Class no 9:

Extending the Algorithm 1 for list decoding RS codes upto $1-2\sqrt{R}$ to $\rho = 1-\sqrt{2R}$ (Algorithm 2).

→ Essential idea: Define $Q(X,Y)$ more cleverly or intelligently

Observation:

Note that to prove (in Algo 1) that every $M(X)$ with $\deg(M(X)) \le k-1$ & $d_H(y, (M(\alpha_1), \ldots, M(\alpha_n))) \le e = \rho n$,

we used a (degree argument (+ no. of zeros)) on $R(X) \triangleq Q(X, M(X))$

(1e) We know that at least $n-e$ distinct zeros exist for $R(X)$.

& thus $R(x) = 0$ (poly) if $\boxed{\underset{\text{no. } f \text{ roots}}{\underline{n-e}} > \deg(R(x))}$

& If $R(x) = 0$ then $(Y - M(x)) \mid Q(x, Y)$

$\Rightarrow$ Correct decoding is true.

Note that $\deg(R(x)) = \deg(Q(x, M(x)))$

$$\boxed{= \deg_x(Q) + (k-1)\deg_Y(Q).}$$

$\Rightarrow n - e >$

Note that $\deg(R(X)) = \deg(Q(X, M(X)))$

Suppose $Q(X,Y) = \sum_{i,j=0} q_{i,j} X^i Y^j$. [only finite terms].

Then $Q(X, M(X)) = \sum_{i,j=0} q_{i,j} X^i (M(X))^j$

$$R(X) = \sum_{i,j=0} \tilde{q}_{i,j} X^{i+(k-1)j}$$

$\deg(R(X)) = \max\{(i+(k-1)j) : \forall i,j \; X^i Y^j$ has a non zero coeff in $Q(X,Y)\}$.

Now see that $\deg(R(X)) \leq \deg_X(Q) + (k-1)\deg_Y(Q)$

[this is with equality provided coeff of $X^{\deg_X(Q)} Y^{\deg_Y(Q)}$ is nonzero in $Q(X,Y)$]

For Algo 2, we will assume a different structure for $Q(X,Y)$ such that the $\deg(R(X)) = \max\{(i+(k-1)j) : X^i Y^j$ exists in $Q(X,Y)$ with nonzero coeffs$\}$
$= D$
is strictly smaller than no of roots of $R(X) = n-e$

$$\boxed{n-e > D} \Rightarrow \boxed{e < n-D}.$$

If D is small, then e can be large.

But if D is too small, then no of coefficients in $Q(X,Y)$ will also be 'too small' $\Rightarrow$ Step 1 (Interpolation step) cannot be executed as for Step 1 we need no of coeff in $Q(X,Y) > n$

no of constraints

constraints $\left[\dfrac{Q(\alpha_i, y_i) = 0}{i = 1, \dots, n} \right]$

Goal for Algo 2:

Define $Q(X,Y)$ so that

(1) $\deg(R(X)) = D$ is 'small enough'

so that $\dfrac{e}{n} < \dfrac{n-D}{n} = 1 - \sqrt{2R}$

(2) Also make sure that no of coeff of $Q(X,Y) > n$ (for Step 1).

Algo 2:

Step 1: (Interpolation): Define $Q(X,Y) \triangleq \sum_{\substack{i,j=0 \\ i+(k-1)j \leq D}} q_{i,j} X^i Y^j$.

Note:

If we define $Q(X,Y)$ like above,

then $\deg(R(X)) = \deg(Q(X, \underset{k-1}{M(X)})) \leq D$

(We will fix D according to requirement that ② condition in the $e < n - D$
$\|$
ρn
$\|$
$(1-\sqrt{2R})n$.)

(C1) Now we want to make sure that to run Step 1, no of coeffs in $Q(X,Y)$ has to $> n$ (no of constraints $\to$ $Q(\alpha_i, y_i) = 0, \forall i = 1 \ldots n$)

We have to pick D such that (C1) is true

& pick D & e such that (C2) is also true

To check C1, we first obtain the no of coeffs in $Q(X,Y)$. First note that as $i + (k-1)j \leq D$ & $i \geq 0, j \geq 0$,

then $j \leq \left\lfloor \frac{D}{k-1} \right\rfloor =: \ell$ (say)

So no of coefficients in $Q(X,Y)$

$$= \underbrace{\sum_{j=0}^{\ell} \sum_{i=0}^{D-j(k-1)}}_{\hookrightarrow \text{ no of } (i,j) \text{ pair allowed by defn of } Q.} 1 \rightarrow \text{no of coeff for any fixed } (i,j) \text{ pair}$$

$$= \sum_{j=0}^{\ell} \left(D - j(k-1) + 1\right) = (D+1)(\ell+1) - (k-1)\left(\sum_{j=0}^{\ell} j\right)$$

$$= (D+1)(\ell+1) - (k-1)\left(\frac{\ell(\ell+1)}{2}\right)$$

$$= \frac{(\ell+1)}{2}\left[2D+2 - \underbrace{(k-1)\ell}_{\leq D}\right] \geq \frac{(\ell+1)}{2}(D+2)$$

$$\ell = \left\lfloor \frac{D}{k-1} \right\rfloor \Rightarrow \left(\ell+1 > \frac{D}{k-1}\right)$$

$$\hookrightarrow \quad \geq \frac{D(D+2)}{2(k-1)} \rightarrow \text{atleast so many coeff in } Q.$$

We want this to be $> n$

Pick D so that

$$\Rightarrow \frac{D(D+2)}{2(k-1)} > n.$$

So we pick $D = \sqrt{2n(k-1)}$

Clearly $\frac{D^2}{2(k-1)} > n \Rightarrow \frac{D(D+2)}{2(k-1)} > n.$

This will ensure Step 1 finds a non-zero $Q(x,y)$.

In Step 2, we find all $\hat{M}(x)$ such that

(a) $\deg(\hat{M}(x)) \leq k-1$

(b) $(y - \hat{M}(x)) \mid Q(x,y)$

(c) $d_H(y, (M(\alpha_1), \ldots, M(\alpha_n))) \leq e$

To verify correct decoding we have to show $R(X) \triangleq Q(X, M(X))$

is zero poly for any $M(X)$ satisfying (a) & (c).

To do this we wanted

$$\deg(R(X)) < \text{no of distinct roots} = n - e$$

This is true as

$$n - e > \underset{\deg(R(X))}{D} = \sqrt{2n(k-1)}$$

$$\Rightarrow \quad e < n - \sqrt{2n(k-1)}$$

$$\Rightarrow \quad \frac{e}{n} < 1 - \sqrt{\frac{2(k-1)}{n}} = 1 - \sqrt{2R}.$$

Correct decoding is ensured upto radius

$$\Rightarrow \quad e \approx 1 - \sqrt{2R}.$$