ON MULTIVARIATE INTERPOLATION DECODING OF FOLDED REED-SOLOMON CODES

Peter V. Trifonov
Saint-Petersburg State Polytechnic University, Russia

Outline

- Folded Reed-Solomon codes
- A multivariate generalization of the Wu list decoding algorithm
- Numeric results

Folded Reed-Solomon Codes

- (n,k,n-k+1) Reed-Solomon code over field F is
  - a cyclic code with generator polynomial $(x-\gamma)(x-\gamma^2)\ldots(x-\gamma^{n-k})$
  - a set of vectors $(f(\gamma^0), f(\gamma^1),\ldots,f(\gamma^{n-1}))$, $\gamma \in F$, $f(x) \in F[x]$ deg $f(x) < k$
- m-folded Reed-Solomon code
  - Consider a block of m adjacent symbols as a single one

$$([f(\gamma^0), f(\gamma^1),\ldots, f(\gamma^{m-1})],\ldots,[f(\gamma^{im}), f(\gamma^{im+1}),\ldots, f(\gamma^{im+m-1})],\ldots)$$

  - A non-linear code of length N=n/m over F^m
  - Application areas
    - Systems with burst noise
    - Concatenated codes
  - Decoding is easier: just find erroneous blocks, no need to look for a particular symbol within them

List Decoding of Folded RS Codes: the Guruswami-Rudra Algorithm

- Problem statement
  - Given: ($[y_{00}, \ldots, y_{0,m-1}], \ldots, [y_{N-1,0}, \ldots, y_{N-1,m-1}]$) - vector to be decoded
  - Goal: find all f(x):
    - deg f(x)<k
    - $f(\gamma^{im+j})=y_{ij}$, j=0..m-1, at least for τ distinct i
- Guruswami-Rudra algorithm
  - A multivariate generalization of the Guruswami-Sudan algorithm
  - Construct a polynomial $W(X,Y_1,\ldots,Y_s)$: $W(\gamma^{im+j},y_{i,j},y_{i,j+1},\ldots,y_{i,j+s-1})=0^r$, j=0..m-s, i=0..N-1
    - $n_0=N(m-s+1)$ interpolation points
    - If f(x): $f(\gamma^{im+j+l})=y_{i,j+l}$, l=0..s-1 ➔ $(x-\gamma^{im+j})^r | W(x,f(x),f(\gamma x),\ldots,f(\gamma^{s-1}x))$

$$\left(\prod_{i:\text{not error}} \prod_{j=0}^{m-s} (x-\gamma^{im+j})^r \right) \mid \underbrace{W(x, f(x), \ldots, f(\gamma^{s-1}x))}_{h(x)}$$

    - deg h(x) ≤ $\text{wdeg}_{(1,k-1,\ldots,k-1)}W(X,Y_1,\ldots,Y_s)$
    - $\text{wdeg}_{(1,k-1,\ldots,k-1)}W(X,Y_1,\ldots,Y_s)<\tau(m-s+1)r$ ➔ $W(x,f(x),f(\gamma x),\ldots,f(\gamma^{s-1}x))=0$
  - Find all f(x): $W(x,f(x),f(\gamma x),\ldots,f(\gamma^{s-1}x))=0$

List Decoding of Folded RS Codes: the Guruswami-Rudra Algorithm -2

- $W(X,Y_1,\ldots,Y_s)=\sum_{j_1=0}^{\rho}\cdots\sum_{j_s=0}^{\rho}w_{j_1,\ldots,j_s}(X)Y_1^{j_1}\cdots Y_s^{j_s}$: $\mathrm{wdeg}_{(1,k-1,\ldots,k-1)}W(X,Y_1,\ldots,Y_s)<D$ exists if the number of interpolation constraints is less than the number of coefficients

$$V(\rho,D,s,k)=\sum_{j=0}^{\rho}(D-j(k-1))\binom{j+s-1}{s-1}=$$

$$\left(D-(k-1)\rho\frac{s}{s+1}\right)\binom{\rho+s}{s}>n\frac{m-s+1}{m}\binom{r+s}{s+1}.$$

$$\rho=\left\lfloor\frac{D-1}{k-1}\right\rfloor.$$

- The fraction of correctable errors $\theta=(N-\tau)/N$:

$$\theta<1-\sqrt[s+1]{\left(\frac{mR}{m-s+1}\right)^s\prod_{j=1}^{s}\left(1+\frac{j}{r}\right)}$$

  - N=n/m, R=k/n, s≤m
- For large m and r: $\theta<1-R^{\frac{s}{s+1}}$
- For small m the $\frac{m}{m-s+1}$ factor severely degrades the error correction capability

Wu List Decoding Algorithm for RS Codes

6

Analytic Continuation of the Berlekamp-Massey Algorithm

- Let r(x)=c(x)+e(x)
- Syndrome of the received sequence: $S_i=r(\gamma^i)$, i=1..n-k
- BM algorithm constructs the shortest LFSR:

$$\Lambda_1 S_{i+w-1}+\Lambda_2 S_{i+w-2}+\ldots+\Lambda_w S_i=-S_{i+w}, i=0..w-$$

- If the number of errors w≤(n-k)/2, th… connection polynomial Λ(x) is th… locator polynomial

$$\Lambda(x)=\prod_{l=1}^{w}(1-X_l x)=\sum^{w}$$

- Otherwise, additio… …re needed to find the true ELP… …quire unknown syndrome component… $S_{n-k+1}, S_{n-k+2}, \ldots,$

$$\Lambda^*(x)=a(x)\Lambda(x)+b(x)xB(x)$$

Decoding Based on Groebner bases

- $r_i=f(\gamma^i)+e_i$, i=0..n-1
- Find f(…
  - …e i-th symbol is not corrupted
  - … if the i-th symbol is corrupted
  - …=σ(x)y-σ(x)f(x)
  - $Q(\gamma^i,r_i)=0$
- All such Q(x,y) are in the module $M_{11}=[\phi(x),y-T(x)] \subset F[x]\times F[x]$

$$\phi(x)=\prod_{i=0}^{n-1}(x-\gamma^i) \qquad T(\gamma^i)=r_i$$

- Let $q_{00}(x)+yq_{10}(x)$, $q_{01}(x)+yq_{11}(x)$ be a Groebner basis of this module
  - $Q(x,y)=a(x)(q_{00}(x)+yq_{10}(x))+b(x)(q_{01}(x)+yq_{11}(x))$

$$\sigma(x)=a(x)q_{10}(x)+b(x)q_{11}(x)$$

Smaller complexity compared to the Guruswami-Sudan algorithm

The number of roots of $\Lambda^*(x)$ and σ(x) must be equal to their degree and number of errors

A Multivariate Generalization of the Wu Algorithm

- Folded RS codes: the error polynomial $\sigma(x)$ must have t blocks of roots γ^{mi+j}, j=0..m-1
- Given: $q_{10}(x)$,$q_{11}(x)$
- Find a(x), b(x):$\sigma(\gamma^{mi+j}) = a(\gamma^{mi+j})q_{10}(\gamma^{mi+j}) + b(\gamma^{mi+j})q_{11}(\gamma^{mi+j}) = 0, j = 0..m-1$
  - for t distinct i
  - deg $\sigma(x) \le tm$
  - deg a(x) $\le d_1 = tm+k-1-\deg q_{00}(x)$;
  - deg b(x) $\le d_2 = tm-\deg q_{11}(x)$
- An observation:

$$\forall \theta \in F : a(\gamma^{mi+j})\theta q_{10}(\gamma^{mi+j}) + b(\gamma^{mi+j})\theta q_{11}(\gamma^{mi+j}) = 0, j = 0..m-1$$

A Multivariate Generalization of the Wu Algorithm - 2

- Construct a partially homogenized polynomial

$$W(X, Y_1, Z_1, \ldots, Y_s, Z_s) = \sum_{j_1=0}^{\rho} \cdots \sum_{j_s=0}^{\rho} w_{j_1,\ldots,j_s}(X) Y_1^{j_1} Z_1^{\rho - j_1} \cdots Y_s^{j_s} Z_s^{\rho - j_s},$$

such that

$$\forall \theta \in F, l = 0..m-s, i = 0..n/m-1:$$

$$W(\gamma^{im+l}, \theta q_{11}(\gamma^{im+l}), -\theta q_{10}(\gamma^{im+l}), \ldots, \theta q_{11}(\gamma^{im+l+s-1}), -\theta q_{10}(\gamma^{im+l+s-1})) = 0^r$$

$$W(x_0, y_1, z_1, \ldots, y_s, z_s) = 0^r \Leftrightarrow \left. \frac{\partial^{i_0+i_1+\ldots+i_s} \sum_{j_1=0}^{\rho} \ldots \sum_{j_s=0}^{\rho} w_{[j_1],\ldots,[j_s]}(x) u_1^{j_1} \ldots u_s^{j_s}}{\partial x^{i_0} \partial u_1^{i_1} \ldots \partial u_s^{i_s}} \right|_{x=x_0, u_1=\hat{u}_1, \ldots, u_s=\hat{u}_s} = 0$$

$i_0+i_1+\ldots+i_s<r$

- $z_i \neq 0$: $[j_i] = j_i; \hat{u}_i = \frac{y_i}{z_i}$
- $z_i = 0$: $[j_i] = \rho - j_i; \hat{u}_i = \frac{z_i}{y_i}$

Hasse derivatives must be used

A Multivariate Generalization of the Wu Algorithm - 3

$$\forall \theta \in F, j = 0..m-s, i = 0..n/m-1:$$

$$W(\gamma^{im+j}, \theta q_{11}(\gamma^{im+j}), -\theta q_{10}(\gamma^{im+j}), \ldots, \theta q_{11}(\gamma^{im+j+s-1}), -\theta q_{10}(\gamma^{im+j+s-1})) = 0^r$$

- For each l, s.t. a(x), b(x): $q_{10}(\gamma^{im+l+j})a(^{im+l+j})+q_{11}(\gamma^{im+l+j})b(^{im+l+j})=0$, $0 \le l \le s-1$

$$(X-\gamma^{im+j})^r \mid W(X, a(X), b(X), a(\gamma X), b(\gamma X), \ldots, a(\gamma^{s-1}X), b(\gamma^{s-1}X))$$

$$\left(\prod_{\substack{i: \\ i\text{-th symbol} \\ \text{is corrupted}}} \prod_{j=0}^{m-s} (X-\gamma^{im+j})^r \right) \mid W(X, a(X), b(X), a(\gamma X), b(\gamma X), \ldots, a(\gamma^{s-1}X), b(\gamma^{s-1}X))$$

- If $wdeg_{(1,d_1,d_2,\ldots,d_1,d_2)} W(X,Y_1,Z_1,\ldots,Y_s,Z_s) < D=t(m-s+1)r$, then $W(X,a(X),b(X),a(\gamma X),b(\gamma X),\ldots,a(\gamma^{s-1}X),b(\gamma^{s-1}X))=0$
  - deg $a(X) \le d_1$,
  - deg $b(X) \le d_2$

Selecting the Parameters

- The coefficients of $W(X,Y_1,Z_1,\ldots,Y_s,Z_s)$ are given by a system of linear equations:

$$W(x_0,y_1,z_1,\ldots,y_s,z_s)=0^r \Leftrightarrow \left.\frac{\partial^{i_0+i_1+\ldots+i_s}\sum_{j_1=0}^{\rho}\ldots\sum_{j_s=0}^{\rho}w_{[j_1],\ldots,[j_s]}(x)u_1^{j_1}\ldots u_s^{j_s}}{\partial x^{i_0}\partial u_1^{i_1}\ldots\partial u_s^{i_s}}\right|_{x=x_0,u_1=\hat{u}_1,\ldots,u_S=\hat{u}_S}=0$$

$i_0+i_1+\ldots+i_s<r$

- The number of equations must exceed the number of terms V in the polynomial

$$W(X,Y_1,Z_1,\ldots,Y_s,Z_s)=\sum_{j_1=0}^{\rho}\cdots\sum_{j_s=0}^{\rho}w_{j_1,\ldots,j_s}(X)Y_1^{j_1}Z_1^{\rho-j_1}\cdots Y_s^{j_s}Z_s^{\rho-j_s}$$

  - Number of terms with $(1,d_1,d_2,\ldots,d_1,d_2)$-weighted degree less than $D=t(m-s+1)r$

$$V=\sum_{j_1=0}^{\rho}\cdots\sum_{j_s=0}^{\rho}\left(D-\sum_{l=1}^{s}(d_1j_i+d_2(\rho-j_i))\right)=\left(D-\frac{d_1+d_2}{2}\rho s\right)(\rho+1)^s=\left(t(m-s+1)r-\frac{2tm+k-1-n}{2}\rho s\right)(\rho+1)^s$$

  - $d_1 = tm+k-1-\deg q_{00}(x), d_2=tm-\deg q_{11}(x)$, $\deg q_{00}(x)+\deg q_{11}(x)=n$

$$\left(r(m-s+1)t-\frac{2tm+k-1-n}{2}\rho s\right)(\rho+1)^s>\frac{n}{m}\frac{m-s+1}{(s+1)!}\prod_{j=0}^{s}(r+j)$$

The Error Correction Capability

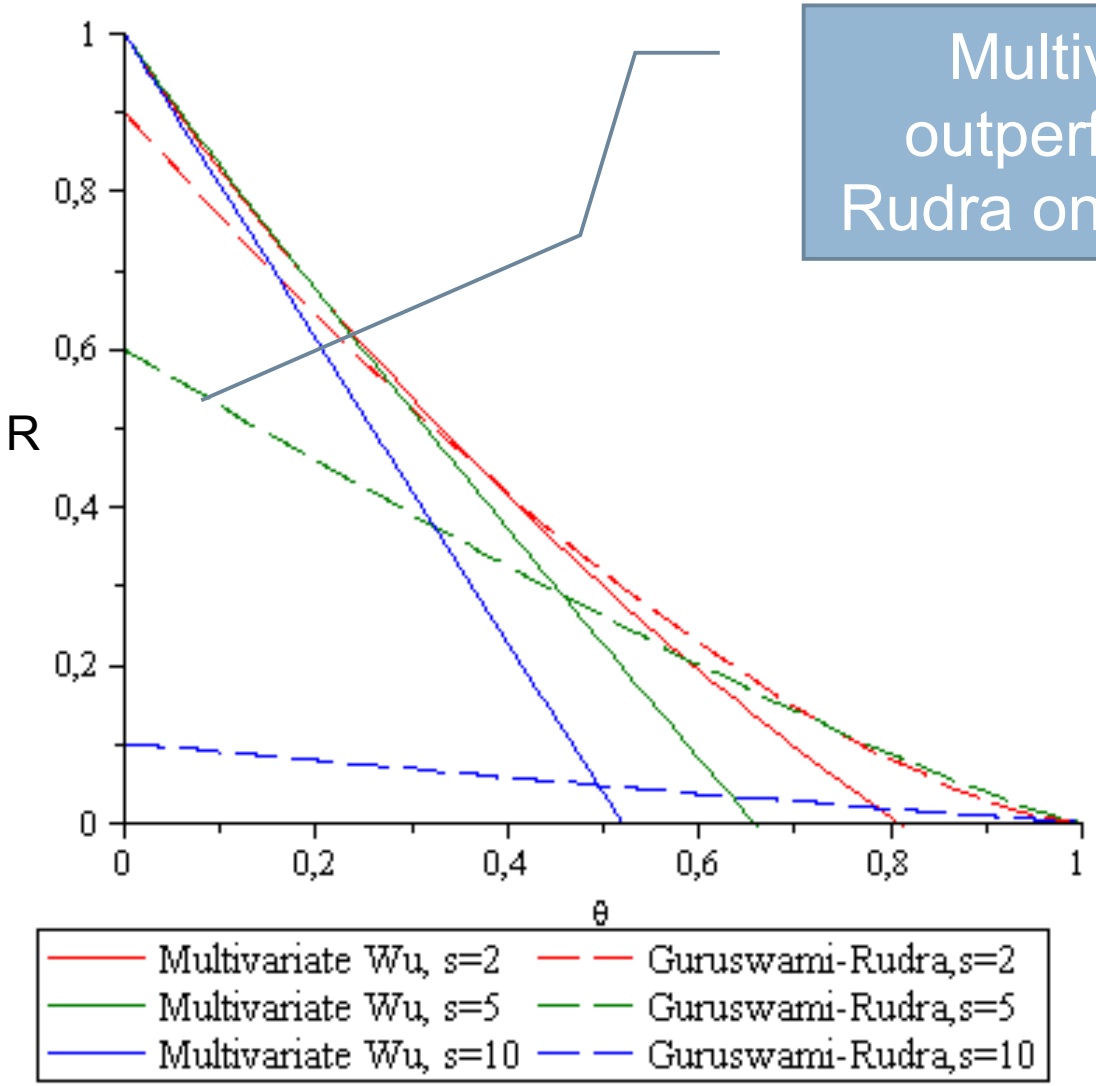
$$R < 1 - 2\theta + 2\frac{m-s+1}{m}\left(\frac{(s+1)!}{(s+1)^{s+1}}\theta^{s+1}\right)^{\frac{1}{s}}$$

- Root multiplicity r → ∞
- R=(k-1)/n
- θ=tm/n – the fraction of correctable errors
- Always better than the classical decoc (R<1-2θ)
- The Guruswami-Rudra algorithm

$$R < \frac{m-s+1}{m}(1-\theta)^{\frac{s+1}{s}}$$

The same for s=1

Comparison of Algorithms (m=10)



Multivariate Wu algorithm outperforms the Guruswami-Rudra one in the high-rate region

$$R_{Wu} < 1 - 2\theta + 2\frac{m-s+1}{m}\left(\frac{(s+1)!}{(s+1)^{s+1}}\theta^{s+1}\right)^{\frac{1}{s}}$$

$$R_{GR} < \frac{m-s+1}{m}(1-\theta)^{\frac{s+1}{s}}$$

Comparison of algorithms(s=2,n=255)

θ	t	m	Guruswami-Rudra				Multivariate Wu			
			R_{opt}	k	r	ρ	R_{opt}	k	r	ρ
0.1	25	1	0.81	208	46	13	0.81	208	5	51
	8	3	0.57	147	19	20	0.82	212	7	18
	5	5	0.68	175	18	19	0.82	211	6	15
0.3	76	1	0.49	125	23	33	0.49	125	9	32
	25	3	0.39	100	12	14	0.50	131	33	43
	15	5	0.47	120	15	18	0.52	136	33	43
0.5	127	1	0.25	64	26	52	0.25	64	25	51
	42	3	0.24	61	22	31	0.22	59	42	42
	25	5	0.28	73	14	20	0.27	72	3421	3455
0.8	204	1	0.04	11	41	209	0.04	11	165	206
	68	3	0.059	16	61	138	-0.15	-	-	-
	40	5	0.071	19	6	14	-0.06	-	-	-

Conclusions

- Multivariate Wu algorithm corrects higher fraction of errors than the Guruswami-Rudra algorithm in the high rate region
- The performance of both algorithms could be improved if more interpolation points could be obtained (eliminate (m-s+1)/m factor)