Distributed Source Coding: Theory, Code Designs, & Applications

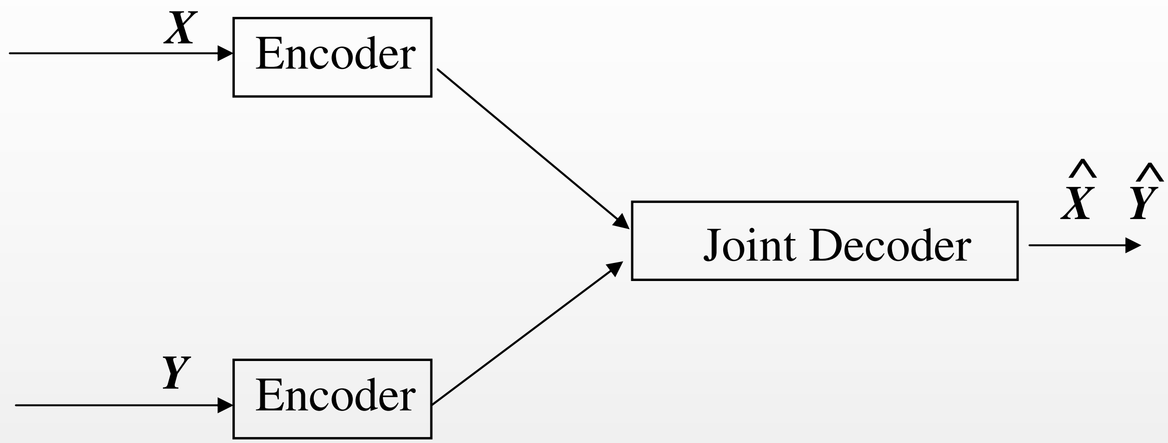
IEEE ICASSP'09
Taipei, Taiwan
April 20 2009

Speakers: Dr Vladimir Stankovic, Dr Lina Stankovic, Dr Samuel Cheng

Contact Details

- Vladimir Stanković
  - Department of Electronic and Electrical Engineering, University of Strathclyde
  - Email: vladimir.stankovic@eee.strath.ac.uk
  - Web: http://personal.strath.ac.uk/vladimir.stankovic
- Lina Stanković
  - Department of Electronic and Electrical Engineering, University of Strathclyde
  - Email: lina.stankovic@eee.strath.ac.uk
  - Web: http://personal.strath.ac.uk/lina.stankovic
- Samuel Cheng
  - Department of Electrical and Computer Engineering, University of Oklahoma
  - Email: samuel.cheng@ou.edu
  - Web: http://faculty-staff.ou.edu/C/Szeming.Cheng-1/

Distributed Source Coding (DSC)

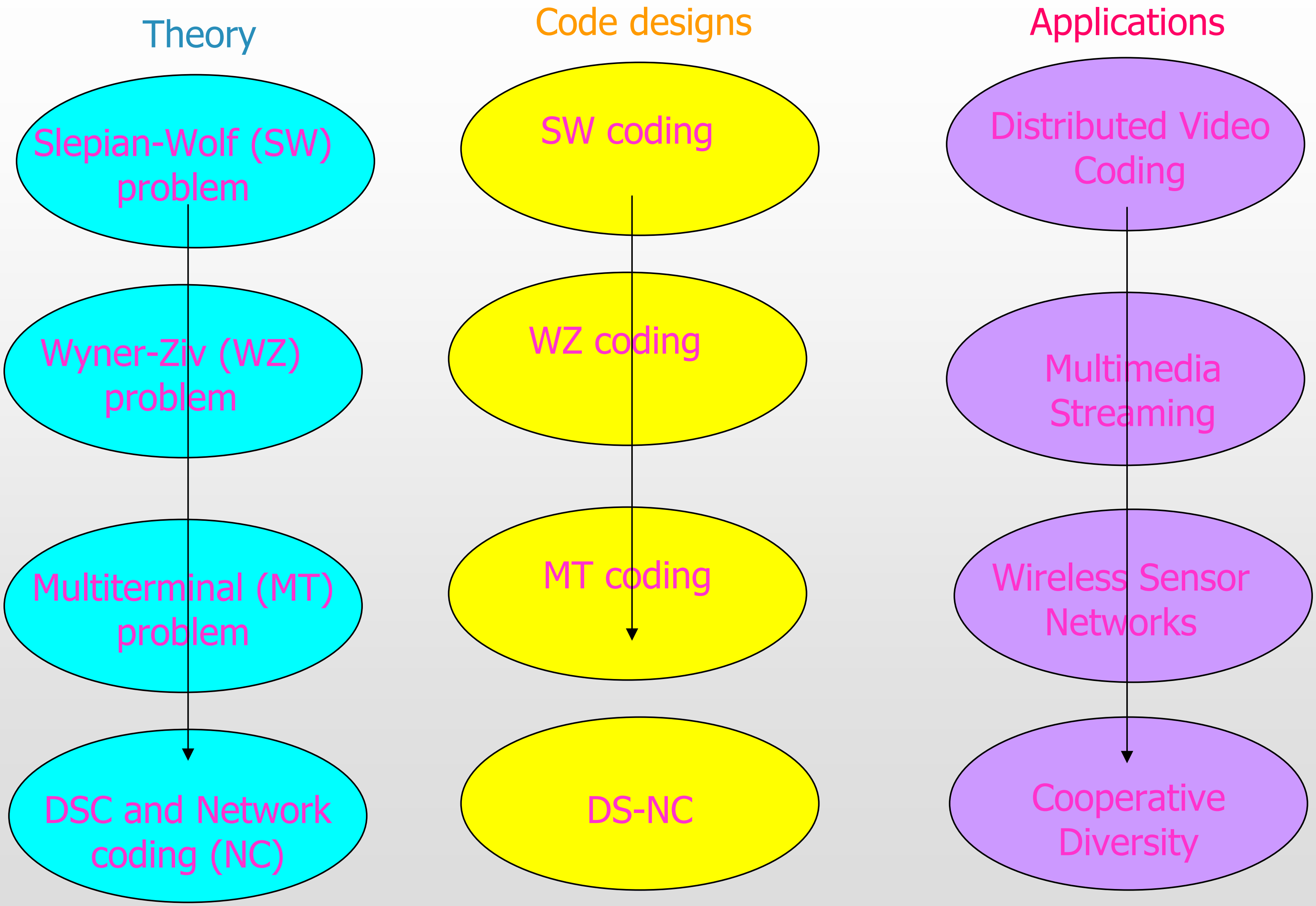


- Compression of two or more physically separated sources
  - The sources do not communicate with each other (hence *distributed coding*)
  - Noiseless transmission to the decoder
  - Decoding is performed jointly
- A compression or source coding problem of network information theory

Motivation

- Increased interest in DSC due to many potential applications
  - Data gathering in wireless sensor networks
  - Distributed (or Wyner-Ziv) video coding
  - Multiple description coding
  - Compressing encrypted data
  - Streaming from multiple servers
  - Hyper-spectral imagining
  - Multiview and 3D video
  - Cooperative wireless communications

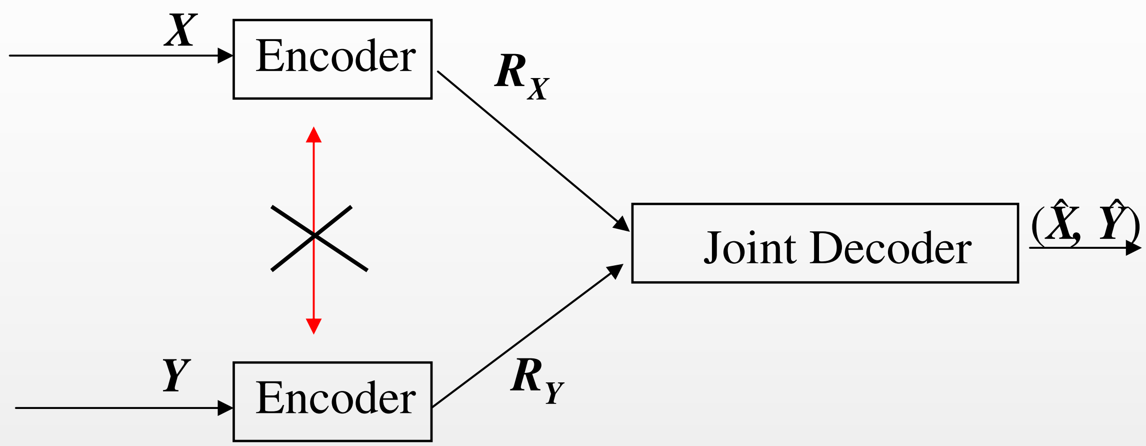
Talk Roadmap



DSC: Problem Setup and Theoretical Bounds

Slepian-Wolf (SW) Problem

X and Y are discrete, correlated sources



R_X and R_Y are compression rates

Joint Encoding:

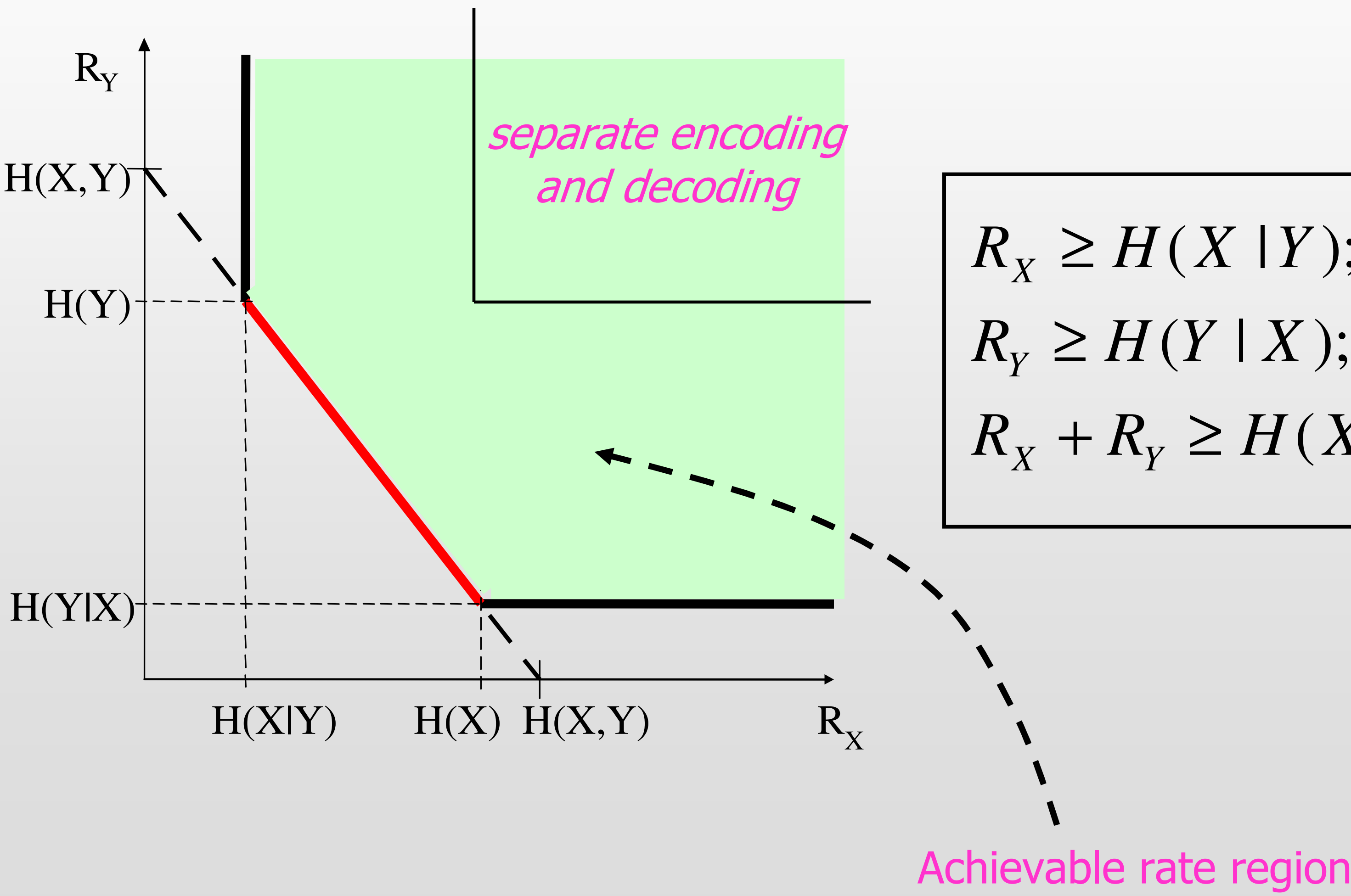
$R=R_Y+R_X=H(X,Y) < H(X)+H(Y)$

Separate Encoding:

$R=R_Y+R_X=H(X,Y) < H(X)+H(Y)$

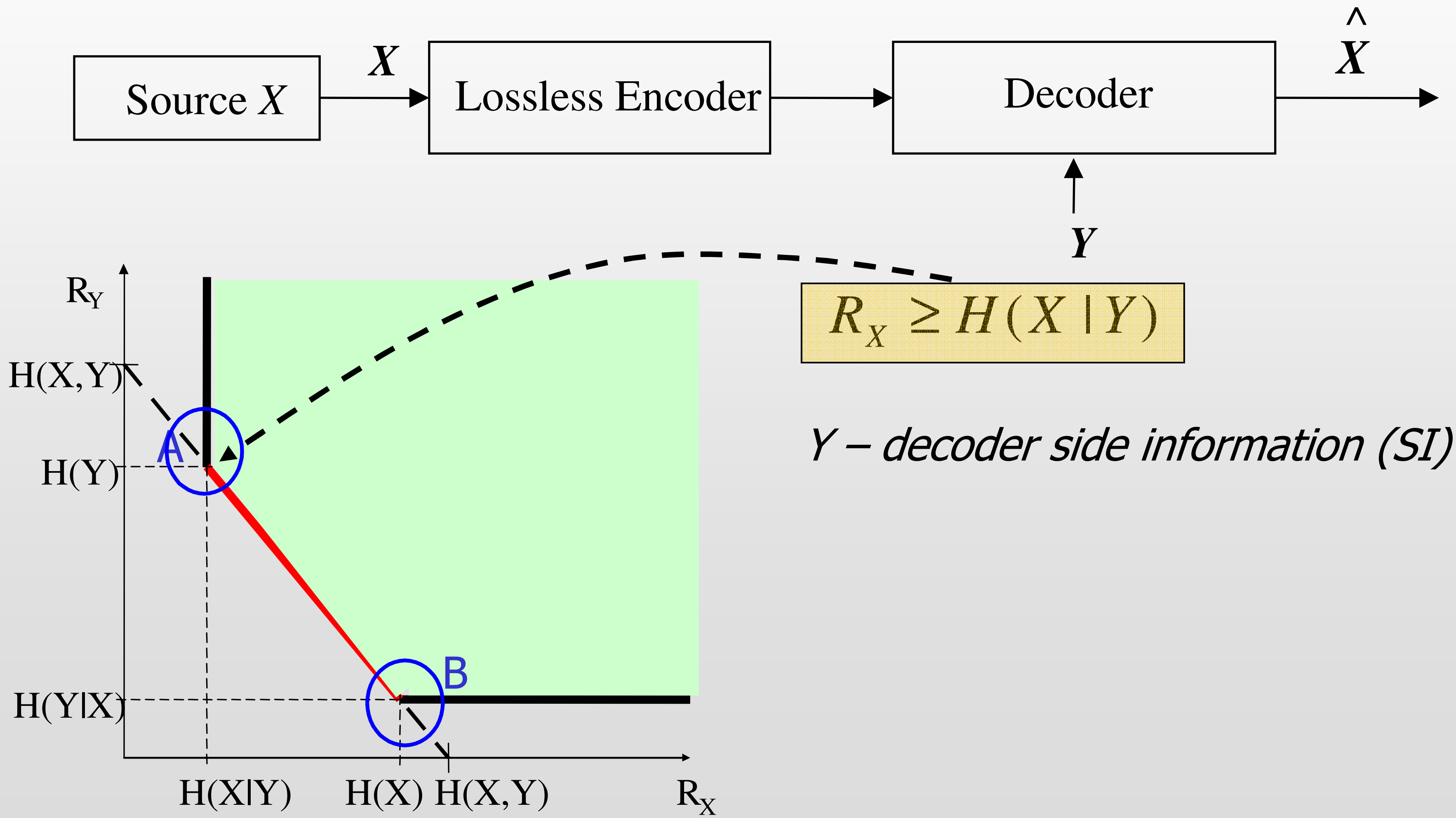
Separate encoding is as efficient as joint encoding!

SW Problem:
The Rate Region



$$R_X \geq H(X \mid Y);$$
$$R_Y \geq H(Y \mid X);$$
$$R_X + R_Y \geq H(X, Y)$$

Source Coding with Decoder Side Information (Asymmetric SW)

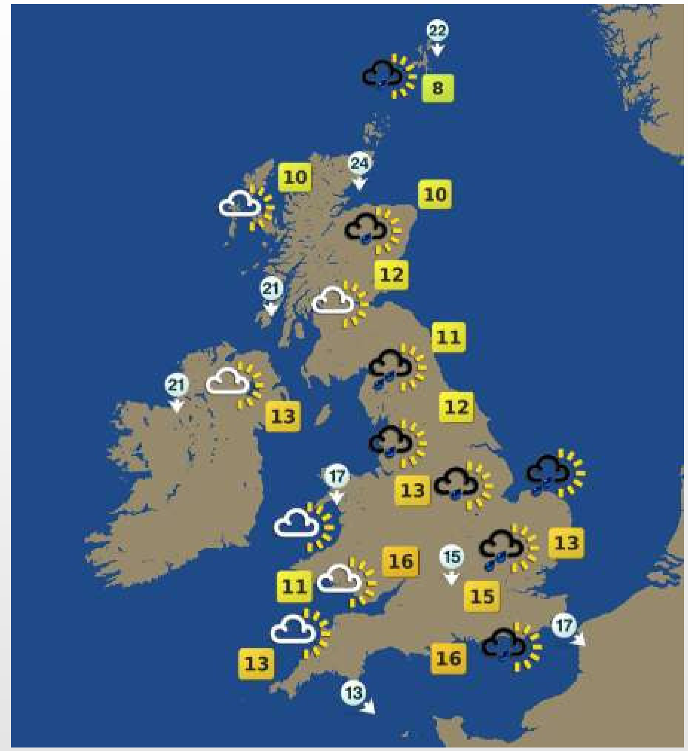


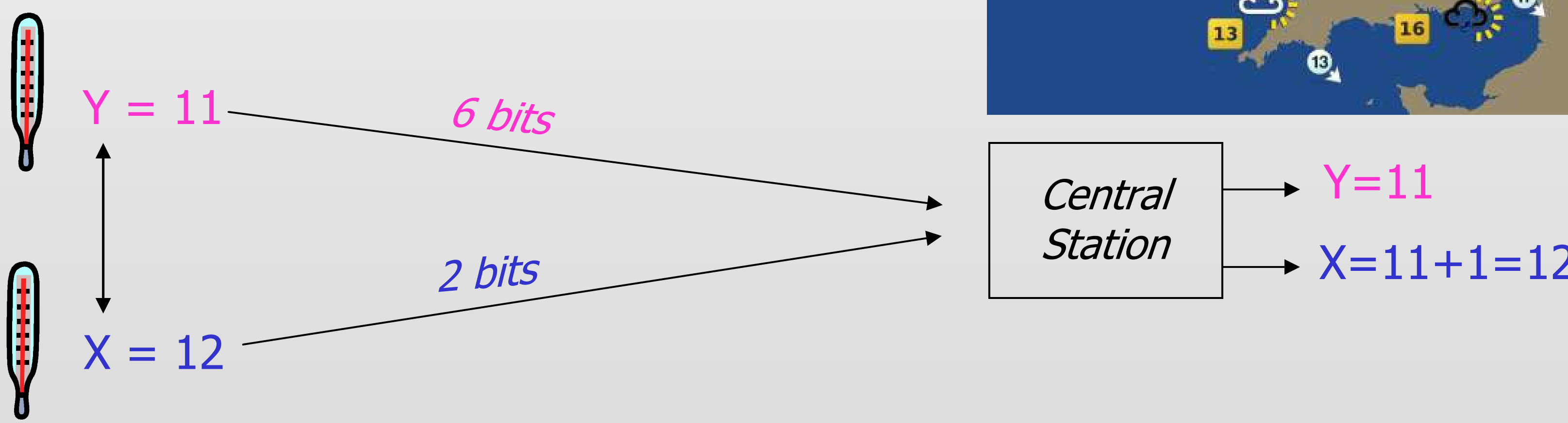
$$R_X \geq H(X|Y)$$

Y – decoder side information (SI)

Compression of Correlated Sources

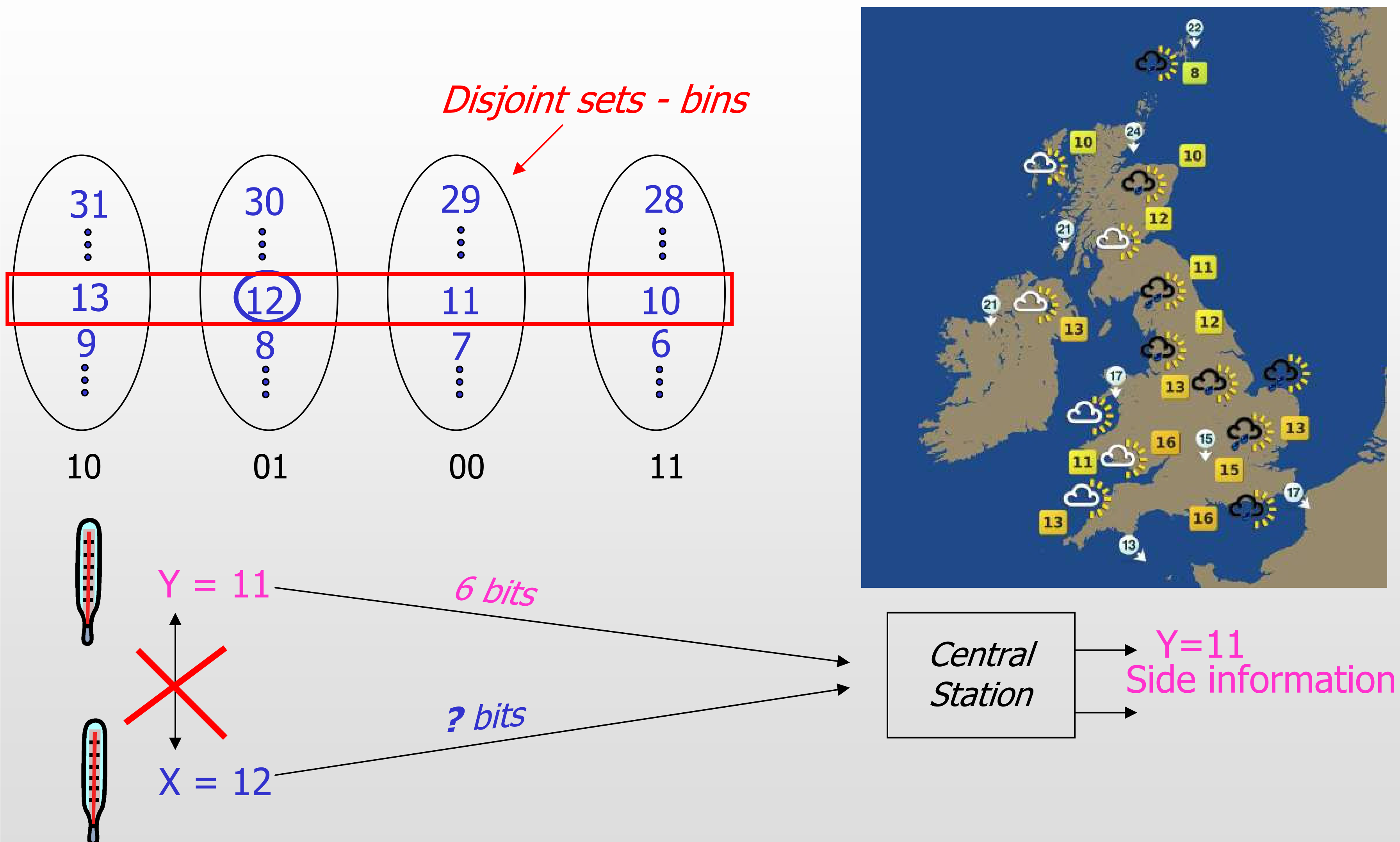
Temp max = 31 degrees + sign bit = 6 bits





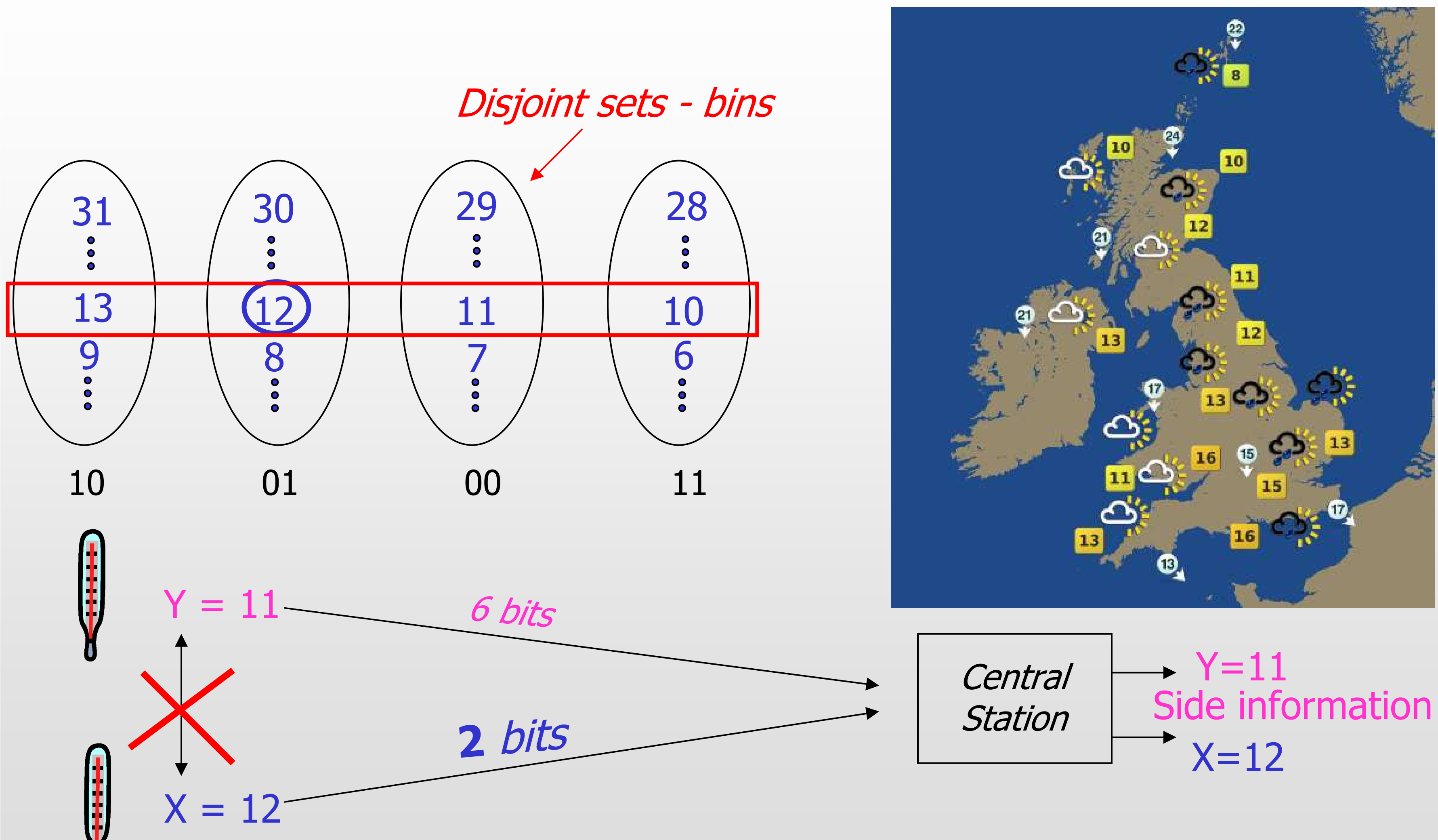
Correlation: X=[Y-1, Y, Y+1, Y+2]

Compression of Correlated Sources



Correlation: X=[Y-1, Y, Y+1, Y+2]

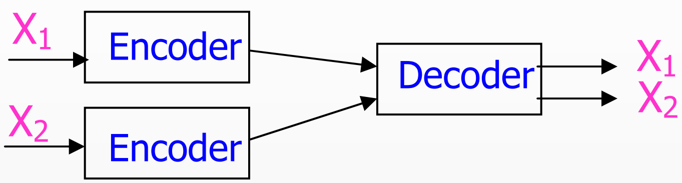
Compression of Correlated Sources



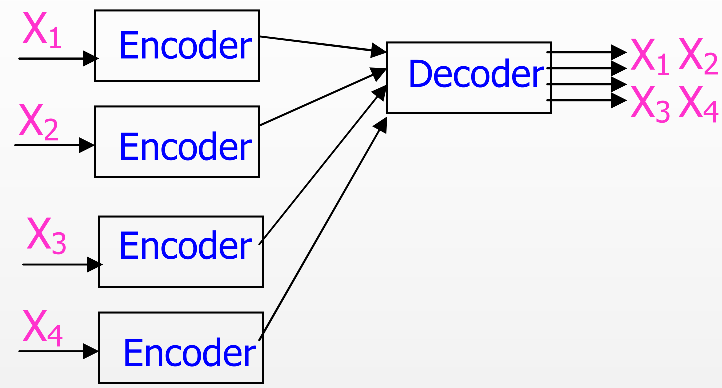
Correlation: X=[Y-1, Y, Y+1, Y+2]

Slepian-Wolf theorem: Still two bits are needed for compressing X!

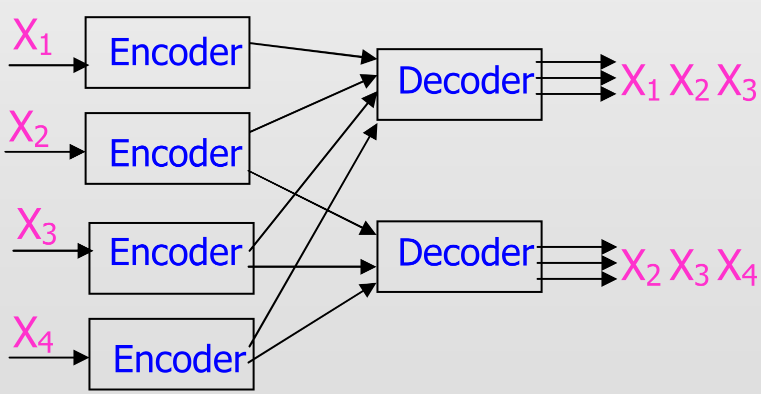
Slepian-Wolf network *(Slepian & Wolf '73)*



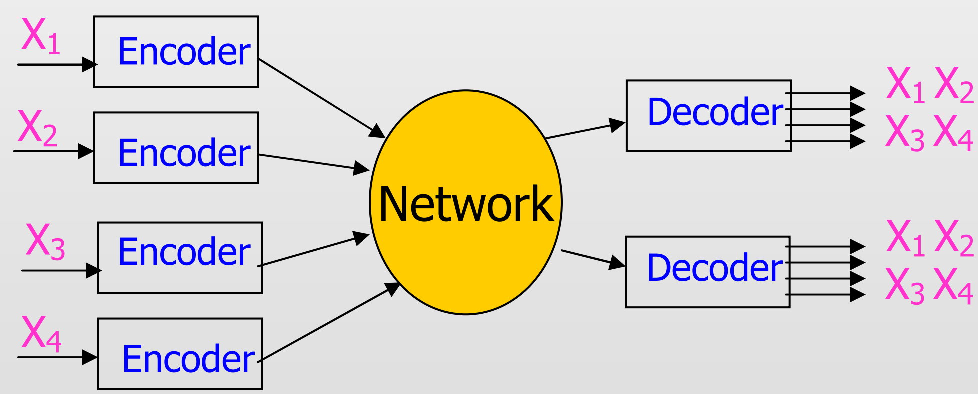
Slepian-Wolf-Cover network *(Wolf '74, Cover '75)*



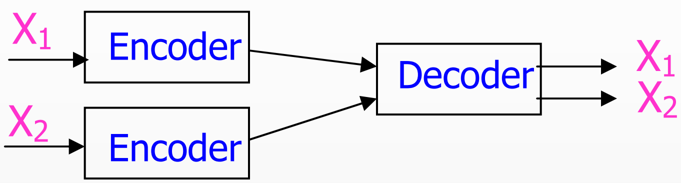
Lossless multiterminal network
(Csiszár & Körner '80, Han & Kobayashi '80)



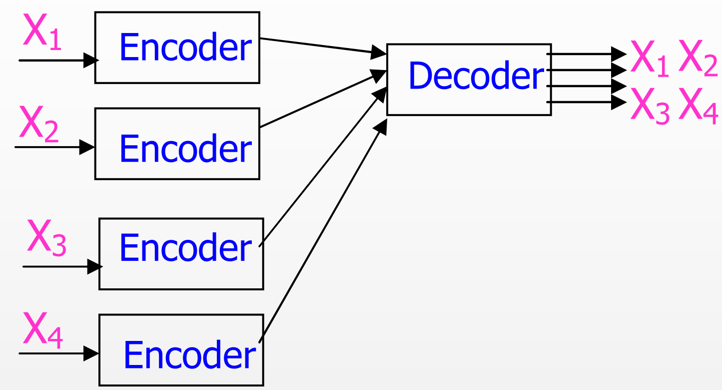
Uncorrelated sources over network *(Ahlswede et al. '00)*



Slepian-Wolf network *(Slepian & Wolf '73)*

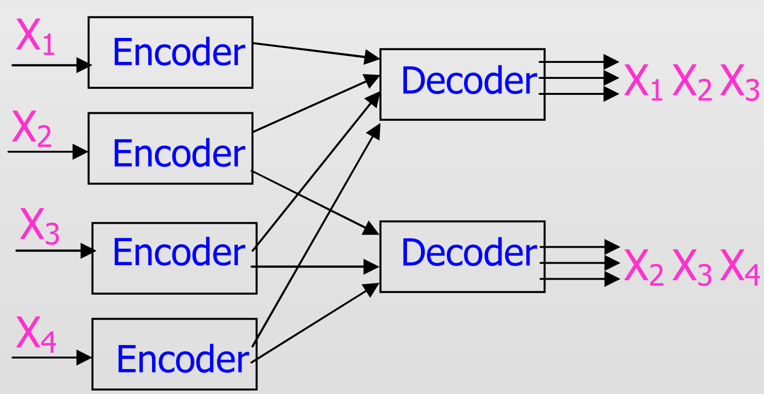


Slepian-Wolf-Cover network *(Wolf '74, Cover '75)*

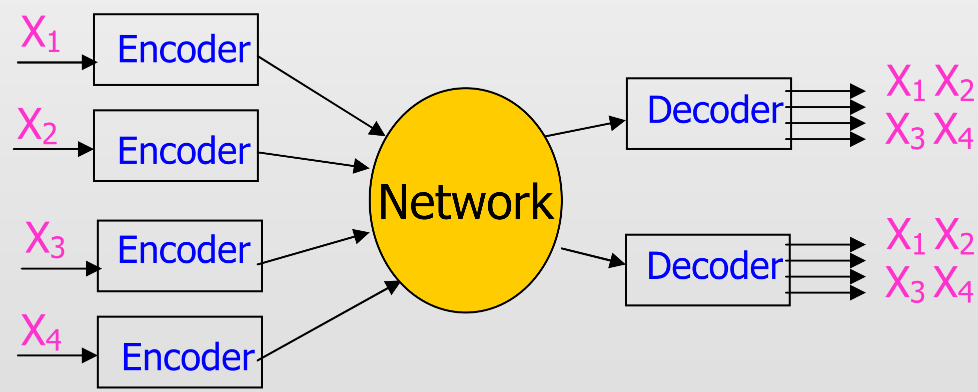


Lossless multiterminal network

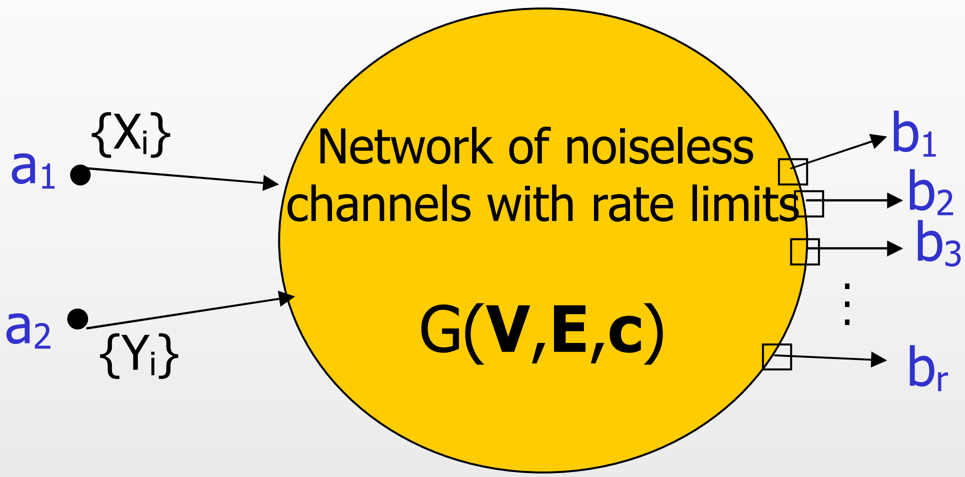
(Csiszár & Körner '80, Han & Kobayashi '80)



Correlated sources over network *(Song & Yeung '01)*



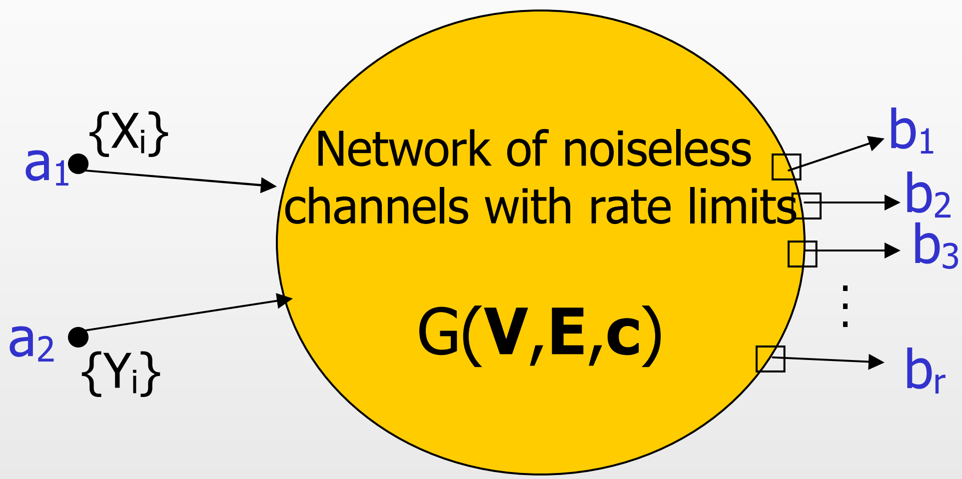
Correlated Sources over Network: Problem Setup

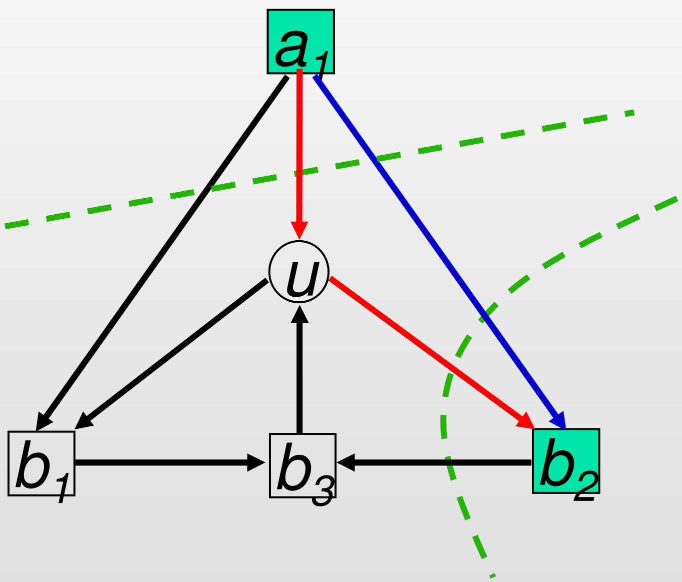


X and Y – discrete, correlated, memoryless sources
V – set of nodes in the network
E – set of edges
Edge $e \in$ **E** – noiseless channel with bit-rate constraint c(e)
c – vector of rate constraints c(e), $\forall e \in$ **E**

- Each destination b_i wants to reconstruct perfectly both X and Y
- How do we determine **c**?

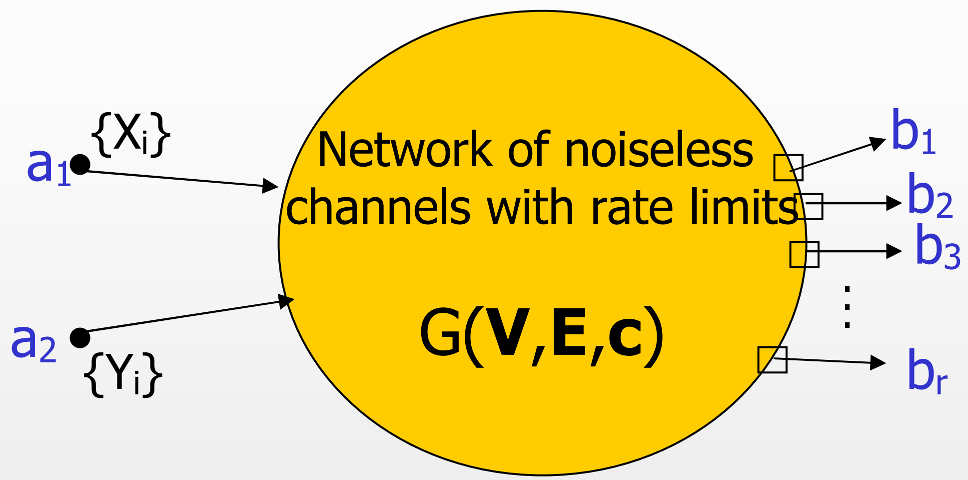
Max-flow=min-cut (Graph Theory)



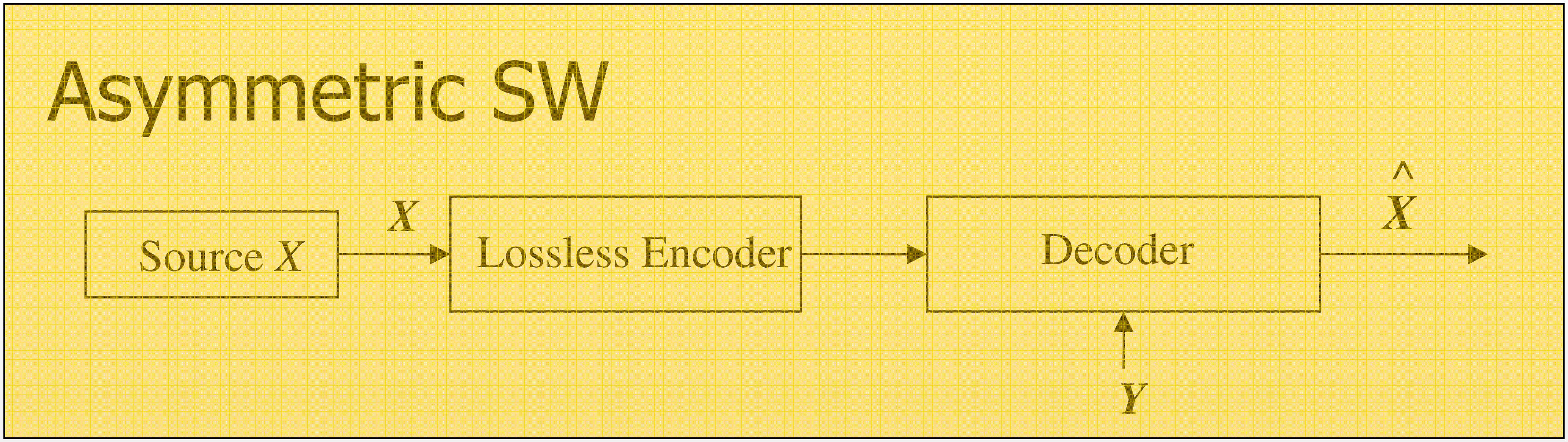


Cuts determine the bit-rate constraint of the links between any two nodes by disconnecting the edges in the graph network

Theoretical Limits *(Han '80, Song & Yeung '01)*



- A rate vector **c** is achievable if and only if :
  - Each cut separating a_1 from any b has at least capacity H(X|Y)
  - Each cut separating a_2 from any b has at least capacity H(Y|X)
  - Each cut separating a_1 and a_2 from any b has at least capacity H(X,Y)

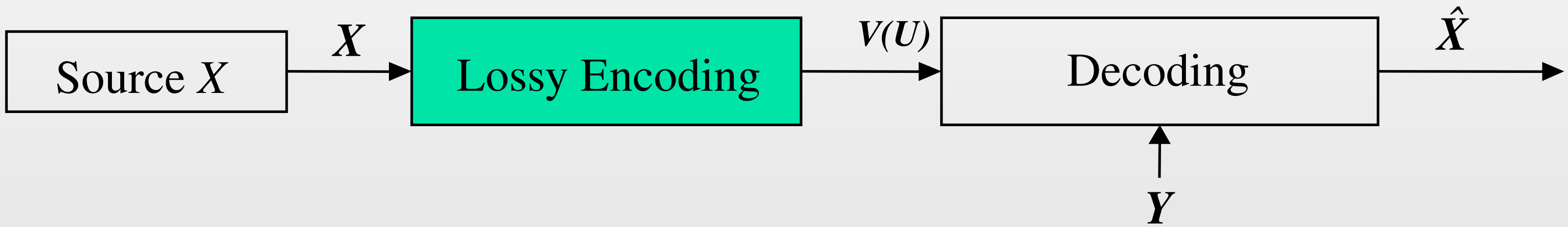


Extending above, i.e., source coding with decoder side information, to lossy coding

Wyner-Ziv coding

Wyner-Ziv (WZ) Problem

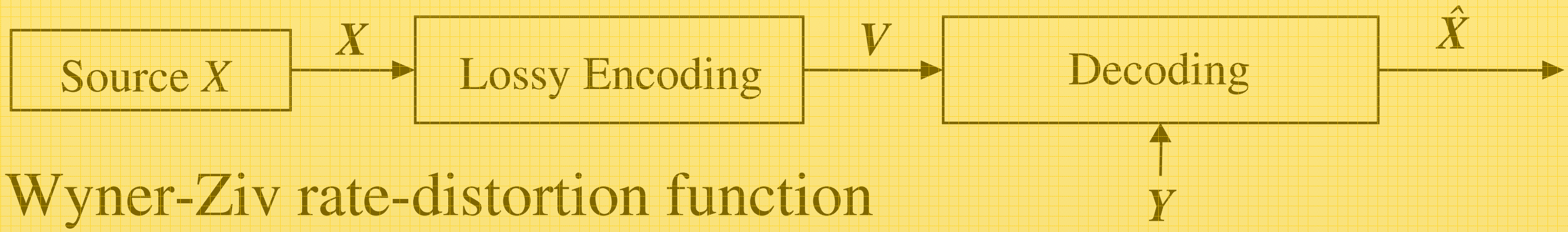
- Lossy source coding of *X* with decoder side information *Y*
- Extension of asymmetric SW setup to rate-distortion theory
- Distortion constraint at the decoder: $E[d(X,\hat{X})] \leq D$



- Wyner-Ziv rate-distortion function

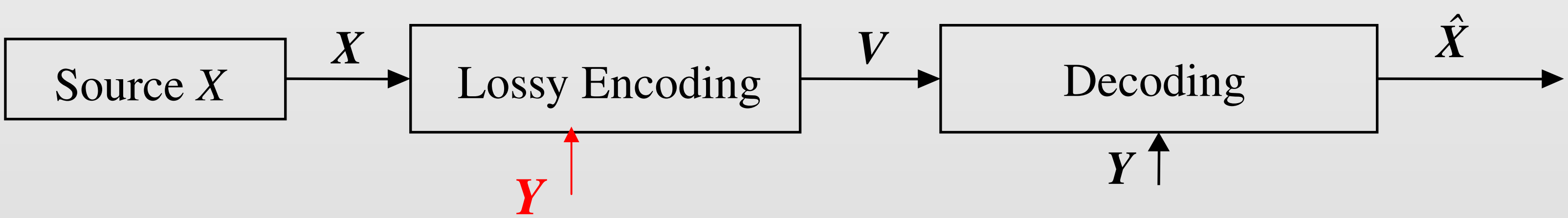
$$R_{WZ}(D) = \inf_{\substack{E[d(X,\hat{X})] \leq D \\ Y \leftrightarrow X \leftrightarrow U \\ \hat{X} \leftrightarrow (U,Y) \leftrightarrow X}} I(X;U|Y)$$

Wyner-Ziv (WZ) Problem



- Wyner-Ziv rate-distortion function

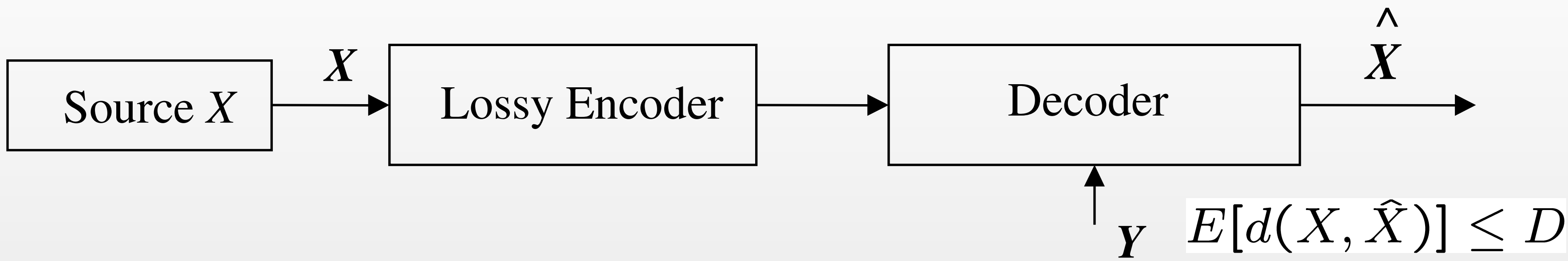
$$R_{WZ}(D) = \inf_{\substack{E[d(X,\hat{X})]\leq D \\ Y\leftrightarrow X\leftrightarrow U \\ \hat{X}\leftrightarrow(U,Y)\leftrightarrow X}} I(X;U|Y)$$



- Conditional rate-distortion function (side information at both sides)

$$R_{X|Y}(D) = \inf_{E[d(X,\hat{X})]\leq D} I(X;\hat{X}|Y)$$

WZ: Lossy Source Coding with Decoder SI

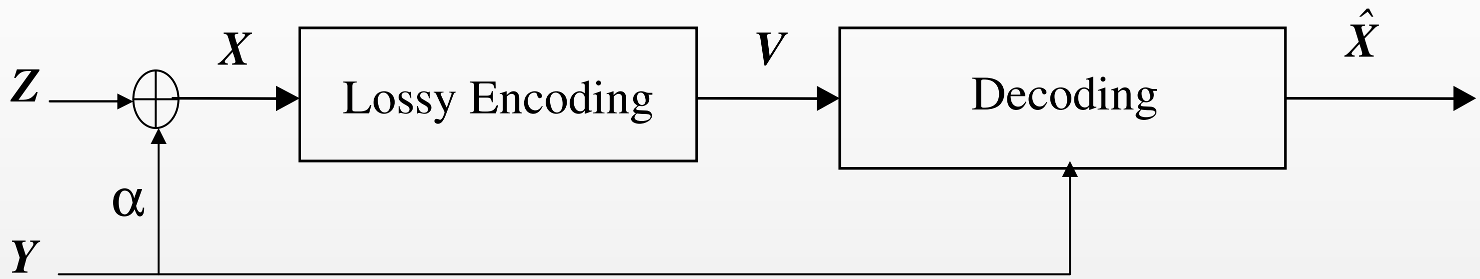


- In general, $R_{WZ} \geq R_{X|Y}$, i.e., there is a rate loss in WZ coding compared to joint encoding

<u>Rate loss</u> (*Zamir '96*) :

- Less than 0.22 bit for binary sources and Hamming measure
- Less than 0.5 bit for continuous sources and mean-square error (MSE) measure

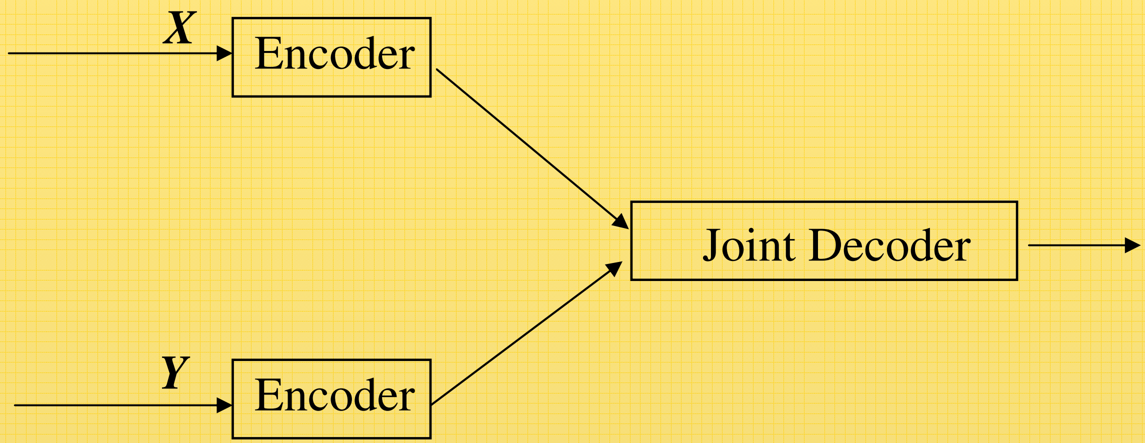
WZ: Jointly Gaussian Sources with MSE Distortion Measure



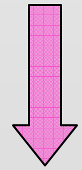
For MSE distortion and jointly Gaussian X and Y, rate-distortion function is the same as for joint encoding and joint decoding

- Correlation model: $X = \alpha Y + Z$, where $Y \sim N(0, \sigma_Y^2)$ and $Z \sim N(0, \sigma_Z^2)$ are independent, Gaussian
- $$R_{WZ} = R_{X|Y}(D) = \frac{1}{2}\log\frac{\sigma_Z^2}{D}$$
- No rate loss in this case compared to joint encoding $R_{X/Y}$

Non-asymmetric SW



Extending above to lossy coding

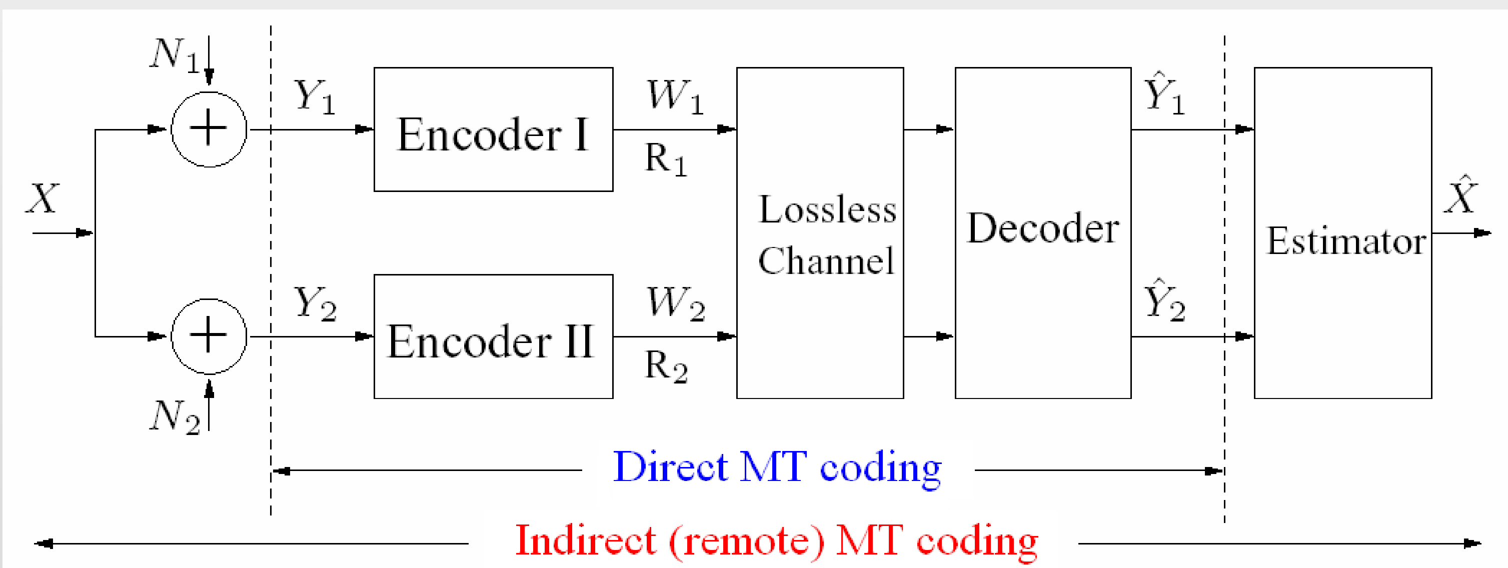


Multiterminal source coding

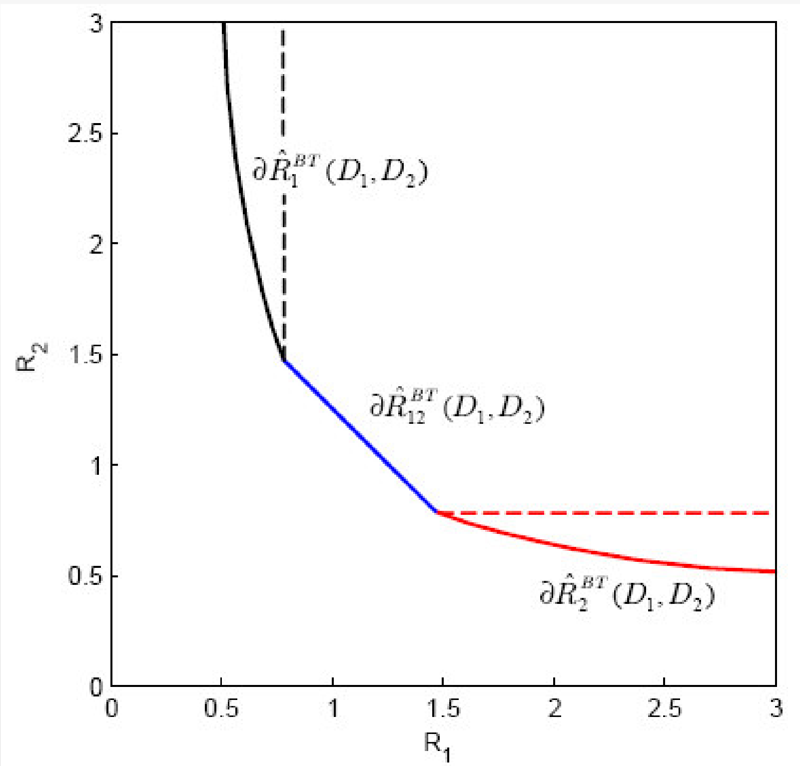
Multiterminal (MT) Source Coding

(Berger & Tung '77, Yamamoto & Itoh '80)

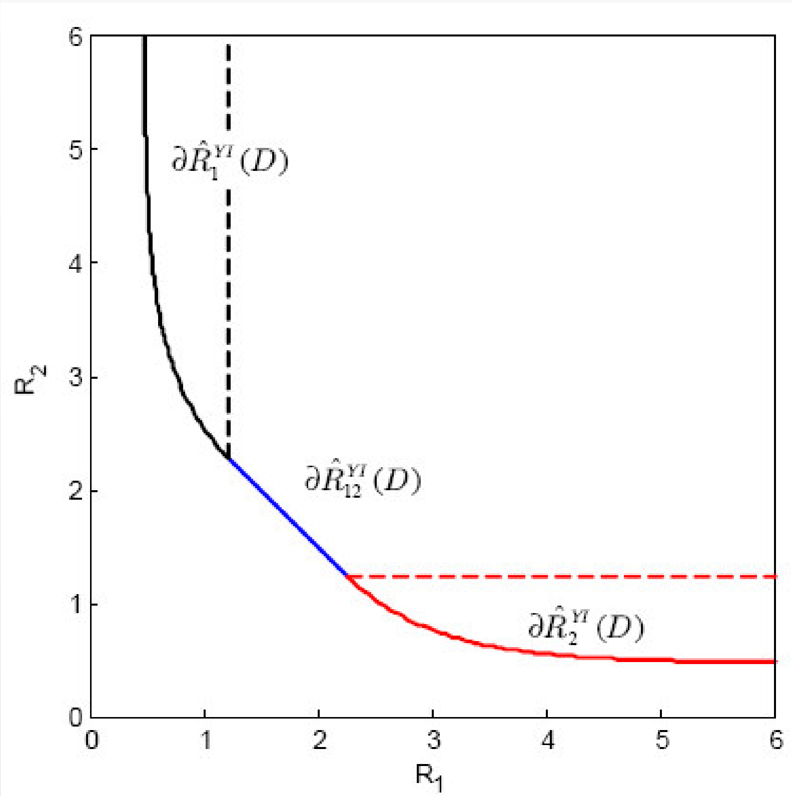
- Non-asymmetric WZ setup
- Extension of the SW setup to rate-distortion theory
- Two types: direct and indirect/remote MT source coding



Quadratic Gaussian MT Source Coding with MSE Distortion



Direct MT Coding



Indirect MT Coding

Quadratic Gaussian Direct MT Source Coding with MSE Distortion

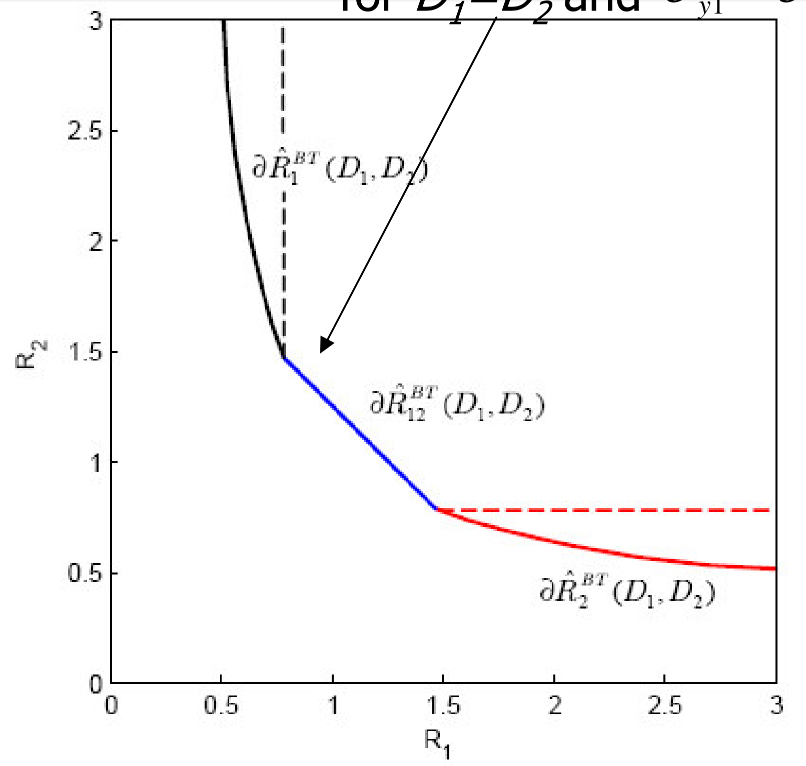
(Wagner '05)

$$\hat{\mathcal{R}}_i^{BT}(D_1,D_2) = \{(R_1,R_2): R_i \geq \frac{1}{2}\log^+[(1-\rho^2+\rho^2 2^{-2R_j})\frac{\sigma_{y_i}^2}{D_i}]\}, i,j=1,2, i\neq j,$$

$$\hat{\mathcal{R}}_{12}^{BT}(D_1,D_2) = \{(R_1,R_2): R_1+R_2 \geq \frac{1}{2}\log^+[(1-\rho^2)\frac{\beta_{max}\sigma_{y_1}^2\sigma_{y_2}^2}{2D_1D_2}]\},$$

$$\beta_{max} = 1+\sqrt{1+\frac{4\rho^2 D_1 D_2}{(1-\rho^2)^2\sigma_{y_1}^2\sigma_{y_2}^2}}$$

for D_1=D_2 and $\sigma_{y1}^2=\sigma_{y2}^2$



Y_1 and Y_2 quadratic Gaussian sources with variances σ_{y1}^2 and σ_{y2}^2

$\rho = \frac{E[Y_1Y_2]}{\sigma_{y_1}\sigma_{y_2}}$. - correlation coefficient

$$\frac{1}{n}\sum_{i=1}^{n} E[d(Y_{1,i},\hat{Y}_{1,i})] \leq D_1+\epsilon,$$

$$\frac{1}{n}\sum_{i=1}^{n} E[d(Y_{1,i},\hat{Y}_{1,i})] \leq D_2+\epsilon$$

- Distortion constraints

Quadratic Gaussian Indirect MT Source Coding with MSE Distortion

(Oohama '05)

$$\hat{\mathcal{R}}_i^{YI}(D) = \{(R_1,R_2): R_i \geq \frac{1}{2}\log^+[\frac{\sigma_x^4(2^{-2R_j}\sigma_x^2+\sigma_{n_i}^2)^2(\sigma_x^2+\sigma_{n_i}^2)^{-1}}{2^{-2R_j}\sigma_x^4(D-\sigma_{n_j}^2)+\sigma_x^2 D(\sigma_{n_1}^2+\sigma_{n_2}^2)-\sigma_{n_1}^2\sigma_{n_2}^2(\sigma_x^2-D)}]\},$$
$$i,j=1,2, i\neq j,$$
$$\hat{\mathcal{R}}_{12}^{YI}(D) = \{(R_1,R_2): R_1+R_2 \geq \frac{1}{2}\log^+[\frac{4\sigma_x^2}{\sigma_{n_1}^2\sigma_{n_2}^2 D(\frac{1}{\sigma_x^2}-\frac{1}{D}+\frac{1}{\sigma_{n_1}^2}+\frac{1}{\sigma_{n_2}^2})^2}]\}.$$

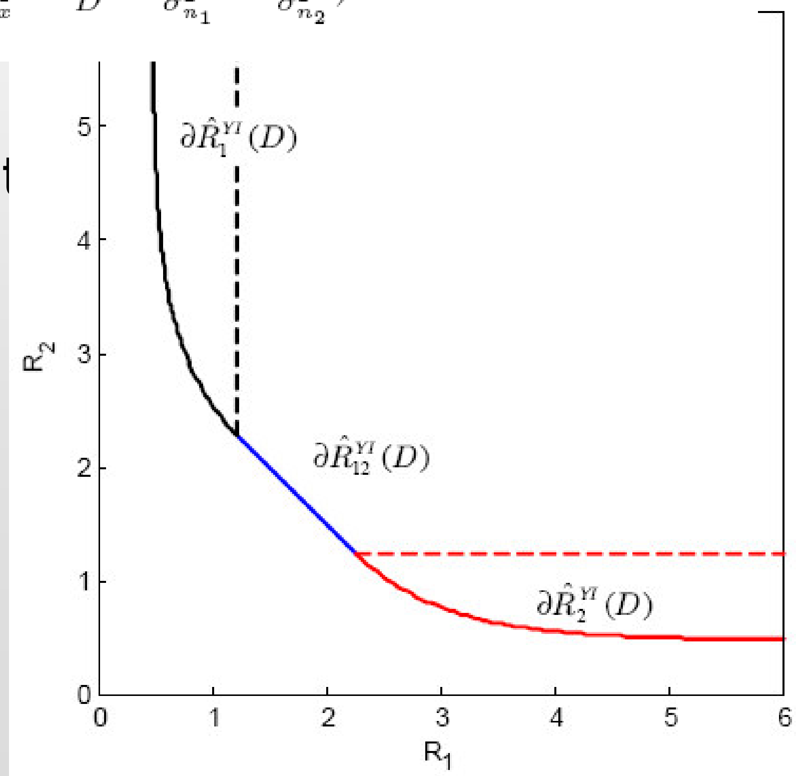
X, N_1, N_2 are zero-mean mutually independent Gaussian random variables with variances σ_x^2, σ_{n1}^2 and σ_{n2}^2.

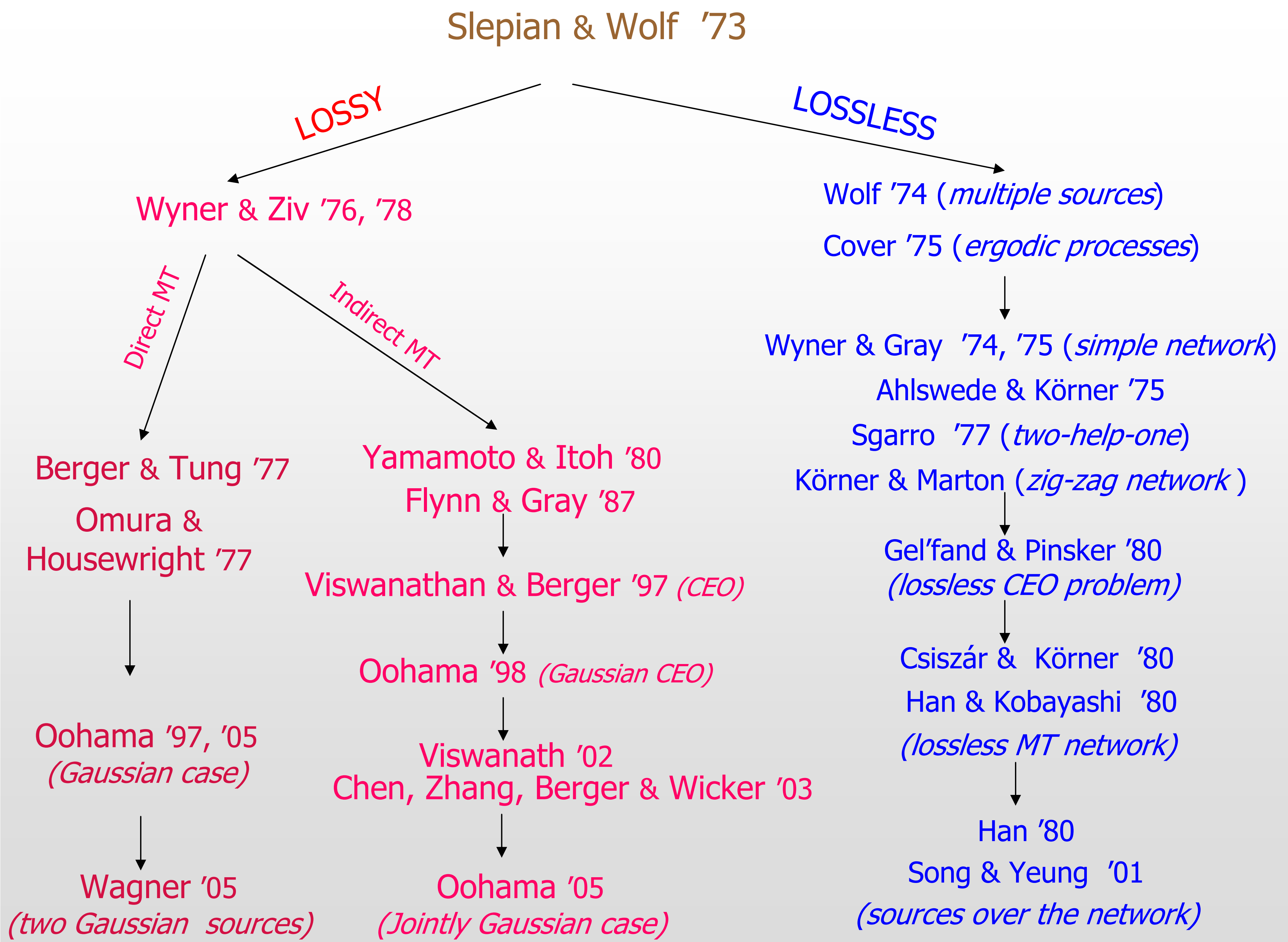
Two noisy observations:

$$Y_1 = X + N_1; \quad Y_2 = X + N_2;$$

$$\frac{1}{n}\sum_{i=1}^{n} E[d(X_i,\hat{X}_i)] \leq D+\epsilon$$

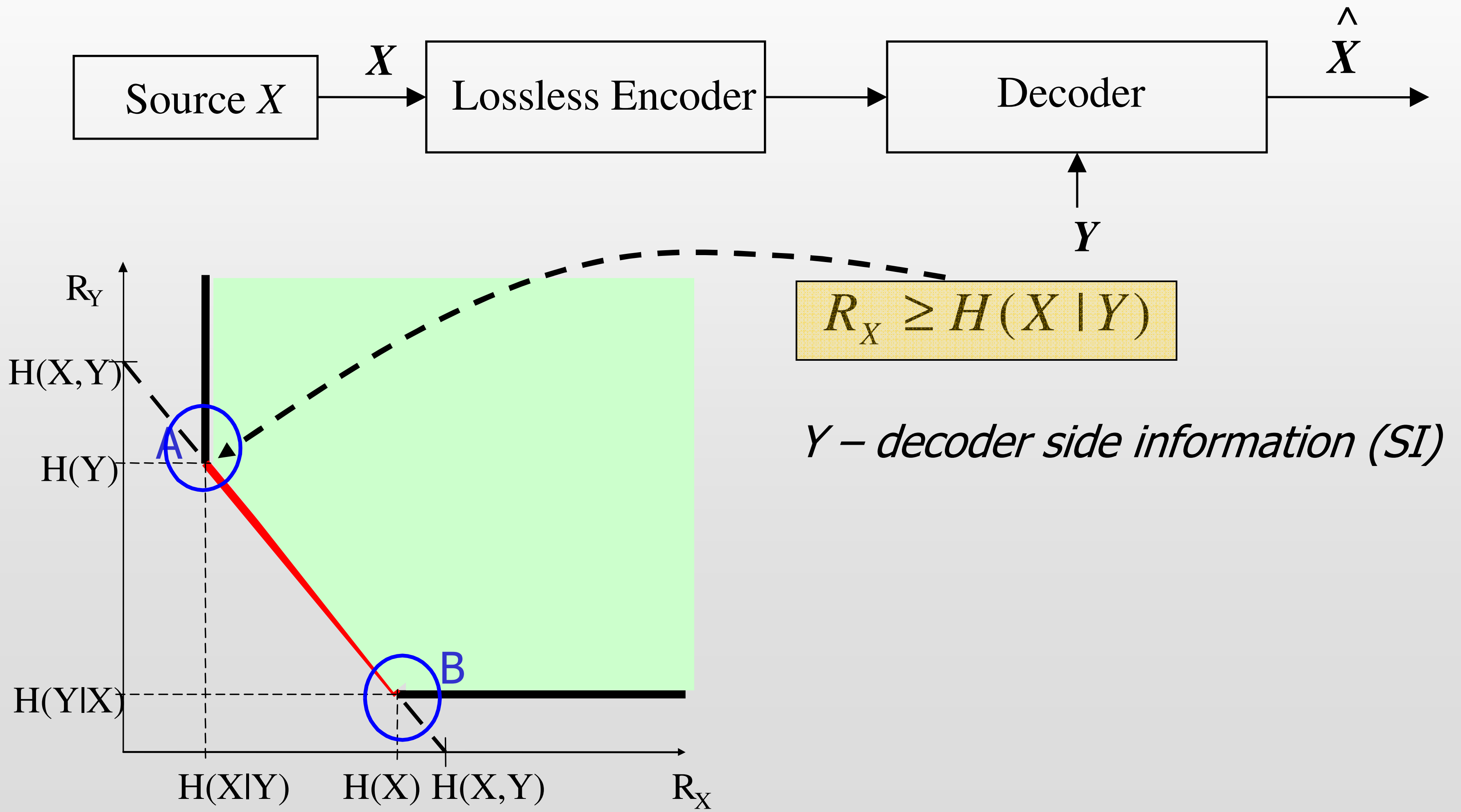
- Distortion constraint




Slepian & Wolf '73
LOSSY
LOSSLESS
Wyner & Ziv '76, '78
Direct MT
Indirect MT
Berger & Tung '77
Omura & Housewright '77
Oohama '97, '05
(Gaussian case)
Wagner '05
(two Gaussian sources)
Yamamoto & Itoh '80
Flynn & Gray '87
Viswanathan & Berger '97 (CEO)
Oohama '98 (Gaussian CEO)
Viswanath '02
Chen, Zhang, Berger & Wicker '03
Oohama '05
(Jointly Gaussian case)
Wolf '74 (multiple sources)
Cover '75 (ergodic processes)
Wyner & Gray '74, '75 (simple network)
Ahlswede & Körner '75
Sgarro '77 (two-help-one)
Körner & Marton (zig-zag network)
Gel'fand & Pinsker '80
(lossless CEO problem)
Csiszár & Körner '80
Han & Kobayashi '80
(lossless MT network)
Han '80
Song & Yeung '01
(sources over the network)

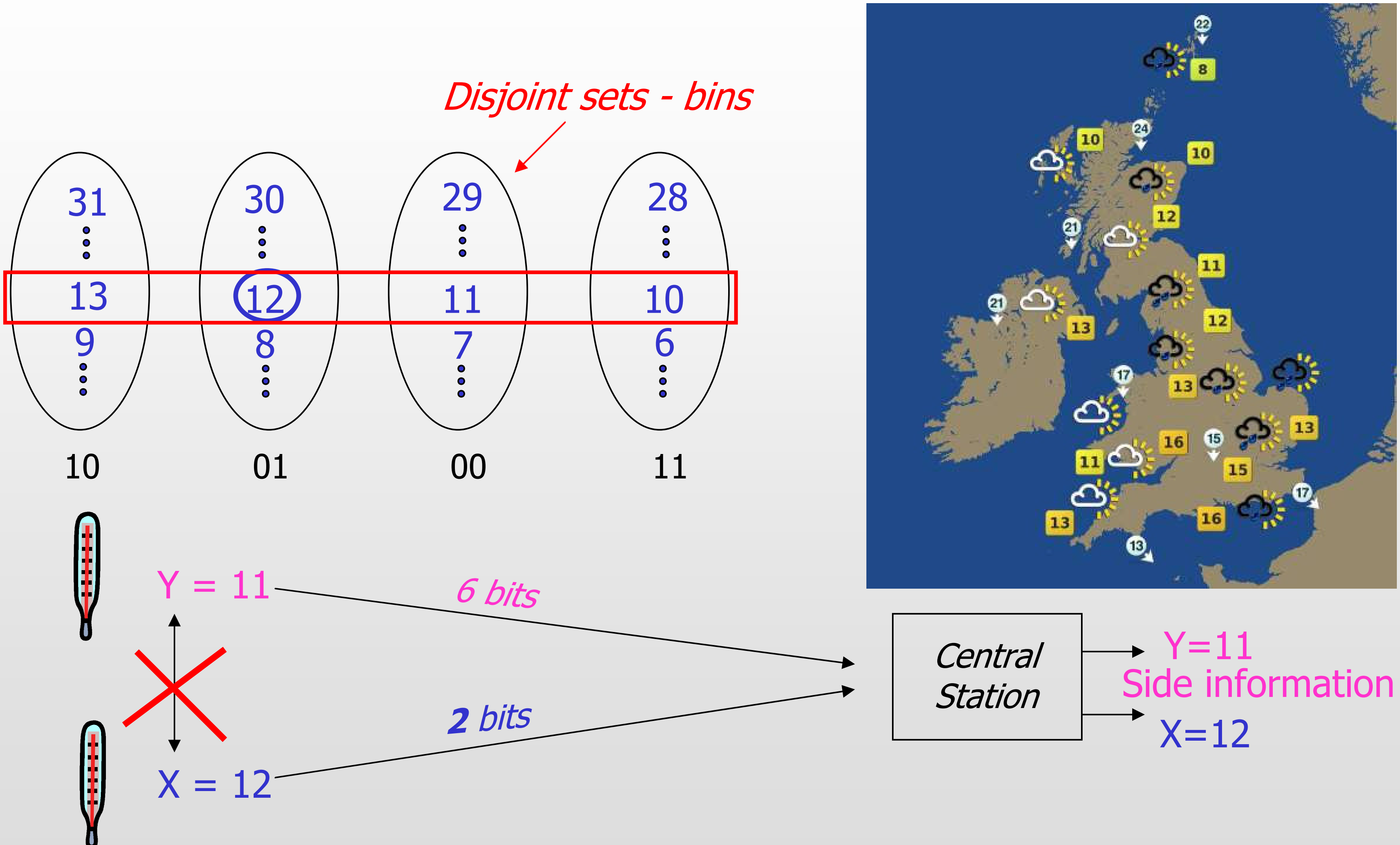
DSC: Code Design Guidelines and Coding Solutions

Source Coding with Decoder Side Information



Y – decoder side information (SI)

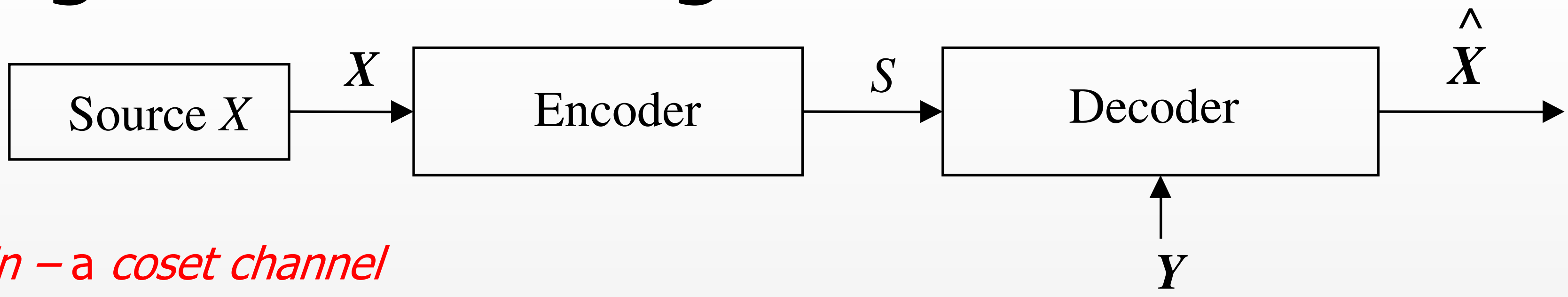
Compression of Correlated Sources

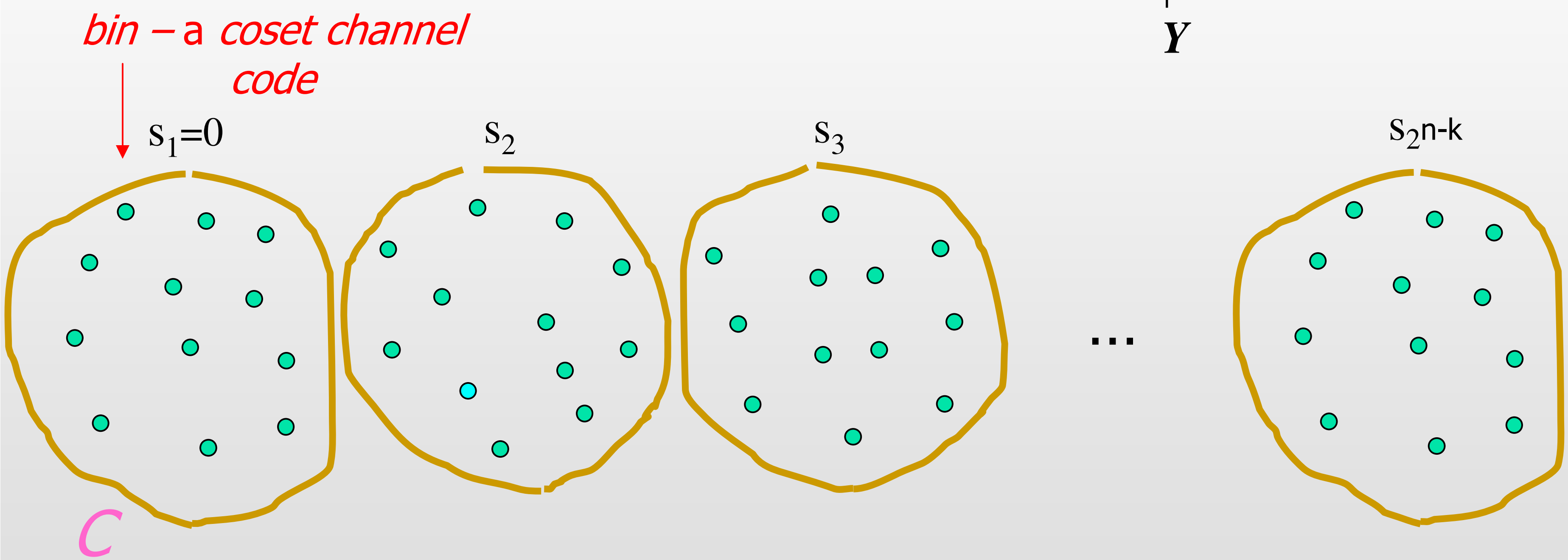


Correlation: $X=[Y-1, Y, Y+1, Y+2]$

Slepian-Wolf theorem: Still two bits are needed for compressing X!

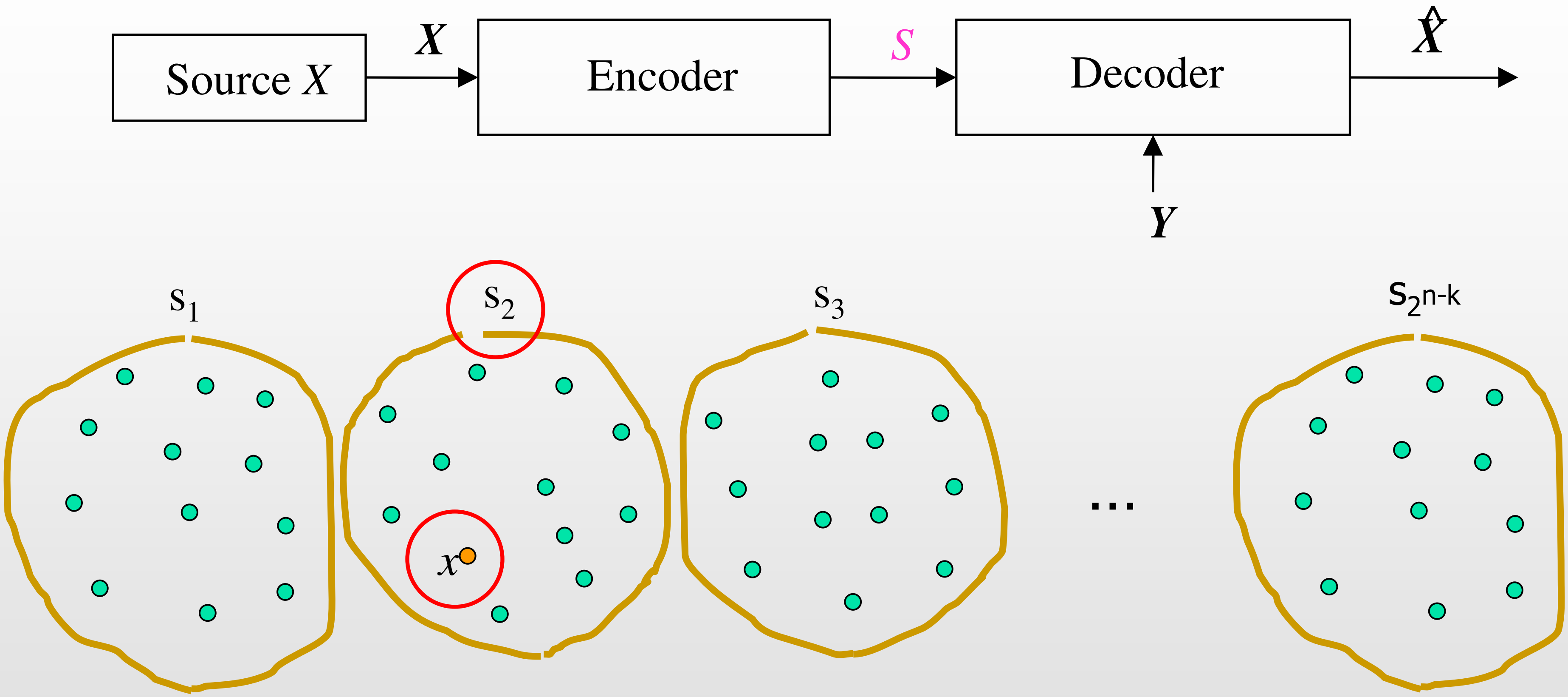
Channel Codes for Compression: Algebraic Binning *(Wyner '74, Zamir et al. '02)*





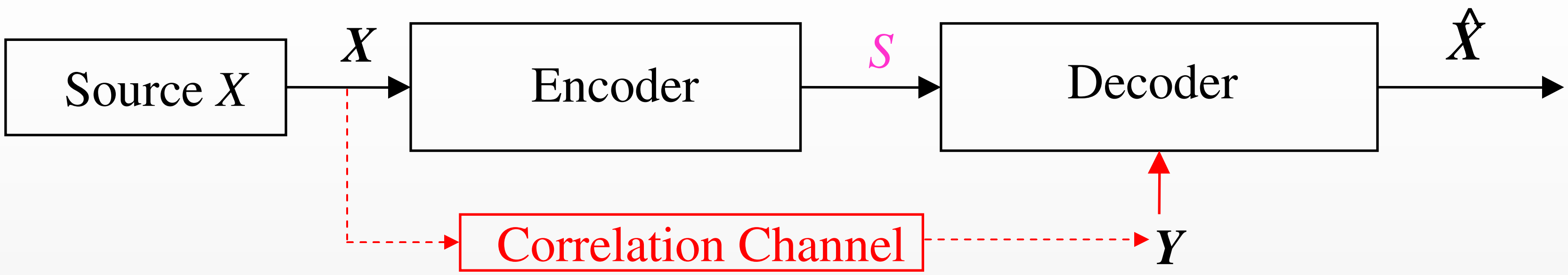
- Distribute all possible realizations of X (of length n) into bins
- Each bin is a *coset* of an (n,k) linear channel code C, with parity-check matrix H of size $(n,n-k)$ indexed by a *syndrome* s

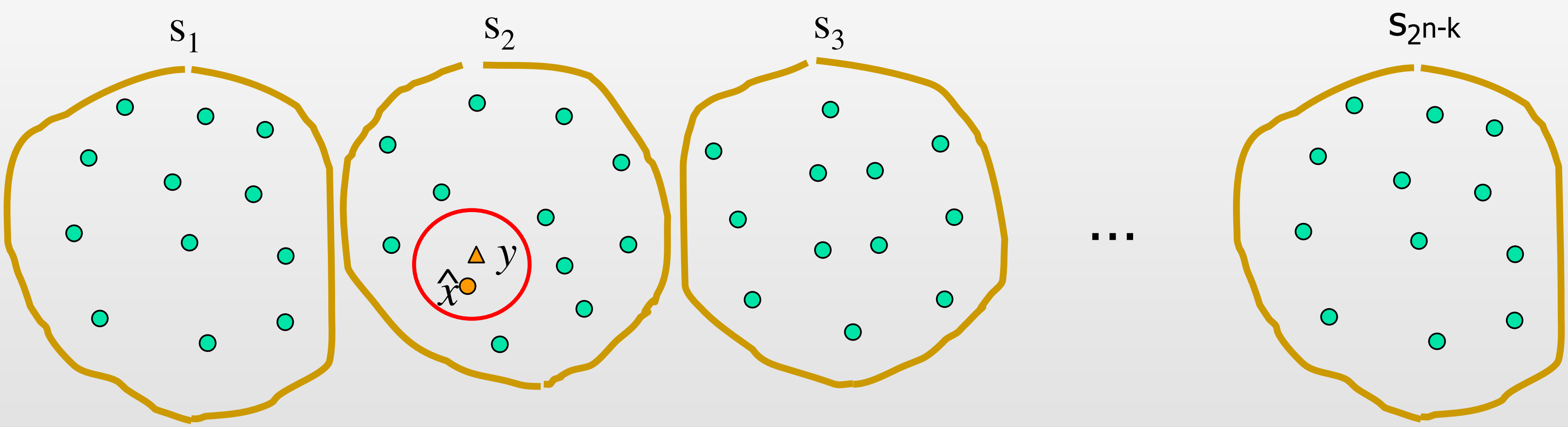
Encoding



- Encoding: For an input x, form a syndrome $s=xH^T$
- Send the resulting syndrome (s_2) to the decoder

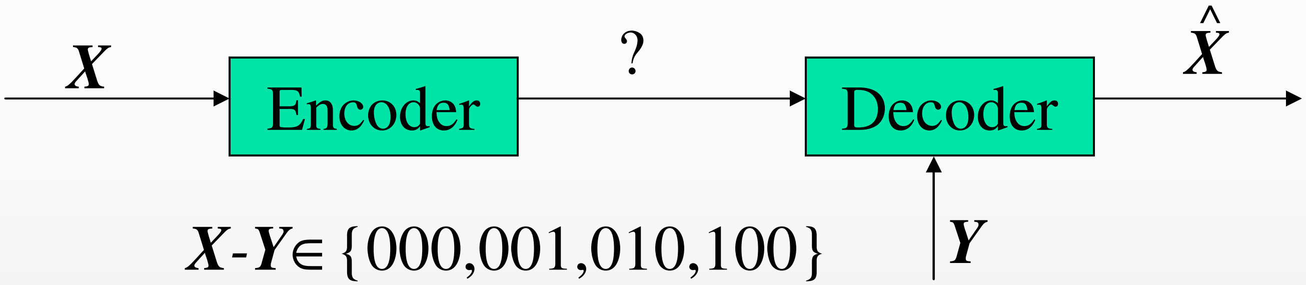
Decoding





- Interpret *y* as a *noisy version* (output of virtual communication channel called *correlation channel*) of *x*
- Find a codeword of the coset indexed by *s* closest to *y* by performing conventional channel decoding

An Example



$X\text{-}Y \in \{000,001,010,100\}$

- Setup
  - ***X*** and ***Y*** are of length 3 bits
  - ***X*** and ***Y*** differ at most in one position ($d_H(\boldsymbol{X},\boldsymbol{Y}) \leq 1$)
- If ***Y*** is also given to the encoder, obviously we can compress ***X*** with 2 bits

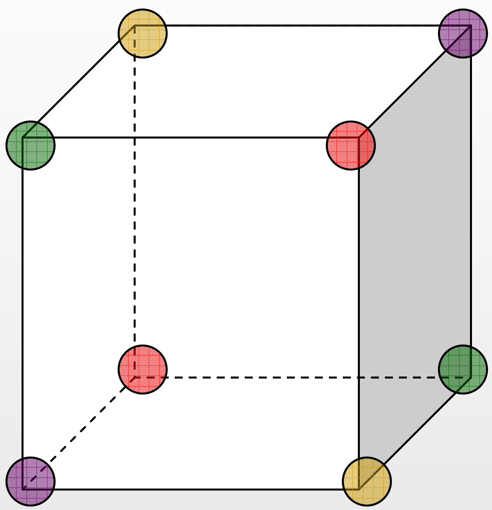
Question: How to do SWC of ***X*** given ***Y*** ?

Or: Given ***Y*** at the decoder, how to compress ***X*** ?

(Pradhan & Ramchandran '99)

Solution:

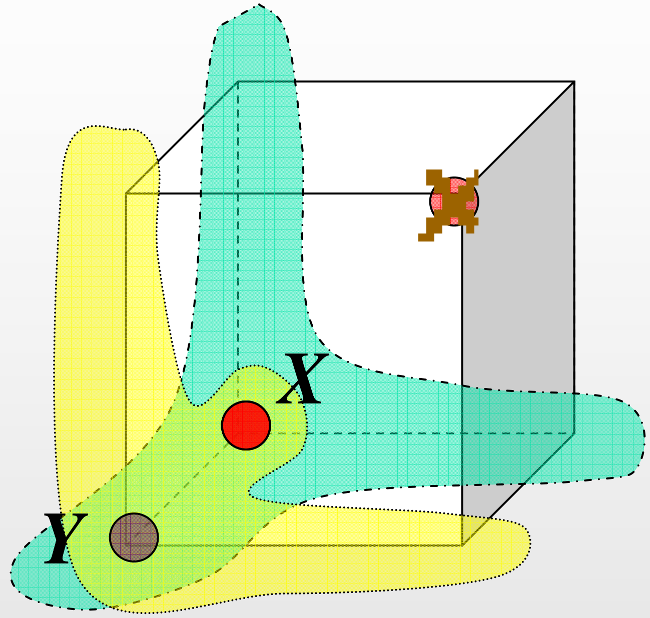
- Let **S**={S_{00},S_{01},S_{10},S_{11}}
  - S_{00}={000,111}
  - S_{01}={001,110}
  - S_{10}={010,101}
  - S_{11}={100,011}



- The encoder can transmit the index of the bin containing X using 2 bits
- With the help of Y, the decoder can recover X correctly

Example

- Assume $\boldsymbol{X}$=000
- Since S_{00} contains $\boldsymbol{X}$, the encoder transmits 00 with 2 bits
- Assume $\boldsymbol{Y}$=001
- With $\boldsymbol{Y}$, the decoder knows $\boldsymbol{X}$=000 instead of 111



$S_{00}=\{000,111\}$

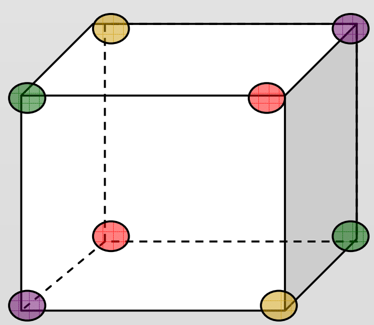
$x \in \{000,111\}: s = xH^T = 00$

$x \in \{010,101\}: s = xH^T = 10$

$x \in \{001,110\}: s = xH^T = 01$

$x \in \{011,100\}: s = xH^T = 11$

$$H = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$



$s_{00}=\{000,111\}$

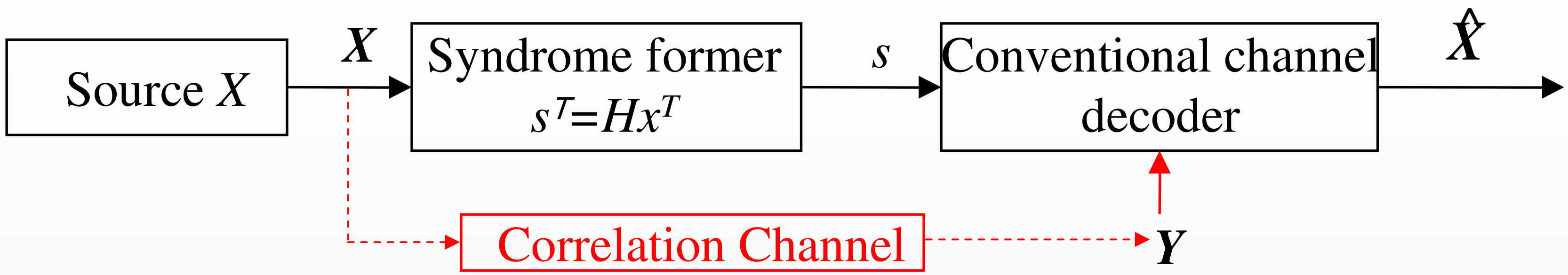
$s_{01}=\{001,110\}$

$s_{10}=\{010,101\}$

$s_{11}=\{100,011\}$

Another Example: Hamming Code

- Two uniformly distributed sources (X and Y)
- Length: $n=7$ bits
- Correlation: $d_H(X,Y) \le 1$ bit
- Slepian-Wolf bound: $R_X+R_Y=nH(X,Y)=10$ bits
- Asymmetric SW coding:
  $R_Y=nH(Y)=7$ *bits*, $R_X=nH(X|Y)=3$ *bits*



Systematic *(7,4) Hamming* code *C (can correct one bit error)*

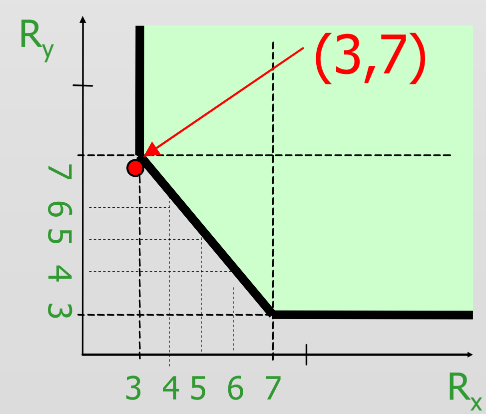
$$G^T = [I_4 \mid P^T] = \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$$H = \left[\begin{array}{c|c} P_{3x4} & I_3 \end{array}\right]$$

Suppose that realizations are:

$x^T = [\ u_1^T \quad u_2^T\ ] = [\ 0010 \quad 110\ ]$

$y^T = [\ v_1^T \quad v_2^T\ ] = [\ 0110 \quad 110\ ]$



Encoding:

$$s_x = Hx = \begin{bmatrix} Pu_1 \oplus u_2 \end{bmatrix} = [0\ 0\ 1]^T$$ ← 3 bits!

$$y = [0\ 1\ 1\ 0\ 1\ 1\ 0]^T$$ ← 7 bits!

Decoding:

1) Form *7*-length vectors :

$$t_1 = \begin{bmatrix} 0_{4x1} \\ \hline Pu_1 \oplus u_2 \end{bmatrix} = [\underbrace{0000}_{\text{padded}}\ 001]^T$$

$$t_2 = y = \begin{bmatrix} v_1^T \\ \hline v_2^T \end{bmatrix} = [0110\ 110]^T$$

2) Find codeword c in C closest to $t = t_1 \oplus t_2 = [0110 \quad 111]$

$$c = t_1 \oplus t_2 \oplus x \oplus y = [0010 \; 111]$$

$$[\, u_1^T \quad u_1^T P^T] = u_i^T G$$

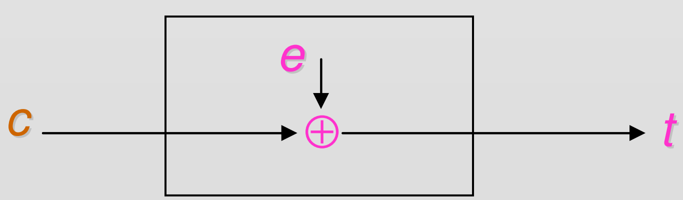
a codeword!

$$t = (t_1 \oplus t_2 \oplus x \oplus y) \oplus (x \oplus y) = c \oplus e$$

$e = x \oplus y$ – correlation noise

3) Reconstruction: $\hat{x} = u_1 G \oplus t_1 = [0010 \; 110] = x$

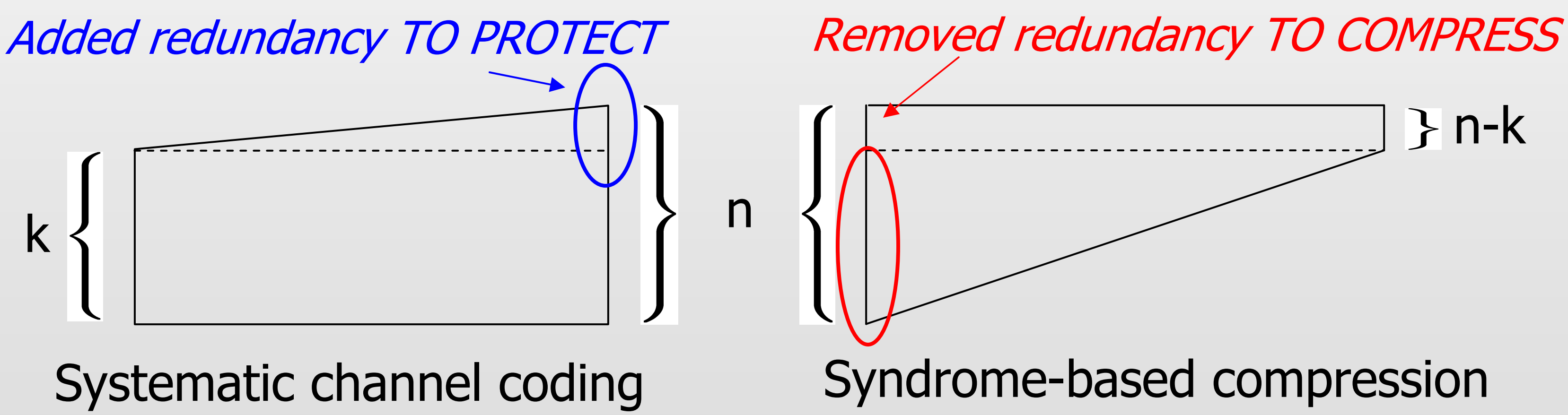
$$[\, u_1 \quad u_2]$$



Hypothetical correlation channel

Asymmetric Syndrome Concept

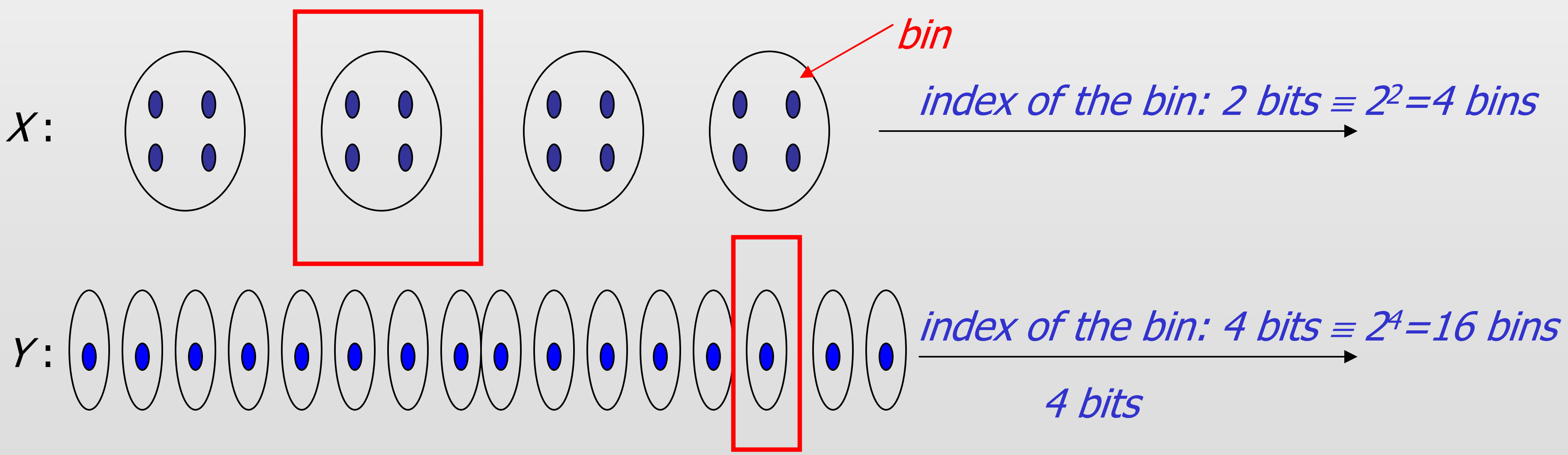
- Take an (n,k) linear channel code to partition the space of n-length source X into cosets indexed by different syndromes (of length $n-k$ bits)



Compression rate: $R=(n-k)/n$

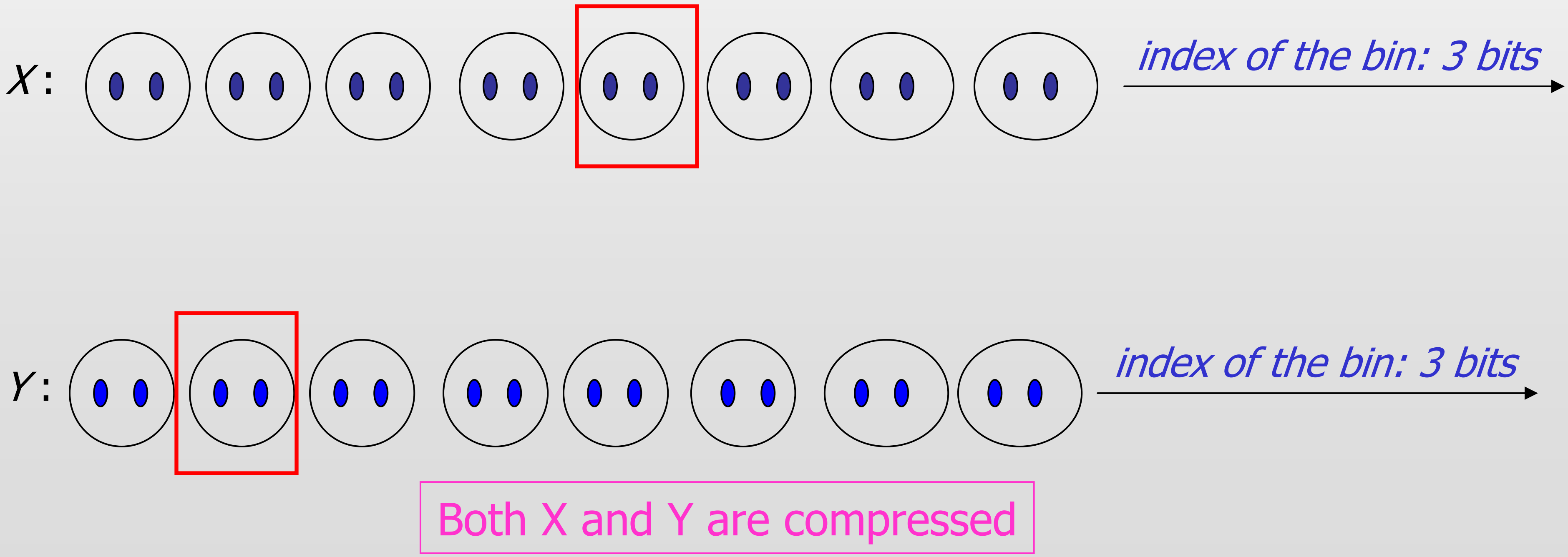
Asymmetric Binning

Example: Suppose that X and Y are i.i.d. uniform sources of length n=4 bits each. Code Y at $R_y=nH(Y)=4$ bits and X at $R_x=nH(X/Y)=2$ bits. Total transmission rate $R_Y + R_X=4+2=6$ bits



From Asymmetric to Symmetric Binning

Code X and Y at $R_x=R_y=3$ bits
Total transmission rate $R_x + R_y=3+3=6$ bits



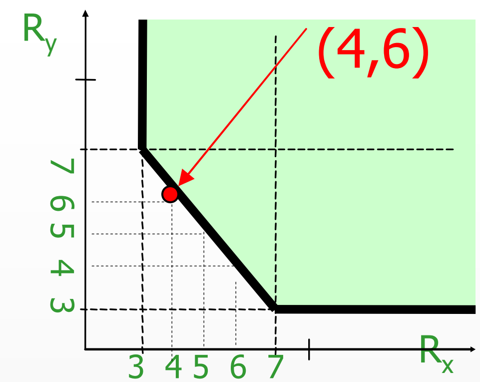
Implementation *(Pradhan & Ramchandran '05)*

- Generate a channel code C and partition it into nonoverlapping subcodes C_1 and C_2
- C_2 : set of cosets representatives of C_1 in C
- Assign subcode C_i to encoder i , $i=1,2$

Hamming Code Example Revisited

- Two uniformly distributed sources (X and Y)
- Length: $n=7$ bits
- Correlation: $d_H(x,y) \leq 1$ bit
- Slepian-Wolf bound: $R_X+R_Y=nH(X,Y)=10$ bits
- Asymmetric coding: $R_Y=nH(Y)=7$ *bits*, $R_X=nH(X|Y)=3$ *bits*
- Non-asymmetric coding: $R_Y=6$ *bits*, $R_X=4$ *bits*
- Symmetric coding: $R_X=R_Y=5$ *bits!*

Systematic *(7,4) Hamming* code *C*



P1 P2

$$H_{3x7}=[P \,\vdots\, I_3]=\begin{bmatrix} 1\;0\;1 & 1 & 1\;0\;0 \\ 0\;1\;1 & 1 & 0\;1\;0 \\ 1\;1\;1 & 0 & 0\;0\;1 \end{bmatrix}$$

$$H1_{4x7}=\begin{bmatrix} O_{1x3} & 1 & O_{1x3} \\ P1_{3x3} & O_{3x1} & I_3 \end{bmatrix} \qquad H2_{6x7}=\begin{bmatrix} I_3 & O_{3x1} & O_{3x3} \\ O_{3x3} & P2_{3x1} & I_3 \end{bmatrix}$$

$$x^T = [\, u_1^T \;\; u_2^T \;\; u_3^T \,] = [\, 001 \;\; 0 \;\; 110 \,]$$
$$y^T = [\, v_1^T \;\; v_2^T \;\; v_3^T \,] = [\, 011 \;\; 0 \;\; 110 \,]$$

Encoding:

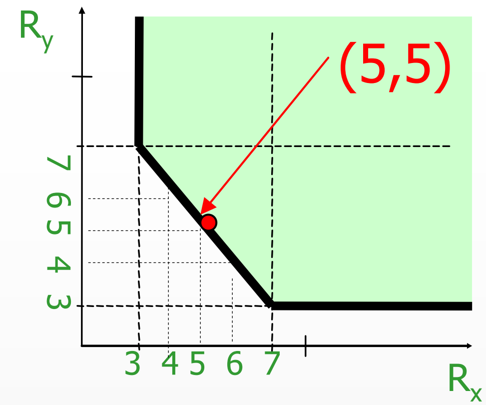
$$s_1 = H1\, x = \begin{bmatrix} u_2 \\ P1 u_1 \oplus u_3 \end{bmatrix} = [0 \;\; 0\;0\;1]^T$$

(coded: 0 0 1) ← 4 bits!

$$s_2 = H2\, y = \begin{bmatrix} v_1 \\ P2 v_2 \oplus v_3 \end{bmatrix} = [0\;1\;1 \;\; 1\;1\;0]^T$$

(coded: 1 1 0) ← 6 bits!

Systematic *(7,4) Hamming* code C



P1 P2

$$H_{3x7}=[P \,|\, I_3]=\left[\begin{array}{cc|cc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

$$H1_{5x7}=\left[\begin{array}{c|c|c} O_{2x2} & I_2 & O_{2x3} \\ P1_{3x2} & O_{3x2} & I_3 \end{array}\right] \qquad H2_{5x7}=\left[\begin{array}{c|c|c} I_2 & O_{2x2} & O_{2x3} \\ O_{3x2} & P2_{3x2} & I_3 \end{array}\right]$$

$x^T = [\, u_1^T \;\; u_2^T \;\; u_3^T \,] = [\, 00 \;\; 10 \;\; 110 \,]$

$y^T = [\, v_1^T \;\; v_2^T \;\; v_3^T \,] = [\, 01 \;\; 10 \;\; 110 \,]$

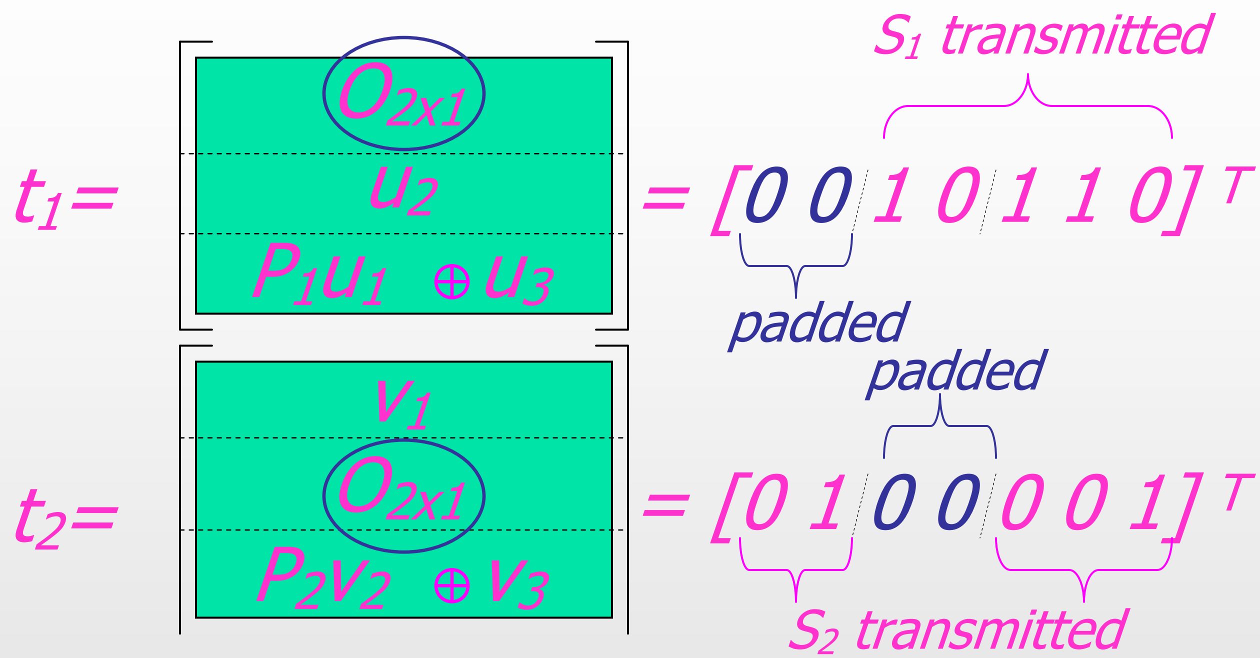
Encoding:

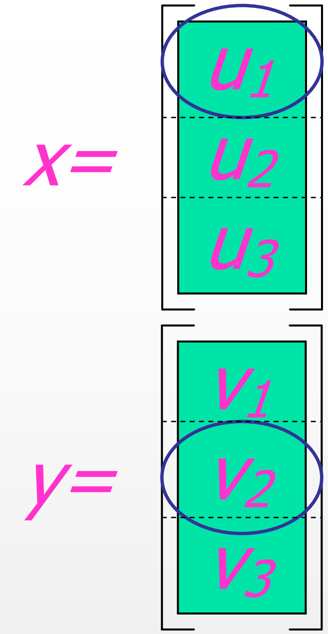
$$s_1 = H1\, x = \left[\begin{array}{c} u_2 \\ \hline P1 u_1 \oplus u_3 \end{array}\right] = [1\;0\;\underbrace{1\;1\;0}_{\text{coded}}]^T \quad \longleftarrow \text{5 bits!}$$

$$s_2 = H2\, y = \left[\begin{array}{c} v_1 \\ \hline P2 v_2 \oplus v_3 \end{array}\right] = [0\;1\;\underbrace{0\;0\;1}_{\text{coded}}]^T \quad \longleftarrow \text{5 bits!}$$

Decoding:

1) Form *7*-length vectors :





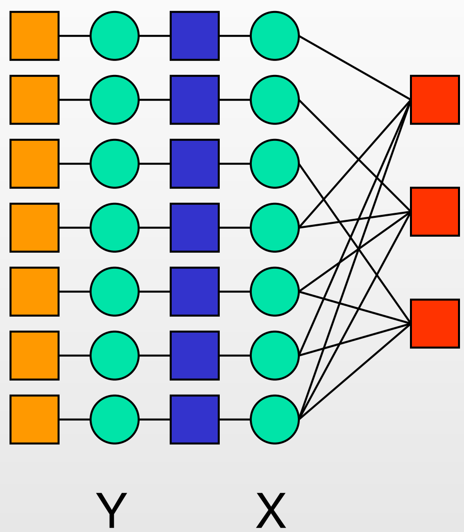
2) Note that $c = x \oplus y \oplus t_1 \oplus t_2 = [0010\ 111\ ]^T$ is a codeword

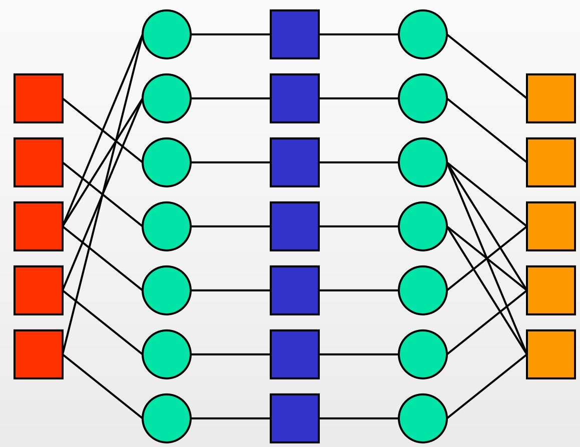
$[\ u_1^T\ v_2^T\ ]\ [u_1^T\ v_2^T]P^T]$

and $x \oplus y$ is small. We can decode $[u_1^T\ v_2^T]$ from $t_1 \oplus t_2$

3) u_3 and v_3 can be found by adding the third parts of t_1 and t_2 by P_1u_1 and P_2v_2

Factor Graph Representation (Hamming Code Example)





$$H_1 = I_{7\times7} \qquad H_2 = \begin{bmatrix} 1001011 \\ 0101101 \\ 0010111 \end{bmatrix}$$

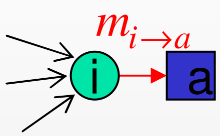
$$H_1 = \begin{bmatrix} 0010000 \\ 0001000 \\ 1100100 \\ 0100010 \\ 1000001 \end{bmatrix} \qquad H_2 = \begin{bmatrix} 1000000 \\ 0100000 \\ 0010100 \\ 0011010 \\ 0011001 \end{bmatrix}$$

What if we consider SW decoding as an inference problem and decode it with belief propagation (BP) directly?

Direct SW Decoding with BP (*Schonberg et al'04, Cheng '09*)

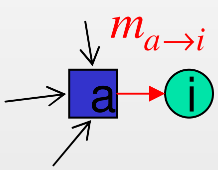
Variable node update

$$m_{i\to a}(x_i) = \prod_{b\in N(i),\ a} m_{b\to i}(x_i)$$



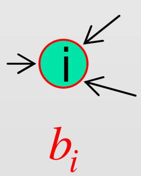
Factor node update

$$m_{a\to i}(x_i) = \sum_{x_a,\ x_i} f_a(x_a) \prod_{j\in N(a),\ i} m_{j\to a}(x_j)$$



Belief update

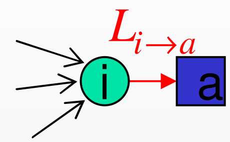
$$b_i(x_i) = \prod_{a\in N(i)} m_{a\to i}(x_i)$$



Direct SW Decoding with BP

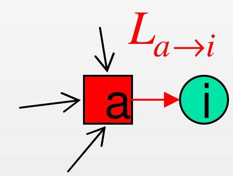
Variable node update

$$L_{i\to a} = \sum_{b\in N(i),\ \ a} L_{b\to i}$$



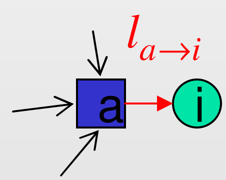
Check node update

$$L_{a\to i} = 2(1-2s(a))\tanh^{-1}\left(\prod_{j\in N(a),\ \ i} \tanh\left(\frac{L_{j\to a}}{2}\right)\right)$$



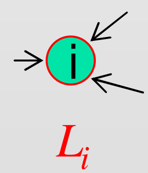
Correlation node update

$$l_{a\to i} = \frac{p+(1-p)l_{j\to a}}{(1-p)+pl_{j\to a}}$$



Belief update

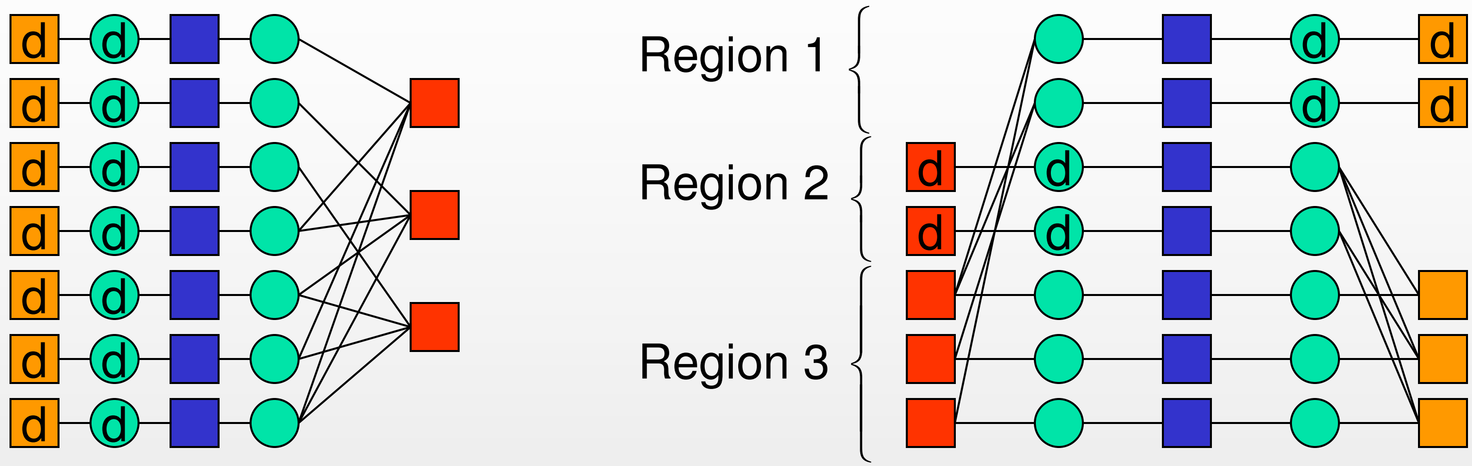
$$L_i = \sum_{a\in N(i)} L_{a\to i}$$



l : likelihood
L : log-likehood

BP is capable of recovering all pairs of *X* and *Y* with less than 1 bit difference

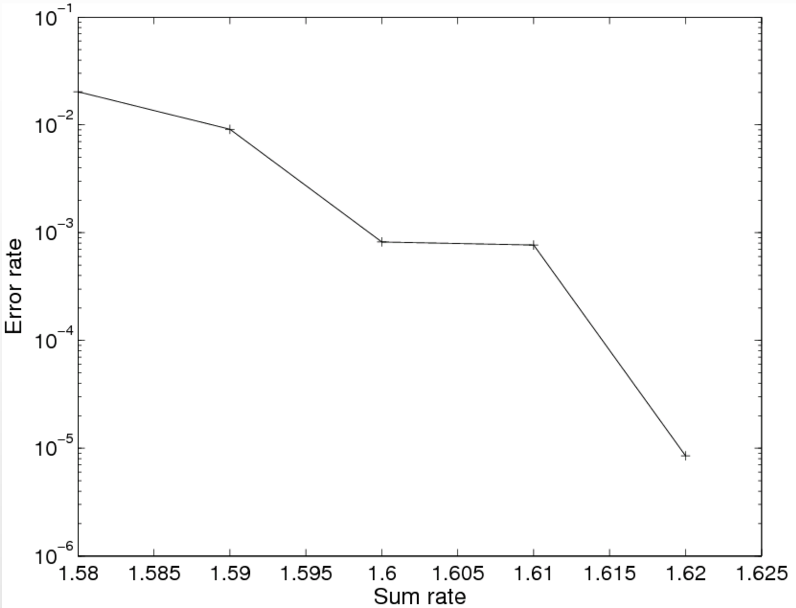
Lesson Learned from Hamming Code

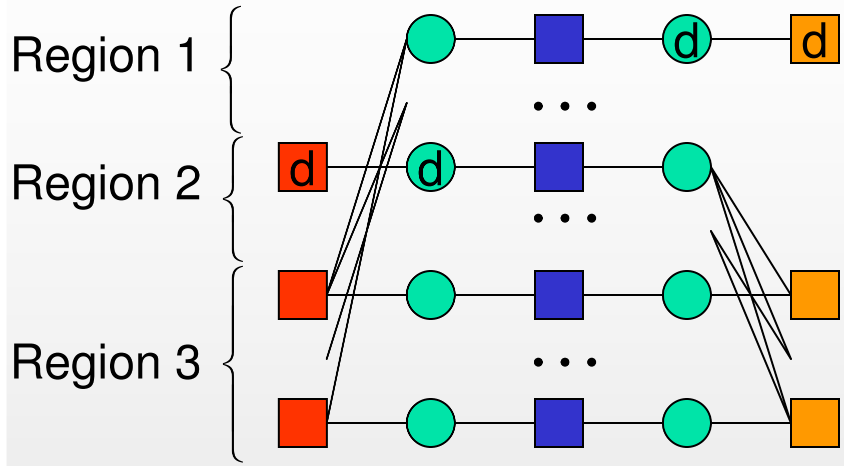


d: doped node, variable sent directly to decoder

- No pair of variable nodes are doped on both sides (obvious waste)
- A variable node on the opposite side of a doped node has higher degree (it gets more info and hence should share)
- Variable nodes that are not doped in either side has degree 1
- Region 1: doped on the right; region 2: doped on the left; region 3: doped on both side

Result with Longer Code Length





n=10,000, $r_1=r_2$, p=0.1, H(X,Y)=1.47

- No pair of variable nodes are doped on both sides (obvious waste)
- A variable node on the opposite side of a doped node has higher degree = 4 (it gets more info and hence should share)
- Variable nodes that are not doped in either side has degree 1
- Region 1: doped on the right; region 2: doped on the left; region 3: doped on both side

Remarks

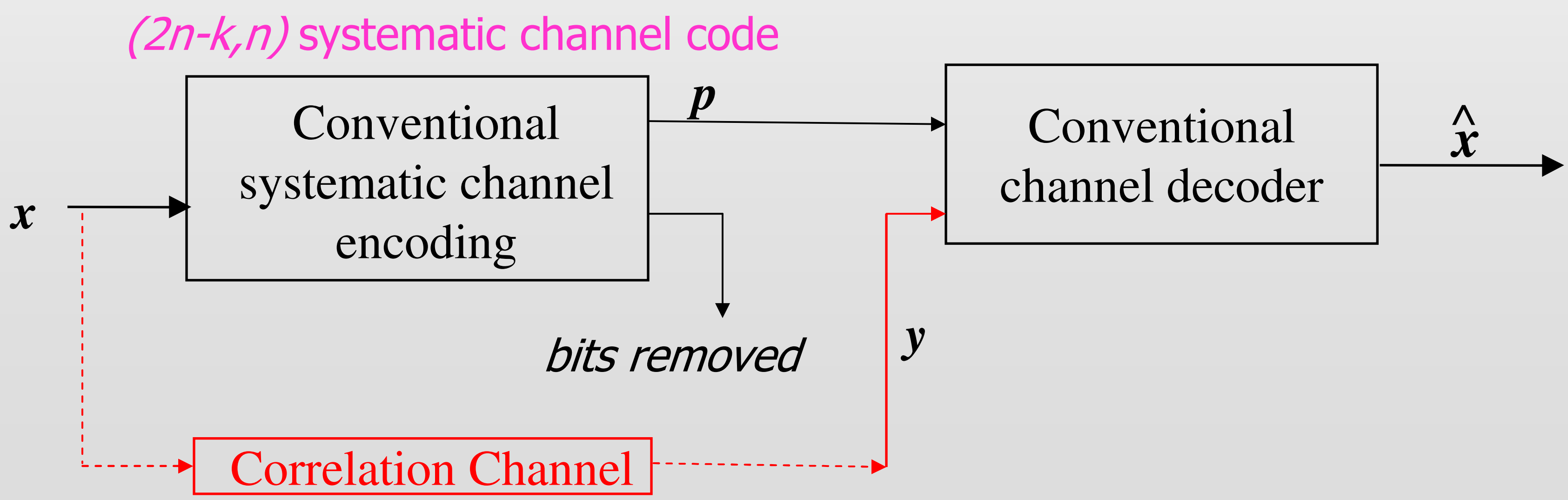
- About 0.15 bits from limit (c.f. 0.04 bits from Stankovic et al ’06). However, no actual “code design” is applied (yet)
- Relationship to channel code splitting (Pradhan et al ’05 and Stankovic et al ’06):
  - For asymmetric case, decoding is identical (both degenerate to SW decoding based on LDPC decoding)
  - However, for non-asymmetric case, decoding is completely different
- Strengths:
  - Can be applied to arbitrary correlation (does not restrict to Bernoulli)
  - Can be easily extended to arbitrary number of terminals Can be easily extended to non-binary case
- Weakness:
  - Higher complexity
  - No counterpart of EXIT chart and density evolution type of analysis (yet)

General Syndrome Concept

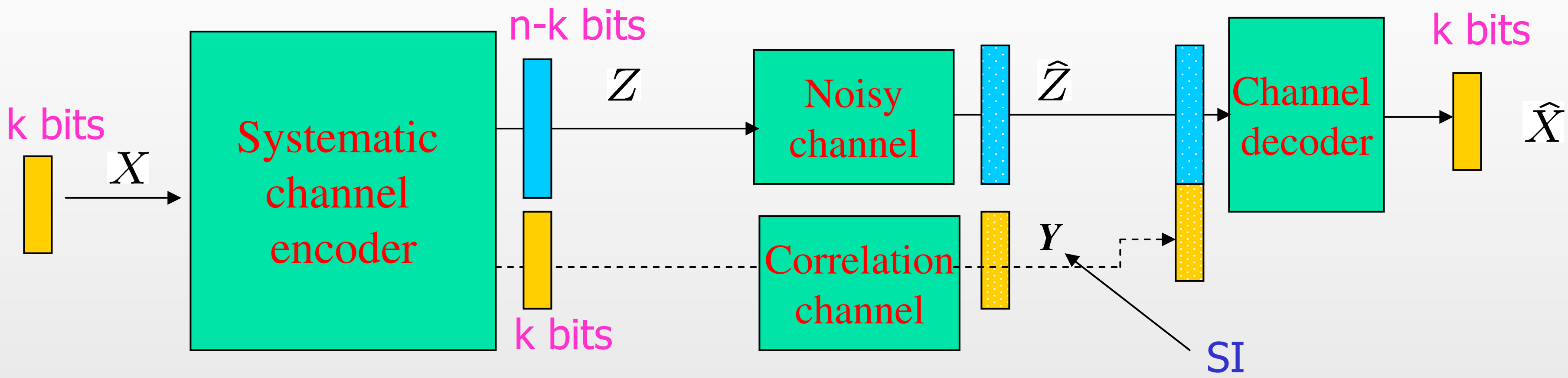
- Applicable to *all linear channel codes* (inc. turbo and LDPC codes)
- Key lies in correlation modeling: if the correlation can be modeled with a simple communication channel, existing channel codes can be used
  - SW code will be good if the employed channel code is good for a "correlation channel"
  - If the channel code approaches capacity for the "correlation channel", then the SW code approaches the SW limit
- Complexity is close to that of conventional channel coding

Parity-based Binning

- Syndrome approach: To compress an n-bit source, index each bin with a syndrome from a linear channel code (n,k)
- Parity-based approach: To compress an k-bit source, index each bin with $(n-k)$ parity bits p of a codeword of a systematic (n,k) channel code
- Compression rate: $R_X=(n-k)/k$



Parity Based SW Coding



- Efficient transmission over two different parallel channels: actual noisy channel and correlation channel between X and Y

Syndrome vs. Parity Binning

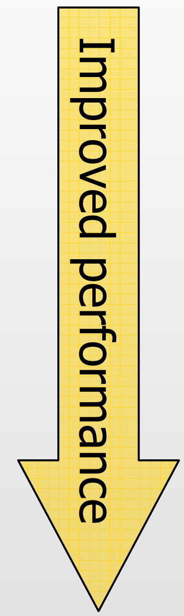
- Syndrome-based approach works better because
  - Code distance property preserved
  - For the same compression length, minimum codeword size is used
  - Good channel code -> good SW code of the same performance
- Parity-based binning has advantages
  - Good for noisy SW coding problem because in contrast to syndromes, parity bits can protect
  - Simpler (conventional encoding and decoding)
  - Simple puncturing mechanism can be used to realize different coding rates

Practical Wyner-Ziv Coding (WZC)

- Practical SW coding with algebraic binning based on channel codes for discrete sources
- In WZ coding, we are dealing with continuous space, hence syndrome approach alone will not work!
- Questions: What is a good choice of binning? How to perform coding efficiently?

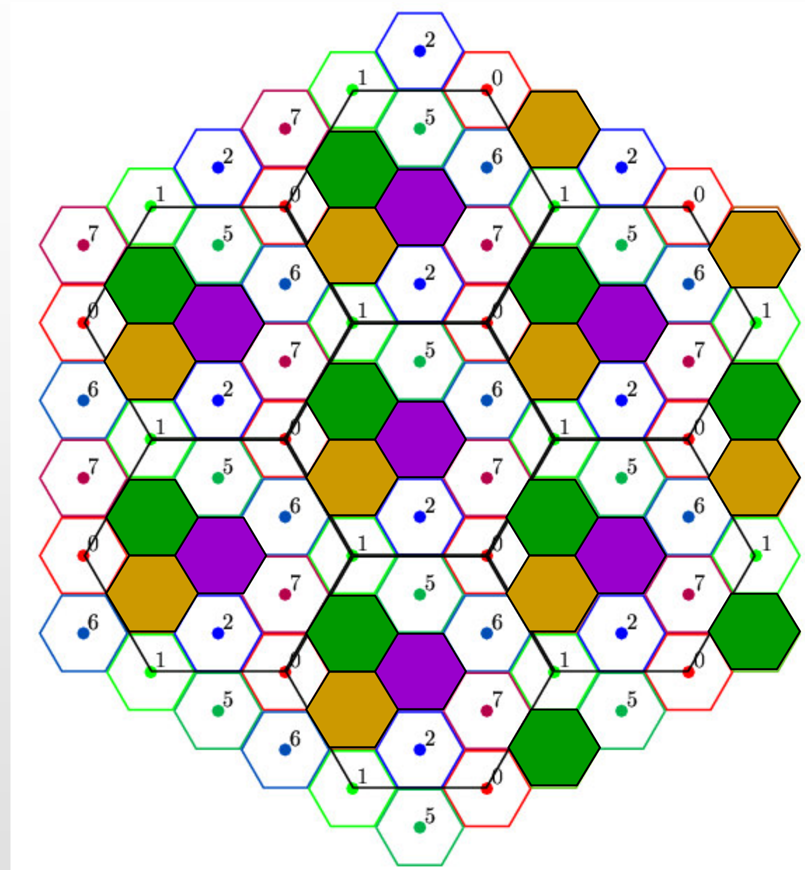
Practical WZC Solutions

- Three types of solutions proposed:
  - Nested quantization
  - Combined quantization and SW coding *(DISCUS, IT March 2003)*
  - Quantization followed by SW coding (Slepian-Wolf coded quantization - SWCQ)
- We will focus on the first and third method
- We will assume correlation model between source X and SI Y: $X=Y+Z$ with $Z \sim N(0,\sigma^2_Z)$



Nested Lattice Quantization (LQ)

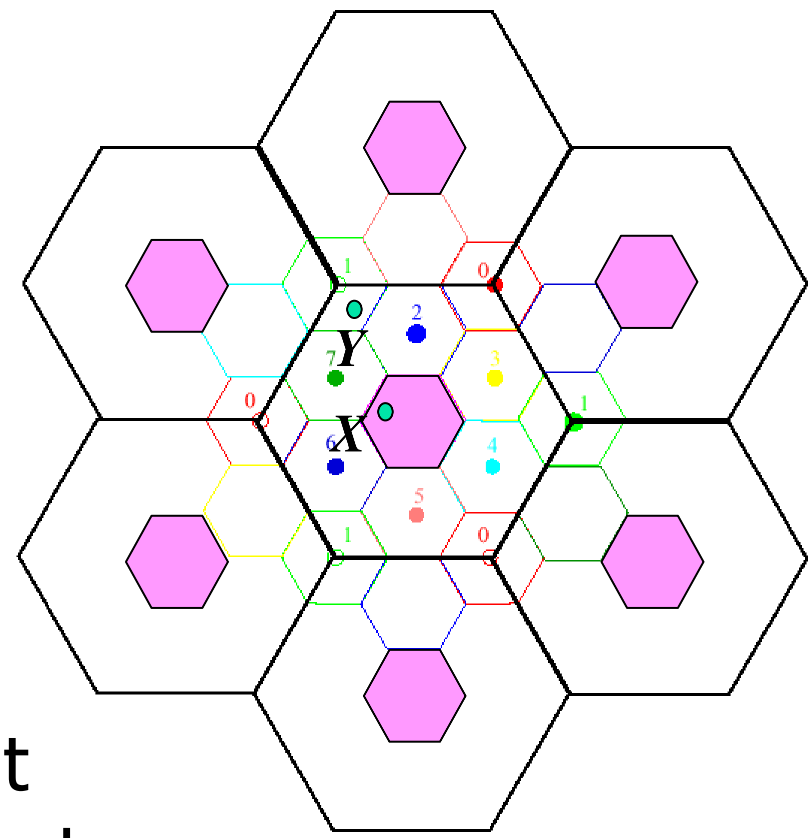
- Nested lattice
  - A (fine) lattice is partitioned into sublattices (coarse lattices)
  - A bin: the union of the original Voronoi regions of points of a sublattice



■ : bin 8
■ : bin 4
■ : bin 3

Nested Lattice Quantization

- Encoding: output index of the bin containing X
  - Quantize X using the fine lattice
  - Output the index V of the coarse lattice containing quantized lattice point

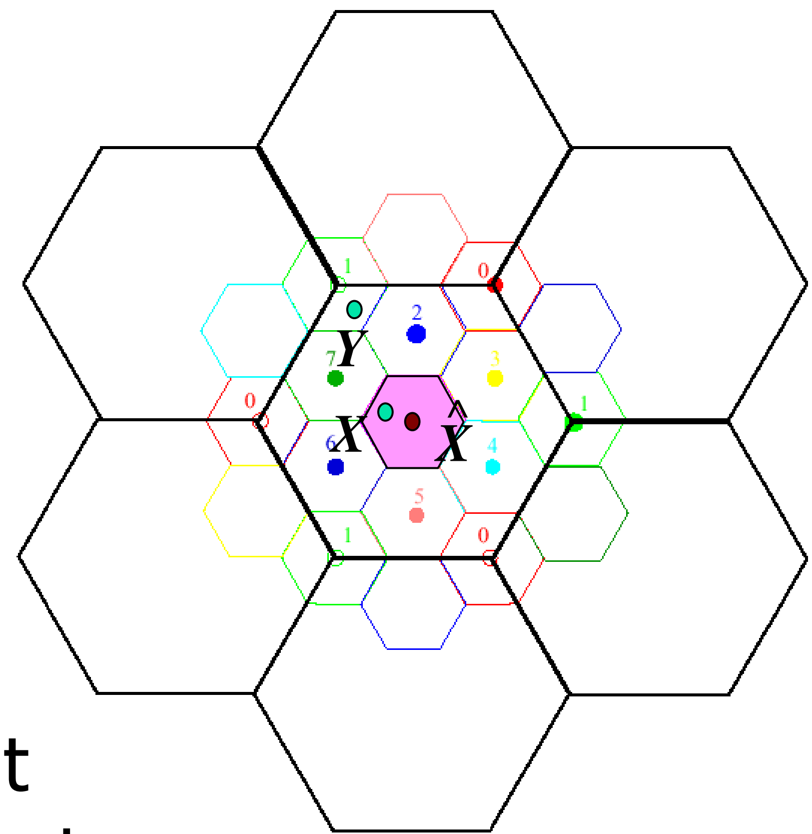


- Decoding: find lattice point of sublattice V that is closest to Y
  - Quantize Y using sublattice V

Bin index: $V = 8$

Nested Lattice Quantization

- Encoding: output index of the bin containing *X*
  - Quantize *X* using the fine lattice
  - Output the index *V* of the coarse lattice containing quantized lattice point

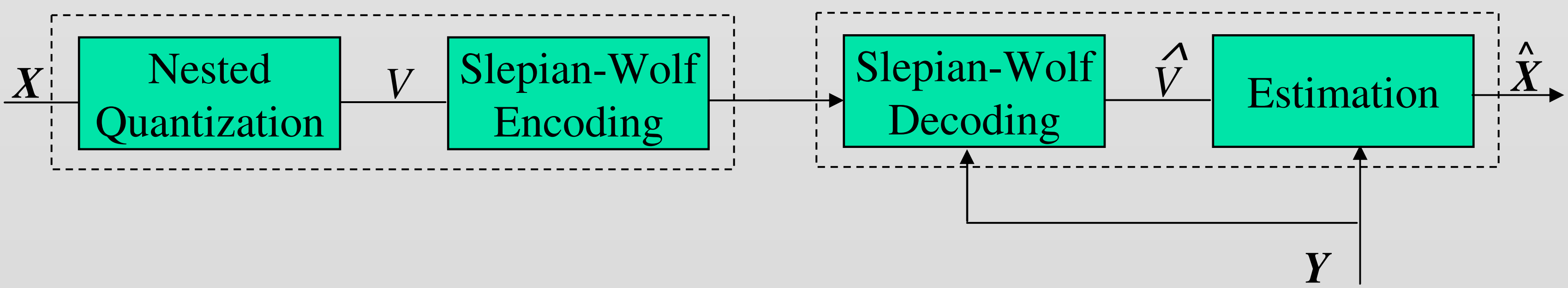


- Decoding: find lattice point of sublattice *V* that is closest to *Y*
  - Quantize *Y* using sublattice *V*

Bin index: $V = 8$

SW Coded Quantization (SWCQ)

- Nested lattice quantization is asymptotically optimal as dimensions go to infinity
  - Difficult to implement even in low dimensions
- The bin index V and the SI Y are still highly correlated, i.e., $H(V) > H(V|Y)$
  - Note that conventional lossless compression techniques (e.g., Huffman coding) are fruitless since Y is not given to the encoder
  - Use SW coding to further compress V!
- Further improvement:
  - Use estimation instead of reconstructing to a lattice point

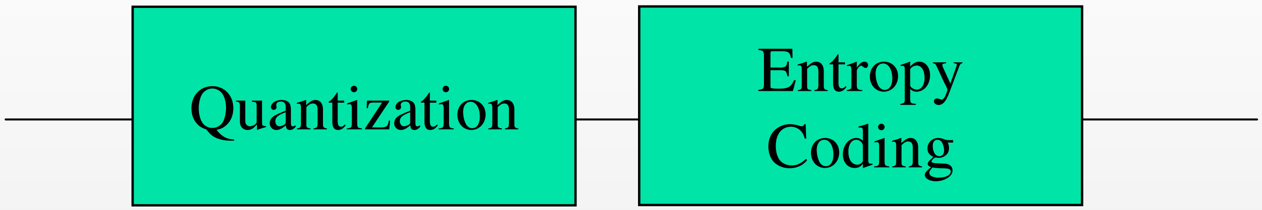


Practical SWCQ

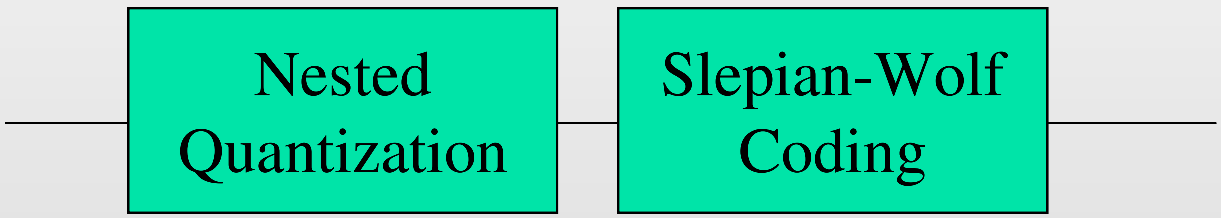
- Practical realization: (nested) quantization followed by channel coding for SW coding
- WZ coding is a source-channel coding problem
  - Quantization loss due to source coding
  - Binning loss due to channel coding
- To approach the WZ limit, one needs
  - Strong source codes (e.g., TCVQ and TCQ)
  - Near-capacity channel codes (e.g., turbo and LDPC)
- Estimation of X based on V and SI helps at low rate, thus rely more on
  - SI Y at lower rates
  - V at higher rates

WZC vs. Classic Source Coding

- Classic entropy-constrained quantization (ECQ)



- Wyner-Ziv coding (SWCQ)



  - Nested quantization: quantization with SI
  - Slepian-Wolf coding: entropy coding with SI

Classic source coding is just a special case of WZ coding
(since the SI can be assumed to be a constant)

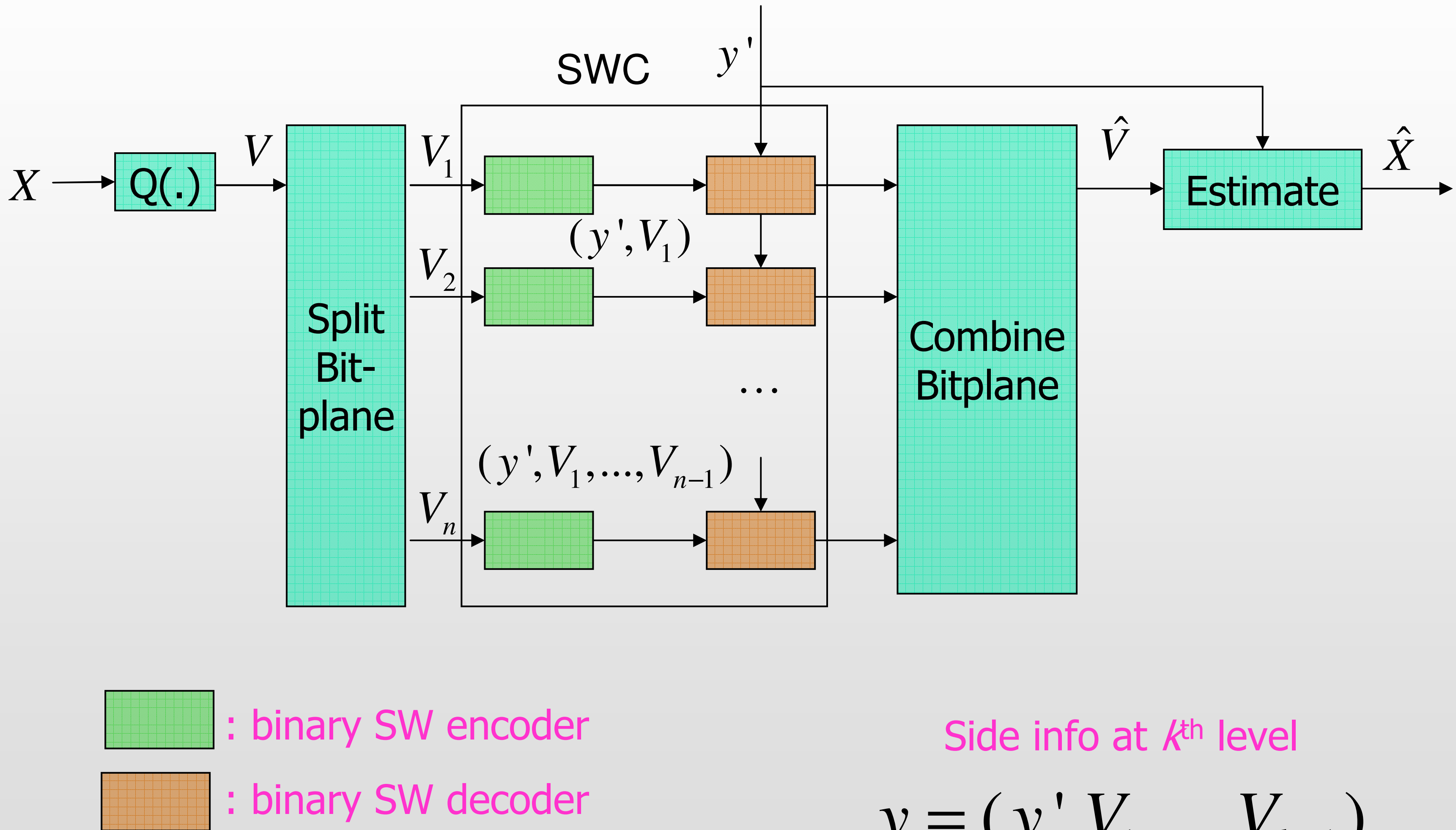
WZC vs. Classic Source Coding (SC)

Classic SC		WZC	
ECQ	Gap to $D_X(R)$	SWCQ	Gap to $D_{WZ}(R)$
ECSQ	1.53 dB	SWC-SQ	1.53 dB
ECLQ (2-D)	1.36 dB	SWC-LQ	1.36 dB
ECTCQ	0.20 dB	SWC-TCQ	0.20 dB

Same performance limits at high rate!

(Assuming ideal entropy coding and ideal SW coding)

Layer WZ Coding



Side info at k^{th} level

$$y = (y', V_1, \ldots, V_{k-1})$$

LDPC Code for binary SW Coding

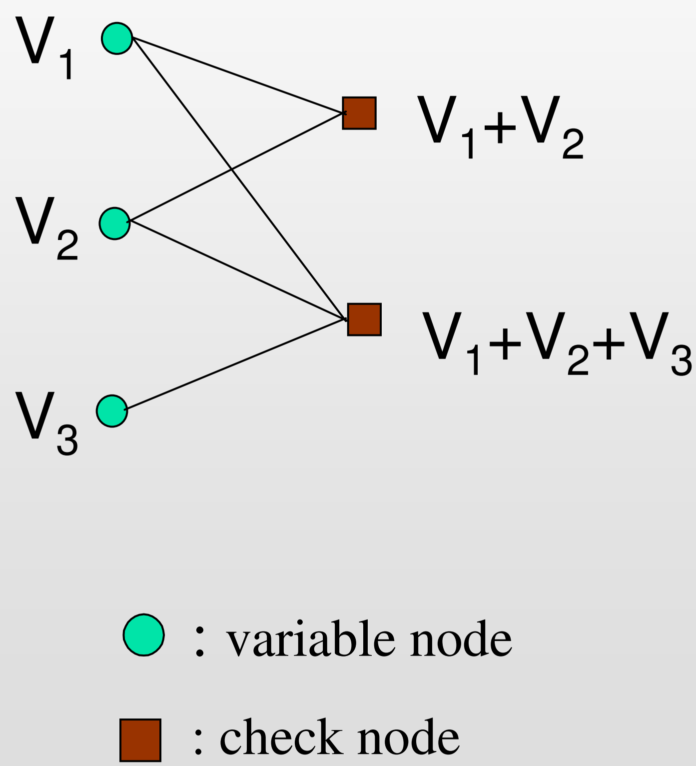
- LDPC code is a linear block code
- LDPC stands for low-density parity-check
  - "Low-density" means its parity-check matrix is sparse
- Message-passing decoding algorithm
  - Suboptimal but effective
- Pros
  - Exists flexible and systematic design techniques for arbitrary channels
  - Designed codes have excellent performance

Tanner Graph

- Consider a length-3 block code with parity check matrix

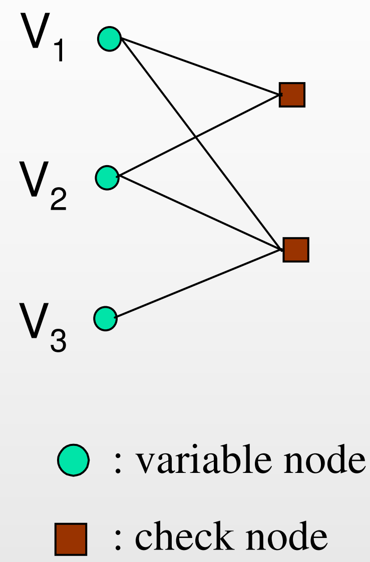
$$H = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

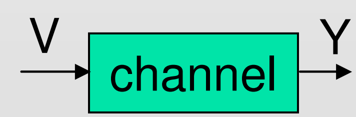
- A binary vector $\mathbf{V}=[V_1,V_2,V_3]$ is a codeword if $H^T\mathbf{V}=\mathbf{0}$



Message Passing Decoding

1. Initialization:
   - Compute the "belief" of actual transmitted bit at each variable node
2. Iteration:
   - Pass beliefs from variable nodes to check nodes; combine beliefs
   - Pass beliefs from check nodes to variable nodes; combine beliefs
3. Exit condition:
   - Estimate variable node values by thresholding current beliefs. Exit if the estimates form a valid codeword; otherwise, back to 2.

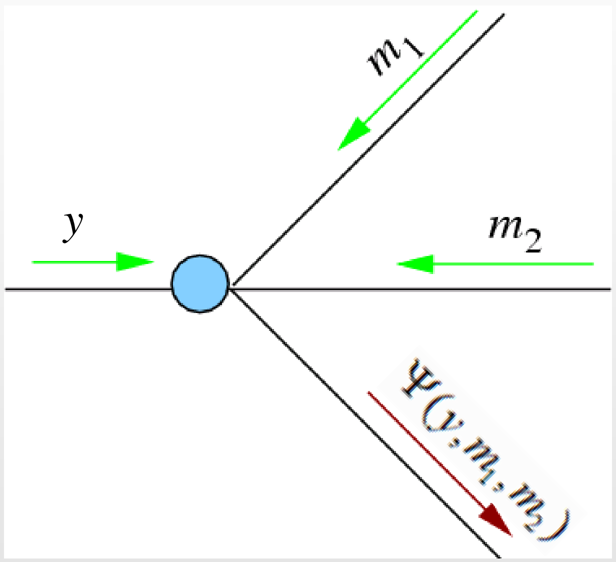


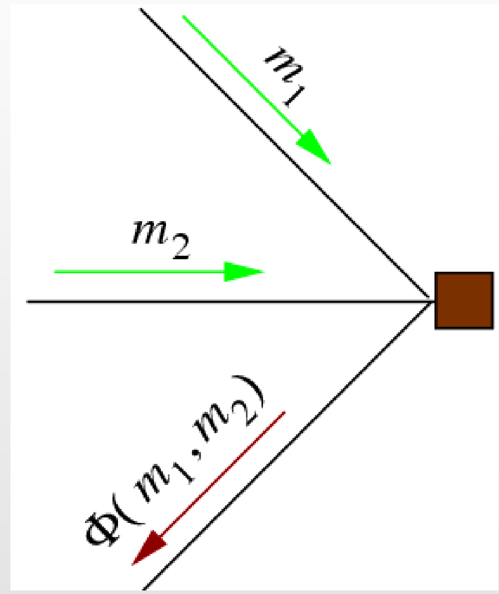


Belief usually in the form of log-likelihood ratio $\left(\log \frac{p(y \mid V=0)}{p(y \mid V=1)} \right)$

Message Passing Decoding

- If we assume all messages are independent





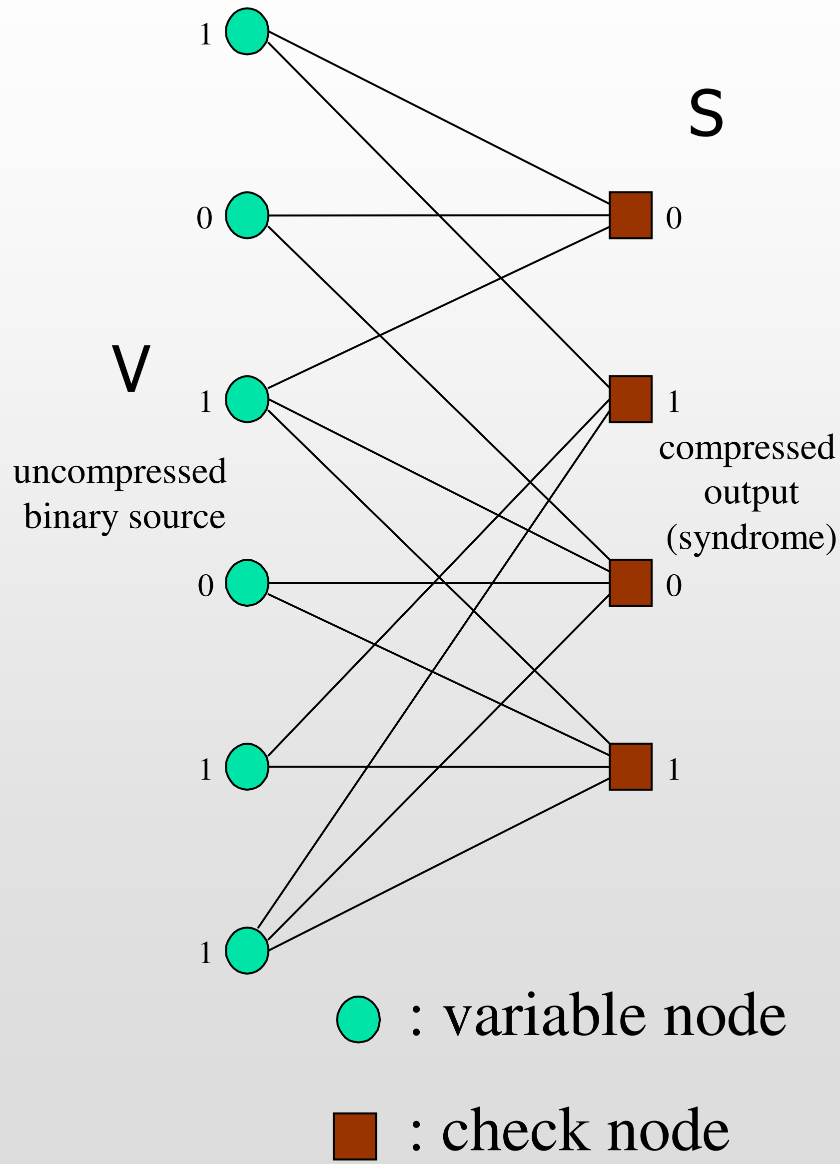
$$\Psi = \log \frac{p(y \mid V=0)}{p(y \mid V=1)} + m_1 + m_2$$

$$\tanh \frac{\Phi}{2} = \tanh \frac{m_1}{2} \tanh \frac{m_2}{2}$$

- Message passing decoding performs well for long block-length code with relatively few connections (low-density)

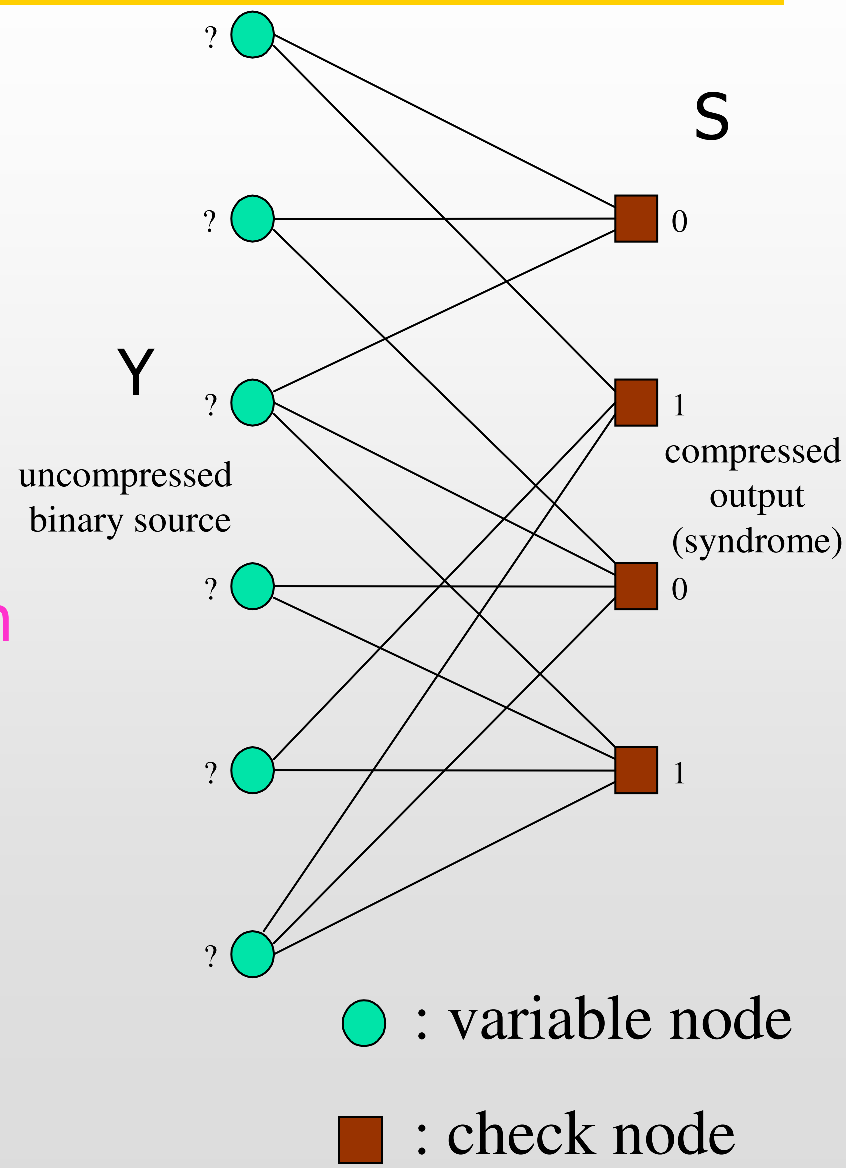
SW Encoding with LDPC Codes

- Encoding:
  - Output check values *S*
- Compression rate:
  - $R=(n-k)/n = 4/6 = 2/3$



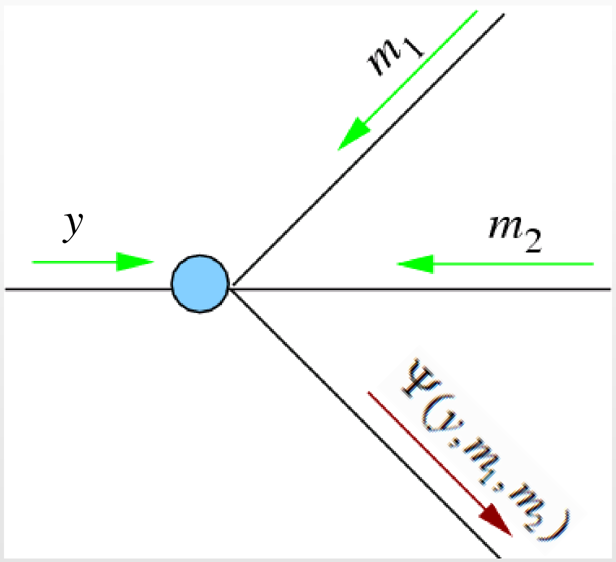
SW Decoding with LDPC Codes

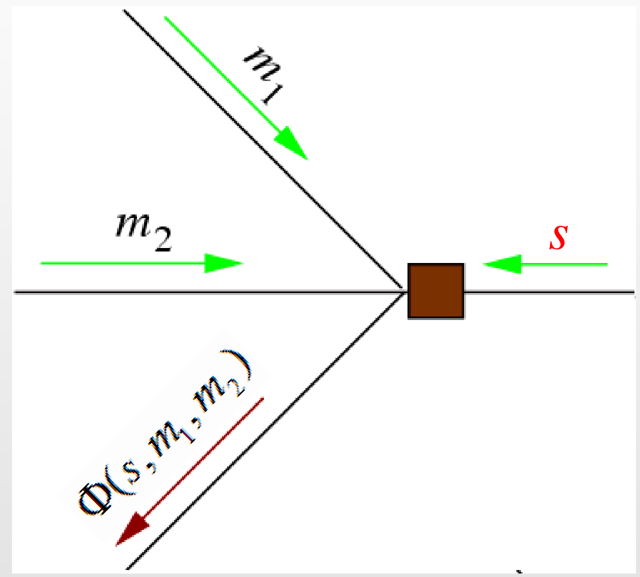
- Decoding:
  - View SI *Y* as hypothetical outputs of a channel
  - Input received *S* as check node values
  - Decode to a code vector with the received syndromes *S* instead of a codeword



Message Passing Decoding

- If we assume all messages are independent





$$\Psi = \log \frac{p(y \mid V=0)}{p(y \mid V=1)} + m_1 + m_2$$

$$\tanh \frac{\Phi}{2} = (1-2s) \tanh \frac{m_1}{2} \tanh \frac{m_2}{2}$$

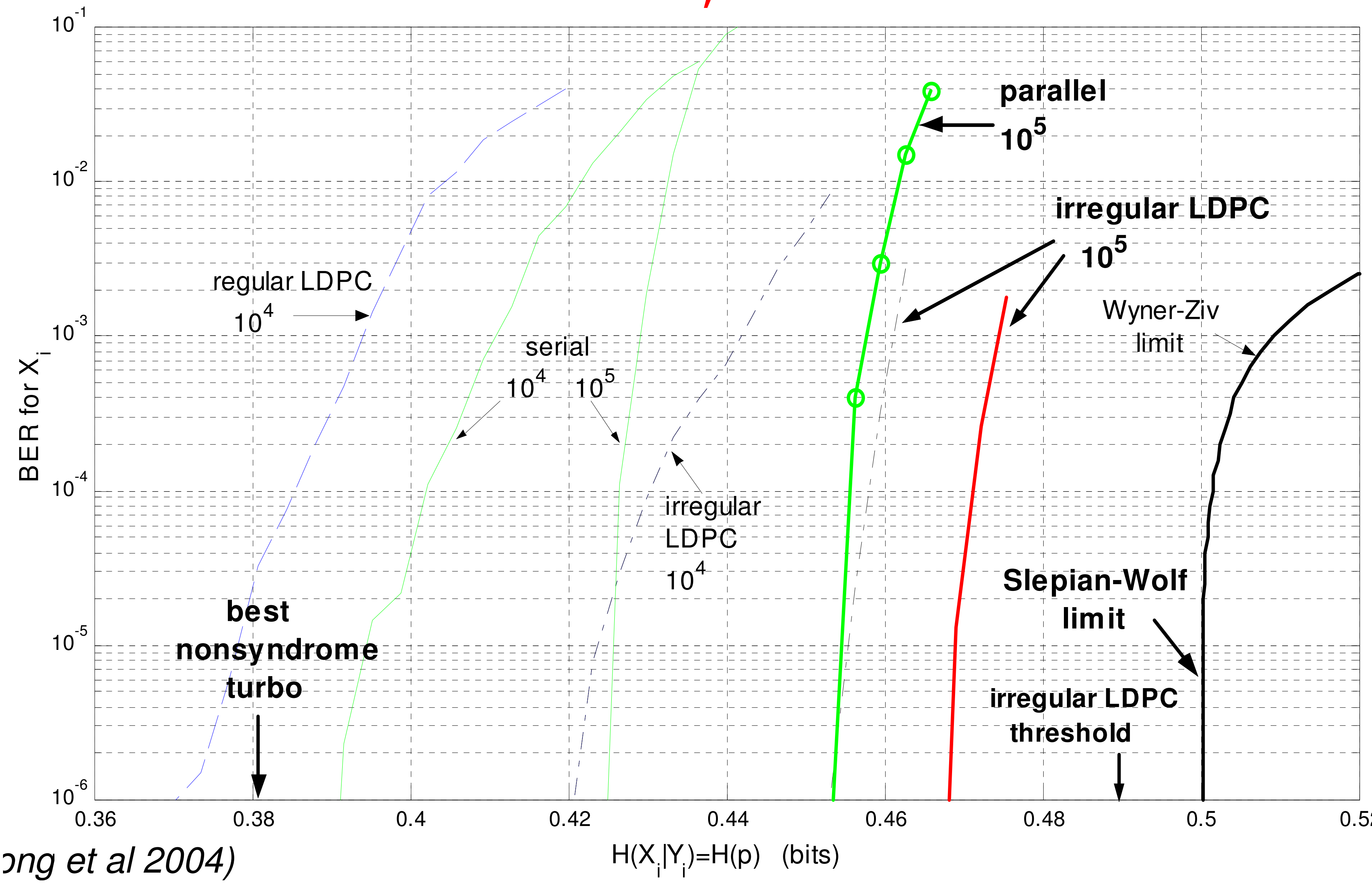
- When **v** is a codeword of the LDPC code (**s** is all-zero sequence), SWC decoding ≡ LDPC channel decoding

Simulation Results

- Asymmetric SW
- Non-asymmetric SW
- Quadratic Gaussian WZ
  - 1D lattice
  - 2D lattice
  - Trellis Coded Quantization (TCQ)
- MT source coding

Asymmetric Binning for SW

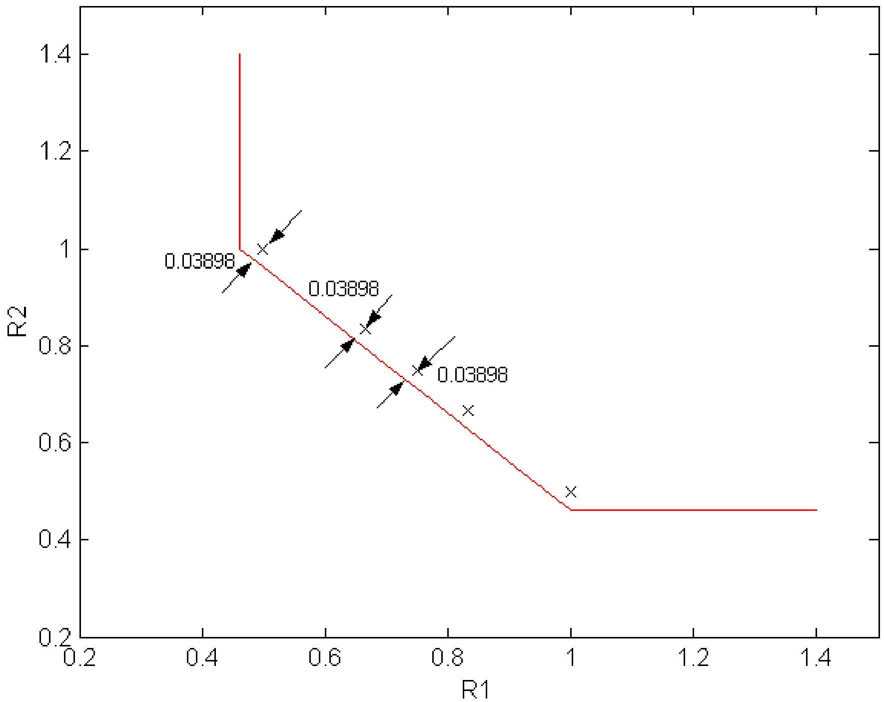
For two sources, code rate = ½

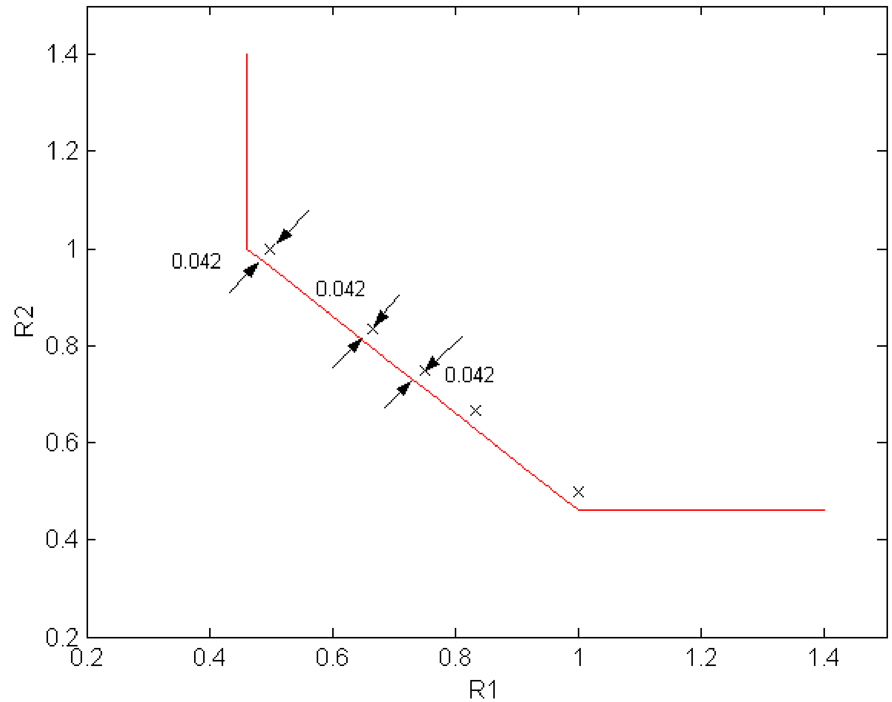


ong et al 2004)

Non-asymmetric Binning for SW

Codeword length 20,000 bits



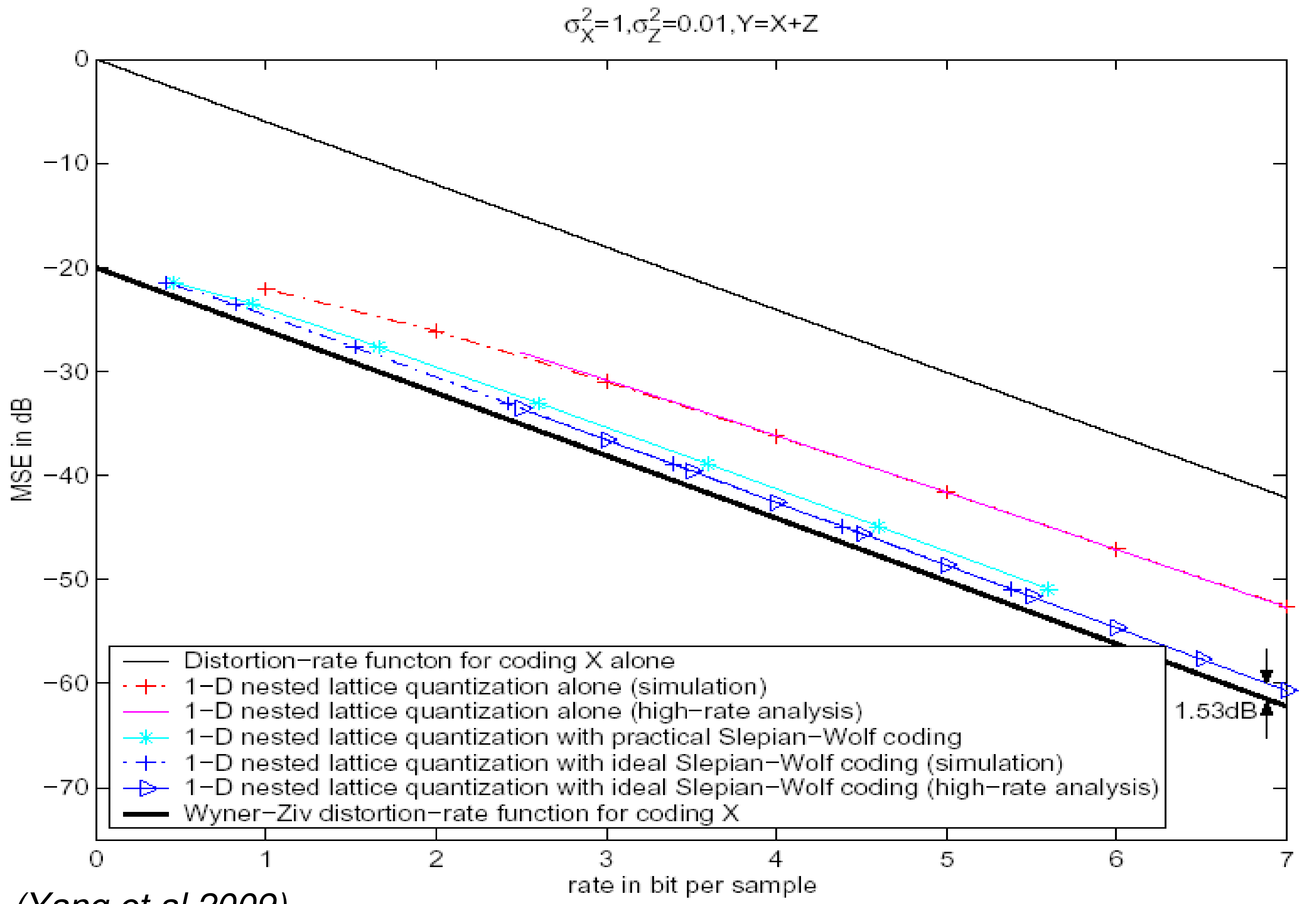


LDPC code

Turbo code

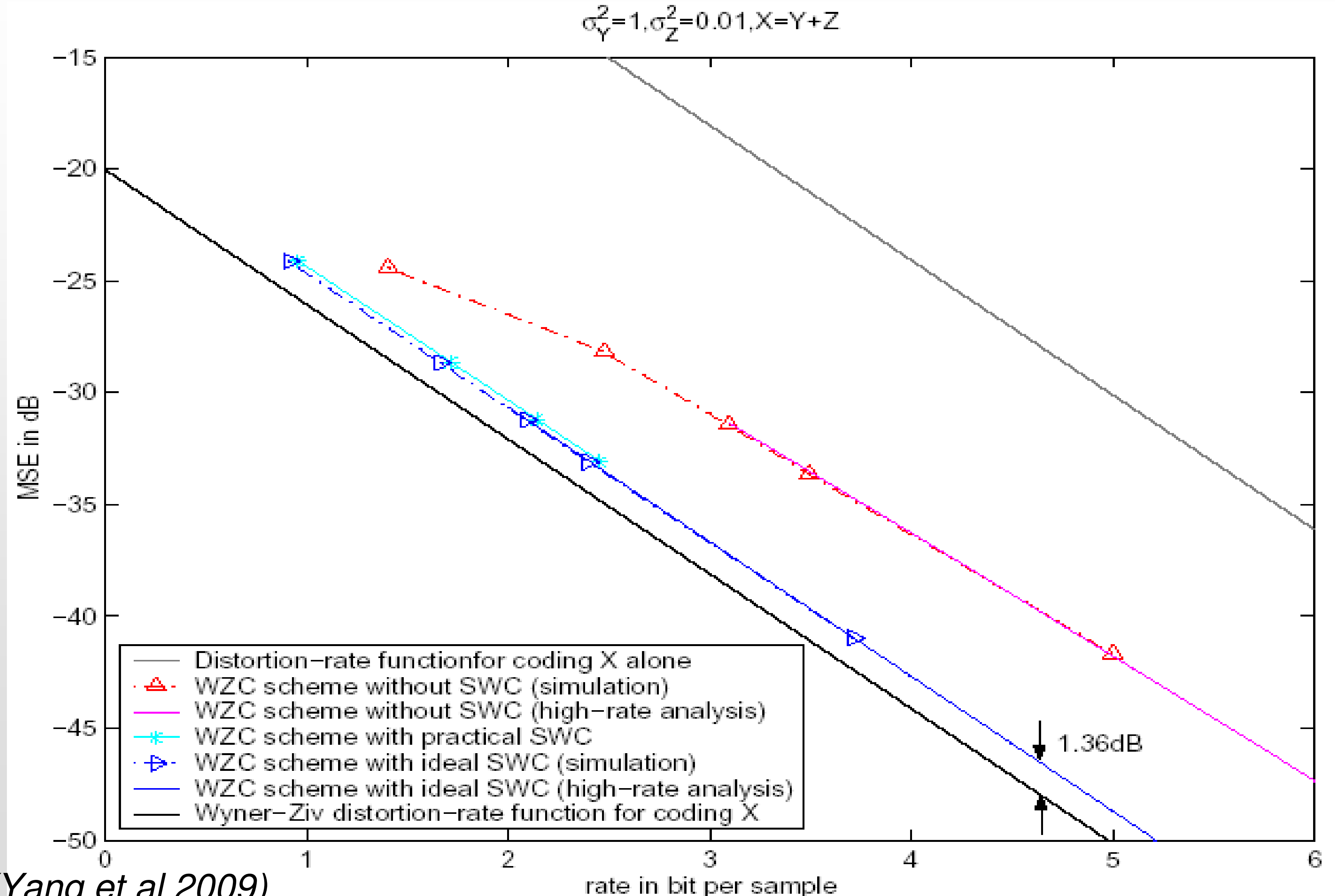
(Stankovic et al 2006)

Gaussian WZC (NSQ 1-D Lattice)



(Yang et al 2009)

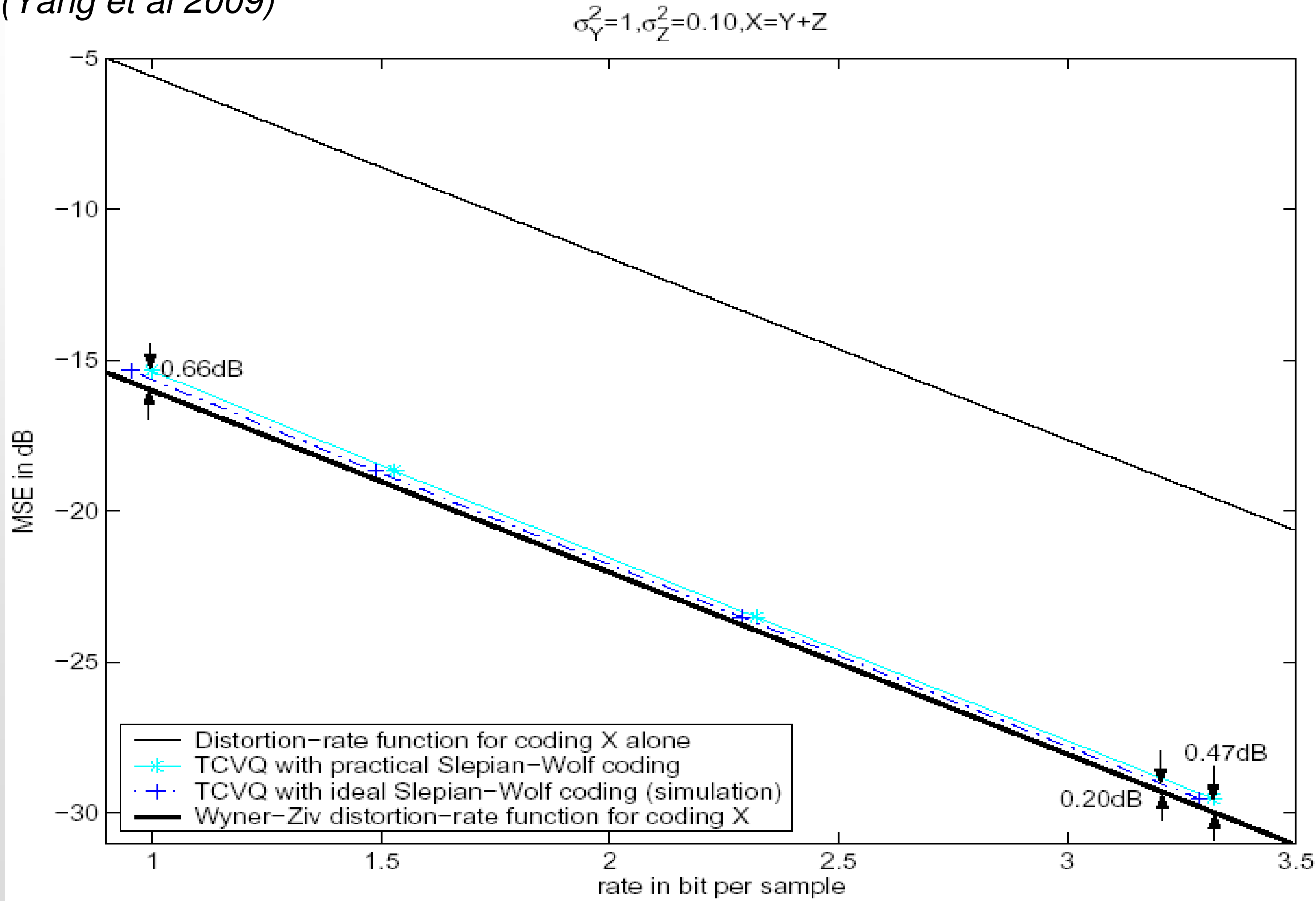
Gaussian WZC (2-D Nested Lattice)



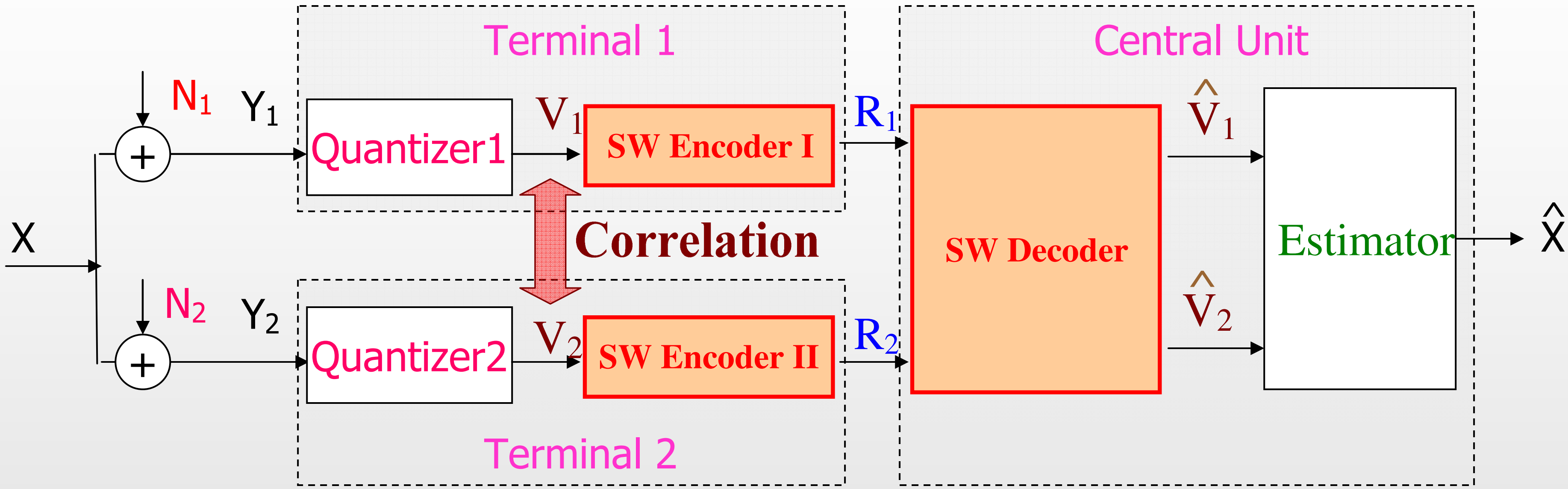
(Yang et al 2009)

Gaussian WZC (with TCVQ)

(Yang et al 2009)



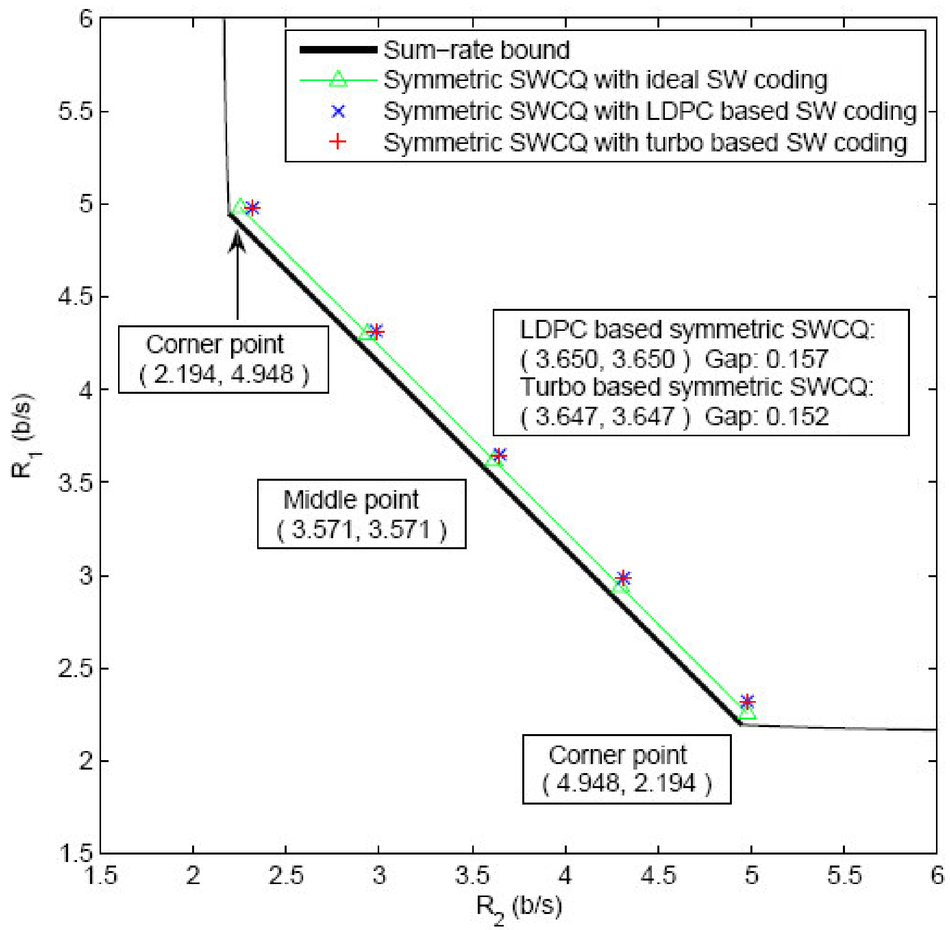
MT Source Code Design

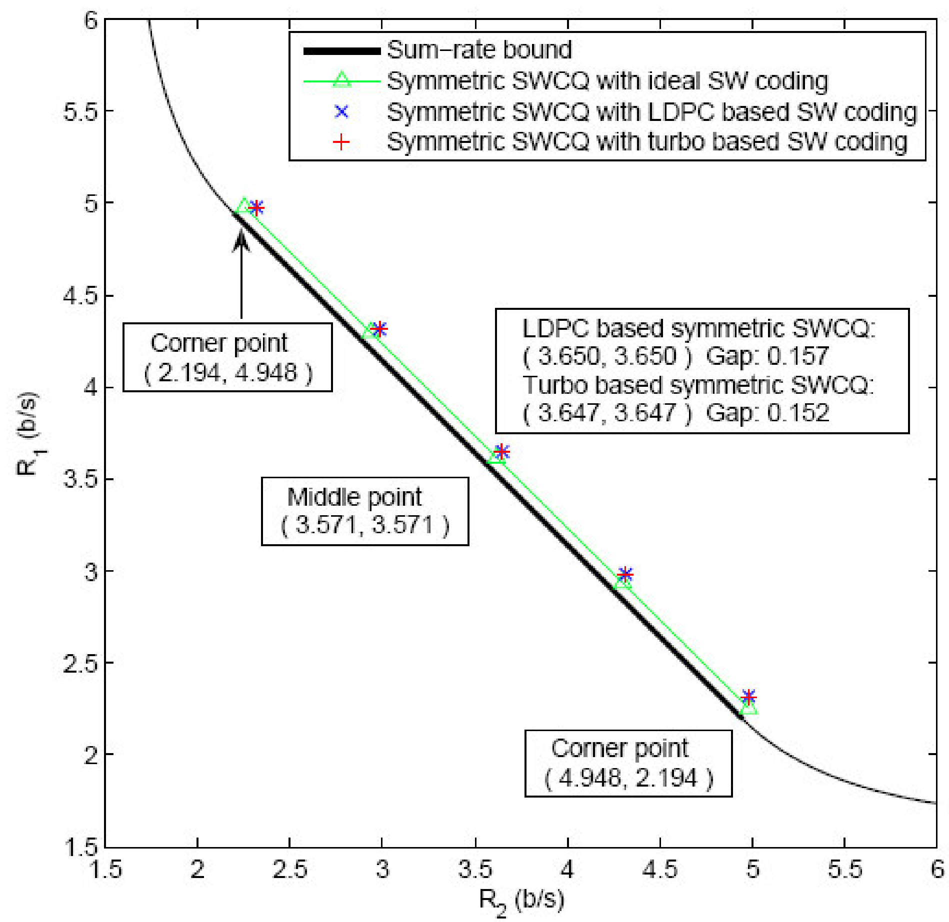


- Conventional quantization + lossless "non-asymmetric" Slepian-Wolf coding of quantization indices V_1 and V_2

(Yang, Stankovic, Xiong, Zhao, IEEE IT, March 2008)

Gaussian MT (with TCQ)





Direct MT $D_1=D_2=-30$ dB, $\rho=0.99$

Indirect MT $D=-22.58$ dB, $\sigma_{n_1}=\sigma_{n_2}=1/99$

DSC: Key Applications

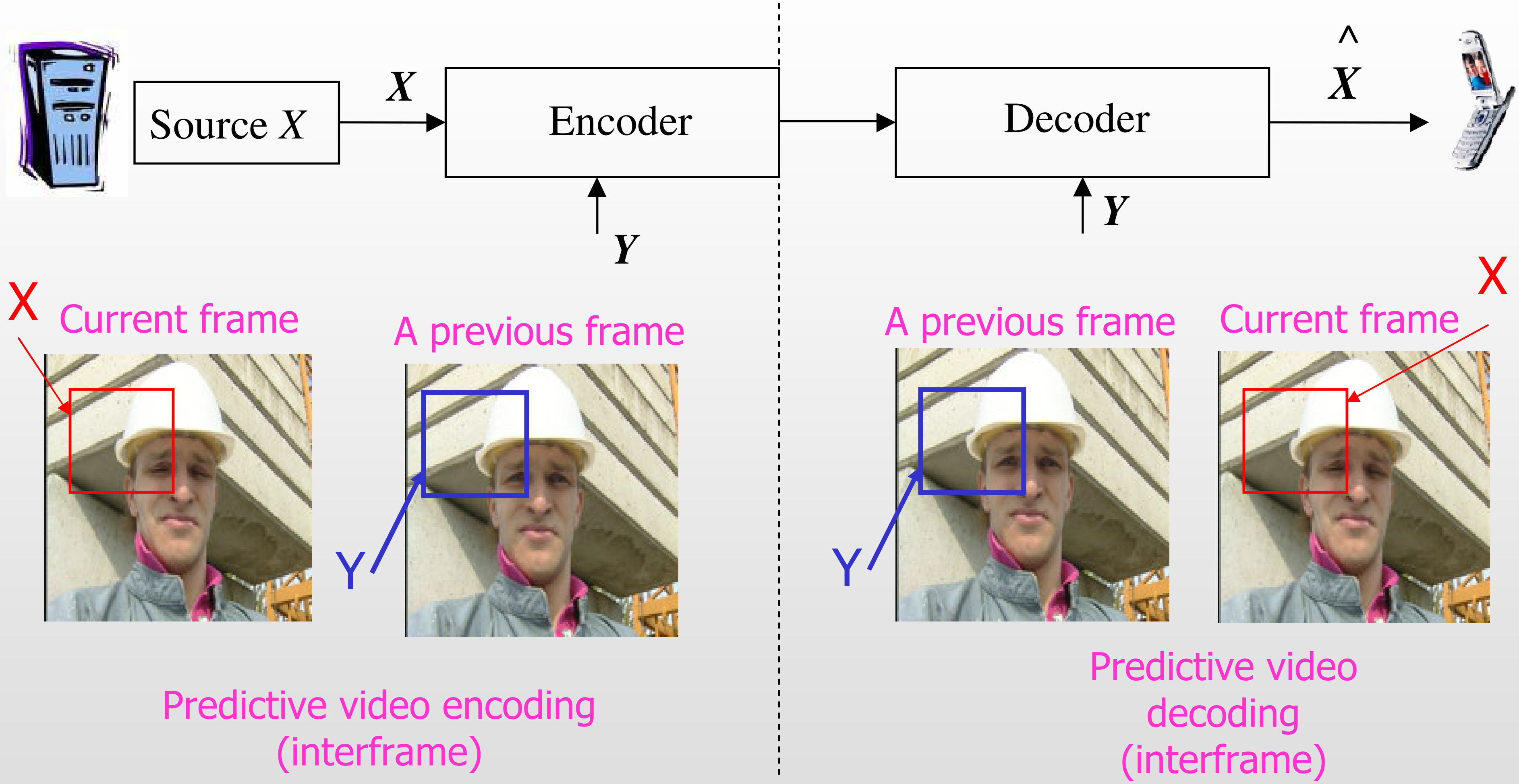
A Step Back: Reality

- Despite great recent theoretical achievements, no commercial product exploits in any way DSC yet
- Practical limitations:
  - *Real sources are not Gaussian, but can be often approximated as Gaussian*
  - *Correlation statistics – varying, difficult to model, track/predict*
  - *To get good performance long block lengths are needed for channel codes, thus resulting in delay and implementation constraints (e.g., memory)*

Applications

- Distributed (WZ) video coding
- Stereo Video Coding
- Multimedia streaming over heterogeneous networks
- Wireless sensor networks
- Spectrum sensing

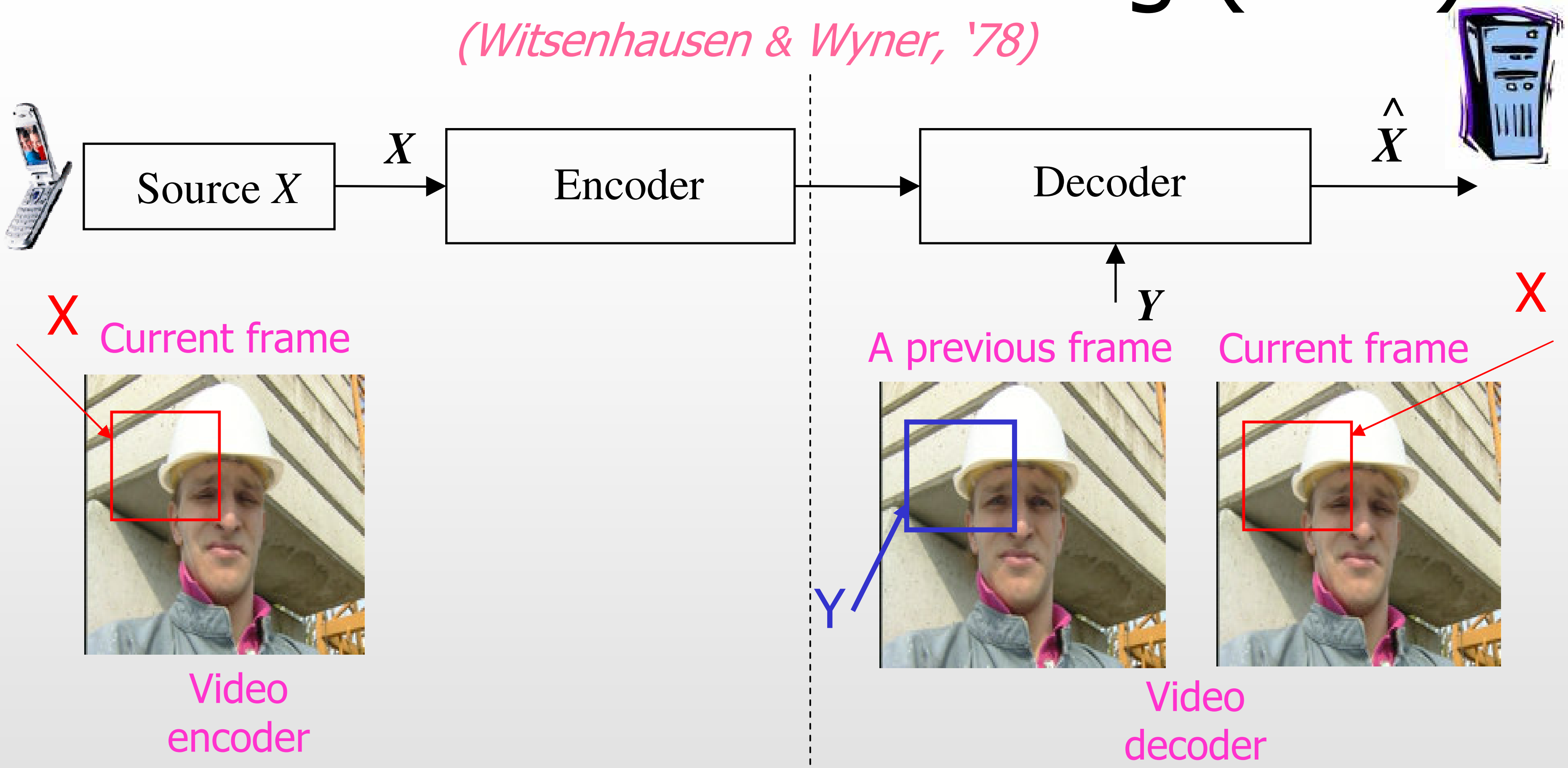
Conventional Video Coding



- High-complexity encoding (TV station, strong server)
- Low-complexity decoding (TV, computer, cell-phone)

Distributed Video Coding (DVC)

(Witsenhausen & Wyner, '78)



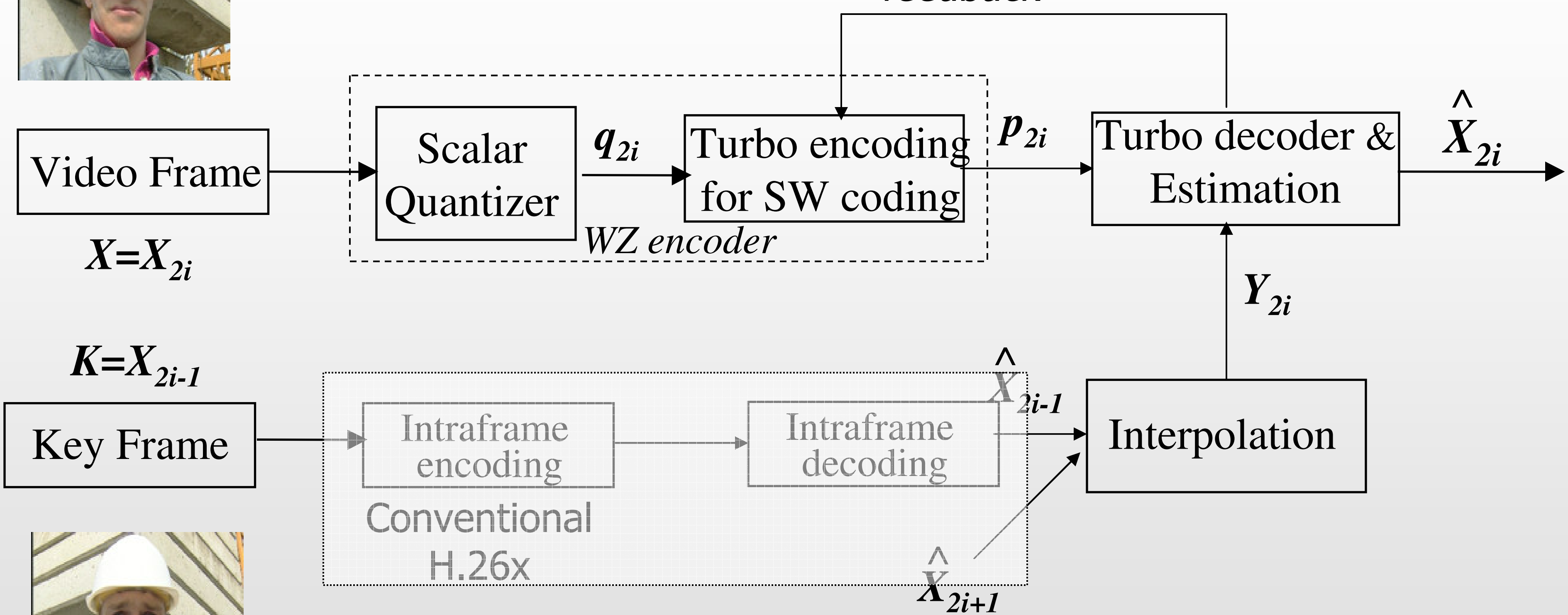
- The encoder does not need to know SI *Y*
- Low-complexity encoding (cell-phone, web-cam)
- High-complexity decoding (computer server)
- Low-complexity network: cell-server (converts to H264) -cell

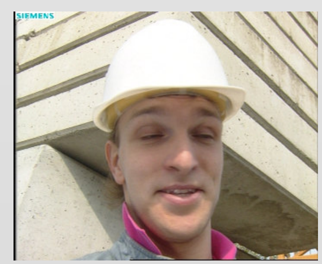
Reported DVC Coders

- Stanford's group
  - Pixel-based DVC
  - DCT-based DVC
- Berkeley's group (PRISM) DCT-based
- TAMU's group Scalable DVC
- Many extensions/improvements of the above, e.g., by the DISCOVER partners

(UPC Spain, IST Portugal, EPFL Switzerland, UH Germany, INRIA France, UNIBIS Italy)

Pixel-domain DVC





(Aaron, Zhang, Girod, 2002)
(Aaron, Rane, Zhang, Girod, 2003)

Decoder side information
generated by interpolation
PSNR 30.3 dB

After Wyner-Ziv decoding
16-level quantization – 1.375 bpp
PSNR 36.7 dB

(Aaron, Zhang, Girod, 2002)
(Aaron, Rane, Zhang, Girod, 2003)

Decoder side information
generated by interpolation
PSNR 24.8 dB

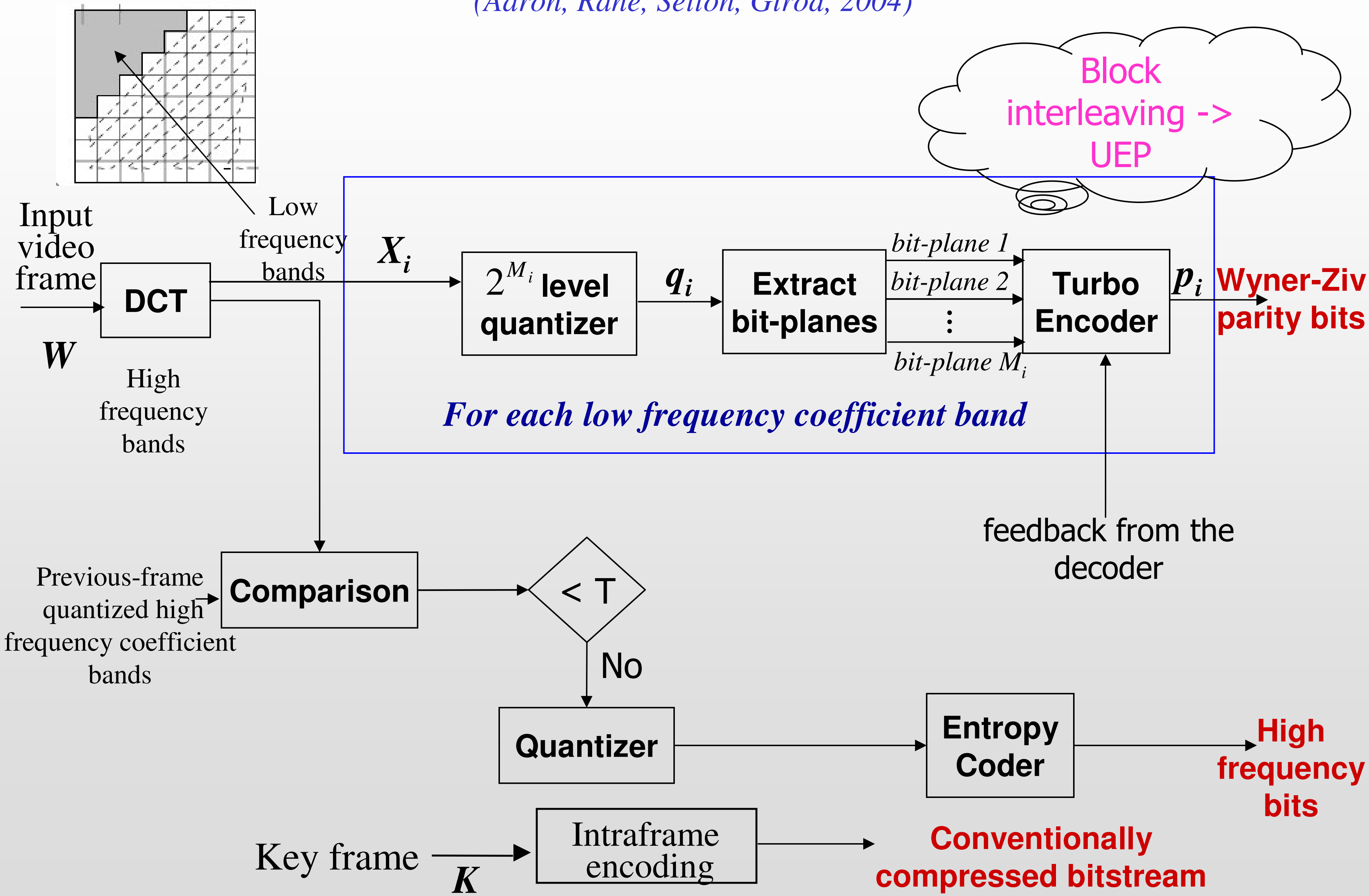
After Wyner-Ziv decoding

16-level quantization – 2.0 bpp
PSNR 36.5 dB

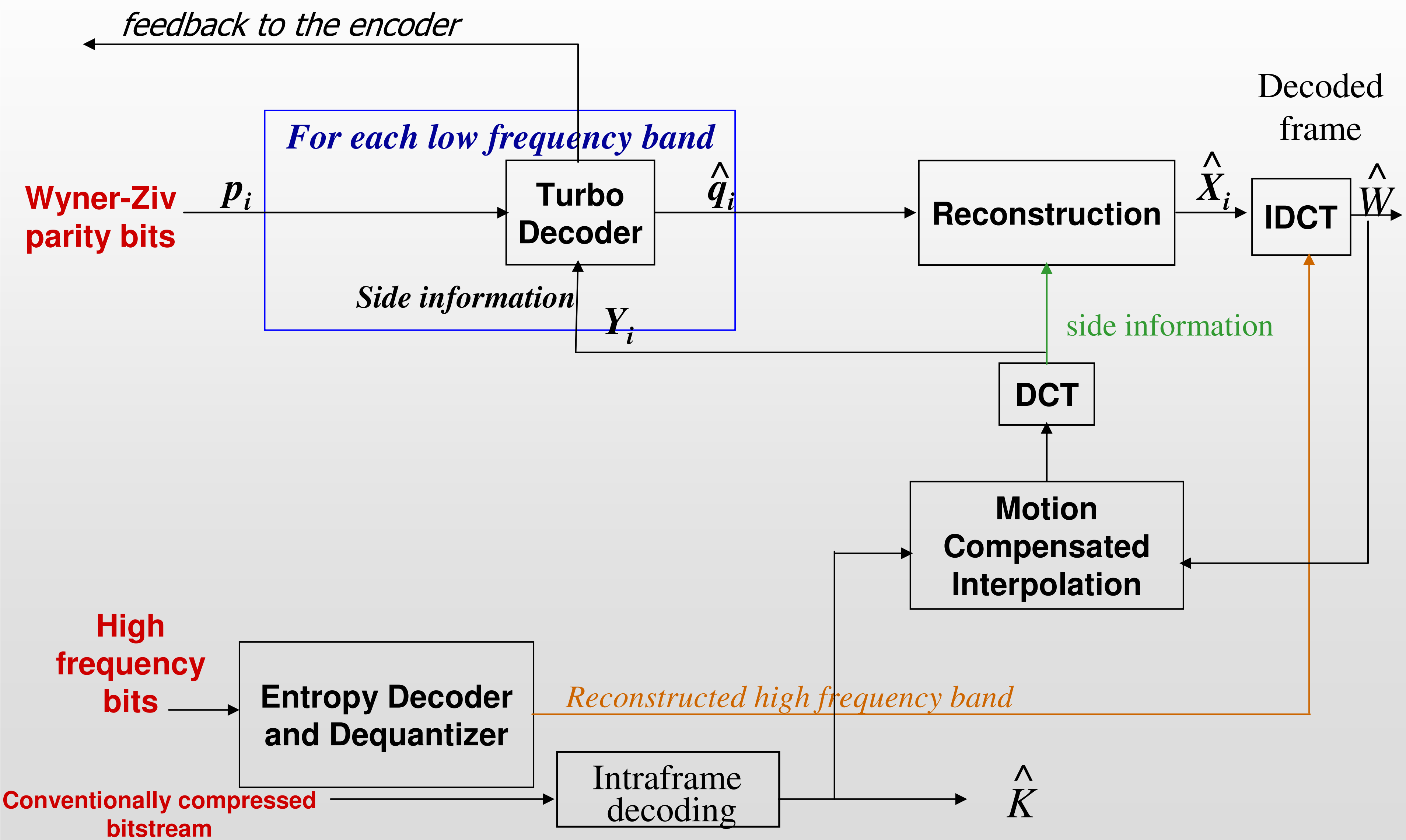
(Aaron, Zhang, Girod, 2002)
(Aaron, Rane, Zhang, Girod,2003)

DCT-domain: Encoder Look

(Aaron, Rane, Setton, Girod, 2004)



DCT-domain: Decoder Look



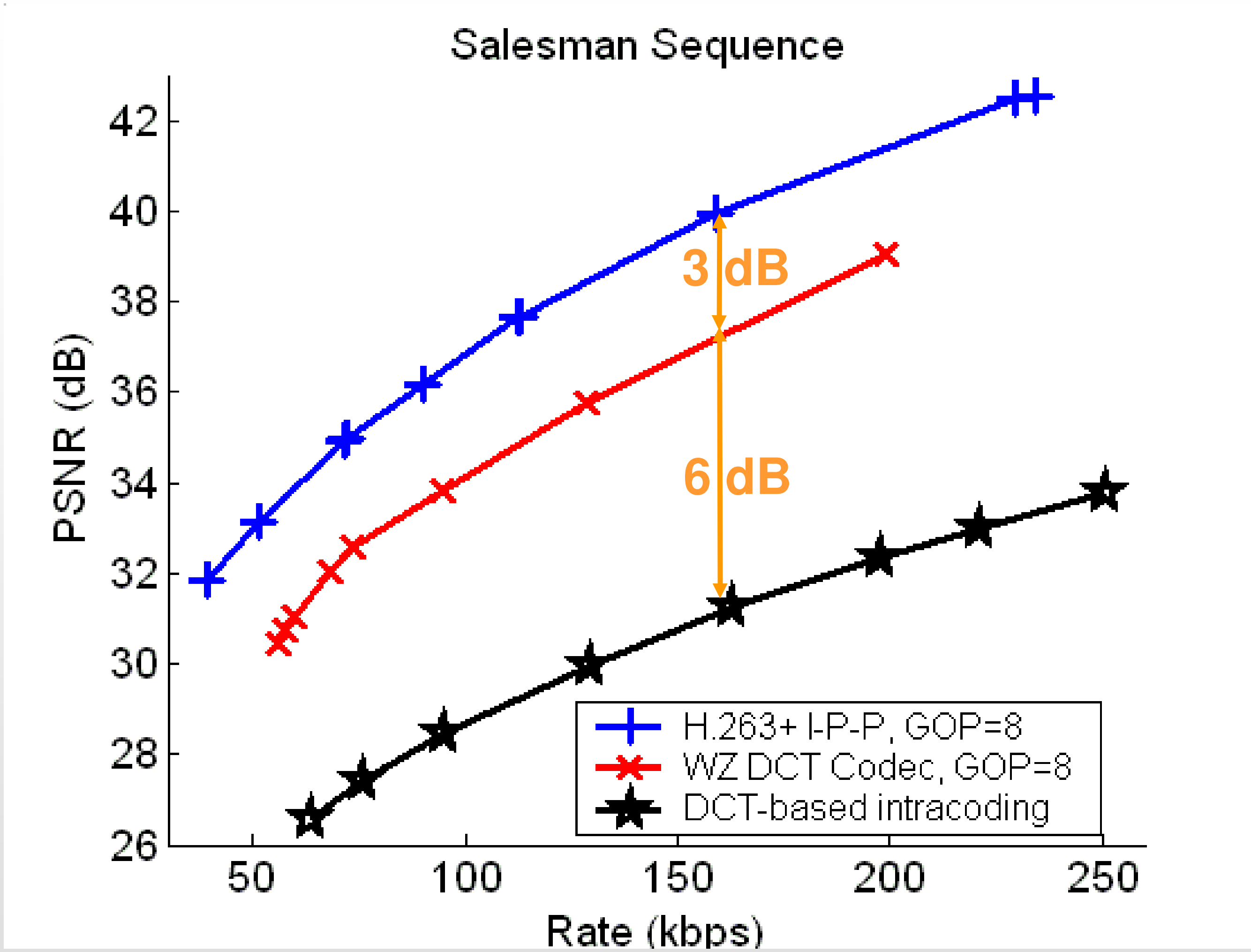
DCT-based Intracoding 149 kbps
$PSNR_Y$=30.0 dB

Wyner-Ziv DCT codec 152 kbps
$PSNR_Y$=35.6 dB. Every 8th frame is a key frame

(Aaron, Rane, Setton, Girod, 2004)

Every 8th frame is a key frame

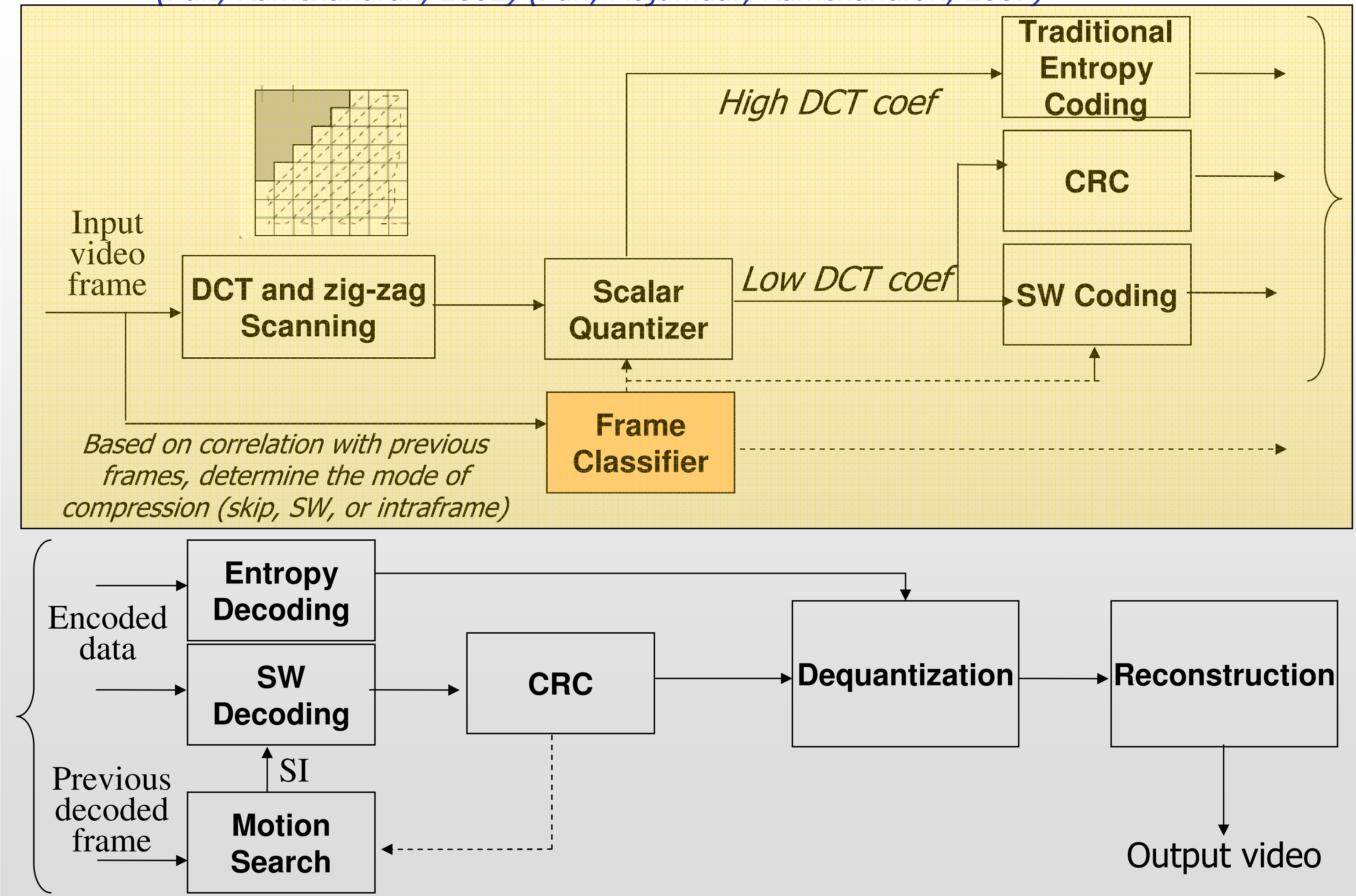
100 frames of *Salesman* QCIF sequence at 10fps



(Aaron, Rane, Setton, Girod, 2004)

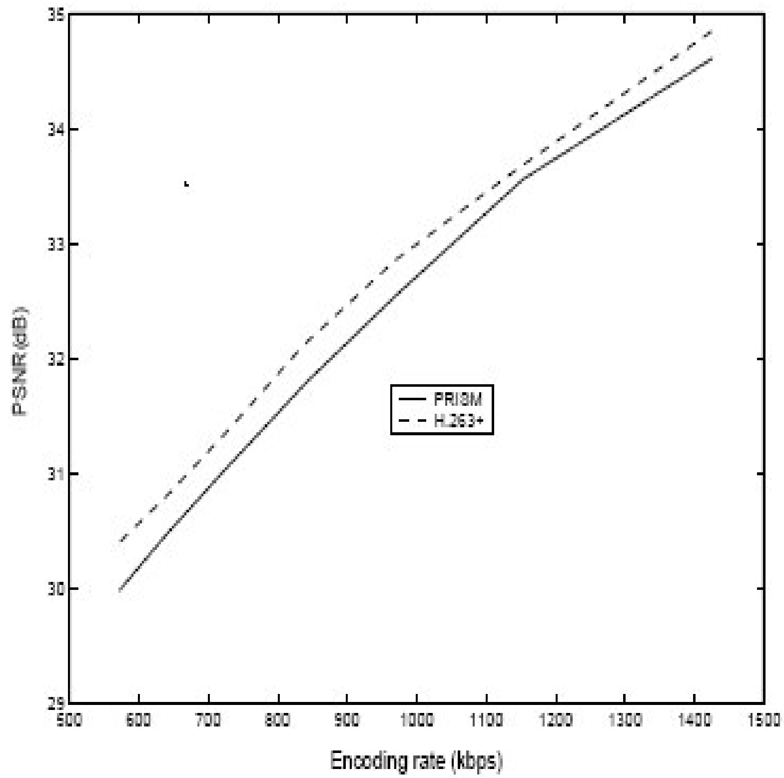
PRISM

(Puri, Ramchandran, 2002) (Puri, Majumdar, Ramchandran, 2007)

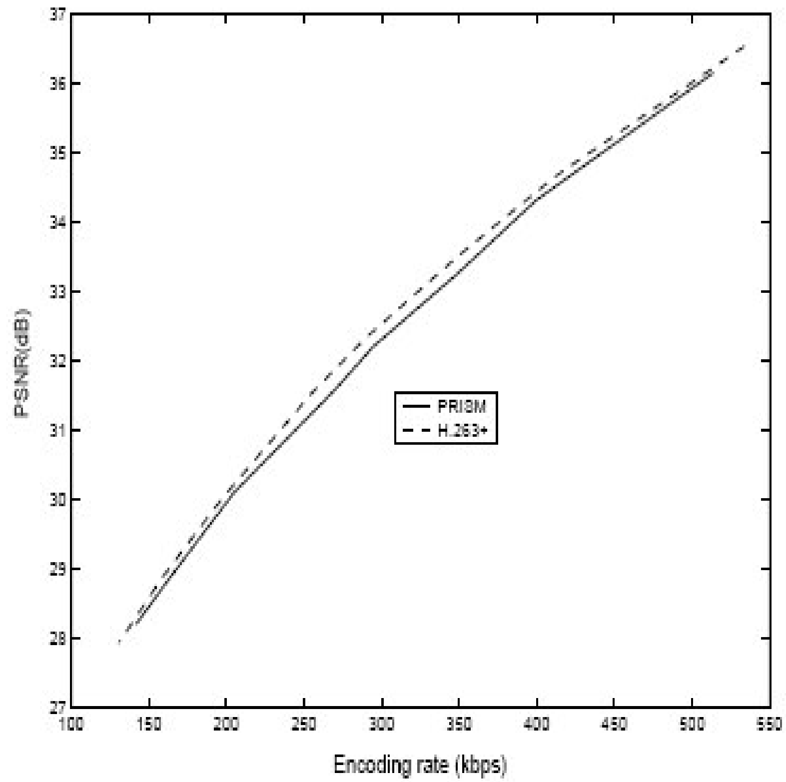


PRISM: Results

Sequence	Rate (bits)	H.263+ PSNR (dB)	PRISM PSNR (dB)
Football	1400000	35.42	34.20
Euronews	1560000	36.91	35.61



CIF Football

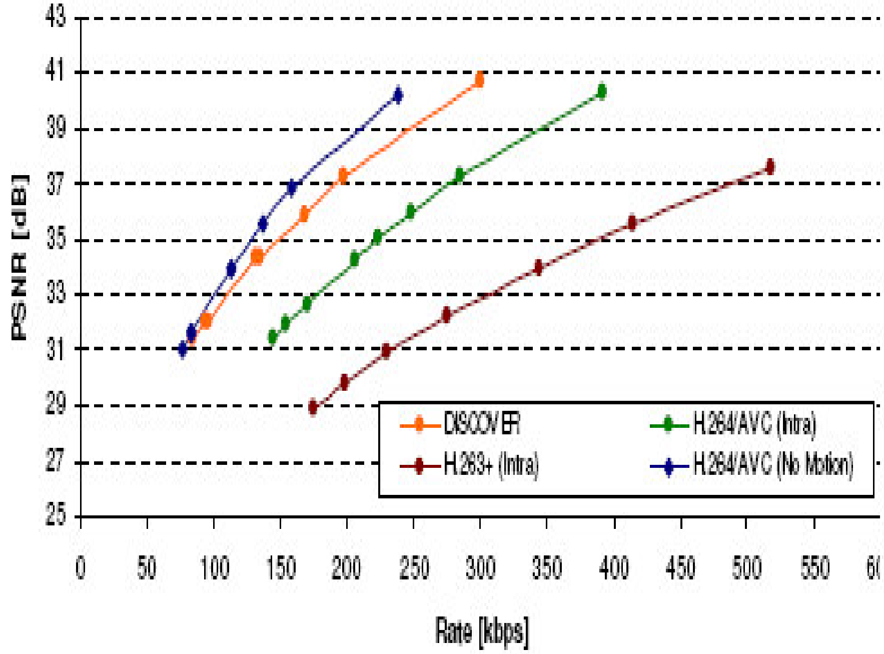


QCIF Stefan

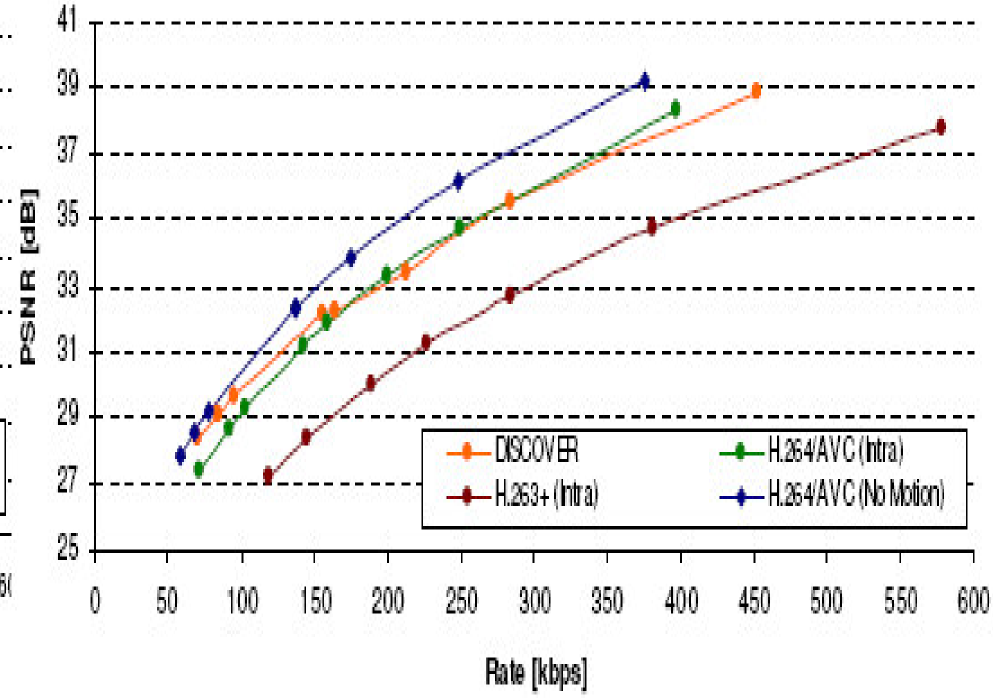
Latest Developments

- Performance improvement
  - Stanford's DCT-based architecture (with DISCOVER improvements) outperforms H.264/AVC intra-coded
  - PRISM outperforms H.263+
- Improved error resilience
- No need for feedback channel
- Extensions to multi-view video

DISCOVER Results



QCIF Hall Monitor

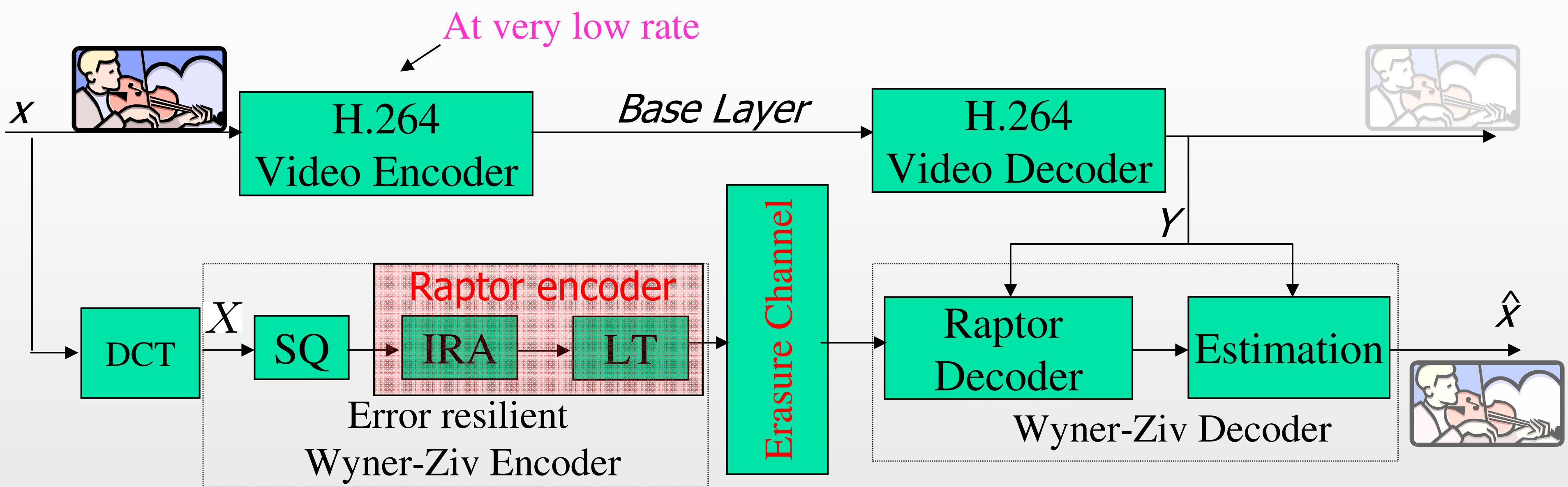


QCIF Foreman

- Much lower encoding complexity than H.264/AVC intra, and comparable decoding complexity

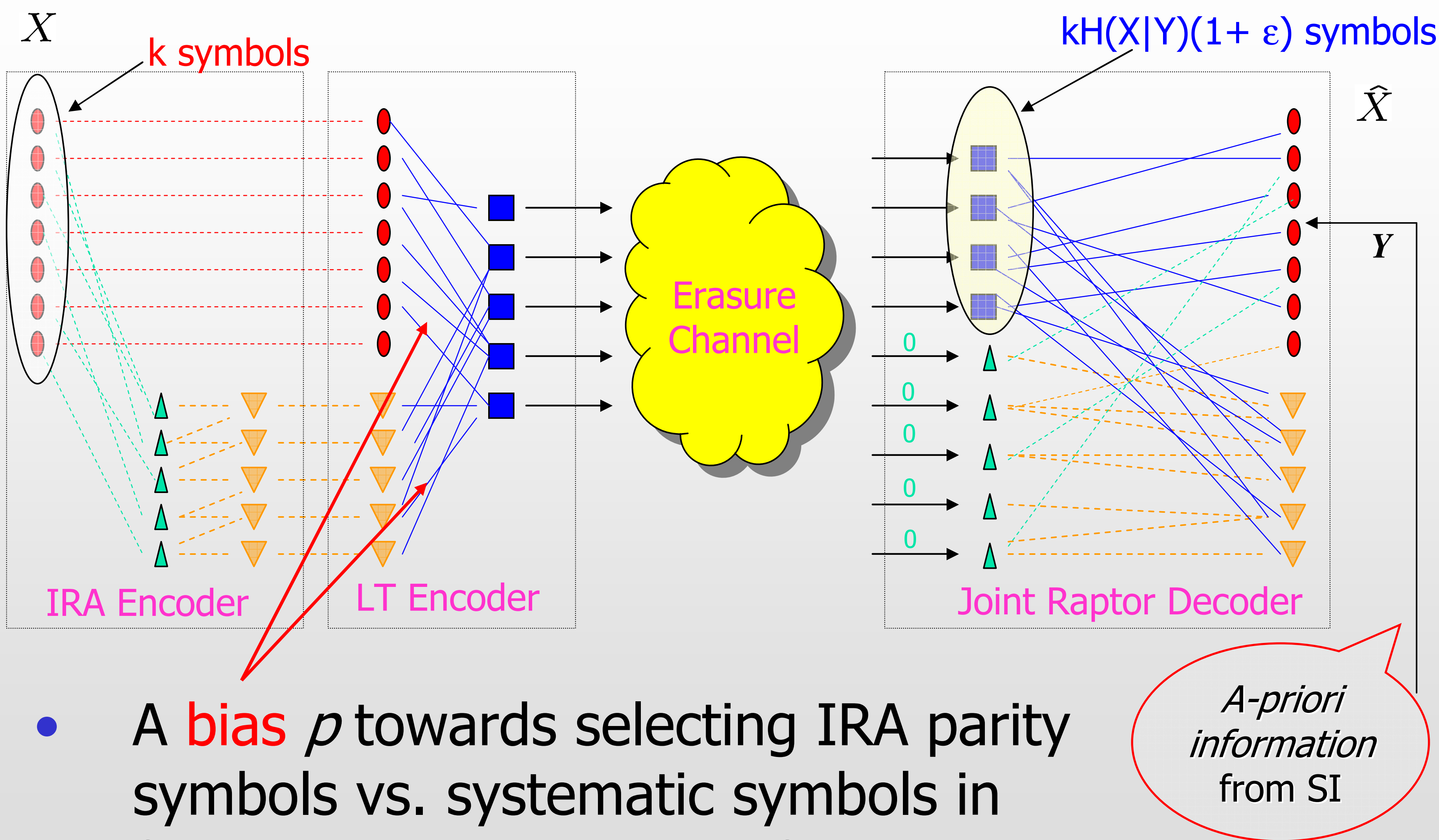
Robust Scalable DVC

(Xu, Stankovic, Xiong, 2005)



1. Encode x at very low bitrate with H.26x and send it to the decoder using strong error protection
2. Decode the received stream and get SI Y
3. x is compress/protected again with a Raptor code assuming Y as SI and erasure packet transmission channel
4. The decoder decodes X using Y as SI.

Raptor Code

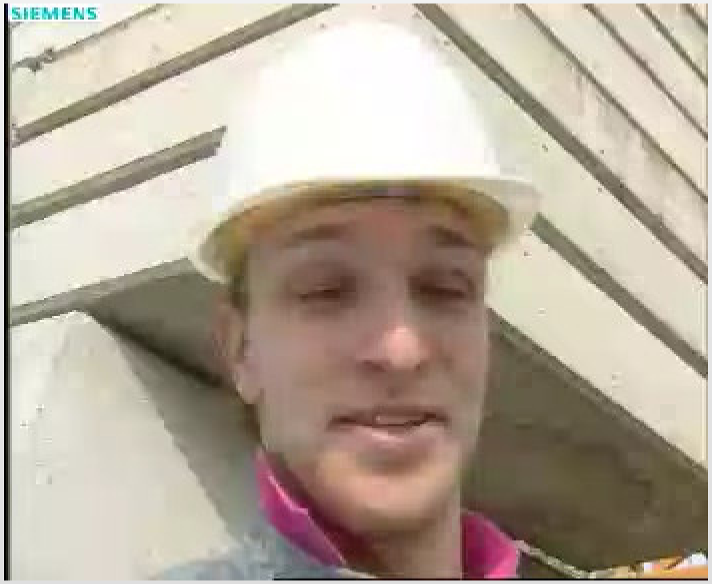


- A bias p towards selecting IRA parity symbols vs. systematic symbols in forming bipartite graph of the LT code

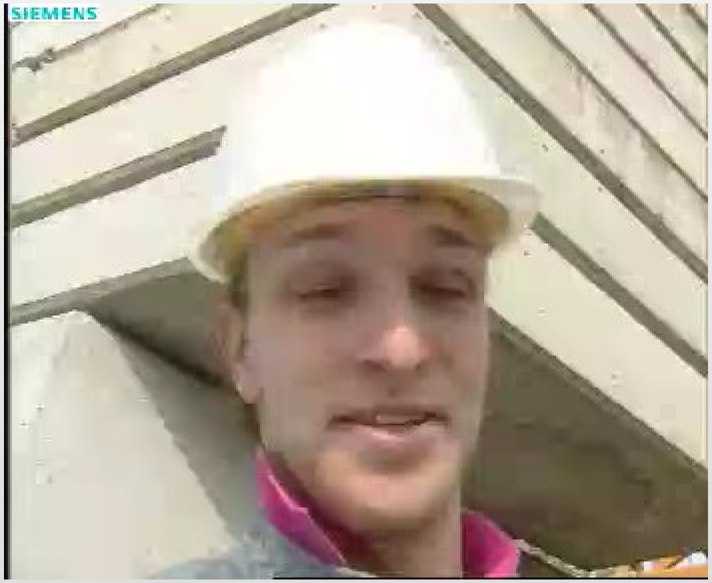
Simulation Example

(Xu, Stankovic, Xiong, 2005)

Transmission rate 256 Kbps
5% macroblock loss rate in the base layer
10% packet loss rate for WZ coding layer



H264 FGS



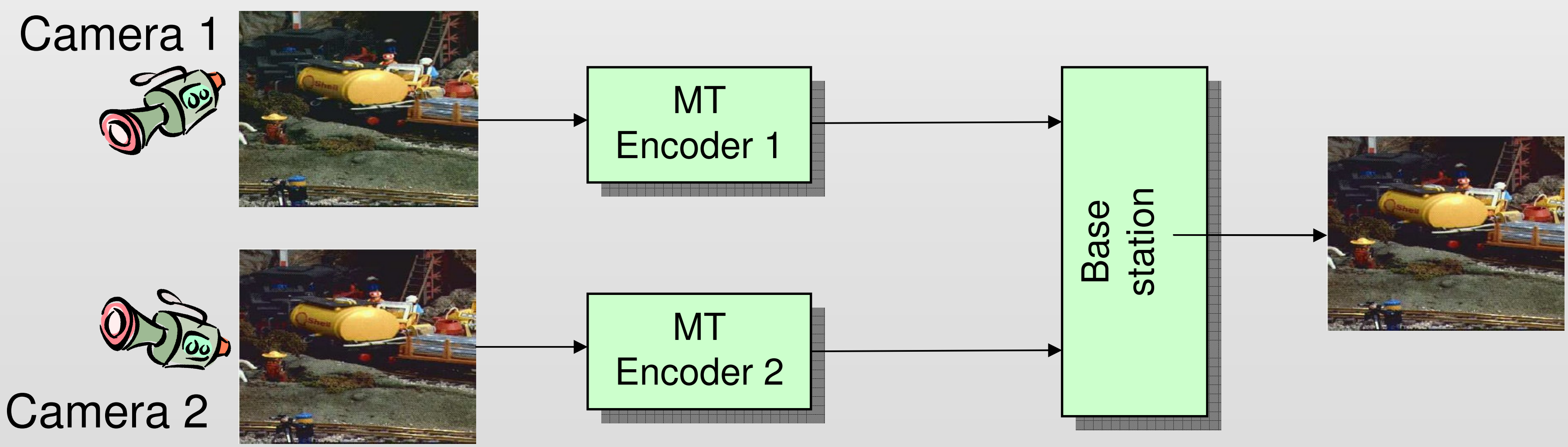
Scalable DVC system

Applications

- Distributed (WZ) video coding
- Stereo Video Coding
- Multimedia streaming over heterogeneous networks
- Wireless sensor networks
- Spectrum sensing

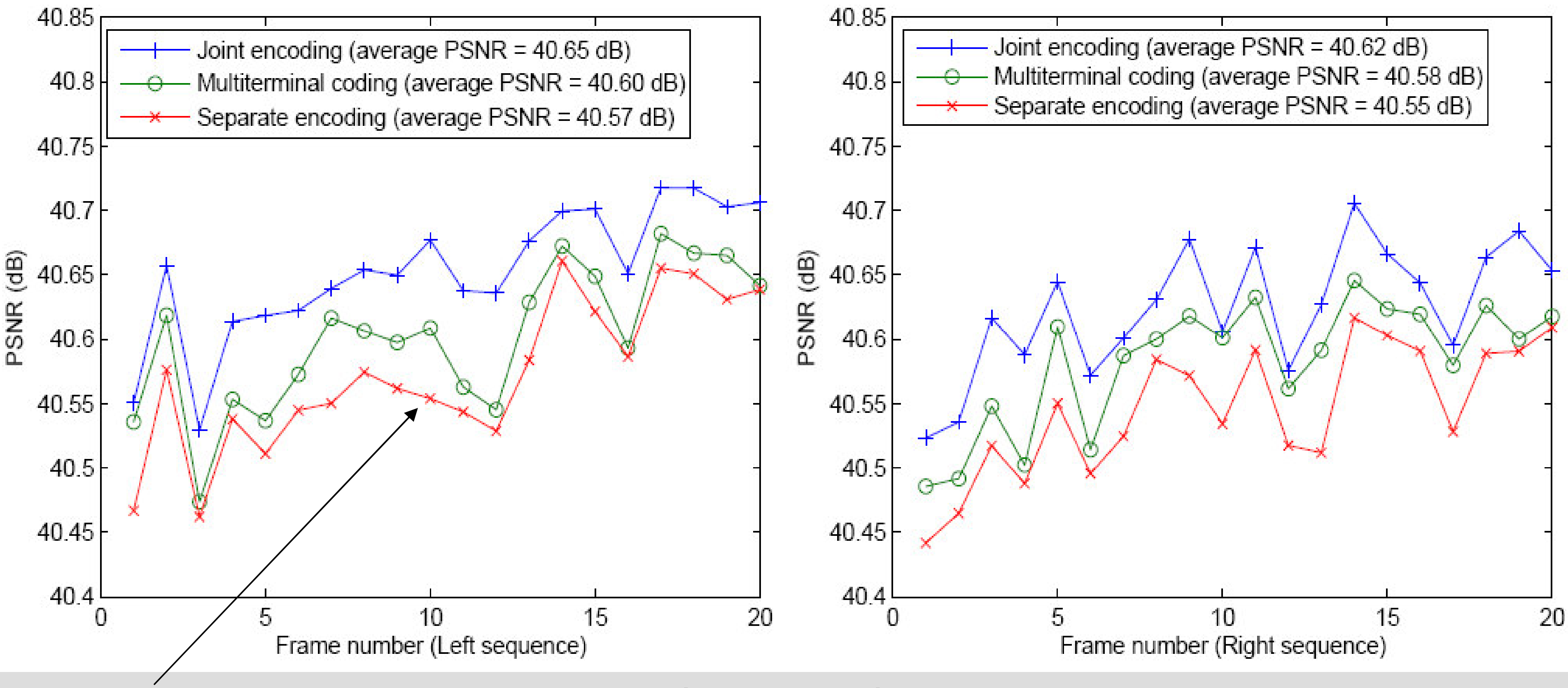
Stereo Video Coding *(Yang, Stankovic, Zhao, Xiong, 2009)*

- The same view encoded independently with two cameras
- High correlation among the views can be exploited with MT source coding



Stereo Video Coding *(Yang, Stankovic, Zhao, Xiong, 2009)*

- Both views compressed with TCQ+LDPC codes using MT source coding scheme



H.264/AVC

Tunnel Stereo Video Sequence

Distributed/Stereo Video Applications

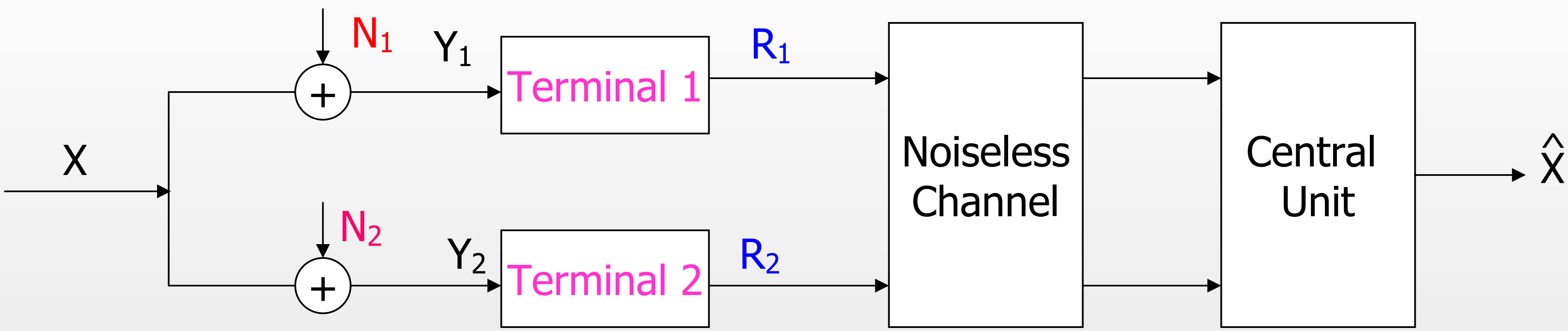
- A new attractive video compression paradigm
  - Video surveillance
  - Low complexity networks
  - Very-low complexity robust video coding
  - Multiview/3D video coding

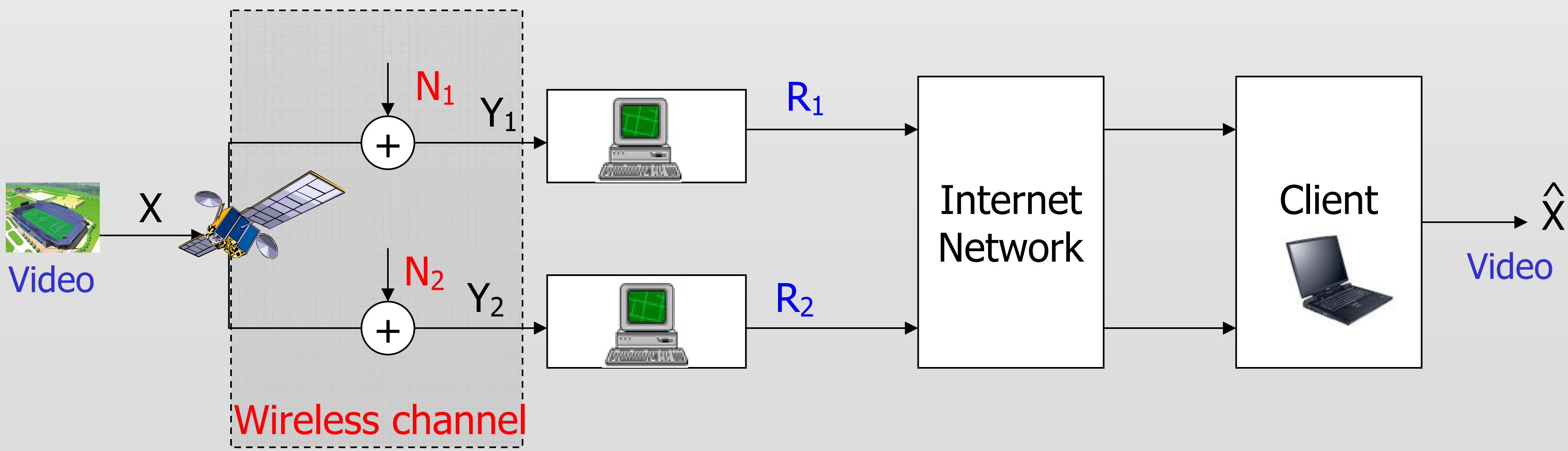
Applications

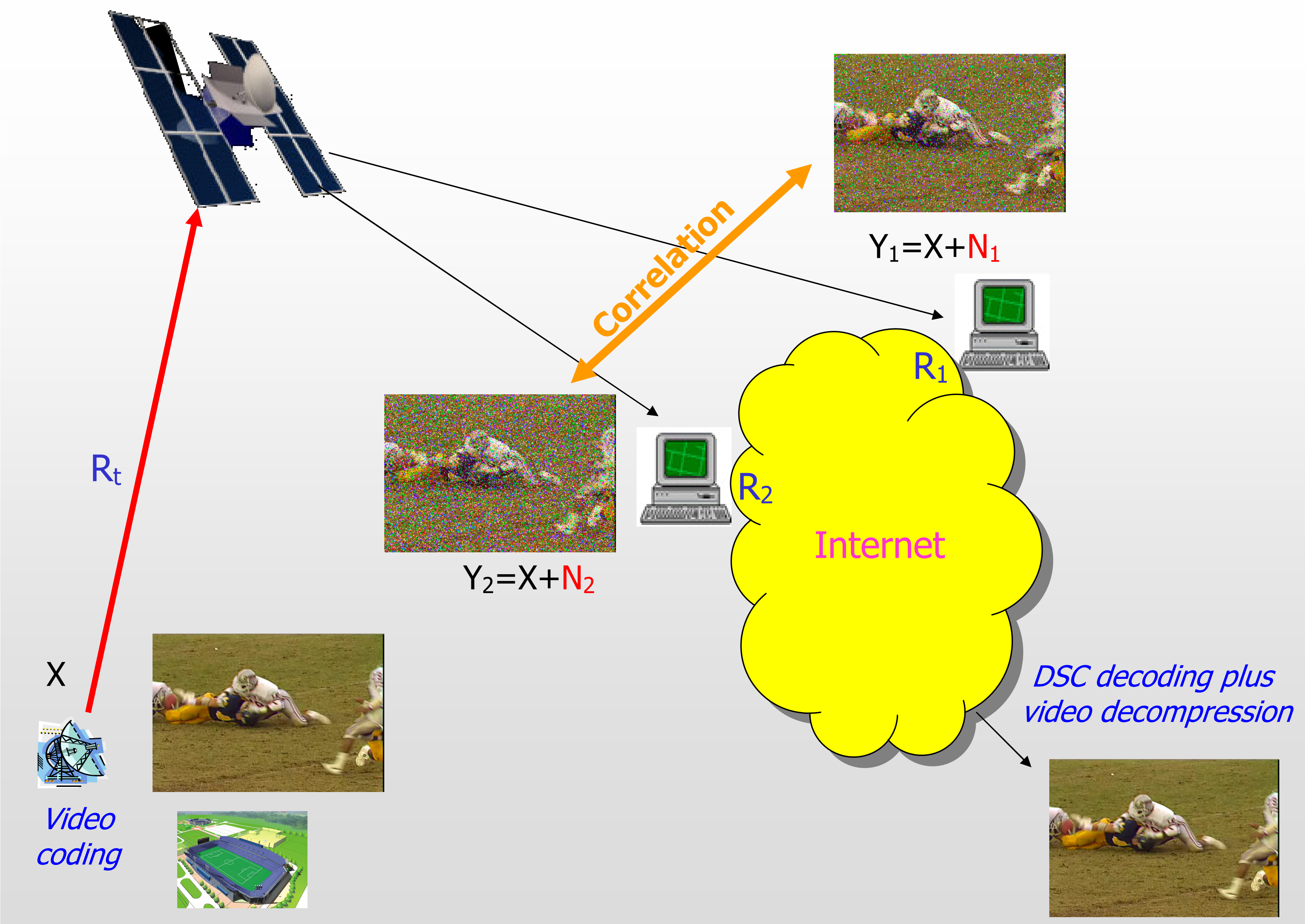
- Distributed (WZ) video coding
- Stereo Video Coding
- Multimedia streaming over heterogeneous networks
- Wireless sensor networks
- Spectrum sensing

- Efficient low-bitrate video coding (e.g., H.264/MPEG-4)
- Strong error protection
- Extremely high compression and efficient congestion control
- Fast encoding/decoding
- QoS: Digital TV quality of video
- Fit into current technologies *(HDTV, best-effort Internet, DVB-S/DVB-T)*

MT Coding-based Streaming





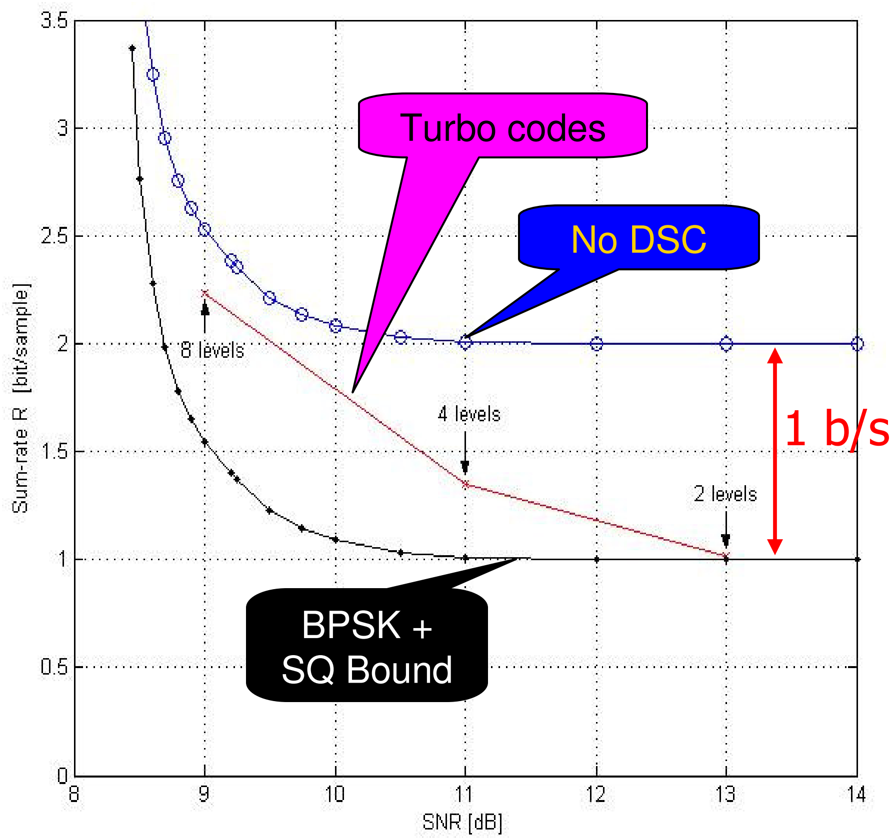


Correlation
$Y_1=X+N_1$
R_1
$Y_2=X+N_2$
R_2
Internet
R_t
X
Video coding
DSC decoding plus video decompression

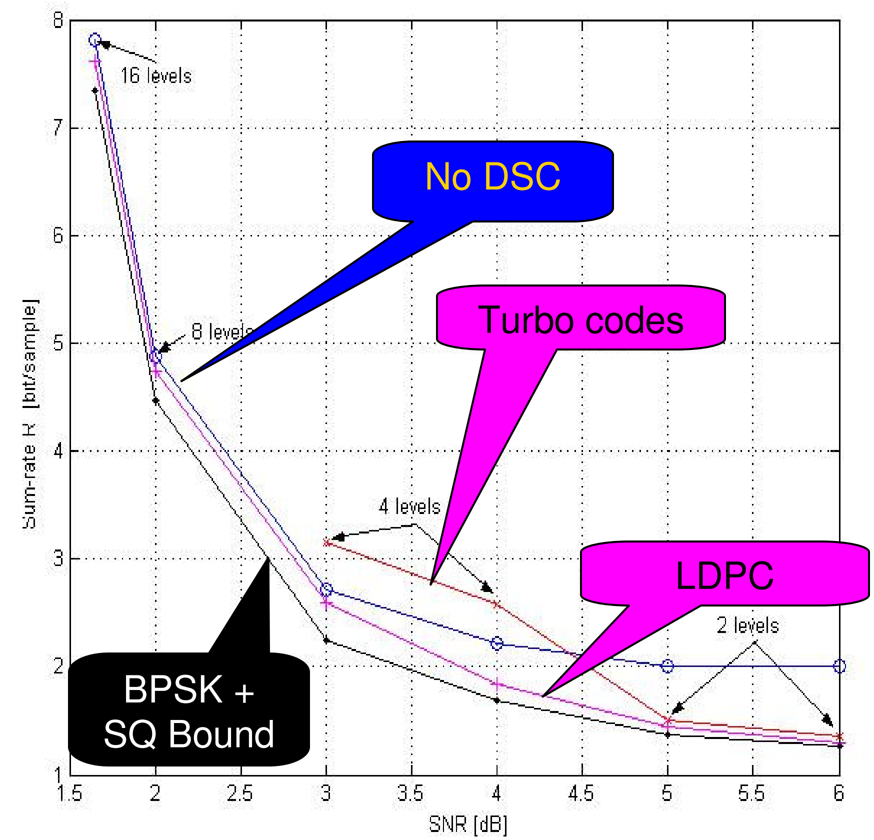
Results: AWGN Channel

(Stankovic, Yang, Xiong, 2007)

Uncoded BPSK



Convolutional codes plus BPSK

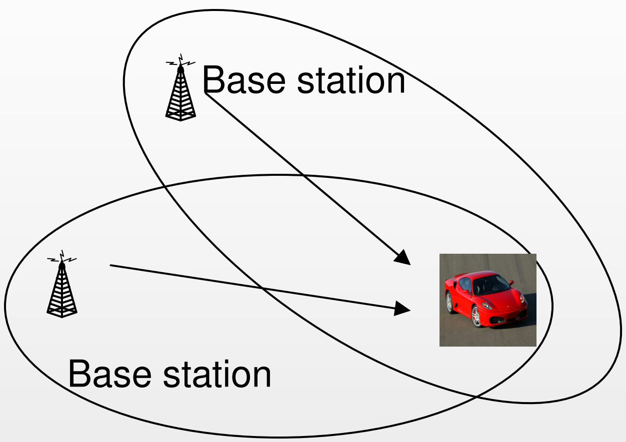


DSC with Scalar Quantization (SQ) + Turbo/LDPC codes

Advantages of the System

- Significant rate savings due to *spatial diversity gain* and *DSC* (without distortion penalty)
- Downloading from multiple servers:
  - Servers evenly loaded
  - Robustness to a server failure
  - Security
- Source-independent transmission (not limited to video or multimedia)
- Acceptable system complexity and flexible design

Multi-station Wireless Streaming



- Increased quality of the reconstructed video due to path diversity gain
- Resilience to station failure
- Traffic control is improved because the servers can be equally loaded

- Problems: multipath fading, interference, noise
- Conventional solution: spread spectrum
- Spread spectrum increases required bandwidth

Solution *(Khirallah, Stankovic et al., TWComm 2009)*

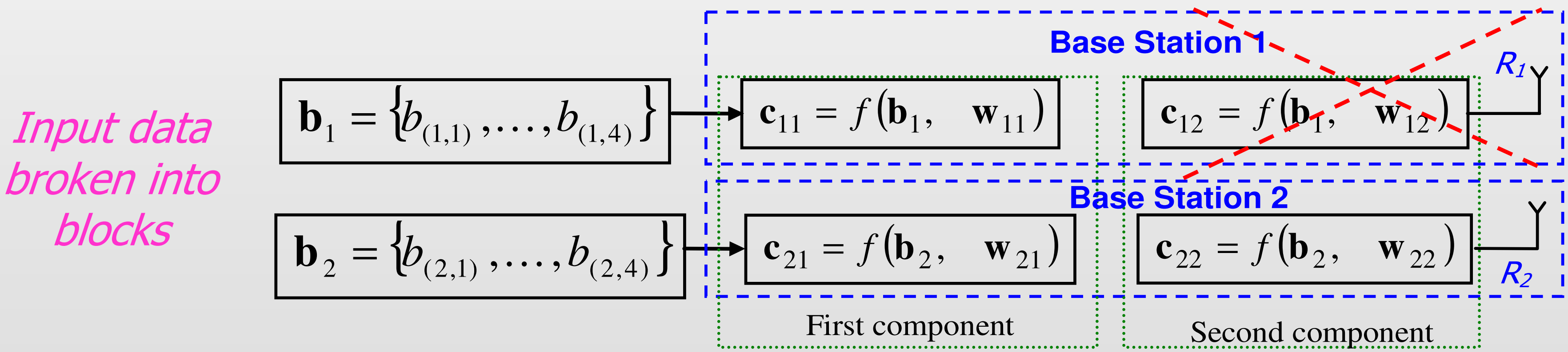
- Idea: Exploit the fact that the two stations stream same/correlated content
- Use Complete Complementary (CC) sequences for spreading at the base station (BS)
- At the encoders, puncture some of the output chips to reduce the rate
- At the decoder, recover the punctured chips using the chips received from the other stations as SI

Extending DSC idea

Puncturing at the Encoder

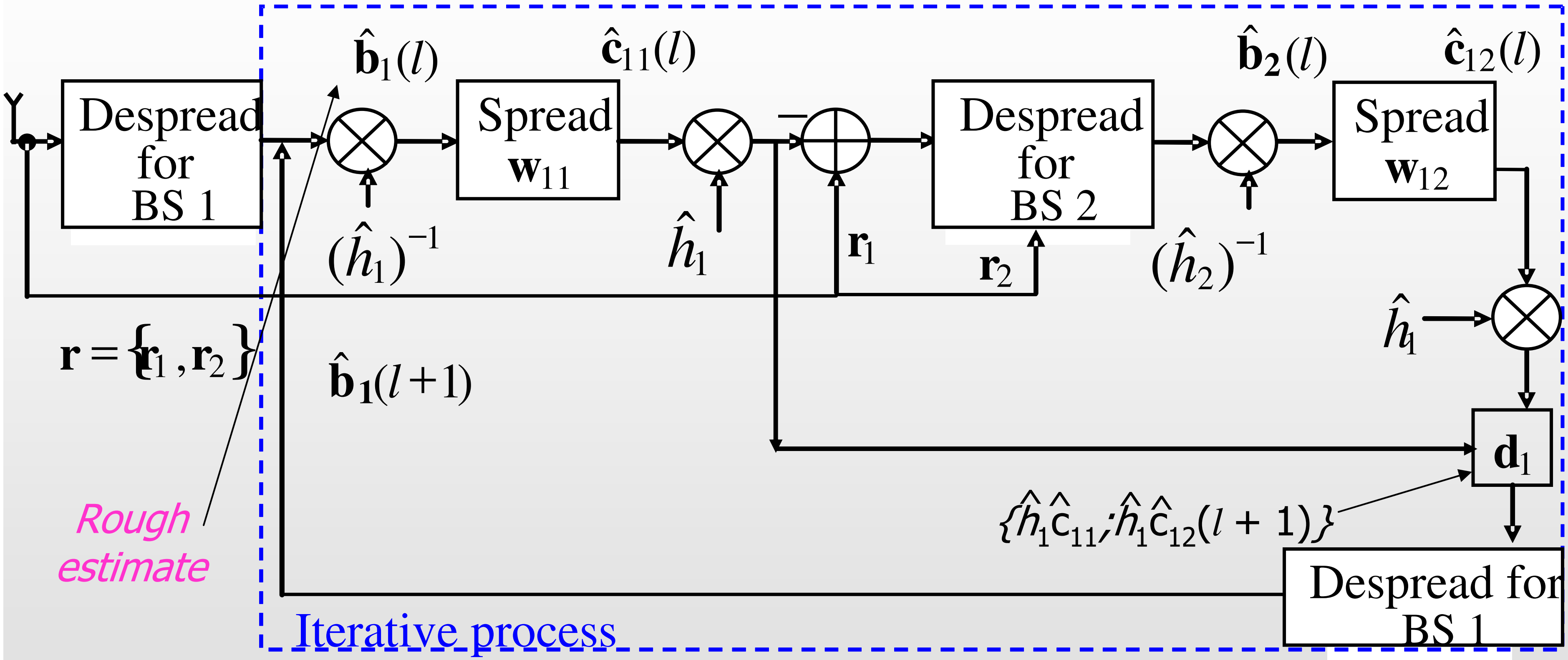
CC sequence sets of size $N = 2$ with $K_{CC} = 4$ chips per each code.

	User 1		User 2	
CC sequences sets $N = \sqrt{K_{CC}} = 2$	$\mathbf{X} = [\mathbf{w}_{11}, \mathbf{w}_{12}]$	$\mathbf{w}_{11} = [+,+,+,-]$ $\mathbf{w}_{12} = [+,-,+,+]$	$\mathbf{Y} = [\mathbf{w}_{21}, \mathbf{w}_{22}]$	$\mathbf{w}_{21} = [+,+,-,+]$ $\mathbf{w}_{22} = [+,-,-,-]$
Auto- correlation function	$\psi_{\mathbf{XX}} = \mathbf{w}_{11} \otimes \mathbf{w}_{11} + \mathbf{w}_{12} \otimes \mathbf{w}_{12} = [0,0,0,8,0,0,0]$		$\psi_{\mathbf{YY}} = \mathbf{w}_{21} \otimes \mathbf{w}_{21} + \mathbf{w}_{22} \otimes \mathbf{w}_{22} = [0,0,0,8,0,0,0]$	
Cross-correlation function	$\psi_{\mathbf{XY}} = \mathbf{w}_{11} \otimes \mathbf{w}_{21} + \mathbf{w}_{12} \otimes \mathbf{w}_{22} = [0,0,0,0,0,0,0]$			



Problem: Puncturing leads to loss of orthogonality => conventional CC decoding is not feasible

Iterative Recovery at the Decoder

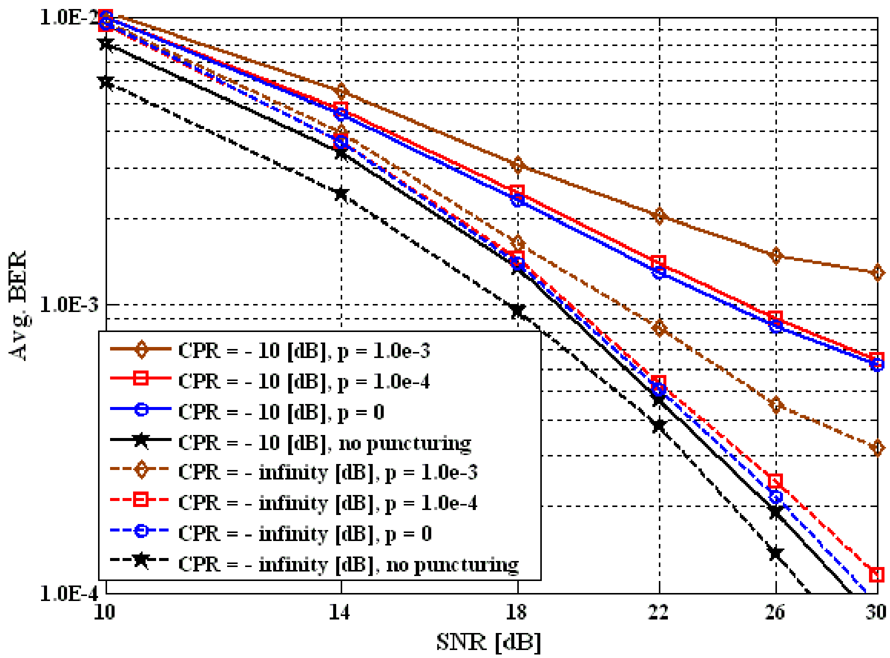


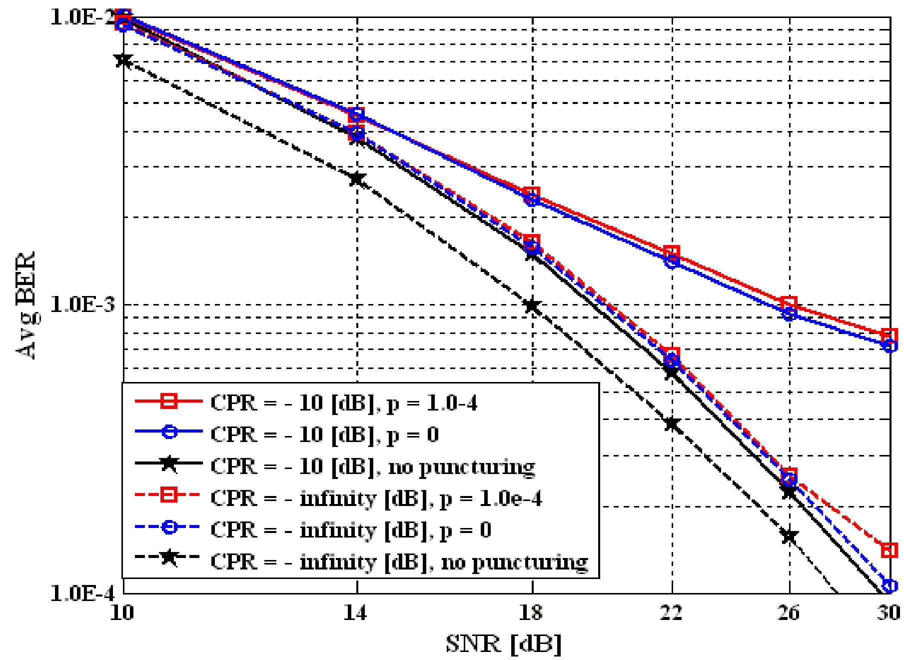
Received signal at frequency f_1: $r_1=h_1c_{11}+h_2c_{21}+z_{i1}$

and f_2: $r_2= \quad 0+ h_2c_{22}+z_2$

h_1, h_2 are fading coefficients, z_1 and z_2 AWGN, l designates iteration number

Results *(Khirallah, Stankovic et al., TWComm 2009)*





Relative speed: 30km/hr

Relative speed: 120km/hr

CPR: channel power ratio (the average power ratio between the second and first path), CPR=10dB frequency selective, CPR=∞ flat-fading channel

$p = 0$: streaming of two identical sources

$p > 0$: streaming of two sources correlated by a binary symmetric channel with crossover probability p

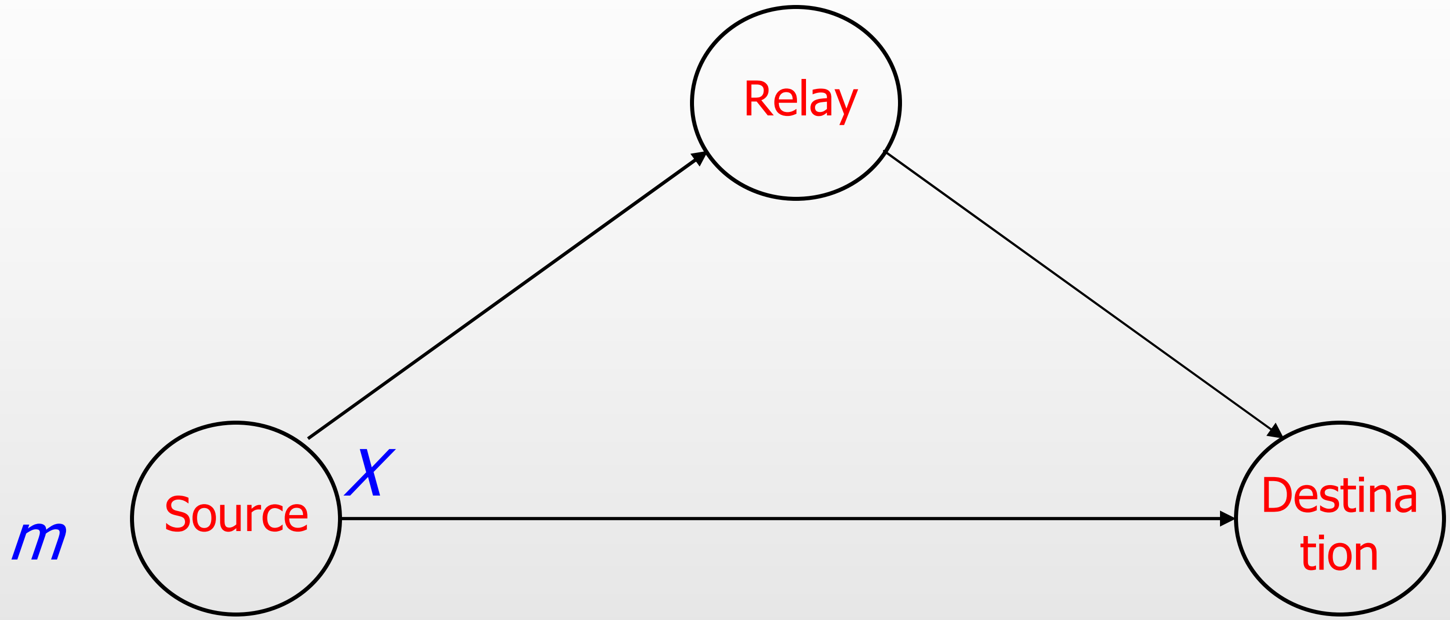
Applications

- Distributed (WZ) video coding
- Stereo Video Coding
- Multimedia streaming over heterogeneous networks
- Wireless sensor networks
- Spectrum sensing

Wireless Sensor Networks (WSN)

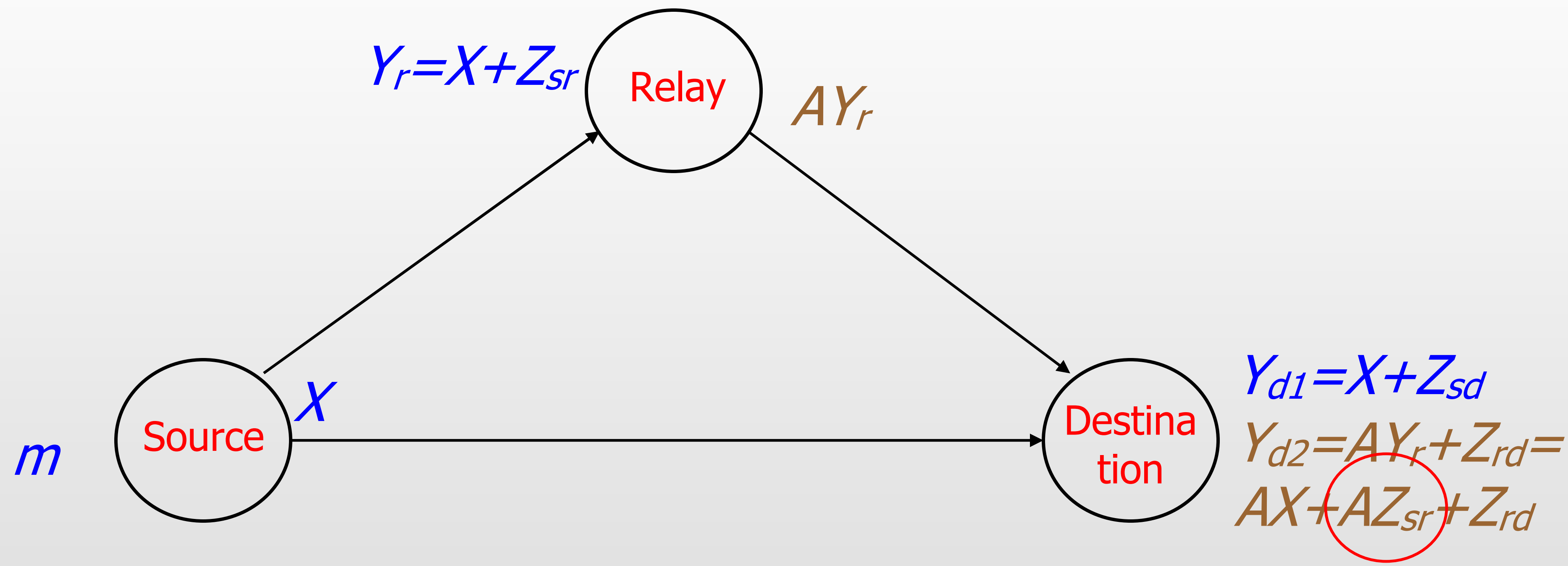
- Networks of numerous tiny, low-power and low-cost devices
- Key requirement: reduce power consumption by reducing communication via efficient distributed compression
- In dense sensor network, measurements of neighbouring sensors are expected to be correlated, hence DSC the most efficient compression choice
  - *R. Cristescu et al. "Networked SW" (joint optimization of placement, routing, and compression in WSN)*
  - *J. Liu et al. "Optimal communication cost in WSN"*

The Relay Channel



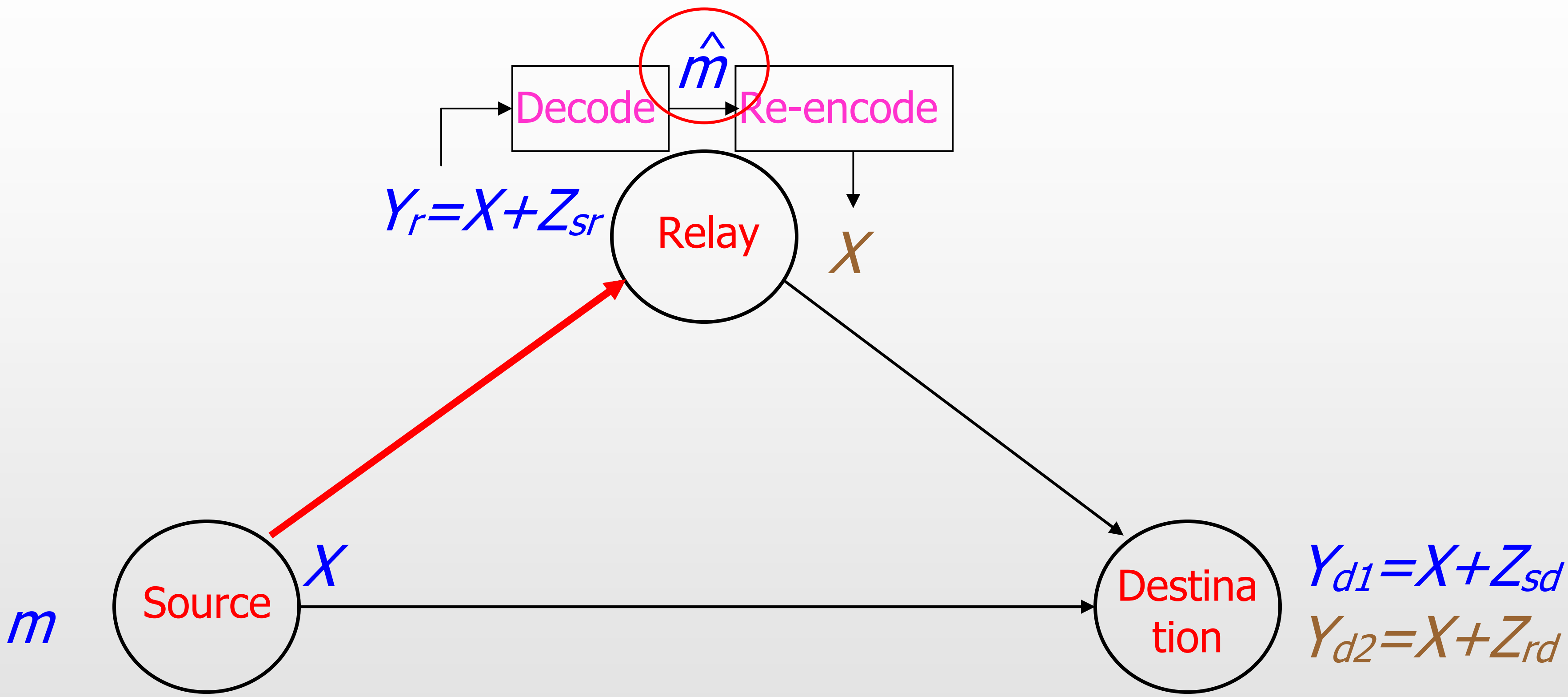
Task: Transmit messages *m* to the destination with the help of a relay. Noisy wireless channels assumed for all three links

The Relay Channel



AMPLIFY-AND-FORWARD CODING

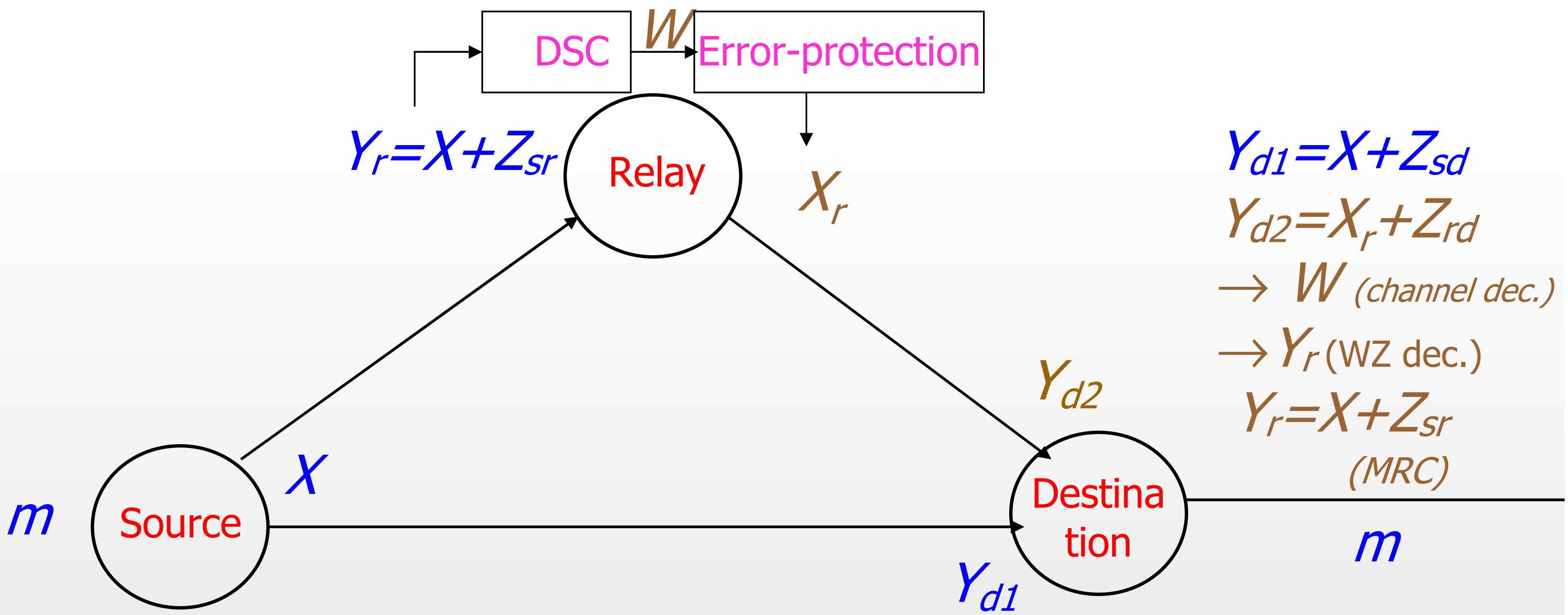
The Relay Channel

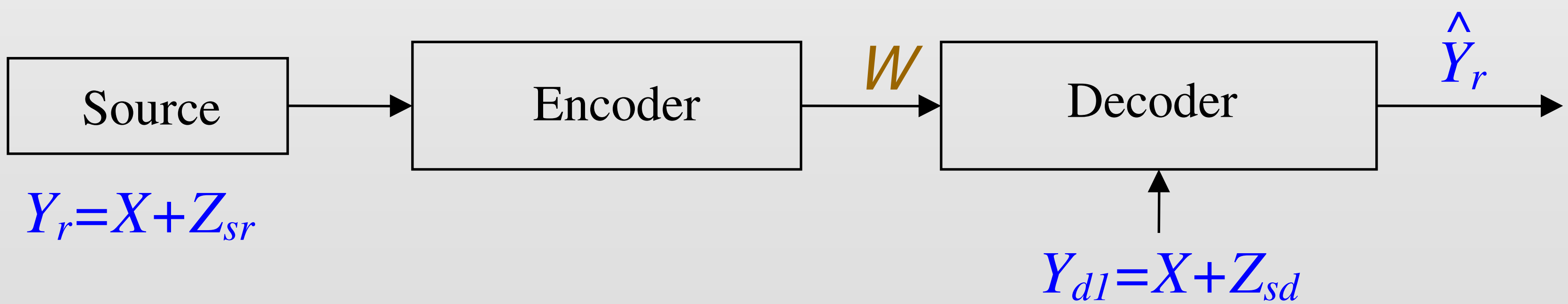


DECODE-AND-FORWARD (DF) CODING

Problem: The rate is limited by the capacity of source-relay link!

Solution: Avoid decoding at the relay node!

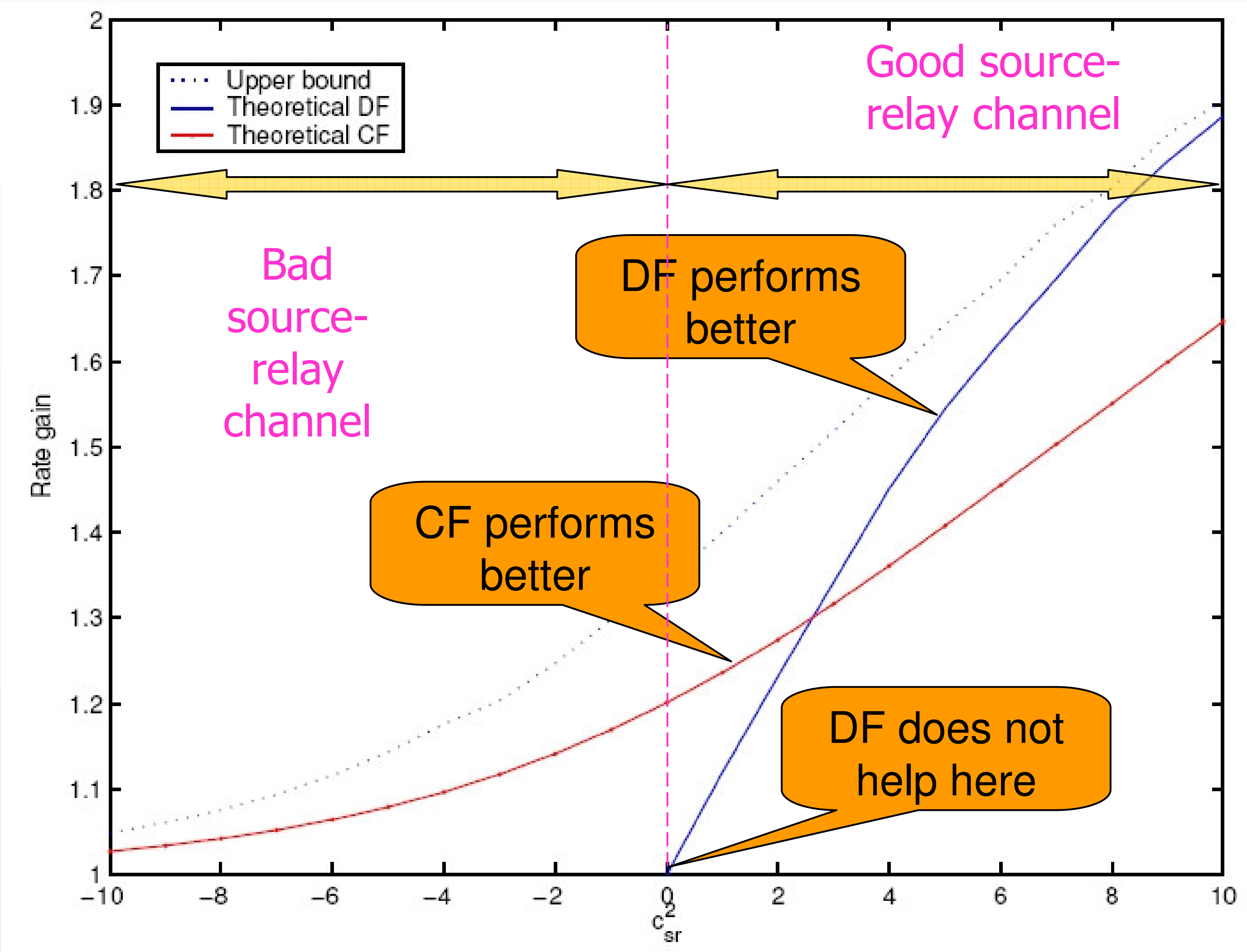




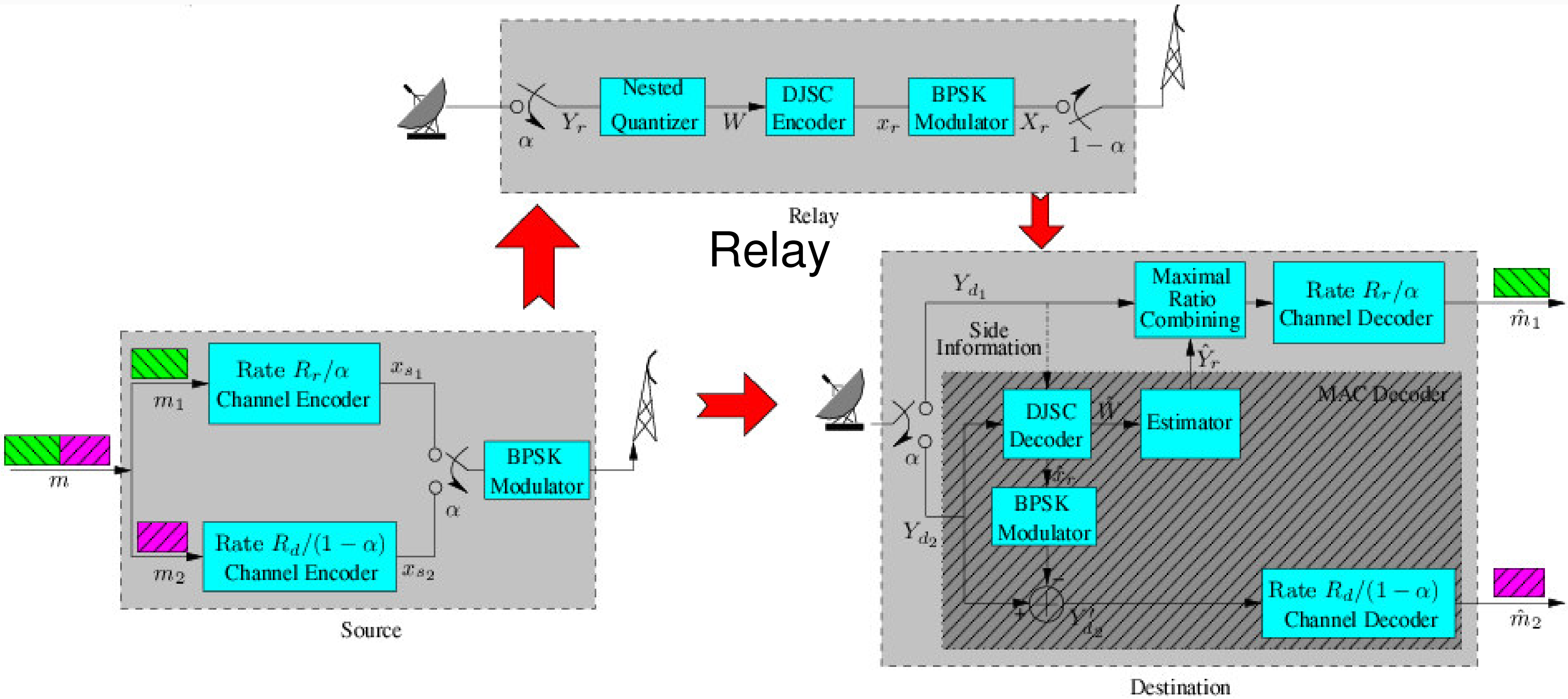
COMPRESS-AND-FORWARD (CF) CODING

(Stankovic et al. SPM 2006)

Half-Duplex Gaussian Relay Channel: The Rate Bounds



Practical CF Code Design

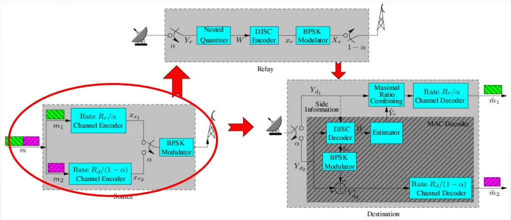


Source

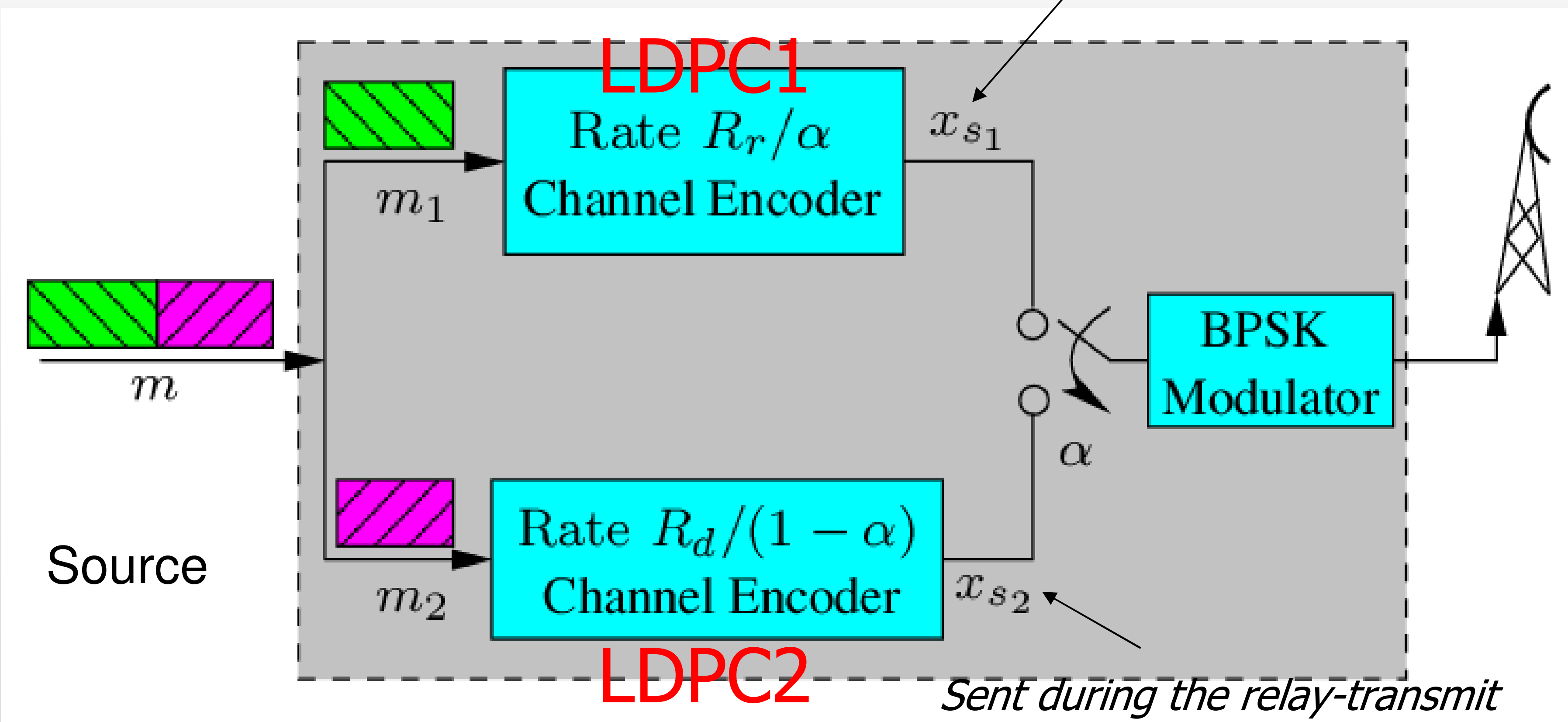
Destination

(Liu, Uppal, Stankovic, and Xiong, ISIT 2008)

Practical CF: The Source

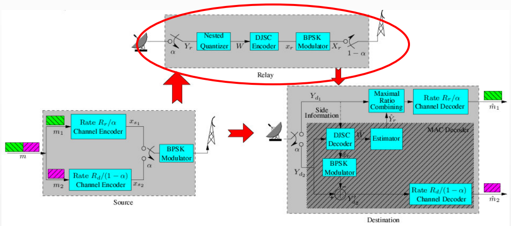


Sent during the relay-receive period αT

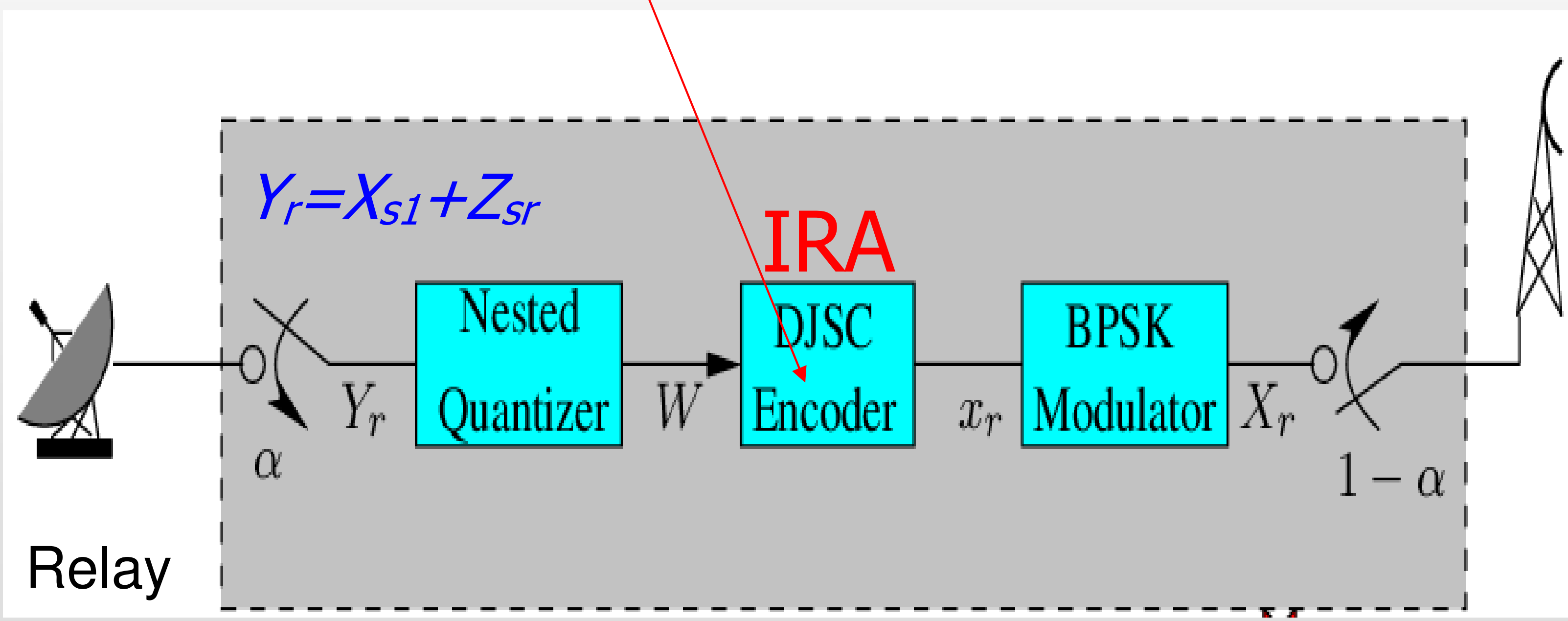


Sent during the relay-transmit period (1-α)T

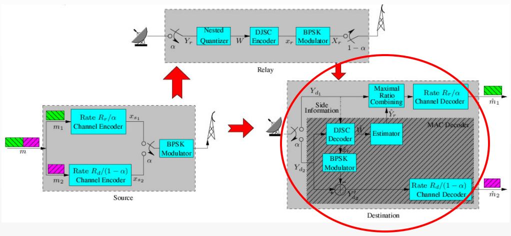
Practical CF: The Relay

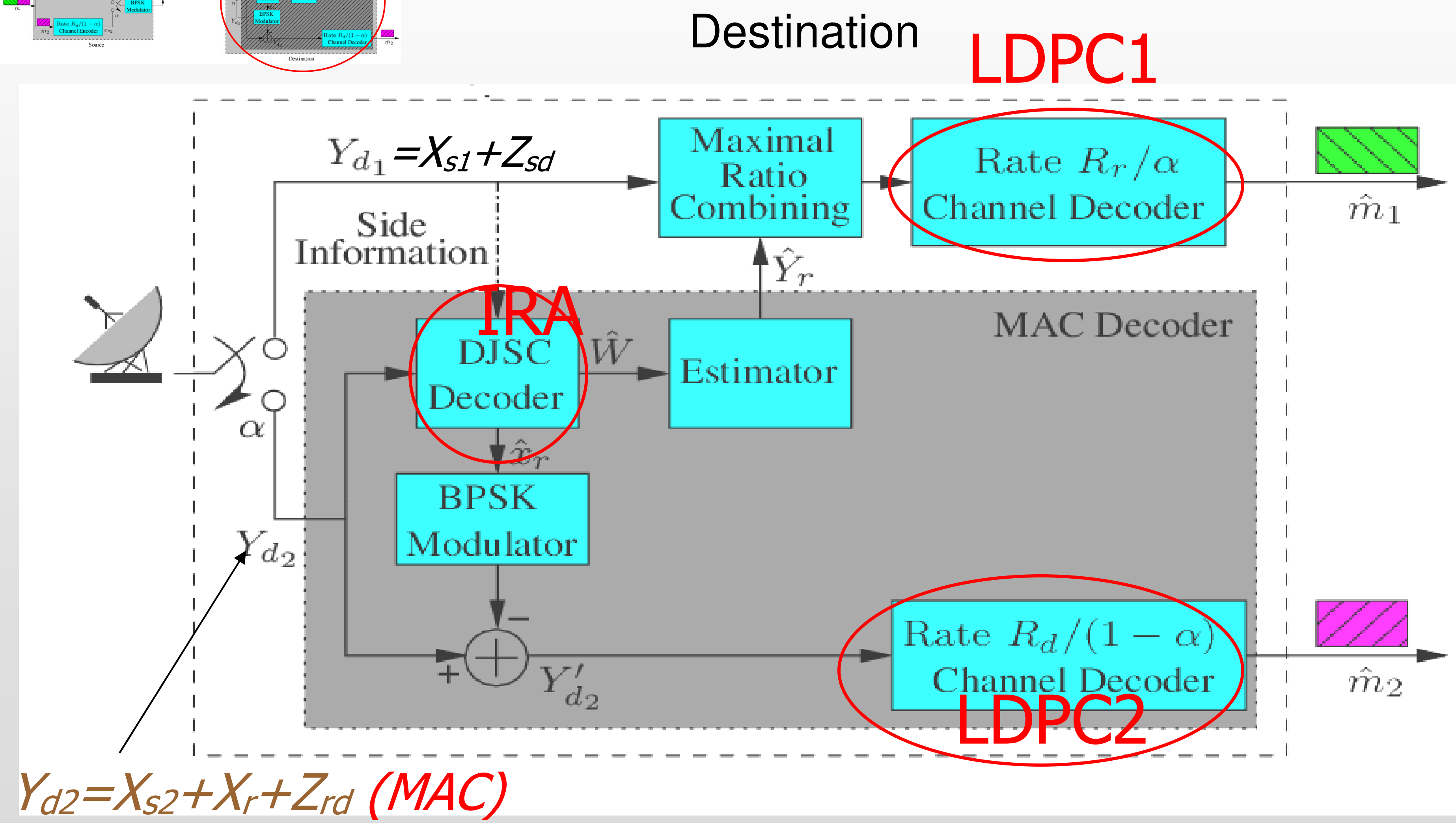


Joint DSC and error protection using a single IRA code

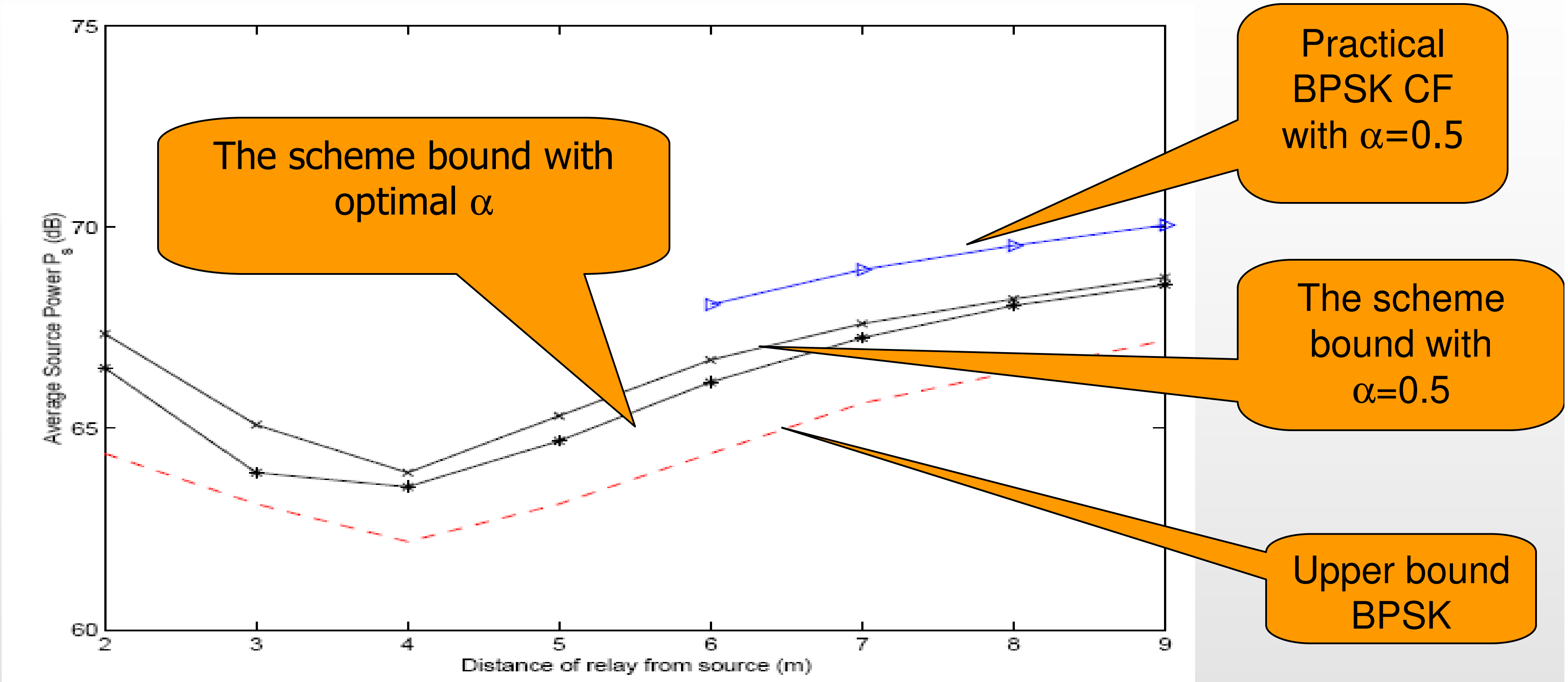


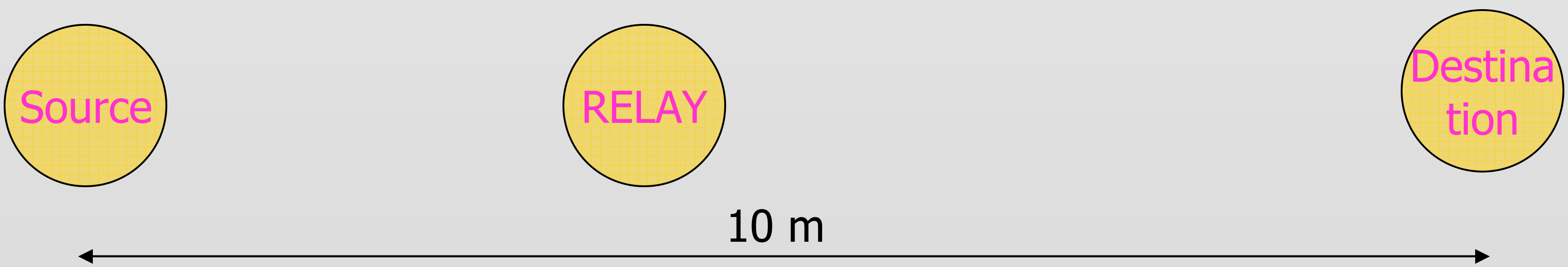
Practical CF: The Destination





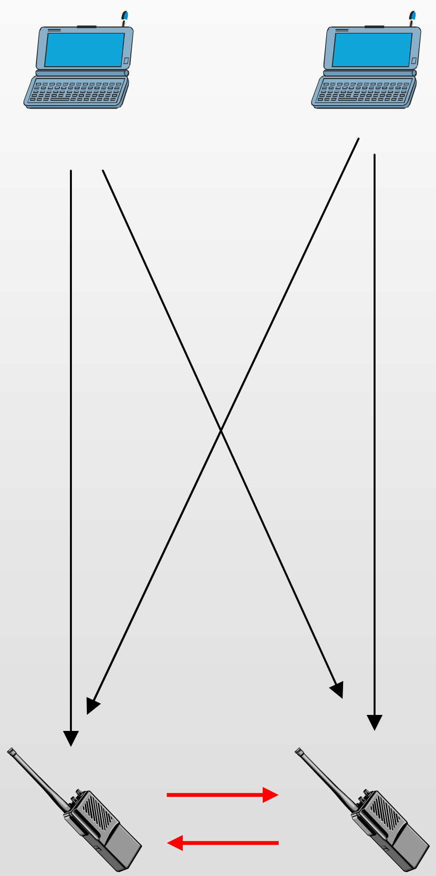
Results: Half-duplex AWGN Relay

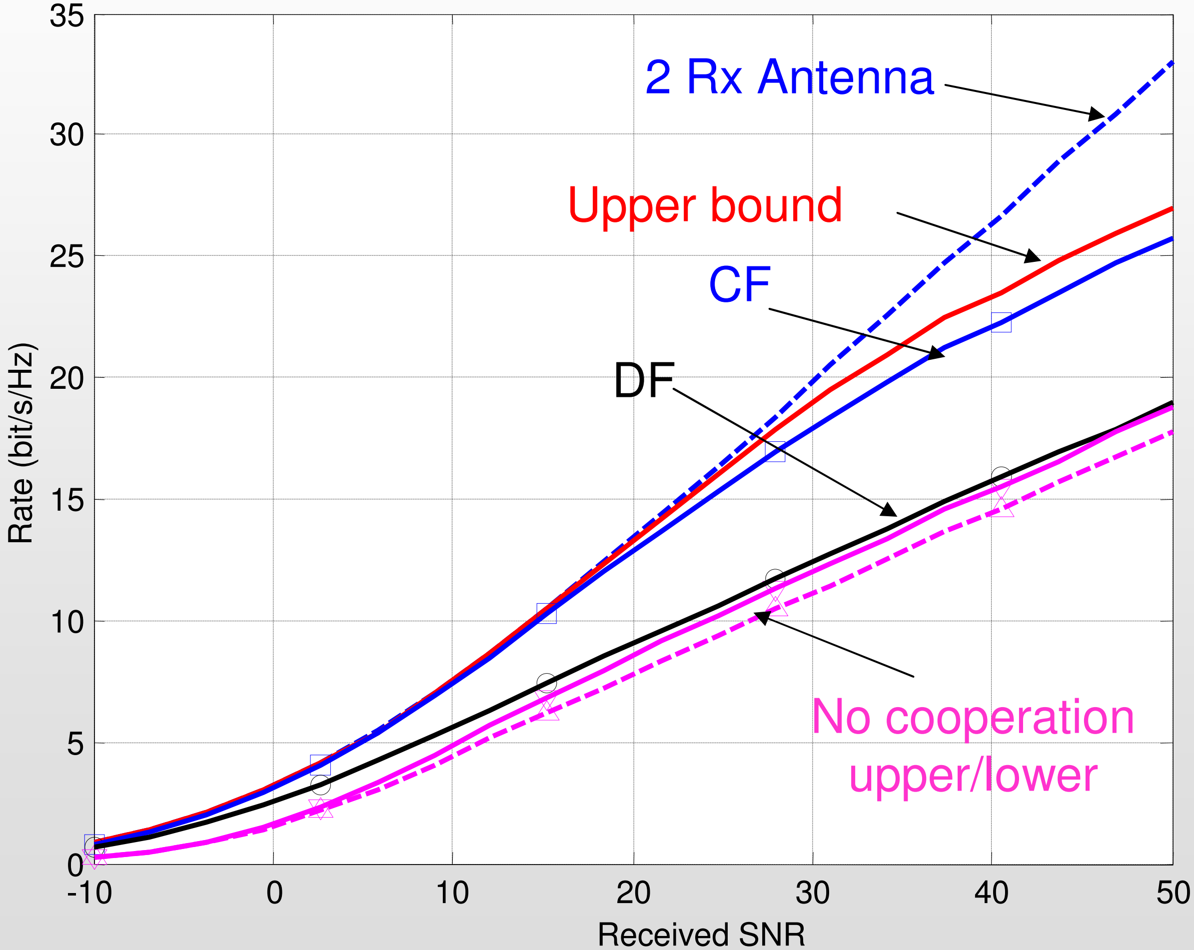




(Liu, Uppal, Stankovic, and Xiong, ISIT 2008)

Receiver Cooperation: Great Promise of CF



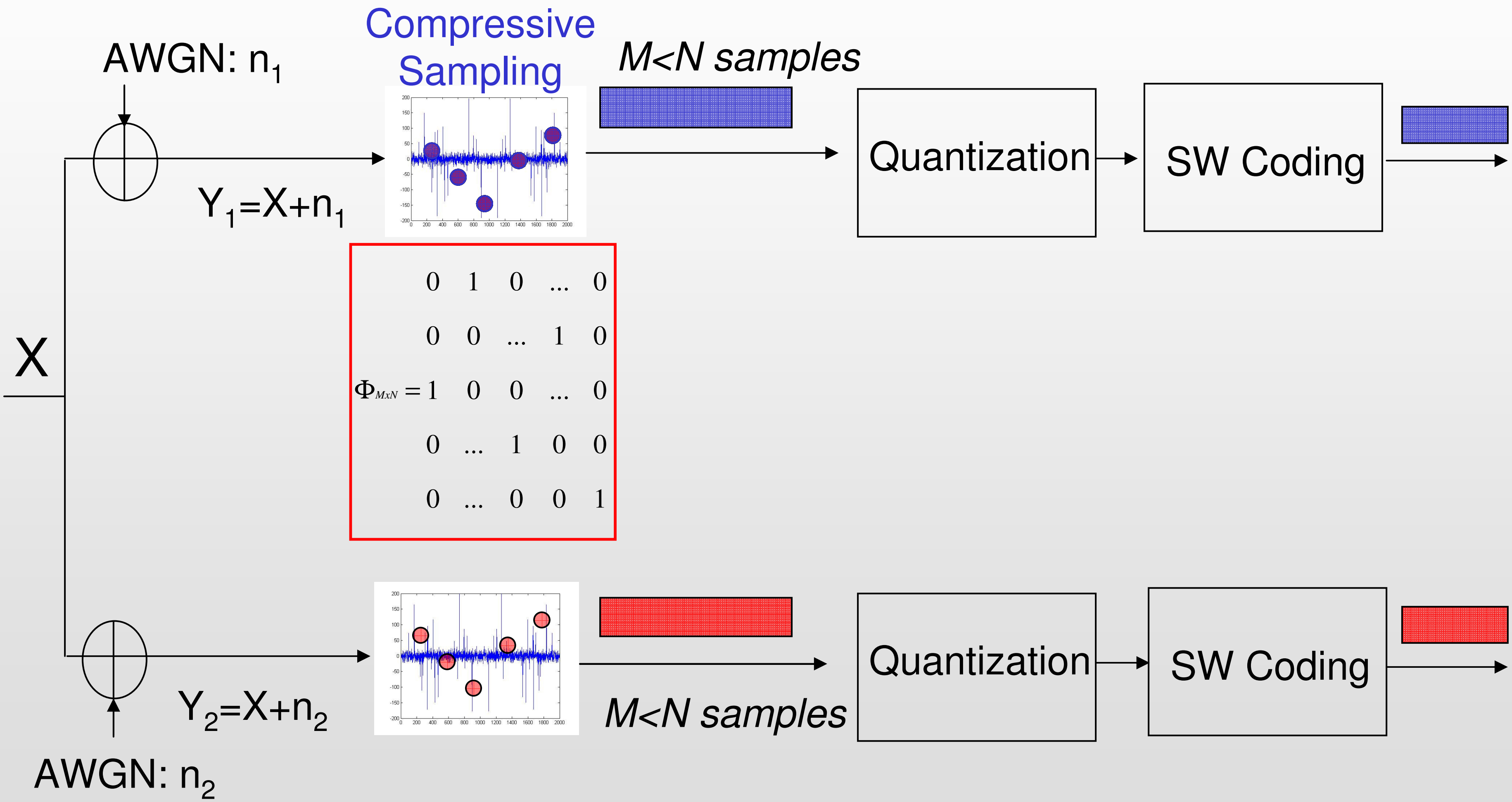


Almost 20dB gain of CF over DF!

Applications

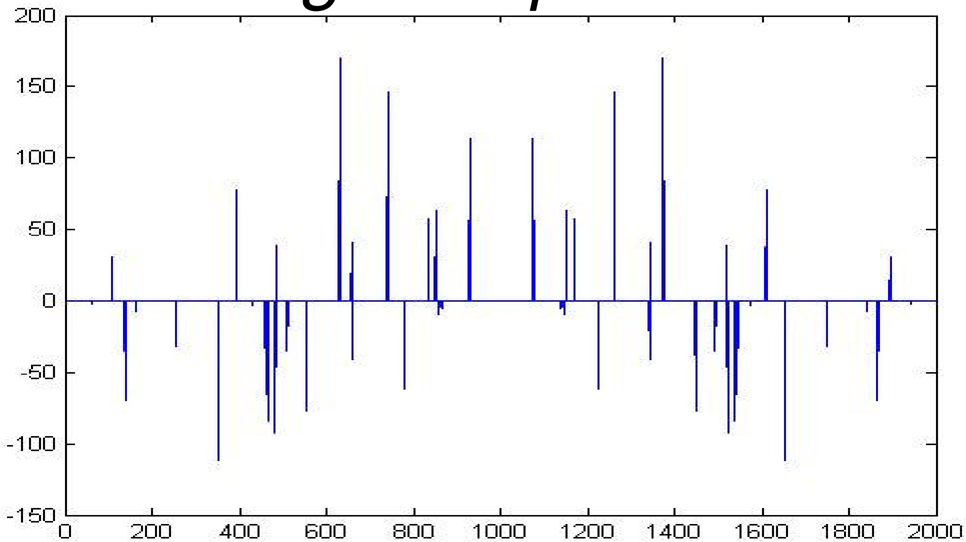
- Distributed (WZ) video coding
- Stereo Video Coding
- Multimedia streaming over heterogeneous networks
- Wireless sensor networks
- Spectrum sensing

Cognitive Radio (CR) Spectrum Sensing



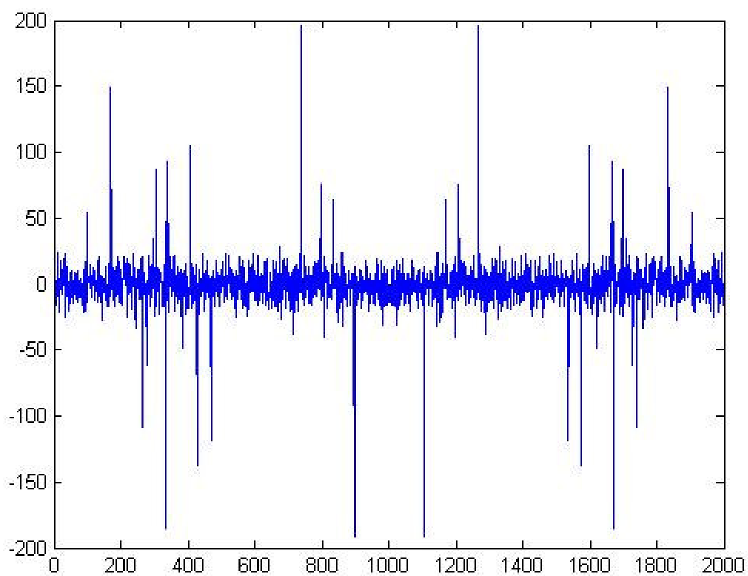
X: Sparse in the frequency domain

Original Spectrum

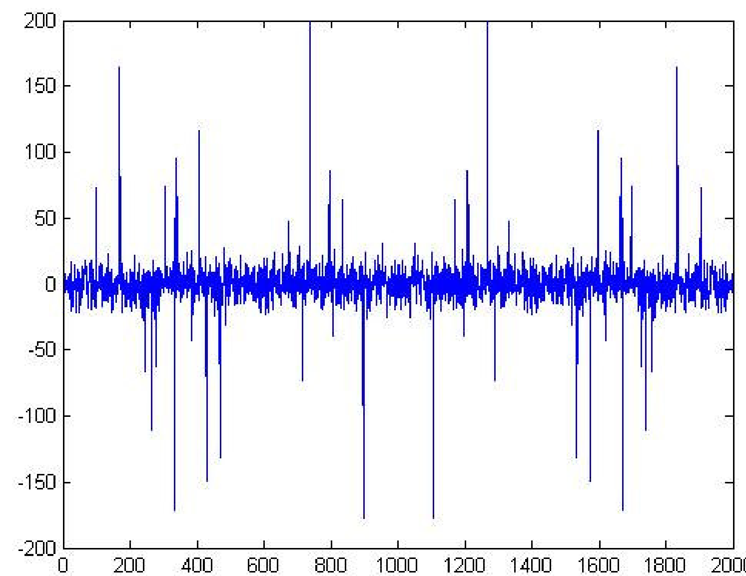


X_f

Observed Spectrum



$Y1_f$



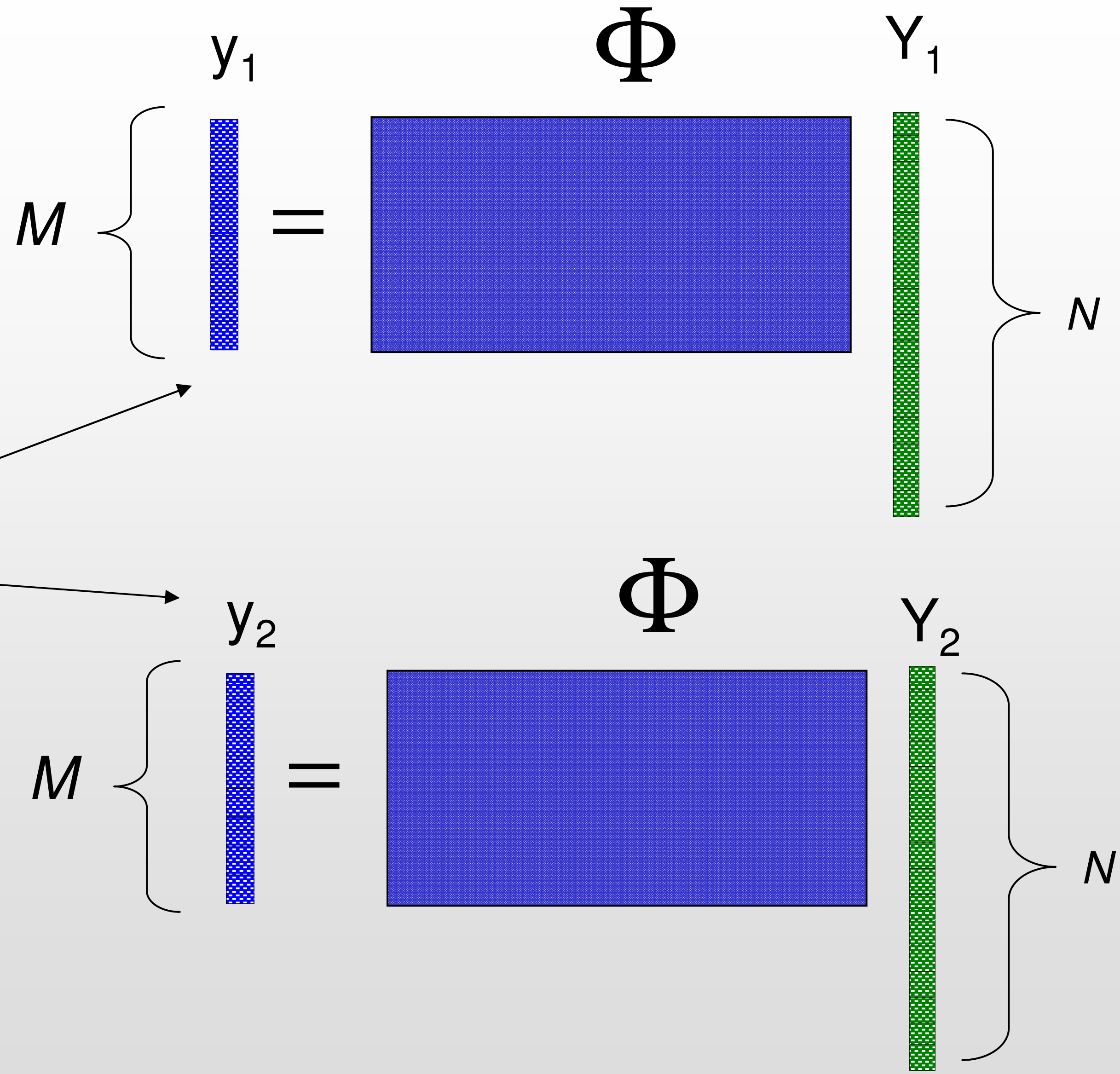
$Y2_f$

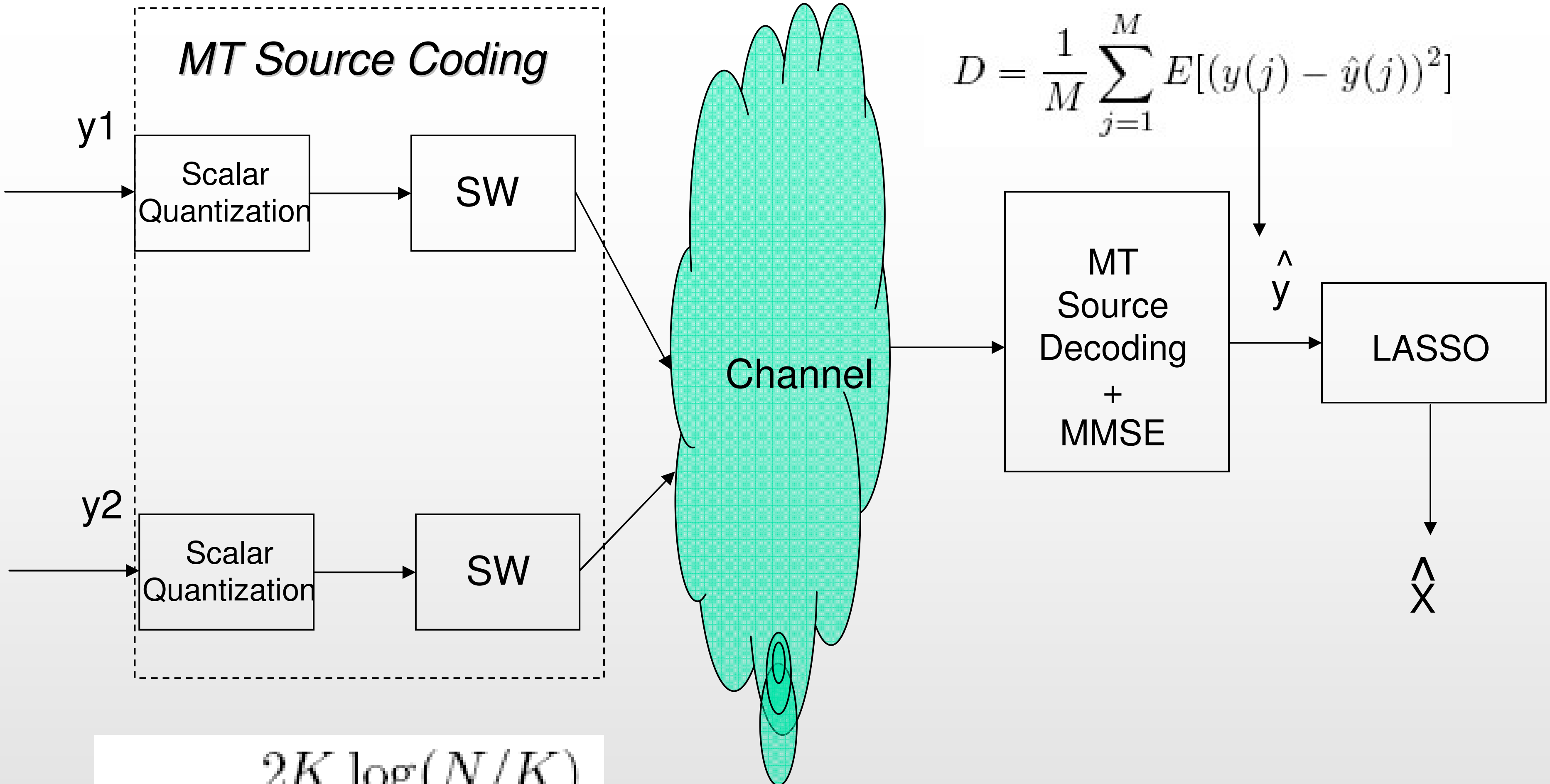
Highly correlated:

$y_1 \sim X + n_1$

$y_2 \sim X + n_2$

Indirect MT source coding





$$D = \frac{1}{M}\sum_{j=1}^{M} E[(y(j) - \hat{y}(j))^2]$$

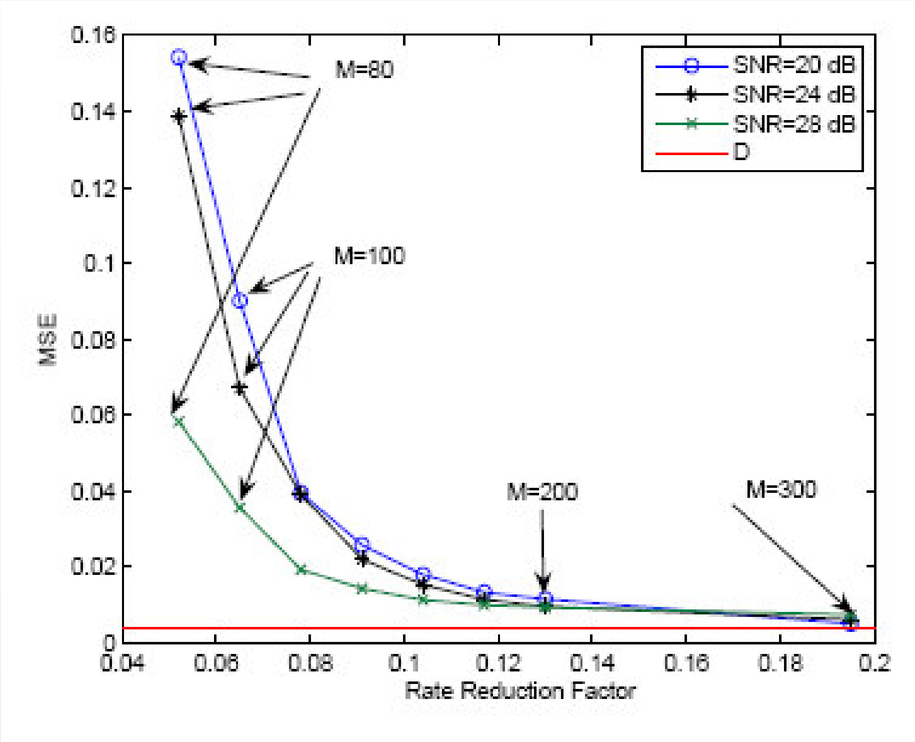
$$M \geq \frac{2K\log(N/K)}{\log(1 + P_y/D)}$$

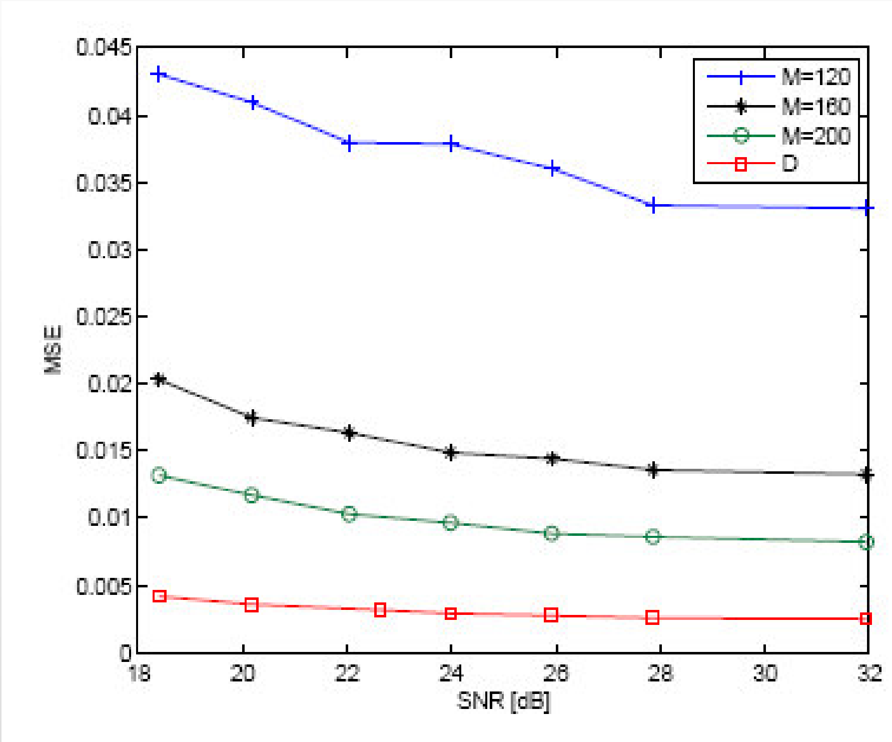
$$R = \frac{1}{2}\log^{+}\frac{4P_y(10^{\sigma/M} - 1)}{(A - P_{n_1}P_{n_2}\frac{10^{\sigma/M}-1}{P_y})^2}$$

MT compression sum-rate

Compression vs. acquisition tradeoff

N = 1000, K=25





MT source coding: Uniform scalar quantization + Turbo codes

(Cheng, Stankovics IEEE SPL 2009)

Final Remarks

- Concept of Distributed Source Coding
  - SW, WZ and MT coding
- Code designs based on channel codes over a "correlation channel"
  - SW: Hamming, Turbo, LDPC, Raptor codes
  - WZ: Nested Quantization + SW
- Many practical challenges
- Still a long way to go (a lot of information theory, not much practice)…