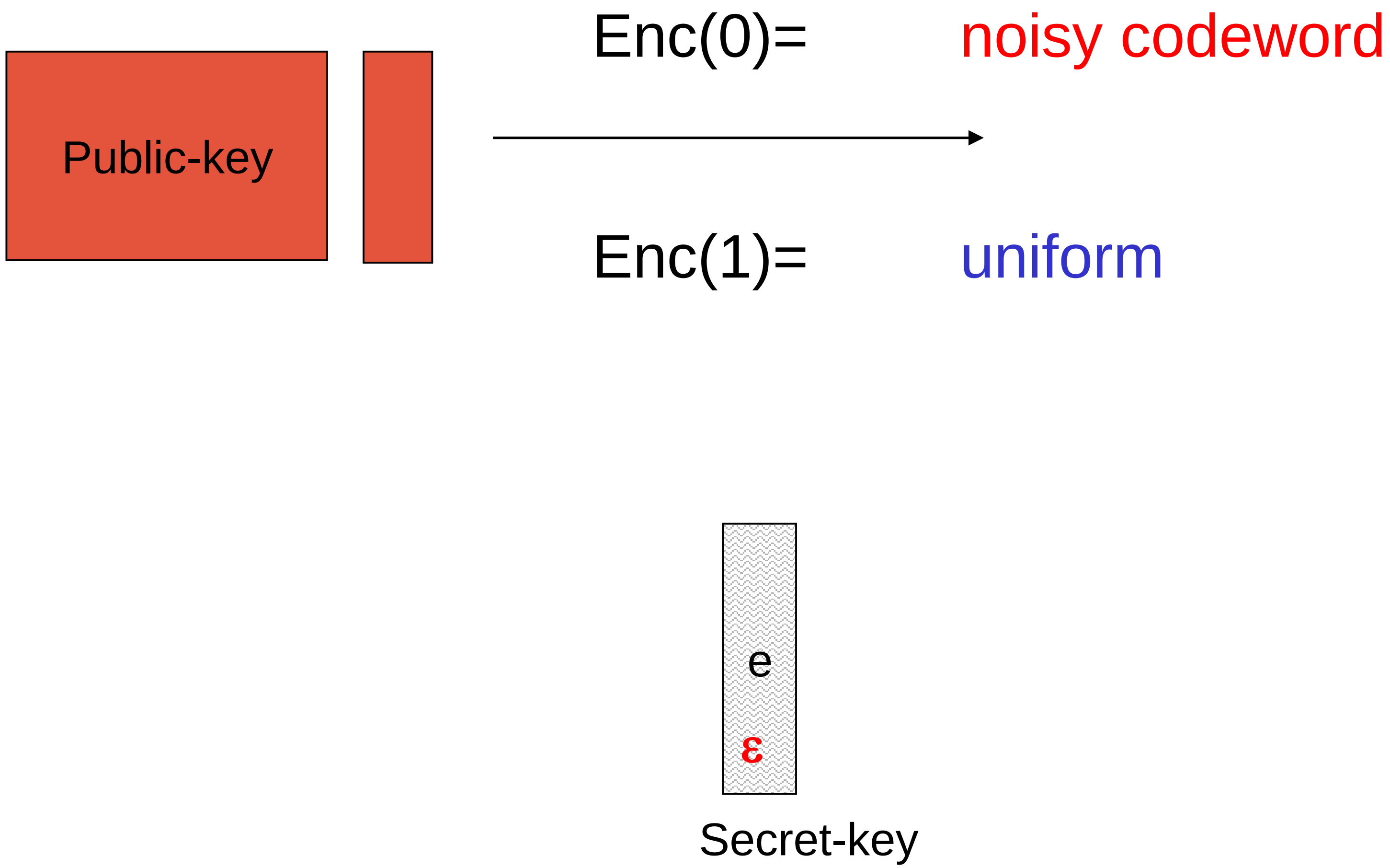
iid noise vector of rate ε

Corollary: Planting Short Vector in Kernel



iid noise vector of rate **ε**

Public-Key Encryption [Alek03]



LPN: Evidence for Hardness

- Search problem, Random-Self Reducibility
- Gaussian-Elimination is noise sensitive
- Well studied in learning/coding community for some parameters
  - "Win-Win" results
  - Provably resist limited attacks
- Robust (Search-to-Decision, leakage-resilient, low-weight secret, circularity)

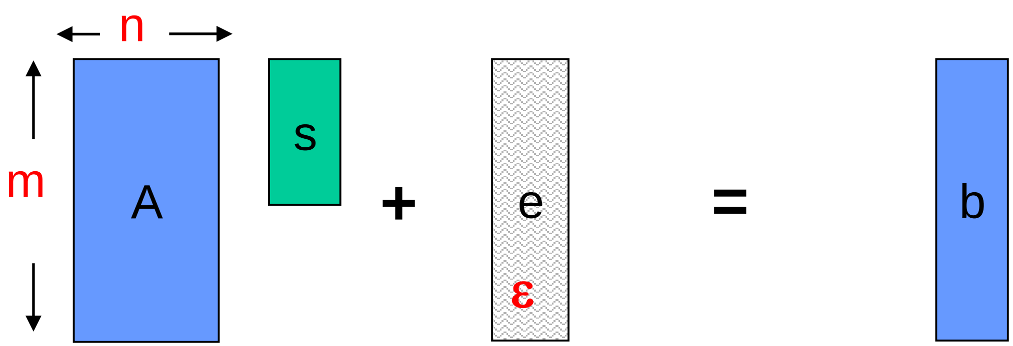
  [BFKL93,AGV09, DKL09, ACPS09, GKPV10, …,] See Pietrzak's survey
- Seems hard even for Quantum algorithms and co-AM algorithms
- "Simple mathematical domain" (compare with factoring/group-based crypto)

LPN: Features

- Simple algebraic structure: “almost linear” function

- Computable by simple (bit) operations
  - exploited by [HB01, …]

Variants



iid noise vector of rate **ε**

- Under-constraint case ($\Rightarrow$ hashing [AHIKV17])

- Changing the matrix distribution
  - Make sure that $\Delta(A)$ is not too small

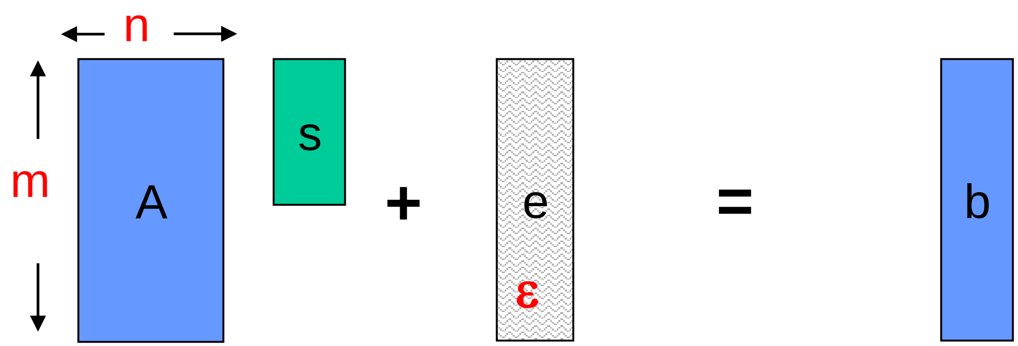
- Noise distribution
  - Fixed weight vector (OK)
  - Structured Noise (may be subject to linearization [AG11])

- Larger Alphabet
  - Noise: Gaussian vs **Bernoulli**

“LPN” over $\mathbb{Z}_q$

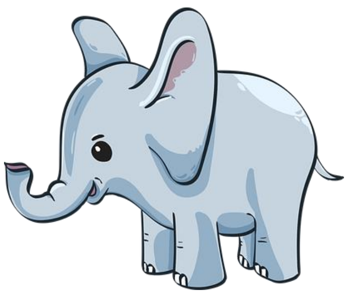
$A \in_R \mathbb{Z}_q^{m\times n}$

$s \in_R \mathbb{Z}_q^{n}$



$$\boldsymbol{e_i} = \begin{cases} \boldsymbol{U_q} & \textbf{w.p} \quad \boldsymbol{\epsilon} \\ \boldsymbol{0} & \textbf{w.p} \quad \boldsymbol{1-\epsilon} \end{cases}$$

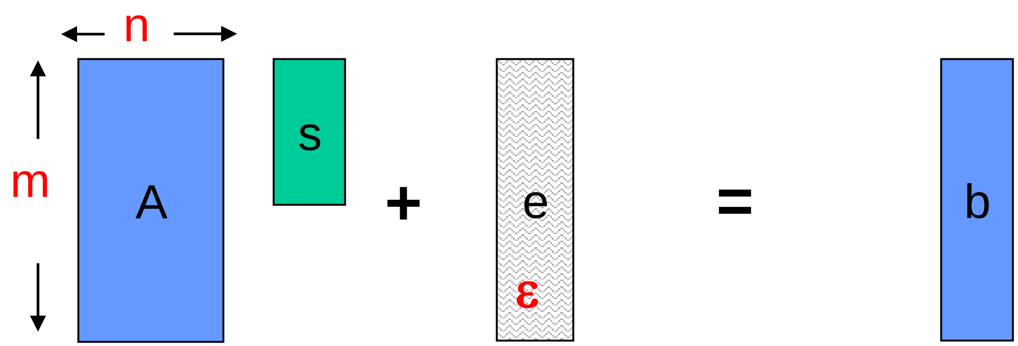
- Decoding over the q-ary symmetric channel (Random-Linear-Code)
- Support(x) = sequence of iid Bernoulli variables
  - Lifting binary-crypto to Arithmetic Crypto [IPS09, AAB15, ADINZ17, BCGI18…]
- Search-RLC(q,n,m,ϵ): hard to find s
- Decision-RLC(q,n,m,ϵ): $(A, b) \approx (U_q^{m\times n}, U_q^{m})$
- Equivalence not known when q is super-polynomial



"LPN" over $\mathbb{Z}_q$

$A \in_R \mathbb{Z}_q^{m \times n}$

$s \in_R \mathbb{Z}_q^{n}$

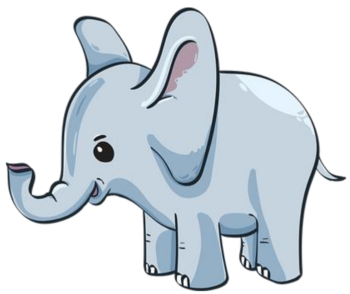


$$e_i = \begin{cases} U_q & \text{w.p } \epsilon \\ 0 & \text{w.p } 1 - \epsilon \end{cases}$$

Seems as hard as binary version (harder?)

- Noisy Linear Algebra is hard
- Large $q \Rightarrow$ less noise cancelations

Powerful assumption: Effective secret is $O_\epsilon(n)$ **bits**
but stretch is $\Omega_\epsilon(m)$ **field elements**

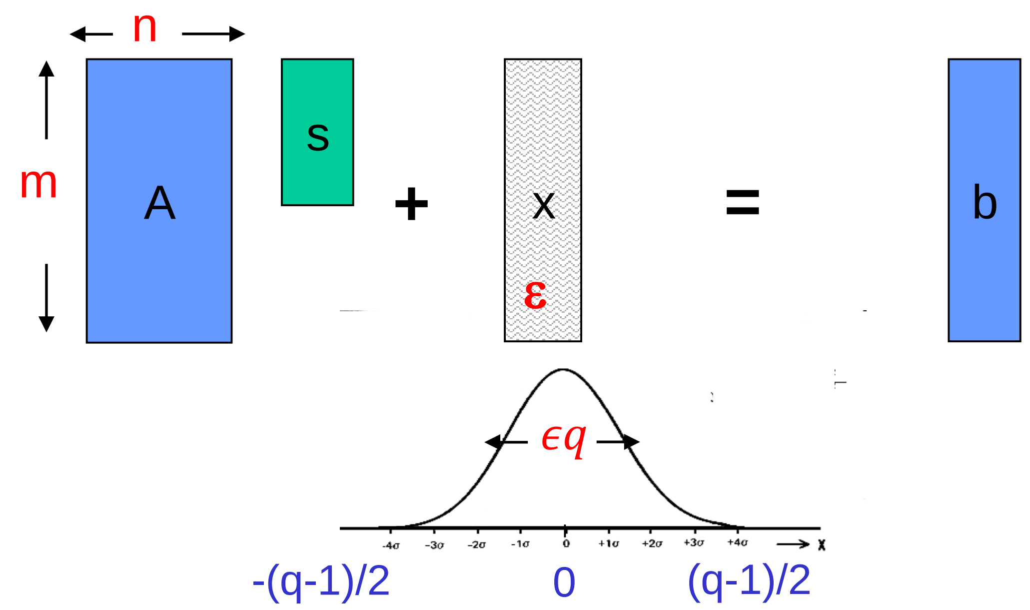


Requires further study especially for polynomial regime

Learning with Errors Variant [Regev05]

$A \in_R \mathbb{Z}_q^{m \times n}$

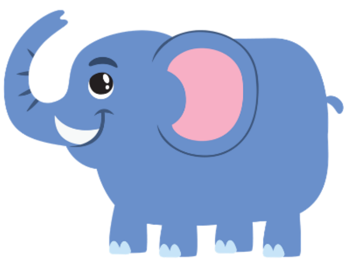
$s \in_R \mathbb{Z}_q^{n}$



Mainstream Crypto Assumption

Noise induces geometry

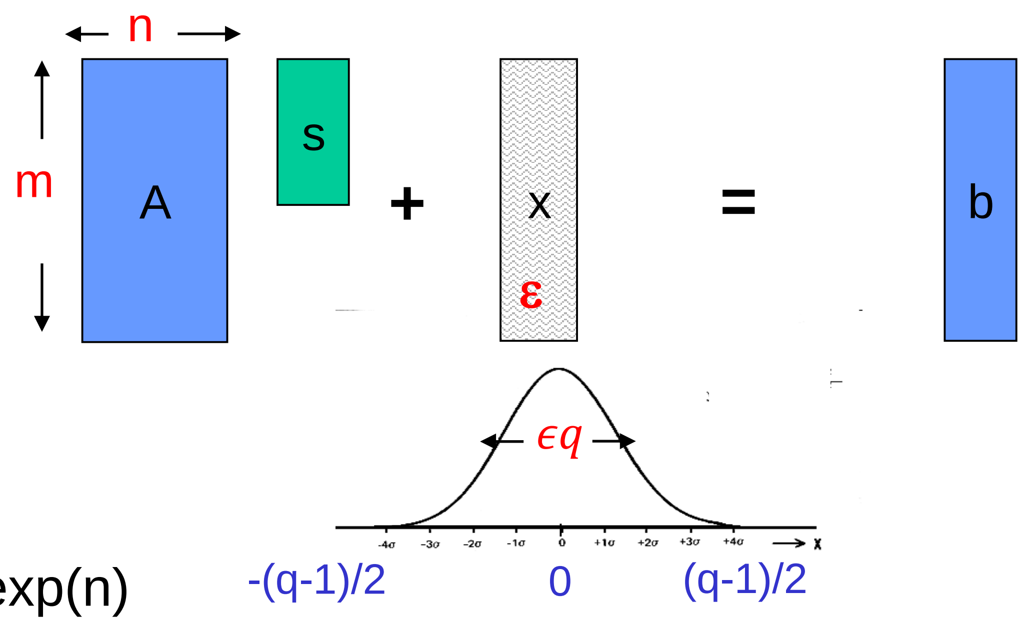
different game



Learning with Errors Variant [Regev05]

$A \in_R \mathbb{Z}_q^{m\times n}$

$s \in_R \mathbb{Z}_q^{n}$



- Modulus poly(n) or exp(n)
- Noise 1/poly(n) or 1/sub-exponential

As hard as worst-case Lattice problems (GAP-SVP) [Reg05,Peik09]

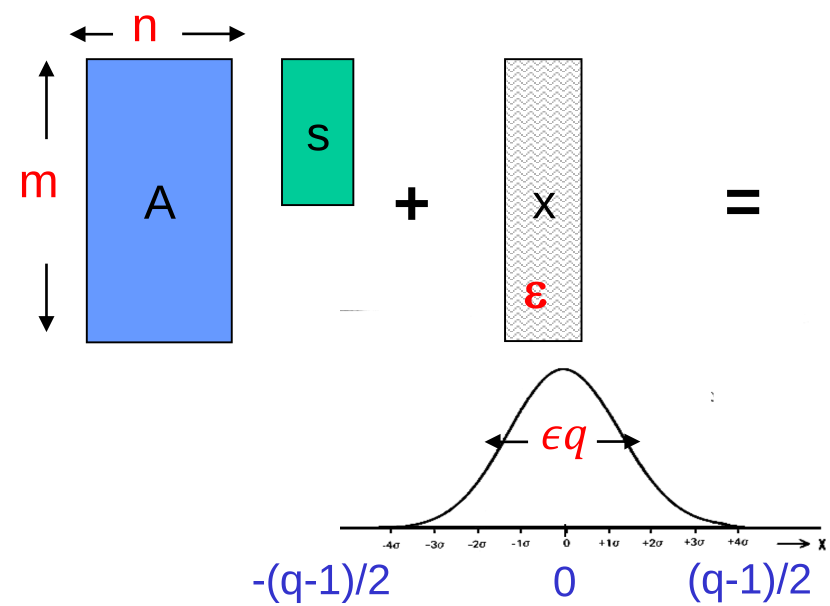
- Approximation factor $\tilde{O}(n/\epsilon)$
- exp-approximation easy via [LLL82]

Believed to be sub-exp secure even against Quantum adversaries

Learning with Errors Variant [Regev05]

$A \in_R \mathbb{Z}_q^{m\times n}$

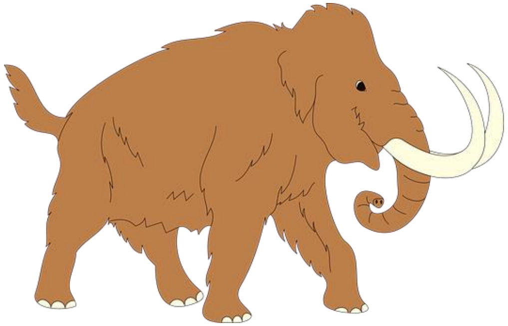
$s \in_R \mathbb{Z}_q^n$



Low noise ⇒ Can repeatedly add noise vectors

- Unlike the Bernoulli variant
- Generate additional equations for free
- Key to many applications [GPV08, …,BV11,…]
- Puts the problem in SZK (“co-NP attacks”) [GG98,MV03]

Local PRGs

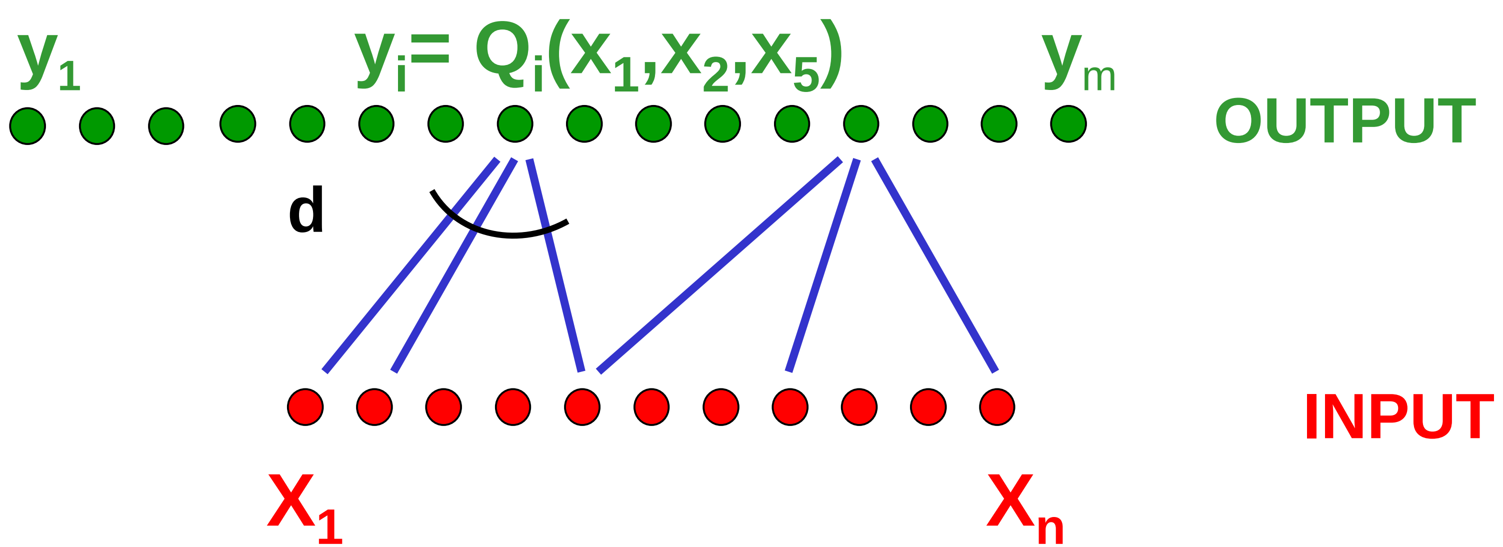


Locally Computable Functions
(NC^0)

Each output depends on constant number of inputs

Function defined by:

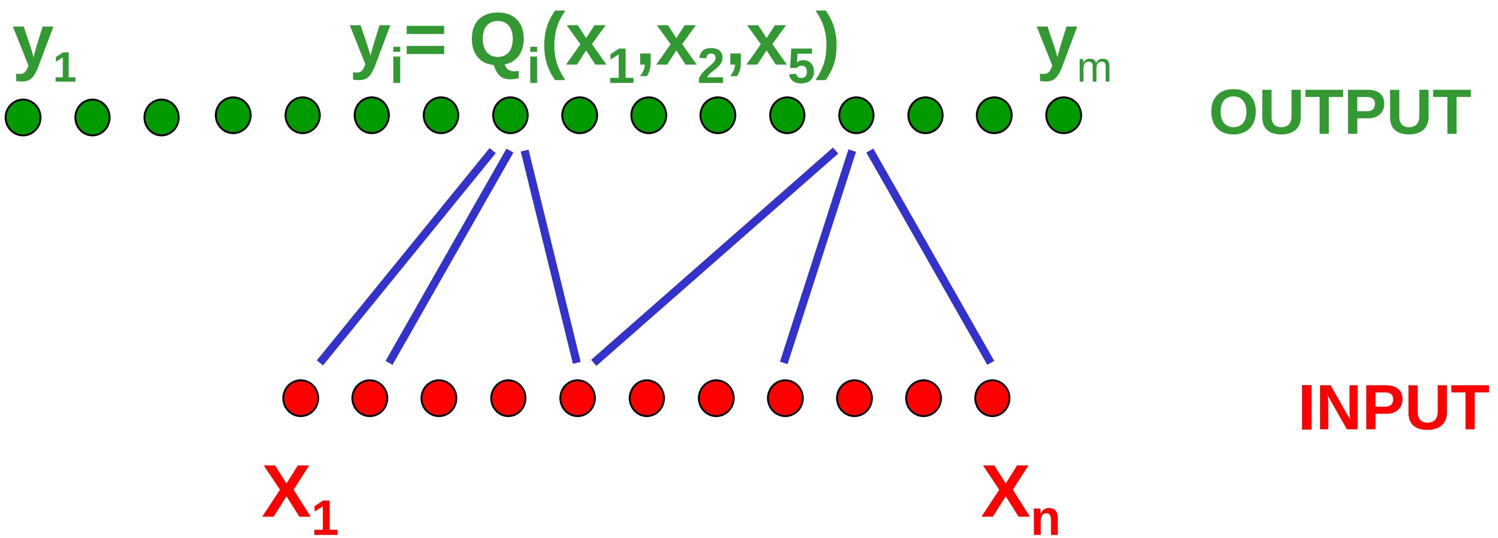
- (m,n,d) graph G
- List of d-local predicates $Q_1,\ldots,Q_m:\{0,1\}^d\rightarrow\{0,1\}$



Locally-Computable PRGs?

Long line of works [CM01,MST02,AIK04,….] see survey [A13]

Stretch matters!

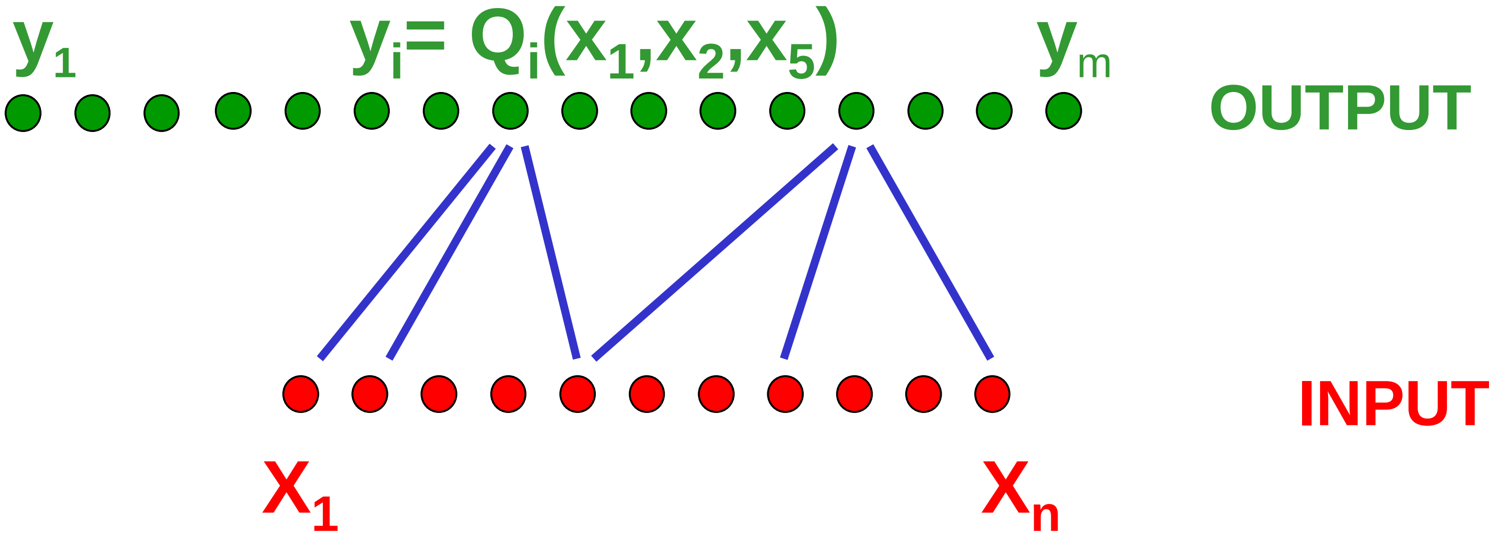


Sub-Linear Local PRG in NC^0

Stretch: $m = n + n^{1-\epsilon}$

Follows from any OWF in NC1 [AIK04]

- Most standard cryptographic assumptions
- Lattices, DLOG, factoring, LPN, asymptotic DES/AES



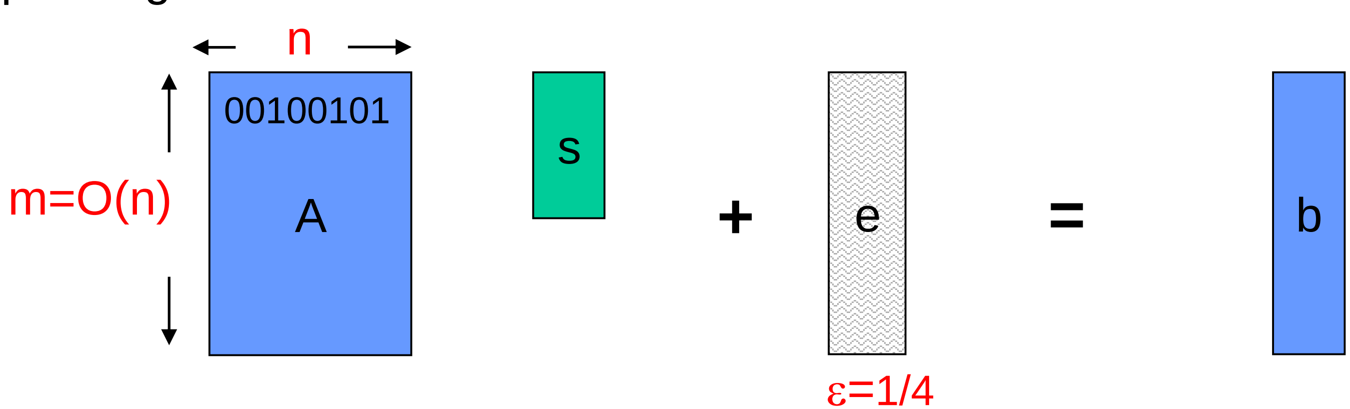
Lin-PRG in NC^0

Linear Stretch: $m = (1+\epsilon)n$

Follows from LPN over sparse matrix [AIK07]

- Assumption made by [Alek03]
- Implies hardness of refuting 3-SAT [Feige02]

Random Sparse Matrix
or
Any sparse expanding matrix

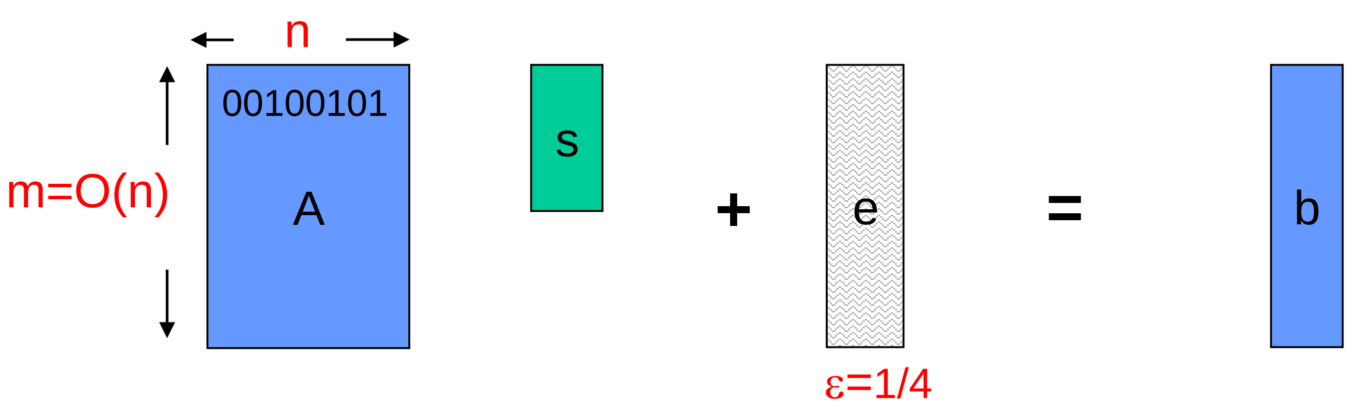


Lin-PRGs in NC^0

[A-17] Also follows from other assumptions

- Any exponentially-hard regular Local OWF (e.g., [Gol00])

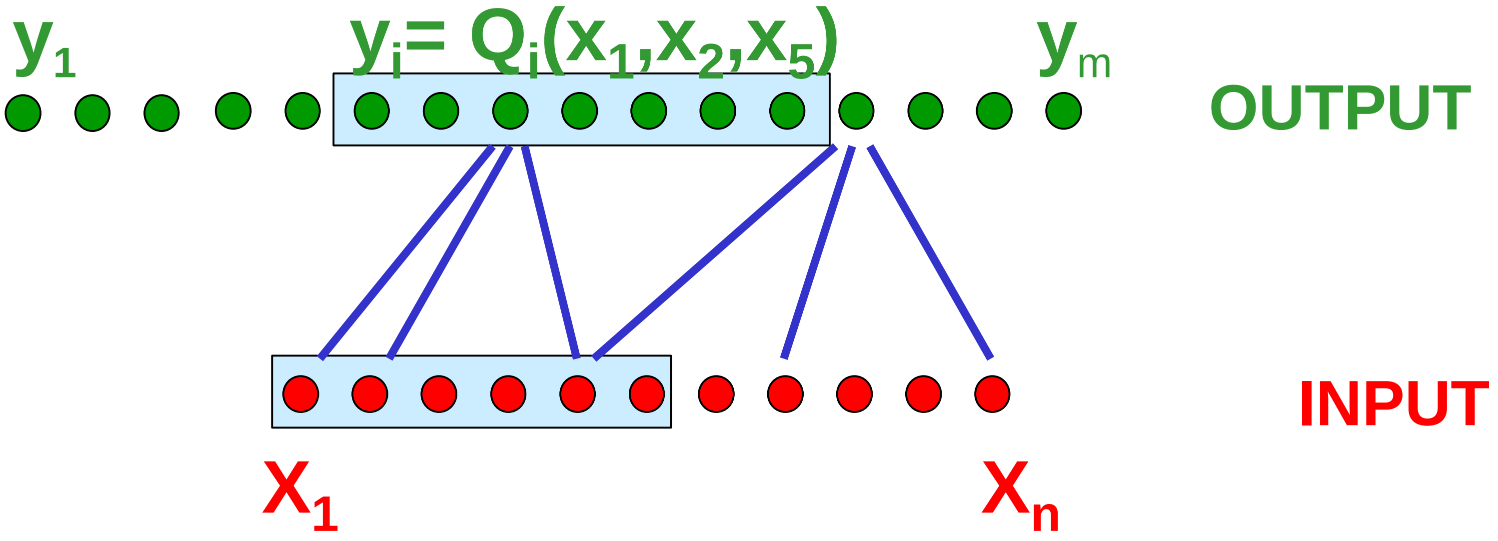
- Exp-hard LPN over O(n)-time computable code, e.g., [DI14]



Lin-PRGs in NC^0

Generic attack [AIK07]

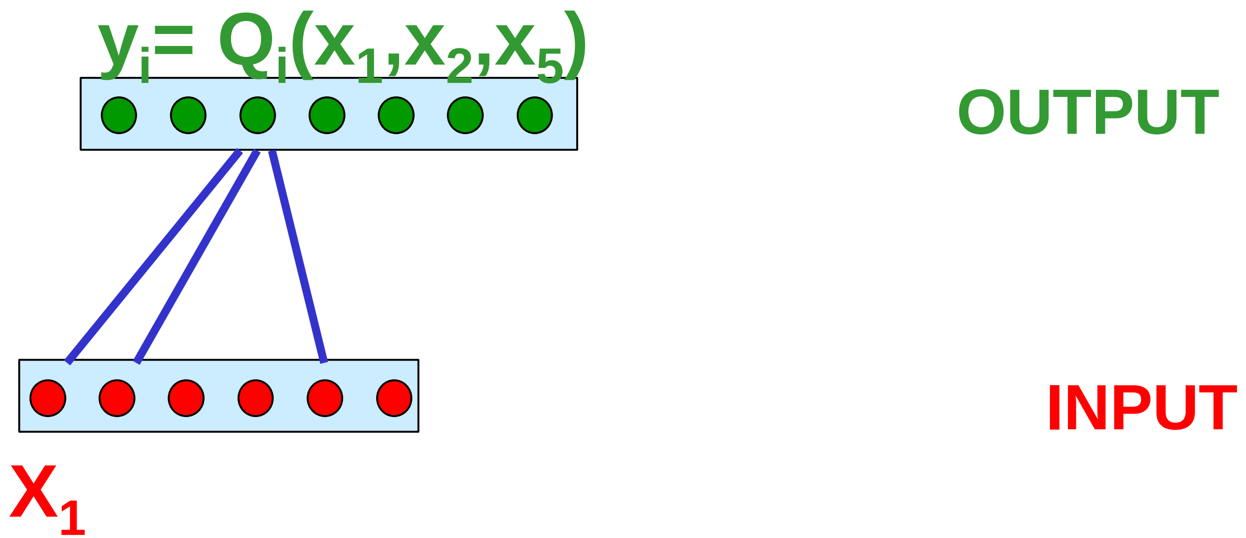
- Find shrinking set
- Enumerate over projected seed



Lin-PRGs in NC^0

Generic attack [AIK07]

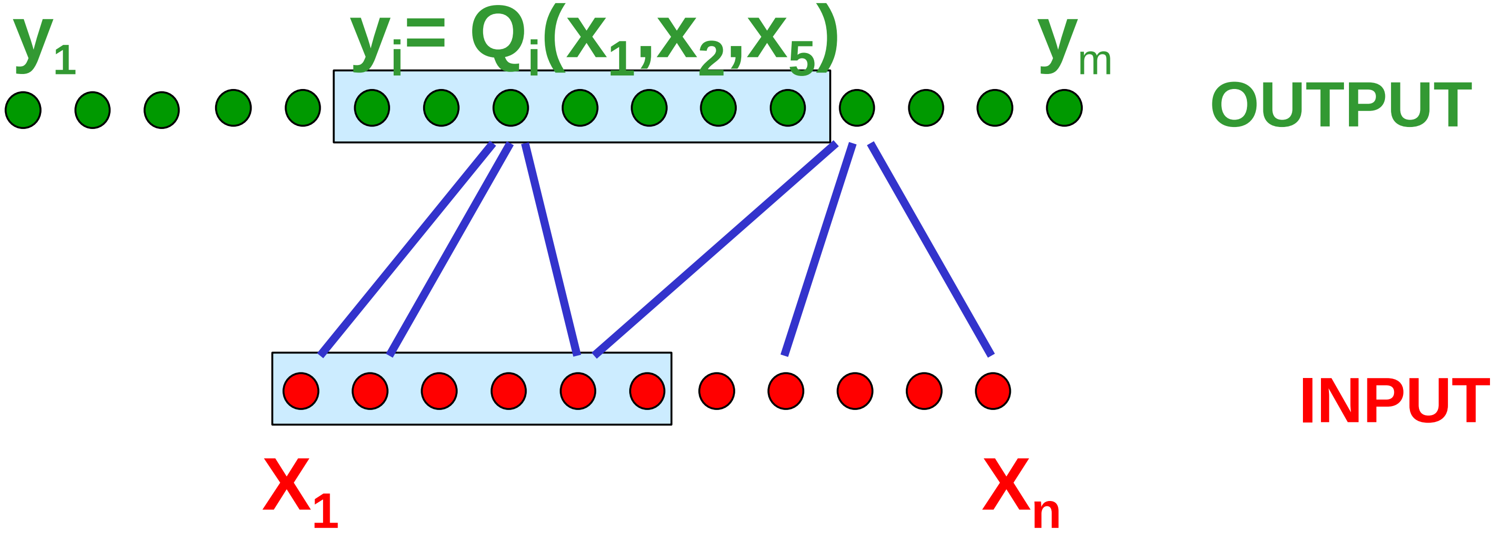
- Find "small" shrinking set of size **k**
- Enumerate over projected seed



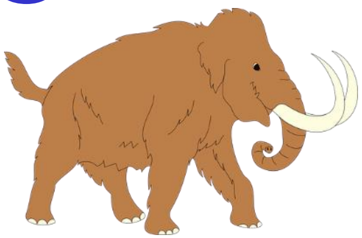
Lin-PRGs in NC^0

Expansion is necessary!

- Plausible to achieve $\exp(n)$ security

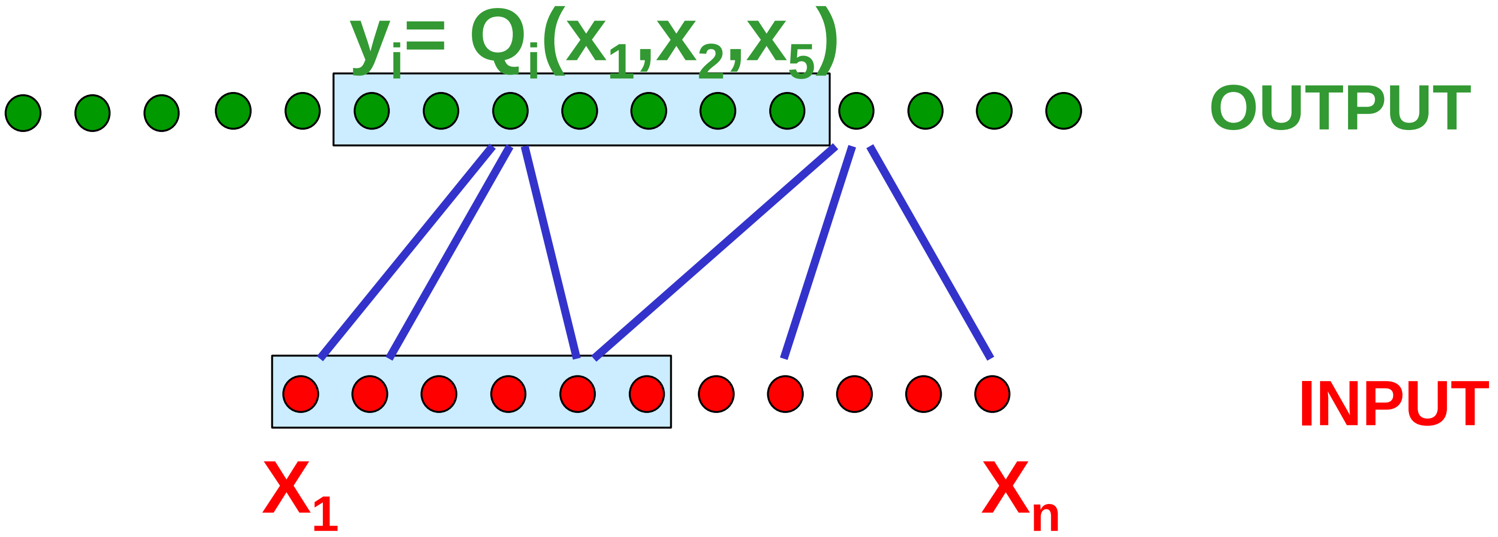


Poly-Stretch PRG in NC^0



Polynomial-Stretch: $m = n^2$

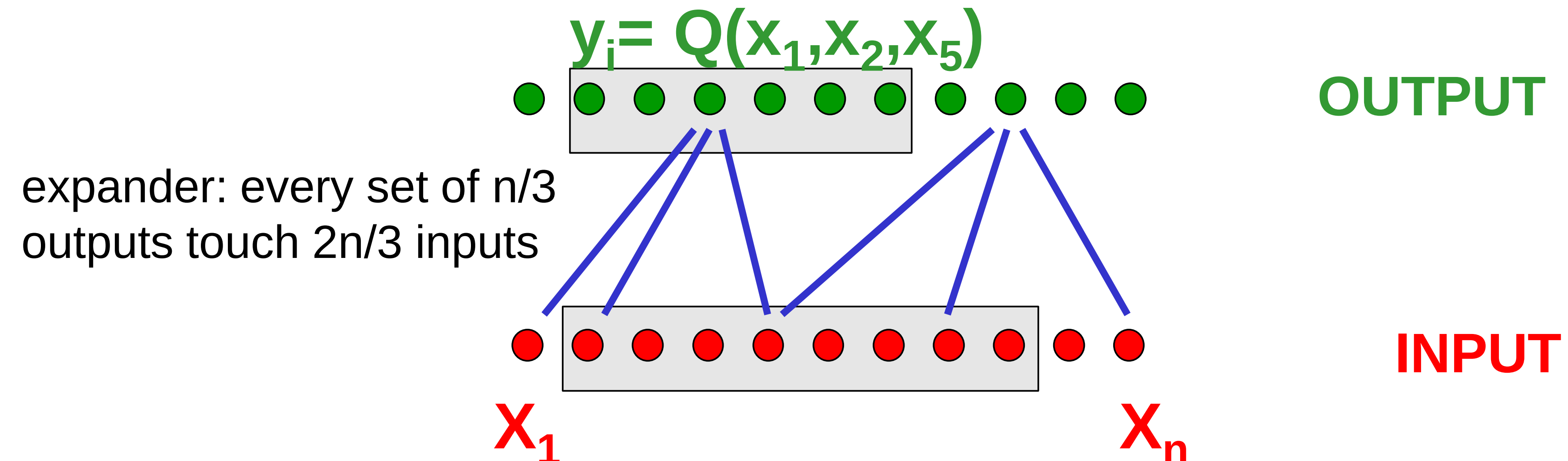
- Can only get $n^{1-\delta}$ expansion $\Rightarrow$ sub-exp security
- Morally should get from sparse-LPN w/ sub-const noise [ABW10]
- All known constructions rely on var's of Goldreich's Assumption



Goldreich's Assumption [ECCC '00]

Conjecture: for **random** predicate **Q** , and ∀ **expander G, m=n** inversion takes **exp(Ω(n))-time**

- First candidate for **optimal one-way function**
- Random local function is whp exp-hard to invert
- Constraint Satisfaction Problems are cryptographically-hard



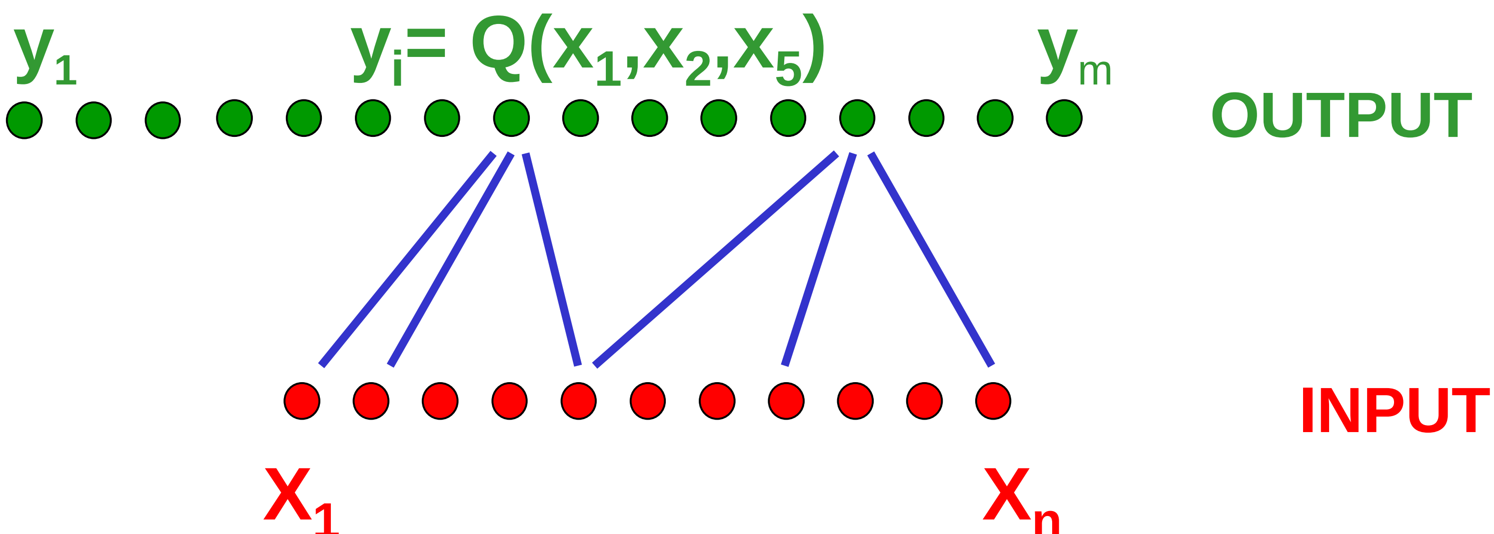
Generalization to Long Output

OW-Conjecture: for **properly chosen** predicate **Q**, any graph **G** inversion complexity is exponential in **the expansion of G**

Params: output length m, predicate Q, locality d, expansion quality

- Larger m ⇒ easier to attack ⇒ security requires more “robust” predicates
- Weaker variant: for random graphs no poly-time inversion
- Strong variant confirmed for many classes of attacks

[CEMT09,ABW10,A12,ABR12,BR11,BQ12,OW14,FPV15,AL16, KMOW16] See survey [A15]



Generalization to Long Output

OW-Conjecture: for **properly chosen** predicate **Q**, any graph **G** inversion complexity is exponential in the **expansion of G**

[A12,AR16]

weak

PRG-Conject: for **properly chosen** predicate **Q**, any graph **G** distinguishing complexity is exp. in **expansion of G**

1/poly-advantage

[AK19]

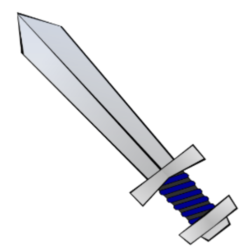
Poly-stretch local PRG

Generalization to Long Output

PRG-Conject: for **properly chosen** predicate **Q**, any graph **G** distinguishing complexity is exp. in **expansion of G**

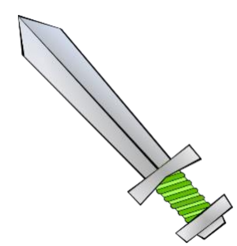
Which predicates yield PRGs?

Resiliency



“Local” attacks

“Degree”

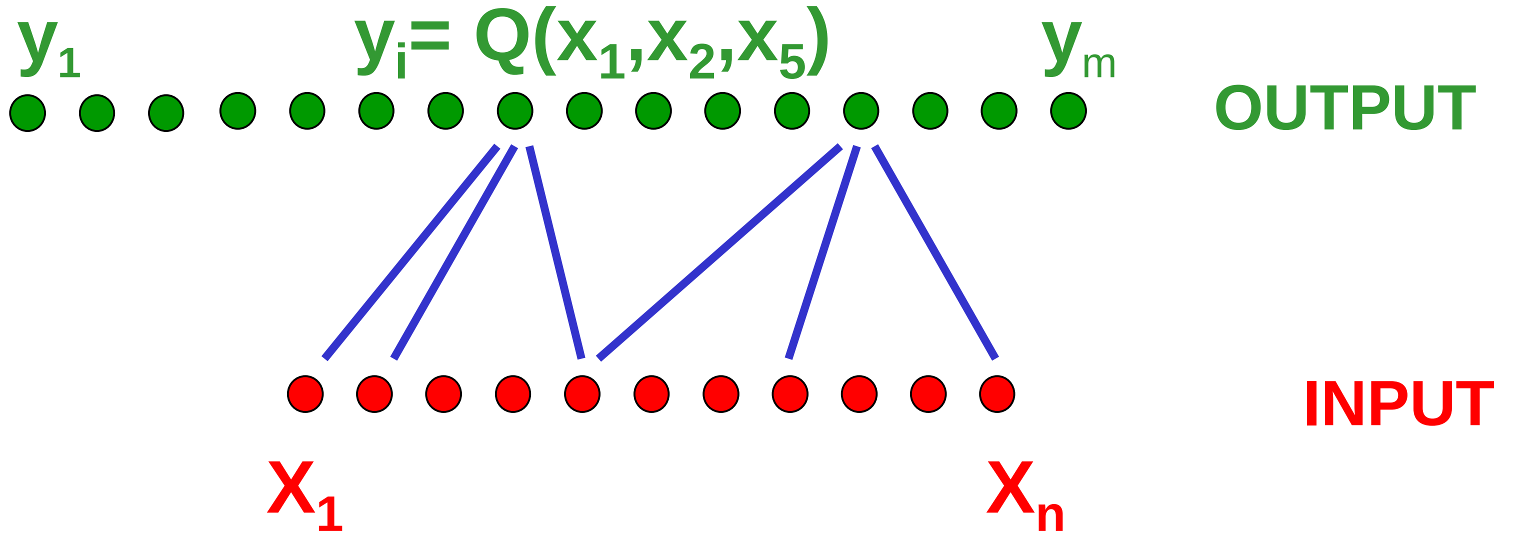


Linear algebra

Goal: Hard to distinguish y from random

More fragile than one-wayness:

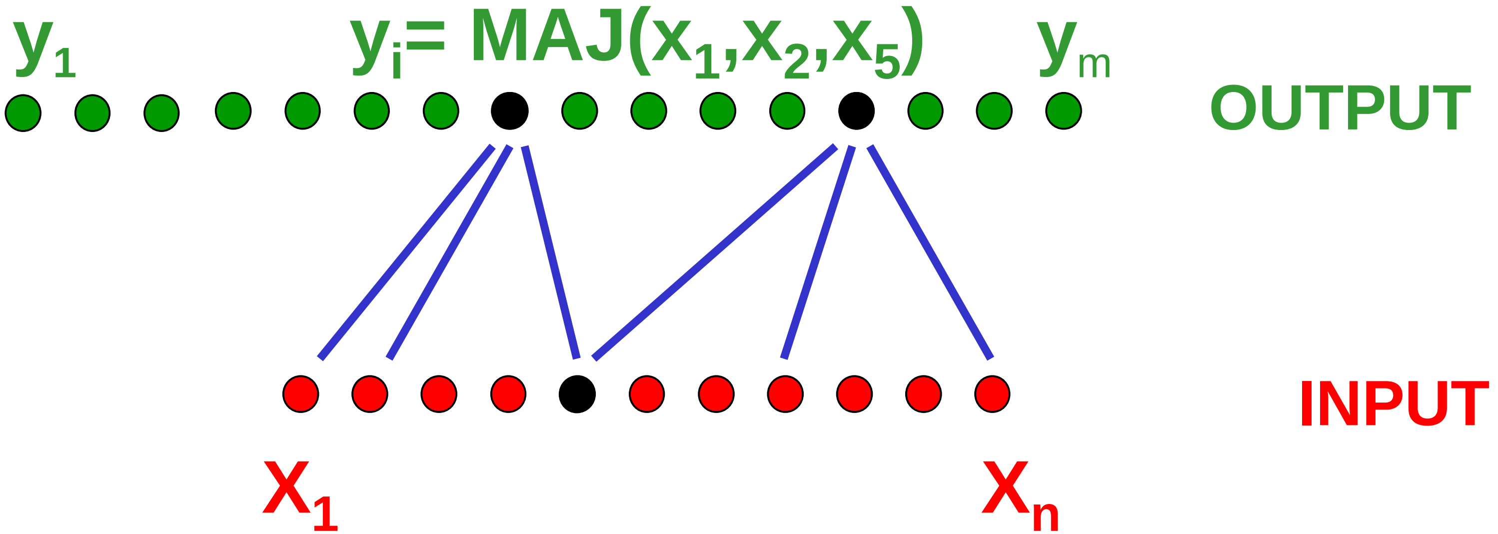
Predicate must be balanced



Goal: Hard to distinguish y from random

More fragile than one-wayness:

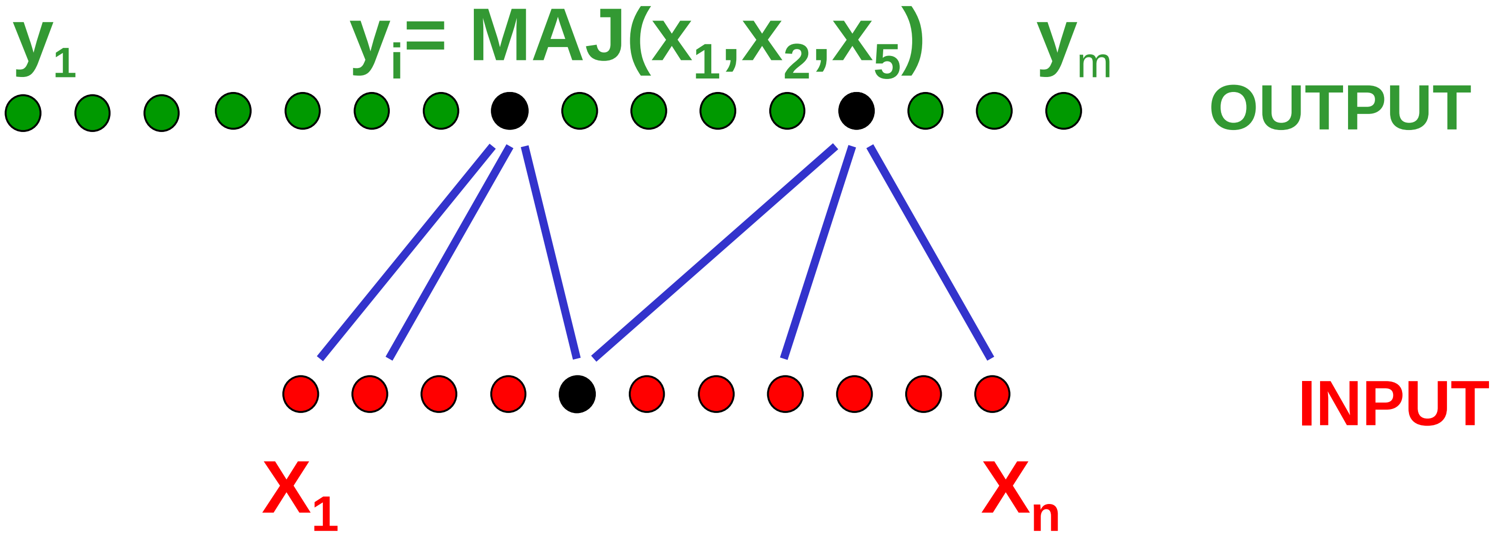
Predicate must be balanced even after fixing single input



Goal: Hard to distinguish y from random

k-resiliency [Cho-Gol-Has-Fre-Rud-Smo]:

Predicate must be balanced even after fixing **k** inputs

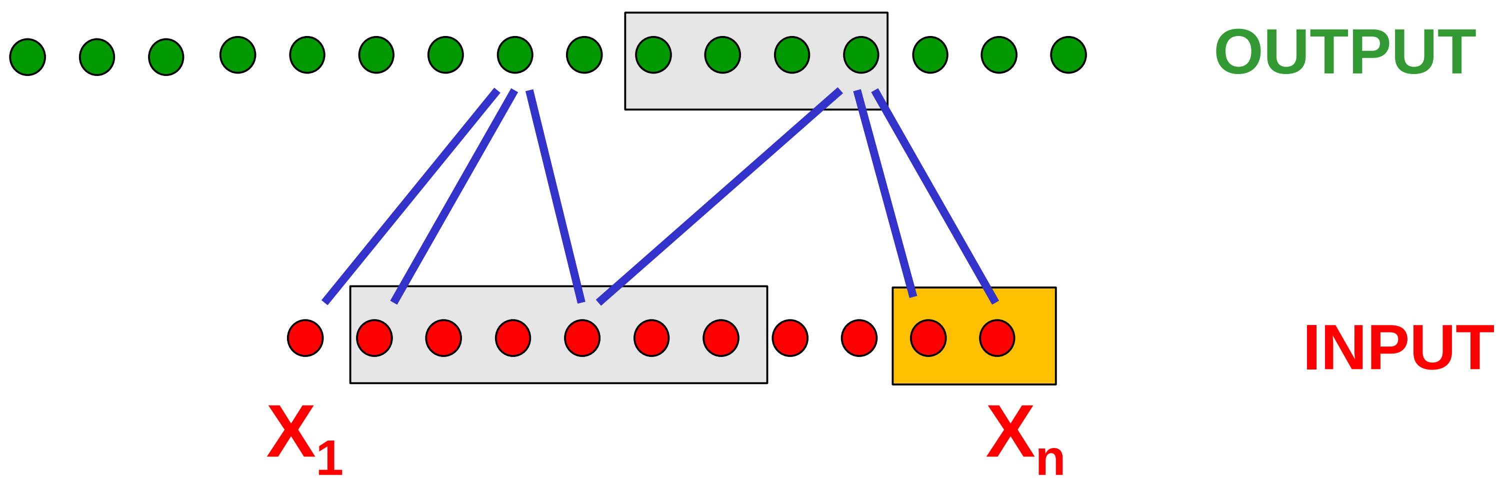


Resiliency defeats local attacks

[Mossel-Shpilka-Trevisan'03]

For $m=n^s$ resiliency of $k=2s-1$ is necessary and sufficient against

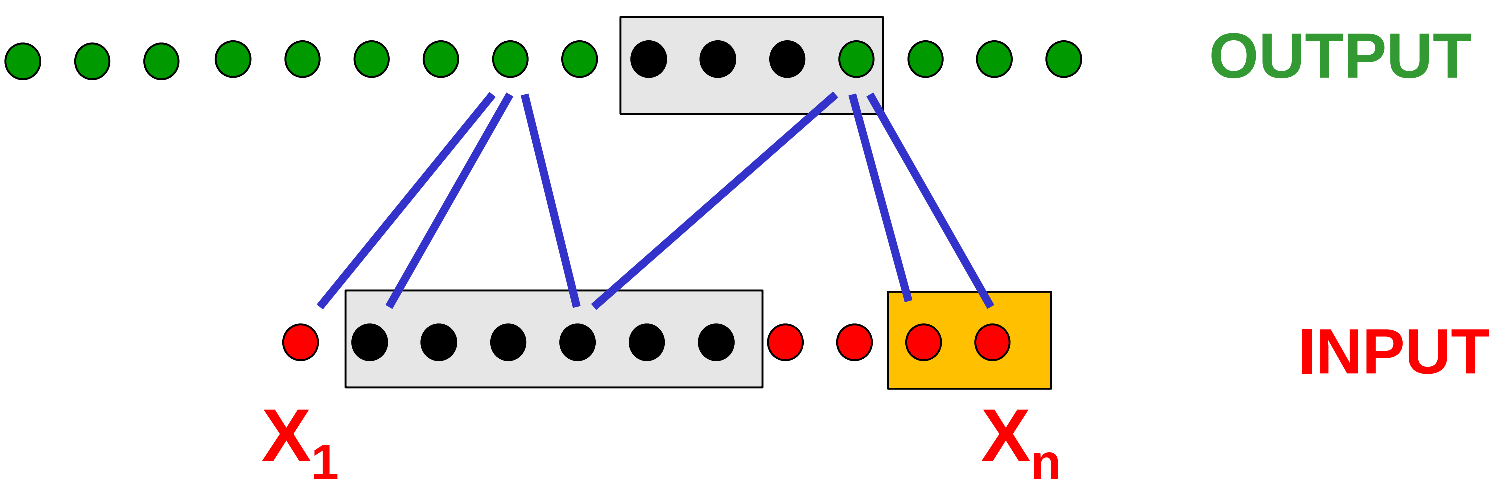
- Sub-exponential AC0 circuits [A-Bogdanov-Rosen12]
- Semidefinite programs [O'Donnel Witmer14]
- Sum of Squares attacks [Kothari Mori O'Donnel Witmer17]
- Statistical algorithms [Feldman Perkins Vempala15]



Resiliency defeats local attacks

For $m=n^s$ resiliency of $k=2s-1$ is necessary and sufficient against

- Sub-exponential AC0 circuits [A-Bogdanov-Rosen12]
- Semidefinite programs [O'Donnel Witmer14]
- Sum of Squares attacks [Kothari Mori O'Donnel Witmer17]
- Statistical algorithms [Feldman Perkins Vempala15]



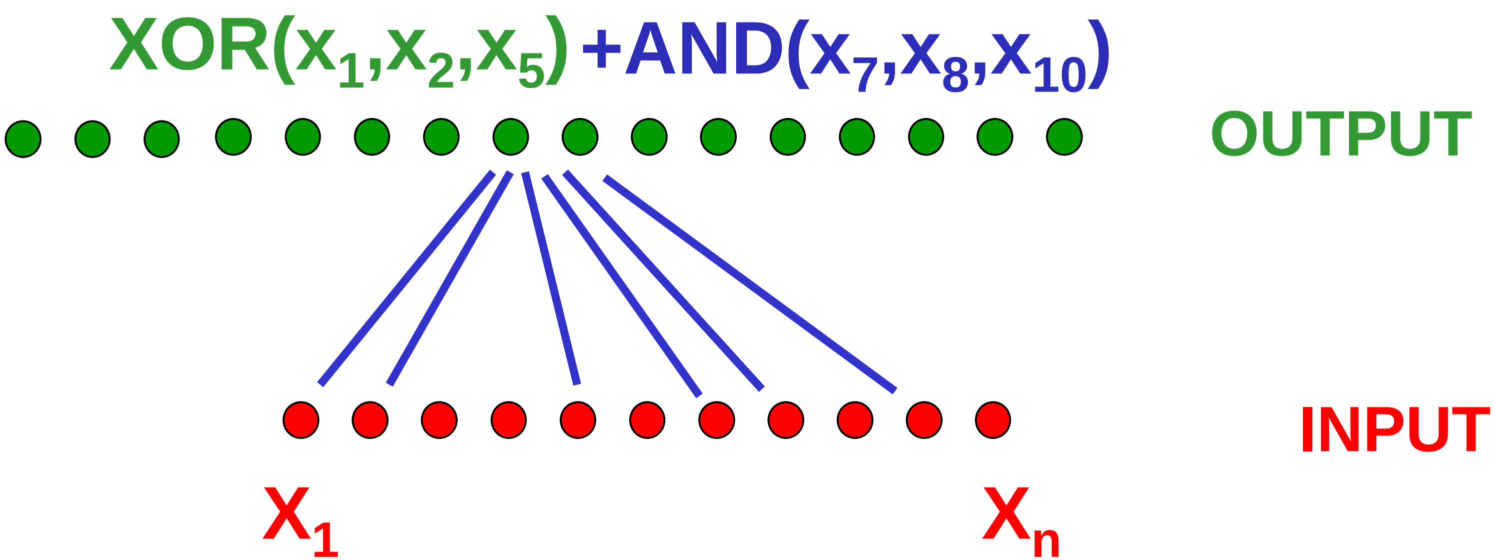
Defeating Linear Algebra

For $m=n^s$ need **algebraic degree** of s

Resiliency+Degree⇒Pseudorandomness? [OW14, A14, FPV15]

- Yes for $m<n^{5/4}$ and linear distinguishers [MST03, ABW10, ABR12]
  i.e., small-bias generator [NN]

- No for larger m’s [A-Lovett16]

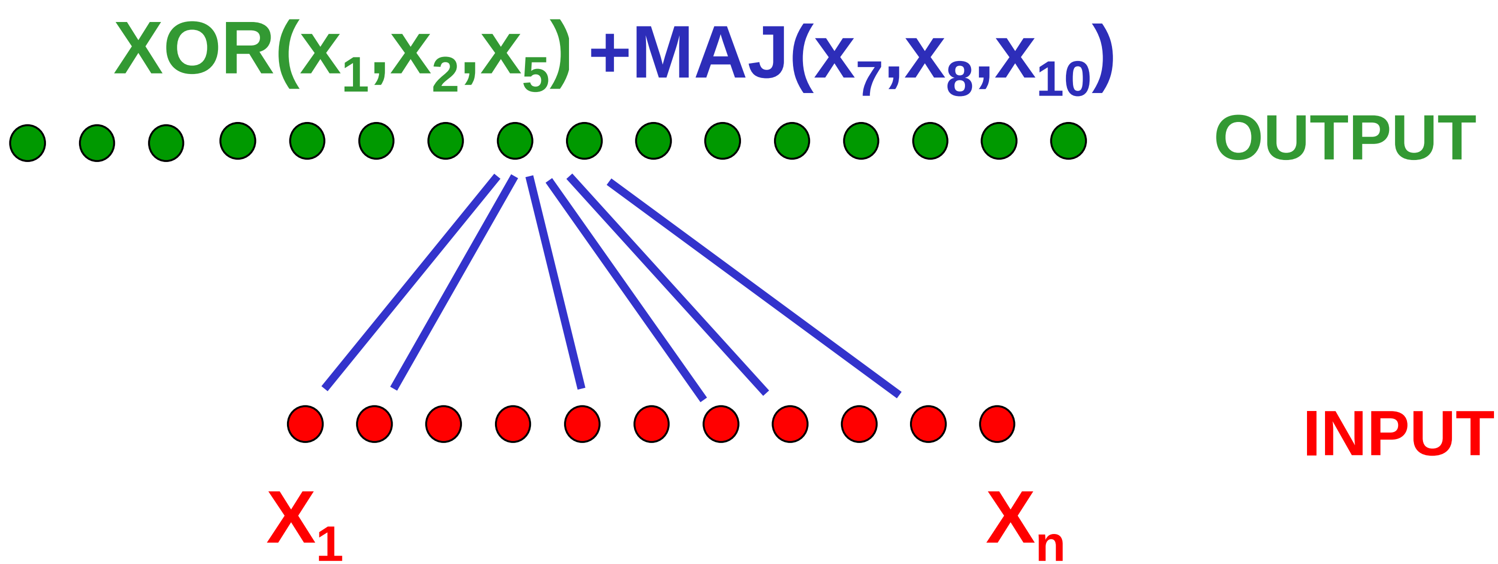


Defeating Linear Algebra [A-Lovett16]

b-**fixing degree**: **algebraic degree** of b **even after fixing** b inputs

Thm: For $m=n^s$, $\Theta(s)$-bit fixing degree
necessary & sufficient against linear distinguishers

A stronger form of **rational-degree** is necessary & sufficient for defeating “algebraic attacks”



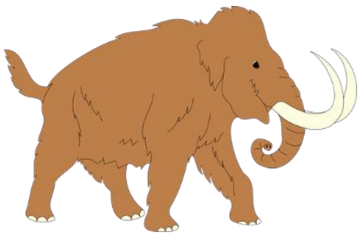
Summary: Local PPRGs

Seem to achieve sub-exp security

- For proper predicate best attack is exponential in expansion
- Concrete security should be further studied, see [CDMRR18]

Interesting TCS applications

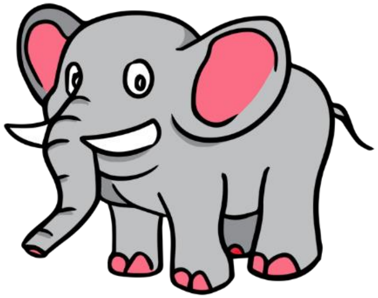
- CSPs are hard to approximate [Feige02, Ale03, AIK07,…,A17]
- Densest-subgraph is hard to approximate [A12]
- Hardness of learning depth-3 AC0 [AR16]



Symmetric eXternal DH [BGdMM05]

$$e: \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$$

- **SXDH:** DDH is hard in both $\mathbb{G}_1$ and $\mathbb{G}_2$
  - $(g^a, g^b, g^{ab}) \approx (g^a, g^b, g^c)$ for $a, b, c \leftarrow \mathbb{Z}_p$
  - where g generates $\mathbb{G}_1$ or $\mathbb{G}_2$

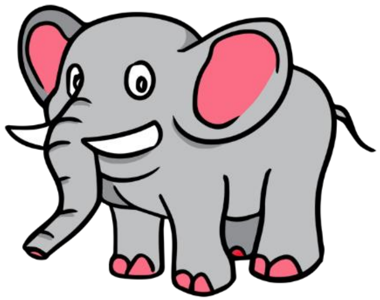


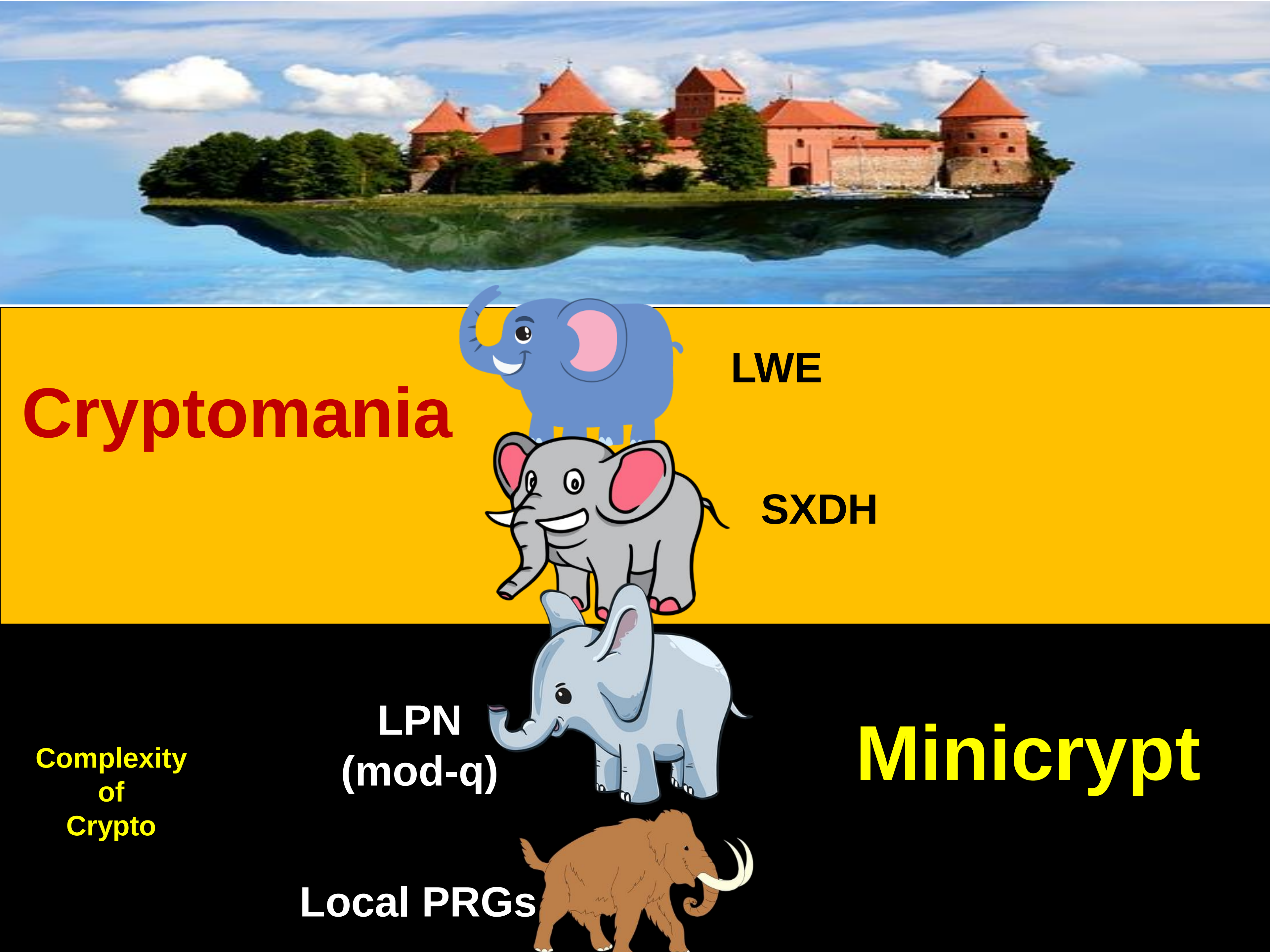
Symmetric eXternal DH [BGdMM05]

$$e: \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$$

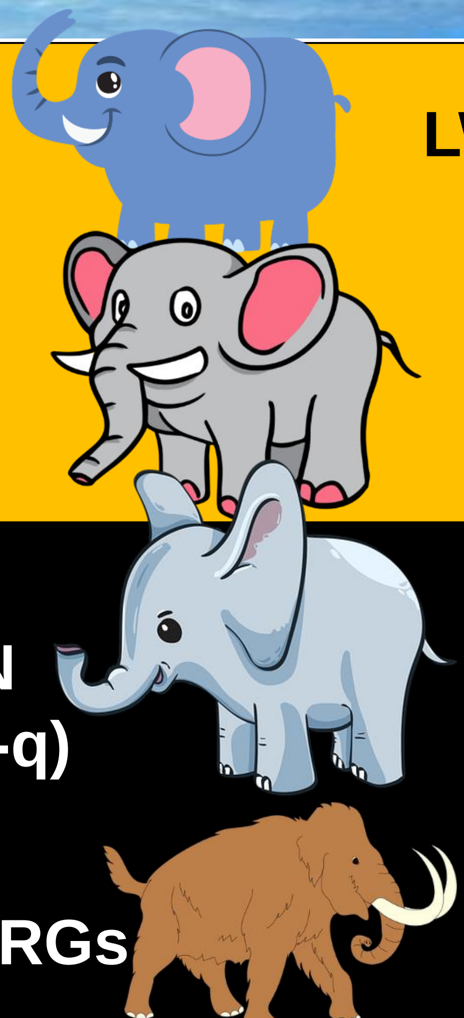
- **SXDH:** DDH is hard in both $\mathbb{G}_1$ and $\mathbb{G}_2$
- Strong form of DDH

  - Can be broken by Quantum adversary
- Standard bilinear assumption
- Groups defined over elliptic curves
- Decisional
- Cryptanalysis by math community?




Cryptomania
LWE
SXDH
LPN
(mod-q)
Minicrypt
Complexity
of
Crypto
Local PRGs

Order?



LWE

SXDH

LPN (mod-q)

Local PRGs

Thank You!