Improved burst-error correction by joint decoding of interleaved RS codes

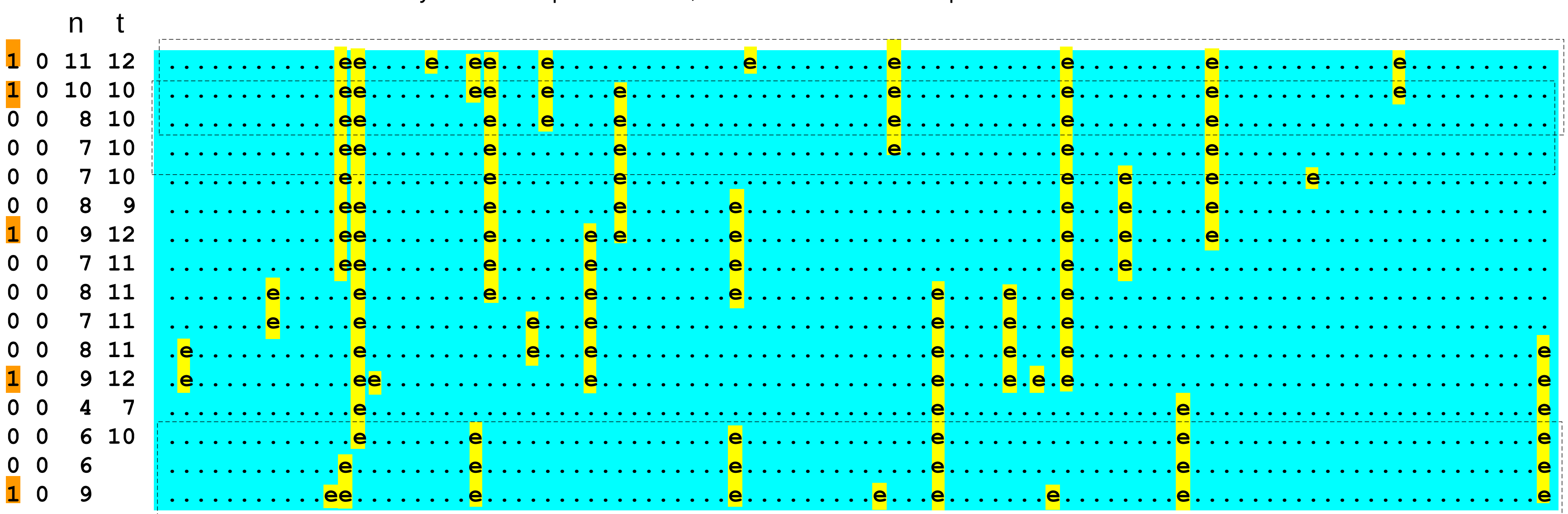
Short version, 27 July 2015

Gottfried Ungerboeck

Preview

D=16 block-interleaved codewords RS(N = 96, K = 80; T = 8) over GF(256), transmitted over **uncoded 16-QAM** channel with **AWGN & burst noise** (SNR_g = 22 dB, SNR_b = 0 dB, P_b=0.05, n_b=8)

n ... # of symbol errors per codeword, t ... # of column errors per subblock of M codewords



Joint decoding of sliding blocks of M = 3 consecutive codewords (t_{max} = 12): all codewords correctly decoded.

e e e e e RS symbol errors

Individual decoding of codewords (t_g = 8): 5 decoding failures

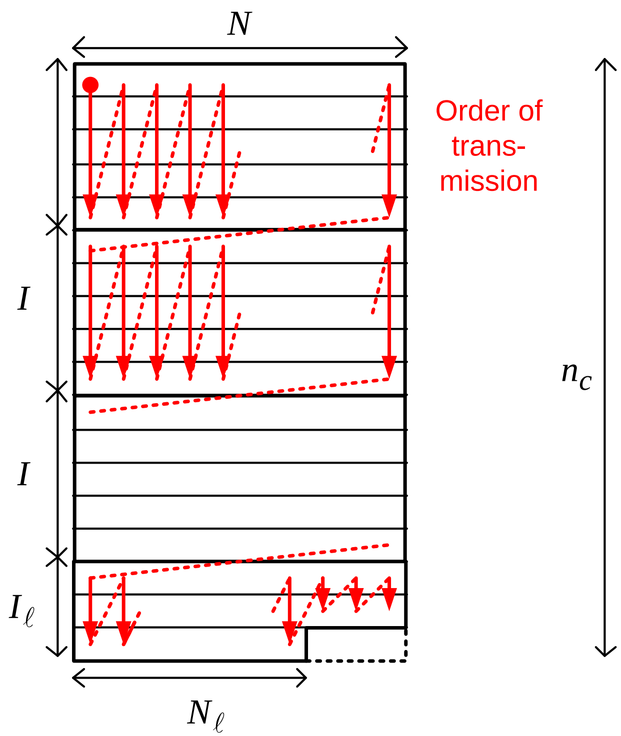
$$\boxed{t_g = \mathrm{floor}[(N-K)/2] = T \;\;;\;\; t_{\max} = \mathrm{floor}[M(N-K)/(M+1)]}$$

Contents

Block interleaving: e.g. in DOCSIS upstream transmission

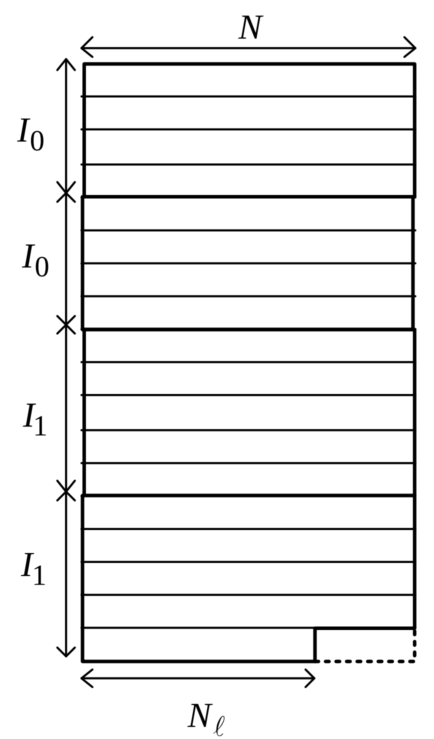
Fixed-configuration block interleaving

$D = I$ or I_ℓ; last block $1 \le I_\ell \le I$



Dynamically-configured block interleaving

$D = I_1$ or $I_0 = I_1 - 1$



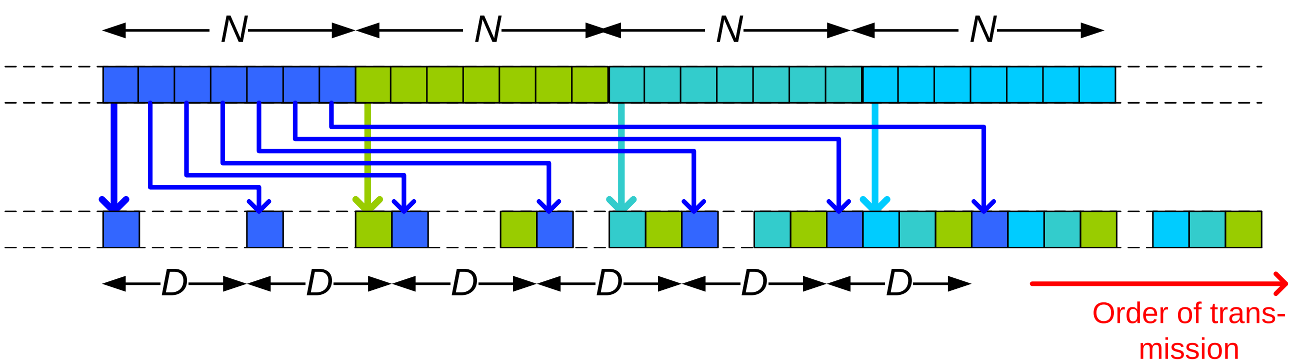
Convolutional interleaving: e.g. in early ADSL (G.992.1, 06/1999)

The ADSL **Transmit PMS-TC Function** generates "FEC Data Frames"

= RS(N,K;T) codewords over GF(256), where $N \leq 255$, odd.

Permitted interleaving depths are $D \in \{2, 4, 8 \cdots 64\}$.

Example: N = 7, D = 4 (collision-free reordering requires gdc(N,D)=1).



Convolutional interleaving: aligning burst errors in columns

For modest interleaving depth D < N, consecutively transmitted RS symbols can be aligned in columns to some extent by shifting received codewords cyclically … .

N = 15, D = 4; I_D = 4: matrix entries = temporal positions $t(i,j)$ of transmitted symbols

codewords not shifted

	j = 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
i=-7,s= 0:	-105	-101	-97	-93	-89	-85	-81	-77	-73	-69	-65	-61	-57	-53	-49
i=-6,s= 0:	-90	-86	-82	-78	-74	-70	-66	-62	-58	-54	-50	-46	-42	-38	-34
i=-5,s= 0:	-75	-71	-67	-63	-59	-55	-51	-47	-43	-39	-35	-31	-27	-23	-19
i=-4,s= 0:	-60	-56	-52	-48	-44	-40	-36	-32	-28	-24	-20	-16	-12	-8	-4
i=-3,s= 0:	-45	-41	-37	-33	-29	-25	-21	-17	-13	-9	-5	-1	3	7	11
i=-2,s= 0:	-30	-26	-22	-18	-14	-10	-6	-2	2	6	10	14	18	22	26
i=-1,s= 0:	-15	-11	-7	-3	1	5	9	13	17	21	25	29	33	37	41
i= 0,s= 0:	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56
i= 1,s= 0:	15	19	23	27	31	35	39	43	47	51	55	59	63	67	71
i= 2,s= 0:	30	34	38	42	46	50	54	58	62	66	70	74	78	82	86
i= 3,s= 0:	45	49	53	57	61	65	69	73	77	81	85	89	93	97	101
i= 4,s= 0:	60	64	68	72	76	80	84	88	92	96	100	104	108	112	116

codewords cyclically shifted to align burst errors in columns

i=-7,s= 2:	-53	-49	-105	-101	-97	-93	-89	-85	-81	-77	-73	-69	-65	-61	-57
i=-6,s= 6:	-54	-50	-46	-42	-38	-34	-90	-86	-82	-78	-74	-70	-66	-62	-58
i=-5,s=10:	-55	-51	-47	-43	-39	-35	-31	-27	-23	-19	-75	-71	-67	-63	-59
i=-4,s=14:	-56	-52	-48	-44	-40	-36	-32	-28	-24	-20	-16	-12	-8	-4	-60
i=-3,s= 3:	3	7	11	-45	-41	-37	-33	-29	-25	-21	-17	-13	-9	-5	-1
i=-2,s= 7:	2	6	10	14	18	22	26	-30	-26	-22	-18	-14	-10	-6	-2
i=-1,s=11:	1	5	9	13	17	21	25	29	33	37	41	-15	-11	-7	-3
i= 0,s= 0:	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56
i= 1,s= 4:	59	63	67	71	15	19	23	27	31	35	39	43	47	51	55
i= 2,s= 8:	58	62	66	70	74	78	82	86	30	34	38	42	46	50	54
i= 3,s=12:	57	61	65	69	73	77	81	85	89	93	97	101	45	49	53
i= 4,s= 1:	116	60	64	68	72	76	80	84	88	92	96	100	104	108	112

RS coding / decoding: selected topics

- Galois field and Fourier transform
- Information encoding into RS(N,K) codewords
- Syndrome-based decoding*
  - Calculation of syndrome values
  - Determining an error locator polynomial (Berlekamp-Massey or Euclidean algorithm)
  - Determining error locations (Chien search)
  - Determining error values (Forney or Horiguchi algorithm)
- Probabilities of decoding results

* syndromes are not required for interpolation-based decoding

Determining the error locations

Assume t symbol errors $e_{i_1}, e_{i_2}, \cdots e_{i_t}$ have occurred. From the observable *N-K* syndrome values $S_0, S_1, \cdots S_{N-K-1}$ find a ***t*-valid* Error-Locator Polynomial** $\Lambda(z)$

$$\lambda(x) = \sum_{i=0}^{N-1} \lambda_i x^i, \quad \lambda_{i_k} = 0, k = 1, \cdots t \quad \Rightarrow \quad \Lambda(z) = \prod_{k=1}^{t} \left(1 - \alpha^{i_k} z\right) = 1 + \Lambda_1 z + \ldots \Lambda_t z^t$$

[* having t distinct zeroes in GF(q)] such that

$$\sum_{i=0}^{N-1} e_i \lambda_i x^i = 0 \quad \Rightarrow \quad E(z)\Lambda(z) \bmod (z^N - 1) = 0 \quad \equiv \quad S(z)\Lambda(z) \bmod (z^N - 1) = 0 \ .$$

This defines N equations. Of these N equations the $N - K - t$ **Key Equations** contain only the observable syndrome values $S_0, S_1, \cdots S_{N-K-1}$:

$$\begin{array}{cccccc}
S_0 \Lambda_t & + S_1 \Lambda_{t-1} & \cdots & + S_{t-1} \Lambda_1 & + S_t & = 0 \\
S_1 \Lambda_t & + S_2 \Lambda_{t-1} & \cdots & + S_t \Lambda_1 & + S_{t+1} & = 0 \\
\vdots & & & & & \vdots \\
S_{N-K-t-1} \Lambda_t & + S_{N-K-t} \Lambda_{t-1} & \cdots & + S_{N-K-2} \Lambda_1 & + S_{N-K-1} & = 0
\end{array}$$

Necessary condition for existence of a unique solution for $\Lambda_1, \Lambda_2, \cdots \Lambda_t$: there are at least as many equations as unknowns, i.e., $N - K - t \geq t \rightarrow t \leq \lfloor (N-K)/2 \rfloor$. Hence correction of at most $t_g = \lfloor (N-K)/2 \rfloor$ symbol errors can be guaranteed.

Joint decoding of multiple RS codewords

- Find common error locator polynomial for a block of Interleaved Reed-Solomon (IRS) codewords:

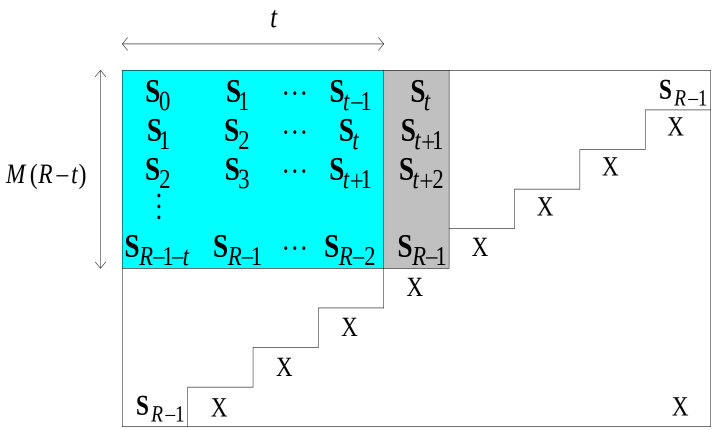
  (a) Number of equations, decoding capability, decoding failure probability

  (b) Multi-sequence Berlekamp-Massey algorithm

G. Schmidt, "Algebraic Decoding beyond half the minimum distance based on shift register synthesis", Ph.D Dissertation, University Ulm, Germany, 2007.

When *t* column errors occur: system of equations

Assume a block of M received RS codewords with t columns containing at least one symbol in error. Let the zeroes of the common error locator polynomial $\Lambda(z) = 1 + \Lambda_1 z + \Lambda_2 z^2 + \cdots \Lambda_t z^t$ coincide with the positions of the error-ed columns. The coefficients of $\Lambda(z)$ satisfy



$$\rightarrow \quad M \underbrace{\begin{bmatrix} \mathbf{S}_0 & \mathbf{S}_1 & \cdots & \mathbf{S}_{t-1} \\ \mathbf{S}_1 & \mathbf{S}_2 & \cdots & \mathbf{S}_t \\ \mathbf{S}_2 & \mathbf{S}_3 & \cdots & \mathbf{S}_{t+1} \\ \vdots & & & \\ \mathbf{S}_{R-1-t} & \mathbf{S}_{R-1} & \cdots & \mathbf{S}_{R-2} \end{bmatrix}}_{\mathbf{U}_{[M(R-t)\times t]}} \begin{bmatrix} \Lambda_t \\ \vdots \\ \Lambda_3 \\ \Lambda_2 \\ \Lambda_1 \end{bmatrix} = -\begin{bmatrix} \mathbf{S}_t \\ \mathbf{S}_{t+1} \\ \mathbf{S}_{t+2} \\ \vdots \\ \mathbf{S}_{R-1} \end{bmatrix}$$

Necessary condition for a unique solution: there are at least as many equations as coefficients to be determined, i.e., $M(R-t) \geq t \;\Rightarrow\; (M+1)t \leq MR \;\Rightarrow\; t \leq \lfloor MR/(M+1) \rfloor = t_{\max}$.

If $\operatorname{rank}(\mathbf{U}) < t$, multiple solutions can exist. One of them will be the correct $\Lambda(z)$. A given decoding algorithm will most likely not decide for the correct $\Lambda(z)$.

When t column errors occur: decoding capability

A decoder determines the correct codeword with probability $P_c(t)$, fails to decode with probability $P_f(t)$, and determines an incorrect codeword with probability $P_e(t)$:

$P_c(t)+P_f(t)+P_e(t)=1$. For IRS codes: $P_f(t) >> P_e(t)$ (as for individual decoding).

- If $t \le t_g = \lfloor R/2 \rfloor$: $\text{rank}\,(\mathbf{U}) = t$ guaranteed. Decoding always successful.
- If $t_g < t \le t_{\max} = \lfloor MR/(M+1) \rfloor$: $\text{rank}\,(\mathbf{U}) \le t$. Decoding successful except when with low probability $\text{rank}\,(\mathbf{U}) < t$. $\text{Prob}[\text{rank}(\mathbf{U}) < t]$ tightly upperbounds $P_f(t)$.
- $t > t_{\max}$: $\text{rank}\,(\mathbf{U}) < t$. Decoding always fails.

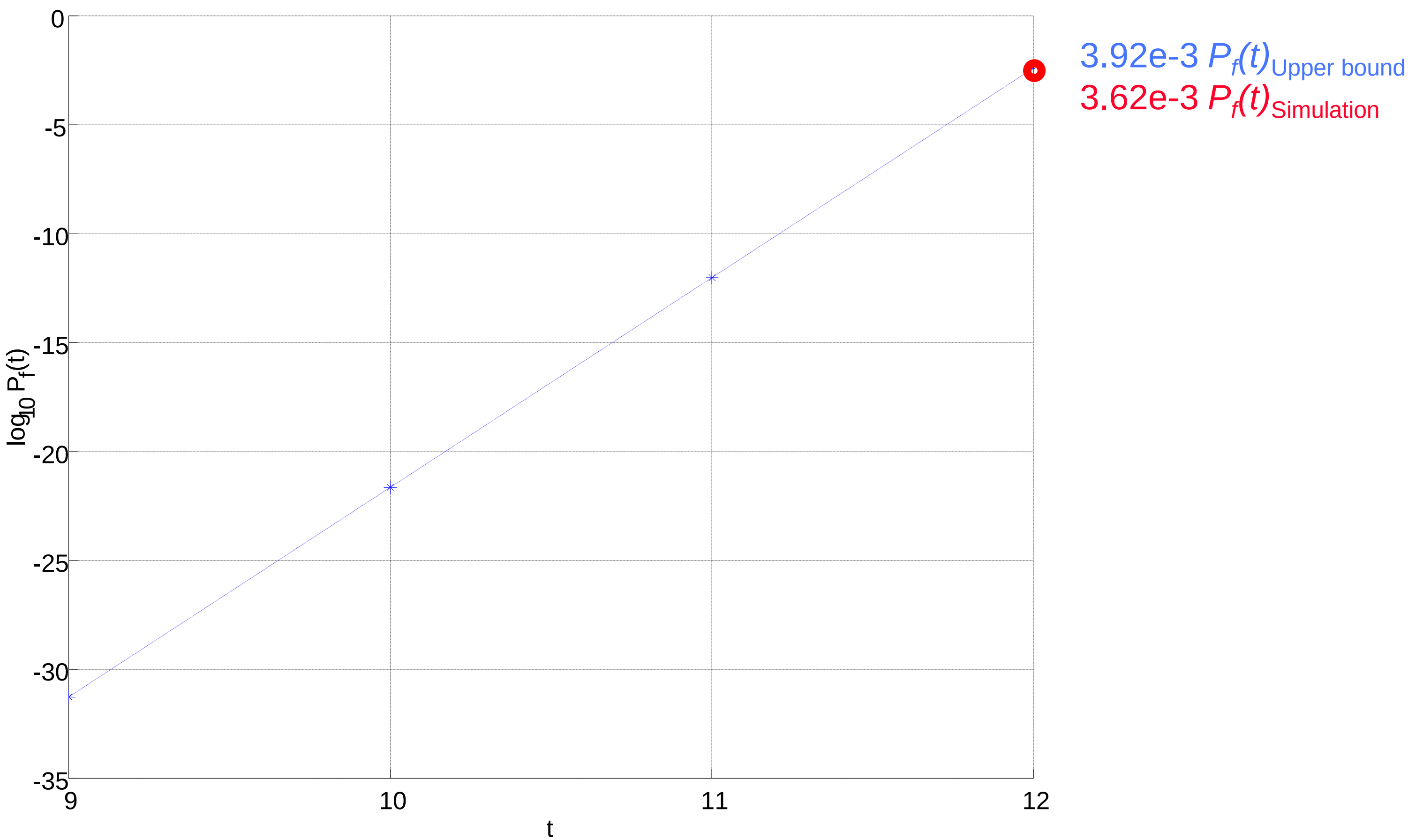
Example: RS(255,239)

Decoding codewords individually: $t \le t_g = (255-239)/2 = 8$ errors can be corrected.

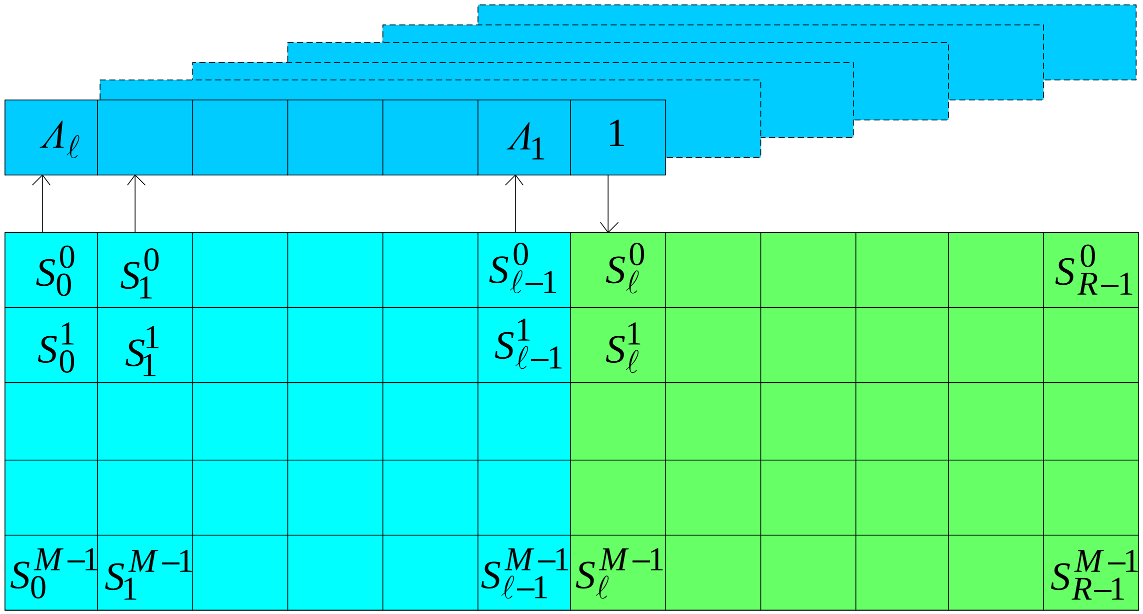
Decoding M = 3 interleaved codewords: errors in $t \le t_{\max} = 3(255-239)/4 = 12$ columns can be corrected *with high probability*.

When t column errors occur: decoding failure probability

M=3, RS(N=255,K=239;T=8) over GF(256): $t_g = 8$, $t_{\max} = 12$



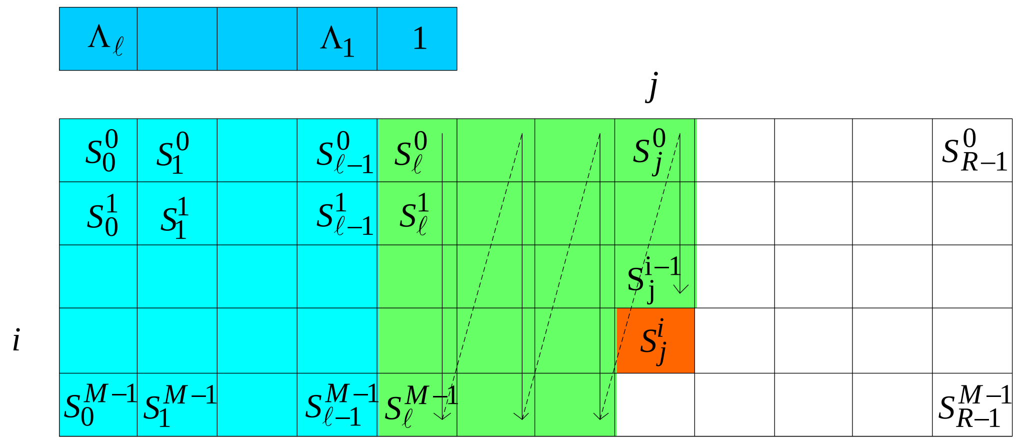
Multi-sequence shift register synthesis



Find the shortest polynomial $\Lambda(z) = 1 + \Lambda_1 z + \cdots \Lambda_\ell z^\ell$ which *generates* from initial syndromes $S_{0\cdots\ell-1}^i, 0 \le i \le M-1$, recursively the remaining syndromes $S_{\ell\cdots R-1}^i$:

$$\sum_{k=1}^{\ell} S_{j-k}^i \Lambda_k = -S_j^i \quad \text{for} \quad 0 \le i \le M-1, \;\; j = \ell, \ell+1, \cdots R-1.$$

Multi-sequence shift register synthesis



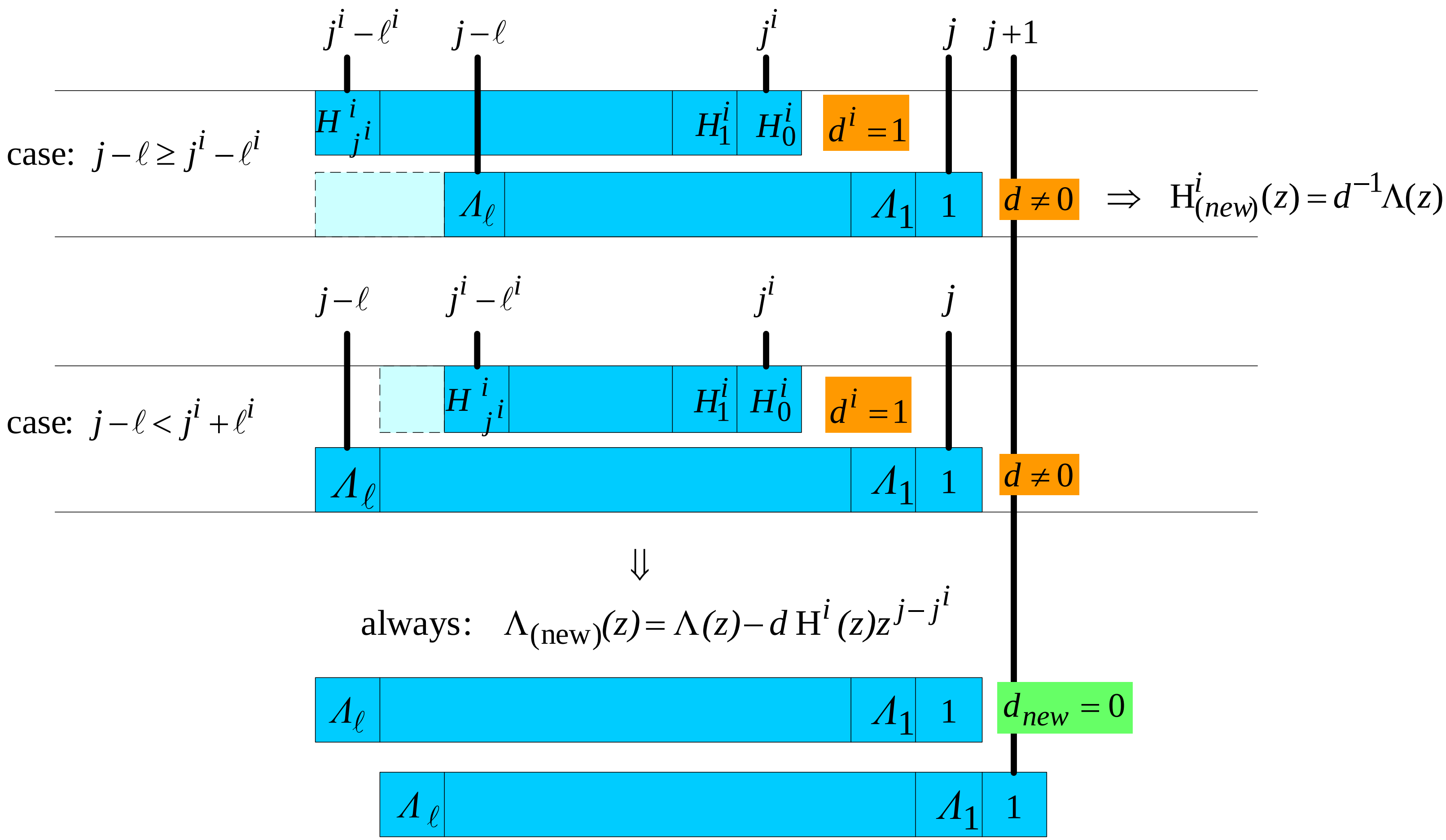
Assume processing has proceeded from (0,0) in column-row order up to (i,j). $\Lambda(z) = 1 + \Lambda_1 z + \cdots \Lambda_\ell z^\ell$ is the shortest polynomial that generates from $S^i_{0 \ldots \ell-1},\ 0 \le i \le M-1$, in continued column-row order the syndromes $S_\ell^0 \cdots\cdots S_j^{i-1}$. However, at ($i$,$j$) a non-zero discrepancy occurs: $d_j^i = \sum_{k=1}^{\ell} S^i_{j-k}\Lambda_k + S^i_j \neq 0$.

Then $\Lambda(z)$ is updated by the currently active *helper polynomial* for the *i-th* row to obtain a new $\Lambda(z)$ with zero discrepancy also at (i,j), and $\Lambda(z)/d_j^i$ may become the new *helper polynomial* $H^i(z), j^i, \ell^i$ for the *i-th* row associated with unit discrepancy, ... (next slide).

(The initialization of the M helper polynomials is simple, however, functionally hard to explain.)

Multi-sequence shift register synthesis

Row i: updating $\Lambda(z)$ and "helper polynomial" $\mathrm{H}^i(z)$



Multi-sequence Berlekamp-Massey algorithm

```
/* Initialization */
```
$\Lambda(z)=1;\ \ell=0;$ // initial error locator polynomial

for $i=0$ to $M-1$ { $\mathrm{H}^i(z)=1;\ j^i=-1;\ \ell^i=0;$ } // initial helper polynomials

```
/* Loop */
```
for $j=0$ to $R-1$

{ for $i=0$ to $M-1$

{ $d=S^i_j+\sum_{k=1}^{\min(j,\ell)}\Lambda_k S^i_{j-k}$ // compute discrepancy

if $d\neq 0$ // discrepancy is non-zero

{ $\tilde{\ell}=\ell;\ \ \tilde{\Lambda}(z)=\Lambda(z);$ // save current error locator polynomial

// update err. loc. poly. with discrepancy and *i-th* helper polynomial

$\ell=\max(\ell, j-j^k+\ell^k);\ \ \Lambda(z)=\Lambda(z)-d\,\mathrm{H}^i(z)\,z^{j-j^i};$

if $\ell>\tilde{\ell}$ // length of err. loc. poly. increased

{ // normalize $\tilde{\Lambda}(z)$ for unity discr. and store as new *i-th* helper poly.

$j^i=j;\ \ \ell^i=\ell;\ \ \mathrm{H}^i(z)=d^{-1}\tilde{\Lambda}(z);$

} }

} }

Performance evaluation by simulation

- Combining individual decoding and joint decoding
- Gilbert model of additive burst-noise channel
- Simulation results: block-interleaved RS coding + uncoded 16-QAM
- Simulation results: block- and convolutionally-interleaved RS coding + 16s4d trellis-coded 32-QAM (as in ADSL)

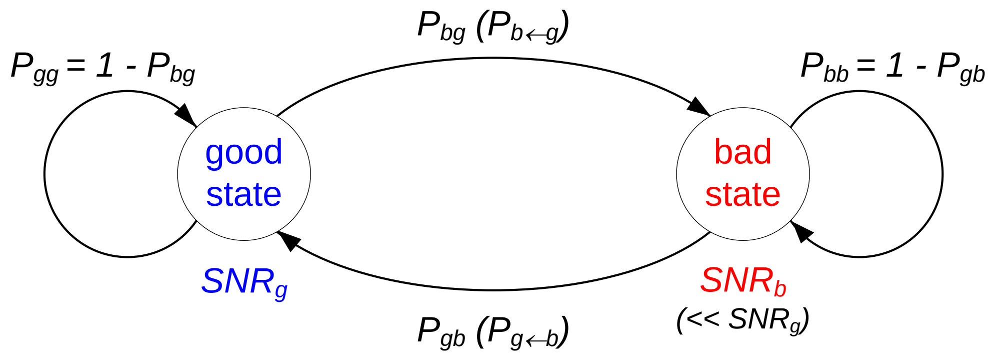
Combining individual decoding and joint decoding

For received *i-th* codeword do   *(i = ... 0, 1, 2, 3, ...)*

1. Decode *i-th* codeword individually: compute syndromes, determine error locator polynomial (ELP), and check ELP validity. Save syndromes in a cyclic buffer for *M* codewords.

2. If for at least one of the last *M* codewords a valid ELP could not be found, determine a joint ELP for the last *M* codewords and check ELP validity.

Gilbert model for additive burst-noise channel



Steady-state probability of being in good/bad state

$$P_g = P_{gb}/(P_{gb}+P_{bg}) \;;\quad P_b = P_{bg}/(P_{gb}+P_{bg}) \text{ (= frequency of noise bursts)}$$

Average number of intervals spend in good/bad state

$$\bar{n}_g = 1/P_{bg} \;;\quad \bar{n}_b = 1/P_{gb} \text{ (= average length of burst noise)}$$

Given: P_b and $\bar{n}_b$. Computation of P_{gb} and P_{bg}

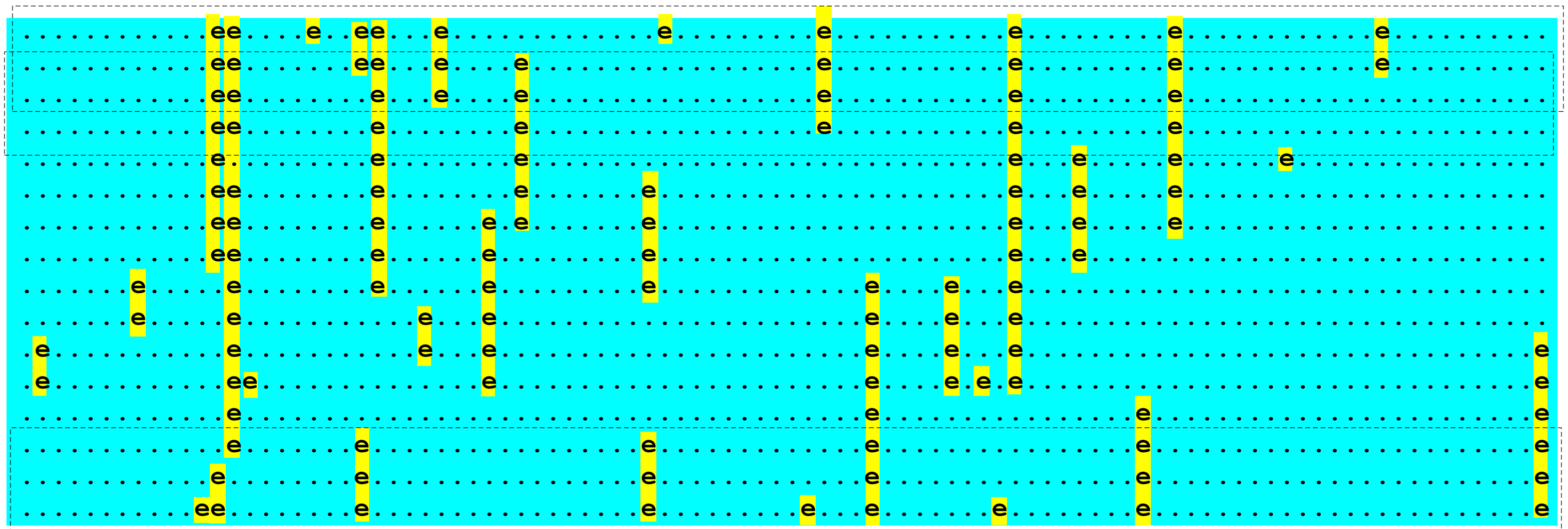
$$P_{gb} = 1/\bar{n}_b \;;\quad P_{bg} = P_b/[\bar{n}_b(1-P_b)]$$

Simulation: uncoded modulation, block interl. RS coding

***D*=16 block-interleaved codewords RS(*N* = 96, *K* = 80; *T* = 8)** over GF(256), transmitted over **uncoded 16-QAM** channel with **AWGN & burst noise** (SNR_g = 22 dB, SNR_b = 0 dB, P_b=0.05, n_b=8)

n ... # of symbol errors per codeword, t ... # of column errors per subblock of M codewords

		n	t
1	0	11	12
1	0	10	10
0	0	8	10
0	0	7	10
0	0	7	10
0	0	8	9
1	0	9	12
0	0	7	11
0	0	8	11
0	0	7	11
0	0	8	11
1	0	9	12
0	0	4	7
0	0	6	10
0	0	6	
1	0	9	



Joint decoding of sliding blocks of M = 3 consecutive codewords (t_{max} = 12): all codewords correctly decoded.

RS symbol errors

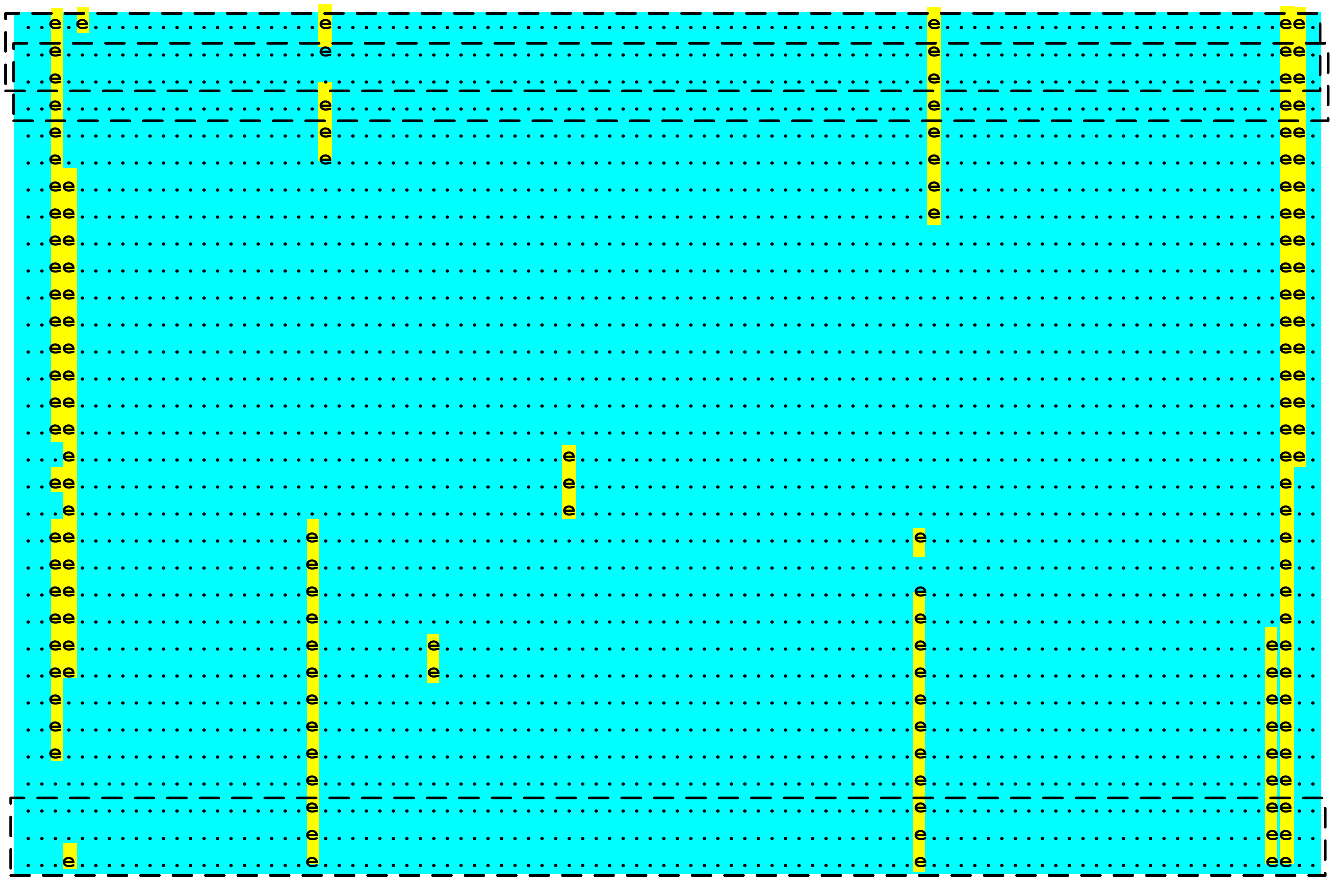
Individual decoding of codewords (t_g = 8): 5 decoding failures

$$t_g = \mathrm{floor}[(N-K)/2] = T \;\; ; \quad t_{\max} = \mathrm{floor}[M(N-K)/(M+1)]$$

Simulation: uncoded modulation, block interl. RS coding

***D*=32 block-interleaved codewords RS(*N* = 96, *K* = 88; *T* = 4)** over GF(256), transmitted over **uncoded 16QAM** channel with **AWGN & burst noise** (SNR_g = 22 dB, SNR_b = 0 dB, P_b=0.04, n_b=30)

1	0	6	6
1	0	5	5
0	0	4	5
1	0	5	5
1	0	5	6
1	0	5	6
1	0	5	5
1	0	5	5
0	0	4	4
0	0	4	4
0	0	4	4
0	0	4	4
0	0	4	4
0	0	4	4
0	0	4	5
0	0	4	5
0	0	4	5
0	0	4	6
0	0	3	6
1	0	5	5
0	0	4	5
1	0	5	7
1	0	5	7
1	1	7	7
1	1	7	7
1	0	5	5
1	0	5	5
1	0	5	5
0	0	4	4
0	0	4	5
0	0	4	
1	0	5	

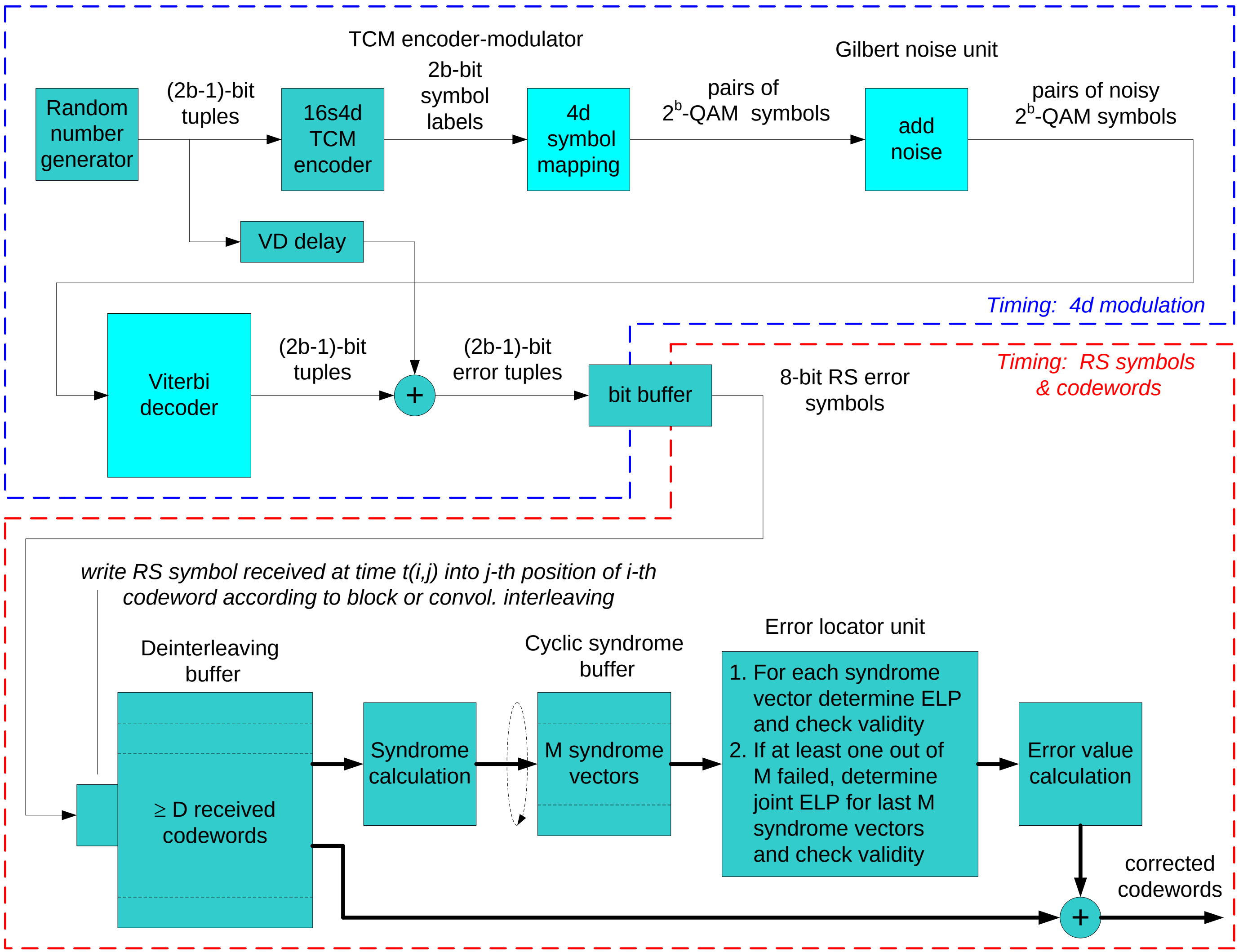


Joint decoding (M = 3, t_{max} = 6): 2 decoding failures

Individual decoding (t_g = 4): 16 decoding failures

Simulated “ADSL”: block or convol. interl. RS, 16s4d TCM, Gilbert burst noise

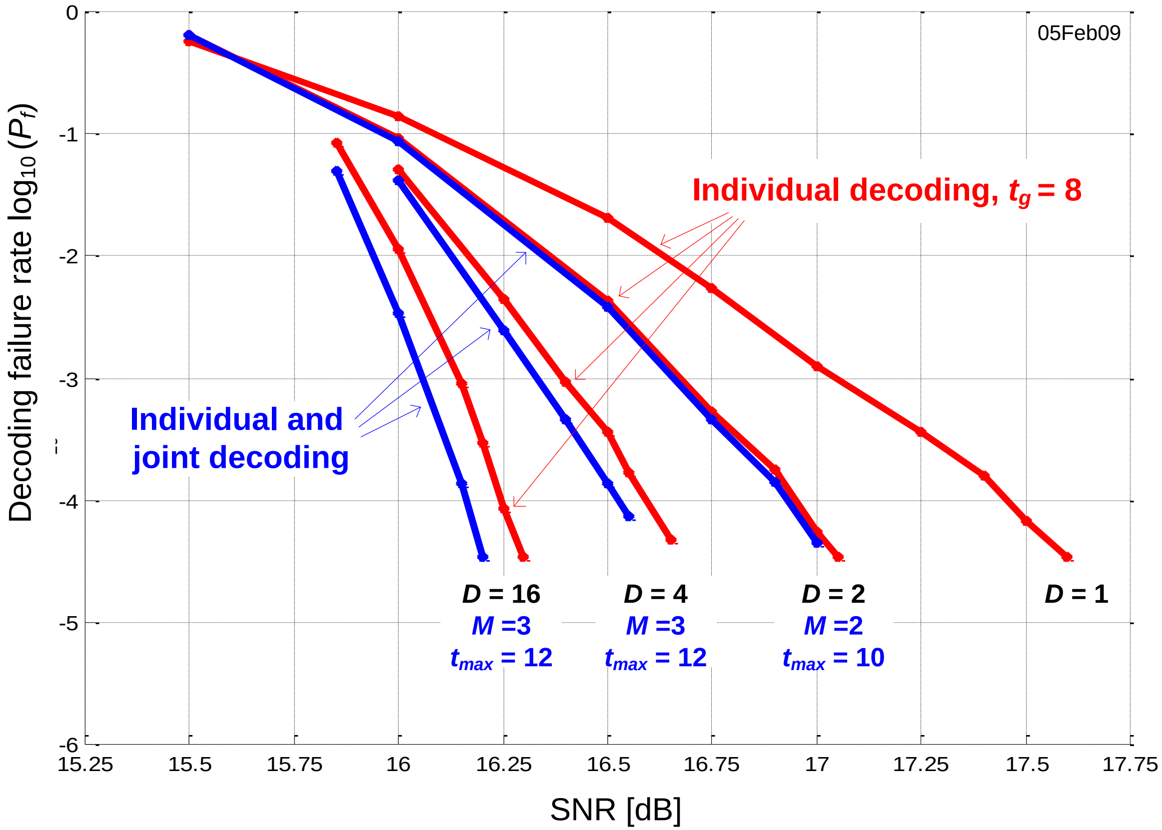
Models transmission of scrambled all-zero RS codewords



Simulation: 16s4d TCM, block interleaved RS coding

16s4d trellis coded 32-QAM, VDdel=32; stationary AWGN (no burst noise)
RS(*N*=255,*T*=8) over GF(256); block interleaving *D* = x; *M* = y;

Decoding failure rate P_f versus SNR, no burst noise

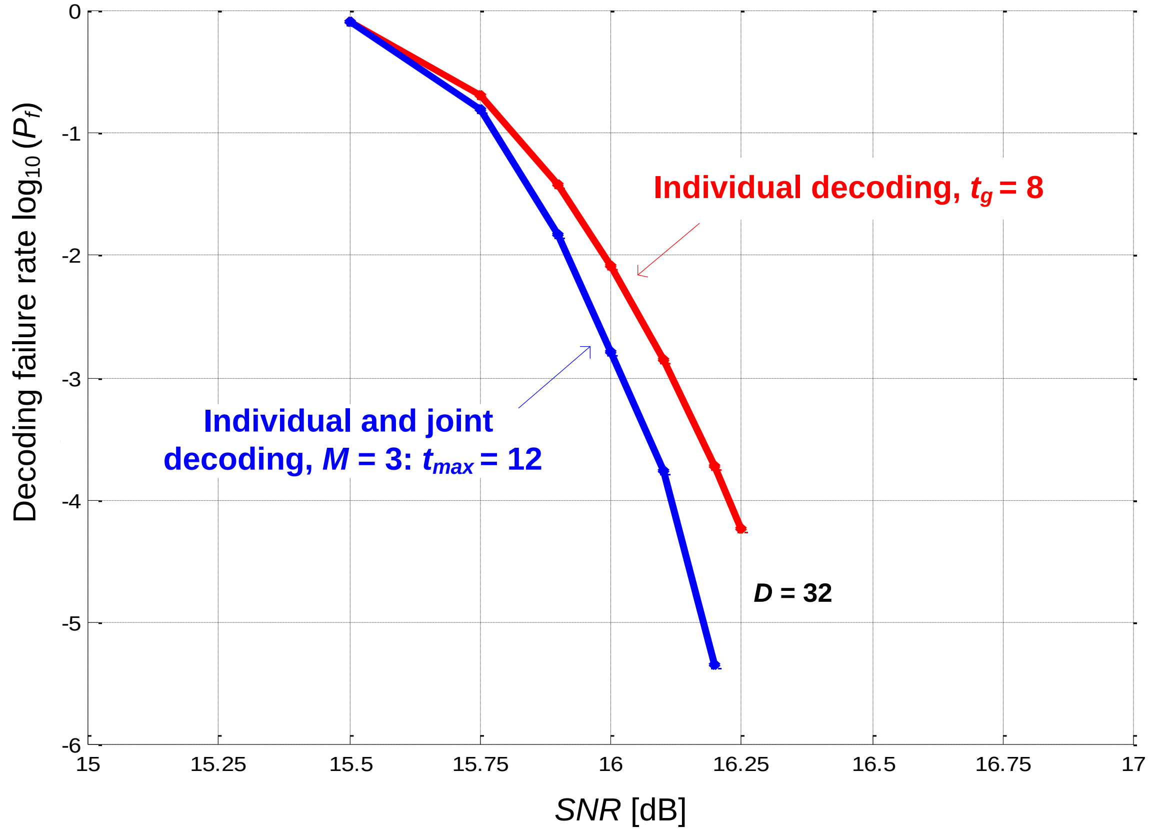


No burst noise: "coding gain" by joint decoding < 0.1 dB

Simulation: 16s4d TCM, convol. interl. RS coding (G.992.1)

16s4d trellis coded 32-QAM, VDdel=32; Gilbert noise $SNR_g = SNR_b$ = x dB;
RS(N=255,T=8) over GF(256); convolutional interleaving D = 32; M = 3;

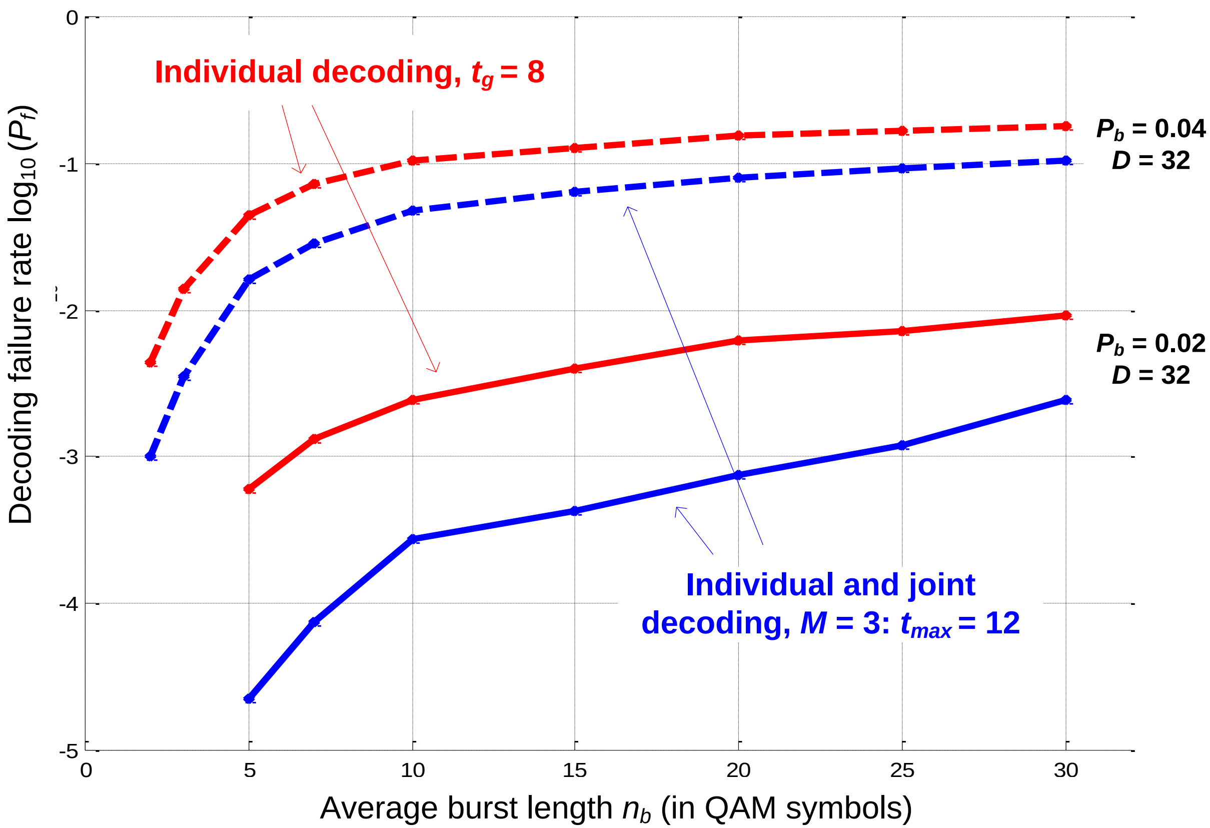
Decoding failure rate P_f versus SNR, no burst noise



Simulation: 16s4d TCM, convol. interl. RS coding (G.992.1)

16s4d trellis coded 32QAM, VDdel=32; Gilbert noise SNR_g = 17dB, SNR_b = 12 dB, P_b = y, n_b = x;
RS(N=255,T=8) over GF(256), convolutional interleaving D = 32; M = 3;

Decoding failure rate P_f versus average burst length n_b



Deep convolutional interleaving (D>>N) and joint decoding of multiple RS codewords

- Correct very long error bursts, e.g., when entire OFDM symbols are wiped out
- Convolutional interleaving in VDSL-2 and in Data-Over-Cable downstream (J.83)
- Illustration of the effect of long error bursts after de-interleaving
- Approaching full erasure-decoding capability without erasure indications

Convolutional interleaving in VDSL2 (G.993.2, 02/2006)

- Codewords: RS(N,K) over GF(256), $N \in \{32, 33, \cdots 255\}$, $R = N - K \in \{0,2,4,6,\cdots 16\}$; $T = R/2$
- Interleaving (general type): *interleaver block length* $I = N/q,\ q \in \{1, 2, 3, \cdots 8\}$, *interleaver depth* $D \le D_{\max}$ such that $\gcd(I, D) = 1,\ D_{\max} = 2048, 3072, \text{or } 4096$.

Let $\mathbf{c}_i = [c_{i,0}, c_{i,1}, \cdots c_{i,N-1}]$ be the *i-th* RS codeword. The temporal position of RS symbol $c_{i,j}$ before interleaving is $s = N\,i + j = N\,i + I\,j_1 + j_0,\ i \in Z,\ 0 \le j = I\,j_1 + j_0 < N \quad (0 \le j_1 < q, 0 \le j_0 < I)$. The interleaver changes the symbol order such that after interleaving $c_{i,j}$ occurs at temporal position $t = N\,i + I\,j_1 + D\,j_0$. The condition $\gcd(I, D) = 1$ ensures collision-free permutation, i.e., $s \ne s'$ implies $t \ne t'$.

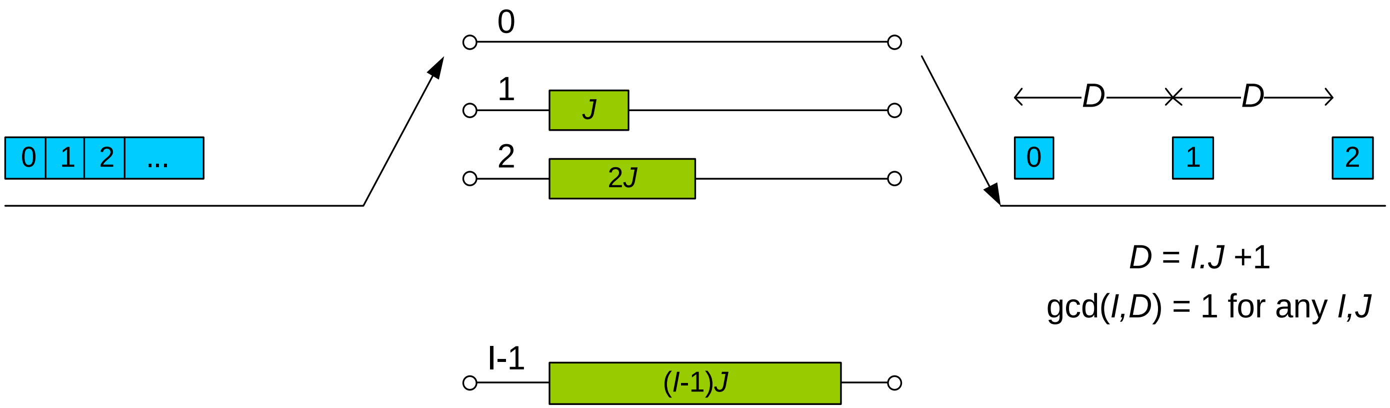
Let $I_D^{-1}\ (1 \le I_D^{-1} < D)$ be the *multiplicative inverse* of I such that $I.I_D^{-1} \bmod D = 1$. Let $D_I^{-1}\ (1 \le D_I^{-1} < I)$ be the *multiplicative inverse* of D such that $D.D_I^{-1} \bmod I = 1$. Note that $I_D^{-1} I + D_I^{-1} D = 1 + I\,D$.

Inverse mapping $t \to (i, j = I\,j_1 + j_0)$: $j_0 = t.D_I^{-1} \bmod I$, $j_1 = [(t - D\,j_0)/I] \bmod q$, $i = (t - N\,j)/N$.

Convolutional interleaving in cable systems (J.83, 12/2007)

- **J.83 Annex B:** Extended codewords RS(N=128,K=122;T=3) over GF(128). Convolutional interleaving, reduced mode: (I,J) = (128,1), (64,2), (32,4), (16,8), (8,16); enhanced mode: (I,J) = (128, 1 to 8). (N/I = q = 1 to 8)

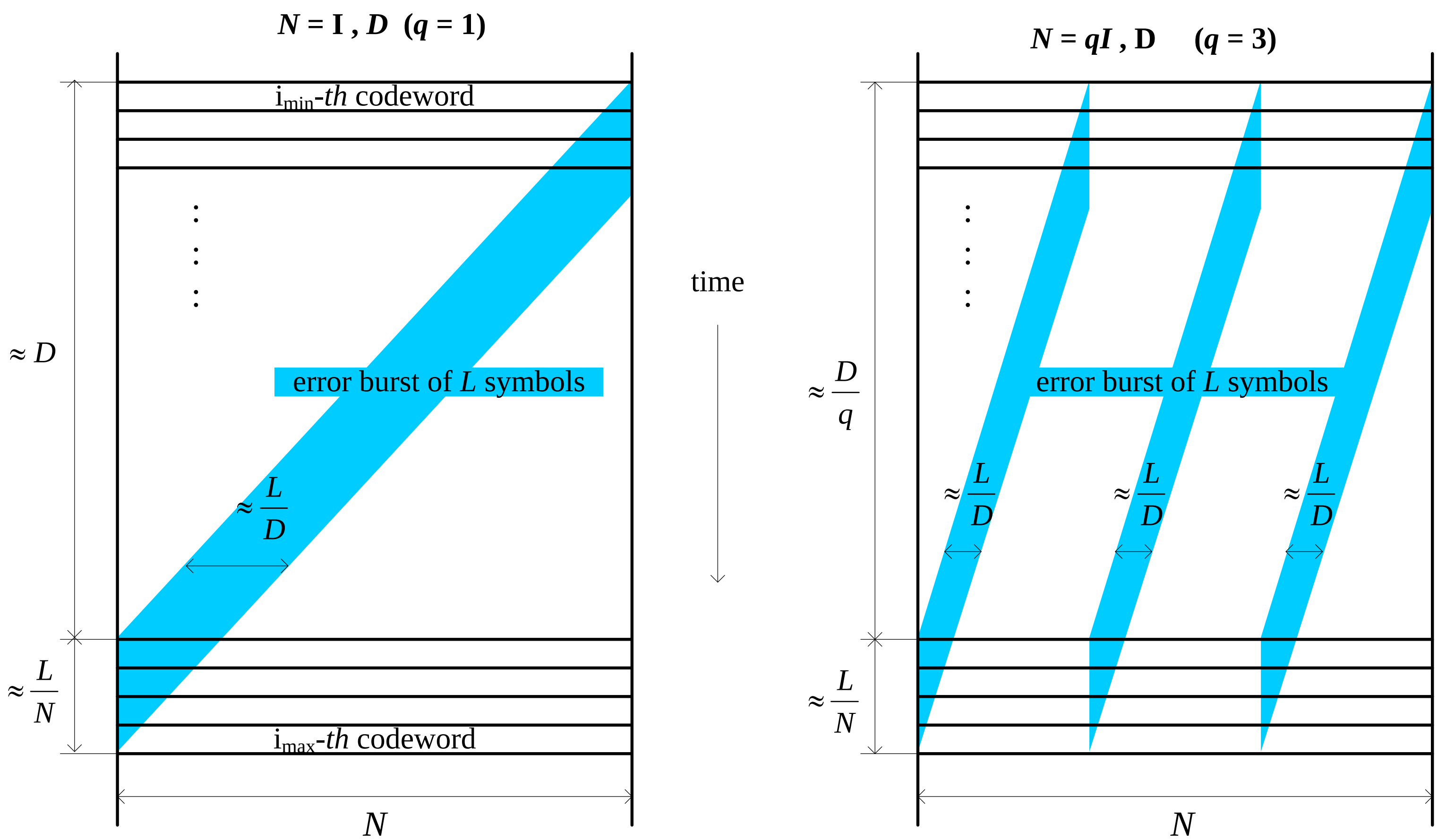
- **J.83 Annex C & D:** Shortened codewords RS(N=204,K=188;T=8) over GF(256). Convolutional interleaving: (I,J) = (12,17). (N/I = q = 17)



Convolutional interleaver: special type (Forney 1971)

Convolutional interleaving: long error burst

N, I, D assumed to be large

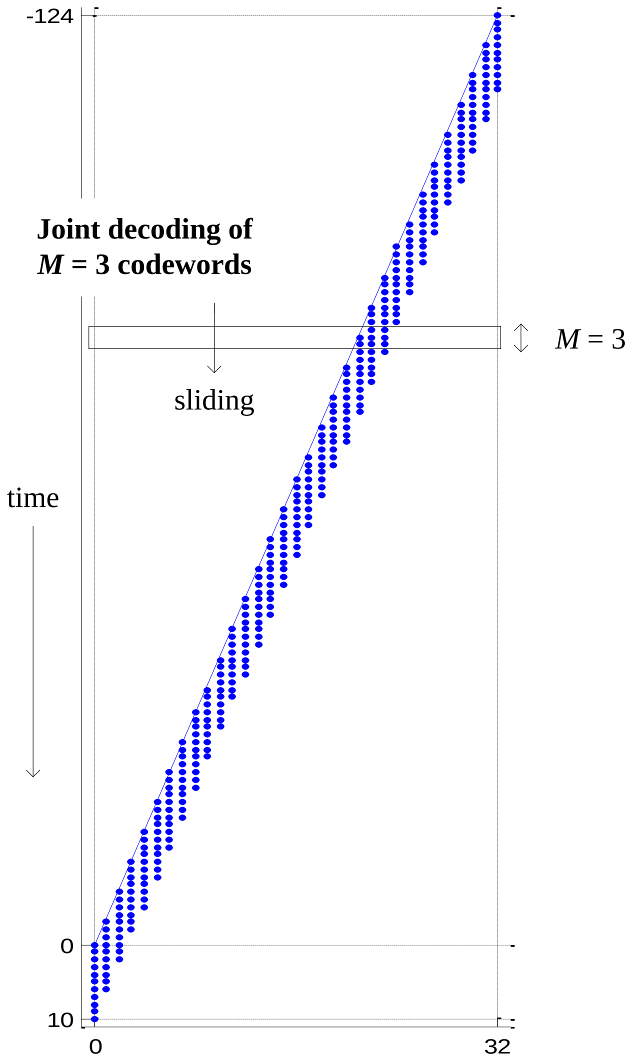


Interleaver-deinterleaver latency $(I-1)(D-1)$

Length of correctable burst error $qL/D \le T \rightarrow L \le TD/q$

***q* > 1: what advantage?**

Convolutional interleaving, joint decoding of *M* codewords



RS(N = 33, K = 29; T=2)

N=I = 33, D = 128

(D >> N, deep interleaving)

Error burst from $t = 0$ to $t = 349$ ($t_0 = 0$, L=350)

Correction by individual decoding fails because $t_g = T = 2$.

Correction by sliding joint decoding with M = 3, $t_{max} = 2T \times M/(M+1) = 3$ succeeds .

Sliding joint decoding achieves burst error correction similar to full-erasure decoding without requiring erasure indications (which often can be wrong).

Plot generated by Matlab function `ConvolInterlVDSL2.m(N,I,D,t0,L)`, 10Feb09

Concluding Remarks (1)

- **Joint decoding of received RS codewords in combination with a first pass of individual decoding reduces RS decoding failures typically by a factor of 10 in the presence of burst noise, provided errors are sufficiently aligned column-wise in consecutively received de-interleaved codewords.**

- **Column-wise alignment of burst errors occurs naturally for block interleaving, but not for convolutional interleaving.**

- **Joint decoding needs to be invoked only when in a sliding window of *M* consecutively received codewords one or more codewords cannot be decoded individually.**

- **The number of jointly decoded codewords can and should be rather small; often it suffices to choose $M = 3$.**

- **The multi-sequence extension of the Berlekamp-Massey algorithm (eBMA) appears to be the most practical algorithm for solving the joint error-locator polynomial problem; complexity of the eBMA is proportional to T^2 and M.**

Concluding Remarks (2)

- **Syndrome calculation and error evaluation can be shared between individual and joint decoding. Additional complexity for joint decoding results from storing *M* syndrome vectors, and occasionally performing the eBMA and checking ELP validity by Chien search.**

- **Convolutional interleaving with modest interleaving depth ($D < N$): consecutively received codewords should be cyclically rotated to align short burst errors in columns. This restricts codeword length to $N = 2^m-1$ such that RS code is cyclic.**

- **Deep convolutional interleaving ($D >> N$) for correction of long error bursts: errors are sufficiently aligned in consecutively received codewords w/o rotation. Sliding joint decoding achieves burst error correction similar to full-erasure decoding (with all-correct erasure indications), but does not require erasure indications.**