List-decoding Reed-Solomon codes: re-encoding techniques and Wu algorithm via simultaneous polynomial approximations

Vincent NEIGER[§,†,‡]

Claude-Pierre JEANNEROD[§] Éric SCHOST[†] Gilles VILLARD[§]

[§]AriC, LIP, École Normale Supérieure de Lyon, France

[†]ORCCA, Computer Science Department, Western University, London, ON, Canada

[‡]Supported by the international mobility grant *Explo'ra doc* from *Région Rhône-Alpes*

Journées nationales de calcul formel
CIRM, Luminy, France, November 5, 2014

Outline

1. Decoding of Reed-Solomon codes via polynomial approximations
2. Re-encoding technique via polynomial approximations
3. Wu reduction via polynomial approximations

Outline

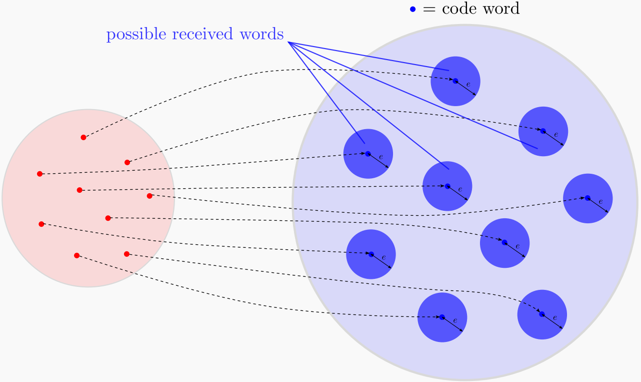
1. Decoding of Reed-Solomon codes via polynomial approximations
2. Re-encoding technique via polynomial approximations
3. Wu reduction via polynomial approximations

Reed-Solomon codes

At most $e = n - t$ errors during transmission of a code word

$$w = w_0 + \cdots + w_k X^k \xrightarrow{\text{encoding}} (w(x_1), \ldots, w(x_n)) \xrightarrow{\text{noise}} y = (y_1, \ldots, y_n)$$

i.e. $\#\{i \mid w(x_i) \neq y_i\} \leqslant e$ or $\#\{i \mid w(x_i) = y_i\} \geqslant t$



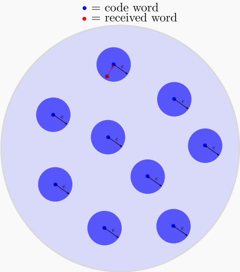
Decoding of Reed-Solomon codes

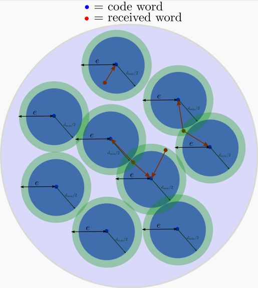
Polynomial Reconstruction

Input: $x_1, \ldots, x_n$ the n distinct evaluation points in $\mathbb{K}$
k the degree bound, $e = n - t$ the error-correction radius
$(y_1, \ldots, y_n)$ the received word in $\mathbb{K}^n$

Output: All polynomials w in $\mathbb{K}[X]$ such that

$$\deg w \leqslant k \qquad \text{and} \qquad \#\{i \mid w(x_i) = y_i\} \geqslant t$$





Key equations & Unique decoding

Master, Interpolation and error-locator polynomials

$$G(X) = \prod_{1 \leqslant i \leqslant n}(X - x_i), \qquad R(x_i) = y_i, \qquad \Lambda(X) = \prod_{i \mid \text{error}}(X - x_i)$$

Key equations: for every i, $\Lambda(x_i)R(x_i) = \Lambda(x_i)w(x_i)$

Modular key equation

$$\Lambda R = \Lambda w \mod G$$

where $\deg(\Lambda) \leqslant e$, $\deg(\Lambda w) \leqslant e + k$, Λ monic.

Unique decoding:
$e + k < n - e \Leftrightarrow e < \frac{n-k}{2} \Rightarrow$ unique rational solution $\frac{\Lambda w}{\Lambda} = w$
computed in $\tilde{\mathcal{O}}(n)$ using e.g. the Extended Euclidean algorithm
[Modern Computer Algebra, von zur Gathen - Gerhard, 2013]

List-decoding: Guruswami-Sudan algorithm

If $e < \frac{n-k}{2}$, unique decoding. If $e < n - \sqrt{kn}$, polynomial-time decoding.
Recall:

$$\deg w \leqslant k \qquad \text{and} \qquad \#\{i \mid w(x_i) = y_i\} \geqslant t$$

[Guruswami - Sudan, 1999]

- Interpolation step
  compute a polynomial $Q(X, Y)$ such that:
  - $Q(X, w)$ has many roots
  - $Q(X, w)$ has small degree

  $\longrightarrow$ w solution $\Rightarrow Q(X, w) = 0$
- Root-finding step
  find all Y-roots of $Q(X, Y)$, keep those that are solutions

Here we focus on the Interpolation step.

The interpolation step

Interpolation With Multiplicities

Input:

number of points n, degree weight k, weighted-degree bound b=mt
points $\{(x_i, y_i)\}_{1 \leqslant i \leqslant n}$ in $\mathbb{K}^2$ (x_i's distinct)
list-size ℓ, multiplicity m $\quad (m \leqslant \ell)$

Output:

a nonzero polynomial Q in $\mathbb{K}[X, Y]$ such that

(i)	$\deg_Y Q \leqslant \ell$,	(list-size condition)
(ii)	$\deg_X Q(X, X^k Y) < b$,	(weighted-degree condition)
(iii)	$\forall i$, $Q(x_i, y_i) = 0$ with multiplicity m	(vanishing condition)

Guruswami-Sudan: $t^2 > kn \Rightarrow$ solution exists for some well-chosen m, ℓ
$\longrightarrow$ linear system, compute a solution in polynomial time

Simultaneous polynomial approximations

[Roth - Ruckenstein, 2000] [Zeh - Gentner - Augot, 2011]

vanishing condition $\Leftrightarrow$ system of modular equations:

write $Q(X,Y) = Q_0(X) + Q_1(X)Y + \cdots + Q_\ell(X)Y^\ell$

for $i \in \{1,\ldots,n\}$, $Q(x_i,y_i) = 0$ with multiplicity m

$$\iff \left\{ \begin{array}{llll} Q_0 + \cdots + Q_{m-1}R^{m-1} + \cdots + Q_\ell R^\ell & = 0 & \bmod & G^m \\ \quad Q_1 + \cdots + Q_{m-1} m R^{m-2} + \cdots + Q_\ell \ell R^{\ell-1} & = 0 & \bmod & G^{m-1} \\ \qquad \ddots \quad \vdots \qquad\qquad \vdots & = 0 & \bmod & G^{\cdots} \\ \qquad\qquad Q_{m-1} + \cdots + Q_\ell \binom{\ell}{m-1} R^{\ell-m+1} & = 0 & \bmod & G \end{array} \right.$$

where $G = \prod_{1 \leqslant i \leqslant n}(X - x_i)$ and $\forall i, R(x_i) = y_i$.

Dimensions of linearized problem:

$M = \frac{1}{2}m(m+1)n$ equations, $\quad N = \sum_{0 \leqslant j \leqslant \ell}(b - jk)$ unknowns

Algorithms based on linearization

Strategy:

- use degree bounds to linearize the problem

$$\left[Q_0^{(0)} \cdots Q_0^{(b-1)} \mid Q_1^{(0)} \cdots Q_1^{(b-k-1)} \mid \cdots \mid Q_\ell^{(0)} \cdots Q_\ell^{(b-\ell k-1)} \right]$$

- vanishing condition $\Leftrightarrow$ solution to an under-determined linear system

[Guruswami - Sudan, 1999]
Structure "not used", cost $\mathcal{O}((m^2 n)^\omega)$ $(\omega =$ exponent of mat. mult.)

[Roth - Ruckenstein, 2000] [Zeh - Gentner - Augot, 2011]
Mosaic-Hankel system, cost $\mathcal{O}(\ell m^4 n^2)$ using [Feng - Tzeng, 1991]

[Chowdhury - Jeannerod - Neiger - Schost - Villard, 2014]
Mosaic-Hankel system, cost $\tilde{\mathcal{O}}(\ell^{\omega-1} m^2 n)$
using [Bostan - Jeannerod - Schost, 2007]

Algorithms based on reduced lattice bases

Based on polynomial lattice reduction
[Alekhnovich, 2002] [Reinhard, 2003] [Beelen - Brander, 2010] [Bernstein, 2011] [Cohn - Heninger, 2011]

- Compute a known basis of approximants
- Use lattice reduction to find a small-degree approximant

Cost $\tilde{\mathcal{O}}(\ell^\omega mn)$ using [Giorgi - Jeannerod - Villard, 2003] (probabilistic) or [Gupta - Sarkar - Storjohann - Valeriote, 2012]

Based on order basis computation

- Mirror all polynomials $\longrightarrow$ simultaneous Hermite-Padé equations
- Compute an order basis of the resulting matrix of power series

Cost $\tilde{\mathcal{O}}(\ell^{\omega-1} m^2 n)$ using [Zhou - Labahn, 2012]

Outline

When some y_i's are zero (case $m = 1$)

Recall $Q(x_i, y_i) = Q_0(x_i) + Q_1(x_i)y_i + \cdots + Q_\ell y_i^\ell$

Assume $y_1 = y_2 = \cdots = y_{i_0} = 0$, then

$$\text{for } i \leqslant i_0, \quad Q(x_i, y_i) = 0 \;\; \Leftrightarrow \;\; Q_0(x_i) = 0$$

Thus

$$(\text{for every } i \leqslant i_0, \quad Q(x_i, y_i) = 0) \;\; \Leftrightarrow \;\; Q_0 = G_0 \widehat{Q}_0$$

for some $\widehat{Q}_0$ of degree $< b - i_0$, where $G_0 \; = \; \prod_{1 \leqslant i \leqslant i_0}(X - x_i)$

$\longrightarrow$ Equations for points $i = 1, \ldots, i_0$ are pre-solved

Then remains an easier approximation problem

$$\widehat{Q}_0 + Q_1 R / G_0 + \cdots + Q_\ell R^\ell / G_0 \;\; = \;\; 0 \quad \text{mod } (G / G_0)$$

Smaller dimensions: $M - i_0$ equations, $N - i_0$ unknowns

Interpolation step with $y_1 = \cdots = y_{i_0} = 0$

Vanishing condition: $Q(x_i, y_i) = 0$ with multiplicity m for $i = 1, \ldots, n$

$$\Leftrightarrow \begin{cases} Q_0 + \cdots + Q_{m-1}R^{m-1} + \cdots + Q_\ell R^\ell & = 0 \mod G^m \\ Q_1 + \cdots + Q_{m-1} m R^{m-2} + \cdots + Q_\ell \ell R^{\ell-1} & = 0 \mod G^{m-1} \\ \ddots \quad \vdots \quad \vdots & = 0 \mod G^{\cdots} \\ Q_{m-1} + \cdots + Q_\ell \binom{\ell}{m-1} R^{\ell-m+1} & = 0 \mod G \end{cases}$$

$Q(x_i, 0) = 0$ with multiplicity m for $i = 1, \ldots, i_0$

$$\Leftrightarrow \begin{cases} Q_{m-1} = G_0 \widehat{Q}_{m-1} & \text{with } \deg \widehat{Q}_{m-1} < b - (m-1)k - i_0 \\ Q_{m-2} = G_0^2 \widehat{Q}_{m-2} & \text{with } \deg \widehat{Q}_{m-2} < b - (m-2)k - 2i_0 \\ \vdots & \\ Q_0 = G_0^m \widehat{Q}_0 & \text{with } \deg \widehat{Q}_0 < b - mi_0 \end{cases}$$

Cost bounds when $y_1 = \cdots = y_{i_0} = 0$

$$Q(x_i, y_i) = 0 \text{ with multiplicity } m \quad \text{for every } i \in \{1, \ldots, n\}$$

$$\Leftrightarrow \begin{cases} Q_{m-1} = G_0 \widehat{Q}_{m-1},\ Q_{m-2} = G_0^2 \widehat{Q}_{m-2},\ \ldots,\ Q_0 = G_0^m \widehat{Q}_0 \\ \forall r < m, \sum_{r \leqslant j < m} \widehat{Q}_j \binom{j}{r} R^{j-r} / G_0^{j-r} \\ \qquad + \sum_{m \leqslant j \leqslant \ell} Q_j \binom{j}{r} R^{j-r} / G_0^{m-r} = 0 \mod (G/G_0)^{m-r} \end{cases}$$

Smaller dimensions: $\widehat{M} = M - \frac{1}{2} m(m+1) i_0$ and $\widehat{N} = N - \frac{1}{2} m(m+1) i_0$

$$\widehat{M} = \frac{1}{2} m(m+1)(n - i_0)$$

Cost bounds:

- Lattice reduction: $\tilde{\mathcal{O}}(\ell^{\omega} m(n - i_0))$
- Order basis / structured system: $\tilde{\mathcal{O}}(\ell^{\omega-1} m^2 (n - i_0))$

Re-encoding technique

[Koetter - Ma - Vardy, 2011]

Decoding: search for all w such that

$$\deg w \leqslant k \qquad \text{and} \qquad \#\{i \mid w(x_i) = y_i\} \geqslant t$$

Re-encoding technique: shift the received word by a code word

$$(y_1, \ldots, y_n) \xrightarrow{\text{shift}} (0, \ldots, 0, y_{k+2} - w_0(x_{k+2}), \ldots, y_n - w_0(x_n))$$

where $\deg w_0 \leqslant k$ and $w_0(x_i) = y_i$ for $1 \leqslant i \leqslant k+1$

- $\widehat{Q}(X, Y) \longleftarrow$ Interpolation step with $\hat{y}_i = y_i - w_0(x_i)$
  taking advantage of $\hat{y}_1 = \cdots = \hat{y}_{k+1} = 0$ $\qquad (i_0 = k+1)$
- Root-finding + filtering step on $\widehat{Q}$, obtaining $\{w^{(1)}, \ldots, w^{(\bar{\ell})}\}$
- Return $\{w^{(1)} + w_0, \ldots, w^{(\bar{\ell})} + w_0\}$

Cost bound: $\mathcal{O}\tilde{~}(\ell^{\omega-1} m^2 (n-k))$

Outline

Central idea

[Wu, 2008] [Trifonov - Lee, 2012] [Beelen - Høholdt - Nielsen - Wu, 2013]

Focus changes from correct locations to erroneous locations

In terms of Key Equations,

$$R = w \mod (G/\Lambda)$$
$$\downarrow$$
$$aB = bA \mod \Lambda$$

Problem changes from polynomial reconstruction to rational reconstruction

$$\deg w \leqslant k \quad \text{and} \quad \#\{i \mid w(x_i) = y_i\} \geqslant t$$
$$\downarrow$$
$$\deg a \leqslant \theta_1, \deg b \leqslant \theta_2, \gcd(a, b) = 1 \quad \text{and} \quad \#\{i \mid a(x_i)z_i' = b(x_i)z_i\} \geqslant e$$

(technical details are omitted, they would explain how to find θ_1, θ_2 and why $\deg \Lambda \leqslant e$ in the key equation has become $\#\{\cdots\} \geqslant e$)

The Interpolation step, revisited

Algo: Guruswami-Sudan via a minor modification of the interpolation step

Interpolation With Multiplicities allowing points at infinity

Input:

number of points n, degree weight θ_0, weighted-degree bound b
points $\{(x_i, z_i : z'_i)\}_{1 \leqslant i \leqslant n}$ in $\mathbb{K} \times (\mathbb{K} \cup \{\infty\})$ (x_i's distinct)
list-size ℓ, multiplicity m $\quad (m \leqslant \ell)$

Output: a nonzero polynomial Q in $\mathbb{K}[X, Y]$ such that

(*i*)	$\deg_Y Q \leqslant \ell$,	(list-size condition)
(*ii*)	$\deg_X Q(X, X^{\theta_0} Y) < b$,	(weighted-deg. condition)
(*iii*)	$\forall i,\ Q(x_i, z_i : z'_i) = 0$ with multiplicity m	(vanishing condition)

Where we have defined when $z_i : z'_i = \infty$,

$$Q(x_i, \infty) = 0 \text{ with multiplicity } m \quad \Leftrightarrow \quad \overline{Q}(x_i, 0) = 0 \text{ with multiplicity } m$$

and $\overline{Q} = Y^\ell Q(X, Y^{-1}) = Q_\ell + Q_{\ell-1} Y + \cdots + Q_1 Y^{\ell-1} + Q_0 Y^\ell$

Simultaneous polynomial approximations

Assume $z_i : z_i' = \infty$ for $i = 1, \ldots, n_\infty$ (with possibly $n_\infty = 0$)

Like in re-encoding technique,

$$Q(x_i, \infty) = 0 \text{ with multiplicity } m \text{ for } i = 1, \ldots, n_\infty$$
$$\Leftrightarrow \quad Q_{\ell-m+1} = G_\infty \widehat{Q}_{\ell-m+1},\ Q_{\ell-m+2} = G_\infty^2 \widehat{Q}_{\ell-m+2},\ \ldots,\ Q_\ell = G_\infty^m \widehat{Q}_\ell$$

where $G_\infty = \prod_{1 \leqslant i \leqslant n_\infty} (X - x_i)$,
with updated degree constraints for $Q_{\ell-m+1}, \ldots, Q_\ell$.

Equations for points $i = 1, \ldots, n_\infty$ are pre-solved,
remains an easier approximation problem without points at infinity

Points at infinity are not a complication but an advantage!

Note: can be combined with re-encoding on $|\theta_0| = |\theta_1 - \theta_2|$ points.
But we expect $\theta_1 \approx \theta_2 \ldots$

Cost bounds

Solving this problem of simultaneous approximations

- Lattice reduction: $\tilde{\mathcal{O}}(\ell^{\omega} m(n - n_{\infty} - |\theta_0|))$
- Order basis / structured system: $\tilde{\mathcal{O}}(\ell^{\omega-1} m^2(n - n_{\infty} - |\theta_0|))$

Recall we expect $n_{\infty} \approx 0$ and $\theta_0 \approx 0\ldots$
$\longrightarrow$ what advantage over original Guruswami-Sudan approach?

Smaller parameter m!

More precisely, $\ell_{\mathrm{Wu}} = \ell_{\mathrm{GS}} =: \ell$, but $m_{\mathrm{Wu}} = \ell - m_{\mathrm{GS}}$

For "well-chosen" parameters, $\ell \approx m_{\mathrm{GS}} t/k \quad \Rightarrow \quad m_{\mathrm{Wu}} \approx m_{\mathrm{GS}}(t/k - 1)$

Cost bounds:

- Lattice reduction: $\tilde{\mathcal{O}}(\ell^{\omega} m_{\mathrm{GS}}(t/k - 1)(n - n_{\infty} - |\theta_0|))$
- Order basis / struct. system: $\tilde{\mathcal{O}}(\ell^{\omega-1} m_{\mathrm{GS}}^2(t/k - 1)^2(n - n_{\infty} - |\theta_0|))$

Conclusion

List-decoding Reed-Solomon codes

$\downarrow$

Simultaneous polynomial approximations

Fast algorithms:

- lattice basis reduction
- solution of structured system
- order basis computation

Can benefit from cost-reducing techniques:

- Re-encoding
- Wu reduction to rational reconstruction

Other applications:

- Interpolation step of soft-decoding [Koetter - Vardy, 2003]