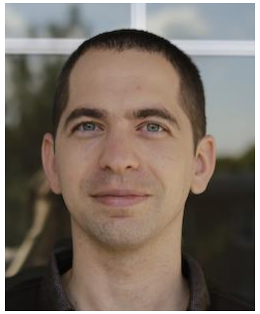
Property Testing with Incomplete or Manipulated Inputs

Sofya Raskhodnikova

BOSTON UNIVERSITY

Based on joint works with:



Omri Ben-Eliezer

Iden Kalemaj

Esty Kelman

Ephraim Linder

Uri Meir

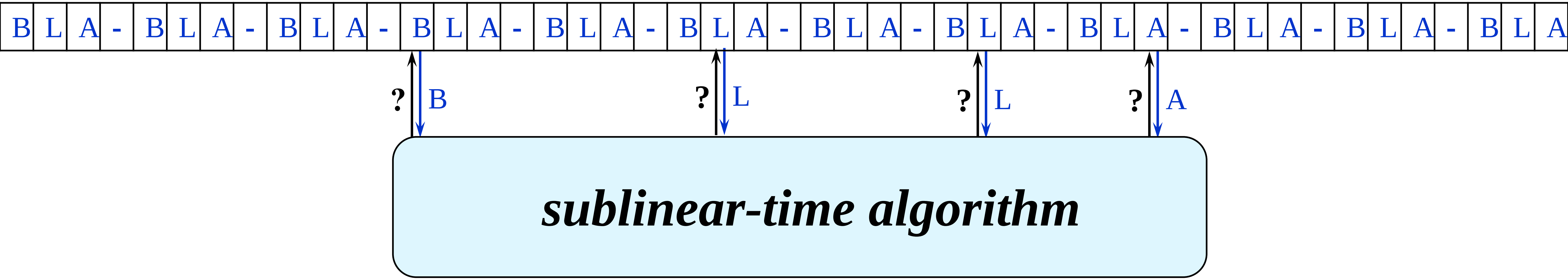
Nithin Varma

Goal: Fundamental Understanding of Sublinear Computation

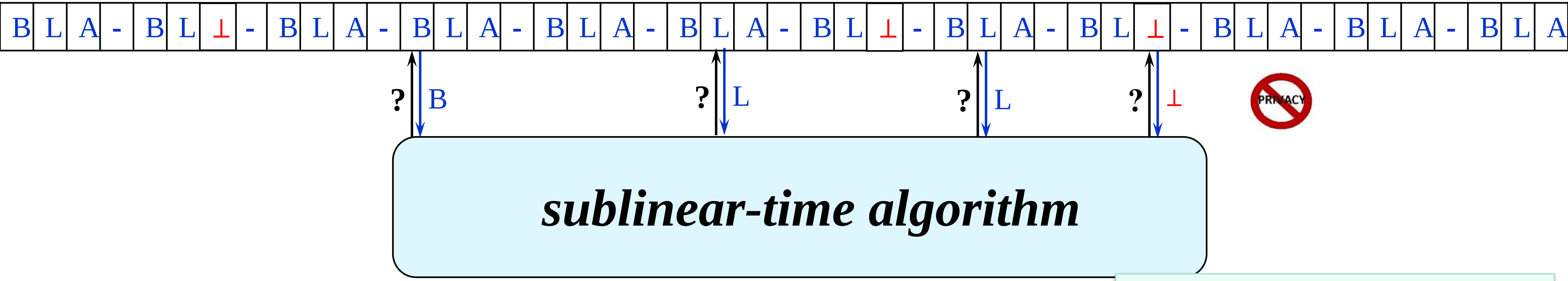
Can we make our computations robust
to adversarial online data manipulations

(specifically, erasures or corruptions)?

Typical access to data



Access to data via an online erasure oracle [Kalemaj Raskhodnikova Varma 22]



erasure budget parameter

- After answering each query, the oracle erases t input characters
- The erasures are performed **adversarially** and **online**, in response to actions of the algorithm

Worst-case analysis circumvents the need to model complex situations

- Oracle knows the description of the algorithm, but not its random coins

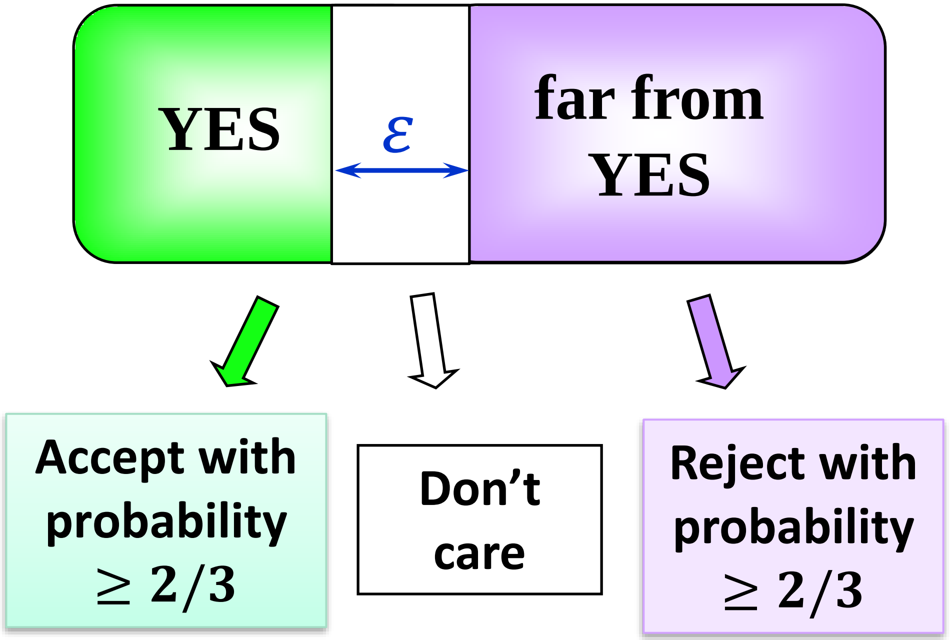
Online corruption oracle is defined analogously, but it modifies the characters instead of erasing them.

Motivating scenarios

- Individuals request that their data be removed from a dataset
  - They are prompted to restrict access to their data after noticing an inquiry into their or other's data (online)
  - General Data Protection Regulation (GDPR) stipulates that data subjects can withdraw previously given consent whenever they want, and their decision must be honored.
- In a criminal investigation / fraud detection setting, a suspect reacts by erasing data after some of their records are pulled by authorities
- In legal setting, an entity is served a subpoena;
  they can destroy related evidence not involved in the subpoena
- In online services, data (such a routes provided by GPS) can change in a complicated way in response to actions of the user

Property testing

Property Tester [Rubinfeld Sudan 96, Goldreich Goldwasser Ron 98]



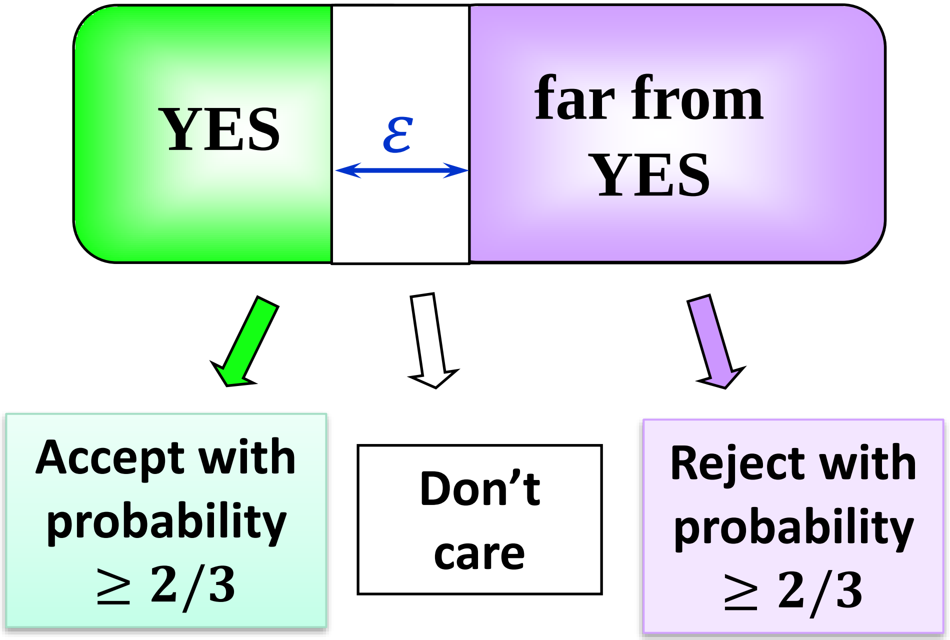
What properties can we test with online erasure/corruption oracle?

How does complexity of testing depend on t?

Two objects are at distance ε = they differ in an ε fraction of places

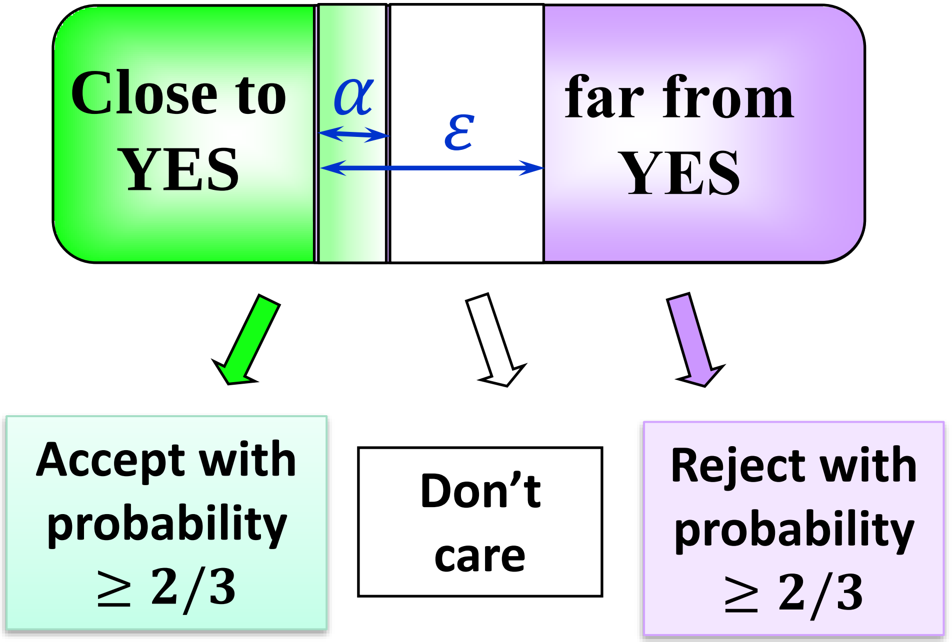
Property testing: offline modifications models

Property Tester [Rubinfeld Sudan 96, Goldreich Goldwasser Ron 98]



Tolerant Property Tester [Parnas Ron Rubinfeld 06]

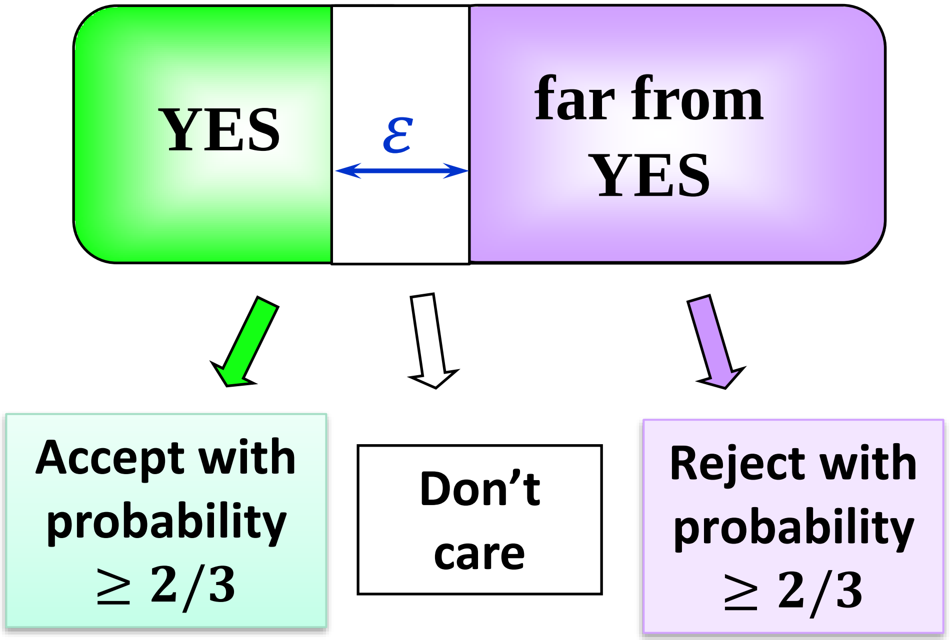
- *Parameters:* $0 < \alpha < \varepsilon < 1$
- $\leq \alpha$ fraction of the input is wrong



Two objects are at distance ε = they differ in an ε fraction of places

Property testing: offline modifications models

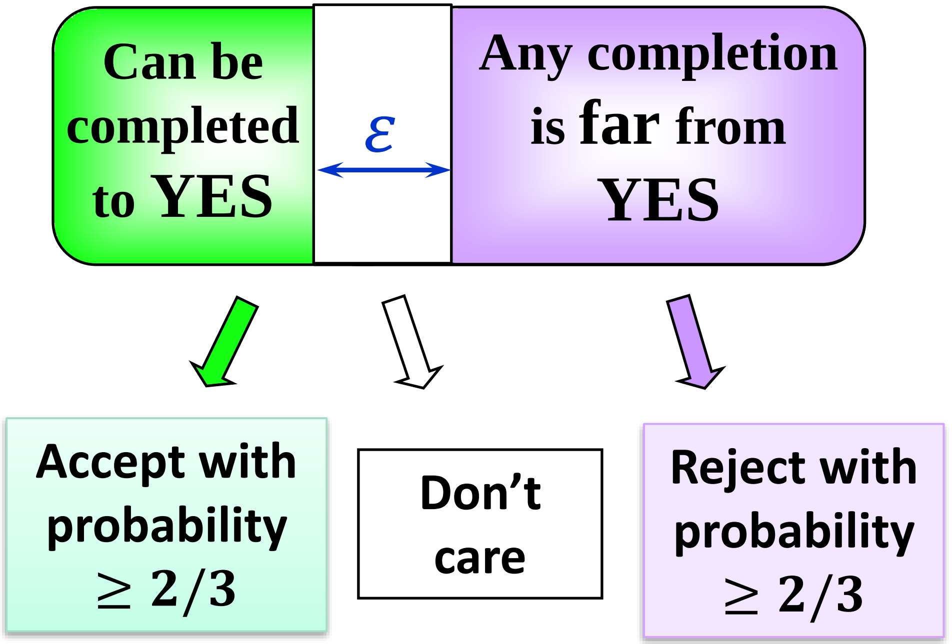
Property Tester [Rubinfeld Sudan 96, Goldreich Goldwasser Ron 98]



Erasure-Resilient Property Tester

[Dixit Raskhodnikova Thakurta Varma 16]

- $\leq \alpha$ fraction of the input is erased adversarially **before** the algorithm runs



Two objects are at distance ε = they differ in an ε fraction of places

Plan: Results in the online-erasures model

- Classical properties that exhibit the extremes in terms of the query complexity
- Separations between the models
- A more nuanced version of the online model
- Connection to Maker-Breaker games

Results in the online erasure model: the extremes

- Some properties can be tested with the ***same*** query complexity as in the standard model (for constant erasure budget t)

  [Kalemaj Raskhodnikova Varma 22, Minzer Zheng 24, Ben-Eliezer Kelman Meir Raskhodnikova 24]:

  – linearity of functions and, more generally, low degree (being of degree at most d)

    - pinning down dependence on t in the query complexity is tricky

- Some properties are ***impossible*** to test, even for $t = 1$ [Kalemaj Raskhodnikova Varma 22]:

  – sortedness and the Lipschitz property of arrays

- Even the simplest tests (i.e., those that sample uniformly and independently at random) cannot necessarily be made resilient to online erasures, even with some loss in query complexity
- The structure of violations to the property plays a role in determining testability

Linearity testing

A function $f\colon\{0,1\}^n \to \{0,1\}$ is linear

- if $f(x) = \sum_{S\subseteq[n]} x[i]$ for some set S of coordinates.
- Equivalently, if $f(x) + f(y) = f(x+y)$ for all x, y in domain.

computations are over $\mathbb{F}_2$

Standard Model	**Online-Erasures Model**
[Blum Luby Rubinfeld 93, Bellare Coppersmith Hastad Kiwi Sudan '96] $\Theta\left(\frac{1}{\varepsilon}\right)$ queries	[Kalemaj Raskhodnikova Varma 22, Ben-Eliezer Kelman Meir Raskhodnikova 24] $\Theta\left(\frac{1}{\varepsilon} + \log t\right)$ queries
BLR Tester: • Sample $x, y \sim \{0,1\}^n$ u.i.r. • Query f on $x, y,$ and $x+y$ • Reject if $f(x) + f(y) \neq f(x+y)$.	Issue with standard linearity tester: • Query x. Receive $f(x)$. • Query y. Receive $f(y)$. • Oracle erases $x+y$.
Thm. If $f\colon\{0,1\}^n \to \{0,1\}$ is ε-far from linear then an $\Omega(\varepsilon)$ fraction of pairs (x, y) violate linearity.	Thm. If $f\colon\{0,1\}^n \to \{0,1\}$ is ε-far from linear then, for all even k, an $\Omega(k\varepsilon)$ fraction of k-tuples $(x_1, x_2, \ldots, x_k)$ violate linearity.

$f(x_1) + \cdots + f(x_k) \neq f(x_1 + \cdots + x_k)$

More options for the algorithm!

Online-erasure-resilient linearity tester

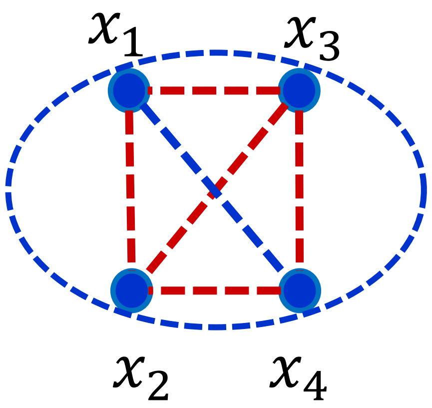
[Kalemaj Raskhodnikova Varma 22, Ben-Eliezer Kelman Meir Raskhodnikova 24]

Tester (**Parameters:** $\epsilon \in (0,1)$, erasure budget t)

1. Query $k = \Theta\left(\frac{1}{\epsilon} + \log t\right)$ points $x_1, \ldots, x_k \in \{0,1\}^n$ u.i.r.
2. Sample a uniformly random $S \subset [k]$ of even size
3. Query $y = \sum_{i \in S} x_i$
4. **Reject** if $\sum_{i \in S} f(x_i) \neq f(y)$ (and all points are non-erased)

Query a reserve of k points

Example:
erasure budget $t = 2$
$k = 4$



$\Theta(2^k)$ options for the last query with our structural theorem instead of $\Theta(k^2)$ with BLR

Takeaways from the analysis of linearity tester

Structural theorem

If $f: \{0,1\}^d \to \{0,1\}$ is ε-far from linear then, for all even k, an $\Omega(k\varepsilon)$ fraction of k-tuples $(x_1, x_2, \ldots, x_k)$ violate linearity.

$f(x_1) + \cdots + f(x_k) \neq f(x_1 + \cdots + x_k)$

- Proved via Fourier analysis
- Gives a new optimal linearity tester in the standard model:

Query a k-tuple $(x_1, \ldots, x_k)$, where $k = \Theta\left(\frac{1}{\epsilon}\right)$ and even, and check if it violates linearity

Non-erasure lemma

The tester is unlikely to query an erased point.

- Intuition for the proof: there are many options for the last query.
- This lemma allows us to show that our linearity tester is online-corruption-resilient

Linearity testing: Lower bound

[Kalemaj Raskhodnikova Varma 22]

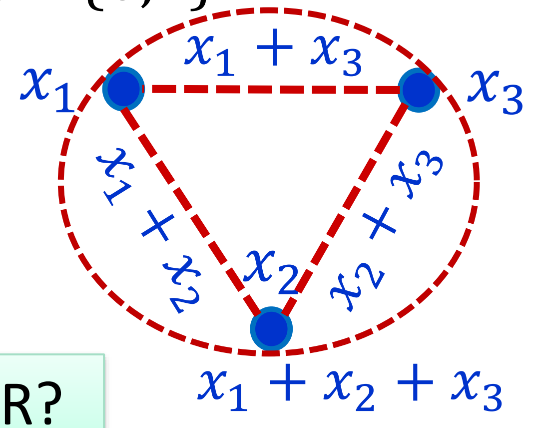
Theorem

t is the erasure budget

Every online-erasure-resilient linearity tester must make $\Omega(\log t)$ queries.

Proof idea (can be made formal via Yao's minimax principle adapted to our setting):

- Oracle $\mathcal{O}$: erase t sums of previous queries of the tester (in some specific order)
- If tester makes $q < \log_2 t$ queries, oracle can erase all their ($< 2^q$) sums
- Tester only sees function values on linearly independent vectors from $\{0,1\}^n$
- The view of the tester is the same whether the input is a random linear function or a random function
- A random function is far from linear.



Question: Could we have used only pair queries in the tester, like in BLR?

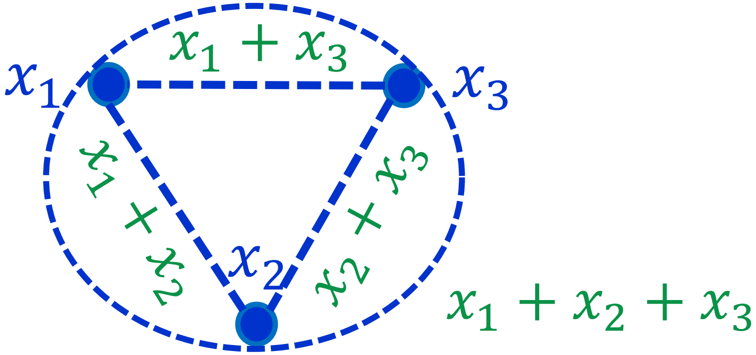
Answer: Then the dependence on t would be at best t, by a similar argument

Low-degree testing

A function $f: \{0,1\}^n \to \{0,1\}$ has degree at most d if it can be expressed as a polynomial of degree at most d in variables $x[1], \ldots, x[n]$.

computations are over $\mathbb{F}_2$

Standard Model	Online-Erasures Model
[..., Alon Kaufman Krivelevich Litsyn Ron 05, Bhattacharyya Kopparty Schoenebeck Sudan Zuckerman 10] $\Theta(1/\varepsilon + 2^d)$ queries	[Minzer Zheng 24, Ben-Eliezer Kelman Meir Raskhodnikova 24] $O\left(\frac{1}{\varepsilon}\log^{3d+3}\frac{t}{\varepsilon}\right)$ and $\Omega(\log^d t)$ queries
AKKLR tester: • *Sample* $d+1$ *points from* $\{0,1\}^n$ *u.i.r.* • *Query f on all their linear combinations* • *Reject if the sum of the returned values is 1*	[Minzer Zheng] tester (idea): • There are many low-degree testers. • Pick points u.i.r. inside an affine subspace of large enough dimension in terms of t *and d* • Find a tester that uses these points.

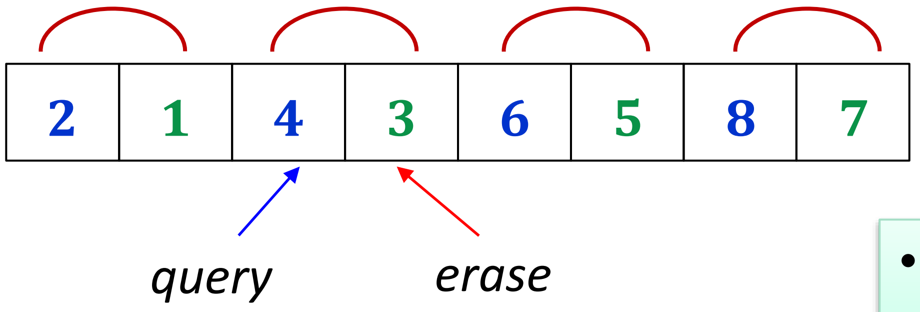


Gives a new tester for the standard model with u.i.r. queries over an affine subspace.

Impossibility of testing sortedness

- An array $f\colon [n] \to \mathbb{N}$ is sorted if $f(x) \le f(y)$ for all $x < y$.

Standard Model	Offline-Erasures Model	Tolerant Testing / Distance Approximation	Online-Erasures Model
[Ergun Kannan Kumar Rubinfeld Viswanathan 00, Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky 99, Fischer 06, Bhattacharyya Grigorescu Jung Raskhodnikova Woodruff 12, Chakrabarty Seshadhri 18, Belovs 18,…] $\Theta\left(\frac{\log \varepsilon n}{\varepsilon}\right)$ queries $O\left(\sqrt{n/\varepsilon}\right)$ uniform iid queries	[Dixit Raskhodnikova Thakurta Varma '18] $O\left(\frac{\log n}{\varepsilon}\right)$ queries	[Saks Seshadhri 17,…] $\left(\frac{1}{\varepsilon}\right)^{O\left(\frac{1}{\varepsilon}\right)}$ polylog n	[Kalemaj Raskhodnikova Varma 22] **Impossible to test**



- This array is $\frac{1}{2}$-far from sorted, but an online tester will see no violations

- Here all violations are disjoint
- In linearity and low-degree, violations overlap with each other

Plan: Results in the online-erasures model

✓ Classical properties that exhibit the extremes in terms of the query complexity

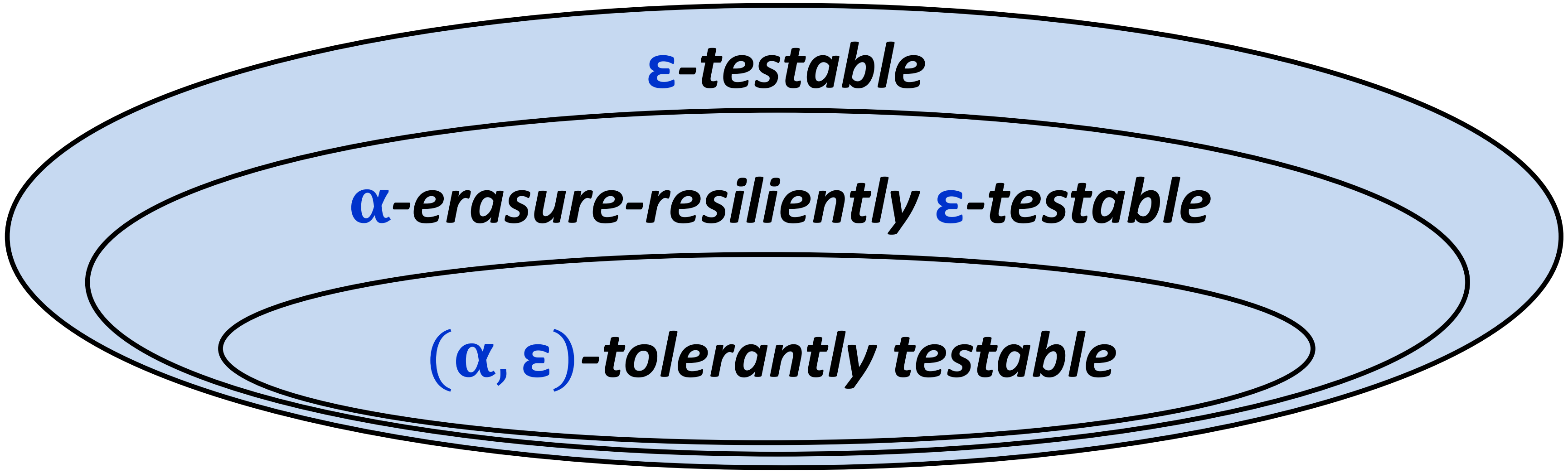
  – linearity, low-degree, sortedness

- Separations between the models
- A more nuanced version of the online model
- Connection to Maker-Breaker games

Comparison: Relationships between offline testing models

Containments are strict:

- [Fischer Fortnow 05]: standard vs. tolerant
- [Dixit Raskhodnikova Thakurta Varma 16]: standard vs. erasure-resilient
  [Ben-Eliezer Fischer Levi Rothblum 20]: improvements in the gap
- [Raskhodnikova Ron-Zewi Varma 19]: erasure-resilient vs. tolerant



Connections to PCPs and locally decodable error-correcting codes.

Separations between the online and offline models

[Kelman Linder Raskhodnikova]

Sortedness is testable with offline erasures, but not with online erasures.

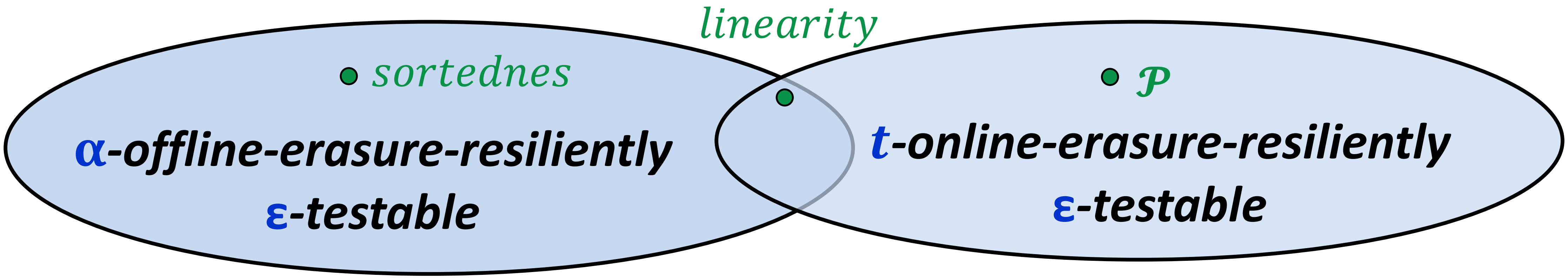
Is the online-erasures model strictly harder?

Answer: **No**, there is a **query separation** in the other direction.

Theorem on query separation

For every $\alpha \in (0,1)$ and $t \in \mathbb{N}$, there exists a property $\mathcal{P}$ on n-bit strings such that

- $\mathcal{P}$ is **online**-erasure-resiliently testable (with t erasures per query) with a constant number of queries.
- Every **offline**-erasure-resilient tester for $\mathcal{P}$ that works with α fraction of corruptions needs $\widetilde{\Omega}\left(\frac{n}{t}\right)$ queries.



Separations between the online and offline models

[Kelman Linder Raskhodnikova]

Online testers we saw use more randomness than offline testers for the same property.

Is it intrinsic?

Answer: **Yes**, there is a **randomness separation**

- In the offline models, only a logarithmic number of random bits is needed:
  [Goldreich Sheffet 10] Any randomized oracle machine that solves a promise problem on input in $[k]^n$ can be simulated using $\log n + \log\log k \ + O(1)$ random bits.

Theorem on randomness separation

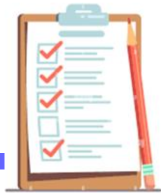
For every $\alpha \in (0,1)$ and $t \in \mathbb{N}$, there exists a property $\mathcal{P}$ which is

- testable with the same query complexity in the online and offline models
- $O(\log n)$ random bits are sufficient **offline**,
  but $\Omega(n^c \log(t+1))$ random bits are needed **online** (for some constant c)

offline

online

Plan: Results in the online-erasures model



- ✓ Classical properties that exhibit the extremes in terms of the query complexity
  - linearity, low-degree, sortedness
- ✓ Separations between the models
  - query separation and randomness separation
- A more nuanced version of the online model
- Connection to Maker-Breaker games

More nuanced version of the online erasure model

[Ben-Eliezer Kelman Meir Raskhodnikova 24]

- Overcomes the impossibility results in [Kalemaj Raskhodnikova Varma 22]

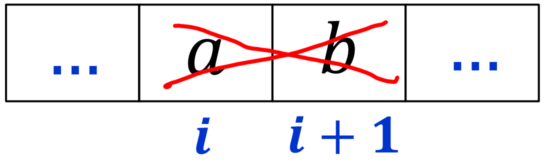
Considers

- *batch queries* (with erasures performed only between the batches)
- rates of erasure less than 1 (e.g., every other query)
- different types of adversarial strategies:

  fixed-rate (as in [KRV22]) vs. *budget-managing* (the adversary can postpone erasures arbitrarily)

Phase transitions for local properties

A property $\mathcal{P}$ of sequences $f:[n] \to \mathbb{R}$ is local if
there exists a family $\mathcal{F}$ of forbidden pairs $(a,b) \in \mathbb{R}^2$ such that
$f \in \mathcal{P} \Leftrightarrow \forall i \in [n-1] \forall (a,b) \in \mathcal{F}: \big(f(i), f(i+1)\big) \neq (a,b)$



Examples

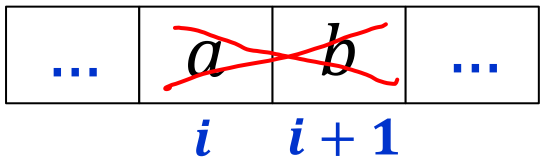
- Sortedness: $\mathcal{F}=\{(a,b): a > b\}$
- Lipschitz: $\mathcal{F}=\{(a,b): |a-b| > 1\}$

[Ben-Eliezer 19], generalizing previous work:

All local properties are testable with $O\left(\frac{\log \varepsilon n}{\varepsilon}\right)$ queries in the standard model.

Phase transitions for local properties

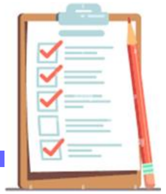
A property $\mathcal{P}$ of sequences $f: [n] \to \mathbb{R}$ is local if
there exists a family $\mathcal{F}$ of forbidden pairs $(a, b) \in \mathbb{R}^2$ such that
$$f \in \mathcal{P} \Leftrightarrow \forall i \in [n-1] \forall (a,b) \in \mathcal{F}: \ (f(i), f(i+1)) \neq (a,b)$$



Batch Size	Fixed-rate adversary	Budget-managing adversary
1	0 to 1: $O\left(\frac{\log \varepsilon n}{\varepsilon(1-t)}\right)$ queries; beyond 1 (t): impossible to test	0 to $\Theta(\varepsilon)$: $O\left(\frac{\log \varepsilon n}{\varepsilon}\right)$ queries; beyond $\Theta(\varepsilon)$ (t): impossible to test
2	0 to $\tilde{\Omega}(\varepsilon^2 n)$ (t): $O\left(\frac{\log \varepsilon n}{\varepsilon}\right)$ queries	

- Phase transition results hold both for erasures and for corruptions

Plan: Results in the online-erasures model

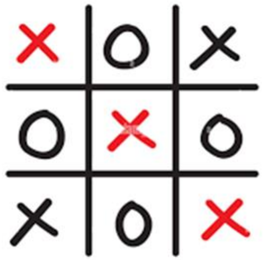


- ✓ Classical properties that exhibit the extremes in terms of the query complexity
  - linearity, low-degree, sortedness
- ✓ Separations between the modelsmodels
  - query separation and randomness separation
- ✓ A more nuanced version of the online models
  - fixed-rate vs. budget-managing adversary; rates of erasure; batch queries
- Connection to Maker-Breaker games

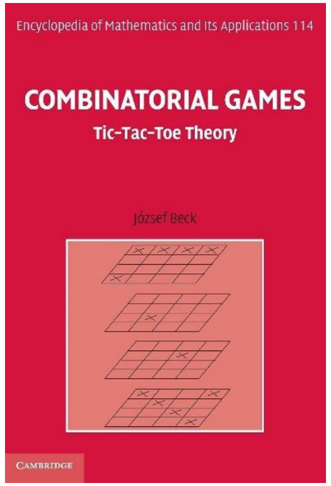
Connection to Maker-Breaker games

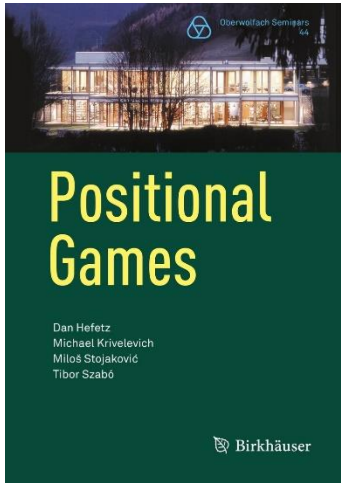
[Ben-Eliezer Kelman Meir Raskhodnikova 24]

- Positional games are central in combinatorics (see textbooks [Beck08, Hefetz Krivelevich Stojaković Szabó 14])



- Maker-Breaker games are a prominent and widely investigated example.



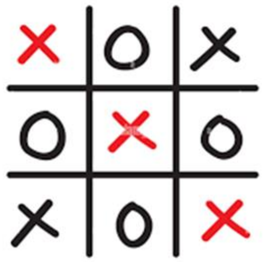


An $(s\colon t)$ Maker-Breaker game is defined by a finite set X of board elements and a family $W \subseteq 2^X$ winning sets.

- Two players, Maker and Breaker, take turns claiming unclaimed elements of X.
- Maker claims s elements on each turn; Breaker claims t
- Maker wins if she manages to claim all elements of a winning set; o.w. Breaker wins

Connection to Maker-Breaker games

[Ben-Eliezer Kelman Meir Raskhodnikova 24]



An $(s{:}\,t)$ Maker-Breaker game

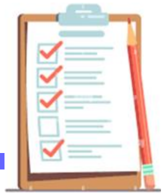
is defined by a finite set X of board elements and a family $W \subseteq 2^X$ winning sets.

- Two players, Maker and Breaker, take turns claiming unclaimed elements of X.
- Maker claims s elements on each turn; Breaker claims t
- Maker wins if she manages to claim all elements of a winning set; o.w. Breaker wins

- In online testing:
  - algorithm is the Maker, adversary is the Breaker
  - the domain of the input function is the set of board elements
  - witness are winning sets.
- A big complication is that the tester does not know in advance which sets are in W.
- A prerequisite for designing an online tester:
  - identify the general structure of the sets in W
  - and a winning strategy for Maker.

Online-erasures model motivates studying new Maker-Breaker games

Plan: Results in the online-erasures model



✓ Classical properties that exhibit the extremes in terms of the query complexity
  - linearity, low-degree, sortedness

✓ Separations between the modelsmodels
  - query separation and randomness separation

✓ A more nuanced version of the online modelmodels
  - fixed-rate vs. budget-managing adversary; batch queries

✓ Connection to Maker-Breaker games

Open questions

- Online manipulation-resilient testers for specific properties
- An investigation of the threshold for t, the rate of erasures, in phase transitions
  - What is t_{max} for which a given property is testable?
  - What is the query complexity as we approach t_{max}?
- Some general characterization of properties testable with online erasures?
  - Maybe, in terms of the structure of witnesses
- More techniques for the online-corruptions model?
  - All testability results so far rely on algorithms that are unlikely to see a manipulated point
- Online-erasure-resilient algorithms for tasks other than property testing?