Algebraic-Geometric Code and its Soft-Decision Decoding Algorithm

- Dr. Li Chen
  - Lecturer, School of Information Science and Technology, Sun Yat-sen University
  - BSc, MSc, PhD, MIEEE
  - Website: http://sist.sysu.edu.cn/~chenli

Personal Background

- Education and employment
  - 2003, BSc in Applied Physics, Jinan University, China
  - 2004, MSc in Communications and Signal Processing, Newcastle University, UK
  - 2008, PhD in Mobile Communications, Newcastle University, Supervisor: Prof. R. A. Carrasco (IEE Fellow)
  - 2007 – 2010, Research Associate, Newcastle University, engaged with an EPSRC project.
  - 2010 -- .., Lecturer, Sun Yat-sen University

- Research Interests
  - Information theory and channel coding
  - Cooperative system

Outline

- Construction of AG codes: Hermitian Codes
- Overview of the algebraic decoding algorithm
- Algebraic soft-decision decoding of Hermitian codes
- Performance evaluation
- Future work

Construction of Hermitian Codes

- Hermitian Curve: $H_w(x, y, z) = x^{w+1} + y^w z + y z^w$
  - Affine component: $H_w(x, y, 1) = x^{w+1} + y^w + y$ – used for code construction!
- Size of GF (GF(q)) decides the degree of the curve: $w = \sqrt{q}$
- Genus of the curve: $g = w(w-1)/2$
- Designed distance of a (n, k) Hermitian code: $d^* = n - k - g + 1$
- Number of affine points $p_i = (x_i, y_i)$, $|p_i| = w^3$
  - It decides the length of the code
- Code parameters that we can achieve:

GF(q) / Paras	GF(4)	GF(16)	GF(64)	GF(256)
deg	3	5	9	17
g	1	6	28	120
n	8	64	512	4096

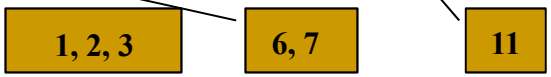
Construction of Hermitian Codes

- Point of infinity p_∞: for points that we can find in $H_w(1, y, z)$, $H_w(x, 1, z)$ and $H_w(x, y, 1)$, the one with the form of $(x_i, y_i, 0)$.
  - Variables x, y, z have a pole order (or weights) at p_∞, $x - w$, $y - w+1$, z -- ? (depends on k).

- Affine points p_i: points on an affine component. E.g. for $H_w(x, y, 1)$, p_i satisfies $H_w(x_i, y_i, 1) = 0$.

- Pole basis L_w: a set of rational functions Φ_α with increasing pole orders
  - Curve H_2 has $L_2 = \{1, x, y, x^2, xy, y^2, x^2y, xy^2, y^3, x^2y^2, xy^3, y^4, \ldots\}$

  - Curve H_4 has $L_4 = \{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, x^4, x^3y, x^2y^2, xy^3, y^4, x^4y, x^3y^2, x^2y^3, xy^4, y^5, \ldots\}$



- Zero basis $Z_{w,pi}$: a set of rational functions $\psi_{w,pi}$ with increasing zero orders at p_i.

Construction of Hermitian Codes

- For a Hermitian code defined on the curve H_w:
  - Find out n affine points on the curve – decide the length of the code
  - Select the first k monomials in L_w – decide the dimension of the code
  - With information symbols $(f_0, f_1, \ldots, f_{k-1}) \in \mathrm{GF}(q)$, the message polynomial can be written as:
    $$f(x, y) = f_0\Phi_0 + f_1\Phi_1 + \ldots + f_{k-1}\Phi_{k-1}$$
  - And the code word is generated by:
    $$(c_0, c_1, \ldots, c_{n-1}) = (f(p_0), f(p_1), \ldots, f(p_{n-1}))$$

- Example: Construct a (8, 4) Hermitian code
  - Curve: $H_2 = x^3 + y^2 + y$
  - Affine points $p_0 = (0, 0)$, $p_1 = (0, 1)$, $p_2 = (1, \sigma)$, $p_3 = (1, \sigma^2)$, $p_4 = (\sigma, \sigma)$, $p_5 = (\sigma, \sigma^2)$, $p_6 = (\sigma^2, \sigma)$, $p_7 = (\sigma^2, \sigma^2)$.
  - Information symbols 1, σ, 1, σ^2, and message polynomial $f(x, y) = 1 + \sigma x + y + \sigma^2 x^2$.
  - Code word $(c_0, c_1, \ldots, c_7) = (1, 0, \sigma, \sigma^2, \sigma, \sigma^2, \sigma^2, \sigma)$.

A Comparison with RS Codes

Properties \ Codes	(n, k) RS code	(n, k) Hermitian code
Algebraic affine curves	$y = 0$	$x^{w+1} + y^w + y = 0$
Pole basis	$1, x, x^2, x^3, \ldots$	$1, x, y, x^2, xy, y^2, \ldots, x^w y, x^{w-1}y^2, \ldots, xy^w, y^{w+1}, \ldots$
Affine points (p)	$x_0, x_1, x_2, \ldots, x_{n-1}$	$(x_0, y_0), (x_1, y_1), (x_2, y_2), \ldots, (x_{n-1}, y_{n-1})$
Transmitted message polynomial (f)	$f(x) = f_0 + f_1 x + f_2 x^2 + \ldots + f_{k-1} x^{k-1}$	$f(x, y) = f_0 + f_1 \phi_1 + f_2 \phi_2 + \ldots + f_{k-1} \phi_{k-1}$
Code word ($\bar{c}$)	$(c_0, c_1, \ldots, c_{n-1}) = (f(x_0), f(x_1), \ldots, f(x_{n-1}))$	$(c_0, c_1, \ldots, c_{n-1}) = (f(p_0), f(p_1), \ldots, f(p_{n-1}))$

A Comparison with RS codes

- Advantage of AG codes: larger codes can be constructed from the same finite field as RS codes, resulting better error-correction capability;

- Example, over GF(64)

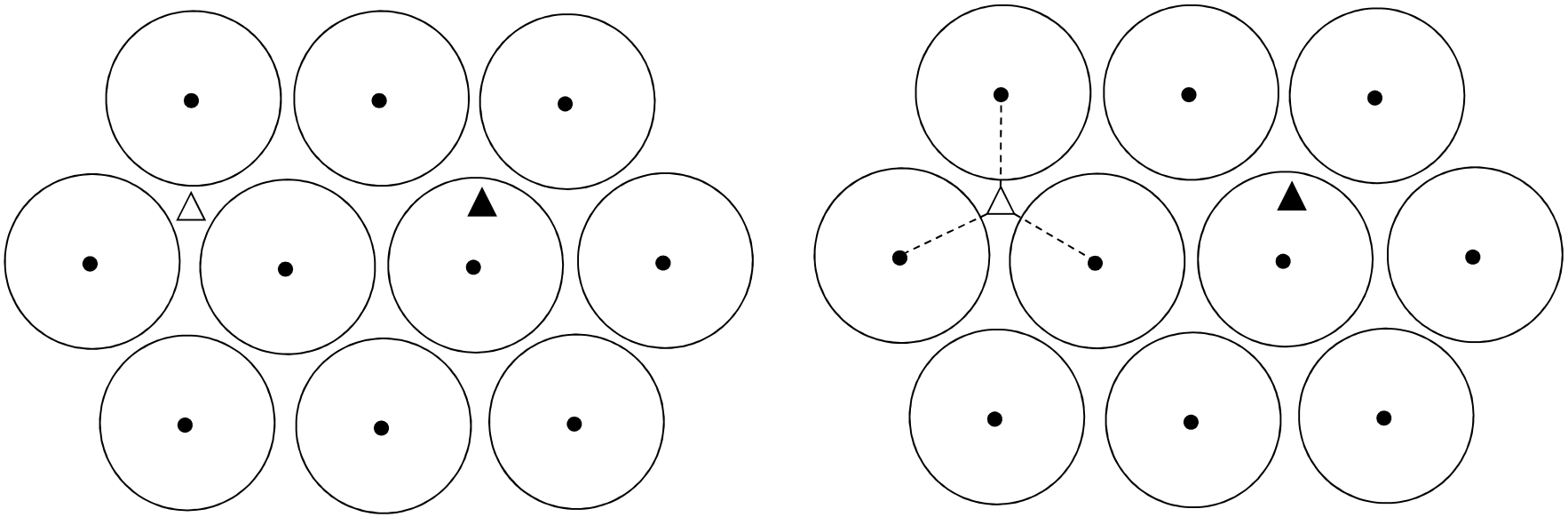
Rate 0.3		Rate 0.56	
Herm (512, 153)	RS (63, 19)	Herm (512, 289)	RS (63, 35)
$d^* = 332$	$d = 45$	$d^* = 196$	$d = 29$
$\tau = 165$	$\tau = 22$	$\tau = 97$	$\tau = 14$
990 bits	132 bits	582 bits	84 bits

- Disadvantage of AG codes: It is not a Maximum Distance Separable (MDS) code.

Overview of the algebraic decoding algorithm

- Decoding philosophy evaluation

Unique decoding ⟶ **List decoding**

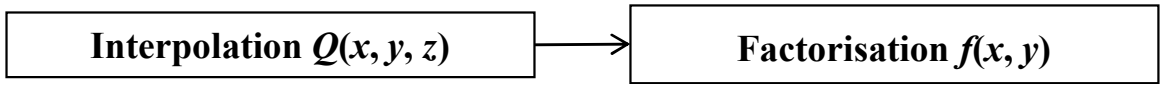


Overview of the list decoding algorithm

- Key processes: Interpolation (construct $Q(x, y, z)$) + Factorisation (find out $f(x, y)$)

- From hard-decision decoding to soft-decision decoding

Hard-decision received word: $\overline{R} = (r_0, r_1, \ldots, r_{n-1})$
Interpolated points: $(p_0, r_0), (p_1, r_1), \ldots, (p_{n-1}, r_{n-1})$
With certain multiplicity value *m*, perform:



Soft-decision reliability matrix Π (→ M)

	p_0 r_0	p_1 r_1	p_2 r_2	p_3 r_3	p_4 r_4	p_5 r_5	p_6 r_6	p_7 r_7	
$\Pi =$	0.96	0.21	0.01	0.46	0.00	0.00	0.69	0.00	**0**
	0.00	0.02	0.00	0.53	0.90	0.03	0.00	0.01	**1**
	0.03	0.74	0.03	0.01	0.10	0.02	0.28	0.99	σ
	0.01	0.03	0.94	0.00	0.00	0.95	0.03	0.00	σ^2
	$(p_0, 0)$	$(p_1, 0)$ (p_1, σ)	(p_2, σ^2)	$(p_3, 0)$ $(p_3, 1)$	$(p_4, 1)$ (p_4, σ)	(p_5, σ^2)	$(p_6, 0)$ (p_6, σ)	(p_7, σ)	

where a multiplicity value m was assigned to the unit

Design of an algebraic soft-decision decoding algorithm

- Key Challenges:
  - How to process trivariate monomials (polynomials)
  - Define the interpolated zero conditions
  - Calculate the corresponding coefficients of a Hermitian curve
  - Prove the validity of the algorithm
  - Optimal performance bound
  - Complexity reduction methods

Trivariate monomials (Polynomials)

- For a code defined on the curve $H_w = x^{w+1} + y^w + y$,
  - monomial $x^i y^j z^k$, $0 \le i \le w$, $j \ge 0$ and $k \ge 0$
  - Decoding a (n, k) Hermitian codes, $\deg_w(z) = \deg_w(\Phi_{k-1})$
  - $\deg_w(x^i y^j z^k) = iw + j(w+1) + k\deg_w(z)$
  - For to monomials $x^{i1} y^{j1} z^{k1}$ and $x^{i2} y^{j2} z^{k2}$

    $$x^{i1} y^{j1} z^{k1} < x^{i2} y^{j2} z^{k2}$$

    if $\deg_w(x^{i1} y^{j1} z^{k1}) < \deg_w(x^{i2} y^{j2} z^{k2})$, or $\deg_w(x^{i1} y^{j1} z^{k1}) = \deg_w(x^{i2} y^{j2} z^{k2})$ and $k1 < k2$.
  - A lexicographic order can be assigned to monomials.

- Polynomials $Q(x, y, z) = \sum_{a,b \in N} Q_{ab} \phi_a(x, y) z^b$, $Q_{ab} \in \mathrm{GF}(q)$
  - Identify the maximal monomial in $Q(x, y, z)$ as $\Phi_{a'} z^{b'}$, then $\deg_w(Q) = \deg_w(\Phi_{a'} z^{b'})$
  - Leading order, $\mathrm{lod}(Q) = \mathrm{ord}(\Phi_{a'} z^{b'})$

- $N_w(\delta) = |\{\phi_a z^b : \deg_w(\phi_a z^b) \le \delta, (a, b, \delta) \in N\}|$ **Define the number of monomials**

  $\Delta_w(v) = \min\{\delta : N_w(\delta) > v, v \in N\}$ **Define the weighted degree of monomials**

Define the Interpolated Zero Conditions

- To interpolate unit (p_i, r_i) (or (x_i, y_i, r_i))
- Recall the zero basis $Z_{w,pi}$ with rational functions $\psi_{pi,\alpha}$ as:

$$\psi_{p_i,\alpha} = \psi_{p_i,\lambda+(w+1)\delta} = (x-x_i)^{\lambda}[(y-y_i)-x_i^{w}(x-x_i)]^{\delta}, (0 \le \lambda \le w, \delta \ge 0)$$

- Zero condition with multiplicity m for polynomial Q(x, y, z) = $\sum_{a,b\in N} Q_{ab}\phi_a(x,y)z^b$
  - It can be written as: $Q(x,y,z) = \sum_{\alpha,\beta\in N} Q_{\alpha\beta}^{(p_i,r_i)}\psi_{p_i,\alpha}(z-r_i)^{\beta}$
  - $Q_{\alpha\beta}^{(p_i,r_r)} = 0$ for $\alpha + \beta < m$.
- Since $\phi_a = \sum_{\alpha\in N}\gamma_{a,p_i,\alpha}\psi_{p_i,\alpha}$ and $z^b = \sum_{\beta\le b}\binom{b}{\beta} r_i^{b-\beta}(z-r_i)^{\beta}$

$$Q_{\alpha\beta}^{(p_i,r_i)} = \sum_{a,b\ge\beta} Q_{ab}\binom{b}{\beta}\gamma_{a,p_i,\alpha} r_i^{b-\beta}$$

A key parameter for determining the polynomial's zero condition!

Calculate the Corresponding Coefficients

- Fact: $\phi_a = \sum_{\alpha \in N} \gamma_{a,p_i,\alpha} \psi_{p_i,\alpha} \longleftrightarrow \psi_{p_t,\alpha} = \sum_{a \in N} \zeta_a \phi_a$ and $\psi_{p_t,\alpha} = \sum_{a \in N, a < L} \zeta_a \phi_a + \phi_L$.

- *Algorithm A:* Determining the corresponding coefficients $\gamma_{a,p_t,\alpha}$ between a pole basis monomial ϕ_a and zero basis functions $\psi_{p_t,\alpha}$.
  Step 1: Initialise all corresponding coefficients $\gamma_{a,p_t,\alpha} = 0$;
  Step 2: Find the zero basis function $\psi_{p_t,\alpha}$ with $LM(\psi_{p_t,\alpha})$ = ϕ_a, and let $\gamma_{a,p_t,\alpha} = 1$;
  Step 3: Initialise function $\hat{\psi} = \psi_{p_t,\alpha}$;
  Step 4: While $(\hat{\psi} \neq \phi_a)$ {
  Step 5: Find the second largest pole basis monomial ψ_{L-1} with coefficient ζ_{L-1} in $\hat{\psi}$;
  Step 6: In Z_{w,p_t}, find a zero basis function $\psi_{p_t,\alpha}$ whose leading monomial $LM(\psi_{p_t,\alpha}) = \phi_{L-1}$, and let the corresponding coefficient $\gamma_{a,p_t,\alpha} = \zeta_{L-1}$;
  Step 7: Update $\hat{\psi} = \hat{\psi} + \gamma_{a,p_t,\alpha}\psi_{p_t,\alpha}$;
  }

[ref] - , Complexity reducing interpolation for list decoding Hermitian codes, IEEE Trans. Wireless Commun, 2008.

Prove the Validity of the Algorithm

- Condition 1: From the perspective of solving a linear equation group

$$N_w(\delta) > C_M$$

Freedom (Nr of coefficients) **Constraints**

- Condition 2: From the perspective of solving equation $Q(x, y, f) = 0$

$$S_M(\overline{C}) > \deg_w(Q(x, y, z))$$

Total zero order of *Q* **Pole order of *Q***

- An important Lemma:

Lemma 1: If $Q \in F_q[x, y, z]$ has a zero of multiplicity at least m over unit (p, ρ) (p is an affine point and $\rho \in \mathrm{GF}(q)$), and there exists a polynomial h in the form of equation (2) such that $h(p) = \rho$, then $Q(x, y, h)$ has a zero order $v_p(Q(x, y, h)) \geq m$ at affine point p, or alternatively $\psi_{p,m} | Q(x, y, h)$ [14].

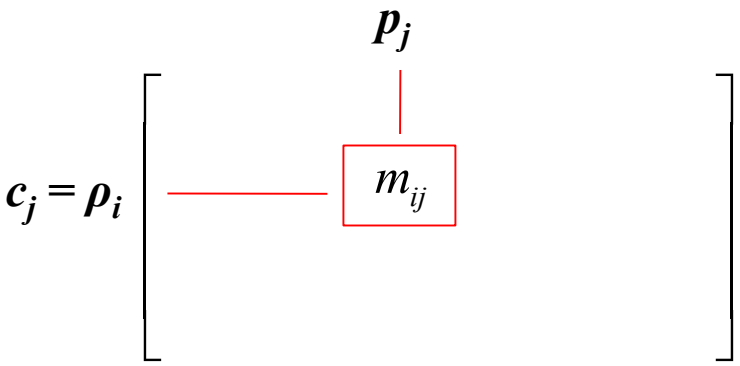
Prove the Validity of the Algorithm

- Theorem 2: Given the multiplicity matrix **M** and the resulting interpolated polynomial $Q(x, y, z)$, if the codeword score $S_M(\underline{c})$ is large enough such that:

$$S_M(\underline{c}) > \deg_w(Q(x, y, z))$$

message polynomial f can be found out by factorising Q as: $z - f \mid Q(x, y, z)$ or $Q(x, y, f) = 0$. → This gives a tight condition of successful list decoding!!!

- Since $f(p_j) = c_j$, according to Lemma 1, in matrix **M**, only …… values will contribute to the total code word score of $Q(x, y, f)$. Those value as marked as $\hat{m}_{i,j}$



$$\psi_{p_0,\hat{m}_{t,0}} \cdot \psi_{p_1,\hat{m}_{t,1}} \cdots \psi_{p_{n-1},\hat{m}_{t,n-1}} | Q(x, y, f).$$

$$\sum_{j=0}^{n-1} v_{p_j}(Q(x, y, f)) \geq S_M(\overline{c}).$$

$$\sum_{j=0}^{n-1} v_{p_j}(Q(x, y, f)) > \deg_w Q(x, y, f)$$

[ref] - , Soft-decision list decodign of Hermitian codes, IEEE Trans. Commun, 2009.

Prove the Validity of the Algorithm

- A corollary that can embrace both of the successful decoding conditions.

  Corollary 3: Message polynomial f can be found out by $(z - f) \mid Q(x, y, z)$ if

  $$S_M(\bar{c}) > \Delta_w(C_M)$$

  Since $\Delta_w(C_M)$ guarantees $N_w(\delta) > C_M$ (Condition 1 is met!)

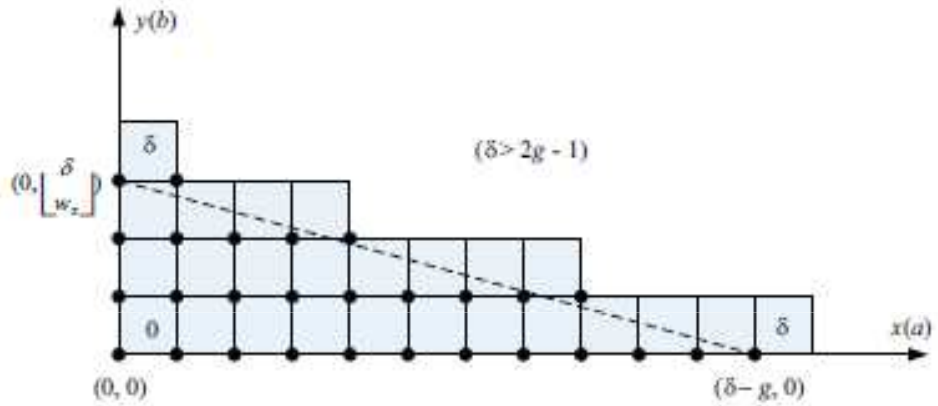
  Since $\deg_w(Q(x, y, z)) \le \Delta_w(C_M)$, if $S_M(\bar{c}) > \Delta_w(C_M)$, $S_M(\bar{c}) > \deg_w(Q)$ (Condition 2 is met!)

  This can be seen later.

- Remark: Solving the linear polynomial group does not give a tight bound on successful list decoding, but solving the polynomial $Q(x, y, f) = 0$ does!

Optimal Performance Bound

- Corollary 4: Let $w_z = \deg_w(\Phi_{k-1})$, $N_w(\delta) > \delta(\delta - g)/2w_z$ given $\delta > 2g - 1$. And $N_w(\delta) = \delta^2/2w_z$ with $\delta \to \infty$.



- With $l \to \infty$, list decoding algorithm's asymptotic optimal performance can be achieved.

  $l \to \infty$, $C_M \to \infty$ and $\Delta_w(C_M) \to \infty$, it results $\Delta_w(C_M) \cong \sqrt{2w_z C_M}$
- Corollary 3 ($S_M(\underline{c}) > \Delta_w$ (CM)) can be interpreted as:

$$\sum_{j=0}^{n-1} \widehat{m}_{i,j} > \sqrt{w_z \sum_{i=0}^{q-1} \sum_{j=0}^{n-1} m_{i,j}(m_{i,j}+1)}.$$

Optimal Performance Bound

- Asymptotic condition (when $C_M \rightarrow \infty$): $\frac{\pi_{i,j}}{n} = \frac{m_{i,j}}{s}$
  - Note $\sum_{i,j} m_{i,j} = s$ and $\sum_{i,j} \pi_{i,j} = n$
- We could further have

$$\frac{s}{n}\sum_{j=0}^{n-1}\widehat{\pi}_{i,j} > \frac{s}{n}\sqrt{w_z \sum_{i=0}^{q-1}\sum_{j=0}^{n-1}\pi_{i,j}(\pi_{i,j}+\frac{n}{s})}.$$

- Since with $s \rightarrow \infty$, $n/s \rightarrow 0$ and

$$\sum_{j=0}^{n-1}\widehat{\pi}_{i,j} > \sqrt{w_z \sum_{i=0}^{q-1}\sum_{j=0}^{n-1}\pi_{i,j}^2}.$$

In KV decoding of RS codes, w_z is replaced by k - 1

Complexity Reduction Methods

- Modified reliability transform algorithm (introducing a stopping criterion)
- Pre-calculation of the corresponding coefficients
- Elimination of the unnecessary polynomials in the group

Complexity Reduction Methods

- Modified reliability transform algorithm

- Problem: In the original KV algorithm, the iterative **Π** → **M** transform process will stop once a pre-set value of **s** = $\sum_{i,j} m_{i,j}$ is met. However, it will be a waste by just increasing the total multiplicity values, but NOT the actual output list size.

- The actual output list size $l_M = \deg_z(Q) = \left\lfloor \frac{\deg_w Q(x,y,z)}{w_z} \right\rfloor$

- Challenge: before the interpolation process, we do not know $Q(x, y, z)$.

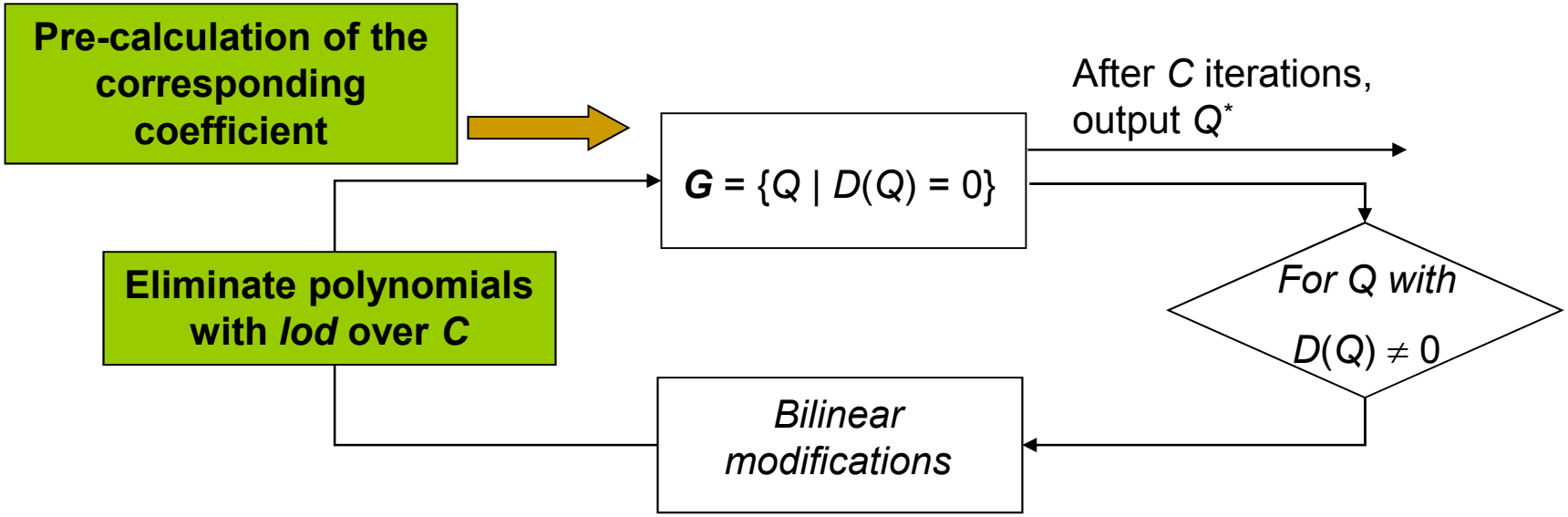
- However, $\deg_w(Q(x, y, z)) \le \Delta_w(C_M)$?

- Corollary: $\Delta_w(C_M) = \deg_w(\Phi_a z^b \mid \mathrm{ord}(\Phi_a z^b) = C_M)$

- $l_M = \left\lfloor \frac{\deg_w(\phi_a z^b \mid ord(\phi_a z^b) = C_M)}{w_z} \right\rfloor$ → if $l_M > l$, then stop! Minimising *s*!

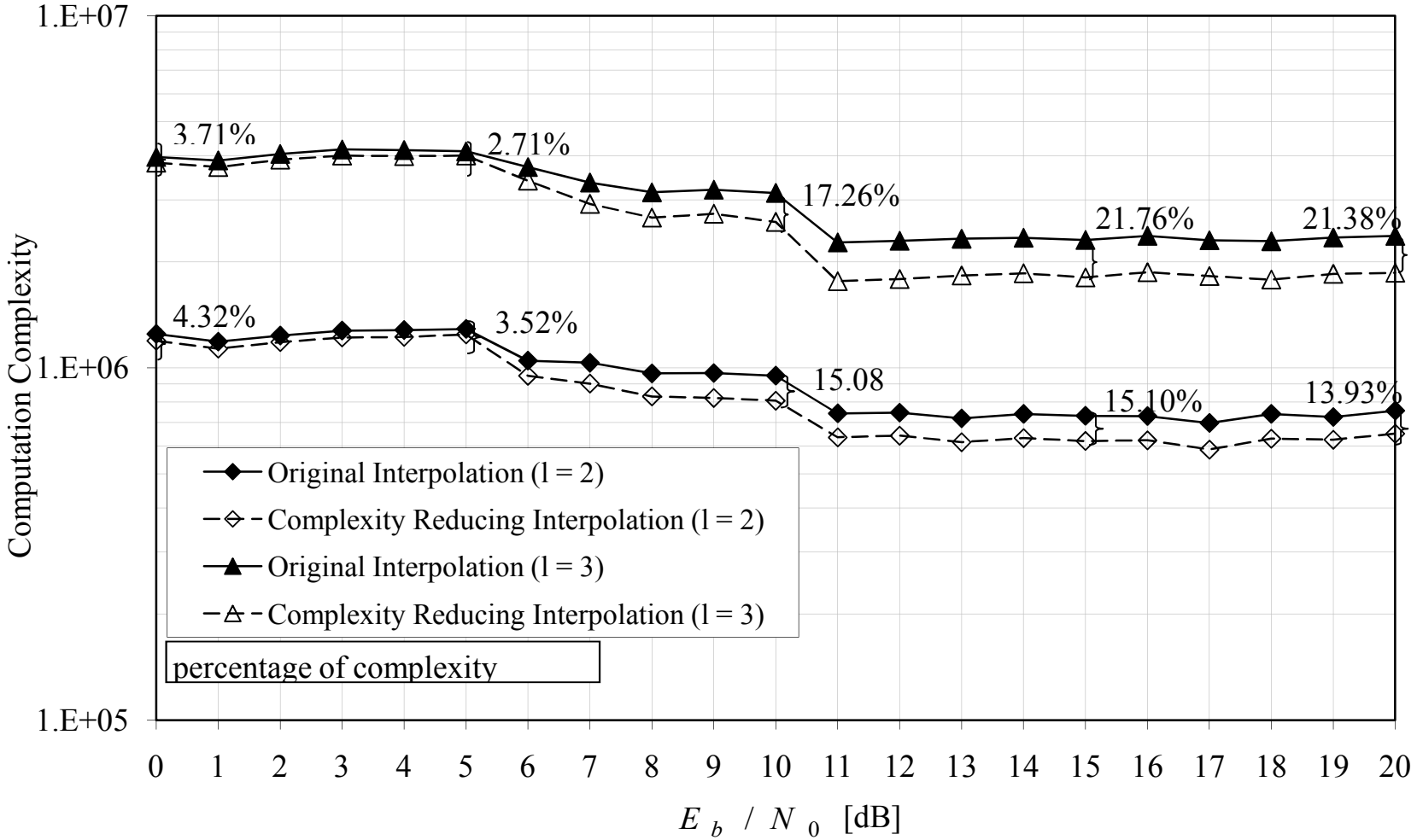
Complexity reducing interpolation

- Pre-calculation of the corresponding coefficients and elimination of the unnecessary polynomials



In the end, the minimal polynomial Q in group G is chosen!

Complexity reducing interpolation



Arising Awareness

- Why Condition 1 ($N_w(\delta) > C_M$) is NOT a tight bound?
- Since $\mathrm{lod}(Q^*) \le C_M$, if $\deg_w(Q^*) = \delta^*$, then

$$N_w(\delta^*) \le C_M \longleftrightarrow N_w(\delta) > C_M$$

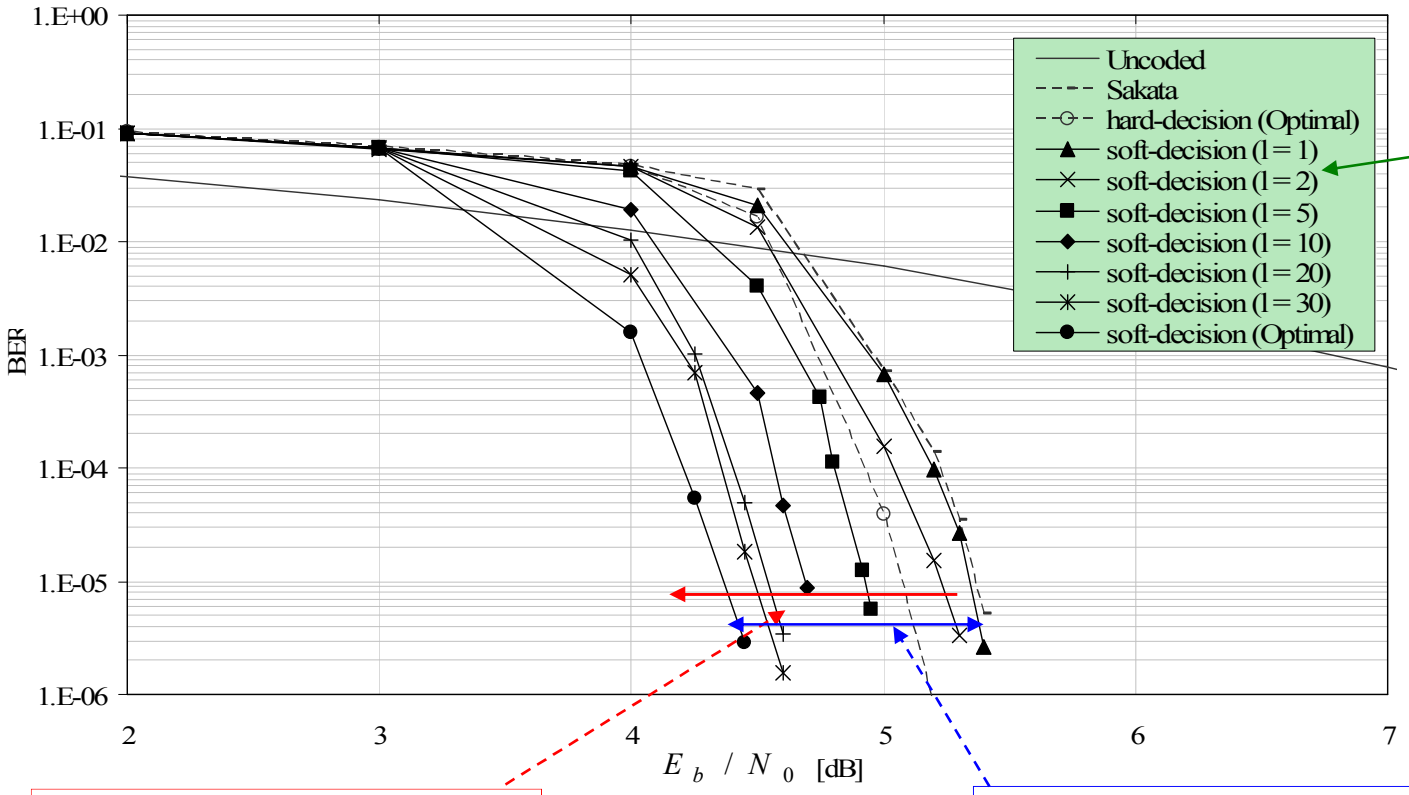
- $N_w(\delta) > C_M$ is the successful decoding criterion w.r.t. the polynomial group G. However, the minimal polynomial in G does not meet this condition.
- To access the decoding performance, only Condition 2 gives a tight bound:

$$S_M(\underline{c}) > \deg_w(Q(x, y, z))$$

- Without performing the interpolation process, the theoretical assessment (e.g. $S_M(\underline{c}) > \Delta_w(C_M)$) produces a relatively negative results.

Performance Evaluation

Hermitian code (512, 289) over AWGN channel



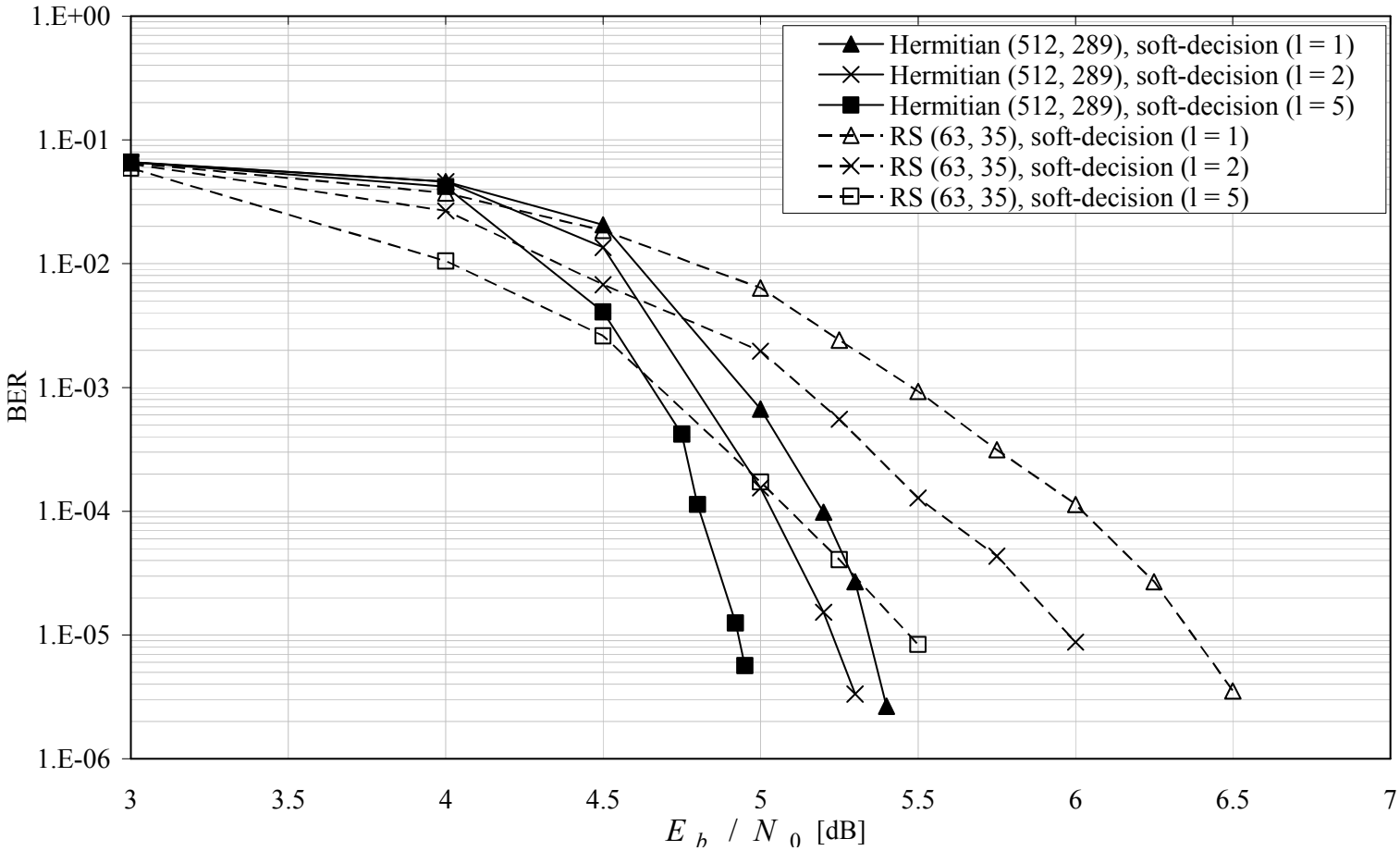
l is the output list size of the list decoder

…. approaching the optimal bound!

…. performance advantage of the list decoding algorithms!

Hermitian code ~ RS code

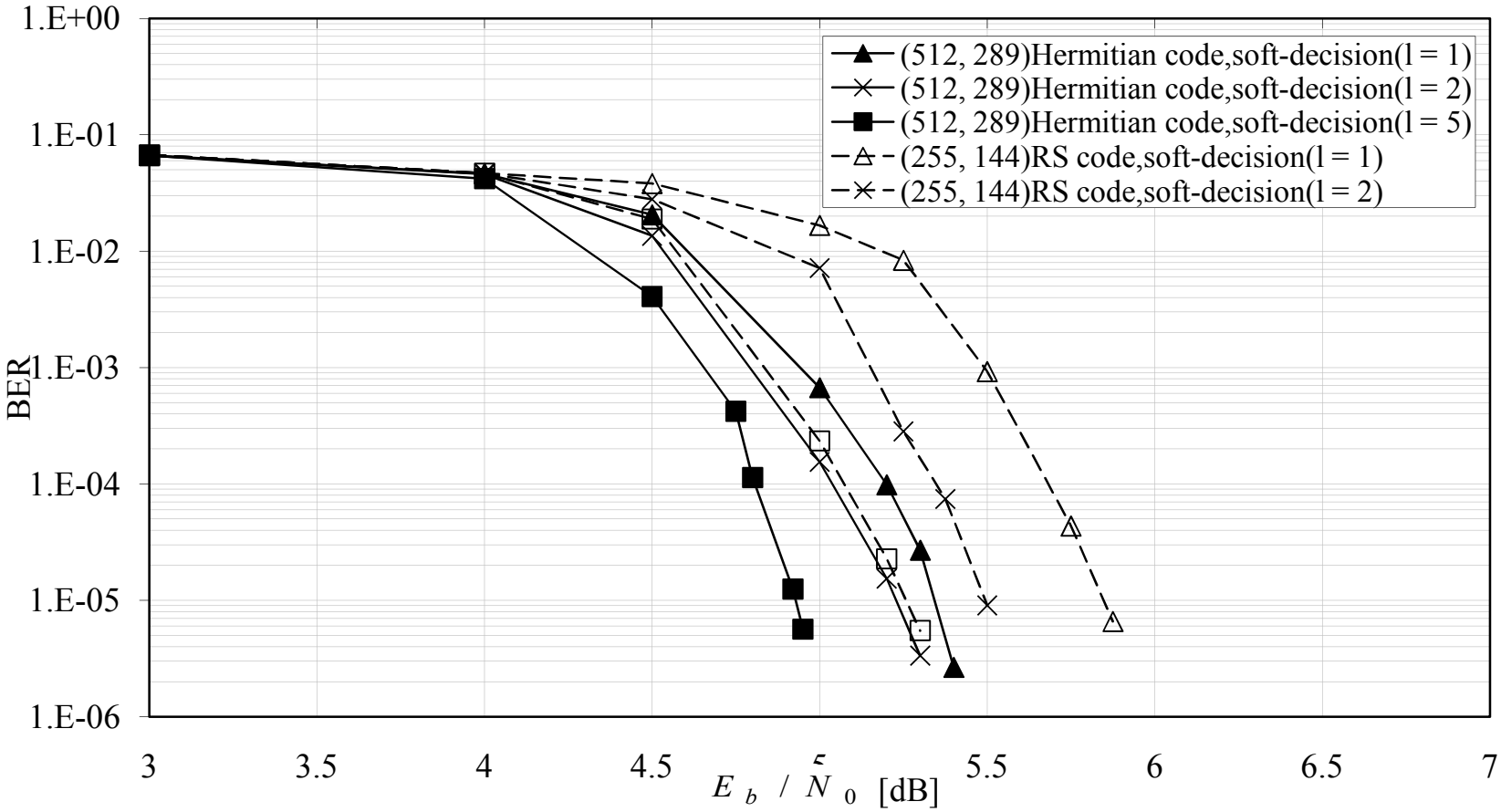
Both codes are defined in GF(64), over AWGN channel



Codes / Output size	Hermitian (512, 289)	RS (63, 35)	RS (255, 144)
$l = 1$	$C = 892$	$C = 103$	$C = 430$
$l = 2$	$C = 1813$	$C = 204$	$C = 859$
$l = 5$	$C = 4602$	$C = 715$	$C = 3004$

Hermitian code ~ RS code

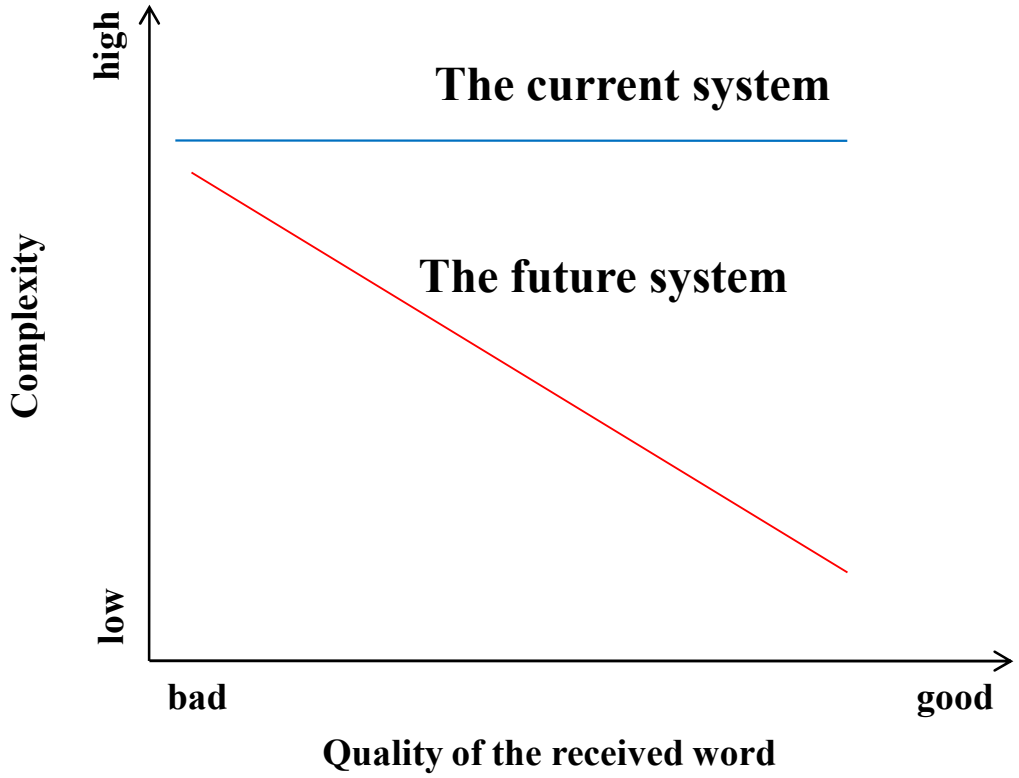
Hermitian code is defined in GF(64) and RS code is defined in GF(256)



Output size \ Codes	Hermitian (512, 289)	RS (63, 35)	RS (255, 144)
$l = 1$	$C = 892$	$C = 103$	$C = 430$
$l = 2$	$C = 1813$	$C = 204$	$C = 859$
$l = 5$	$C = 4602$	$C = 715$	$C = 3004$

A rebound thinking

- A common phenomenon:



Inspiration: Can we design a list decoder which can **SMARTLY ADAPT** its complexity according to the quality of the received word?

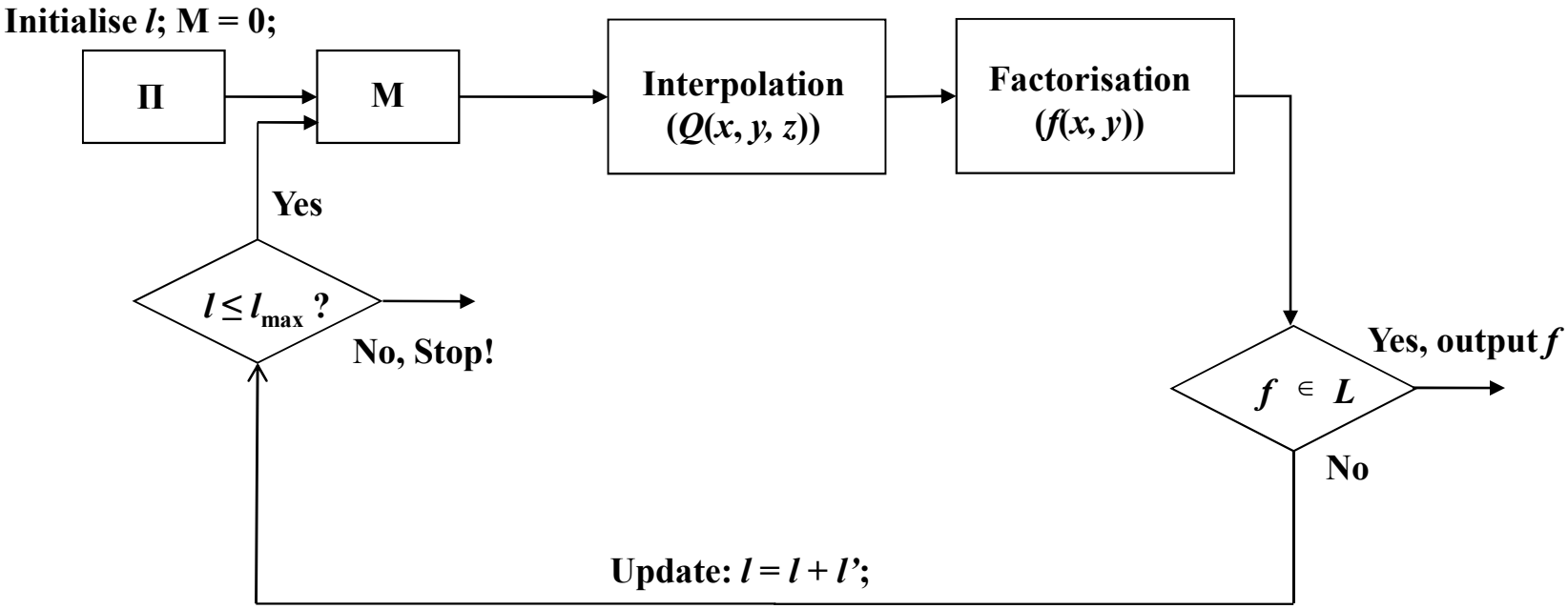
We can 'borrow' the idea of iterative decoding!

Coming soon ….

A rebound thinking

- Iterative algebraic soft-decision decoding

l, designed output list size at each iteration;
l_{max}, designed maximal output list size;
l', update step.



Current funding: Advanced coding technology for future storage devices, National Science Foundation of China (NSFC), 61001094, PI.

Conclusions

- Construction of a Hermitian code and some of its properties;
- Algebraic decoding system (hard → soft)
- Algebraic soft-decision decoding of Hermitian codes
- Performance evaluation and comparison with RS codes
- Future research direction.