Coding Theory in **Almost Linear Time** and **Sublinear Space**

Dana Moshkovitz

UT Austin

Based on joint works with
Joshua Cook (UT Austin)

Algorithmic Coding Theory

$C:\{0,1\}^k \rightarrow \{0,1\}^n$

- Constant rate $n=O(k)$
- Constant relative distance $d=\Omega(n)$
- Encoding complexity
- Decoding complexity

Time $n^{1+o(1)}$

Space $n^{o(1)}$

Our Results

1. Code with deterministic **encoding** in time $n^{1+o(1)}$ and space ~logn.
   - Impossible without random access to input.
2. Code with deterministic **decoding** in time $n^{1+o(1)}$ space $n^{o(1)}$.
   - Follows from locally correctable codes and new efficient derandomization.

Non uniform

Still open: A code that can be encoded and decoded simultaneously in efficient time-space.

Time-Space Efficient **Randomized** Correction

Follows from locally decodable codes (correctable)

There are efficient randomized decoders with $n^{o(1)}$ queries for asymptotically good codes [Kopparty-Saraf-Yekhanin'11, Guo-Kopparty-Sudan'13, Hemenway-Ostrovsky-Wootters'13,Kopparty-Meir-RonZewi-Saraf'16].

We can decrease their error probability to $\ll 1/n$ by repetition. Then they give **randomized** decoders in time $n^{1+o(1)}$ and space $n^{o(1)}$.

Time-Space Efficient **Deterministic** Decoding

- Existing locally correctable codes give **non-adaptive** (non-uniform) **deterministic** decoders that run in time $n^{2+o(1)}$ and space $n^{o(1)}$.
  - Since only $O(n)$ randomness strings are needed for the $\exp(n)$ possible corrupted codewords.
- Gronemeier '06: **Non-adaptive deterministic** decoders that run in time $n^{1+\delta}$ must use space at least $n^{1-\delta}$.
- Is there a quadratic time lower bound for **all deterministic** decoders that use space $n^{o(1)}$?

Randomization Speed-up?

- **Efficient derandomization** [Nisan-Wigderson'88, Impagliazzo-Wigderson'97,…,Doron-Moshkovitz-Oh-Zuckerman'20, Chen-Tell'21-22]: Under plausible assumptions:

  time-**t** space-**s** randomized ⇒ time≈**tn** space-**s** deterministic

Is this tight?

	Randomized Time with $n^{o(1)}$ space	Deterministic Time with $n^{o(1)}$ space
Local Decoding	$n^{o(1)}$	$\Omega(n)$

$n^{1+o(1)}$ vs. $n^{2-o(1)}$ for PIT only known under **NSETH**: #SAT requires $2^{(1-o(1))n}$ non-det time [Williams'16]

Theorem: There exists an asymptotically good error correcting code with a (non-uniform) decoder running in time $n^{1+o(1)}$ and space $n^{o(1)}$.

Typical locally correctable code

Perfect completeness;
Non-adaptive;
Smooth; Systematic;

→

Time $n^{1+o(1)}$ space $n^{o(1)}$ (non-uniform) decoder

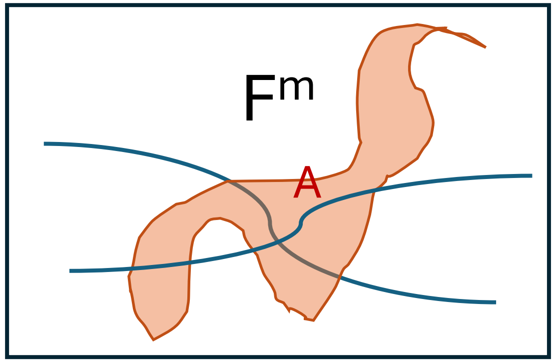
Uniform Decoders?

To get **uniform** decoder for Reed-Mueller code need better **curve samplers**.

Specifically, $|F|^{m+O(k)}$ degree-k curves in F^m so for every $A \subseteq F^m$ of fraction μ, it holds

$$P_c(|c \cap A| >> \mu|F|) < \mu|F|^{-\Omega(k)}.$$

[TaShma-Umans'06, Guo'13]: $|F|^{O(m+k)}$ curves of degree poly(k) with sampling error $|F|^{-\Omega(k)}$.



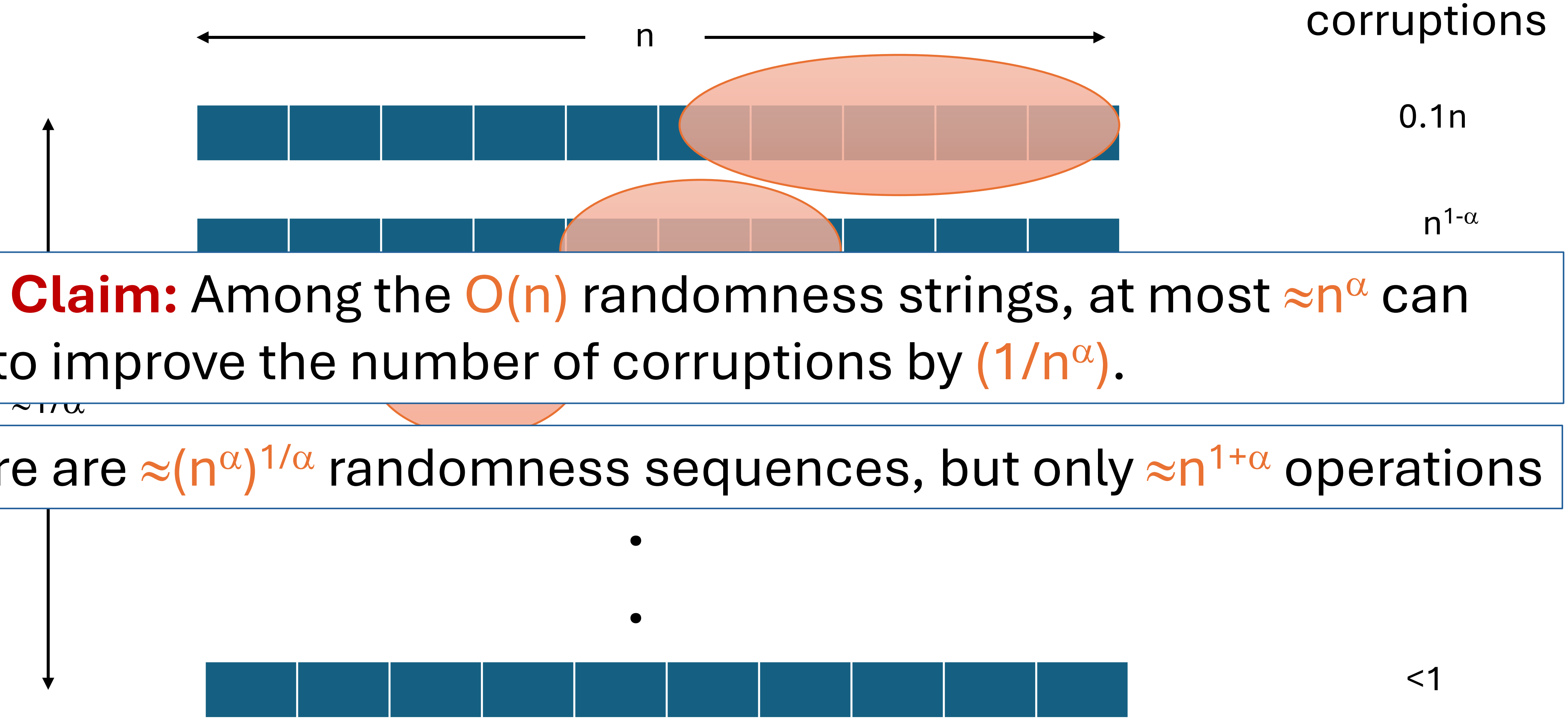
Locally Testing Typical Locally Correctable Codes

Lemma: For a typical locally correctable code C (perfect completeness, non-adaptive, smooth, systematic), local T that **for w with $dist(w,C)<0.1$**,

$$\tfrac{1}{2}dist(w,C) < P(T \text{ accepts}) < 2dist(w,C).$$

Again, $O(n)$ randomness strings suffice since there are $exp(n)$ possible w. Hence, one can estimate $dist(w,C)$ **deterministically** non-uniformly in time $n^{1+o(1)}$ and space $n^{o(1)}$.

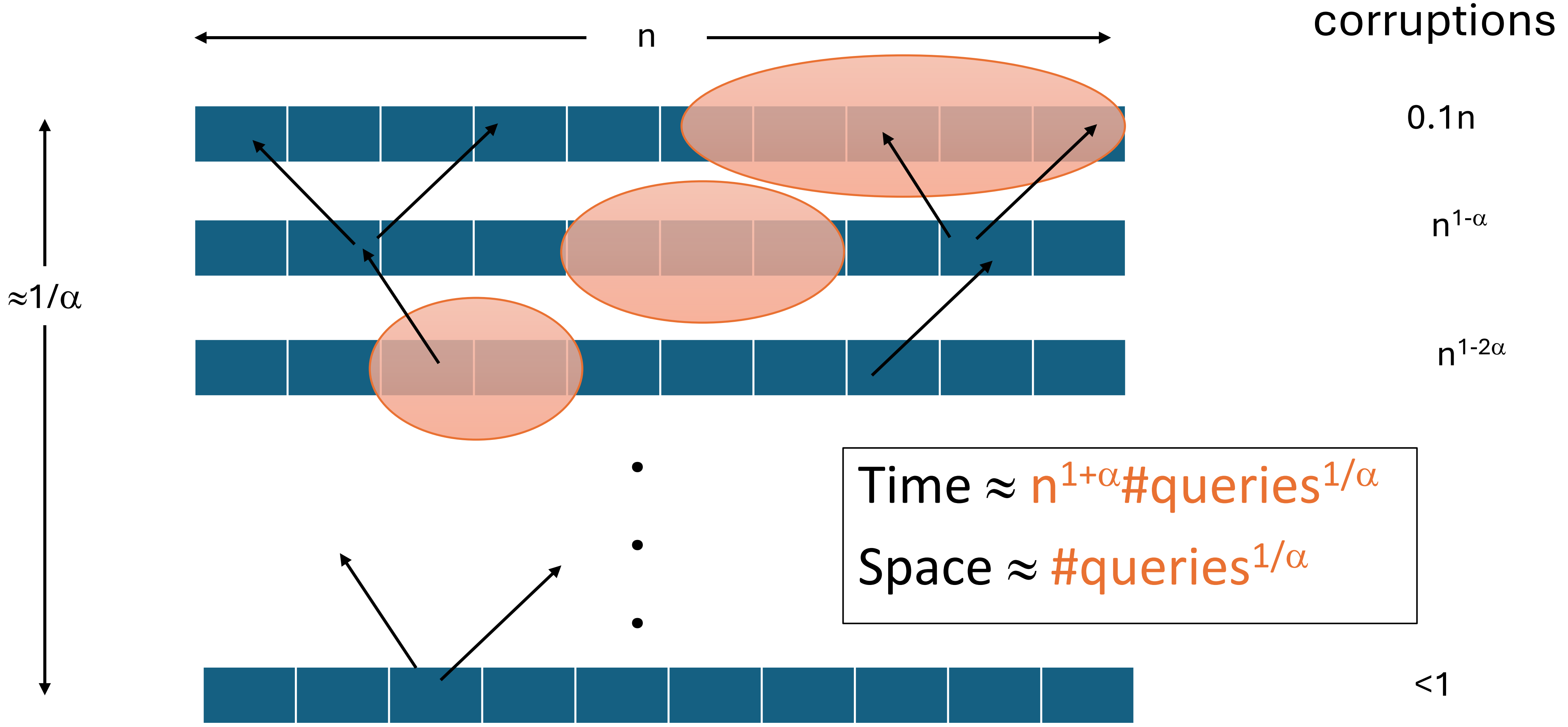
The Iterative Correction Method



Key Claim: Among the $O(n)$ randomness strings, at most $\approx n^{\alpha}$ can fail to improve the number of corruptions by $(1/n^{\alpha})$.

There are $\approx (n^{\alpha})^{1/\alpha}$ randomness sequences, but only $\approx n^{1+\alpha}$ operations

Time-Space Efficient Deterministic Decoder



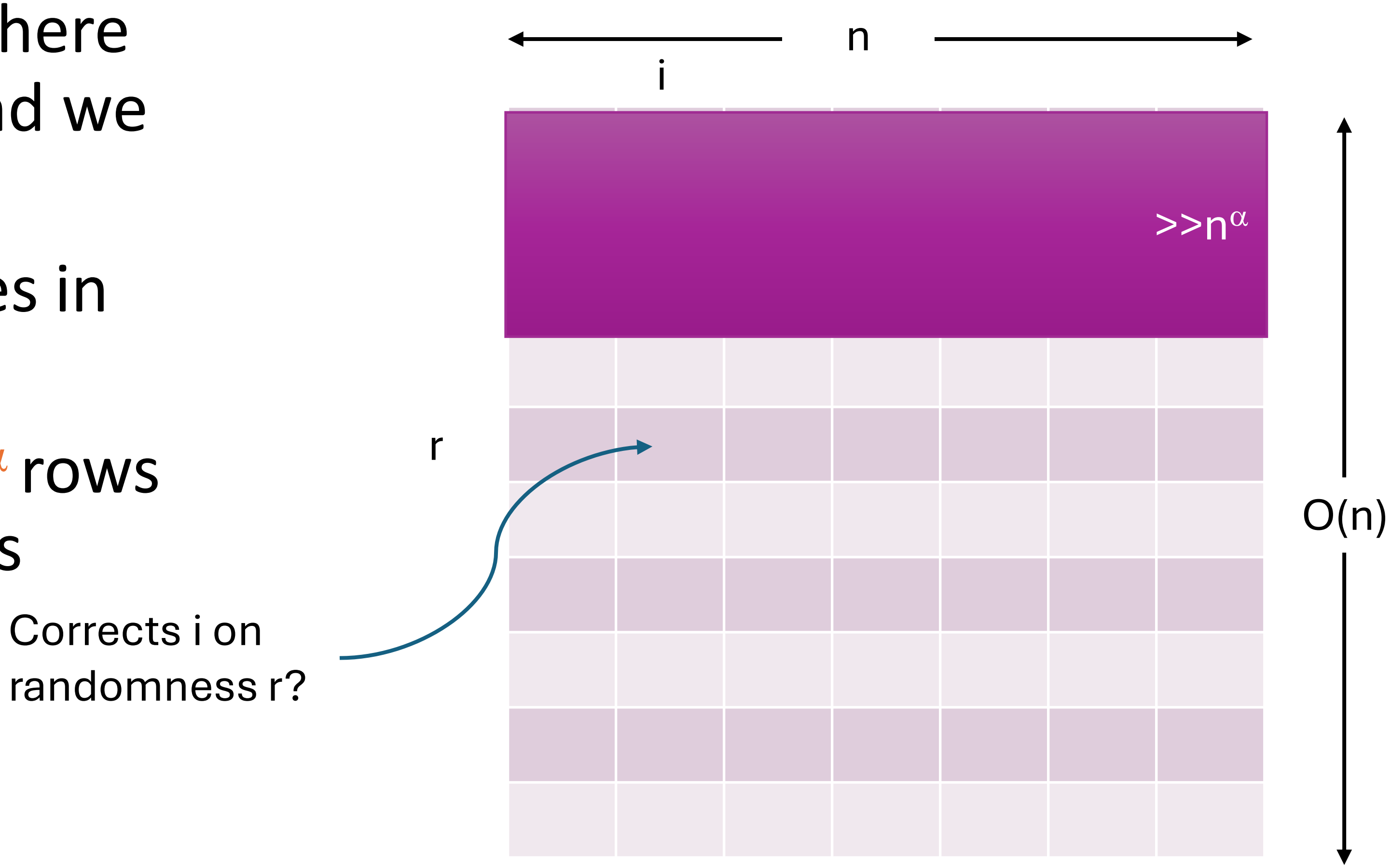
Time ≈ $n^{1+\alpha}\#\text{queries}^{1/\alpha}$

Space ≈ $\#\text{queries}^{1/\alpha}$

Proof of Key Claim: $\approx n^\alpha$ Randomness Strings Suffice For $1/n^\alpha$ Less Corruptions

For simplicity, assume there are $\Omega(n)$ corruptions and we want $O(n^{1-\alpha})$.

- $O(n)$ correction failures in the entire table.
- Thus, can't have >> n^α rows contribute $n^{1-\alpha}$ failures each.



Our Results

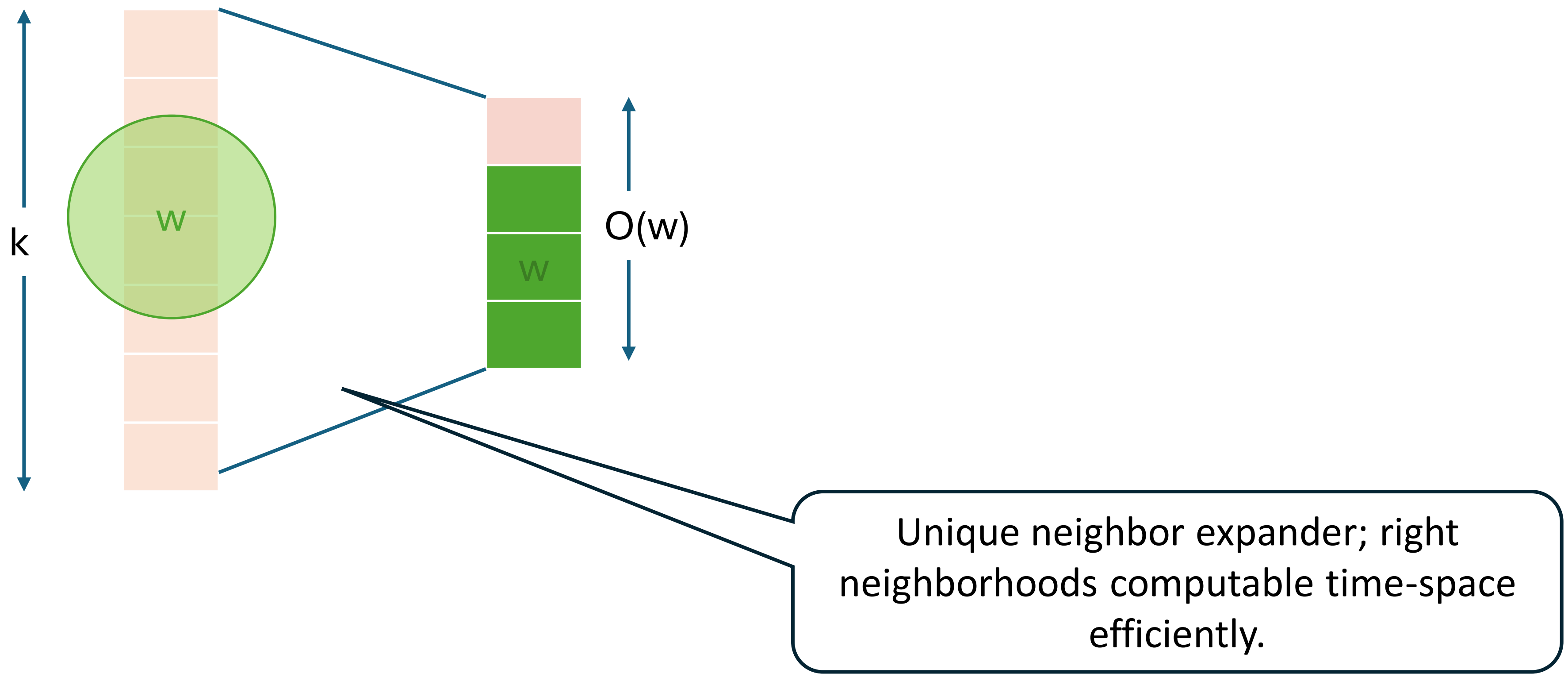
1. Code with deterministic **encoding** in time $n^{1+o(1)}$ and space ~logn.
   - Impossible without random access to input.
2. Code with deterministic **decoding** in time $n^{1+o(1)}$ space $n^{o(1)}$.
   - Follows from locally correctable codes and new efficient derandomization.

Non uniform

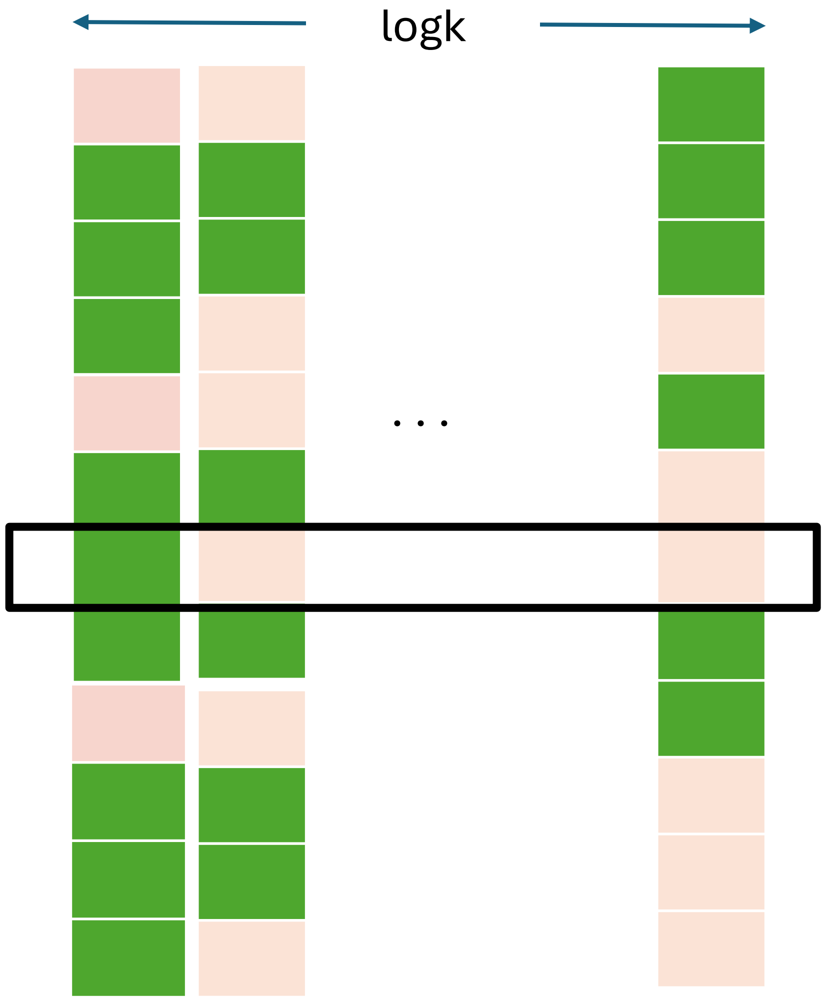
Still open: A code that can be encoded and decoded simultaneously in efficient time-space.

Time-Space Efficient Encoding via Expanders

We'll construct a linear code. Assume message has w=o(k) non-zeros.



Time-Space Efficient Encoding



1. Per approximate weight w, hash.
2. Repeat so n bits per w.
3. Encode each row.