High Rate Polynomial Evaluation Codes

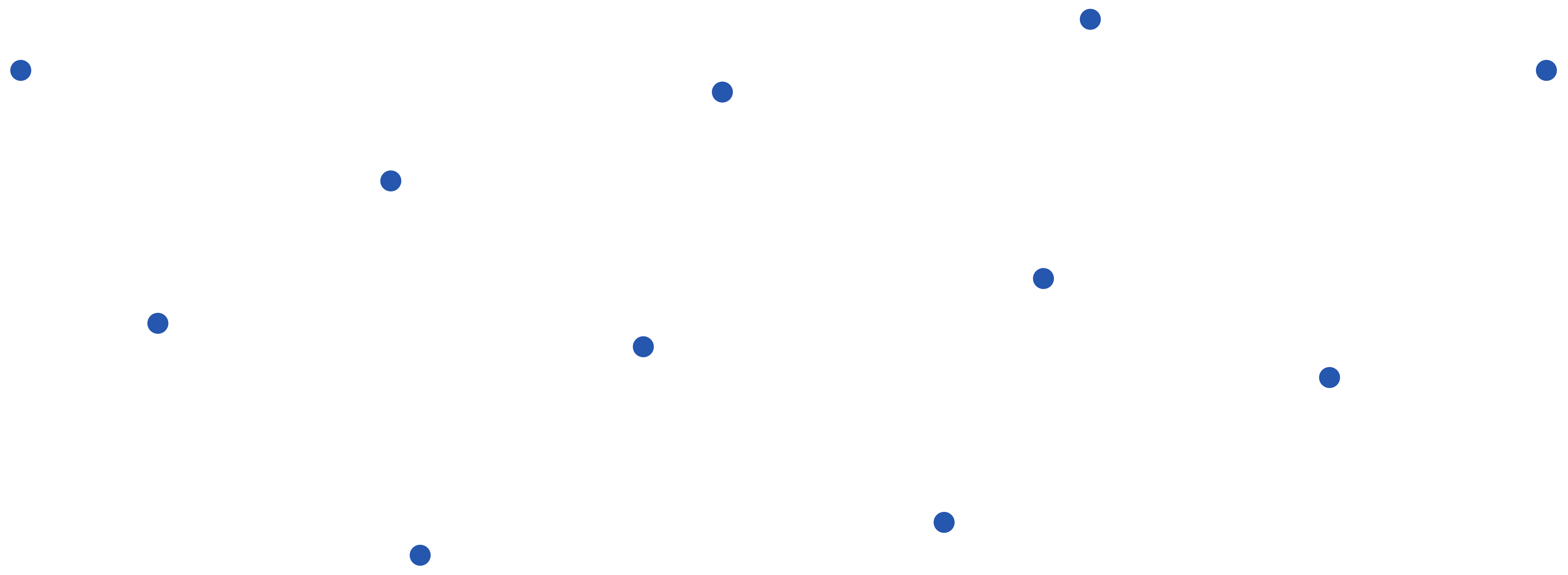
Swastik Kopparty, Mrinal Kumar, and Harry Sha

Error Correcting Codes

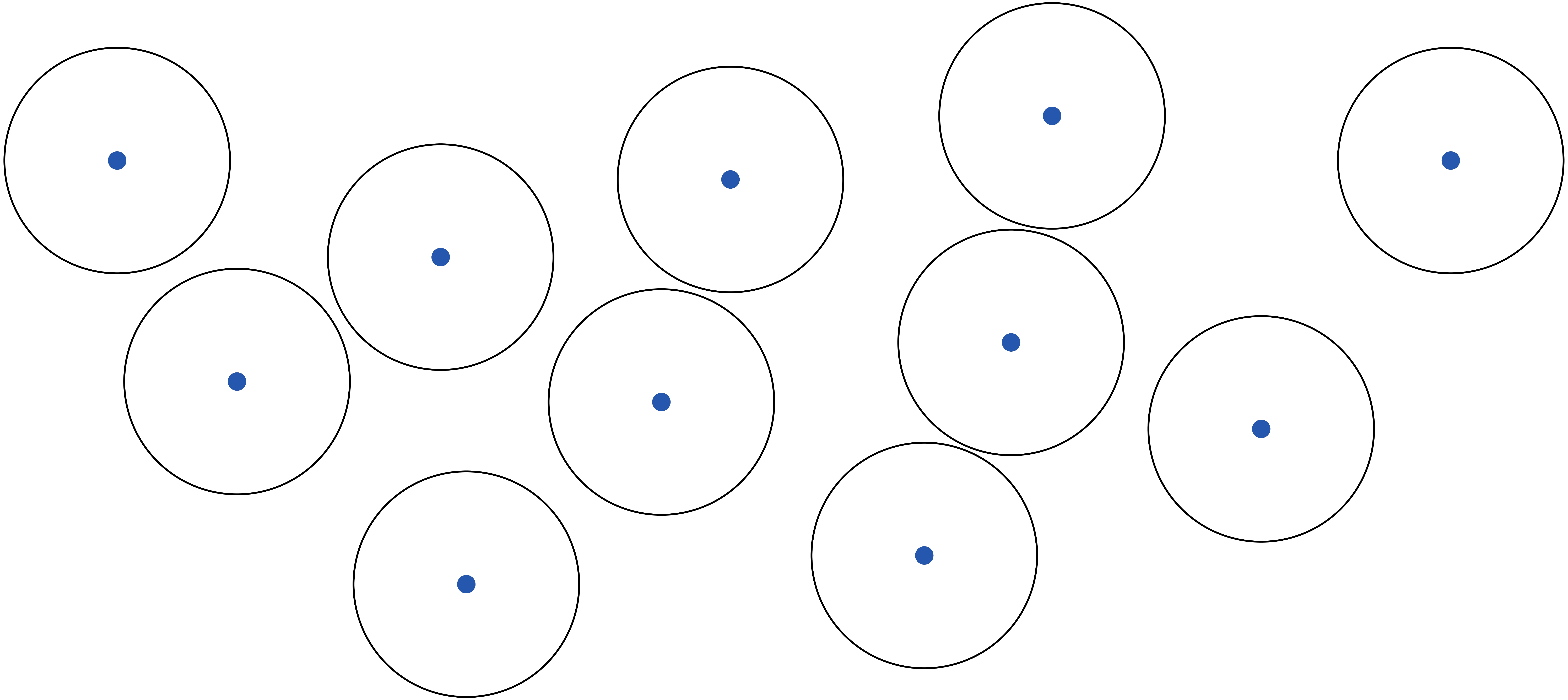
- **Goal:** Want to encode messages into codewords such that even if there are some corruptions, we can still recover the original message.

- This corresponds to the mathematical problem of finding a subset $C \subseteq \Sigma^n$, such that for every distinct $x, y \in C$, x and y are far in the Hamming distance (differ in many coordinates).

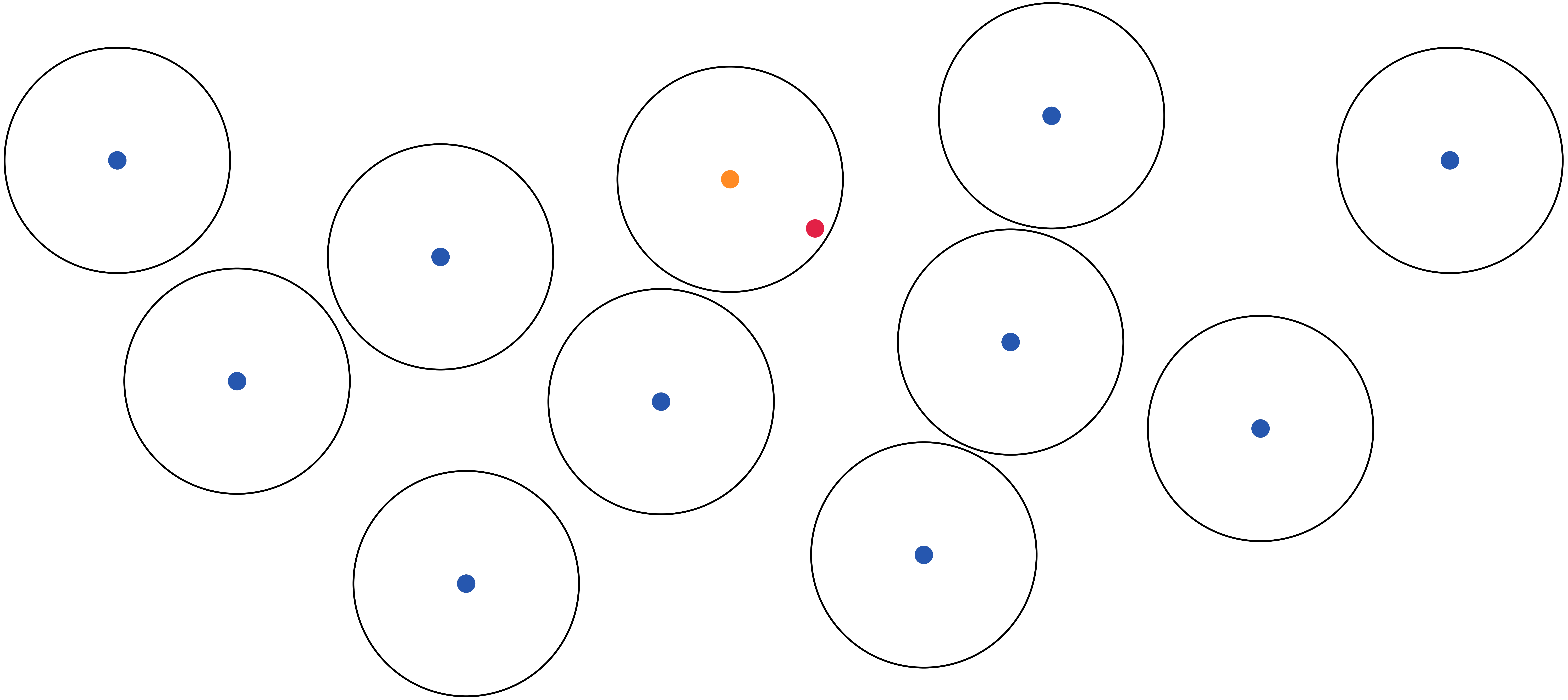
Picture



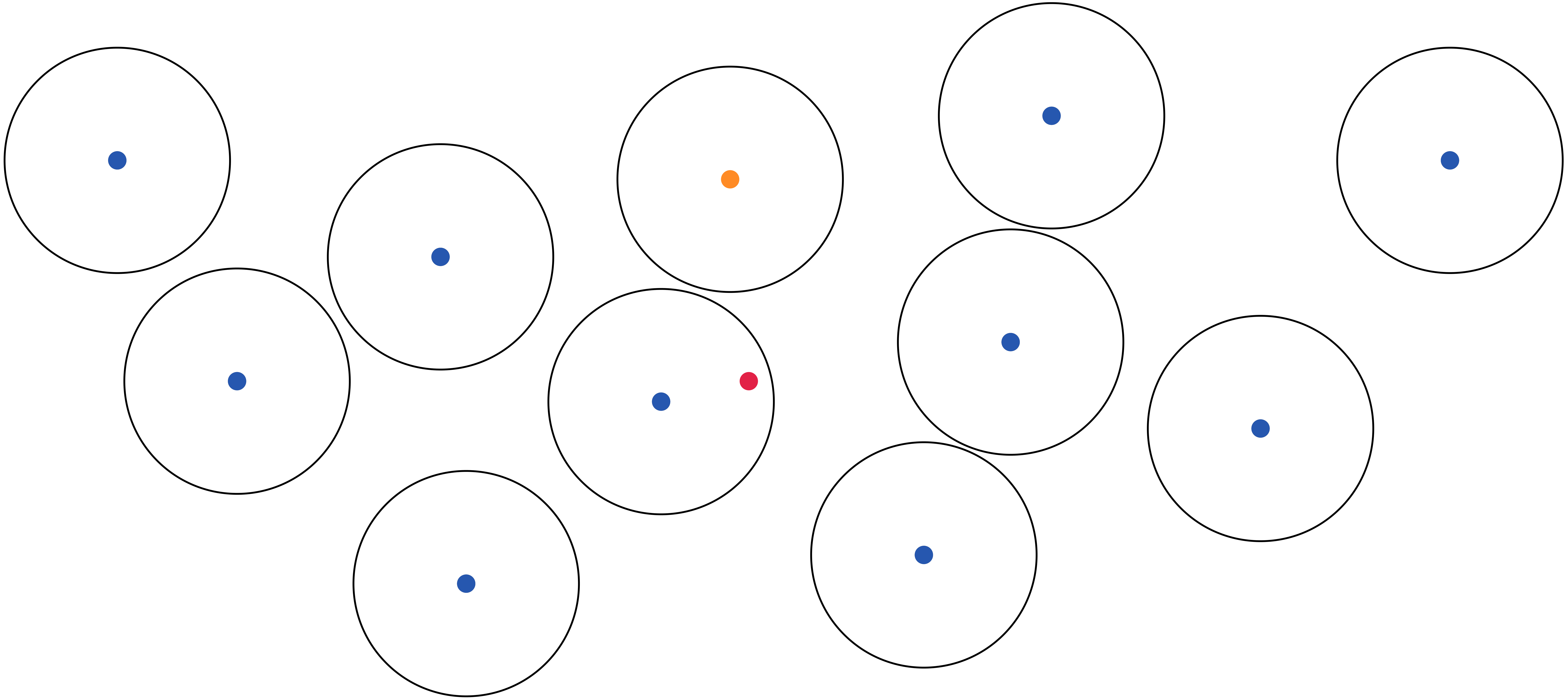
Picture



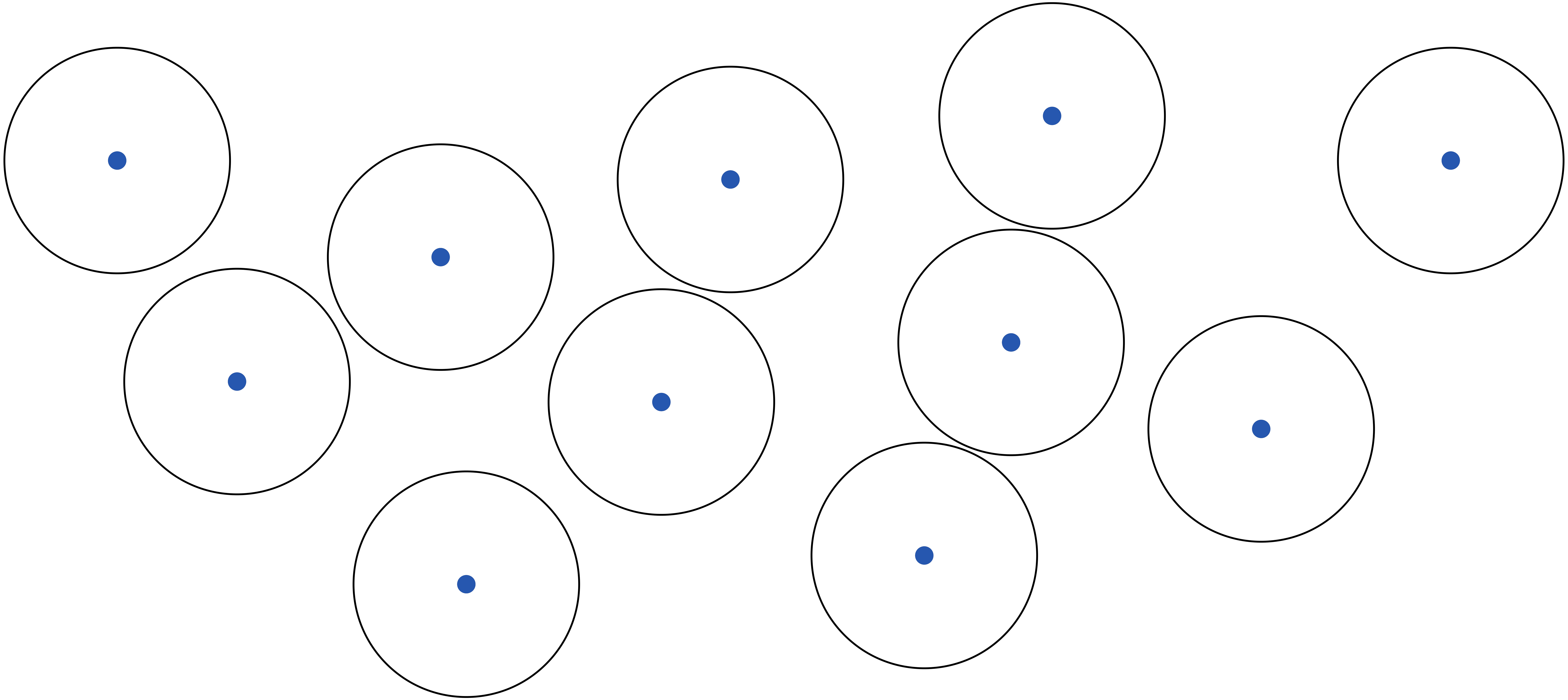
Picture



Picture



Tradeoffs



Parameters

- $k = \log_{|\Sigma|}(|C|)$ = message length (dimension)
- n = codeword length
- $R = k/n$, rate
- d = minimum (Hamming) distance between two codewords
- $\delta = d/n$, relative distance

What we want from codes

- High rate (low overhead)
- High distance (robust to many errors)
- Efficient decoding/encoding algorithms
- List decodable, locally testable, locally decodable…

Polynomial Evaluation Codes

- The messages are all m-variate polynomials of degree at most d.
- A polynomial f is encoded by evaluating f on each point in some evaluation set $S \subset \mathbb{F}^m$

$$f \to (f(\mathbf{x}))_{\mathbf{x} \in S}$$

The Evaluation Set

- Since the difference of two polynomials of degree at most d is again a polynomial of degree at most d, the minimum distance between two codewords is the minimum number of non-zeros of any non-zero degree $\leq d$ polynomial on S.
- **Want:**
    - **High distance:** All non-zero polynomials of degree $\leq d$ have many non-zeros in S
    - **High rate:** S is as small as possible.

The most famous code

Reed Solomon Codes, $m = 1$

$S \subseteq \mathbb{F}$

- Have the optimal rate-distance tradeoff $R = 1 - \delta$.
- Decodable [WB86], List Decodable [GS99]...

Another example

Reed Muller Codes

$S = A^m$, where $A \subseteq \mathbb{F}$.

- Suboptimal rate-distance tradeoff: $R \approx (1 - \delta)^m / m!$
  - In particular, $R \leq 1/m!$
- Decodable [KK17], List decodable [PW04]
- Locally testable [RS96, AS03]

Goal

Construct high-rate multivariate polynomial evaluation codes.

Related Work

Polynomial Identity Testing

Problem: Given query access to a polynomial $f \in \mathbb{F}[X_1, \ldots, X_m]$, of degree d, determine if $f \equiv 0$.

Classic test: Sample a random point $\mathbf{x}$ from S. Accept iff $f(\mathbf{x}) = 0$.

If $f \equiv 0$, then the test is always correct

If $f \not\equiv 0$ the test is correct iff $f(\mathbf{x}) \neq 0$.

- Randomness efficiency corresponds to $|S|$
- Low error corresponds to a non-zero f having many non-zeros in S

Polynomial Identity Testing

Individual degree bounds

- Chen-Kao [CK97], Lewin-Vadhan [LV98], Agrawal-Biswas [AB03]

Sparse polynomials

- Klivans-Spielman [KS01]

$d << m$

- Bläser-Pandey [BP20]

Pseudorandom Generators Against Polynomials

Want a generator G such that for any polynomial f for degree at most d,

$$f(U) \sim f(G(s))$$

Intuition: if those distributions looks similar, any non-zero f should be non-zero on many points of the form $G(s)$, since f is non-zero on most of $\mathbb{F}^m$.

In fact, there is a reduction from polynomial evaluation codes to pseudorandom generators.

Pseudorandom Generators Against Polynomials

Constructions from Dvir-Shpilka [DS11], Viola [Vio08], Bogdanov-Viola [BV10] work in the setting of large *m*, constant *d*, and small field size.

Main Results

Theorem A. *For any constant* $R \in (0,1)$, $m \geq 1$*, there exist* m*-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*

Theorem B. *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*

Theorem C. m*-variate GAP codes are locally testable with* $O(n^{2/m})$ *queries.*

High rate polynomial evaluation codes

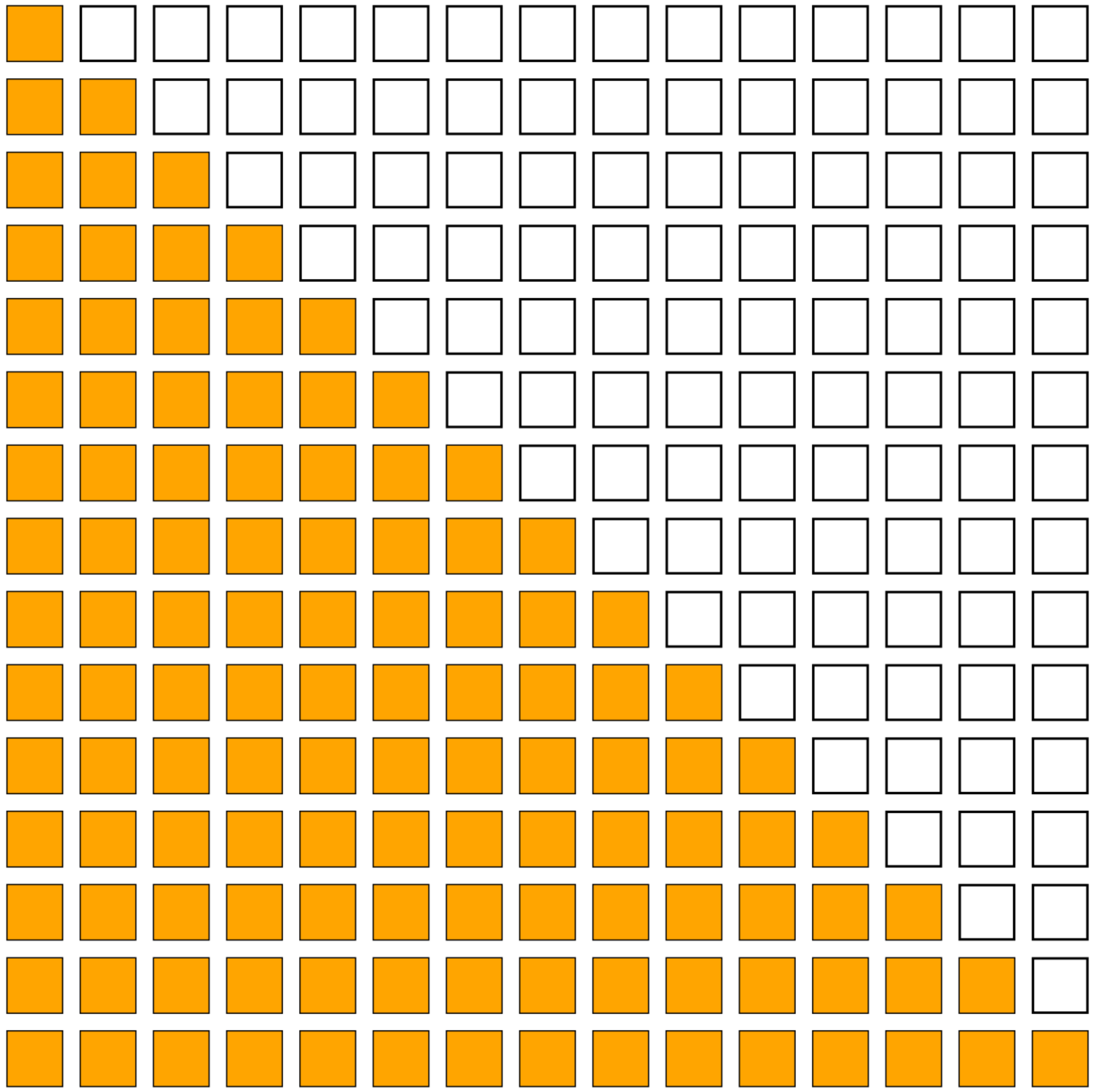
Two constructions

- CAP (Combinatorial Arrays for Polynomials)
- GAP (Geometric Arrays for Polynomials)

- This talk: Constructions of bivariate CAP and GAP codes.

CAP Codes

CAP Codes



Distance of CAP Codes

The distance of CAP codes is obtained by a generalization of the Schwartz-Zippel Lemma.

How many zeros are there in a $\ell \times \ell$ grid?

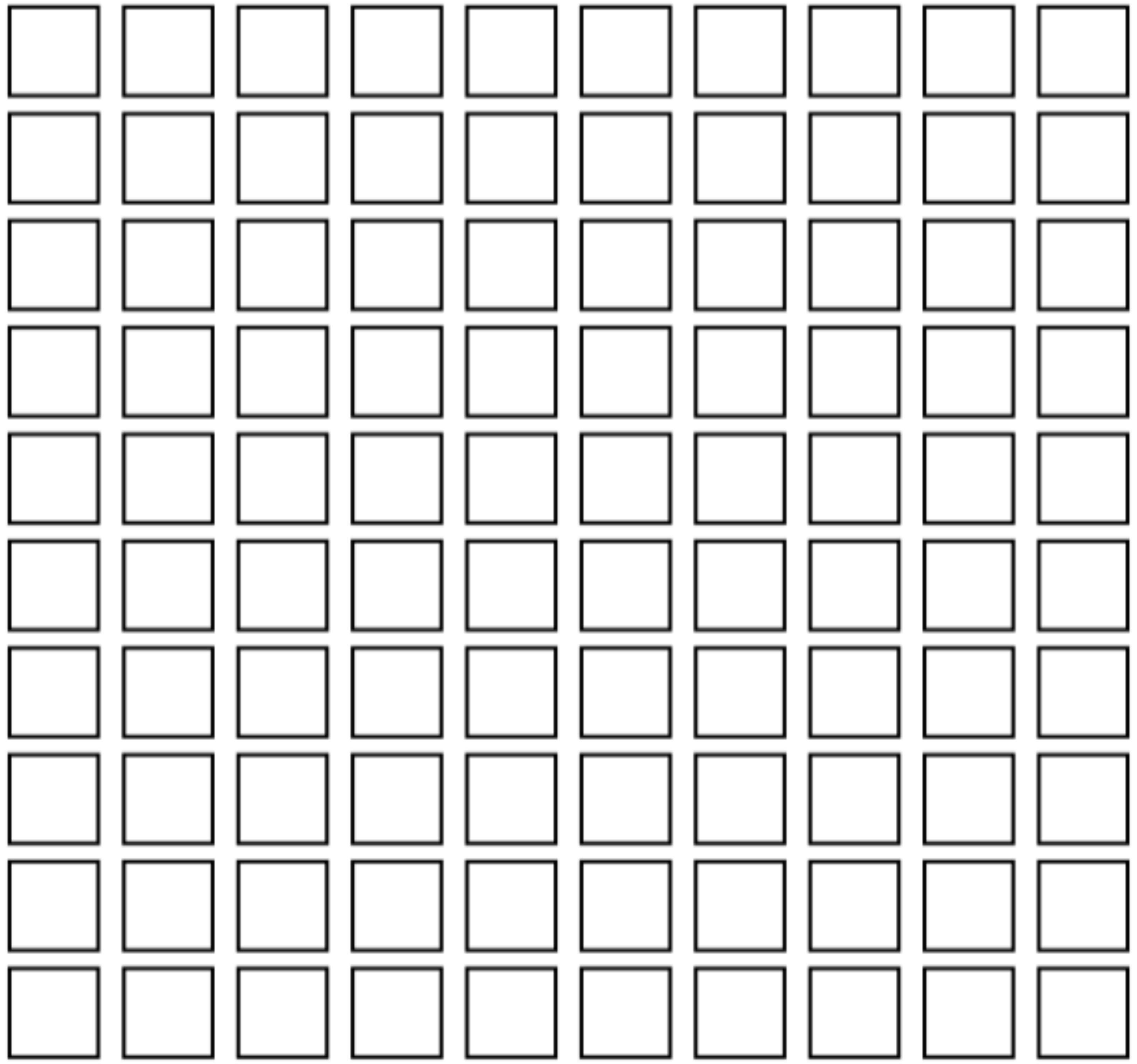
Recap: Schwartz-Zippel

Let $f(X, Y) = \sum_{i=0}^{d_Y} c_i(X) Y^i$,

How many zeros are there in the ath column?

1. If $c_{d_Y}(a) \neq 0$, then $f(a, Y)$ is a univariate polynomial in Y of degree d_Y.
2. If $c_{d_Y}(a) = 0$, all bets are off, since $f(a, Y)$ might be identically zero.

Case 2 happens at most $d - d_Y$ times, so the total number of zeros is at most $d_Y(\ell - d + d_Y) + \ell(d - d_Y) \leq d\ell$

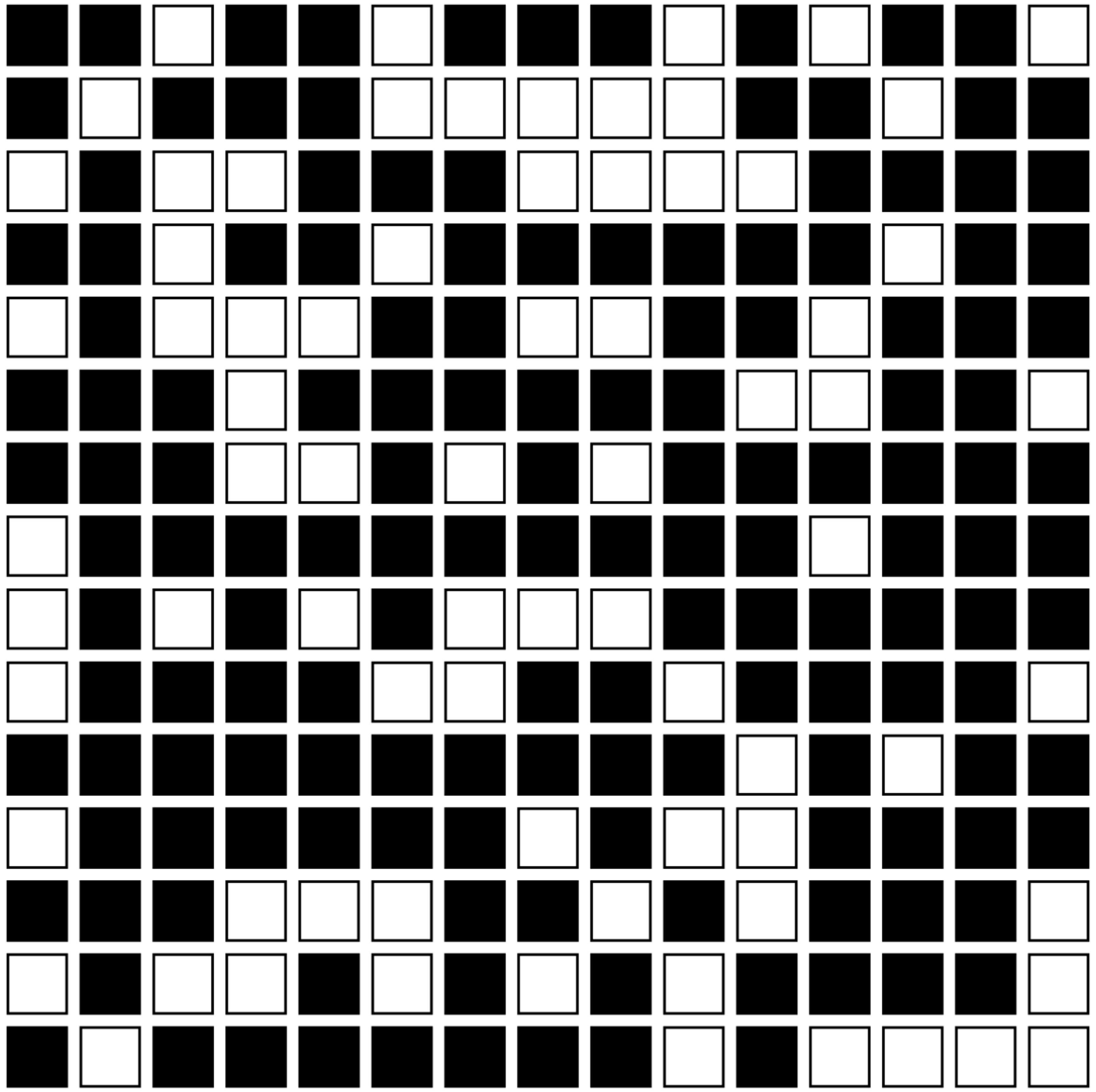


Number of zeros

E.g. $\ell = 15, d = 10$

There are ≤ 150 zeros in the grid...

Zeros are **filled** squares



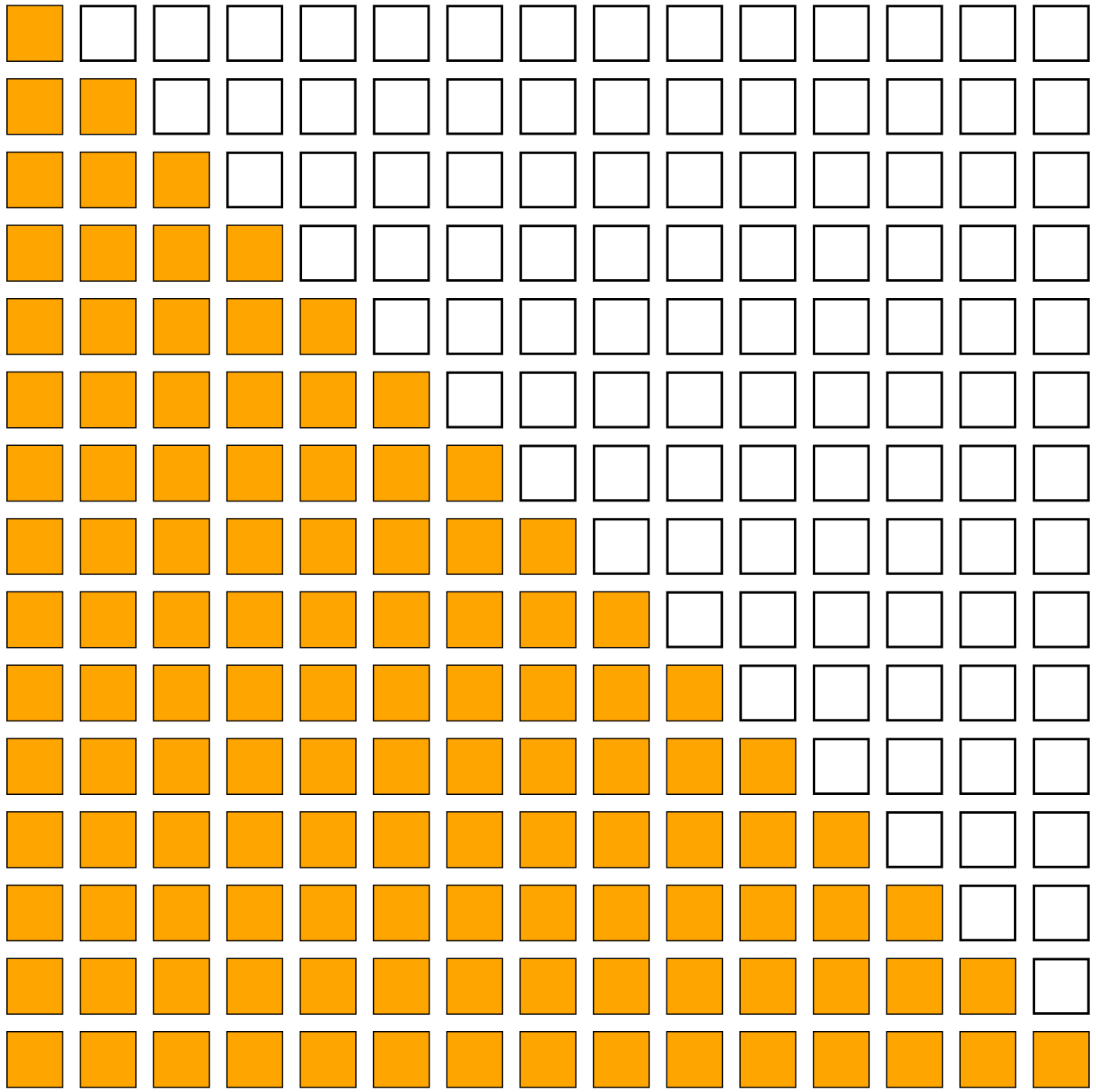
CAP Codes

Zeros in the triangle

E.g. $\ell = 15, d = 10$

… but that's useless because there are only

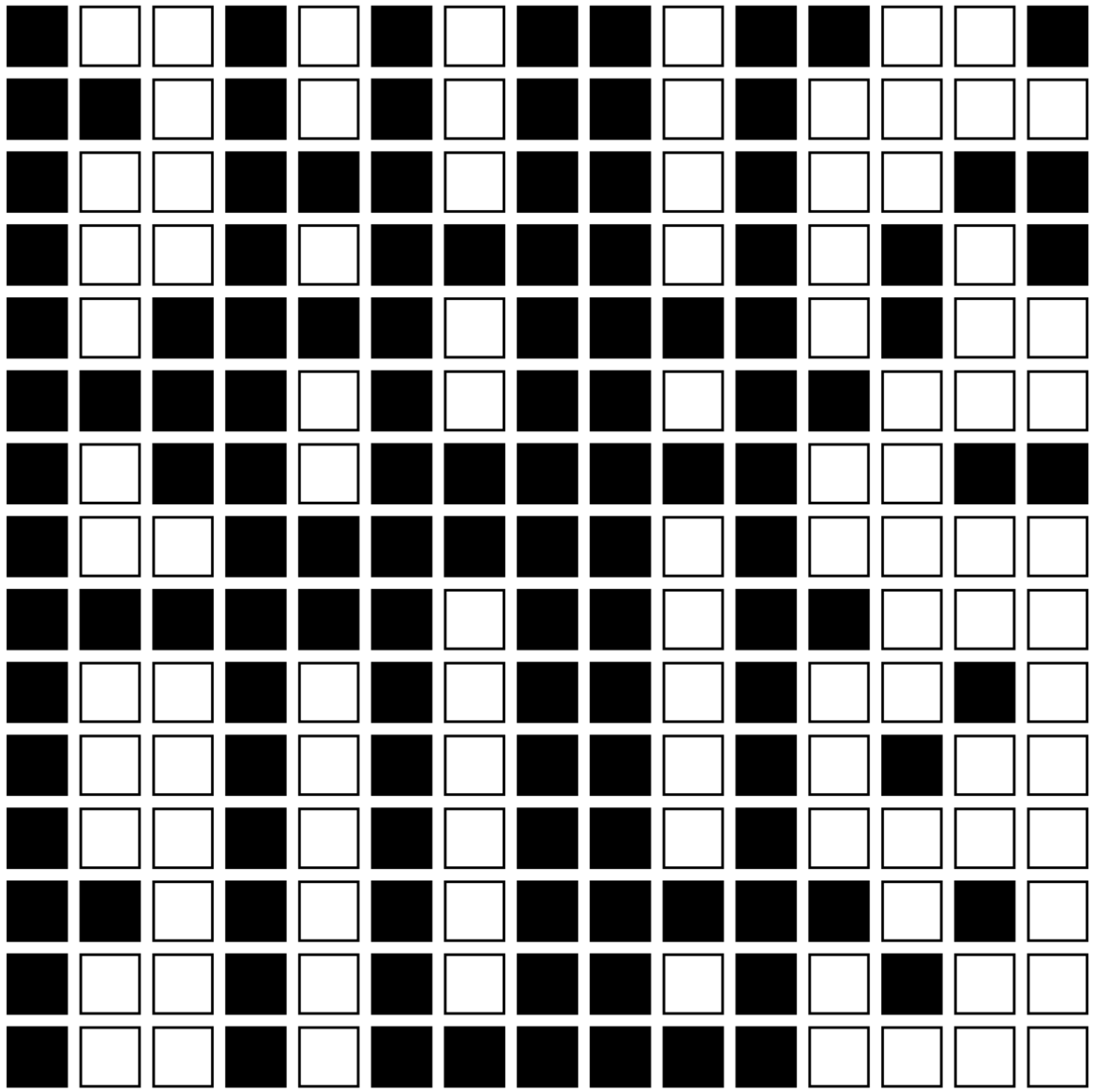
$\binom{15+1}{2} = 120$ points in the triangle!



There exists $d_Y \in \{0,1,\ldots,d\}$ (which is the Y-degree of f) such that at most $d - d_Y$ columns are entirely zero, and the remaining columns have at most d_Y zeros each.

Shape of zeros

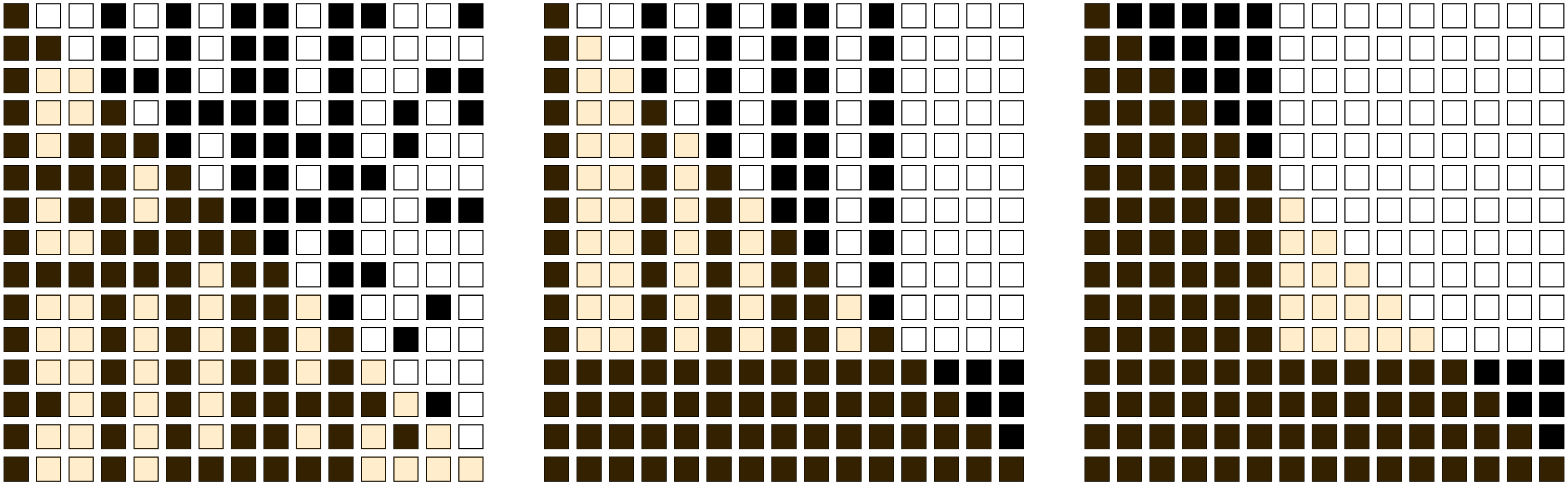
E.g. $\ell = 15, d = 10, d_Y = 4$



Zeros are **filled** squares

Counting zeros in the triangle

Shifting zeros down and to the left can only increase the number of zeros in the triangle



At least $\binom{\ell - d + 1}{2}$ non-zeros in the triangle!

Rate and distance calculation

$$\delta = \frac{\binom{\ell + d + 1}{2}}{\binom{\ell + 1}{2}} \geq \left(1 - \frac{d}{\ell}\right)^2$$

$$R = \frac{\binom{d + 1}{2}}{\binom{\ell + 1}{2}} \geq \left(\frac{d}{\ell}\right)^2$$

$$R \geq \left(1 - \sqrt{\delta}\right)^2$$

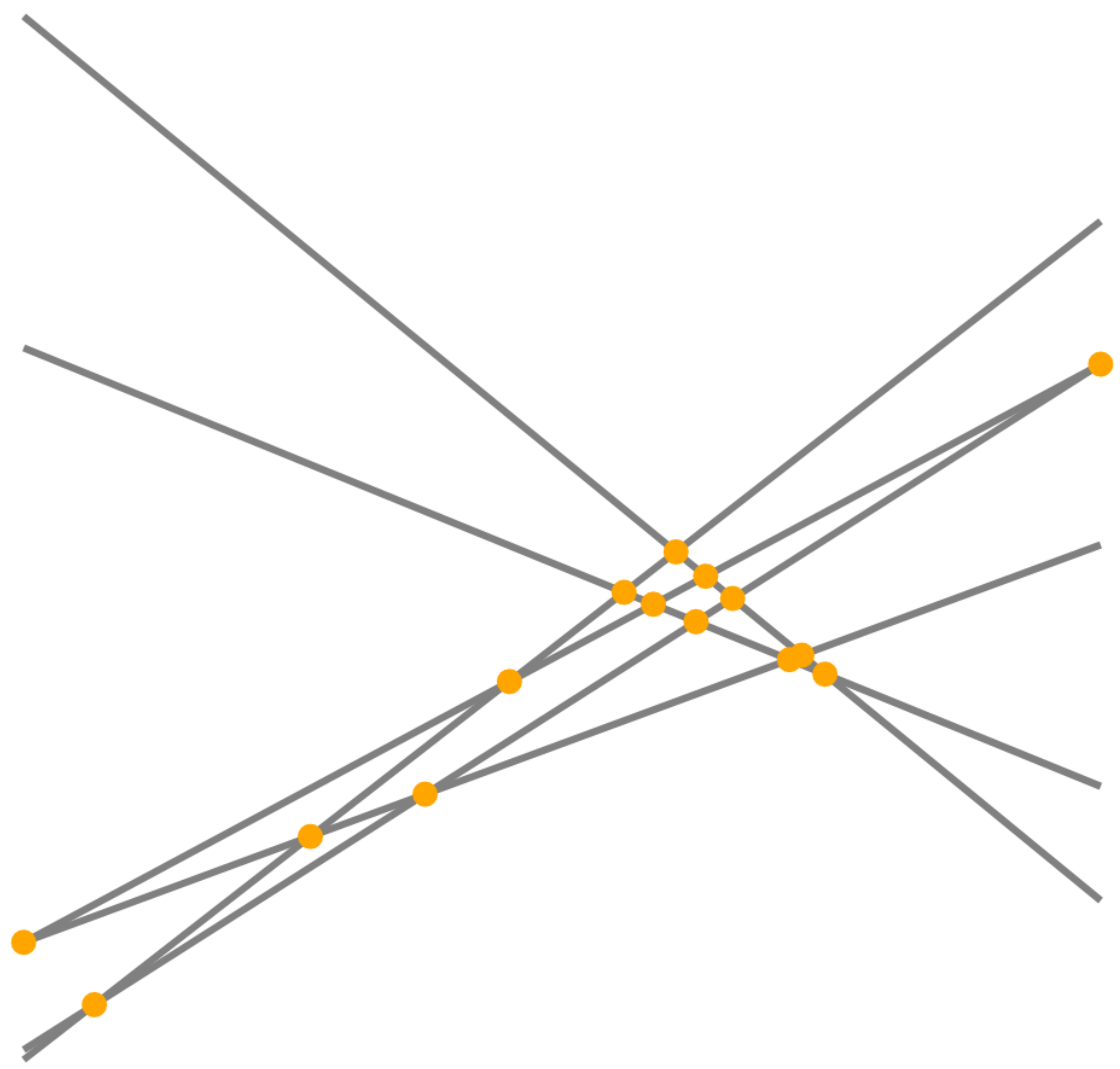
$$\sqrt{R} + \sqrt{\delta} \geq 1$$

GAP Codes

GAP Codes

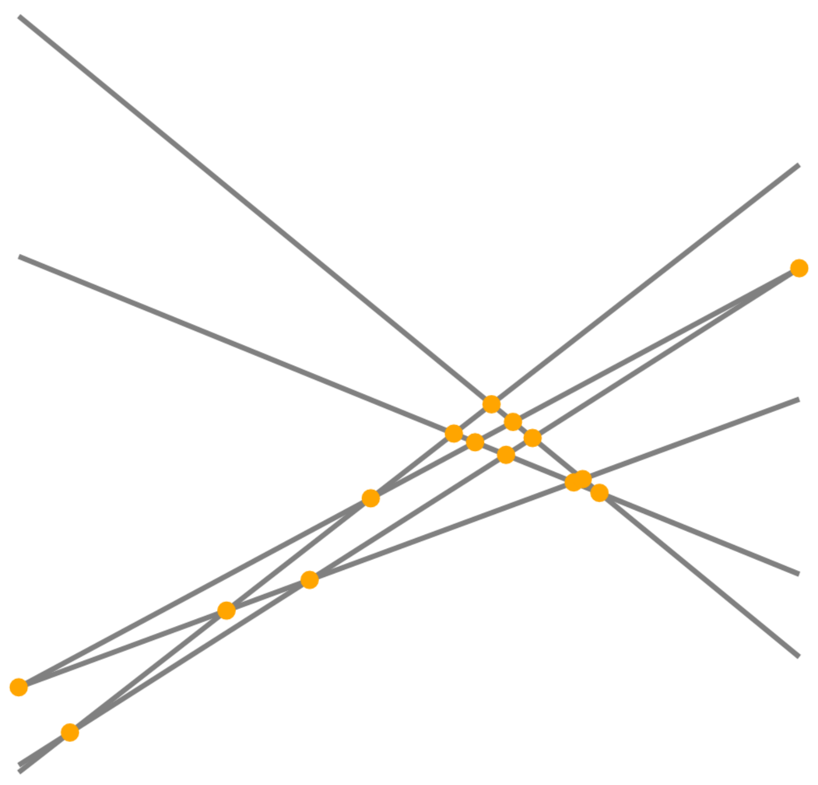
A geometric construction

Take the intersection points of t lines in general position.



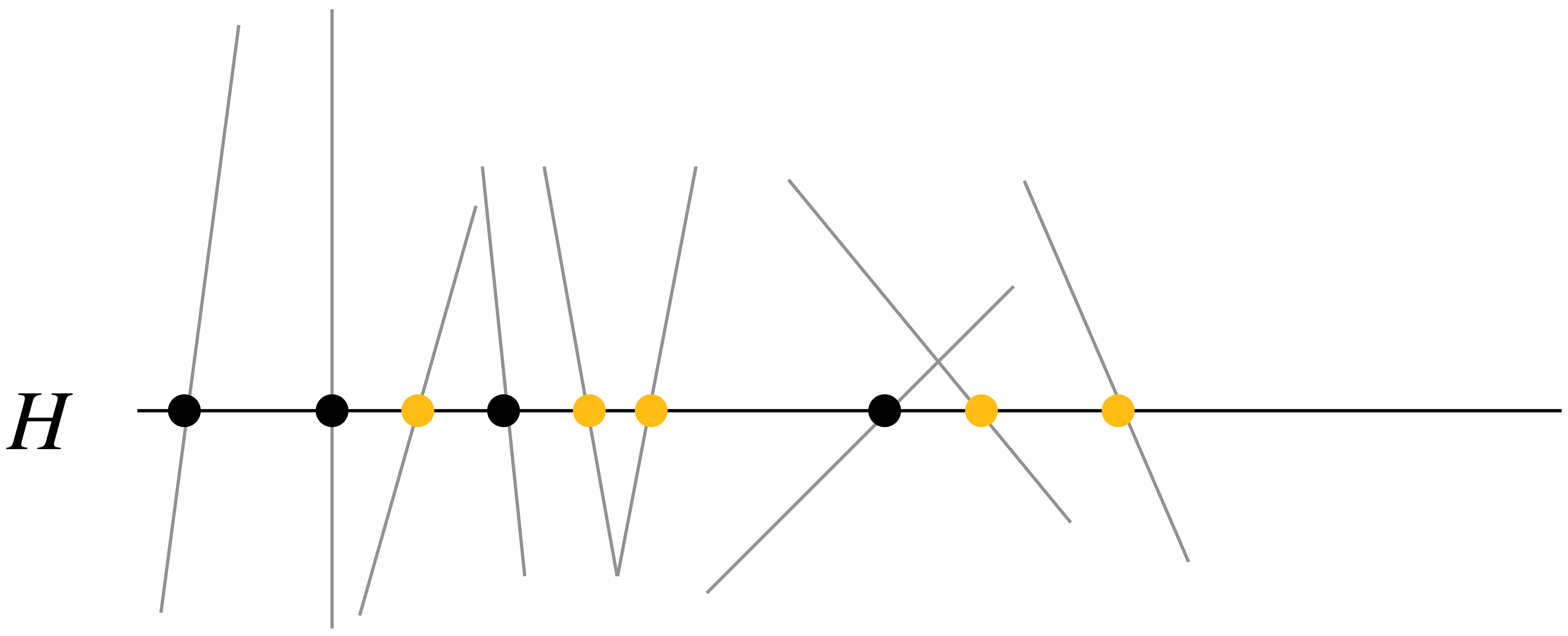
Distance of GAP Codes

Zoom in on a single line



Zoom in on a particular line containing a non-zero of f

Call the line H, and suppose it's defined by the equation $Y = mX + b$

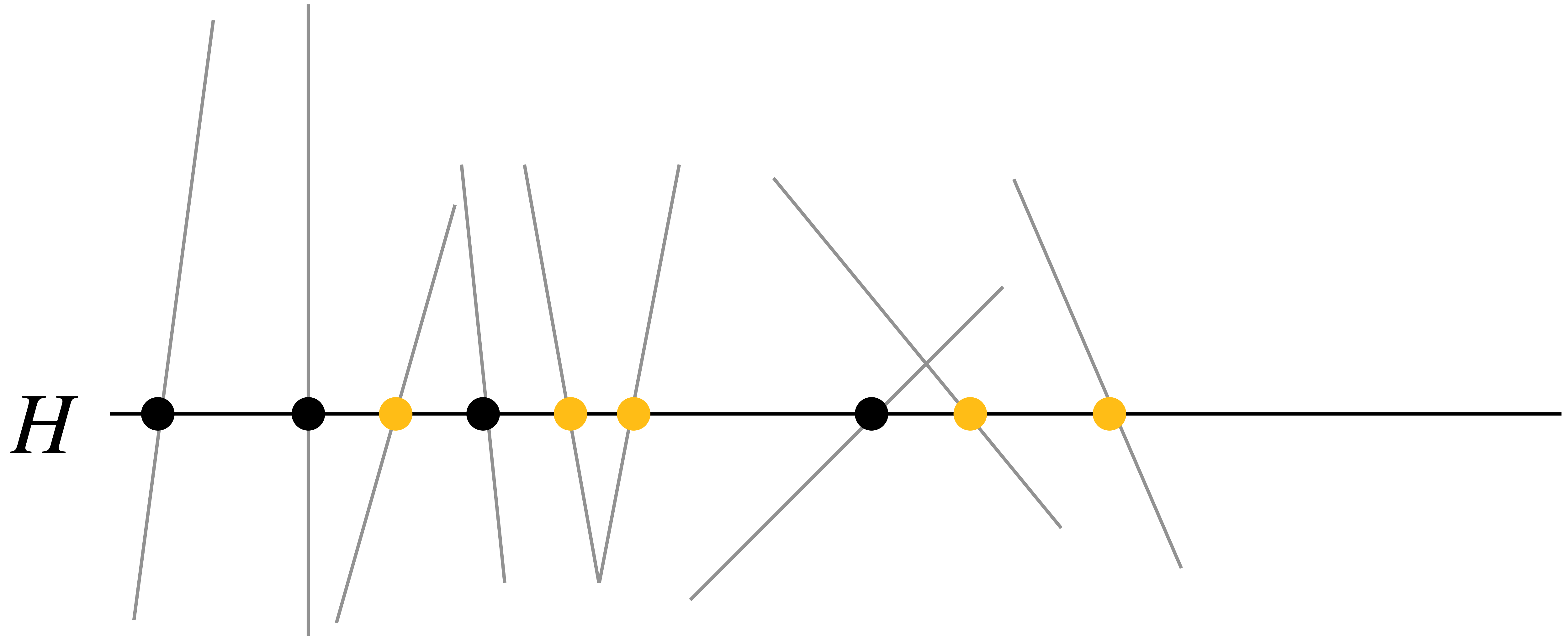


Distance of GAP Codes

Count the number of non-zeros on H

Then the polynomial $g(X) = f(X, mX + b)$ is a non-zero univariate polynomial of degree at most d.

Hence, there are at least $t - 1 - d$ non-zeros on this line



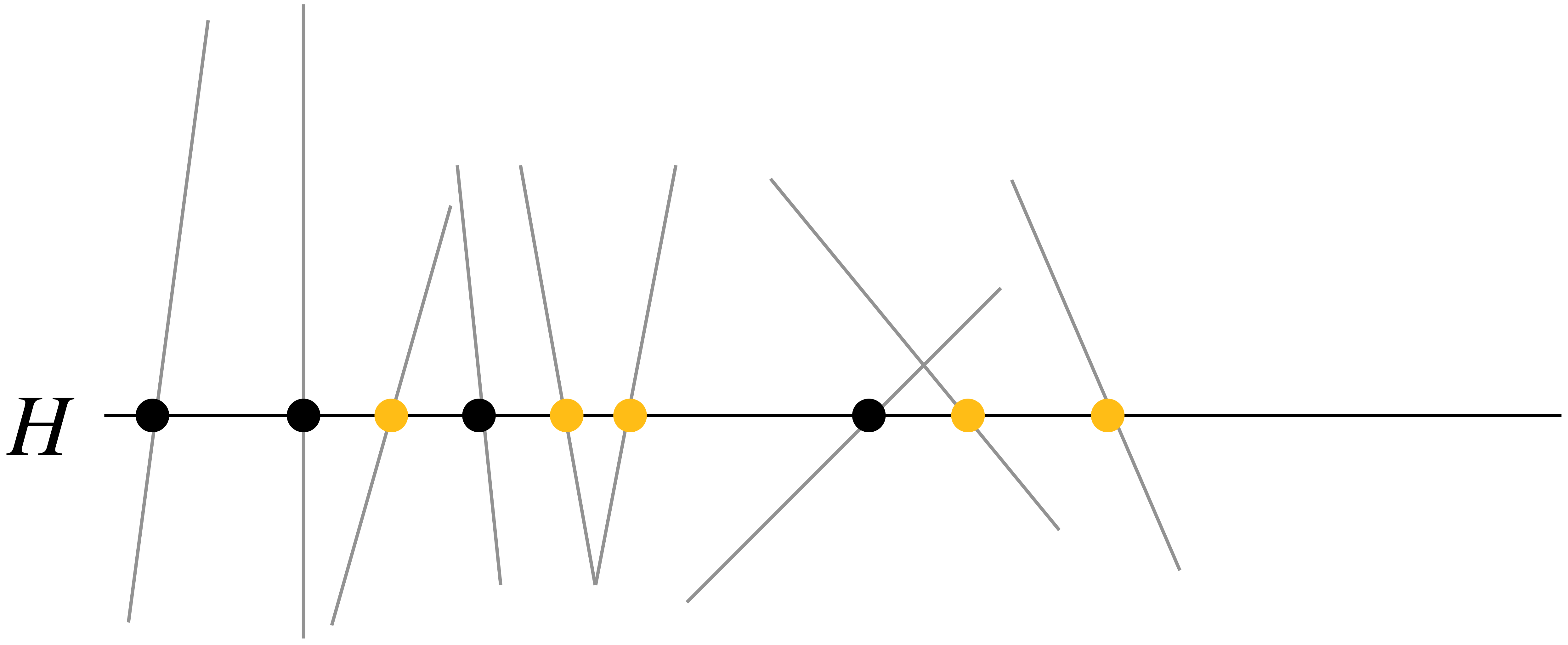
Distance of GAP Codes

Repeat this logic for the other line going through a non-zero point

Each line going through a non-zero point on H contains at least $t - d - 1$ non-zeros. So we found $(t - d - 1)^2$ non-zeros.

However, each non-zero point not on H was counted twice. Thus, the actual number of non-zeros is at least

$$\frac{(t-d-1)^2 + t - d - 1}{2} = \binom{t-d}{2}$$



R vs. δ calculation

There are $\binom{t}{2}$ points. Thus, we have

$$\delta = \binom{t-d}{2} / \binom{t}{2} \approx (1 - d/t)^2,$$

and

$$R = \binom{d+2}{d} / \binom{t}{2} \approx (d/t)^2.$$

The tradeoff is $\sqrt{\delta} + \sqrt{R} = 1$

Evaluation Sets for Higher m

CAP Codes

- Triangle → m-dimensional simplex

GAP Codes

- Intersections of lines in general position → Intersections of hyperplanes in general position.

Tradeoff: $R^{1/m} + \delta^{1/m} = 1$

Main Results

Theorem A. *For any constant* $R \in (0,1)$, $m \geq 1$*, there exist* m*-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*

Theorem B. *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*

Theorem C. m*-variate GAP codes are locally testable with* $O(n^{2/m})$ *queries.*

Main Results

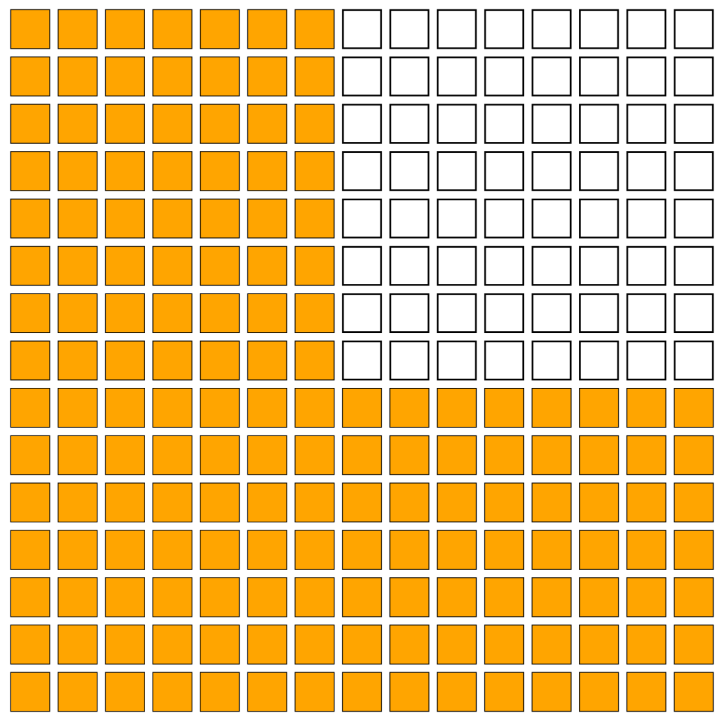
~~**Theorem A.** *For any constant $R \in (0,1)$, $m \geq 1$, there exist m-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*~~

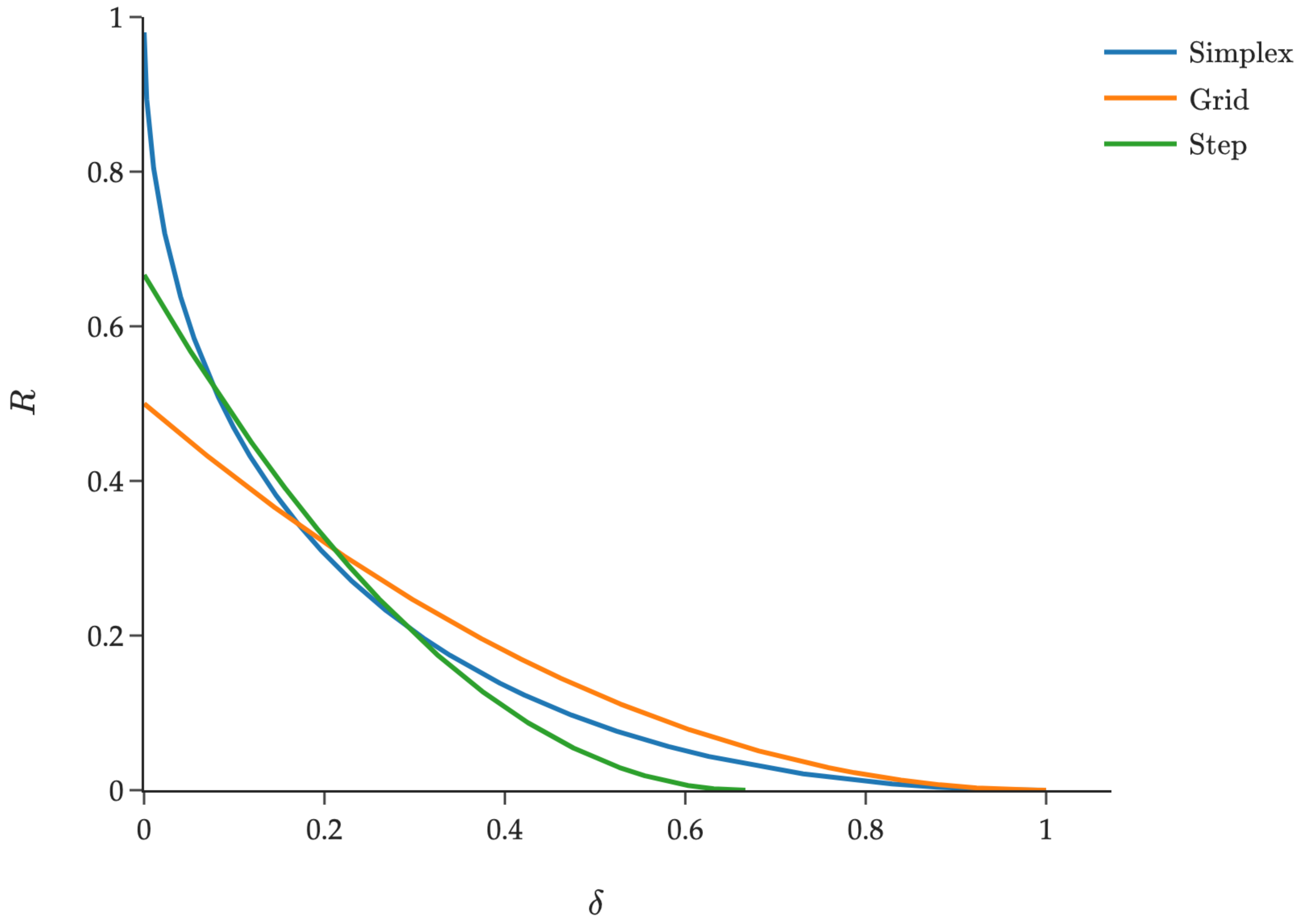
Theorem B. *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*

Theorem C. *m-variate GAP codes are locally testable with $O(n^{2/m})$ queries.*

Future directions

- Better tradeoffs: $R = 1 - \delta$?
- Other properties
- Growing m
- Better CAP codes





Thank you!

Longer talk video

Link to Paper

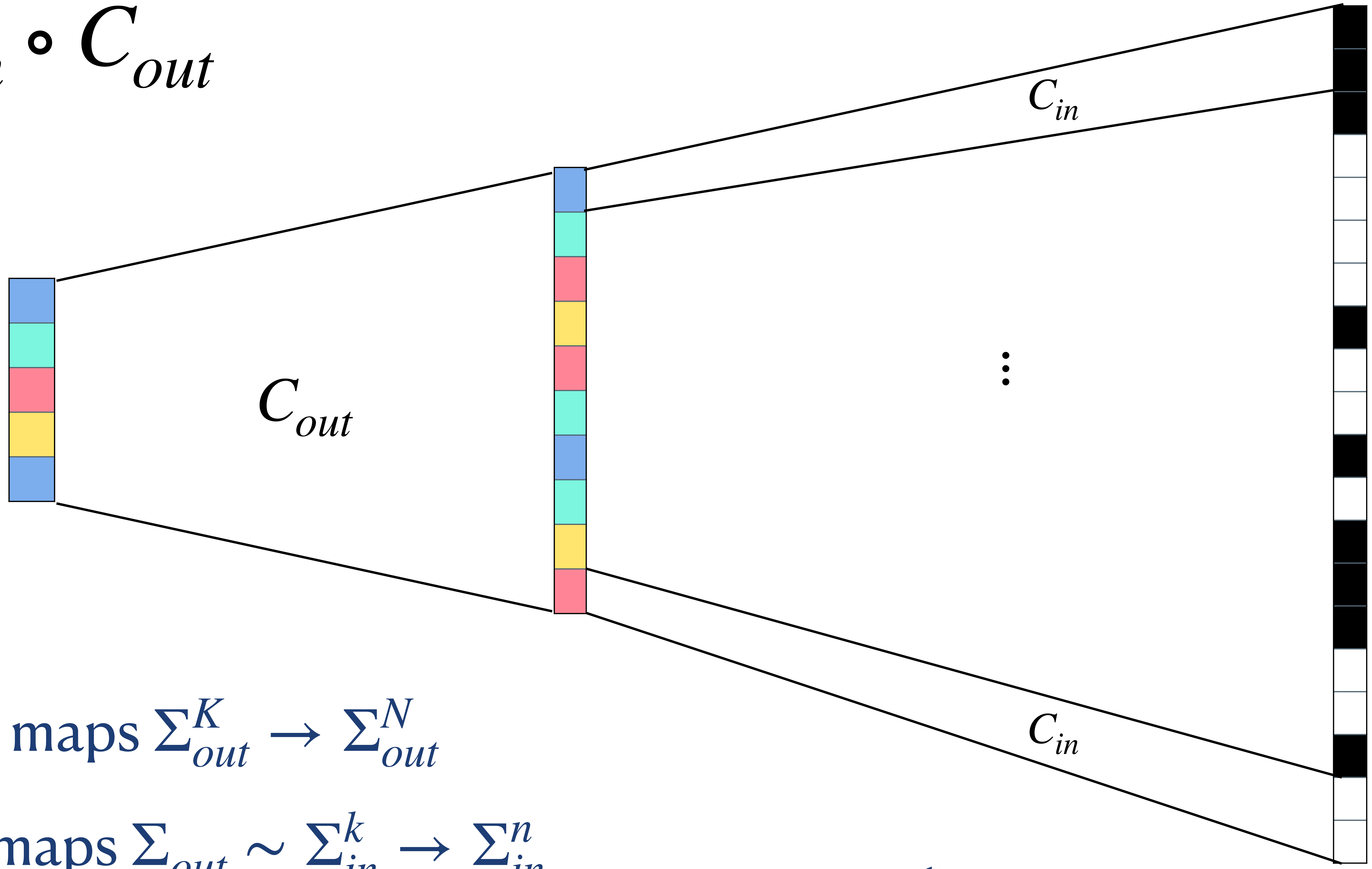
Unique Decoding

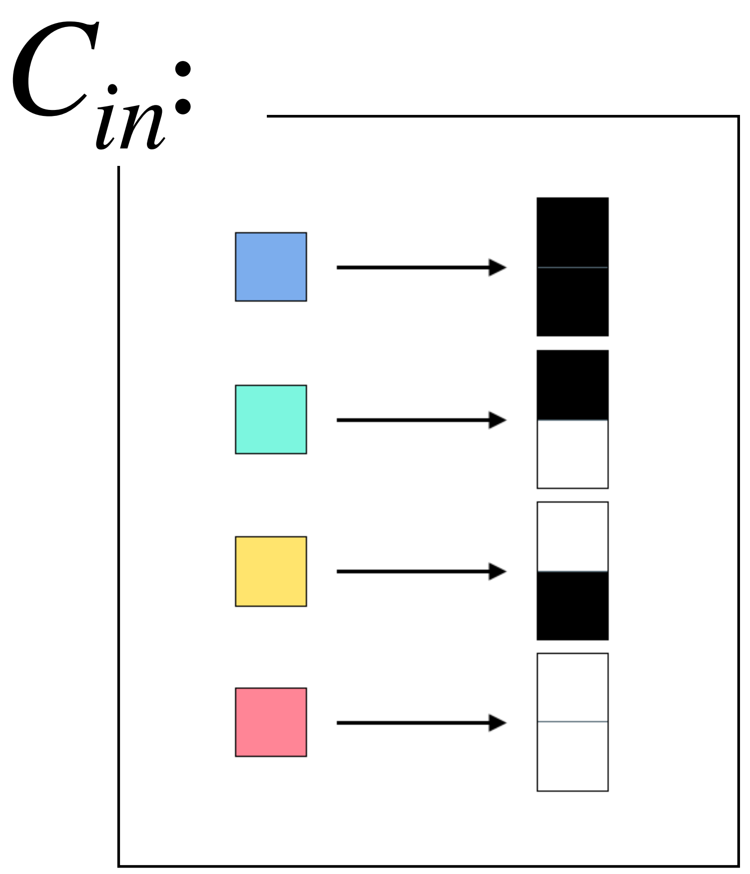
Concatenated Codes

- A key component to our decoding algorithms is code concatenation, and the GMD algorithm, which is a general way to decode concatenated codes.
- Decoding GAP codes can be done almost directly using GMD.
- Decoding CAP codes requires a new variant of the GMD algorithm.

Concatenated Codes

$C_{in} \circ C_{out}$





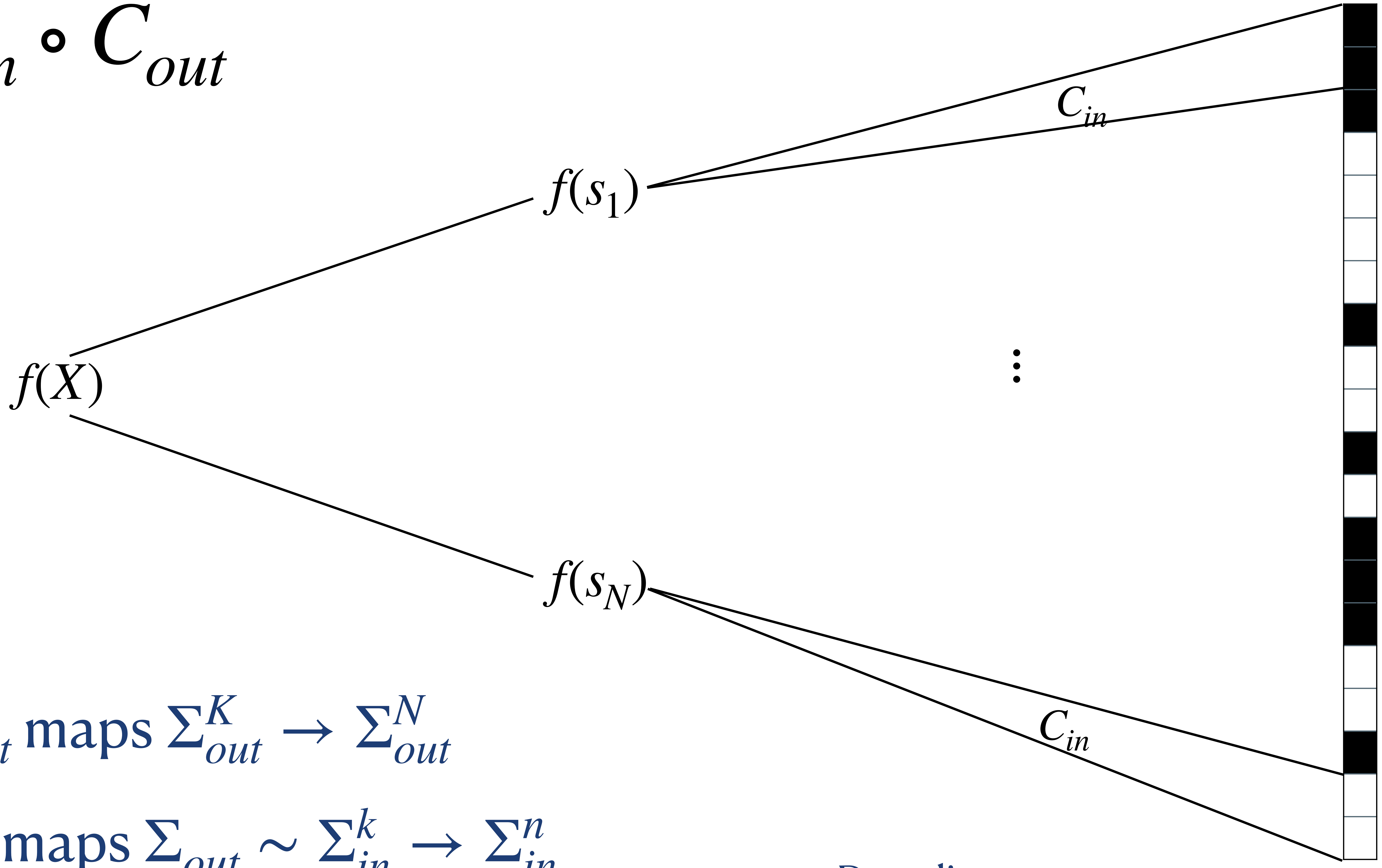
C_{out} maps $\Sigma_{out}^{K} \rightarrow \Sigma_{out}^{N}$

C_{in} maps $\Sigma_{out} \sim \Sigma_{in}^{k} \rightarrow \Sigma_{in}^{n}$

Decoding

Example: RS as the outer code

$C_{in} \circ C_{out}$

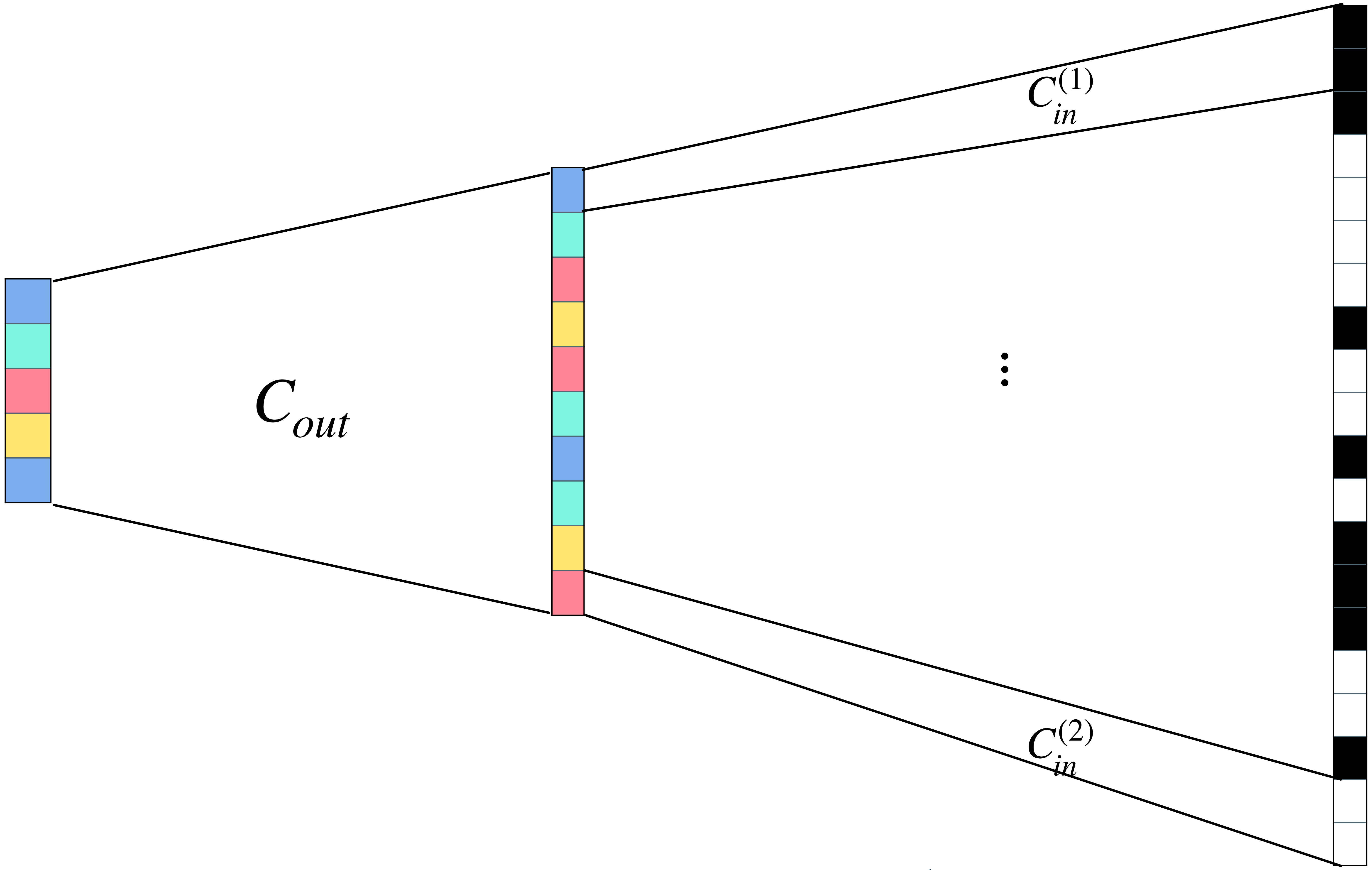


C_{out} maps $\Sigma_{out}^K \to \Sigma_{out}^N$

C_{in} maps $\Sigma_{out} \sim \Sigma_{in}^k \to \Sigma_{in}^n$

Decoding

Concatenated Codes



Decoding

More generally, each inner code can be different!

Concatenated Codes

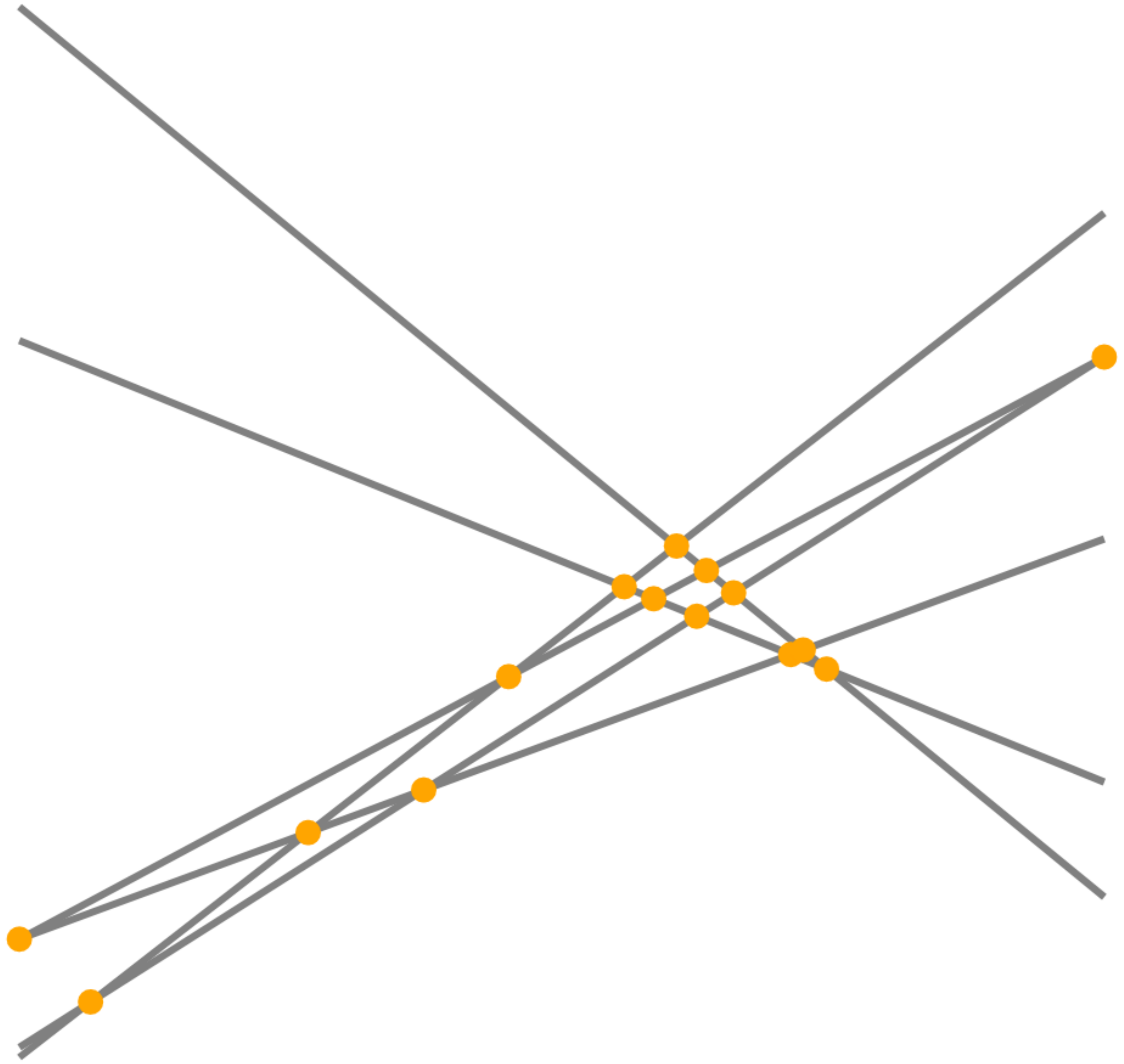
- If C_{in} is a [n, k, d] code and C_{out} is a [N, K, D], code, then $C_{out} \circ C_{in}$ is a [Nn, Kk, Dd] code.

Theorem (GMD Decoding) [For66]. Suppose C_{out} , C_{in} can be decoded optimally*,
Then, $C_{in} \circ C_{out}$ can be decoded optimally.

*optimally as in most number of errors we can hope to decode from, which is < distance /2

Decoding GAP Codes

- Recall GAP codes are evaluated on the m-wise intersections of hyperplanes $H_1, ..., H_t$. Let's think of $m = 2$ for now.



GAP codes as concatenated codes

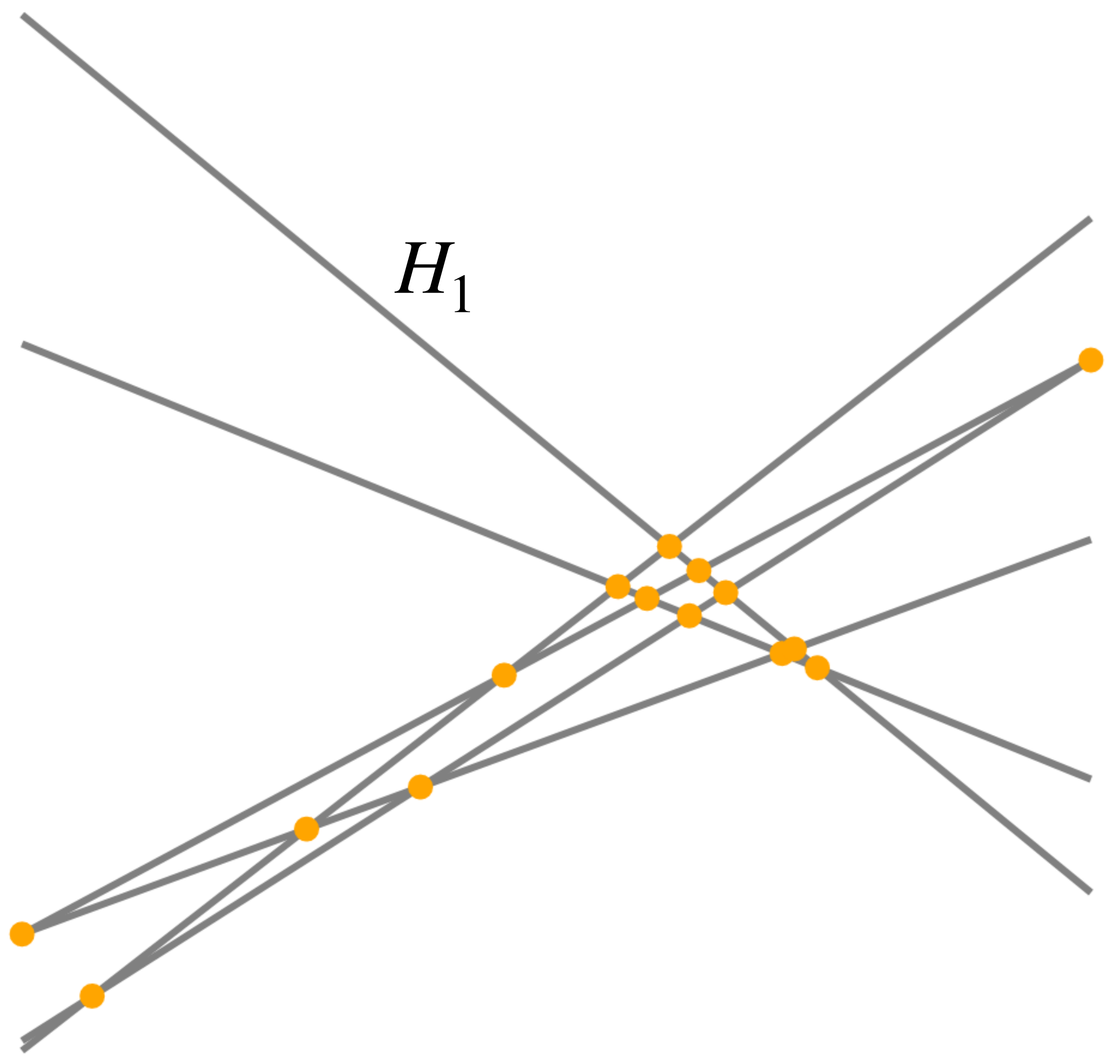
$$f(X,Y) = \sum_{i=0}^{k} f_i(X) Y^i$$

Shorthand: $g(A) = [g(a_1), \ldots, g(a_{t-1})]$

$$g_1(X) = f(X, m_1 X + b_1) \quad — \quad g_1(H_1 \cap S)$$

$$f(X, Y) \qquad \vdots \qquad \vdots \qquad \vdots$$

$$g_t(X) = f(X, m_t X + b_t) \quad — \quad g_t(H_t \cap S)$$



$$f(X, Y) = \sum_{i=0}^{k} f_i(X) Y^i$$

- The outer code is a RS code where elements are from $\mathbb{F}(X)[Y]$
- The inner code is an RS code.

Decoding CAP Codes

Decoding of CAP codes is based on [KK17].

The main new ingredient is an "uneven" version of the classic GMD algorithm for decoding concatenated codes. The proof is based on ideas from [BHKS23].

Lemma (Uneven GMD). *Let C_{out} be a code with block length N, and distance D. Let $C_1, \ldots, C_N$ be codes with distance d_i. Let $C = (C_1, \ldots, C_N) \circ C_{out}$. Then C has minimum distance at least $\min_{S \subset [N] : |S| = D} \sum_{i \in S} d_i$. Furthermore, if there exist optimal unique decoding algorithms for $C_{out}, C_1, \ldots, C_N$, then there exists an optimal unique decoding algorithm for C.*

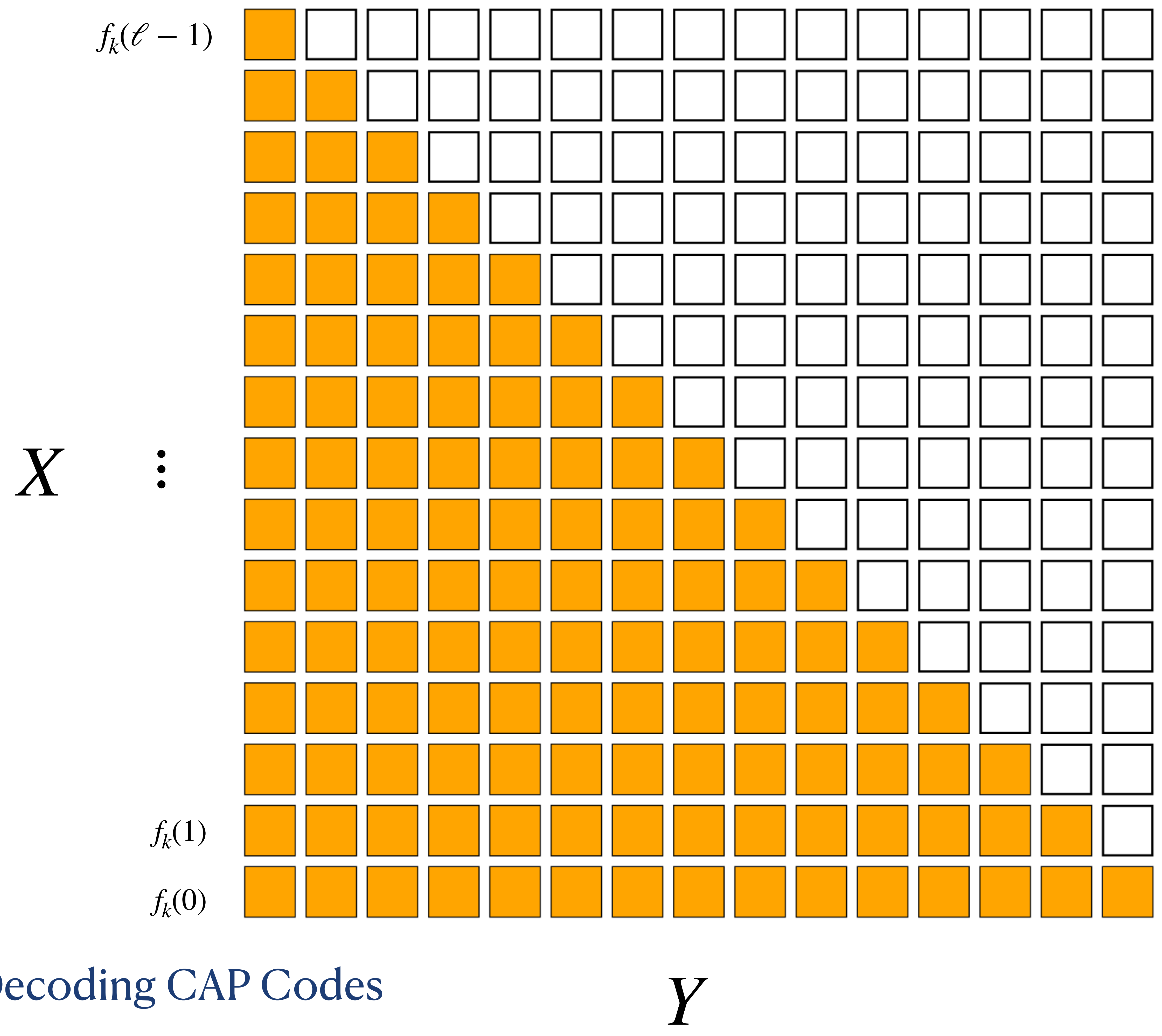
Decoding CAP Codes

Viewing CAP codes as a concatenated code

$$f(X, Y) = \sum_{i=0}^{k} f_i(X) Y^i$$

The key is to view the codeword as an encoding of f_k under concatenated code $\{C_1, \ldots, C_\ell\} \circ C_{out}$, where

- C_{out} evaluates f_k on $0,1,\ldots,\ell-1$
- C_x maps $\alpha \to \alpha Y^k + \sum_{i=0}^{k-1} f_i(x) Y^i$ and evaluates that polynomial on $0,1,\ldots,\ell - x - 1$.



Decoding CAP Codes

Distance Calculation

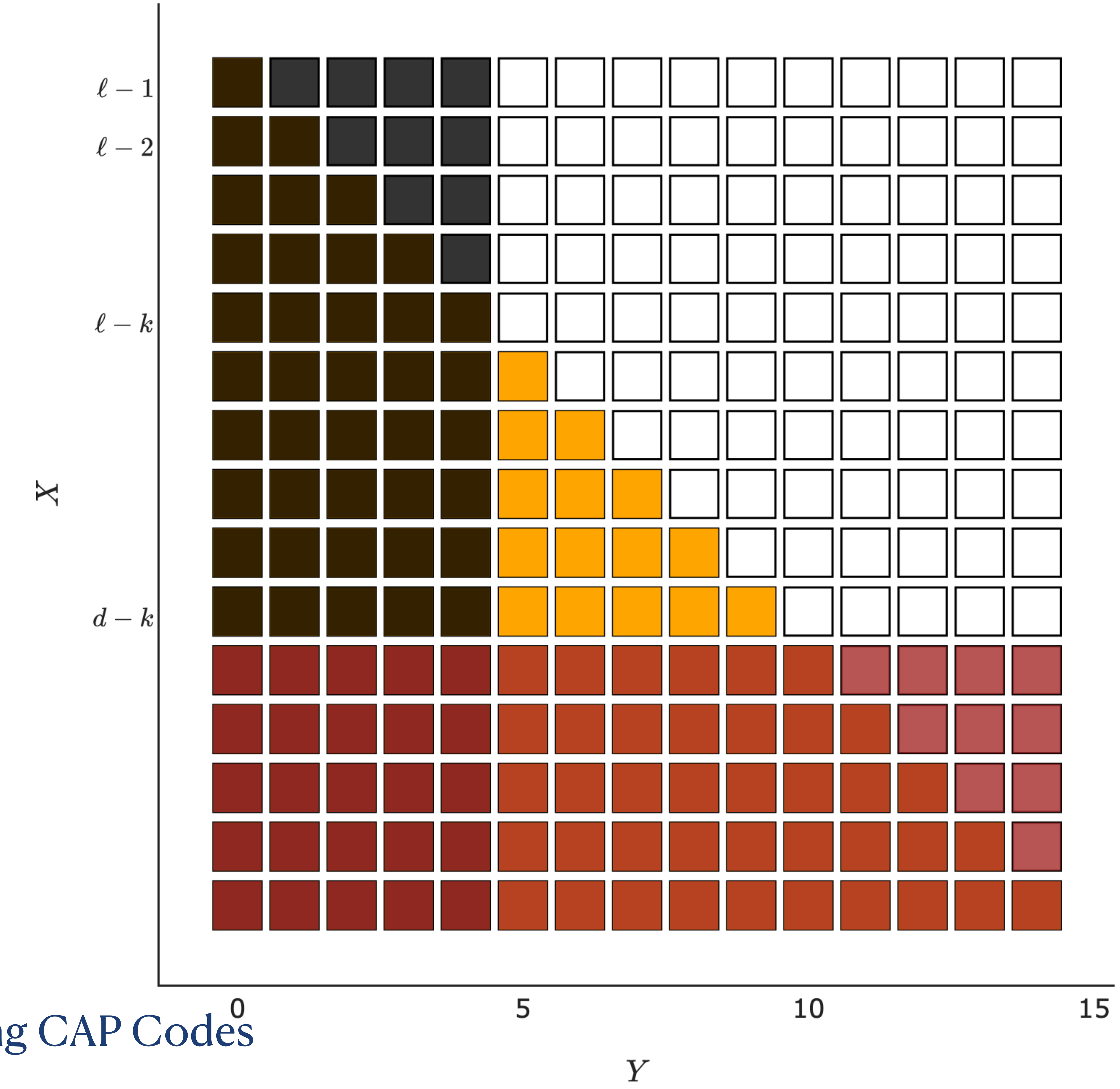
C_{out} evaluates f_k on $0,1,\ldots,\ell-1$

C_x maps $\alpha \to \alpha Y^k + \sum_{i=0}^{k-1} f_i(x) Y^i$

- The outer distance, D, is $\ell - (d-k)$
- The xth inner distance, d_i is $\ell - x - k$

- Top k inner codes have distance 0, next $\ell - d$ codes have distance $1, 2, \ldots, \ell - d$, so the distance of the concatenated code is

$$\binom{\ell - d + 1}{2}$$



Decoding CAP Codes

Recurse

- Thus, we can recover f_k using GMD
- Then, subtract f_k from the received word, and recurse to find $f_{k-1}, \ldots, f_0$

Main Results

~~**Theorem A.** *For any constant $R \in (0,1)$, $m \geq 1$, there exist m-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*~~

Theorem B. *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*

Theorem C. *m-variate GAP codes are locally testable with $O(n^{2/m})$ queries.*

Main Results

~~**Theorem A.** *For any constant* $R \in (0,1)$, $m \geq 1$, *there exist* m*-variate polynomial evaluation codes (CAP and GAP codes) with rate* R *and constant relative distance.*~~

~~**Theorem B.** *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*~~

Theorem C. m*-variate GAP codes are locally testable with* $O(n^{2/m})$ *queries.*

Local Testability of GAP Codes

Local Testing

- Motivation: Although decoding algorithms are polynomial time, it can still be an expensive process, especially if the message is long.
- A local test is an algorithm you can run on a received word to quickly check if it is "close" to a valid codeword or far from all valid codewords.

Local Testing for GAP Codes

Theorem. There exists a test such that

- **Completeness.** if f is a codeword of the GAP code, then the test passes with probability 1.
- **Soundness**. There exists constants T, Q such that if the test rejects f with probability $p \leq T$, then $\delta_C(f) \leq Q \cdot p$.

Common Tests

For, e.g., Reed Muller Codes

Line/Plane-point test(f):

1. Pick a random line/plane, P
2. Let g_P be the closest degree d bivariate polynomial to $f|_P$
3. Sample a random point $\mathbf{x}$ on the line/plane and accept iff $g_P(\mathbf{x}) = f(\mathbf{x})$

Local Testing Intuition

- Reed-Solomon codes are not locally testable because all small sets of evaluations are consistent with some codeword.
- On the other hand, Reed-Muller codes resemble univariate polynomials in every line - a significant restriction.
- For Reed Muller codes, to ensure these tests work, we typically need to use the entire dataset as the evaluation set; otherwise, a random line or plane may not be contained in the evaluation set.
- Thus, the fact that GAP codes are high rate and locally testable is surprising and interesting to us.

Local Testing for GAP Codes

Recall GAP codes are evaluated on the m-wise intersections of hyperplanes $H_1, \ldots, H_t$.

- m-wise intersections are points, $m-1$-wise intersections are lines, and $m-2$-wise intersections are planes.

Plane-point test(f):

1. Pick a random 2-D plane (intersection of a random subset of $m-2$ of the H_i), P
2. Let g_P be the closest degree d bivariate polynomial to $f|_P$
3. Sample a random point $\mathbf{x}$ on the plane and accept iff $g_P(\mathbf{x}) = f(\mathbf{x})$

Local Testing for GAP Codes

Plane-point test(f):

1. Pick a random 2-D plane (intersection of a random subset of $m-2$ of the H_i), P
2. Let g_P be the closest degree d bivariate polynomial to $f|_P$
3. Sample a random point $\mathbf{x}$ on the plane and accept iff $g_P(\mathbf{x}) = f(\mathbf{x})$

Completeness. If f is a codeword, then the plane-point test passes with probability 1

Soundness. There exist constants T, Q such that if the test rejects f with probability $p \leq T$, then $\delta_C(f) \leq Q \cdot p$.

Local Testing for GAP Codes

Robust local characterization

Let g_i be the closest $m - 1$ variate polynomial to $f|_{H_i}$.

Lemma (Robust local characterization). *"If many pairs of g_i are consistent, then some m-variate polynomial h is consistent with many of them."*

The proof is similar to [BSS06]

Soundness

Lemma (Robust local characterization).
"If many pairs of g_i are consistent, then some m-variate polynomial h is consistent with many of them."

Soundness. If the test accepts f with high probability, the f is close to the code. The proof of soundness is by induction. Suppose the test works for $m-1$ variate GAP codes.

1. If the tests accept f with high probability. Then, the probability the test accepts, given that the test queries a plane lying on H_i is also high.
2. Let g_i be the polynomial that is close to $f_i = f|_{H_i}$ (using the IH)
3. Since each g_i is close to f_i, many of them are consistent with each other.
4. Obtain a polynomial h consistent with most of them using the lemma.
5. h is a codeword that is close to f.

Main Results

~~**Theorem A.** *For any constant $R \in (0,1)$, $m \geq 1$, there exist m-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*~~

~~**Theorem B.** *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*~~

Theorem C. *m-variate GAP codes are locally testable with $O(n^{2/m})$ queries.*

Main Results

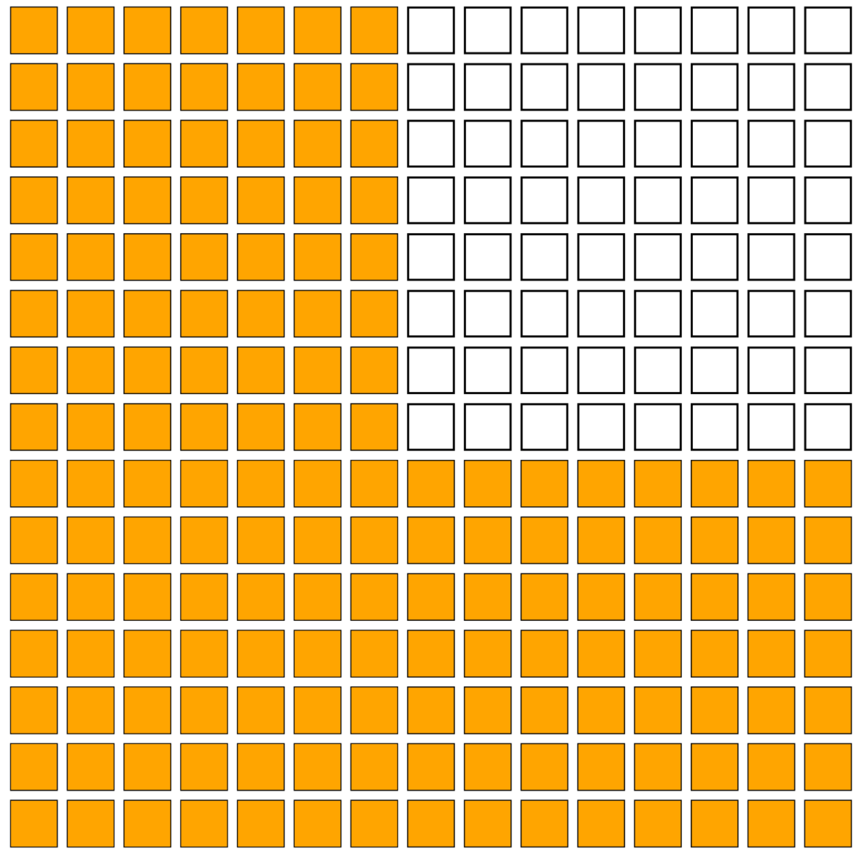
~~**Theorem A.** *For any constant $R \in (0,1)$, $m \geq 1$, there exist m-variate polynomial evaluation codes (CAP and GAP codes) with rate R and constant relative distance.*~~

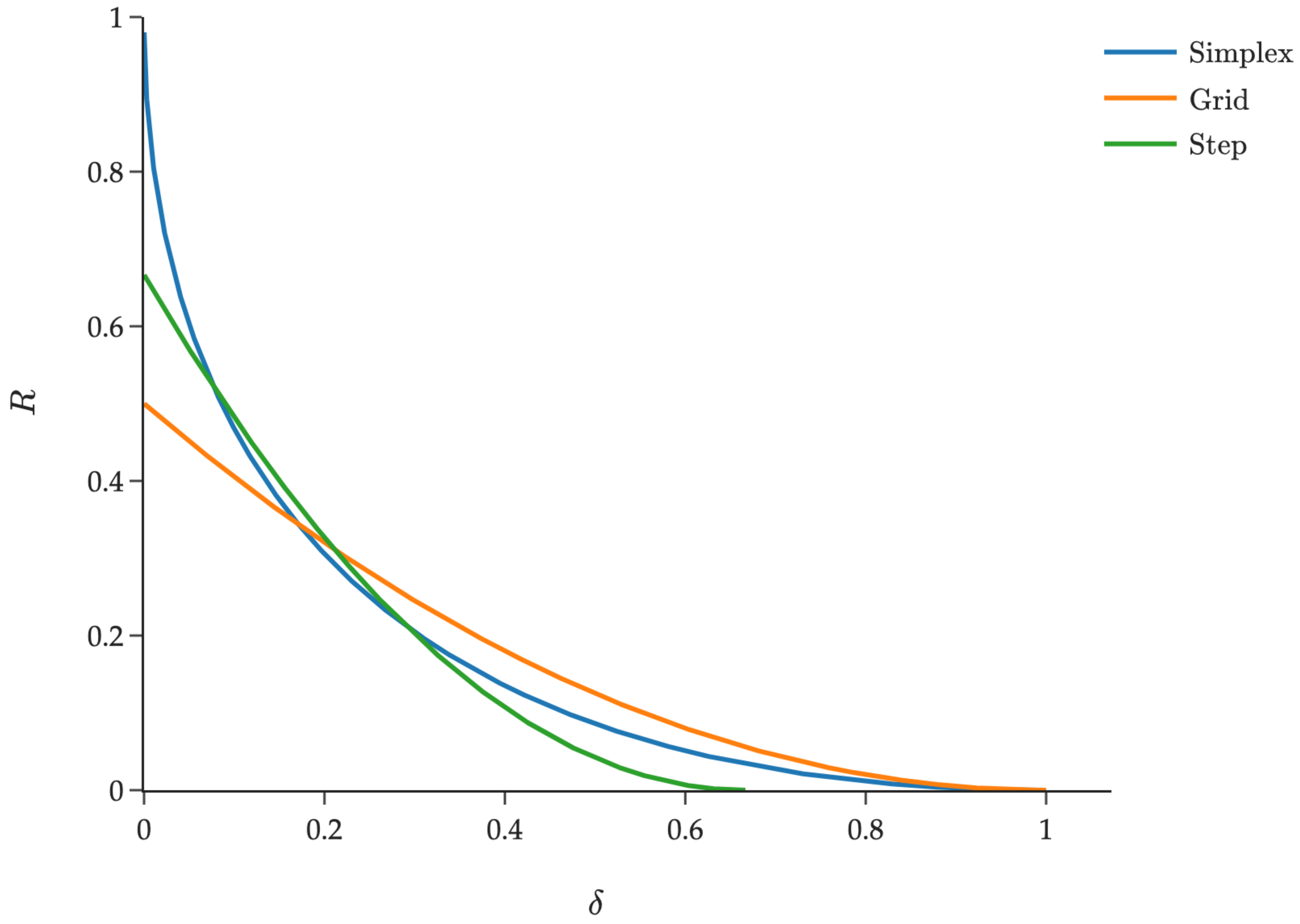
~~**Theorem B.** *CAP and GAP codes can be uniquely decoded in polynomial time from up to half of the minimum distance.*~~

~~**Theorem C.** *m-variate GAP codes are locally testable with $O(n^{2/m})$ queries.*~~

Future directions

- What other properties do CAP and GAP codes have?
- Growing m
- Better CAP codes





Thank you!