Decoding linear codes via systems solving: complexity issues

Stanislav Bulygin (joint work with Ruud Pellikaan)

University of Kaiserslautern

June 19, 2008

Outline

Outline of the talk

- Introduction: codes and decoding problem
- Quadratic system method
- Complexity issues: Macaulay matrix, extended linearization, some estimates

Introduction

Codes recap

- Let $\mathbb{F}_q$ be a field with q elements. A *linear code* C is a linear subspace of $\mathbb{F}_q^n$ endowed with the *Hamming metric*
- *Hamming distance* between $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n : d(\mathbf{x}, \mathbf{y}) = \#\{i | x_i \neq y_i\}$. *Hamming weight* of $\mathbf{x} \in \mathbb{F}_q^n : \mathrm{wt}(\mathbf{x}) = \#\{i | x_i \neq 0\}$.
- Minimum distance of the code $C : d(C) := \min_{\mathbf{x},\mathbf{y} \in C, \mathbf{x} \neq \mathbf{y}}(d(\mathbf{x}, \mathbf{y}))$.
- The code C of dimension k and minimum distance d is denoted as $[n, k, d]$.
- A matrix G whose rows are the basis vectors of C is a *generator matrix*.
- A matrix H with the property $\mathbf{c} \in C \iff H\mathbf{c}^T = 0$ is a *check matrix*.

Introduction

Decoding problem

- *Complete decoding*: Given $\mathbf{y} \in \mathbb{F}_q^n$ and a code $C \subseteq \mathbb{F}_q^n$, so that $\mathbf{y}$ is at distance $d(\mathbf{y}, C)$ from the code, find $\mathbf{c} \in C : d(\mathbf{y}, \mathbf{c}) = d(\mathbf{y}, C)$.

- *Bounded up to half the minimum distance*: Additional assumption $d(\mathbf{y}, C) \leq (d(C) - 1)/2$. Then a codeword with the above property is unique.

Introduction

Decoding via systems solving

One distinguishes between two concepts:

- *Generic decoding*: Solve some system $S(C)$ and obtain some "closed" formulas F. Evaluating these formulas at data specific to a received word $\mathbf{y}$ should yield a solution to the decoding problem. For example for $f \in F : f(syndrome(\mathbf{y}), x) = poly(x)$. The roots of $poly(x) = 0$ yield error positions – general error-locator polynomial f.
- *Online decoding*: Solve some system $S(C, \mathbf{y})$. The solutions should solve the decoding problem.

Computational effort

- Generic decoding. Preprocessing: very hard. Decoding: relatively simple.
- Online decoding. Preprocessing: – . Decoding: hard.

Quadratic system method

Unknown syndrome

Let $\mathbf{b}_1, \ldots, \mathbf{b}_n$ be a basis of $\mathbb{F}_q^n$ and let B be the $n \times n$ matrix with $\mathbf{b}_1, \ldots, \mathbf{b}_n$ as rows. The *unknown syndrome* $\mathbf{u}(B, \mathbf{e})$ of a word $\mathbf{e}$ w.r.t B is the column vector $\mathbf{u}(B, \mathbf{e}) = B\mathbf{e}^T$ with entries $u_i(B, \mathbf{e}) = \mathbf{b}_i \cdot \mathbf{e}$ for $i = 1, \ldots, n$.

Structure constants

For two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$ define $\mathbf{x} * \mathbf{y} = (x_1 y_1, \ldots, x_n y_n)$. Then $\mathbf{b}_i * \mathbf{b}_j$ is a linear combination of $\mathbf{b}_1, \ldots, \mathbf{b}_n$, so there are constants $\mu_l^{ij} \in \mathbb{F}_q$ such that $\mathbf{b}_i * \mathbf{b}_j = \sum_{l=1}^{n} \mu_l^{ij} \mathbf{b}_l$. The elements $\mu_l^{ij} \in \mathbb{F}_q$ are the *structure constants* of the basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$.

MDS matrix

Let B_s be the $s \times n$ matrix with $\mathbf{b}_1, \ldots, \mathbf{b}_s$ as rows ($B = B_n$). Then $\mathbf{b}_1, \ldots, \mathbf{b}_n$ is an *ordered MDS basis* and B an *MDS matrix* if all the $s \times s$ submatrices of B_s have rank s for all $s = 1, \ldots, n$.

Quadratic system method

Check matrix

Let C be an $\mathbb{F}_q$-linear code with parameters $[n, k, d]$. W.l.o.g $n \leq q$. H is a check matrix of C. Let $\mathbf{h}_1, \ldots, \mathbf{h}_{n-k}$ be the rows of H. One can express $\mathbf{h}_i = \sum_{j=1}^{n} a_{ij}\mathbf{b}_j$ for some $a_{ij} \in \mathbb{F}_q$. In other words $H = AB$ where A is the $(n-k) \times n$ matrix with entries a_{ij}.

Known syndrome

Let $\mathbf{y} = \mathbf{c} + \mathbf{e}$ be a received word with $\mathbf{c} \in C$ and $\mathbf{e}$ an error vector. The syndromes of $\mathbf{y}$ and $\mathbf{e}$ w.r.t H are equal and known: $s_i(\mathbf{y}) := \mathbf{h}_i \cdot \mathbf{y} = \mathbf{h}_i \cdot \mathbf{e} = s_i(\mathbf{e})$. They can be expressed in the unknown syndromes of $\mathbf{e}$ w.r.t B: $s_i(\mathbf{y}) = s_i(\mathbf{e}) \sum_{j=1}^{n} a_{ij} u_j(\mathbf{e})$ since $\mathbf{h}_i = \sum_{j=1}^{n} a_{ij}\mathbf{b}_j$ and $\mathbf{b}_j \cdot \mathbf{e} = u_j(\mathbf{e})$.

Quadratic system method

Linear forms

Let B be an MDS matrix with structure constants μ_l^{ij}. Define U_{ij} in the variables $U_1, \ldots, U_n$ by $U_{ij} = \sum_{l=1}^{n} \mu_l^{ij} U_l$.

Quadratic system

The ideal $J(\mathbf{y})$ in $\mathbb{F}_q[U_1, \ldots, U_n]$ is generated by

$$\sum_{l=1}^{n} a_{jl} U_l \quad - \quad s_j(\mathbf{y}) \quad \text{for} \quad j = 1, \ldots, r$$

The ideal $I(t, \mathcal{U}, V)$ in $\mathbb{F}_q[U_1, \ldots, U_n, V_1, \ldots, V_t]$ is generated by

$$\sum_{j=1}^{t} U_{ij} V_j \quad - \quad U_{it+1} \quad \text{for} \quad i = 1, \ldots, n$$

Let $J(t, \mathbf{y})$ be the ideal in $\mathbb{F}_q[U_1, \ldots, U_n, V_1, \ldots, V_t]$ generated by $J(\mathbf{y})$ and $I(t, \mathcal{U}, V)$.

Quadratic system method

Main result

Let B be an MDS matrix with structure constants μ_l^{ij}. Let H be a check matrix of the code C such that $H = AB$ as above. Let $\mathbf{y} = \mathbf{c} + \mathbf{e}$ be a received word with $\mathbf{c} \in C$ the codeword sent and $\mathbf{e}$ the error vector. Suppose that $\mathrm{wt}(\mathbf{e}) \neq 0$ and $\mathrm{wt}(\mathbf{e}) \leq \lfloor (d(C) - 1)/2 \rfloor$. Let t be the smallest positive integer such that $J(t, \mathbf{y})$ has a solution $(\mathbf{u}, \mathbf{v})$ over $\overline{\mathbb{F}_q}$. Then

- $\mathrm{wt}(\mathbf{e}) = t$ and the solution is unique satisfying $\mathbf{u} = \mathbf{u}(\mathbf{e})$.
- the reduced Gröbner basis G for the ideal $J(t, \mathbf{y})$ w.r.t any monomial ordering is

$$\begin{cases} U_i - u_i(\mathbf{e}), i = 1, \ldots, n, \\ V_j - v_j, j = 1, \ldots, t, \end{cases}$$

where $(\mathbf{u}(\mathbf{e}), \mathbf{v})$ is the unique solution.

Quadratic system method

Features

- No field equations.
- The same result holds for the complete decoding.
- The solution lies in the field $\mathbb{F}_q$.
- The equations are at most quadratic.
- After solving $J(t, \mathbf{y})$ decoding is simple: $B^{-1}\mathbf{u}(B, \mathbf{e}) = B^{-1}B\mathbf{e}^T = \mathbf{e}^T$.

Quadratic system method

Analysis

From $J(\mathbf{y})$ one can express some $n-k$ U-variables via k others. Substitution of those in $I(t, \mathcal{U}, V)$ yields a systems of n quadratic equations in $k+t$ variables, thus obtaining *overdetermined* system. Easier to solve when

- With constant k and t, n increases.
- With constant n and t, k decreases.

Simulations

For example for random binary codes with $n = 120$ and $k = 10, \ldots, 40$ one can correct $5 - 20$ errors in ≤ 1000 sec. via computing the reduced Gröbner basis in $\mathrm{SINGULAR}$ or MAGMA.

Complexity issues

Diagonal representation (joint with S.Ovsienko)

Our system is equivalent to

$$\begin{aligned} HX^T &= \mathbf{s} \\ X_i Y_i &= 0, i = 1, \ldots, n \\ \hat{H}_t Y^T &= \hat{\mathbf{s}}_t, \end{aligned}$$

where H is a check matrix of the code C, $\mathbf{s}$ a known syndrome, $X = (X_1, \ldots, X_n)$ and $Y = (Y_1, \ldots, Y_n)$ are new variables, $\hat{H}_t$ is a check matrix of a code with the generator matrix B_t, $\hat{\mathbf{s}}_t$ is a syndrome of the vector $\mathbf{b}_{t+1}$ w.r.t to $\hat{H}_t$.

Complexity issues

Macaulay matrix

Like above one can obtain a system Sys with n quadratic equations and $k+t$ variables, w.l.o.g $X_1, \ldots, X_k, Y_1, \ldots, Y_t$. The monomials that appear in the system are $X_i Y_j, 1 \leq i \leq k, 1 \leq j \leq t$, $X_1, \ldots, X_k, Y_1, \ldots, Y_t$. The total number of monomials appearing in the system is $kt + k + t = (k+1)(t+1) - 1$. One can consider the *Macaulay matrix* of Sys: rows are indexed by the equations, columns by the monomials. Denote the matrix by $M(\text{Sys})$.

Linearization

If $n \geq kt + k + t$ and $M(\text{Sys})$ is full-rank, one can find X_i's by applying Gaussian elimination to $M(\text{Sys})$.

Complexity issues

Macaulay matrix is full-rank

Let C be a random $[n, k]$ code over $\mathbb{F}_q$, defined e.g. by a random full-rank $(n-k) \times n$ check matrix H and let $\mathbf{e}$ be a random error vector over $\mathbb{F}_q$ of weight t. Let $\mathrm{Sys} = \mathrm{Sys}(n, k, t)$ be the corresponding system as above. Then the probability of the fact that $M(\mathrm{Sys})$ has full-rank tends to 1 as n tends to infinity. Experiments suggest that already for small values of n (e.g. 20-30) the statement also holds with quite high probability.

Idea of the proof

Degeneracy of $M(\mathrm{Sys})$ is reduced to the fact that

$$\mathbf{e}_I + C_I \subseteq (\widetilde{B_{t+1}})^{\perp}.$$

Here $\mathbf{e}_I$ and C_I are the vector $\mathbf{e}$ and the code C resp. restricted to some I positions from $\{1, \ldots, n\}$ and $\widetilde{B_{t+1}}$ is a code equivalent to the code B_{t+1} restricted to the same I positions as before.

Complexity issues

Extended linearization

One can try to go further and apply *extended linearization*. Consider binary case, so $X_i^2 = X_i$ for all i. Multiply the system Sys with all monomials in $X_1, \ldots, X_k$ of degree $s < k$. A system, call it Sys_s, obtained in this way has $n(1 + \binom{k}{1} + \cdots + \binom{k}{s})$ equations and $C_s := C_{s-1} + \binom{k}{s+1}(t+1)$ monomials. Denote $\binom{k}{0} + \binom{k}{1} + \cdots + \binom{k}{s} =: f(k,s)$. If we assume that $M(\mathrm{Sys}_s)$ is full-rank, then if

$$n\left(1+\binom{k}{1}+\cdots+\binom{k}{s}\right) = nf(k,s) \geq C_s - 1 = (t+1)f(k,s+1) - 1,$$

then successfull application of Gaussian elimination to $M(\mathrm{Sys})$ is possible.

Complexity issues

Complexity coefficient

If an algorithm $\mathcal{A}$ has complexity $\mathcal{O}(2^{\alpha n})$ in the length of input n, then we say that α is a *complexity coefficient* of algorithm $\mathcal{A}$. Denote $CC_{\mathcal{A}}$.

Minimum distance of a random binary code

Let C be a binary random code with parameters $[n, Rn, \Delta n]$, where $0 < R < 1$ is given. If $n \to \infty$, then almost all codes have $\Delta = H^{-1}(1 - R)$, where $H(\cdot)$ is a *binary entropy function*: $H(x) = -x \log_2 x - (1 - x) \log_2 (1 - x), 0 < x < 1$.

Complexity issues

Estimating degree of extended linearization

If $k = Rn, t = \delta n, s = \lambda n, 0 < R, \delta, \lambda < 1$, then it is sufficient to take

$$\lambda = \delta R$$

for extended linearization to work.

Upper bound for complexity

The upper bound on the complexity coefficient with the extended linearization as above is

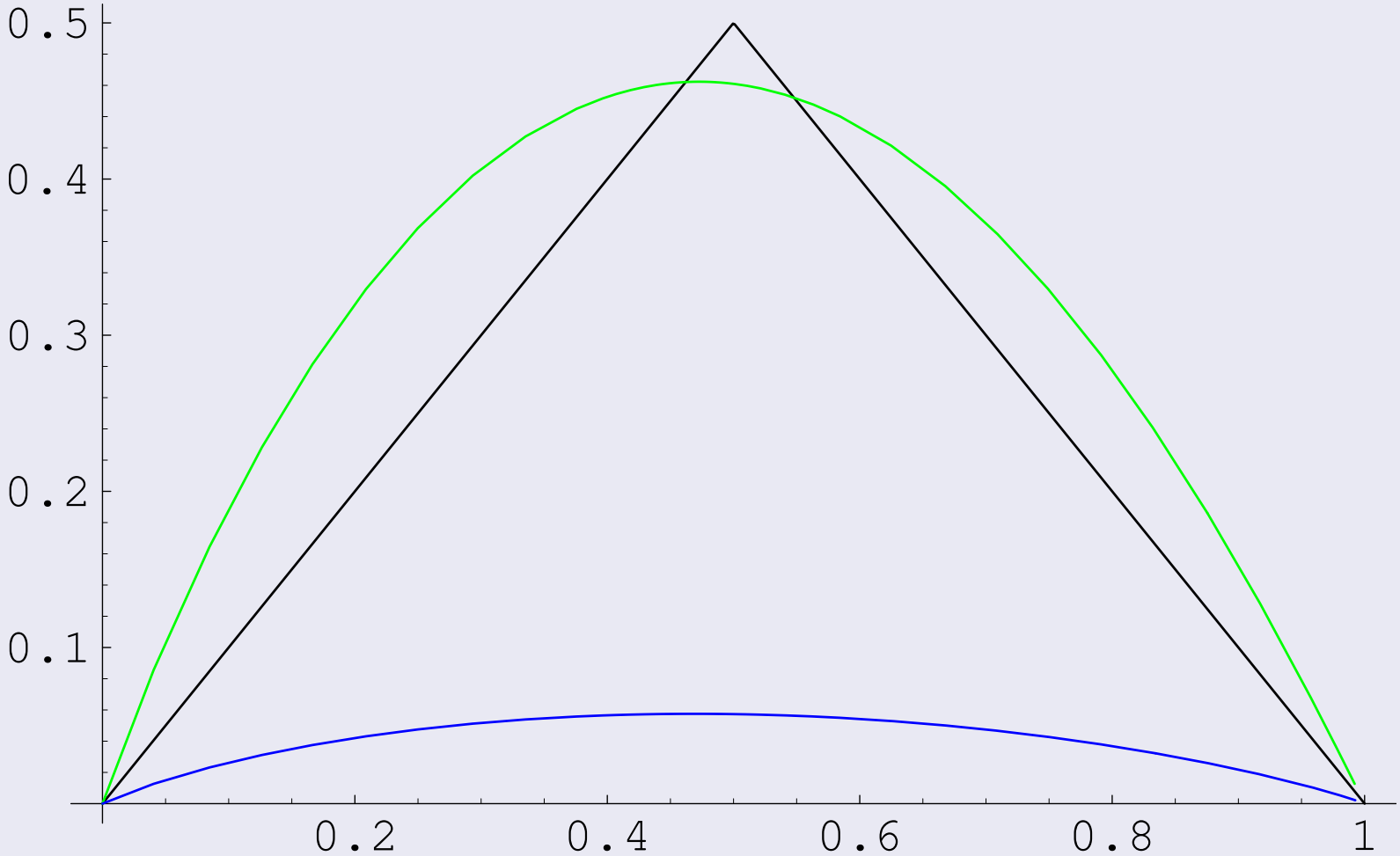
$$CC_{QED}(R) = \omega R H_2(\delta),$$

where ω is the exponent of Gaussian elimination and $\delta \leq \delta_e = H^{-1}(1 - R)/2$ such that $t = \delta n$ is the number of errors occurred.

Complexity issues

Comparing complexities

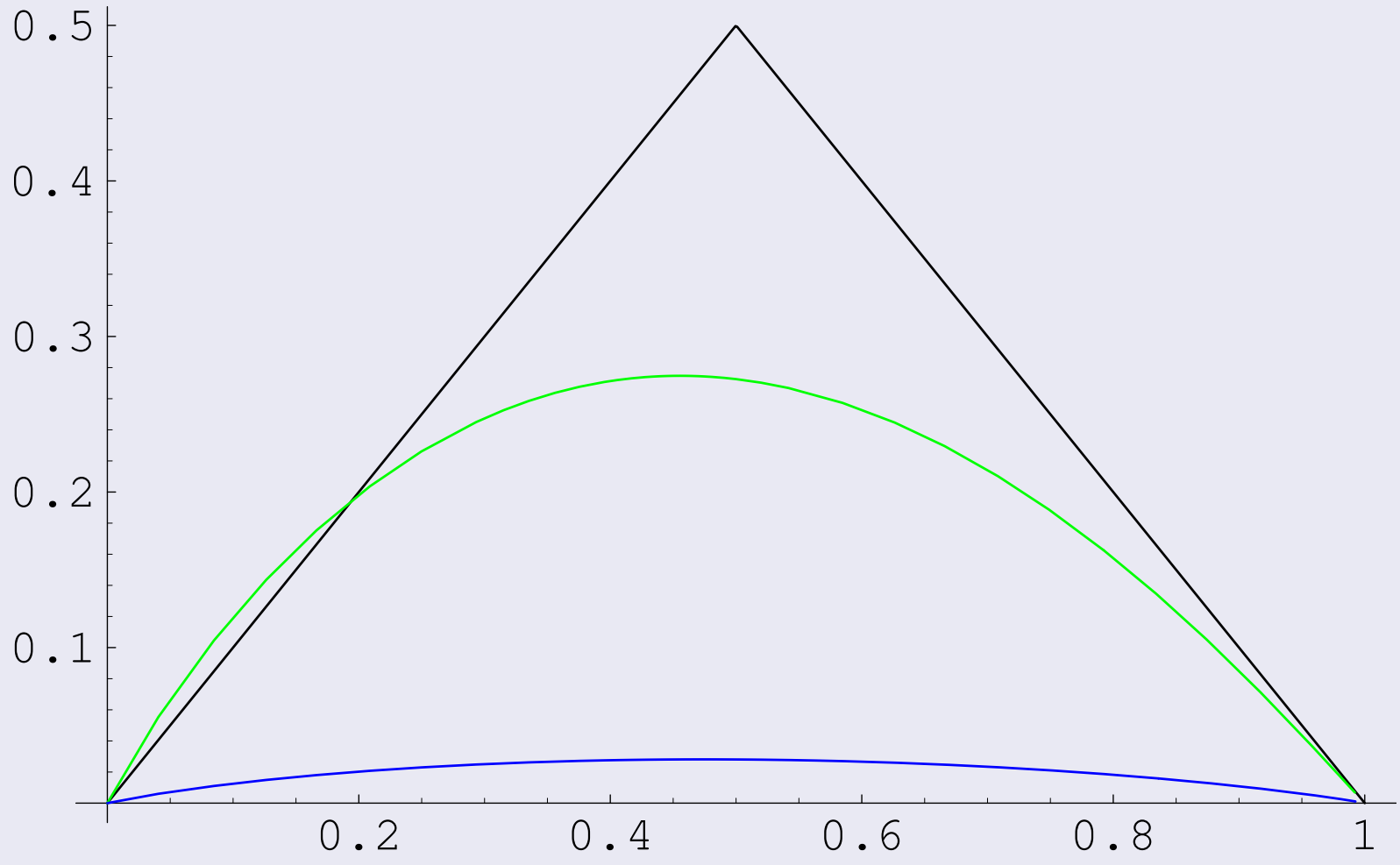
Let $t = \delta_e n$ so the full error-correcting capacity is used. $x-$axis is information rate R, y-axis is the complexity coefficient.



Complexity issues

Comparing complexities II

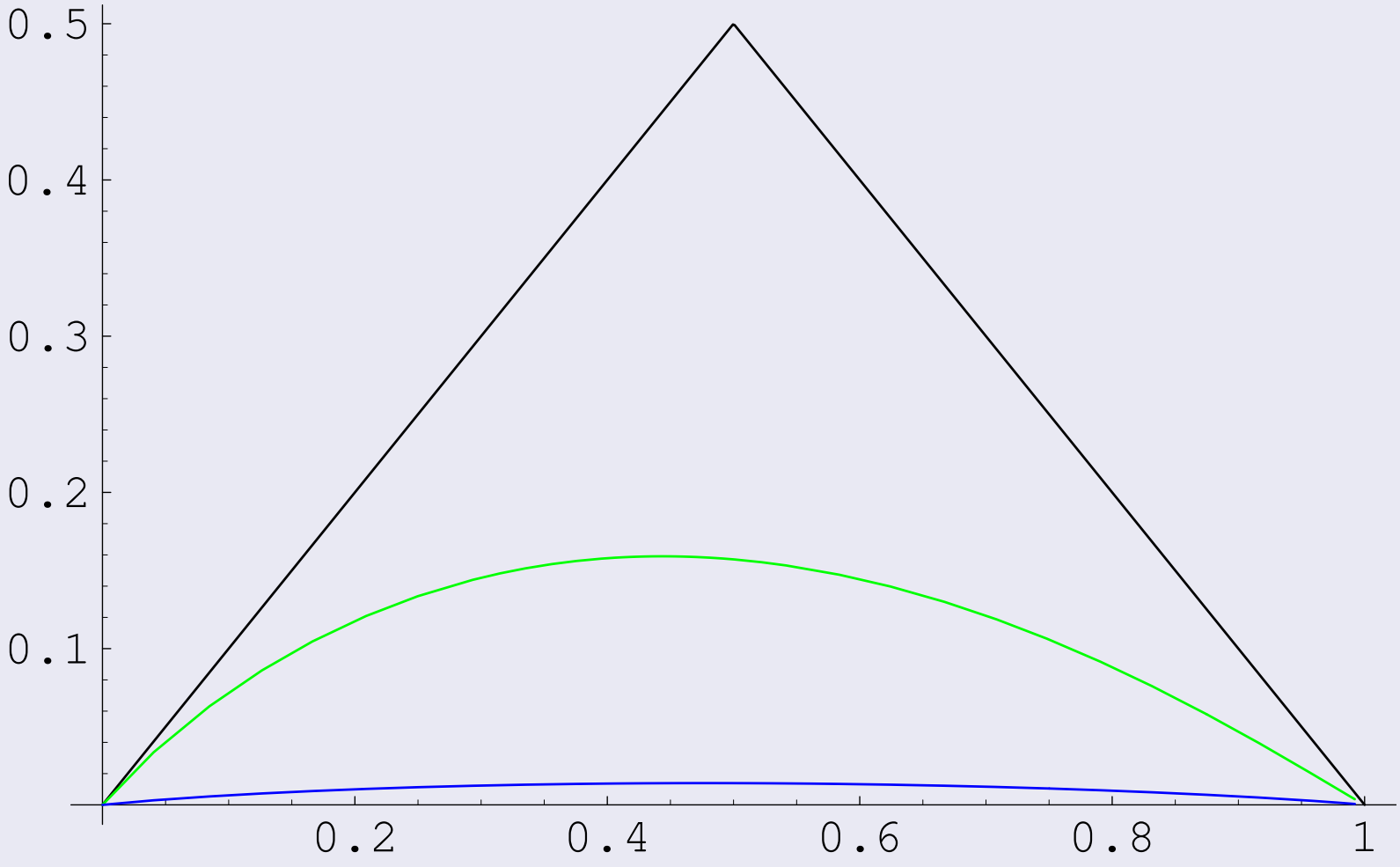
Relax requirement on t a little. Take $t = \delta n$ for $\delta = \delta_e/2$.



Complexity issues

Comparing complexities III

Relax a little more. Take $t = \delta n$ for $\delta = \delta_e/4$.



Complexity issues

Highest degree in GB computation I

In the table below we compare highest degrees when computing GB w.r.t DEGREVLEX with STD command in SINGULAR. For every code the left column shows degree for the "diagonal" system, and the right one for the initial system.

no. of err.	[120,40]		[120,30]		[120,20]		[120,10]		[150,10]	
2	4	4	3	3	3	3	3	3	3	3
3	7	6	5	5	3	3	3	3	3	3
4	6	5	5	6	3	3	3	3	3	3
5	-	-	6	5	6	6	3	3	3	3
6	-	-	5	5	6	5	3	3	3	3
7	-	-	5	5	5	5	3	3	3	3
8	-	-	-	-	5	5	3	3	3	3
9	-	-	-	-	5	5	3	3	3	3
10	-	-	-	-	5	5	3	3	3	3

Complexity issues

Highest degree in GB computation I

no. of err.	[120,40]		[120,30]		[120,20]		[120,10]		[150,10]	
11	-	-	-	-	33	19	6	6	3	3
12	-	-	-	-	-	-	6	5	3	3
13	-	-	-	-	-	-	5	5	4	4
14	-	-	-	-	-	-	5	5	5	6
15	-	-	-	-	-	-	5	5	6	5
16	-	-	-	-	-	-	5	5	5	5
17	-	-	-	-	-	-	5	5	6	5
18	-	-	-	-	-	-	5	5	5	5
19	-	-	-	-	-	-	5	5	5	5
20	-	-	-	-	-	-	6	6	5	5
21	-	-	-	-	-	-	6	5	5	5
22	-	-	-	-	-	-	-	-	5	5
23	-	-	-	-	-	-	-	-	5	5
24	-	-	-	-	-	-	-	-	5	5

Complexity issues

Highest degree in GB computation II

In the next table we show degrees that appear during GB computation with F4 implementation of MAGMA (GroebnerBasis command)

no. of err.	[120,40]	[120,30]	[120,20]	[120,10]	[150,10]
2	3	3	3	3	3
3	3	3	3	3	3
4	4	3	3	3	3
5	-	4	3	3	3
6	-	4	3	3	3
7	-	4	3	3	3
8	-	-	4	3	3
9	-	-	4	3	3
10	-	-	4	3	3
11	-	-	4	3	3

Complexity issues

Highest degree in GB computation II

no. of err.	[120,40]	[120,30]	[120,20]	[120,10]	[150,10]
12	-	-	-	3	3
13	-	-	-	4	3
14	-	-	-	4	3
15	-	-	-	4	3
16	-	-	-	4	4
17	-	-	-	4	4
18	-	-	-	4	4
19	-	-	-	4	4
20	-	-	-	4	4
21	-	-	-	4	4
22	-	-	-	-	4
23	-	-	-	-	4
24	-	-	-	-	4

Complexity issues

Highest degree in GB computation III

We also present degree estimate with extended linearization as above.

no. of err.	[120,40]	[120,30]	[120,20]	[120,10]	[150,10]
2	3	2	2	2	2
3	3	3	2	2	2
4	3	3	2	2	2
5	-	3	3	2	2
6	-	3	3	2	2
7	-	4	3	2	2
8	-	-	3	2	2
9	-	-	3	2	2
10	-	-	3	2	2
11	-	-	4	3	2
12	-	-	-	3	2

Complexity issues

Highest degree in GB computation III

no. of err.	[120,40]	[120,30]	[120,20]	[120,10]	[150,10]
13	-	-	-	3	2
14	-	-	-	3	3
15	-	-	-	3	3
16	-	-	-	3	3
17	-	-	-	3	3
18	-	-	-	3	3
19	-	-	-	3	3
20	-	-	-	3	3
21	-	-	-	3	3
22	-	-	-	-	3
23	-	-	-	-	3
24	-	-	-	-	3

Complexity issues

Highest degree in GB computation IV

Although highest degree of polynomials occurring during GB computations is an important parameter for complexity, it is not the only one. For example in our experiments with SLIMGB we never obtained degree larger than 3, and still running time was much worse than the ones of STD or F4.

Complexity issues

Comparing with different random systems

Consider different types of random systems:

- R_1 is a system of n quadratic equations that has the same monomials as Sys, but the corresponding coefficients are randomly taken from $\mathbb{F}_q$. Require that R_1 has a unique solution in $\overline{\mathbb{F}_q}$.
- R_2 is a system that has the same properties as R_1, but the requirement on uniqueness of a solution is dropped out.
- R_3 is a fully random system of n quadratic equations, i.e. it has all possible monomials of degree ≤ 2 and the corresponding coefficients are random from $\mathbb{F}_q$

Note that R_2 and R_3 do not have solutions in general.

Complexity issues

Experiments

Using some experimental evidence we **conjecture** that there are following relations between the complexities for solving Sys, R_1, R_2, and R_3 with "general methods"

$$Compl(Sys) \approx Compl(R_1) \approx Compl(R_2) \ll Compl(R_3).$$

Semi-regular sequences

Solving R_3-systems has to do with the *semi-regular sequences* introduced by M.Bardet *et.al.* Complexity estimates for the $\mathrm{F5}$ algorithm are available. These results are not applicable for our situation, since the quadratic homogeneous part in our systems has positive dimension, whereas the results there are valid only for zero-dimensional case.

Further research

Further research

The possible directions of research:

- Complexity analysis of solving, e.g. via the analysis of R_2 systems.
- In the complexity analysis take into account a special form of the Macaulay matrix.
- Algorithmic questions connected with the existence of Generalized Newton identities for arbitrary linear codes.
- Decoding algorithms of polynomial complexity for some classes of codes that follow the idea of cyclic codes.