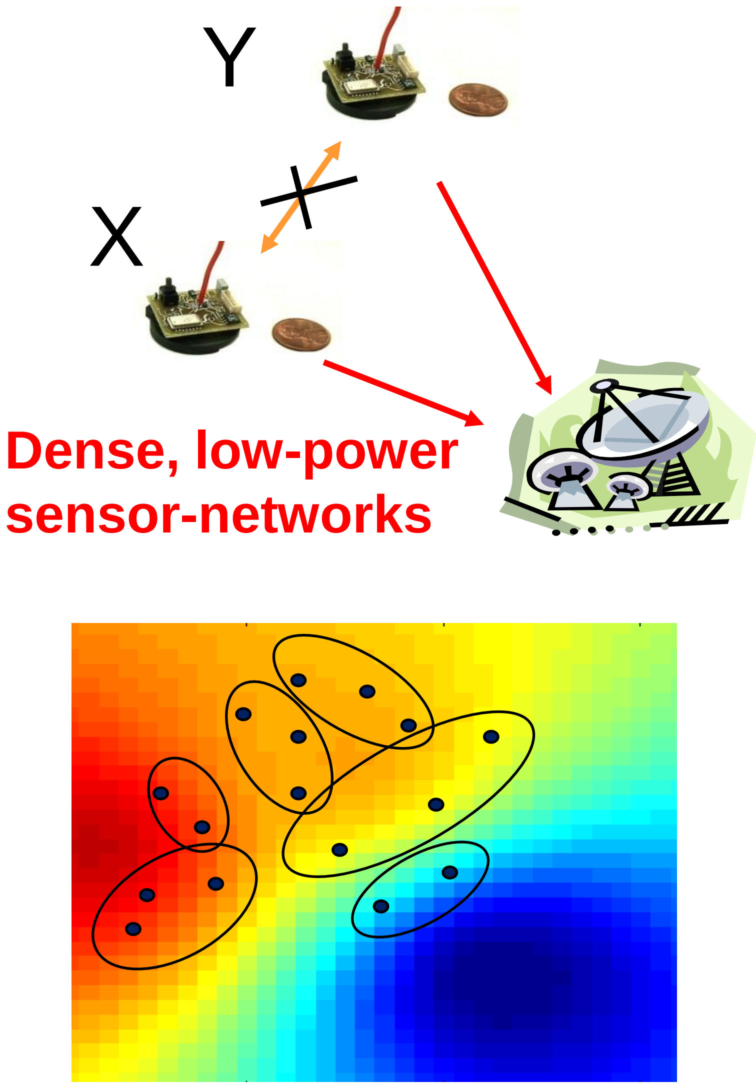
Distributed Source Coding: Foundations, Constructions and Applications

Kannan Ramchandran, UC Berkeley
S. Sandeep Pradhan, UM Ann Arbor

Acknowledgements

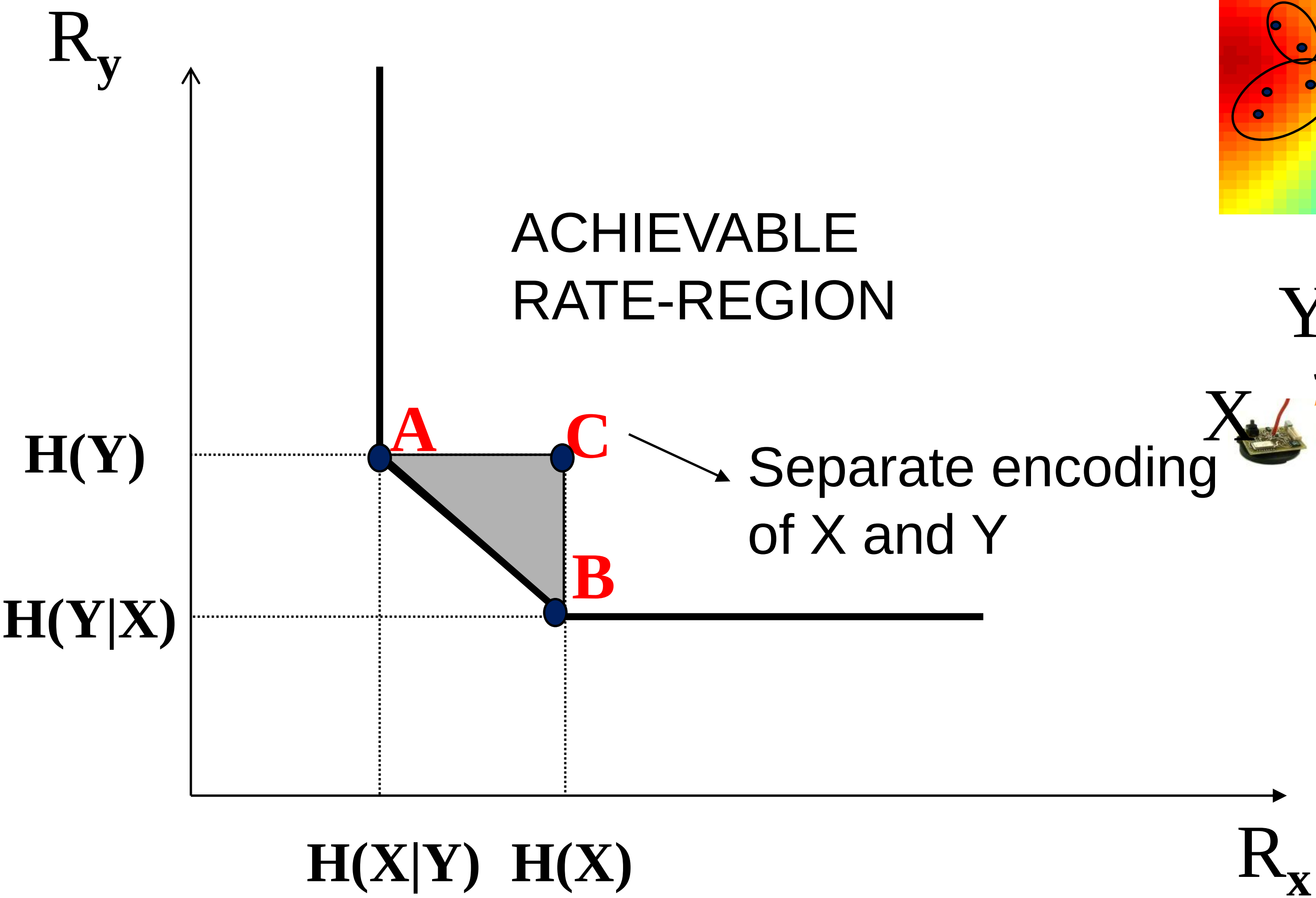
- Dinesh Krithivasan (UM)
- Prof. Prakash Ishwar (BU, formerly UCB)
- Prof. David Neuhoff (UM)
- Dr. Vinod Prabhakaran (UIUC, formerly UCB)
- Dr. Rohit Puri (Cisco, formerly UCB)
- June Wang (UCB)
- Chuohao Yeo (UCB)
- …
- as well as our collaborators on this topic….

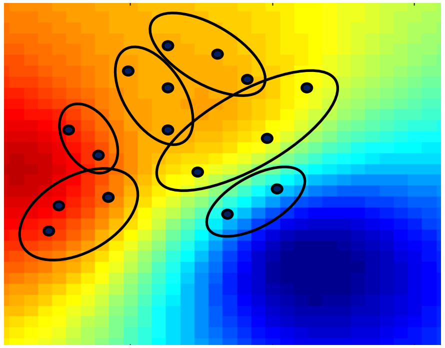
Motivation: sensor networks

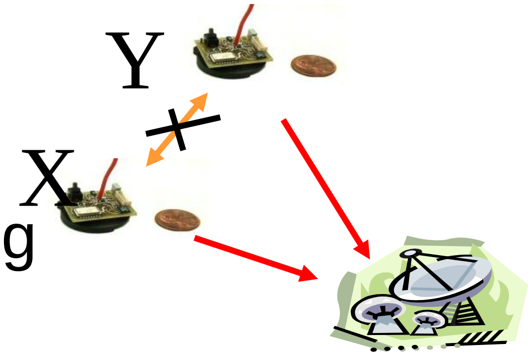


- Consider correlated nodes X, Y
- Communication between X and Y expensive.
- Can we exploit correlation without communicating?
- Assume Y is compressed independently. How to compress X close to H(X|Y)?
- Key idea: discount I(X;Y).
  $H(X|Y) = H(X) - I(X;Y)$

Distributed source coding: Slepian-Wolf '73

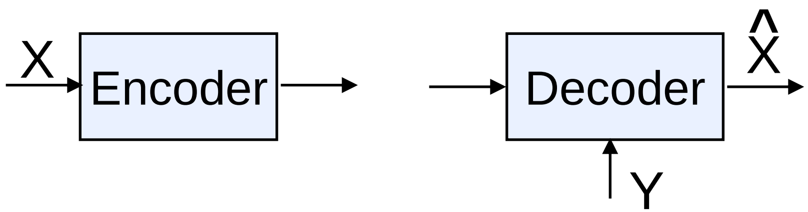






Distributed source coding

Source coding with side information: (Slepian-Wolf, '73, Wyner-Ziv, '76)

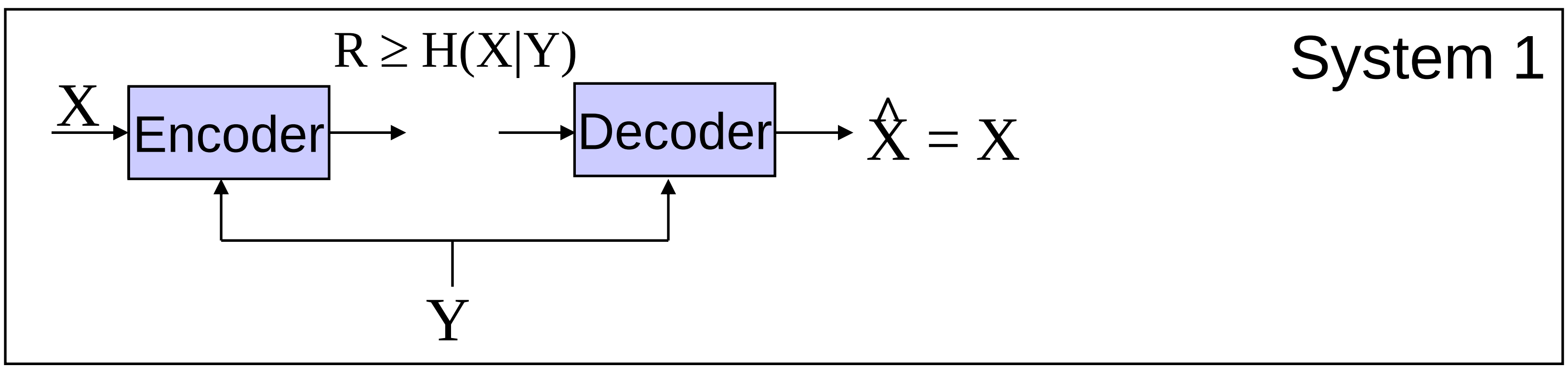


- X and Y are correlated sources.
- Y is available only to decoder.

- ***Lossless coding (S-W):*** **no loss of performance over when Y is available at both ends if the statistical correlation between X and Y is known.**
- ***Lossy coding (W-Z):*** **for Gaussian statistics, no loss of performance over when Y known at both ends.**
- **Constructive solutions:** *(Pradhan & Ramchandran (DISCUS) DCC '99 , Garcia-Frias & Zhao Comm. Letters '01, Aaron & Girod DCC '02, Liveris, Xiong & Georghiades DCC '03,…)*
- **Employs statistical instead of deterministic** mindset.

Example: 3-bit illustration

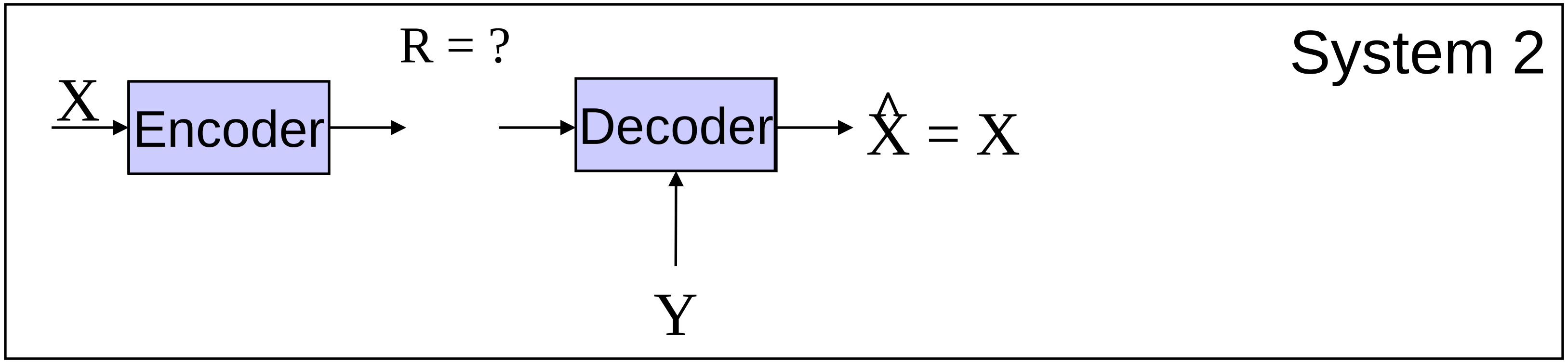
- Let X and Y be length-3 binary data (equally likely), with the correlation: Hamming distance between X and Y is at most 1.
- Example: When X=[0 1 0], Y is equally likely to be [0 1 0], [0 1 1], [0 0 0], [1 1 0].



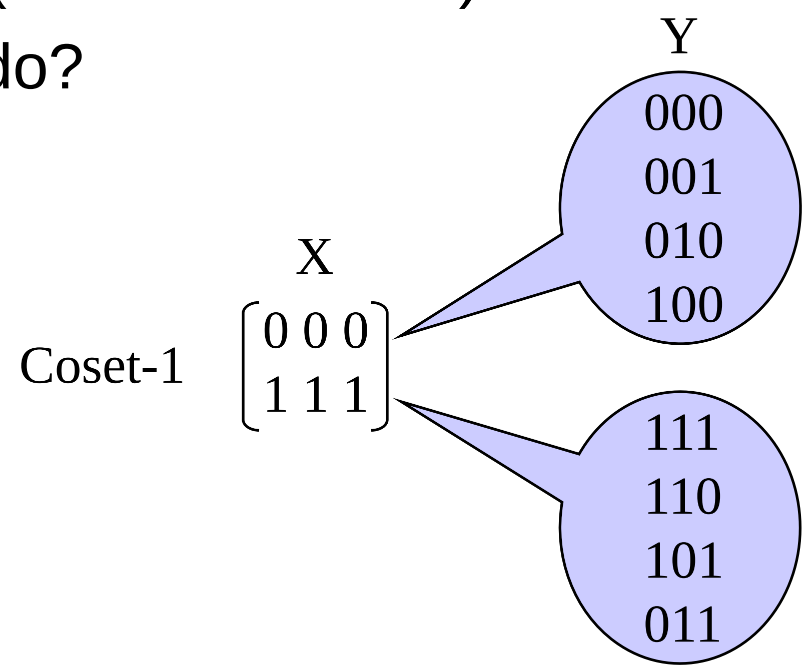
$$X+Y=\begin{cases} 0\ 0\ 0 \\ 0\ 0\ 1 \\ 0\ 1\ 0 \\ 1\ 0\ 0 \end{cases}$$

Need 2 bits to index this.

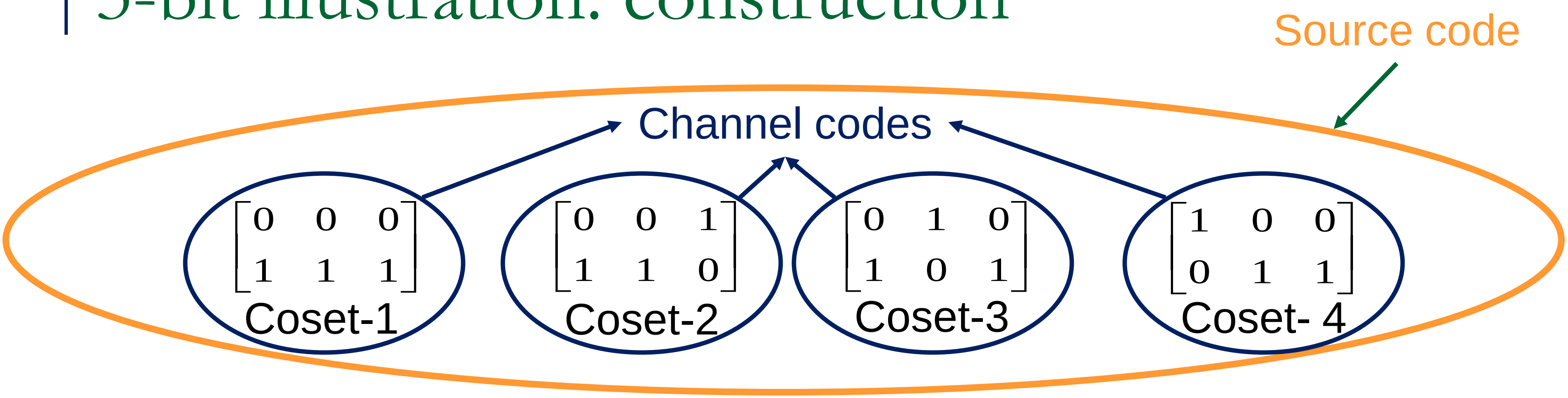
Example: 3-bit illustration



- X and Y are correlated
- Y is available only at decoder (side information)
- What is the best that one can do?
  - The answer is still 2 bits!
  - How?



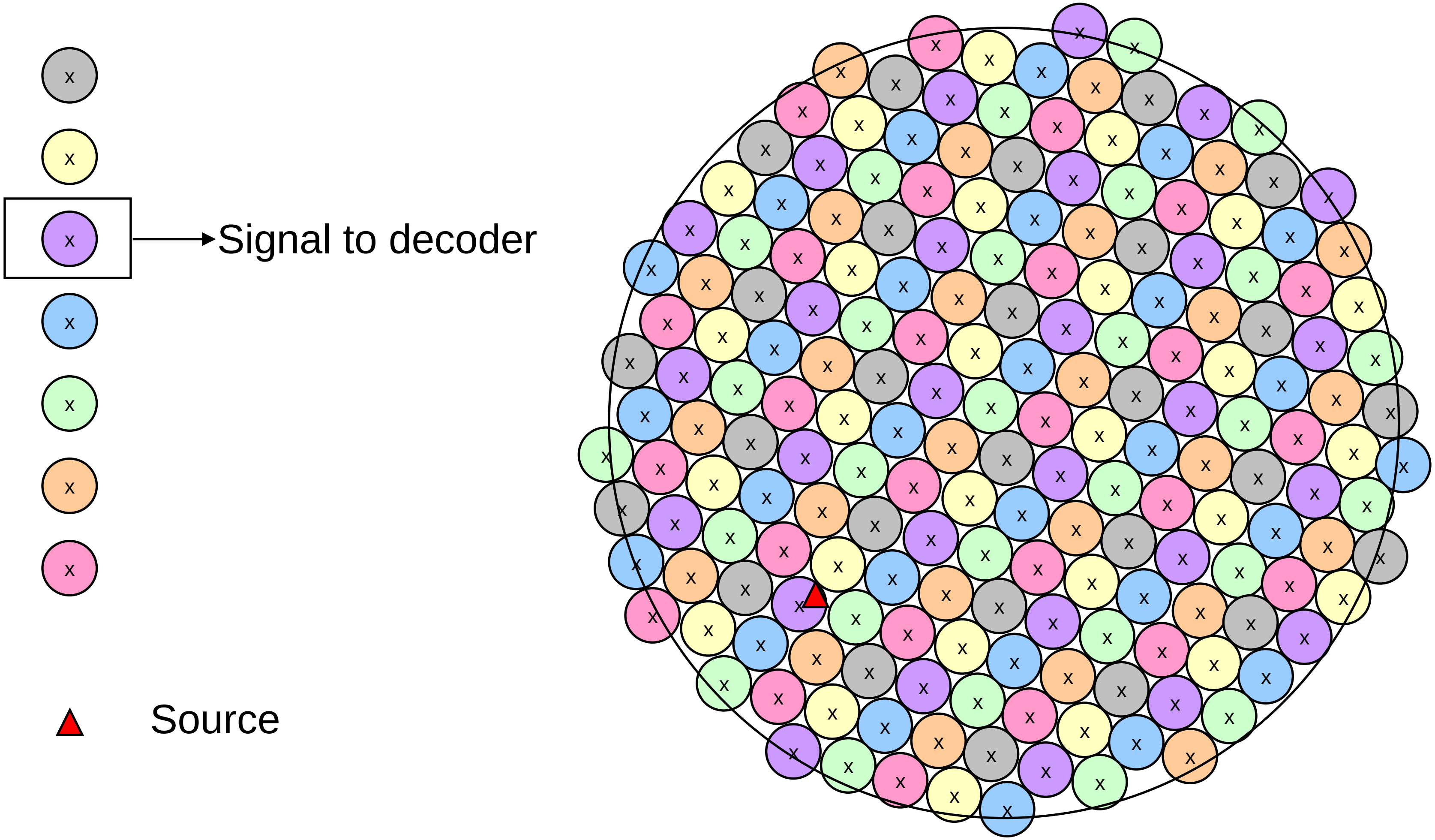
3-bit illustration: construction



- Encoder: sends the index of the coset (bin) containing X.
- Decoder: using index and Y, decode X without error.
- Coset 1 is a length-3 repetition code
- Each coset has a unique associated “syndrome”

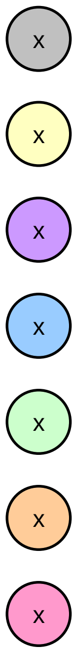
Use of syndromes in IT literature: Wyner ’74, Csiszar ‘82
Practical code construction (DISCUS): SP& KR ‘99

Example: geometric illustration



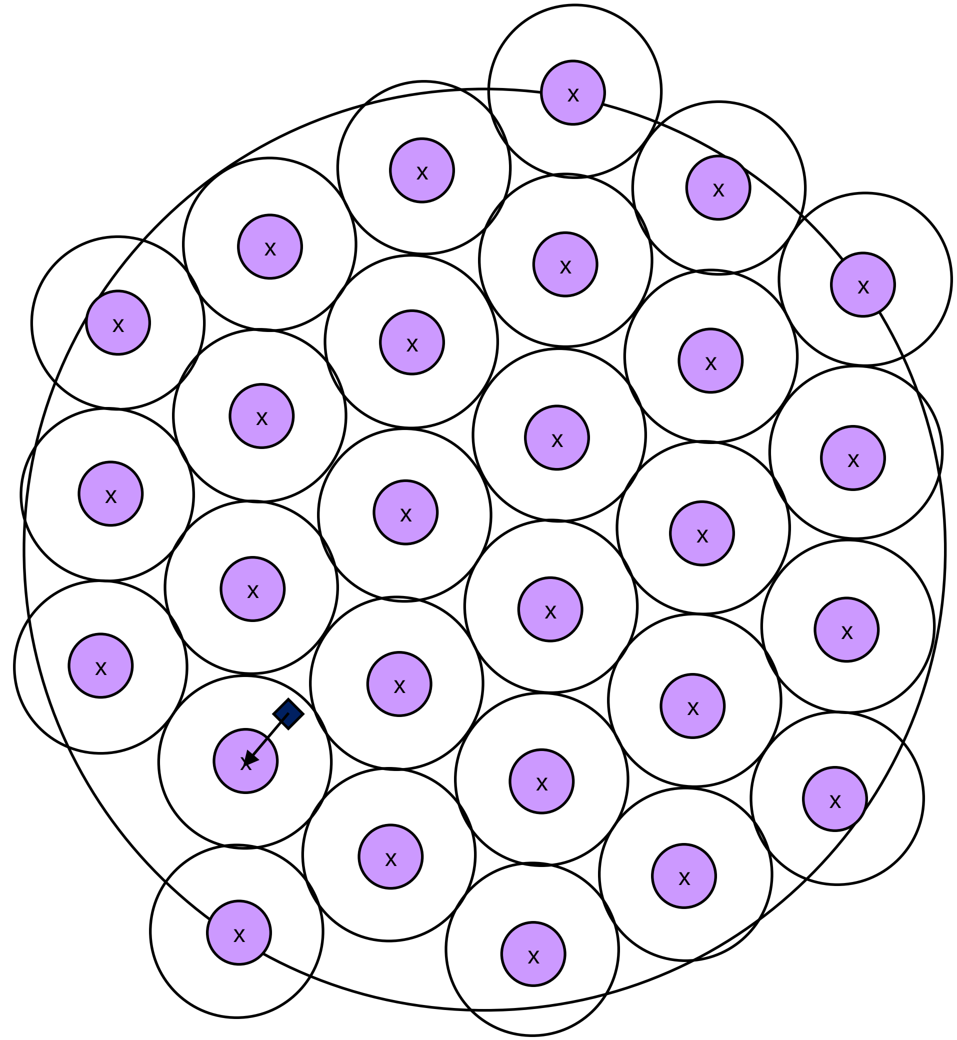
Assume signal and noise are Gaussian, iid

Example: geometric illustration



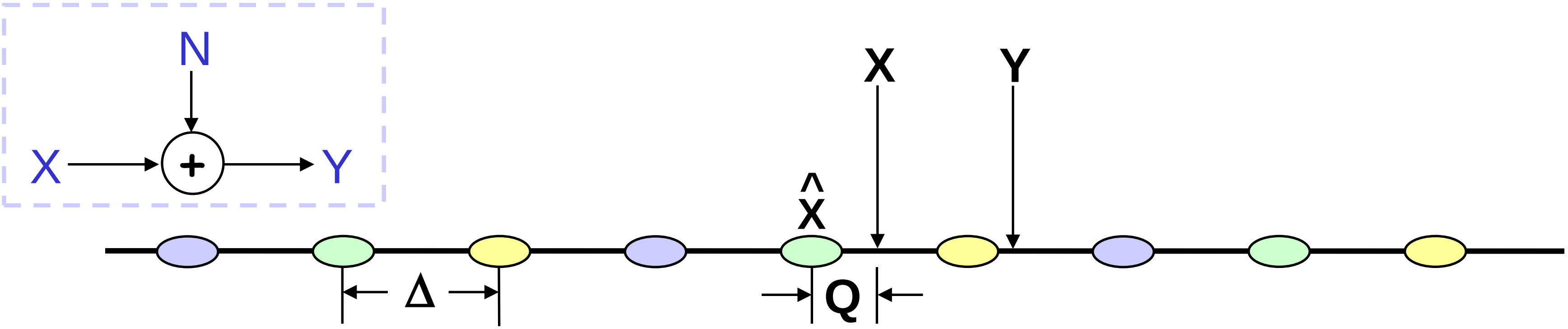
Source

Side information

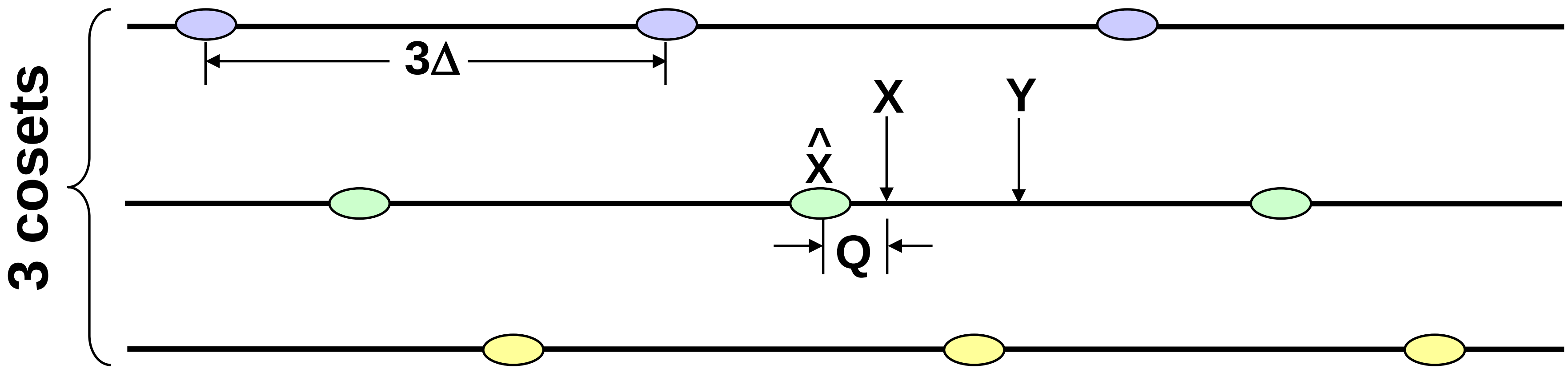


Assume signal and noise are Gaussian, iid

Example: scalar Wyner-Ziv



Partition



- Encoder: send the index of the coset (log23 bits)
- Decoder: decode X based on Y and signaled coset

Outline

- Session I. Introduction and theory : 9.00 am-10.00 am
  - Motivation and intuition
  - Distributed source coding foundations
  - Break: 10.00-10.10 am
- Session II. Constructions: 10.10 am-10.50 am
  - Structure of distributed source codes
  - Constructions based on trellis codes
  - Constructions based on codes on graphs
  - Break: 10.50-11.00 am
- Session III. Connections and Applications: 11.00 am-12.00 noon
  - Overview of connections and applications with snippets
  - Compression of encrypted data
  - Distributed video coding

Outline

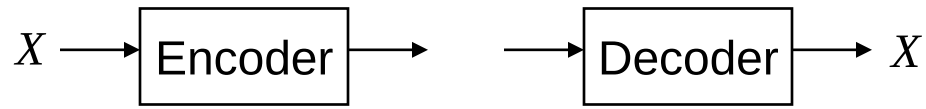
- Session I. Introduction and theory : 9.00 am-10.00 am
  - Motivation and intuition
  - Distributed source coding foundations
  - Break: 10.00-10.10 am

- Session II. Constructions: 10.10 am-10.50 am
  - Structure of distributed source codes
  - Constructions based on trellis codes
  - Constructions based on codes on graphs
  - Break: 10.50-11.00 am

- Session III. Connections and Applications: 11.00 am-12.00 noon
  - Overview of connections and applications with snippets
  - Compression of encrypted data
  - Distributed video coding

Source coding: lossless case [Shannon '49]

- Source alphabet $\mathcal{X}$
- Source distribution ~ $p_X(x)$

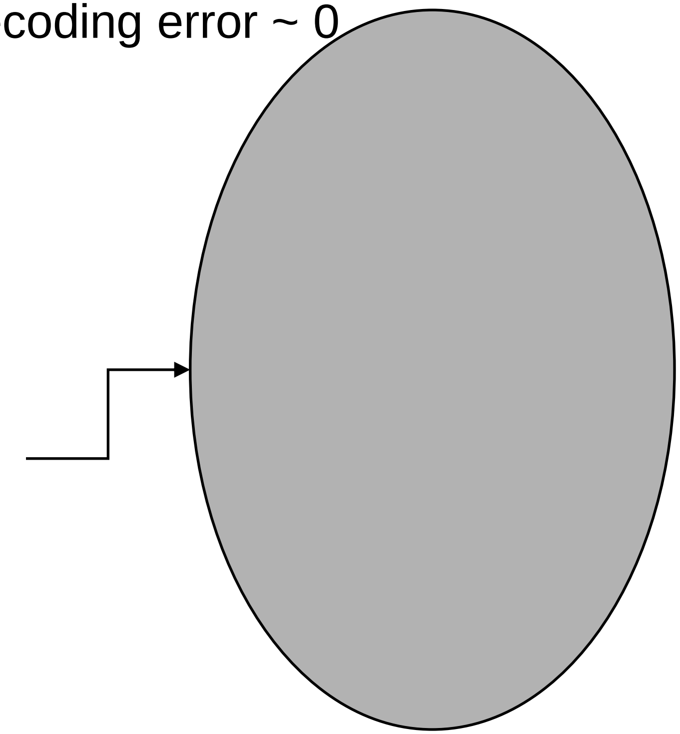


- Encoder: e: $\mathcal{X}^N \to \{1, 2, \ldots, 2^{\mathrm{NR}}\}$
- Decoder: f: $\{1, 2, \ldots, 2^{\mathrm{NR}}\} \to \mathcal{X}^N$

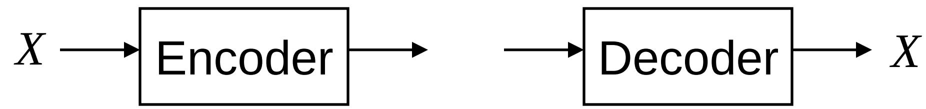
- Goal: minimize rate R such that probability of decoding error ~ 0
- Answer: $R \geq H(X)$

- Idea: index only typical sequences

Set of all N-length sequences
(Size $\approx 2^{N\log|\mathcal{X}|}$)



Source coding: lossless case

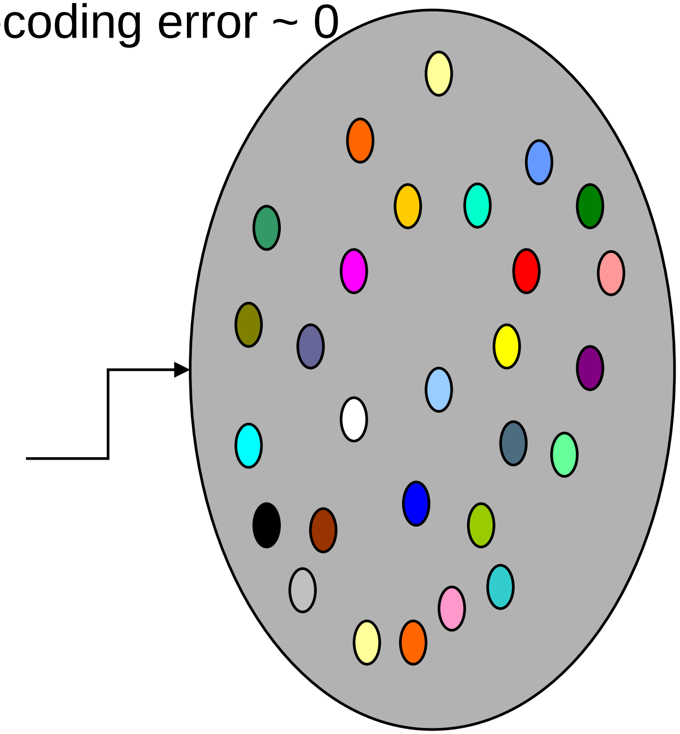


- Source alphabet $\mathcal{X}$
- Source distribution ~ $p_X(x)$

- Encoder: e: $\mathcal{X}^N \rightarrow \{1, 2, \ldots, 2^{NR}\}$
- Decoder: f: $\{1, 2, \ldots, 2^{NR}\} \rightarrow \mathcal{X}^N$

- Goal: minimize rate R such that probability of decoding error ~ 0
- Answer: $R \geq H(X)$

- Idea: index only typical sequences
  - Probability of typical set ~ 1

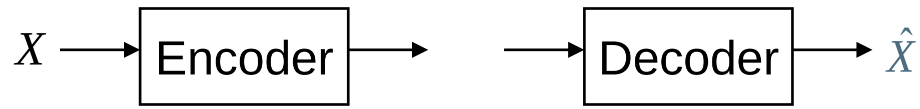


Set of all N-length sequences
(Size $\approx 2^{N\log|\mathcal{X}|}$)

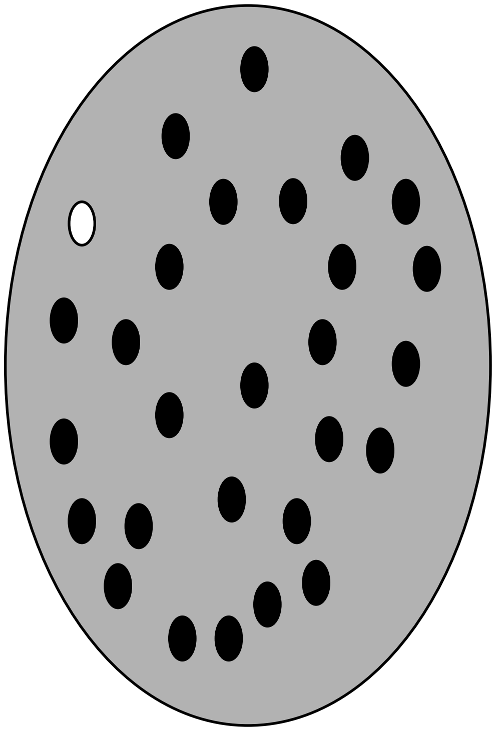
Set of typical sequences
(Size $\approx 2^{NH(X)}$)

Source coding: lossy case [Shannon '58]

- Distortion function: $d(x,\hat{x})$

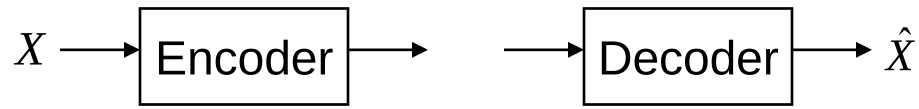


- Goal: minimize rate R such that expected distortion $< D$
- Answer: $R \geq R(D) = \min_{p(\hat{x}|x): Ed(X,\hat{X}) \leq D} I(X;\hat{X})$
- Idea
  - Cover typical set with "spheres" of radius D,
  - Index these "spheres"
  - Size of each "sphere" $\approx 2^{NH(X|\hat{X})}$
  - Rate $= H(X) - H(X\,|\,\hat{X}) = I(X;\hat{X})$
  - Sequences which get the same index are nearby

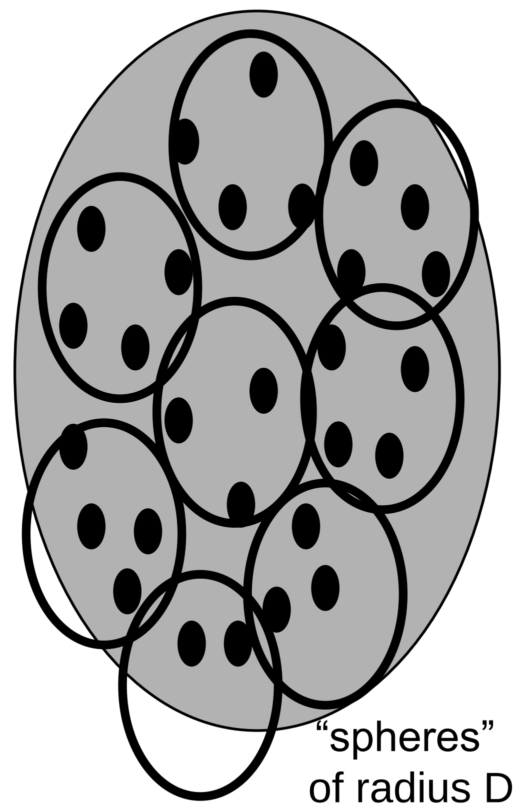


Source coding: lossy case [Shannon '58]

- Distortion function: $d(x, \hat{x})$

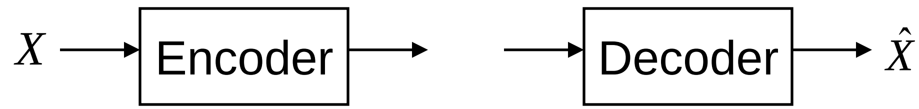


- Goal: minimize rate R such that expected distortion $< D$
- Answer: $R \geq R(D) = \min_{p(\hat{x}|x):Ed(X,\hat{X}) \leq D} I(X;\hat{X})$
- Idea
  - Cover typical set with "spheres" of radius D,
  - Index these "spheres"
  - Size of each "sphere" $\approx 2^{NH(X|\hat{X})}$
  - Rate $= H(X) - H(X \mid \hat{X}) = I(X;\hat{X})$
  - Sequences which get the same index are nearby

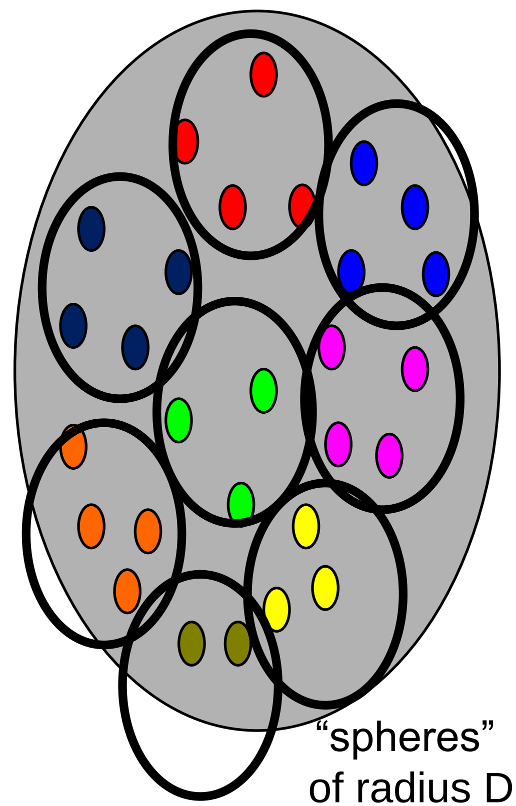


Source coding: lossy case [Shannon '58]

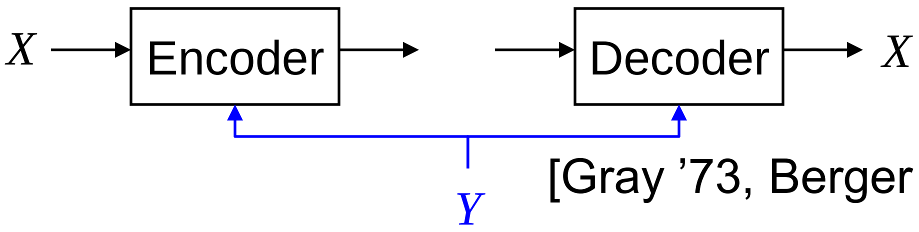
- Distortion function: $d(x, \hat{x})$



- Goal: minimize rate R such that expected distortion $< D$
- Answer: $R \geq R(D) = \min_{p(\hat{x}|x): Ed(X,\hat{X}) \leq D} I(X;\hat{X})$
- Idea
  - Cover typical set with "spheres" of radius D,
  - Index these "spheres"
  - Size of each "sphere" $\approx 2^{NH(X|\hat{X})}$
  - Rate $= H(X) - H(X\,|\,\hat{X}) = I(X;\hat{X})$
  - Sequences which get the same index are nearby

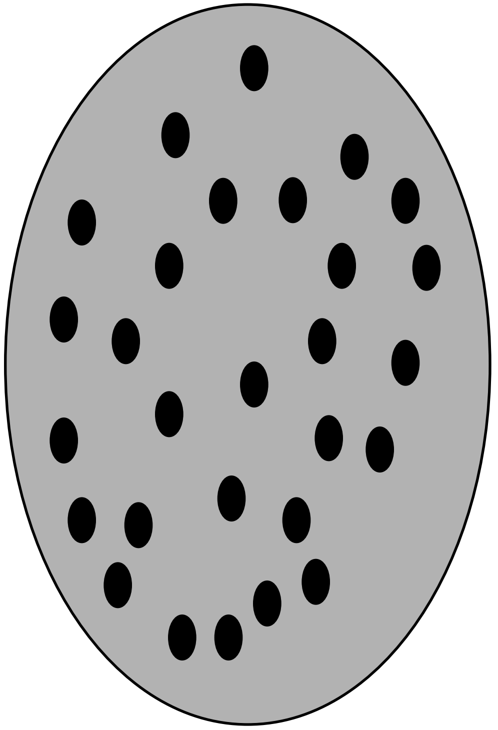


Source coding w/side information: lossless case



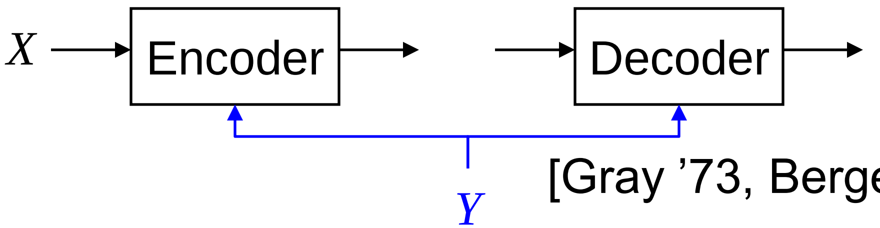
[Gray '73, Berger '71]

- Source X
- Side information Y
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Answer: $R = H(X|Y)$
- Idea
  - Given side information sequence Y^N, index conditionally typical sequences of X^N given Y^N

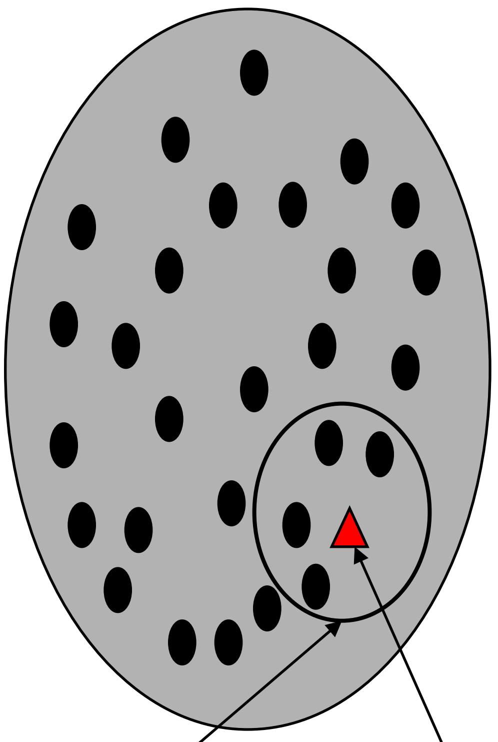


Source coding w/side information: lossless case

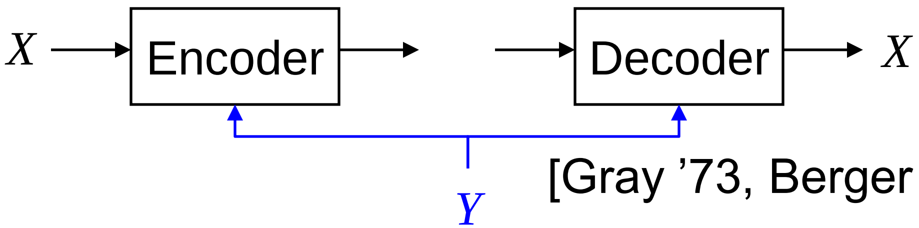
- Source X
- Side information Y
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Answer: $R = H(X|Y)$
- Idea
  - Given side information sequence Y^N, index conditionally typical sequences of X^N given Y^N



[Gray '73, Berger '71]

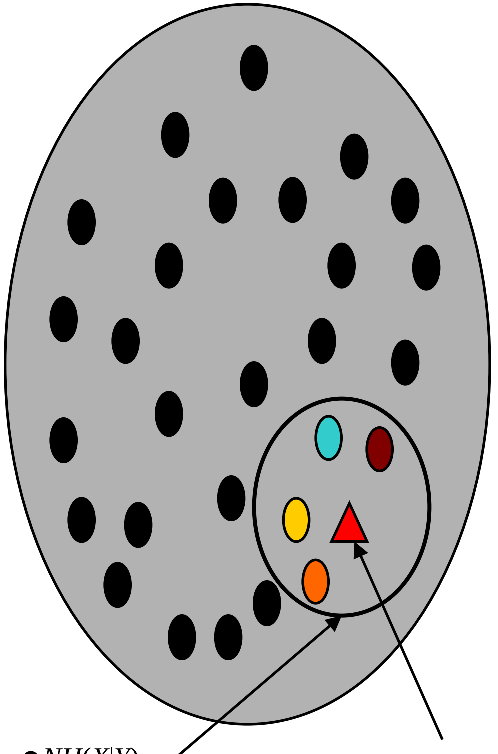


Source coding w/side information: lossless case



[Gray ’73, Berger ’71]

- Source X
- Side information Y
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Answer: $R = H(X|Y)$
- Idea
  - Given side information sequence Y^N, index conditionally typical sequences of X^N given Y^N

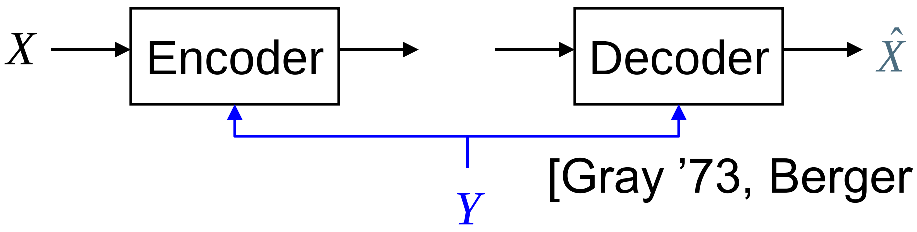


Conditionally typical set of size $\approx 2^{NH(X|Y)}$

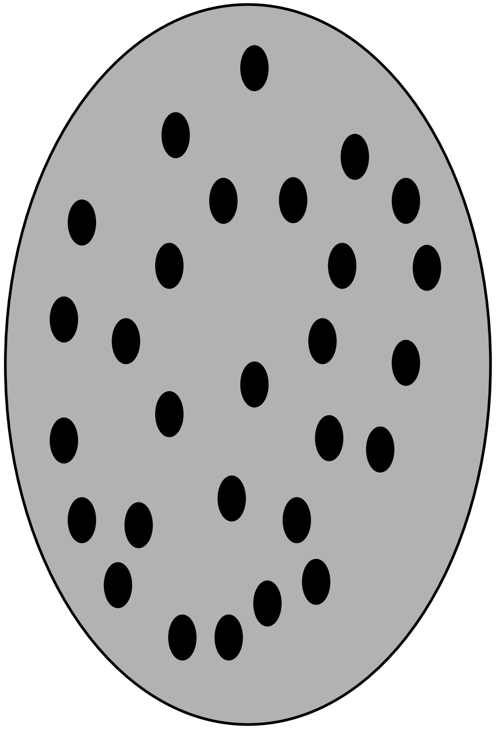
Y-sequence

Source coding w/side information: lossy case

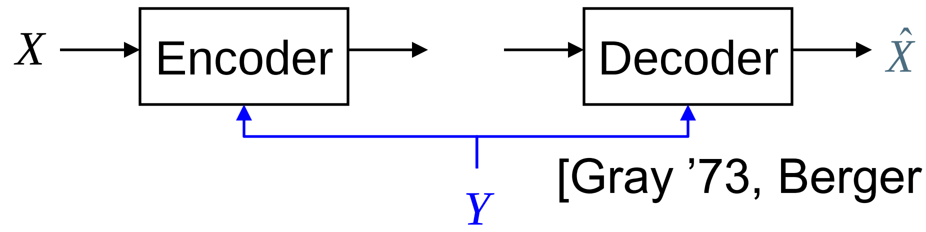
- Source X
- Side information Y
- Source distribution ~ $p_{X|Y}(x|y)$
- Goal: minimize rate R such that expected distortion $< D$



[Gray ’73, Berger ’71]



Source coding w/side information: lossy case

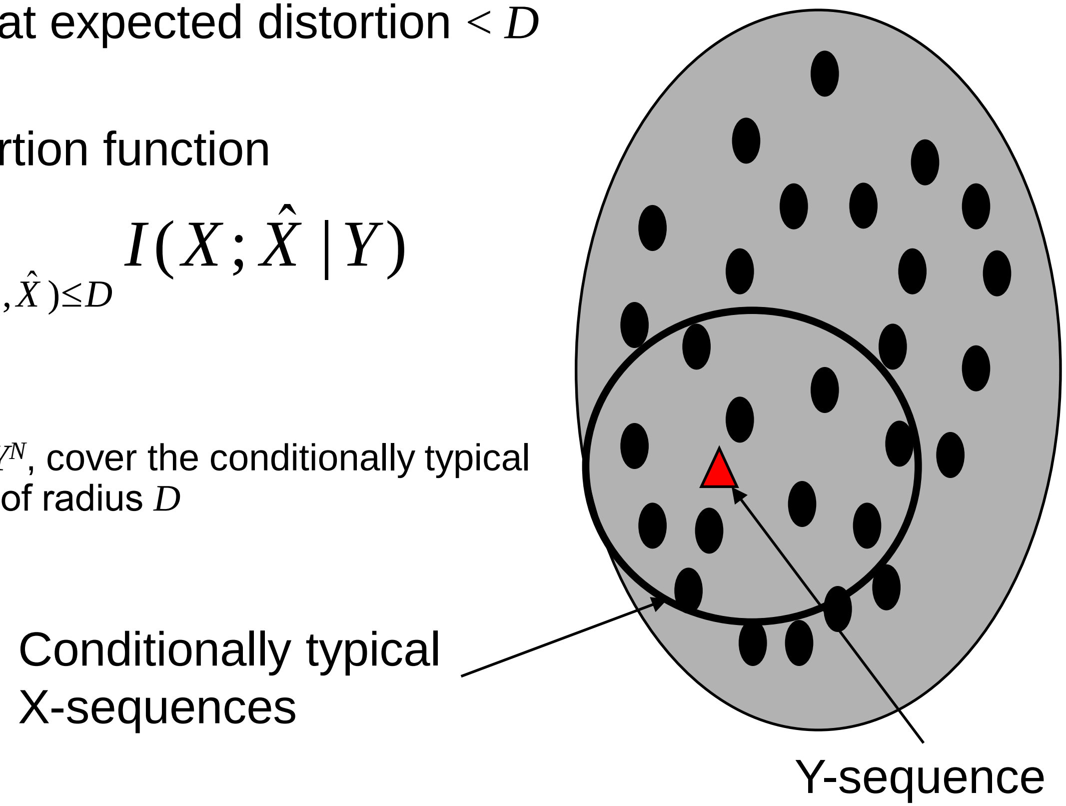


[Gray '73, Berger '71]

- Source X
- Side information Y
- Source distribution ~ $p_{X|Y}(x|y)$
- Goal: minimize rate R such that expected distortion < D
- Answer: conditional rate-distortion function

$$R \geq R_{X|Y}(D) = \min_{p(\hat{x}|x,y):Ed(X,\hat{X}) \leq D} I(X;\hat{X}|Y)$$

- Idea
  - Given side information sequence Y^N, cover the conditionally typical set of X^N given Y^N using "spheres" of radius D
  - Index these spheres

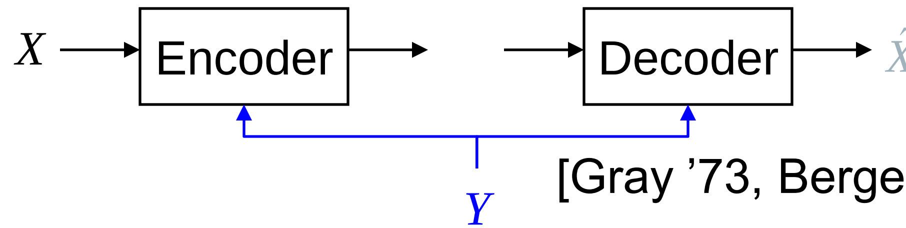


Source coding w/side information: lossy case

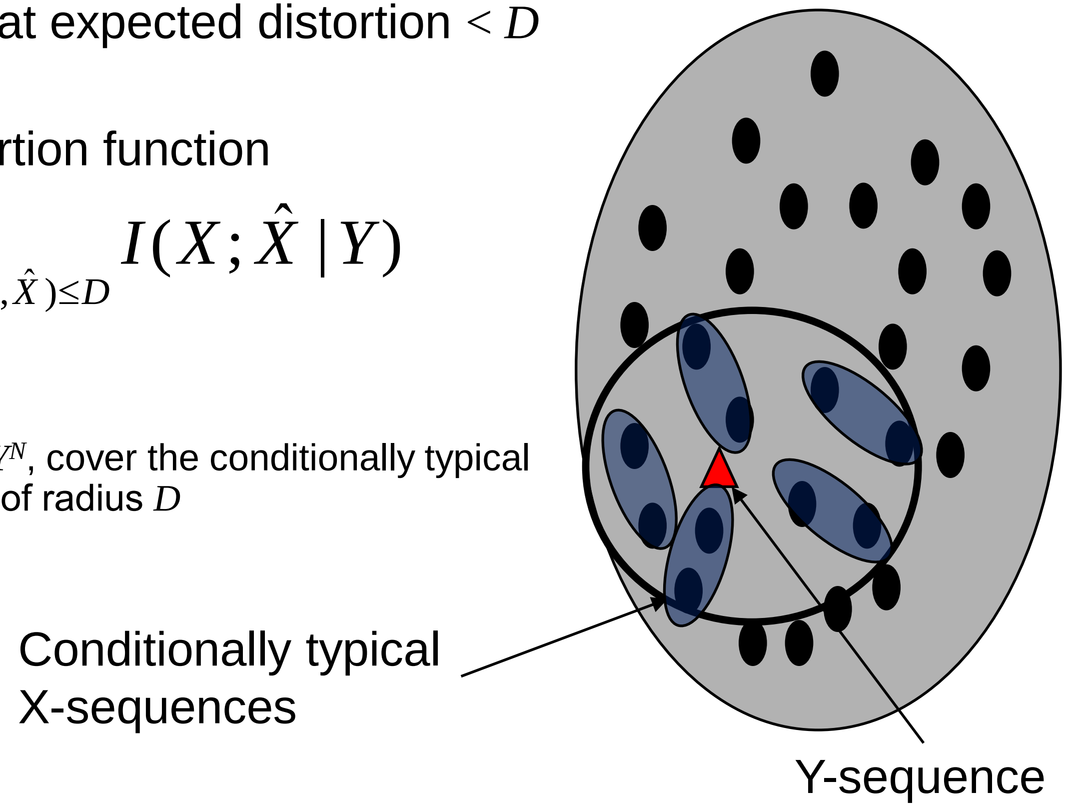
- Source X
- Side information Y
- Source distribution $\sim p_{X|Y}(x|y)$
- Goal: minimize rate R such that expected distortion $< D$
- Answer: conditional rate-distortion function

$$R \geq R_{X|Y}(D) = \min_{p(\hat{x}|x,y):Ed(X,\hat{X})\leq D} I(X;\hat{X}\,|\,Y)$$

- Idea
  - Given side information sequence Y^N, cover the conditionally typical set of X^N given Y^N using "spheres" of radius D
  - Index these spheres

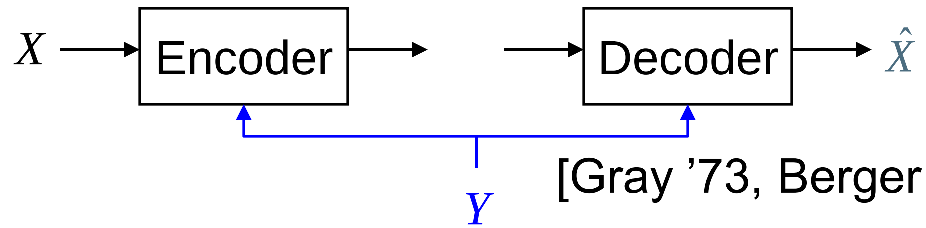


[Gray '73, Berger '71]



Source coding w/side information: lossy case

- Source X

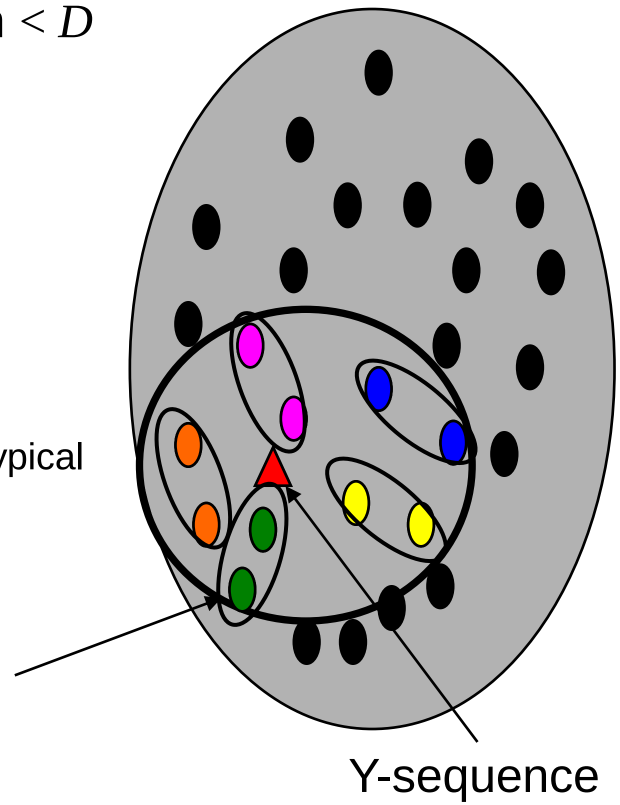


[Gray '73, Berger '71]

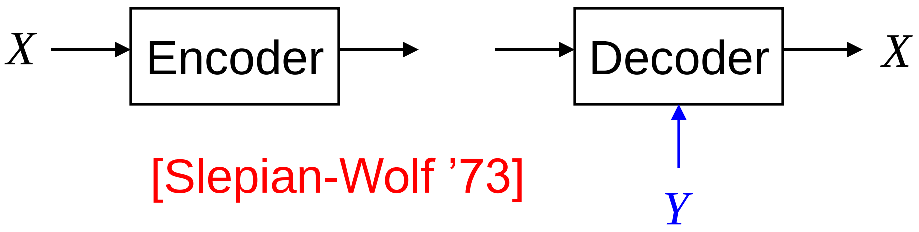
- Side information Y
- Source distribution $\sim p_{X|Y}(x|y)$
- Goal: minimize rate R such that expected distortion $< D$
- Answer: conditional rate-distortion function

$$R \geq R_{X|Y}(D) = \min_{p(\hat{x}|x,y): Ed(X,\hat{X}) \leq D} I(X;\hat{X}\,|\,Y)$$

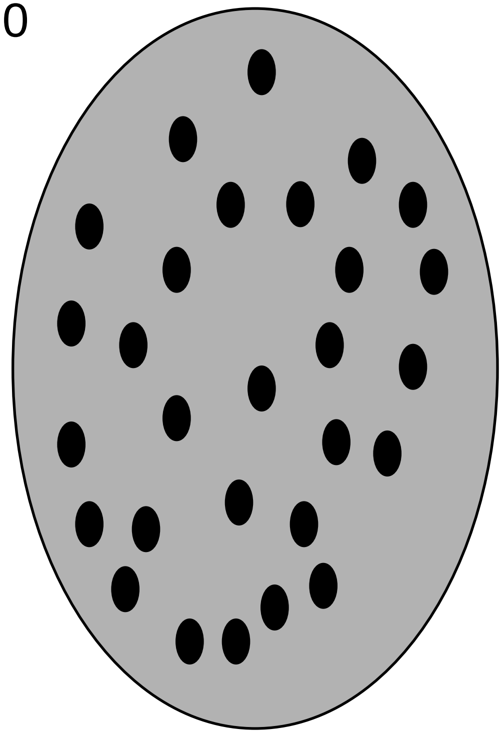
- Idea
  - Given side information sequence Y^N, cover the conditionally typical set of X^N given Y^N using "spheres" of radius D
  - Index these spheres



Source coding w/ SI at decoder only: lossless

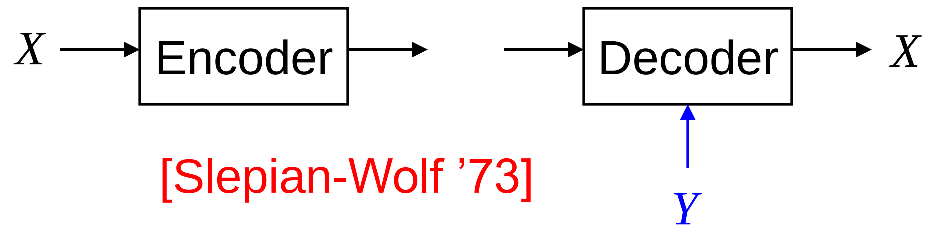


- Source X
- Side information Y
- Source distribution ~ $p_{X|Y}(x|y)$
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Idea
  - Typical X-sequences which are far apart given the same index
  - Induces a partition on the space of X : binning
  - Any valid Y-sequence $\rightarrow$ there do not exist more than one conditionally typical X-sequence having the same index

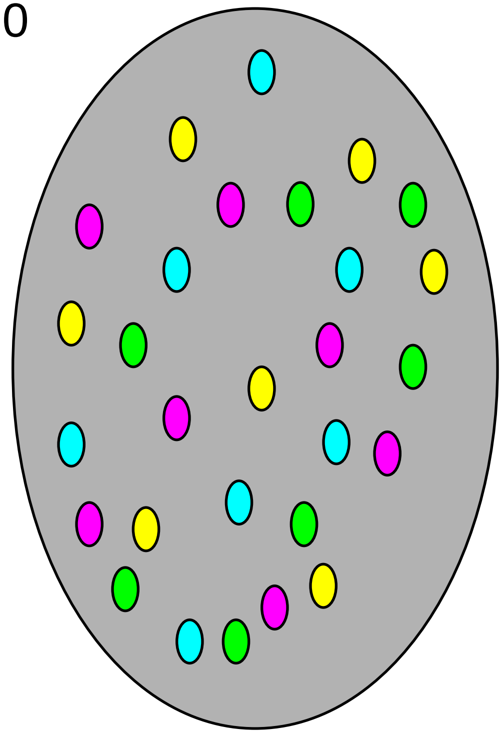


Source coding w/ SI at decoder only: lossless

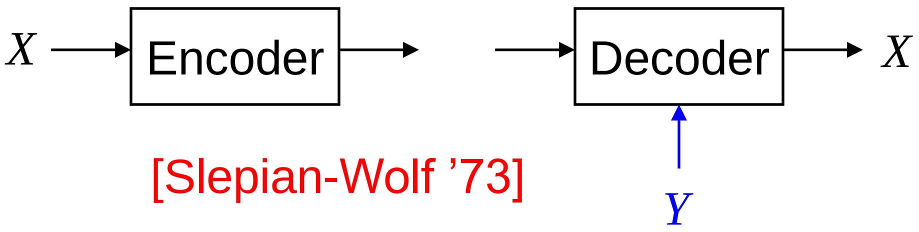
- Source X
- Side information Y
- Source distribution ~ $p_{X|Y}(x|y)$
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Idea
  - Typical X-sequences which are far apart given the same index
  - Induces a partition on the space of X : binning
  - Any valid Y-sequence $\rightarrow$ there do not exist more than one conditionally typical X-sequence having the same index



[Slepian-Wolf '73]

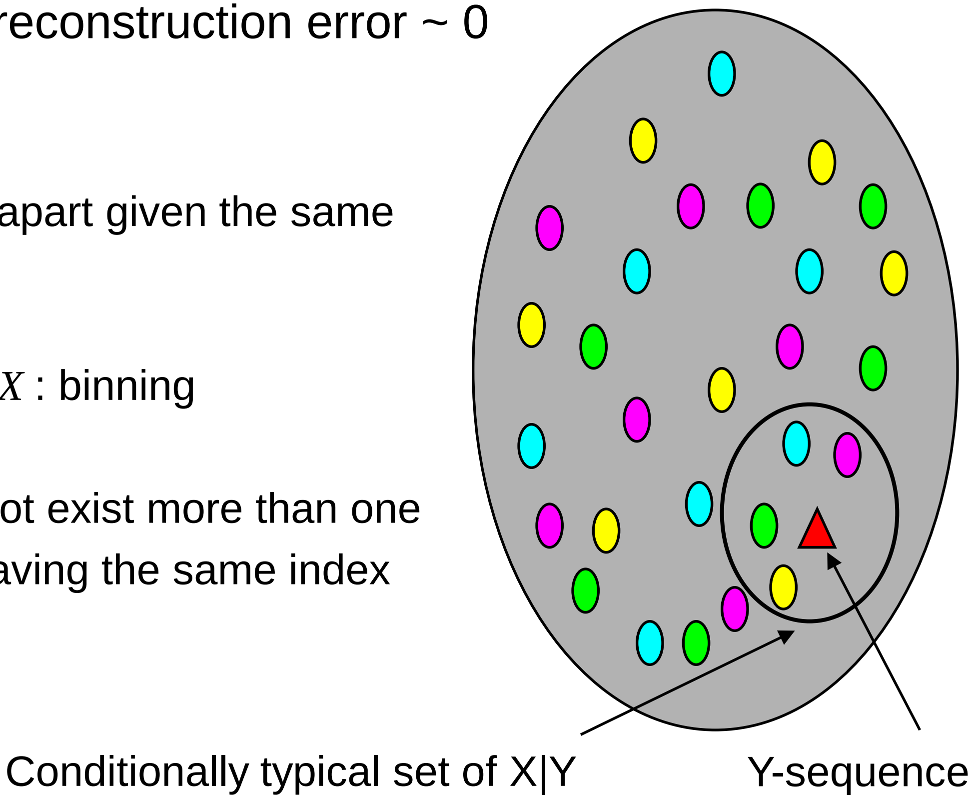


Source coding w/ SI at decoder only: lossless

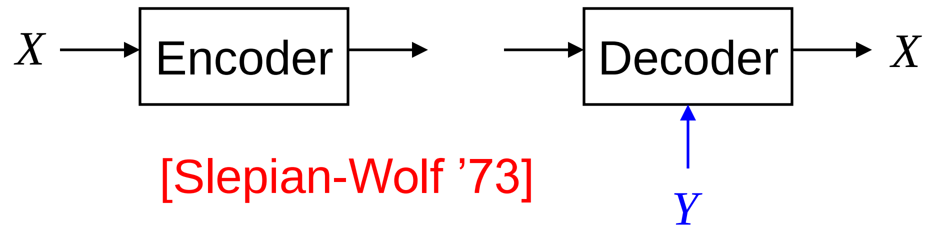


[Slepian-Wolf ’73]

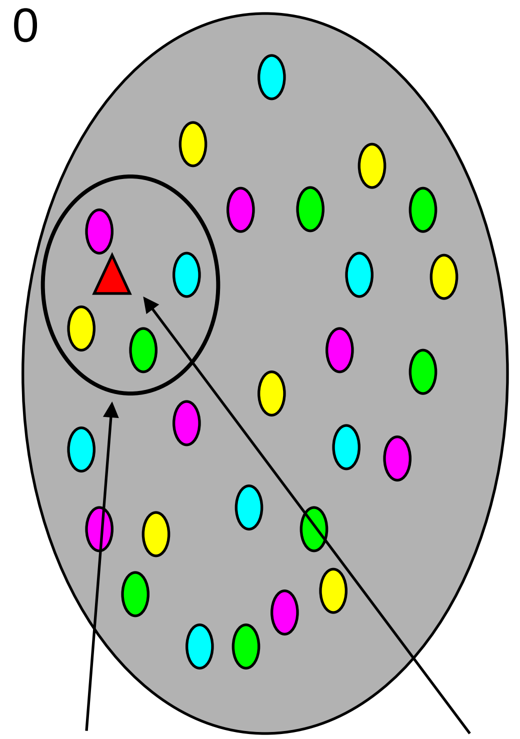
- Source X
- Side information Y
- Source distribution $\sim p_{X|Y}(x|y)$
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Idea
  - Typical X-sequences which are far apart given the same index
  - Induces a partition on the space of X : binning
  - Any valid Y-sequence $\rightarrow$ there do not exist more than one conditionally typical X-sequence having the same index



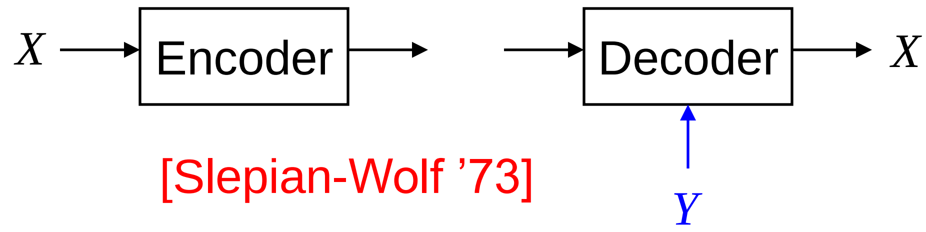
Source coding w/ SI at decoder only: lossless



- Source X
- Side information Y
- Source distribution ~ $p_{X|Y}(x|y)$
- Goal: minimize rate R s.t. prob. of reconstruction error ~ 0
- Idea
  - Typical X-sequences which are far apart given the same index
  - Induces a partition on the space of X : binning
  - Any valid Y-sequence → there do not exist more than one conditionally typical X-sequence having the same index



Source coding w/ SI at decoder only: lossless



[Slepian-Wolf '73]

- Set of X-sequences that get the same index $\leftrightarrow$ channel code for the fictitious channel with input X, output Y

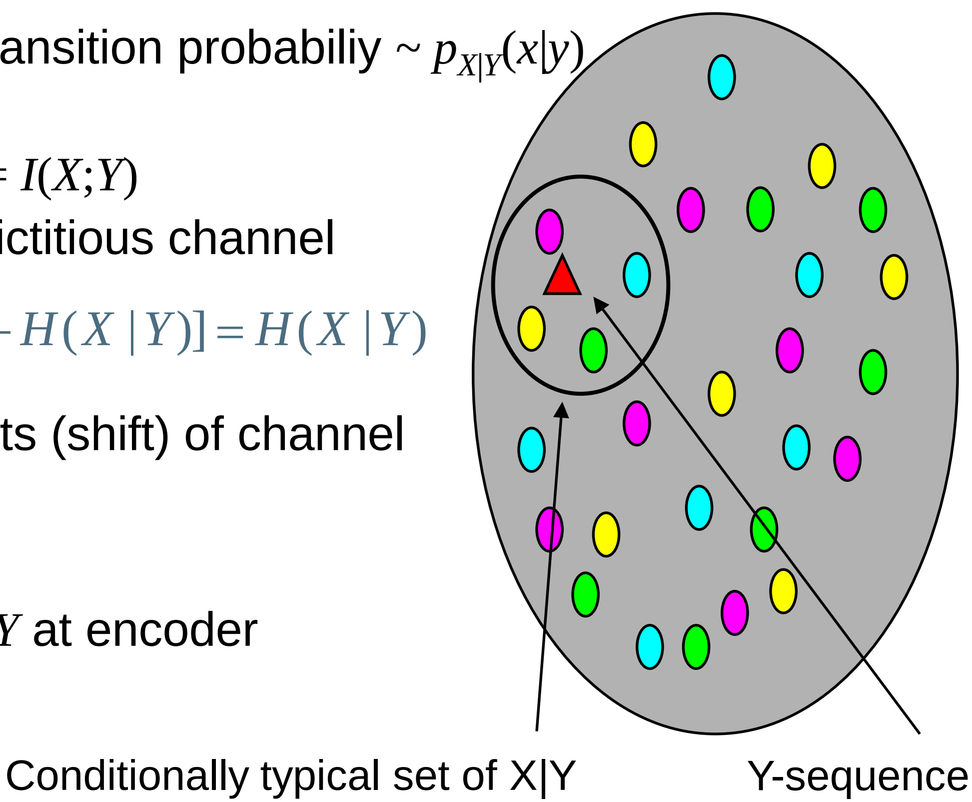
- Channel: input distribution $p_X(x)$, transition probabiliy ~ $p_{X|Y}(x|y)$

- Max. reliable rate of transmission = $I(X;Y)$
- This rate comes for free from this fictitious channel

$$\Rightarrow R = H(X) - I(X;Y) = H(X) - [H(X) - H(X|Y)] = H(X|Y)$$

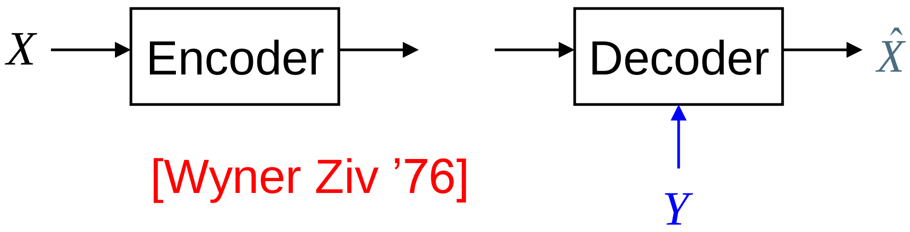
- Source space partitioned into cosets (shift) of channel codebooks

- No loss in performance for lack of Y at encoder

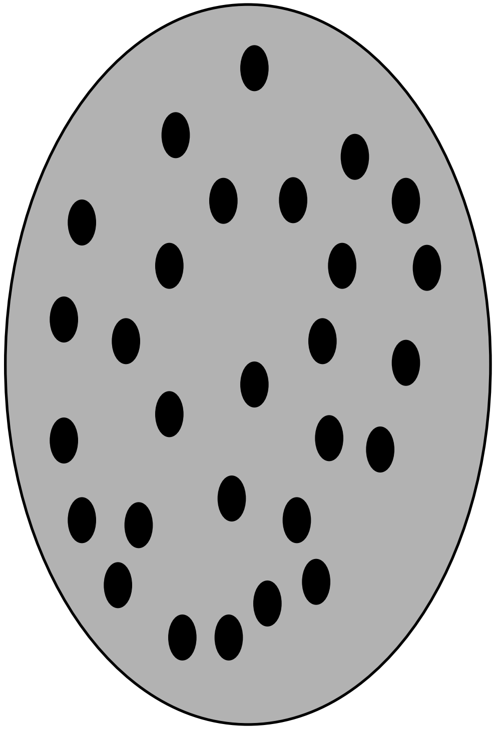


Source coding w/ SI at decoder only: lossy

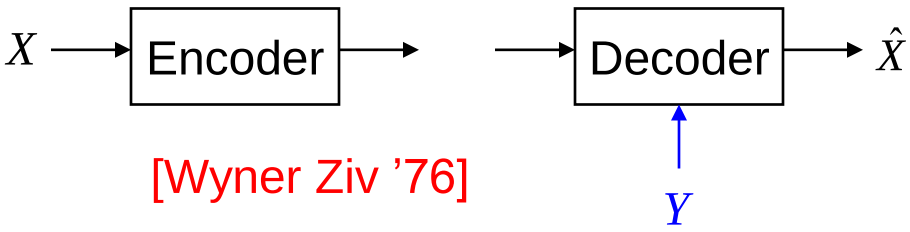
- Source X
- Side information Y
- Distortion function $d(x,\hat{x})$
- Goal: minimize rate R such that expected distortion $< D$



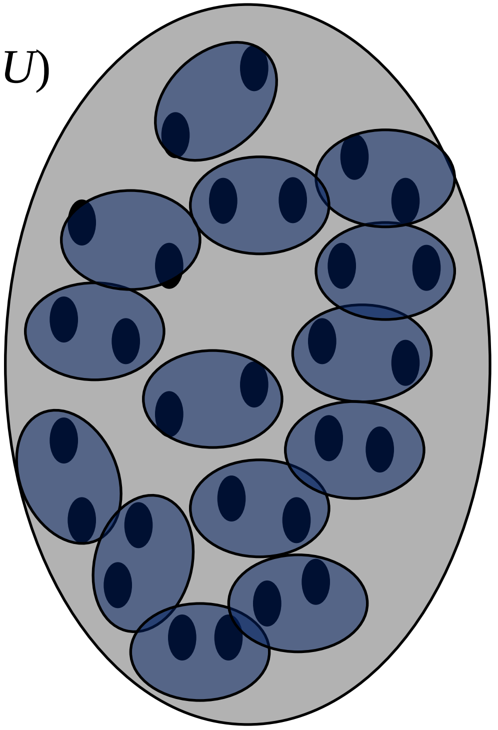
[Wyner Ziv '76]



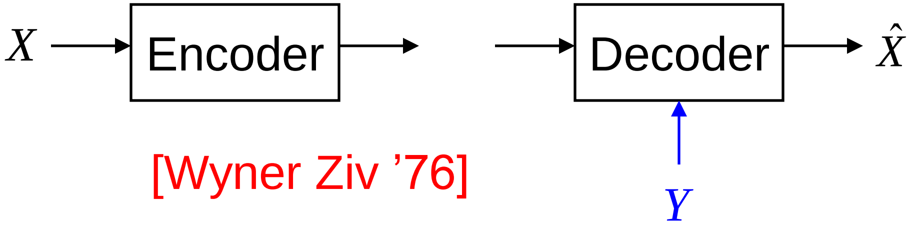
Source coding w/ SI at decoder only: lossy



- Quantize X to some intermediate reconstruction U
- From standard R-D theory, this would incur a rate of $I(X; U)$
- Apply source coding with SI idea losslessly
- New fictitious channel has input U, and output Y
- This gives a rebate in rate of $I(U; Y)$
- Total rate = $I(X; U)$ - $I(U; Y)$

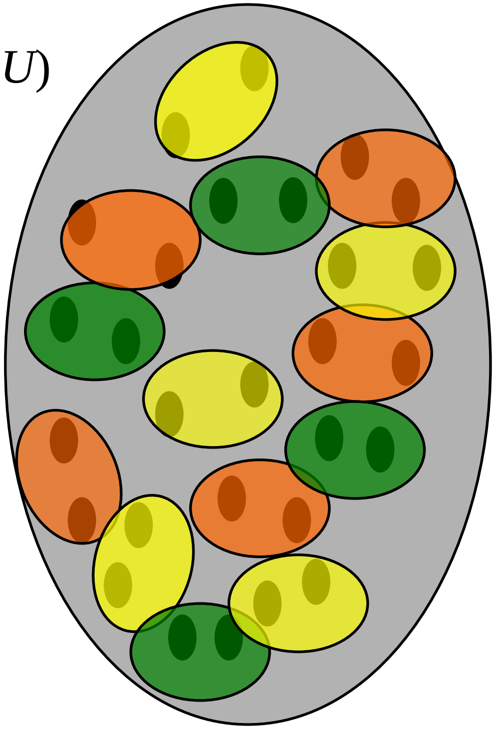


Source coding w/ SI at decoder only: lossy

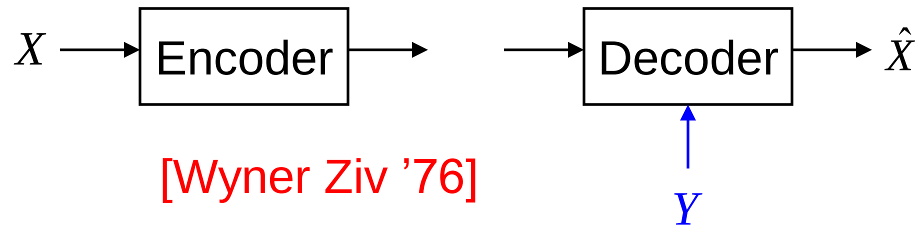


[Wyner Ziv '76]

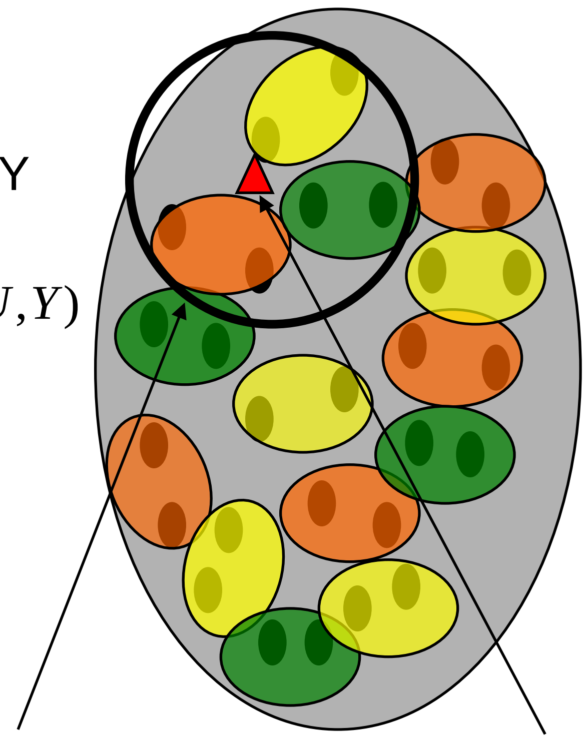
- Quantize X to some intermediate reconstruction U
- From standard R-D theory, this would incur a rate of $I(X; U)$
- Apply source coding with SI idea losslessly
- New fictitious channel has input U, and output Y
- This gives a rebate in rate of $I(U; Y)$
- Total rate = $I(X; U) - I(U; Y)$



Source coding w/ SI at decoder only: lossy



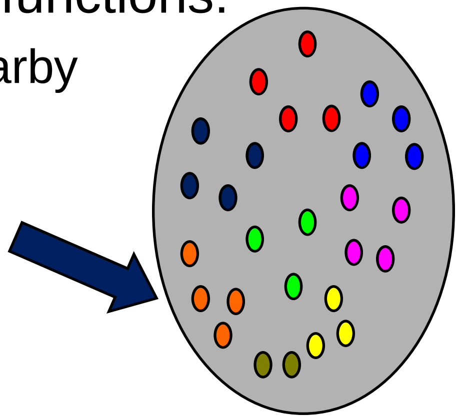
- Encoder does not observe Y
- Choosing $p(u|x)$ fixes the joint distribution of X,Y,U using Markov chain condition $Y \to X \to U$ as $p(y)p(x|y)p(u|x)$
- The decoder has two looks at X: through U, through Y
- Get an optimal estimate of X given U and Y: $\hat{X} = g(U,Y)$
- SI Y is used twice: recovering U, estimating X
- $R \geq R_{WZ}(D) = \min_{p(u|x): Ed(X,\hat{X}) \leq D} I(X;U) - I(U;Y)$



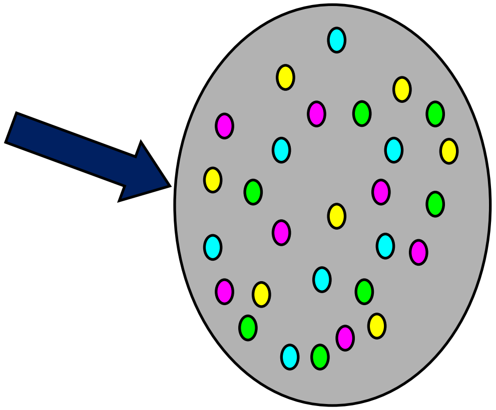
Remark

- Quantizer for the source X is partitioned into cosets (shift) of channel codebooks for the fictitious channel with i/p U and o/p Y

- Contrast between two kinds of many-to-one encoding functions:
  - Quantization: sequences that get the same index are nearby



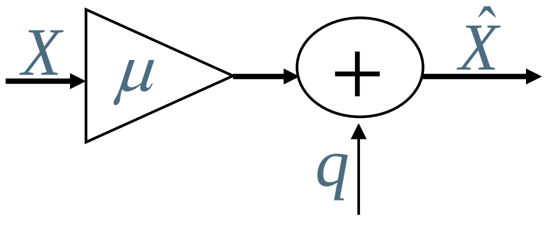
  - Binning: sequences that get the same index are far apart



Example: Gaussian with quadratic distortion

Lossy source coding with no side information

- X is zero-mean Gaussian with variance σ_x^2
- Quadratic distortion: $d(x,\hat{x}) = (x-\hat{x})^2$
- $R(D) = \frac{1}{2}\log\left(\frac{\sigma_x^2}{D}\right)$
- Test channel is given by:



$$\Rightarrow p(\hat{x}\,|\,x) = \text{Gaussian}$$

$$\text{mean} = \mu\, x$$

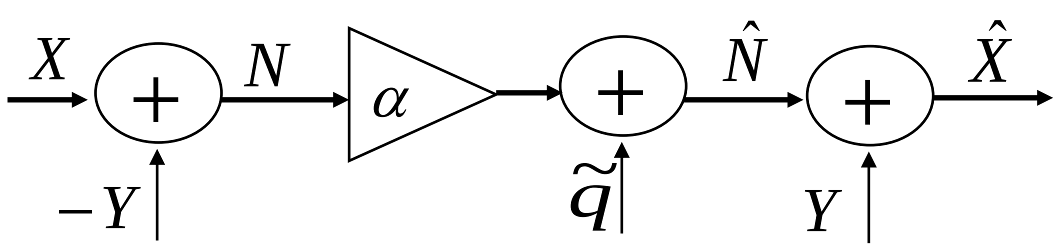
$$\text{var} = \frac{D(\sigma_x^2 - D)}{\sigma_x^2}$$

$$\mu = \frac{\sigma_x^2 - D}{\sigma_x^2}$$

Example: Gaussian with quadratic distortion

Lossy source coding with side information

- X=Y+N, where N is zero-mean Gaussian with variance σ_N^2
- Y is arbitrary and independent of N
- Quadratic distortion: $d(x,\hat{x}) = (x-\hat{x})^2$
- $R_{X|Y}(D) = \frac{1}{2}\log\left(\frac{\sigma_n^2}{D}\right)$
- Test channel is given by:



$$\Rightarrow p(\hat{x}\,|\,x,y) = \text{Gaussian}$$

$$\text{mean} = \alpha x + (1-\alpha)y$$

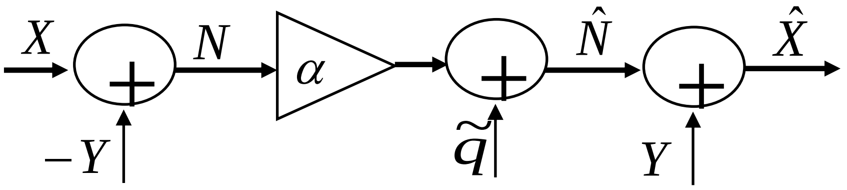
$$\text{var} = \frac{D(\sigma_n^2 - D)}{\sigma_n^2}$$

$$\alpha = \frac{\sigma_n^2 - D}{\sigma_n^2}$$

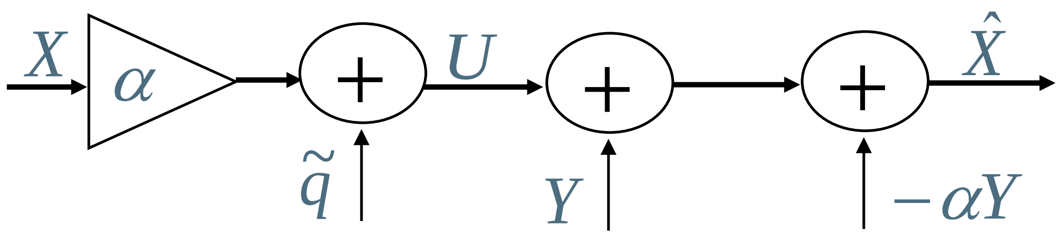
Example: Gaussian with quadratic distortion

Lossy source coding with side information at decoder only

- Source, side information, and distortion as before
- $R_{WZ}(D) = \frac{1}{2}\log\left(\frac{\sigma_n^2}{D}\right)$ $\rightarrow$ no performance loss for lack of Y at encoder
- Test channel when SI is present at both ends



- Test channel when SI is present at decoder only



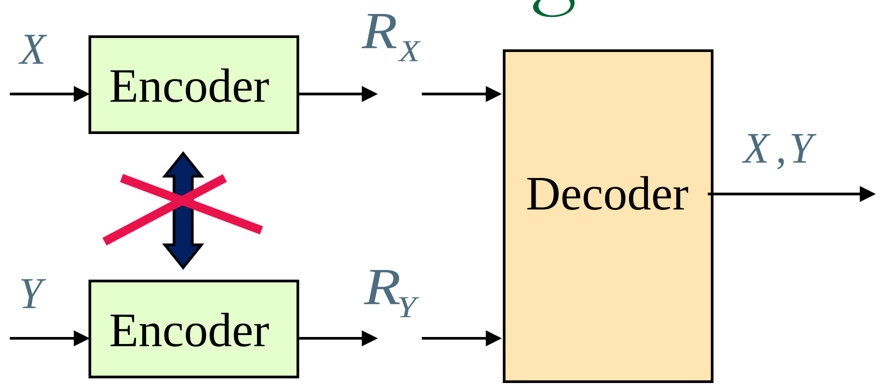
$$\Rightarrow p(u \mid x) = \text{Gaussian}$$

$$\text{mean } = \alpha\, x$$

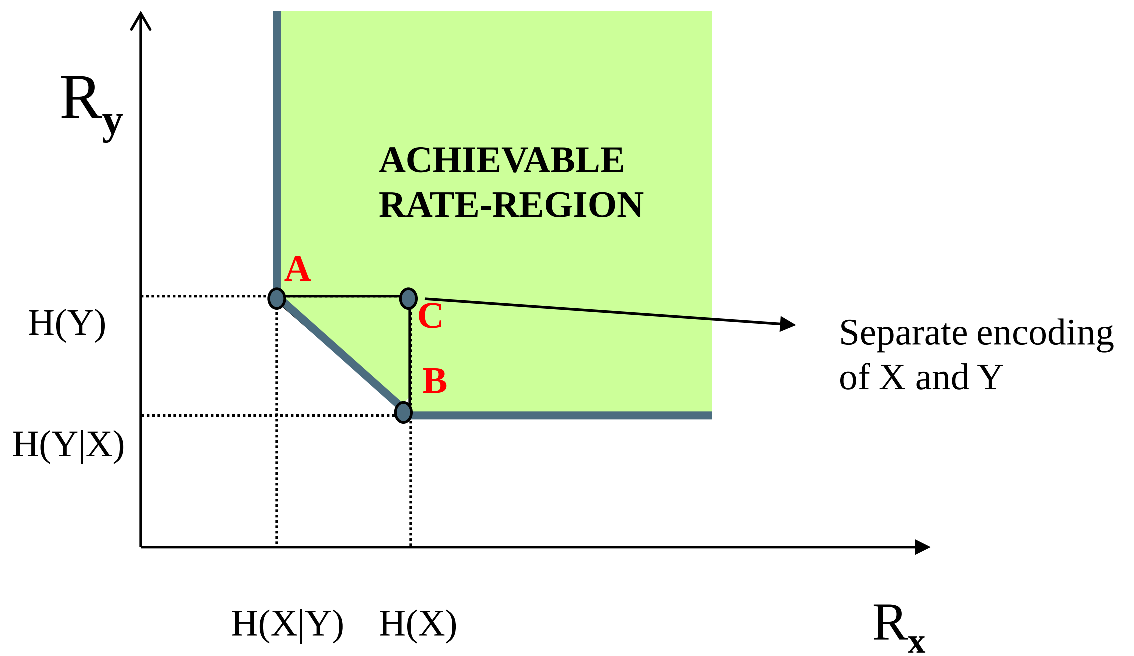
$$\text{var} = \frac{D(\sigma_n^2 - D)}{\sigma_n^2}$$

$$\hat{X} = U + (1-\alpha)Y$$

Distributed source coding: lossless case

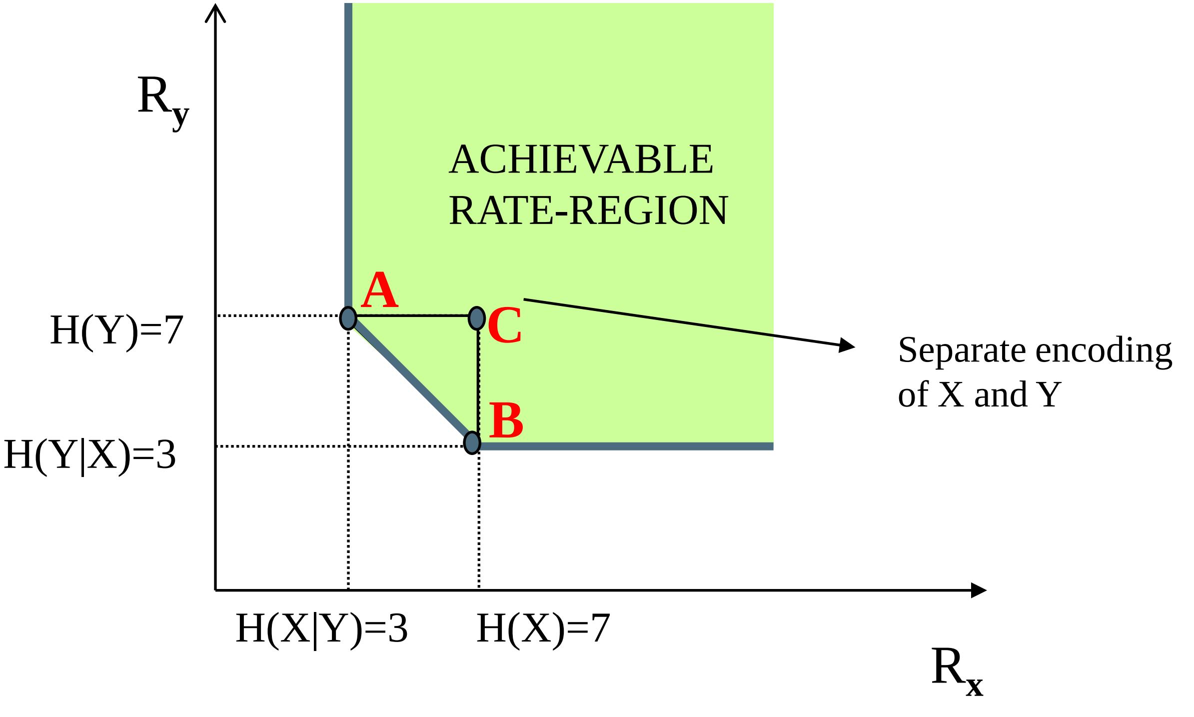


- Minimize rate pair R_X, R_Y such that probability of decoding error ~ 0

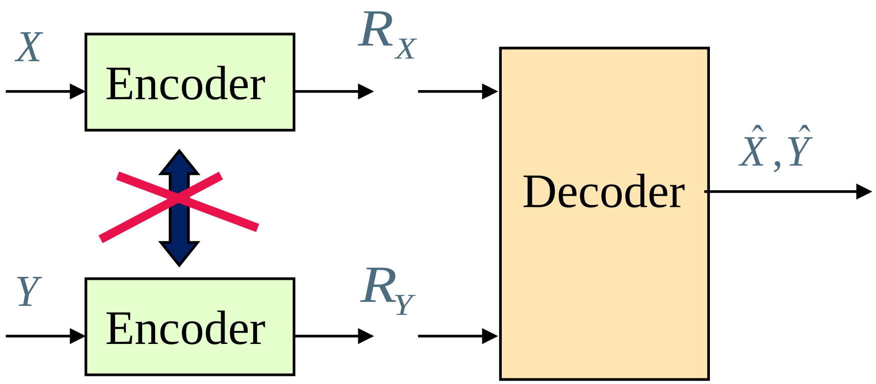


Example

- X and Y → length-7 equally likely binary data with Hamming distance between them at most 1.
  - H(X)= 7 bits
  - H(Y|X)= 3 bits = H(Y|X)
  - H(X,Y)=10 bits



Distributed source coding: lossy case



- Minimize rate pair R_X, R_Y such that

$$E[d_X(X,\hat{X})] \le D_X, \text{ and } E[d_Y(Y,\hat{Y})] \le D_Y$$

- Optimal performance limit: open problem!

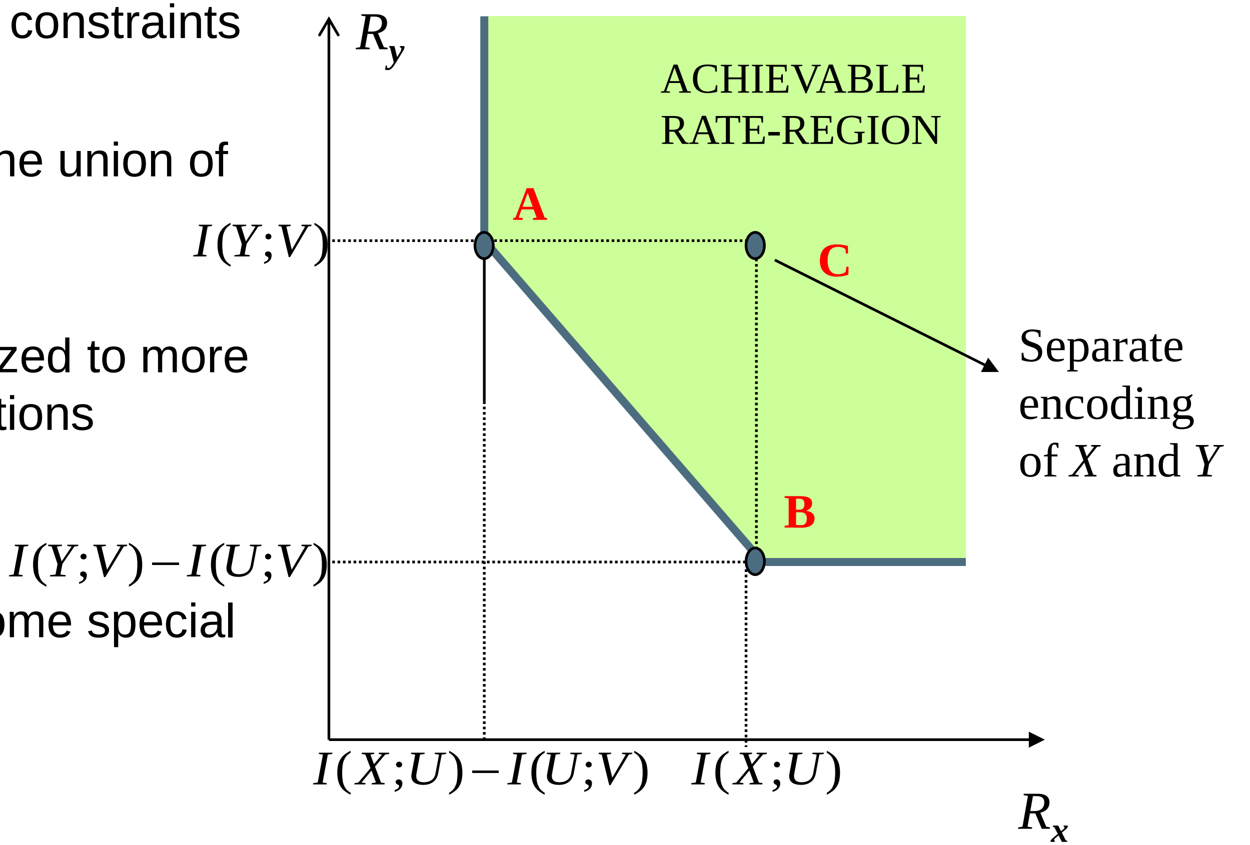
- Approach: [Berger-Tung '77]
  - Quantize Y to V
  - Treat V as side information

Distributed source coding: lossy case

- Berger-Tung achievable rate region

$$R_X \geq I(X;U) - I(U;V)$$
$$R_Y \geq I(Y;V) - I(U;V)$$
$$R_X + R_Y \geq I(X;U) + I(Y;V) - I(U;V)$$

- For every choice of $p_{U|X}(u|x), \quad p_{V|Y}(v|y)$ that satisfies distortion constraints
- Overall rate region is the union of such regions
- Can be easily generalized to more general distortion functions
- Shown to be tight in some special cases



Remarks

- All results $\rightarrow$ random quantization and binning
- Structured random codes may give a better performance than unstructured random codes [Korner-Marton '79]
- Structured codes for quantization and binning is a topic of active research.

BREAK
10.00 AM-10.10 AM

Outline

- Session I. Introduction and theory: 9.00 am-10.00 am
  - Motivation and intuition
  - Distributed source coding foundations
  - Break: 10.00-10.10 am

- Session II. Constructions: 10.10 am-10.50 am
  - Structure of distributed source codes
  - Constructions based on trellis codes & codes on graphs
  - Break: 10.50-11.00 am

- Session III. Connections and Applications: 11.00 am-12.00 noon
  - Overview of connections and applications with snippets
  - Compression of encrypted data
  - Distributed video coding

Active area of recent research

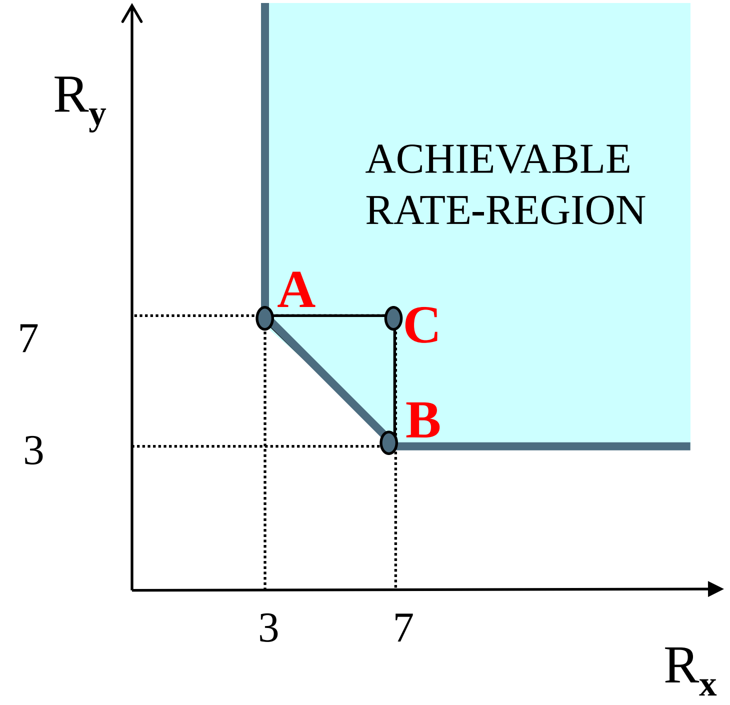
- Theory
  - D. Slepian & J. Wolf ('73)
  - A. Wyner ('74)
  - A. Wyner & J. Ziv ('76)
  - I. Csiszar ('82)
  - Zamir et al ('02, '03, '04)
  - S. Pradhan & K. Ramchandran ('05)
  - Many More…
- Source coding with side information – I.I.D. Sources
  - A. Orlitsky ('93, Graph Theoretic)
  - S. Pradhan & K. Ramchandran ('99)
  - Y. Zhao & J. Garcia-Frias ('02, larger alphabets)
  - A. Liveris, Z. Xiong, & C. Georghiades ('02)
  - D. Schonberg, S. Pradhan, & K. Ramchandran ('02)
  - P. Mitran & J. Bajcsy ('02)
  - A. Aaron & B. Girod ('02)
  - A. Liveris, Z. Xiong, & C. Georgihades ('03)
  - J. Li, Z. Tu, & R. Blum ('04)
  - M. Sartipi & F. Fekri ('05)
- Source coding with side information – Correlated Sources
  - J. Garcia-Frias & W. Zhong ('03)
  - D. Varodayan, A. Aaron, & B. Girod ('06)

Example

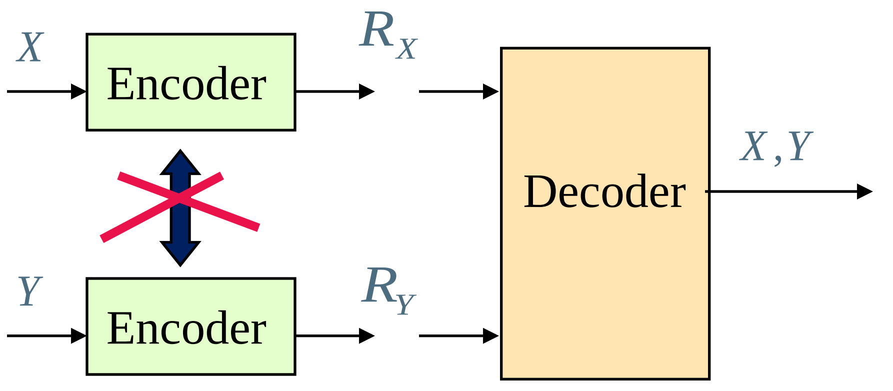
- X and Y -> length-7 equally likely binary data with Hamming distance between them at most 1.
  - H(X)= ? bits
  - H(Y|X)= ? bits = H(Y|X)
  - H(X,Y)=? bits

Answer:

- •H(x)=H(Y)=7 bits, H(X,Y)=10 bits
- •Use (7,4,3) Hamming code
- •Send Y as is (7 bits)
- •Send syndrome for X (3 bits)



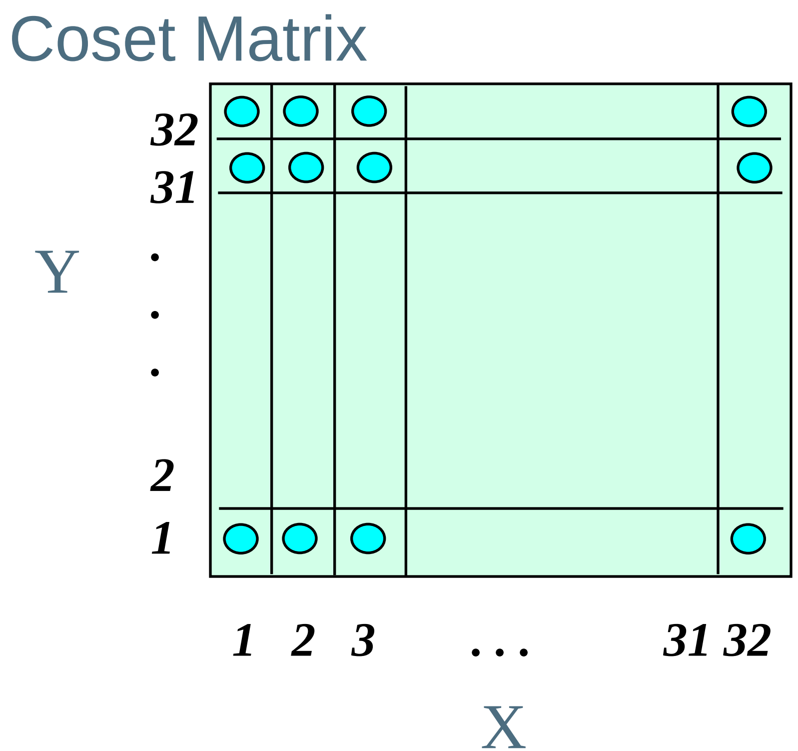
Symmetric Coding



- Example:
  - X and Y -> length-7 equally likely binary data.
  - Hamming distance between X and Y is at most 1
- Solution 1:
  - Y sends its data with 7 bits.
  - X sends syndromes with 3 bits.
  - { (7,4) Hamming code } -> Total of 10 bits
- Solution 2: source splitting [Willems ‘88, Urbanke-Rimoldi ’97]
- Can correct decoding be done if X and Y send 5 bits each ?

Symmetric Coding

- Solution: Map valid (X,Y) pairs into a coset matrix [SP & KR ‘00]



- Construct 2 codes, assign them to encoders
- Encoders → send index of coset of codes containing the outcome

Symmetric Coding

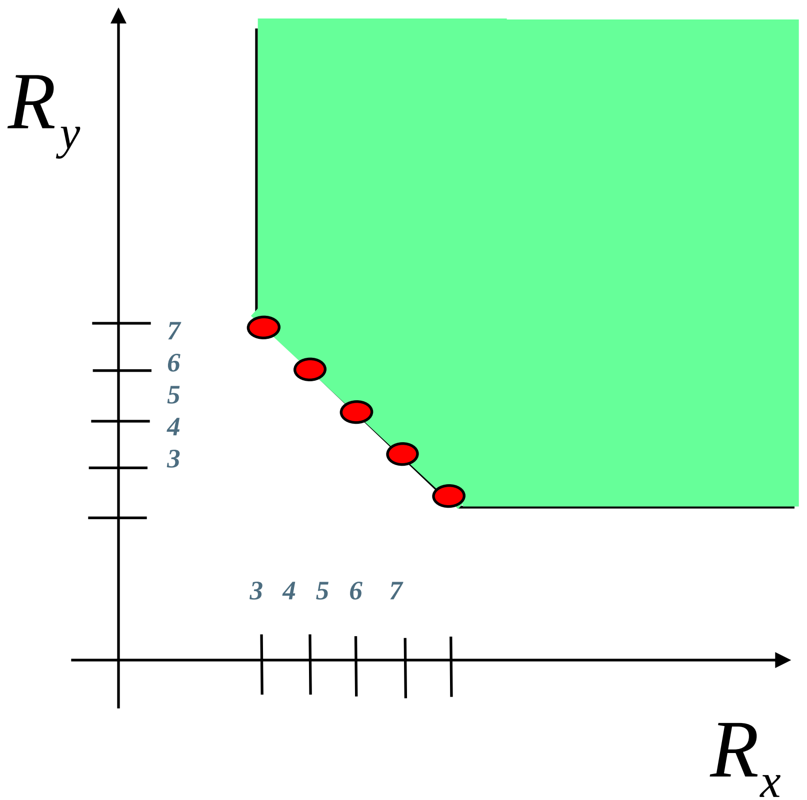
$$G = \left(\begin{array}{ccccccc} 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right) \Longrightarrow \quad G_1 = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$G_1, G_2 \Rightarrow 2 \times 7 \Rightarrow$ Syndromes are 5 bits long

- Decoder: Find a pair of codewords (one from each coset) that satisfy the distance criterion
- There exists a fast algorithm for this
- This concept can be generalized to Euclidean-space codes.

Rate Region

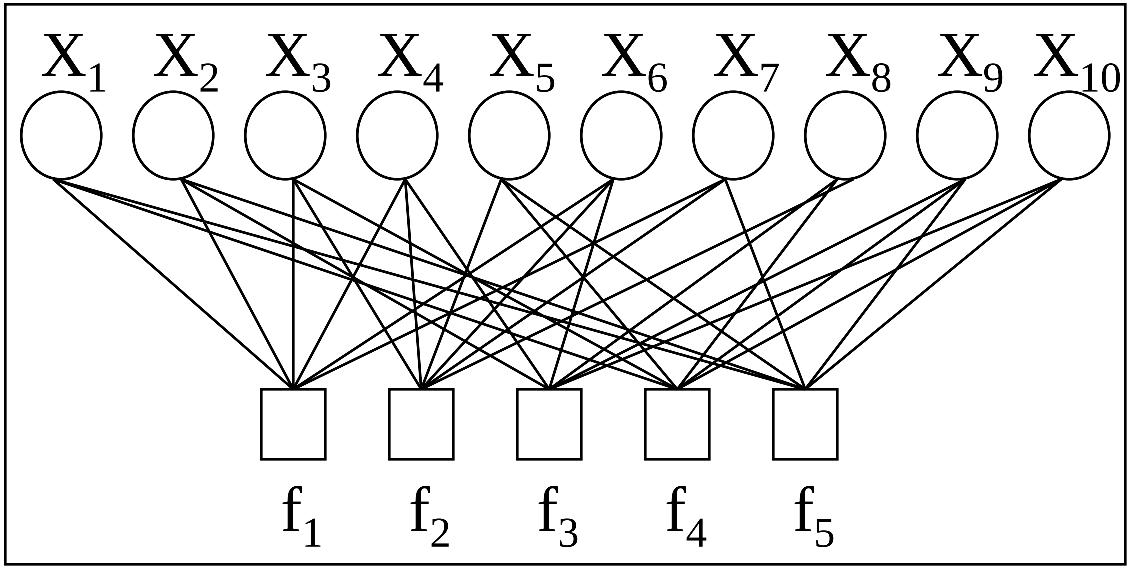
The rate region is: $\left\{ \begin{array}{ll} R_x, R_y : & R_x \geq 3, R_y \geq 3 \\ & R_x + R_y \geq 10 \end{array} \right\}$



- All 5 optimal points can be constructively achieved with the same complexity.
- All are based on a single linear code
- Can be generalized to arbitrary statistics [Schonberg et al. 2002]

LDPC Codes: Brief Overview

- Need linear codes $\Rightarrow$ use LDPC codes.
- Class of capacity approaching linear block codes.
- Sparse parity check matrix depicted by Tanner graph
  - Circles represent bits.
  - Squares represent constraints.

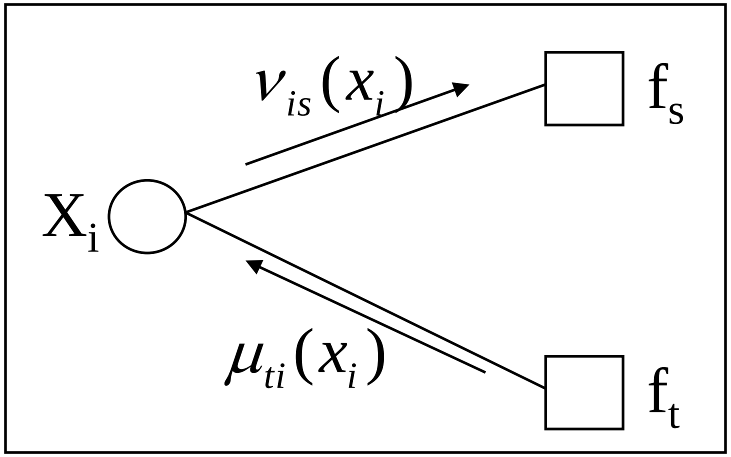


LDPC Codes Overview: decoding

- Decoded via message passing algorithm.
- Messages passed in two phases.
  - Update rules:

$$\nu_{is}(x_i) = \prod_{t \in N(i) \backslash s} \mu_{ti}(x_i)$$

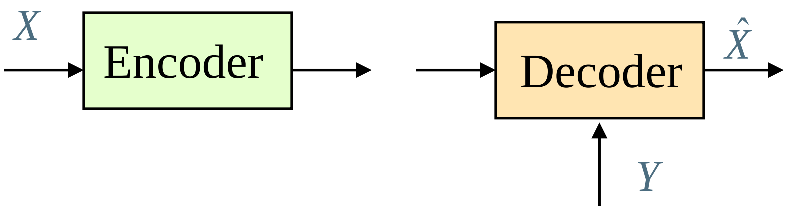
$$\mu_{si}(x_i) = \sum_{x_{N(s) \backslash i}} \left(f_s(x_{N(s)}) \prod_{j \in N(s) \backslash i} \nu_{js}(x_j) \right)$$



- Distribution of each variable estimated after n iterations.

$$p(x_i) = \frac{1}{Z} \prod_{s \in N(i)} \mu_{si}(x_i)$$

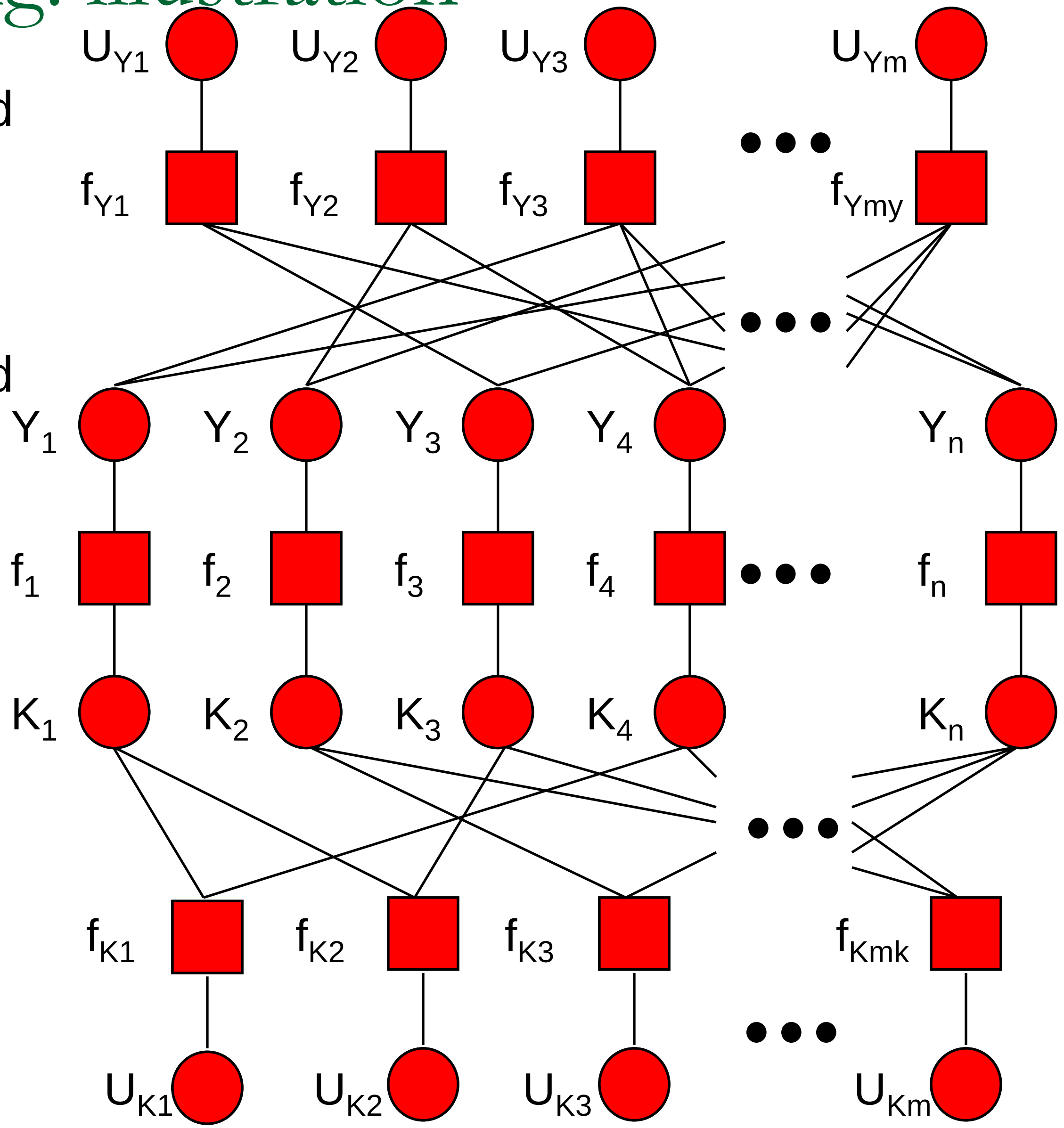
Source coding w/ side information at decoder



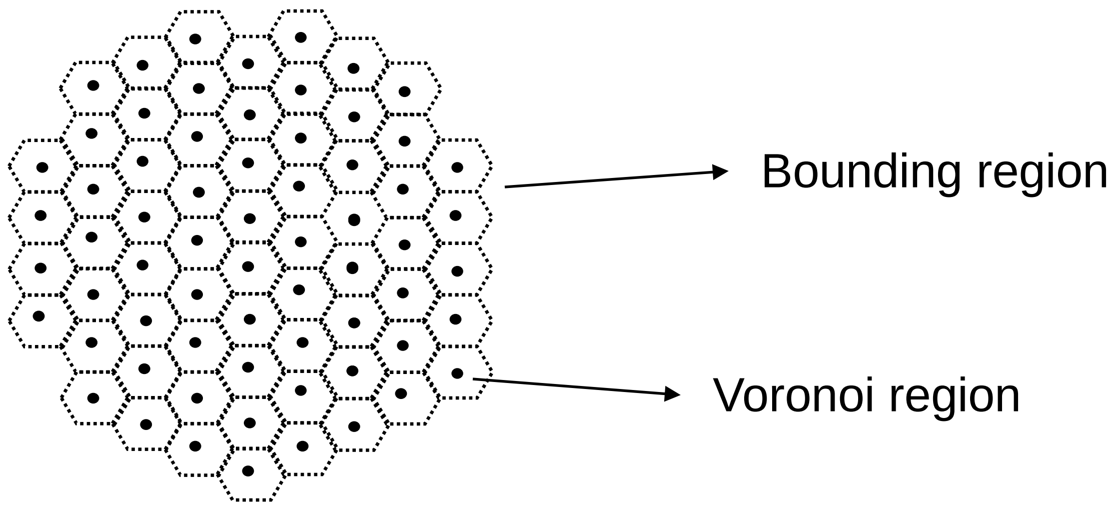
- X=Y+N, Y is arbitrary
- N is zero-mean Gaussian with variance σ_n^2
- Y and N are independent
- Quadratic distortion: $d(x, \hat{x})$
- Performance limit: $R_{WZ}(D) = \frac{1}{2}\log\left(\frac{\sigma_n^2}{D}\right)$
- Key idea: source codebook partitioned into cosets of channel codebooks
- Goal: computationally efficient way to construct
  - Source codebook (quantizer) with an encoding procedure
  - Partition of the quantizer into cosets of channel codebooks

Symmetric Coding: illustration

- Source bits, compressed bits, and LDPC code applied to Y
- Source bits, compressed bits, and LDPC code applied to K
- Correlation constraints



Standard source coding	Distributed source coding
▪ Boundary gain ~ 6 - 9 dB: achieved by entropy coding	▪ Coding gain ~ 6 - 9 dB: achieved by partition using LDPC channel codes
▪ Granular gain ~ 1.53 dB: achieved by vector quantization	▪ Granular gain ~ 1.53 dB: achieved by vector quantization

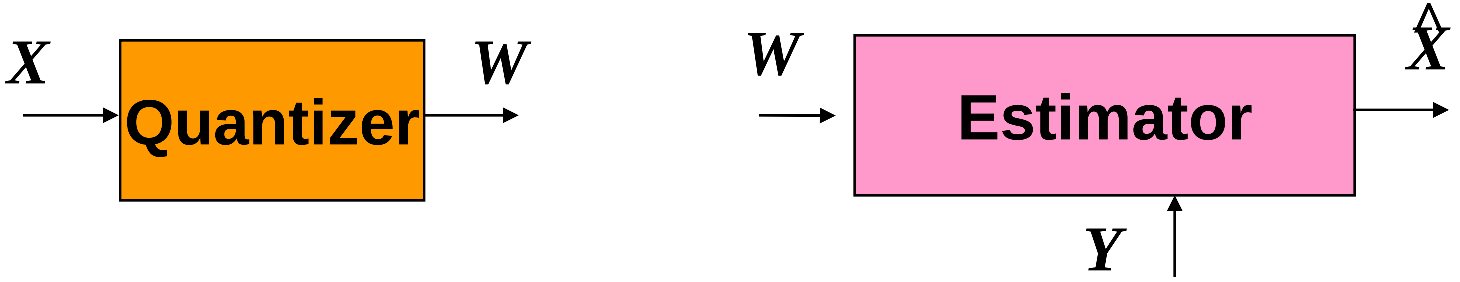


Standard source coding	Distributed source coding
▪ VQ and entropy coding can be done independently	▪ VQ and partition using channel codes cannot be done independently
▪ TCQ gets 1.36 dB of granular gain => within 0.15 dB from R-D limit	▪ Algebraic structure of TCQ does not "gel" well with that of LDPC codes
	▪ Need new block-coded quantization techniques!

Role of "source codes"

"Source Code": Desired distortion performance.

Joint quantization and estimation:



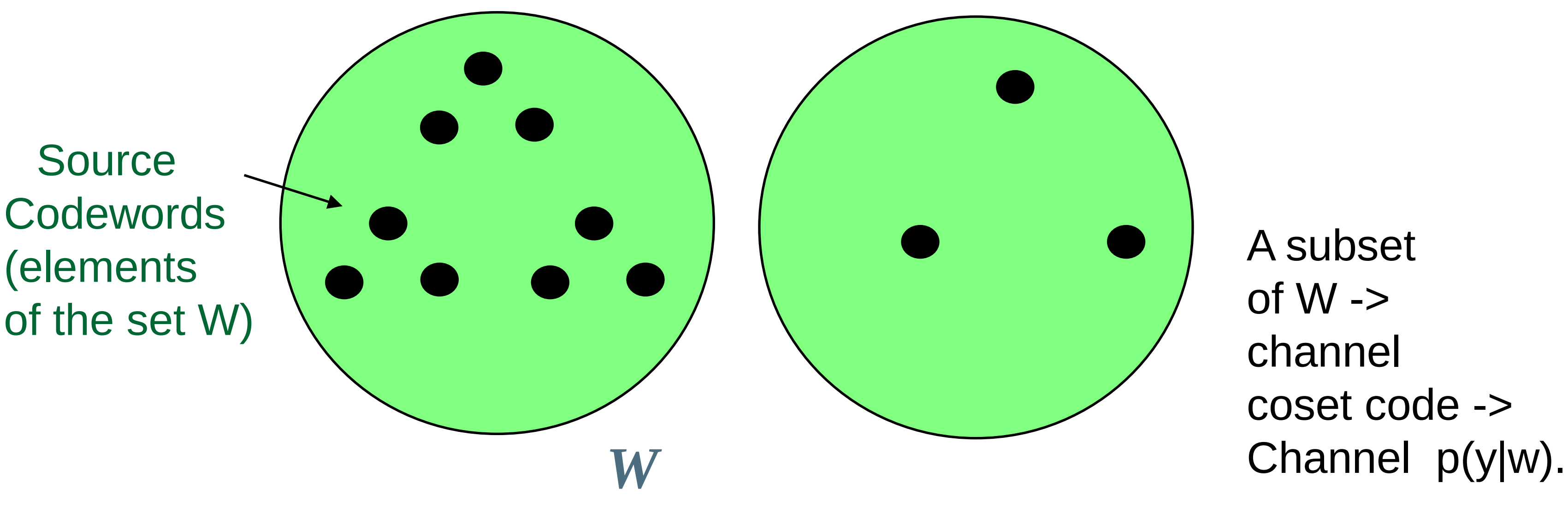
Active source codeword: Codeword -> X is quantized.

Quantization:
- Quantize X to W
- W => "Source Codes"

Estimation: Estimate X using W and Y.

Role of "channel codes"

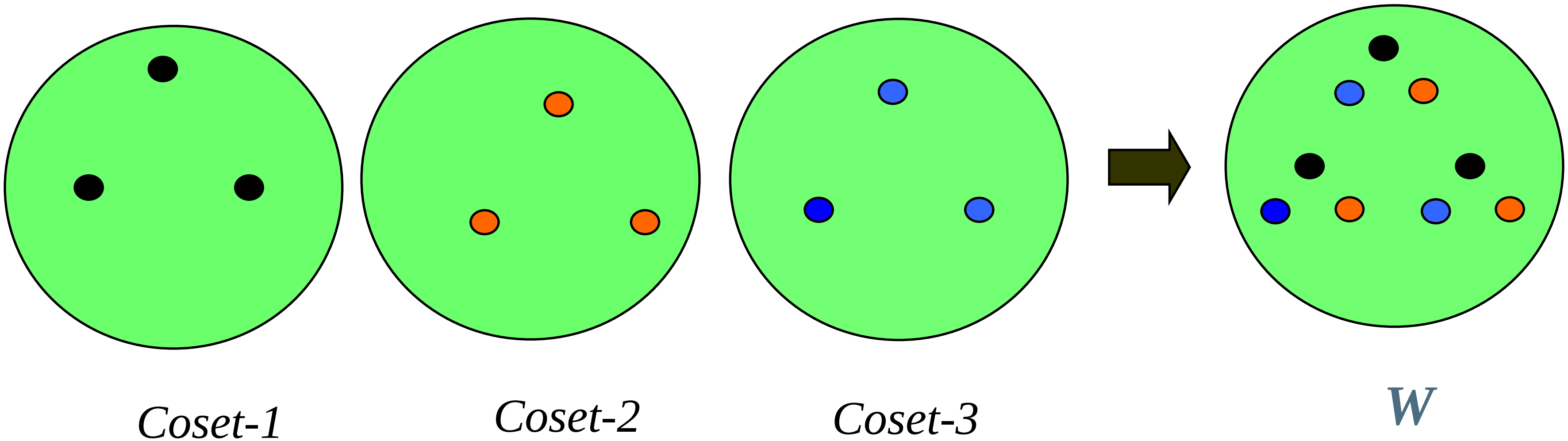
- "Channel Codes": Reduce source coding rate by exploiting correlation
- Partition of the "source codes" into cosets of "channel codes":



- Y and W are correlated=>induces an equivalent channel p(y|w).
- Build "channel coset codes" on W for channel p(y|w)

Role of “channel codes”

- Partition W into cosets of such “channel codes”.

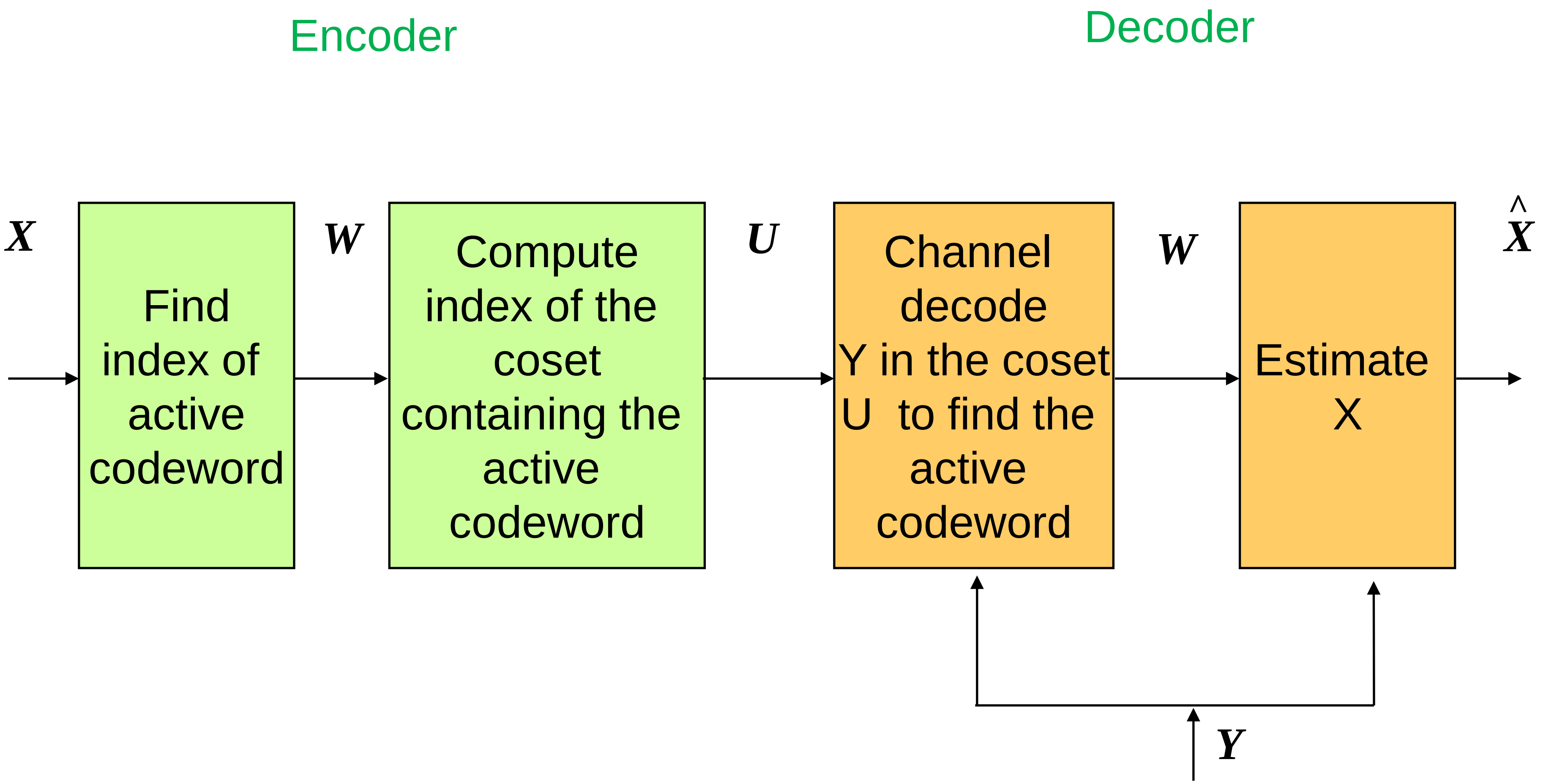


- Source code => a collection of channel codes.
- Send index of coset

Decoder:

- Recovers active source codeword by channel decoding Y in given coset
- Channel decoding fails => Outage

Encoder and Decoder Structure



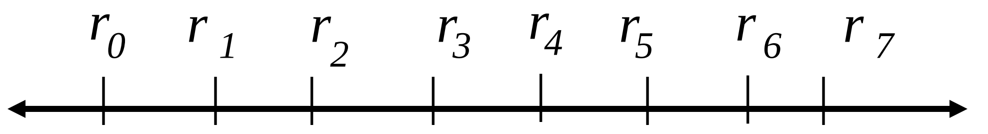
Distributed Source Coding Theory

- Source coding theory
  - Quantization
  - Indexing
  - Fidelity criterion
- Channel coding theory
  - Algebraic structure
  - Minimum distance
  - Prob. of decoding error
- Estimation theory
  - Estimation with rate constraints

Intricate Interplay

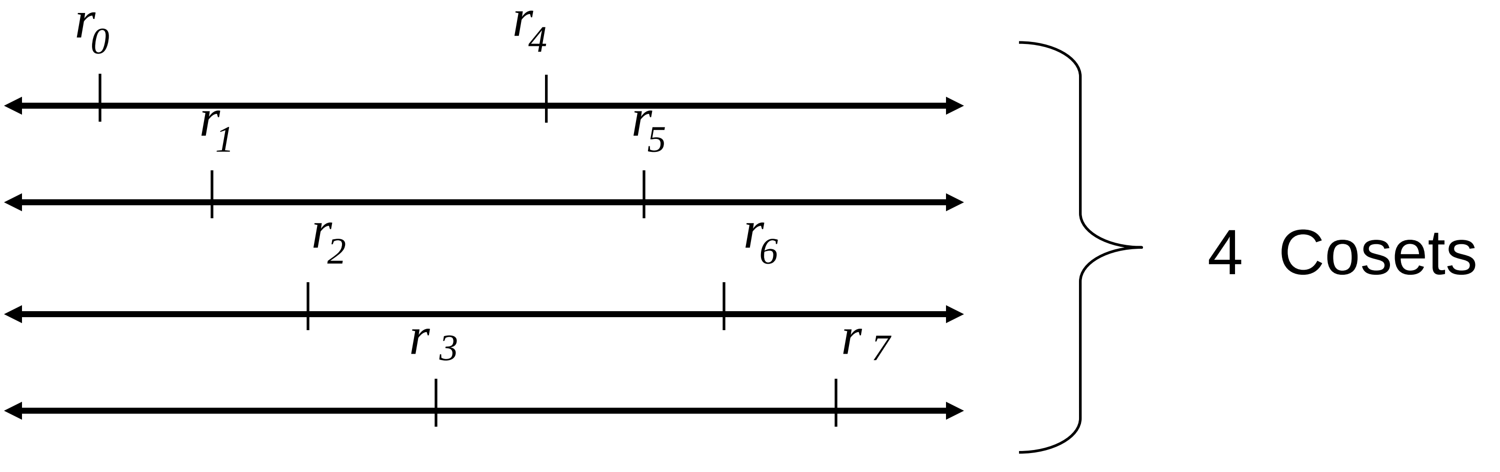
Basic Concept

Illustrative Example:



Consider a fixed-length scalar quantizer (say with 8 levels)

Partition:

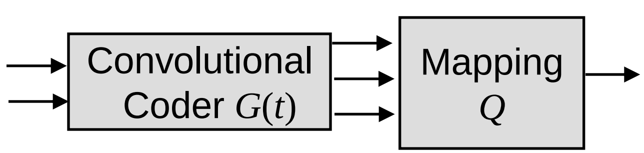


Trellis based coset construction

Example: Rate of transmission= 1 bit/ source sample.
Quantizer: fixed-length scalar quantizer ->8 levels.

- $C = \{ r_0, \ r_1, \ r_2, \ r_3, \ r_4, \ r_5, \ r_6, \ r_7 \}$ =>set of codewords.
- C^N has 2^{3N} sequences
- Partition C^N into 2^N cosets each containing 2^{2N} sequences.
- Use Ungerboeck trellis for such partitioning.

Trellis Coding:

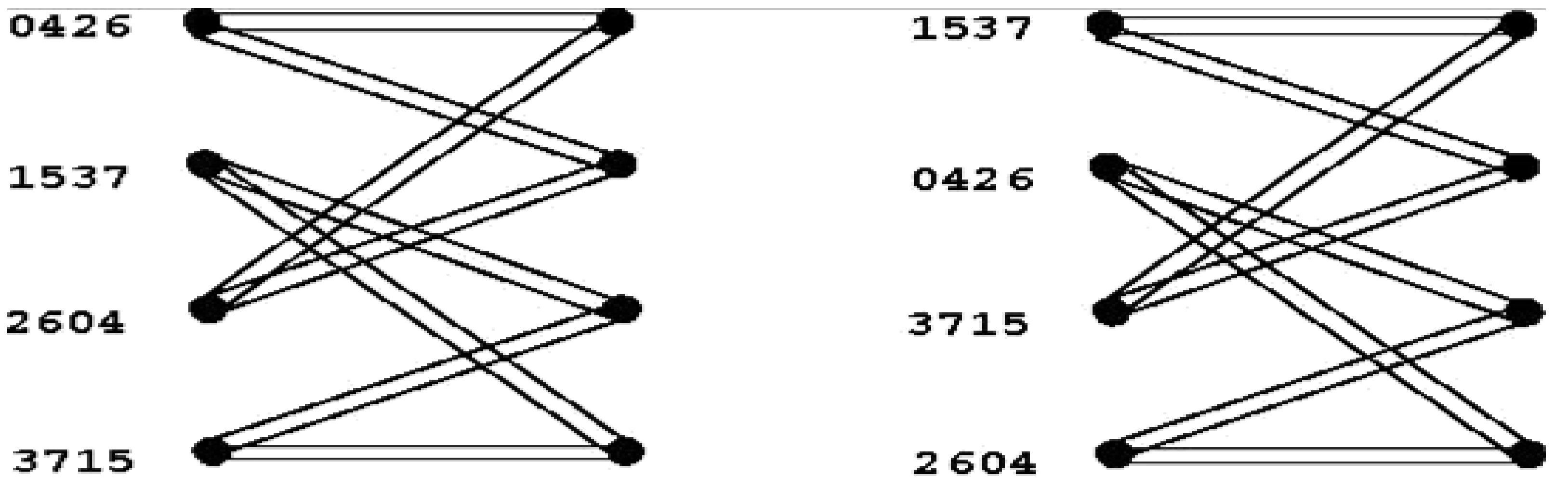


$$Q : \{0,1\}^3 \to C$$

- Sequences generated by this machine form a coset in space C^L
- Coset -> 2^{2N} sequences.

Trellis Partitioning:

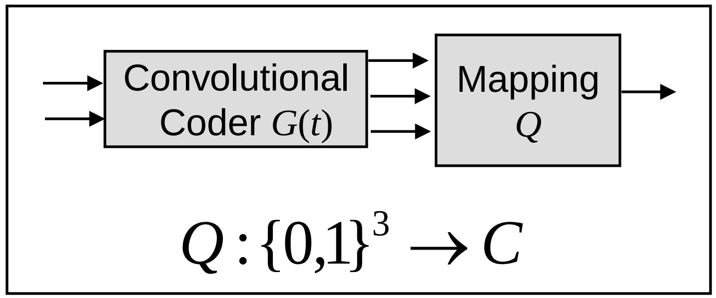
- Source Codebook = C^N
- Channel Codebook= set of sequences generated by the finite state machine
- Task: partition C^N into 2^N cosets, containing 2^{2N} sequences (in a computationally efficient way)



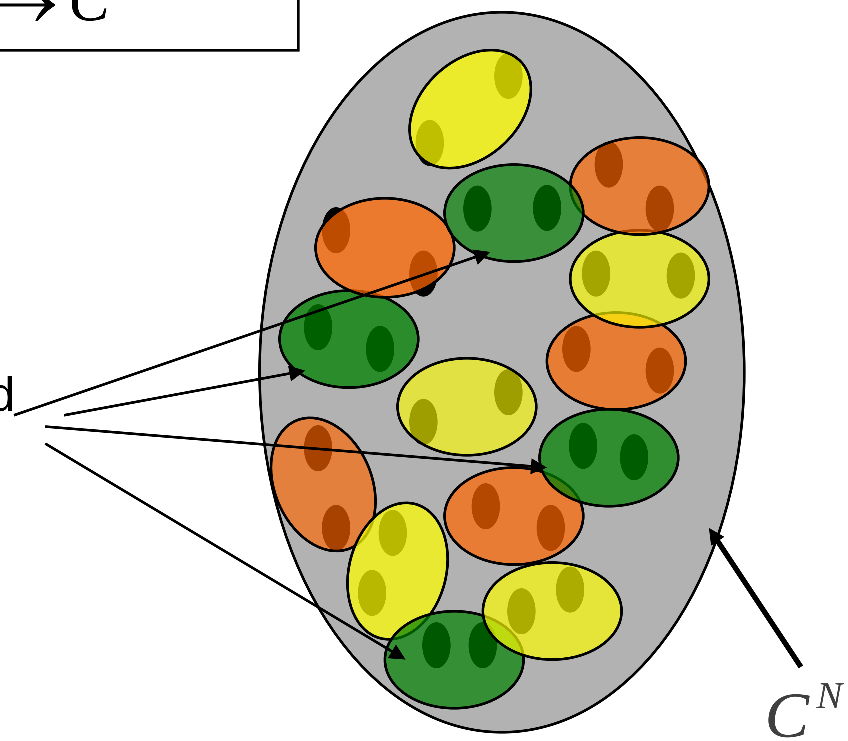
- Fast Encoding: Send syndrome sequence of active codeword
- Fast Decoding: Modified Viterbi algorithm using relabeling.

Trellis Partitioning

- Connection with earlier picture
  $C = \{r_0, r_1, \ldots, r_7\} \rightarrow$ codewords of scalar quantizer



Set of N-length sequences generated by the finite state machine



Simulation Results

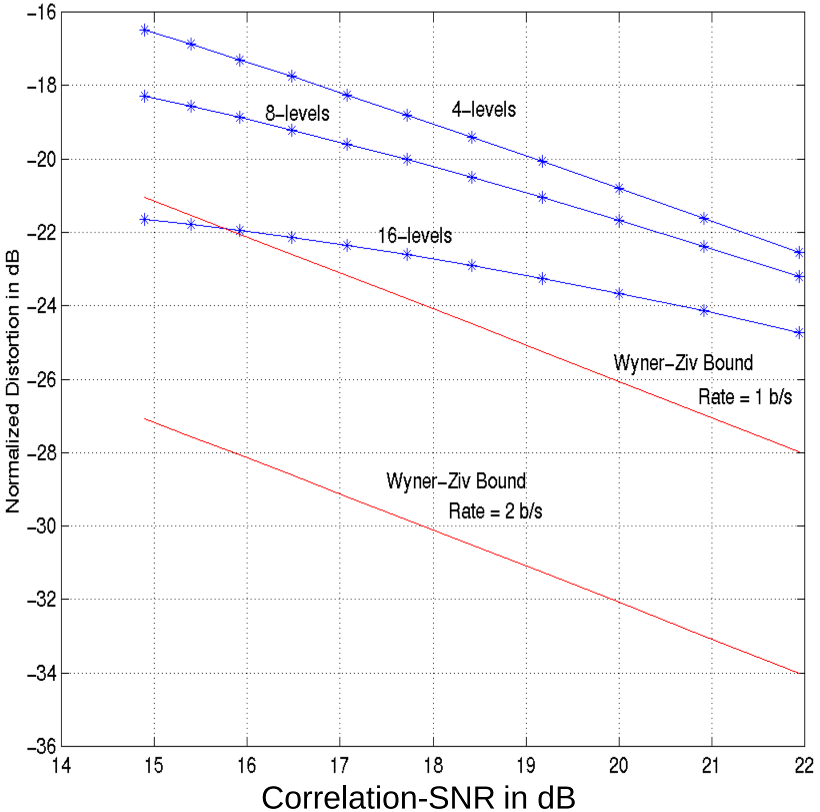
- Model:
  - Source: X~ i.i.d. Gaussian
  - Side information: Y= X+N, where N ~ i.i.d. Gaussian
  - Correlation SNR: ratio of variances of X and N
  - Normalized distortion: ratio of distortion and variance of X
  - Effective source coding rate = 1 bit per source sample

- Quantizers:
  - Fixed-length scalar quantizers with 4, 8 and 16 levels

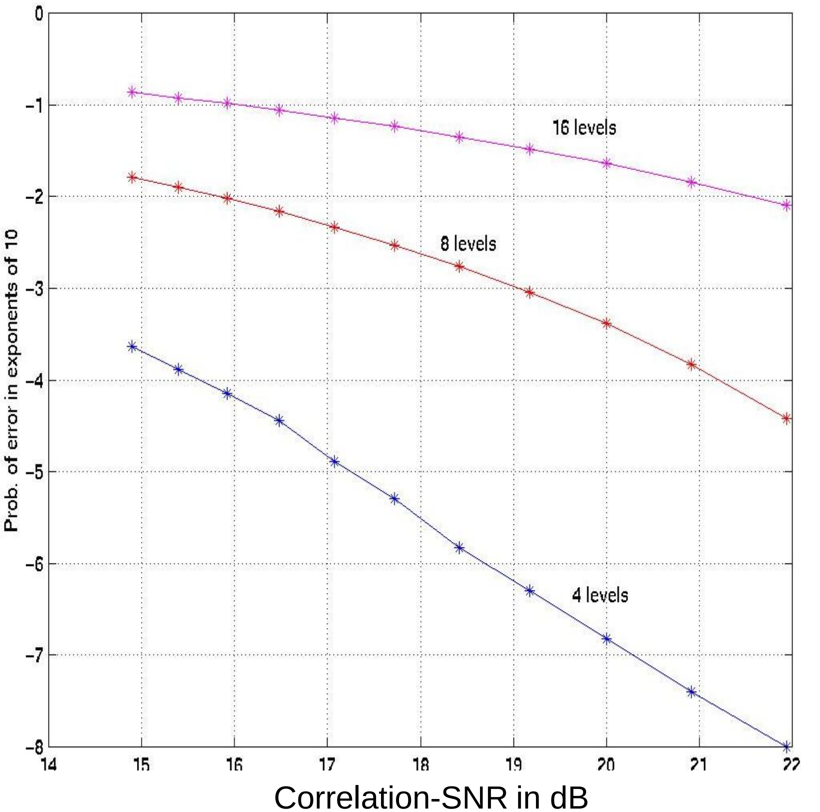
- Shannon R-D Bound: distortion= -6.021 dB at 1 bit/sample.

Simulation Results

- Distortion Performance

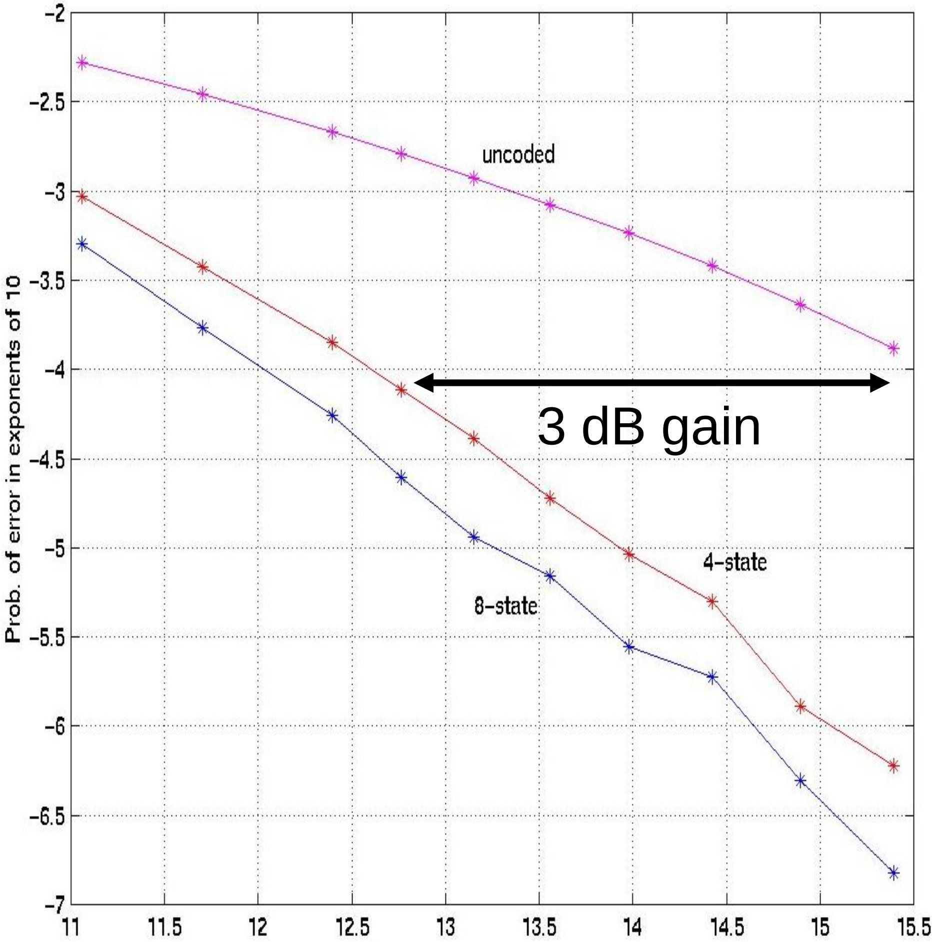


- Probability of error (uncoded system)

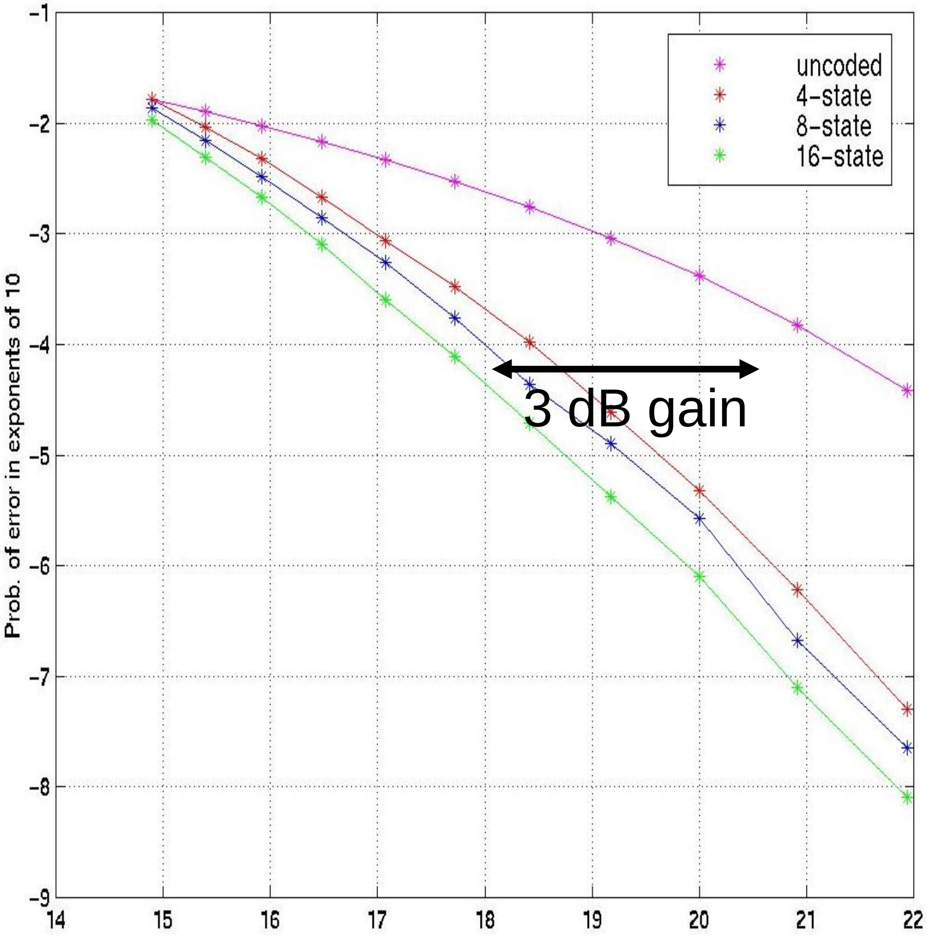


Probability of Error: Trellis coset coded system

4-level root scalar quantizer



8-level root scalar quantizer



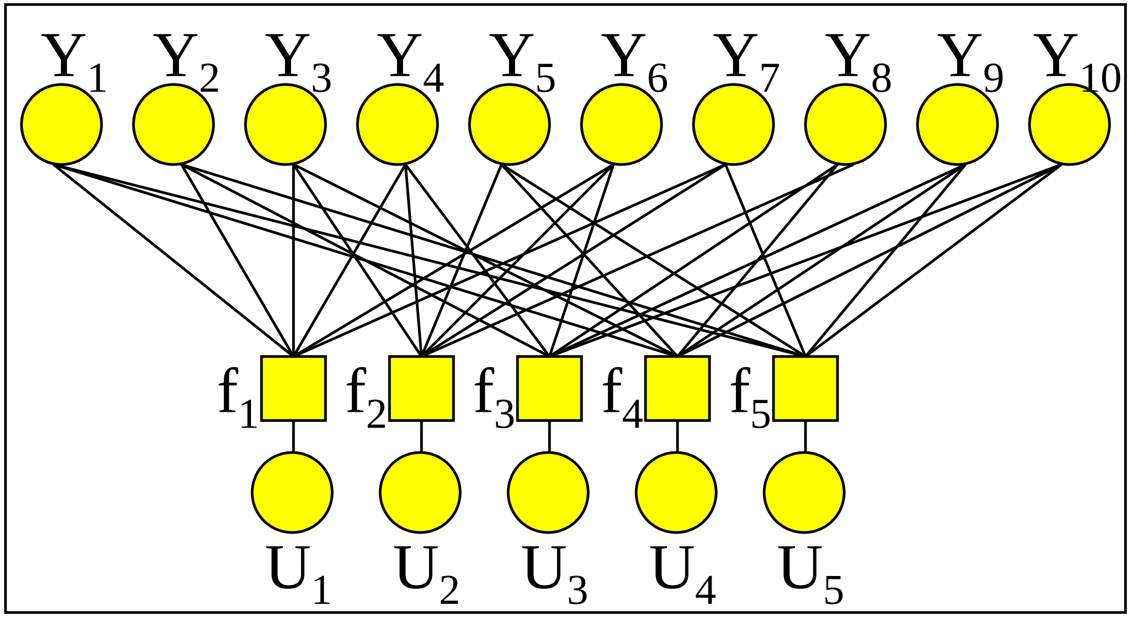
Gains (at C-SNR=18 dB) : Theoretical = 18 dB.
over Shannon bound DISCUS = 14 dB at Prob. of error $\leq 10^{-4}$

Approaches based on codes on graph

- Trellis codes → codes on graph to effect this partition
- Need good source code and good channel code
- Start with simple (not so good) source codebook and very good channel codebooks.
- Use belief propagation at the decoder to recover active source codeword

Reminder: graphical models

- Factor Graphs
  - Circles: Variables, Squares: Constraints
- Graphical representation for linear transformation
  - Y – source bits, U – compressed bits
  - Squares – Linear transformation Equations
- Transformation inversion: Belief propagation
  - Iterative application of inference algorithm



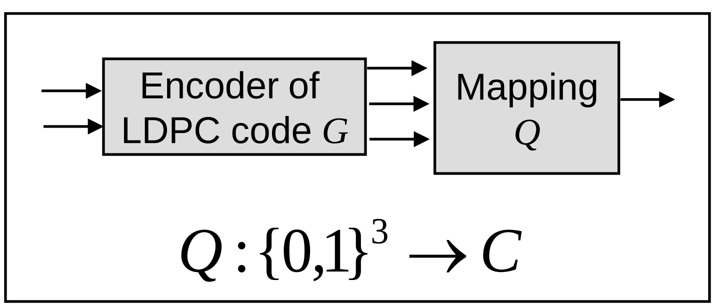
Approaches based on codes on graph

Xiong et al., Garcia-Frias et al.

Example: Rate of transmission= 1 bit/ source sample.
Quantizer: fixed-length scalar quantizer ->8 levels.

- $C = \{ r_0 , \; r_1 , \; r_2 , \; r_3 , \; r_4 , \; r_5 , \; r_6 , \; r_7 \}$ =>set of codewords.
- C^N has 2^{3N} sequences
- Partition C^N into 2^N cosets each containing 2^{2N} sequences.

Multi-level Coding using binary block codes of code rate 2/3

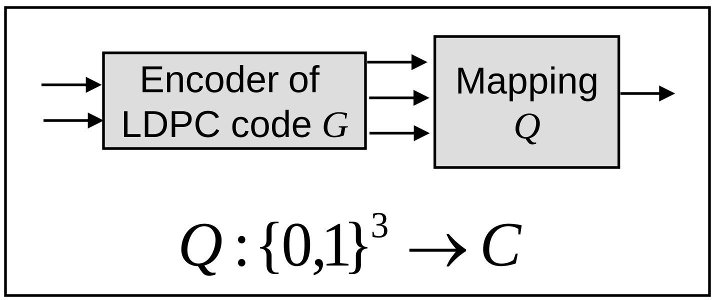


- Sequences generated by this machine form a coset in space C^L
- Coset -> 2^{2N} sequences.
- within 1.53 dB from R-D limit

Partitioning based on LDPC codes

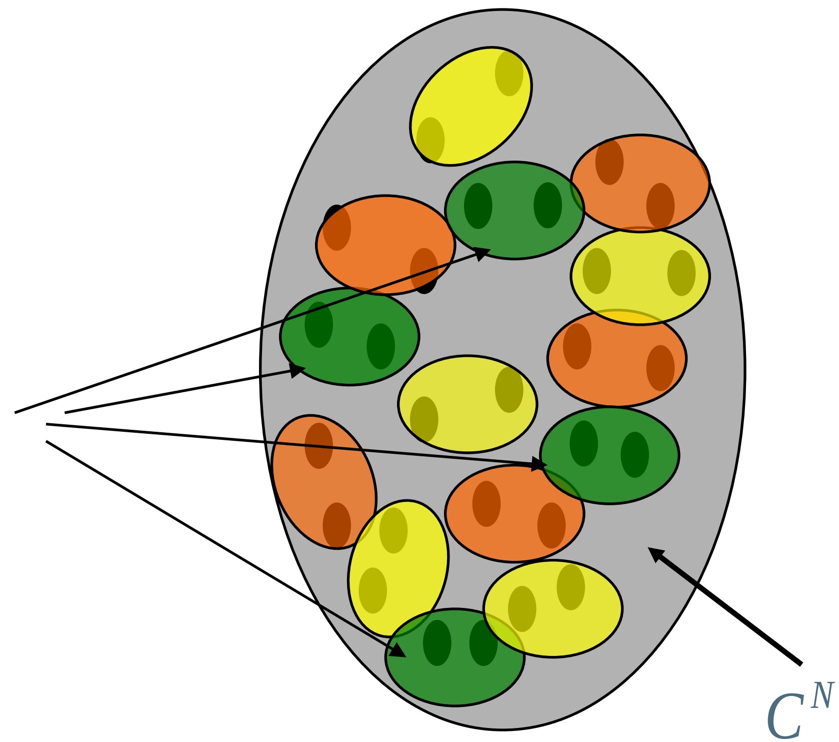
Connection with earlier picture:

$$C = \{ r_0, \; r_1, \; r_2, \; r_3, \; r_4, \; r_5, \; r_6, \; r_7 \}$$ => codewords of scalar quantizer

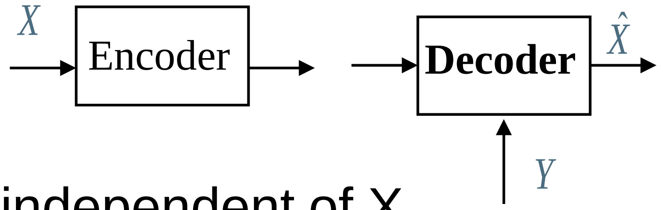


$$Q : \{0,1\}^3 \rightarrow C$$

Set of N-length sequences generated by the block code



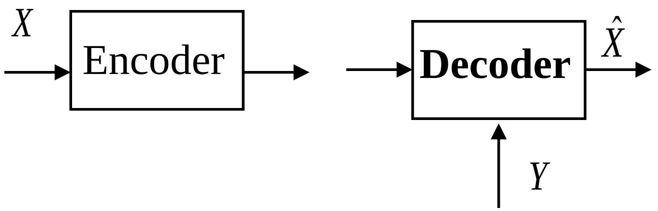
Binary Memoryless Sources



- X, Y: binary symmetric correlated sources
- Correlation: $Y = X \oplus Z$, Z is Bernoulli(p) and independent of X
- Distortion: $d(x, \hat{x}) = w_H(x \oplus \hat{x})$
- Goal:
  - Build a quantizer for X (U represents the quantized version)
  - Build a channel code for the channel with i/p U and o/p Y
  - Put a linear structure on both quantizer and channel code
  - Channel code is a subcode of the quantizer => induces a coset partition

Binary Memoryless Sources

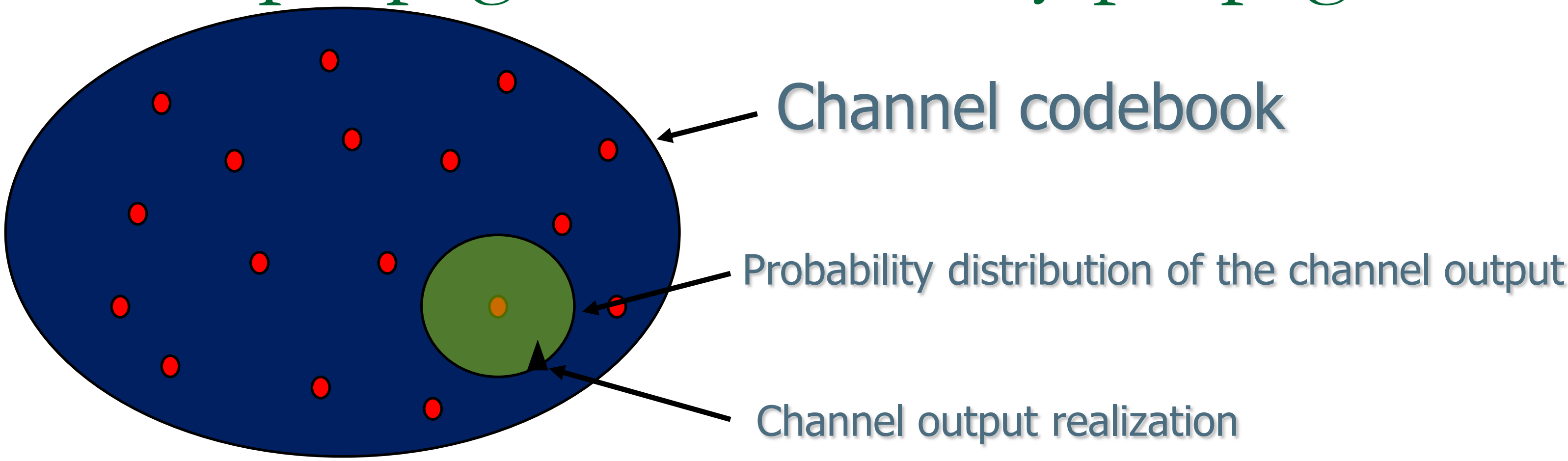
- Linear codes:



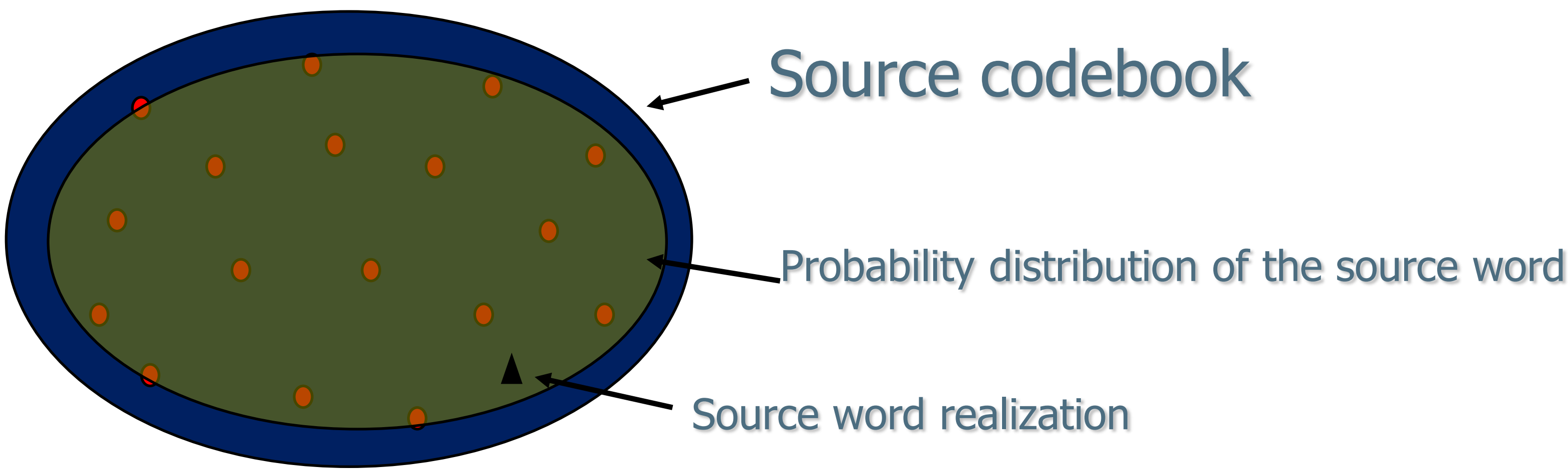
- Channel code:
  - Theory of binary linear channel codes → well-developed
  - LDPC codes with belief propagation (BP) algorithm
  - Gets the ultimate rebate I(U;Y)

- Block quantizer:
  - LDPC codes are not good quantizers, BP fails for quantization
  - A new theory of binary block quantizers
  - LDGM (low-density generator matrix) codes
  - Survey propagation (SP) algorithm [Mezard 2002, Wainwright-Martinian 2006]]

Belief propagation vs Survey propagation



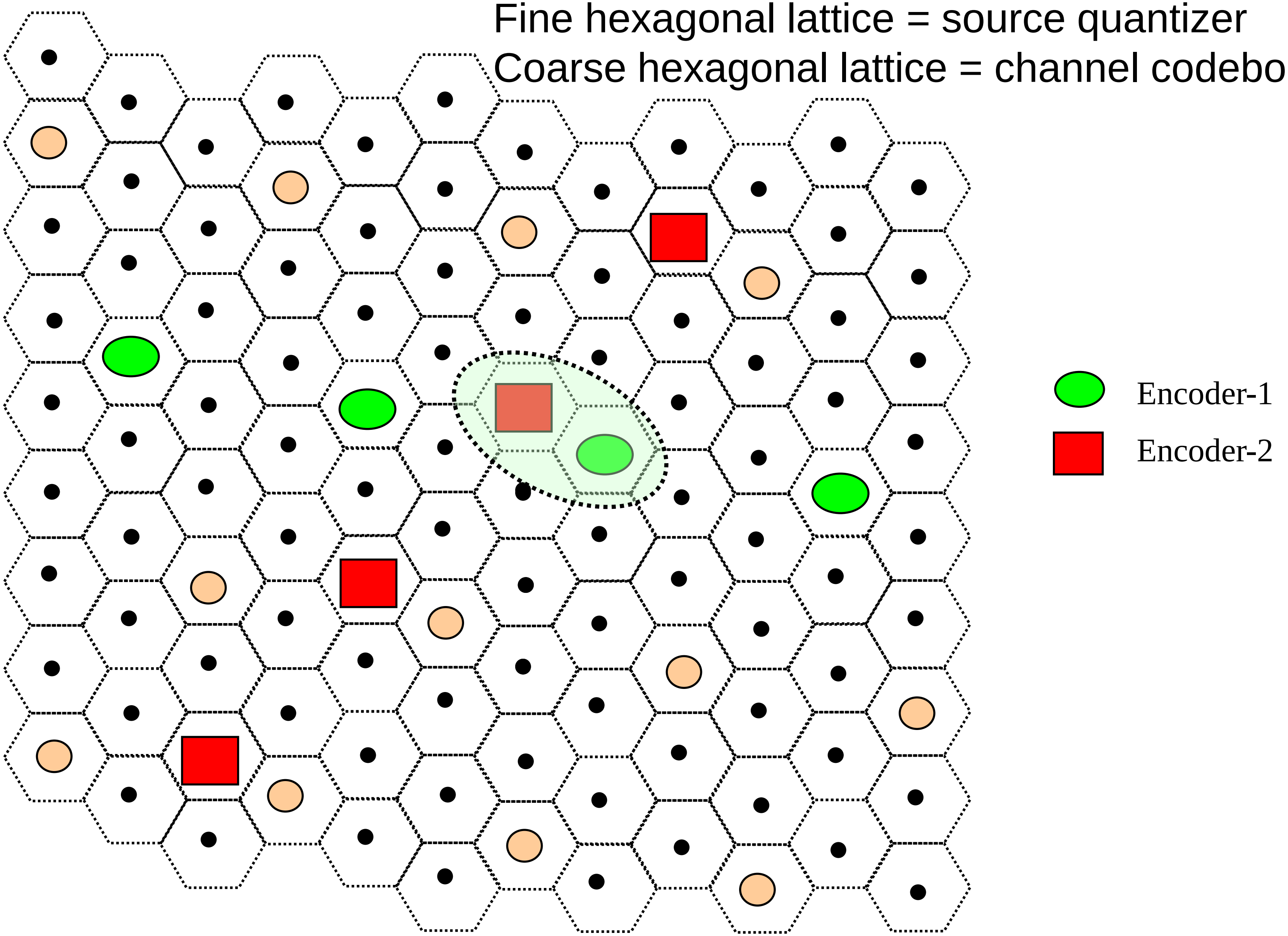
Channel decoding: belief propagation approximates min. distance decoding



Quantization: survey propagation approximates min. distance encoding

Lattice codes:

Fine hexagonal lattice = source quantizer
Coarse hexagonal lattice = channel codebook



BREAK
10.50 AM-11.00 AM

Outline

- Session I. Introduction and theory : 9.00 am-10.00 am
  - Motivation and intuition
  - Distributed source coding foundations
  - Break: 10.00-10.10 am
- Session II. Constructions: 10.10 am-10.50 am
  - Structure of distributed source codes
  - Constructions based on trellis codes & codes on graphs
  - Break: 10.50-11.00 am
- Session III. Connections and Applications: 11.00 am-12.00 noon
  - Overview of connections and applications with snippets
  - Compression of encrypted data
  - Distributed video coding

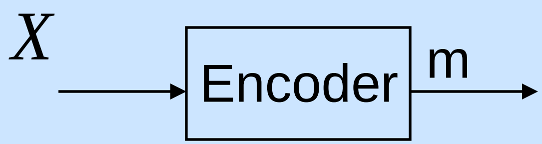
Connections and Applications

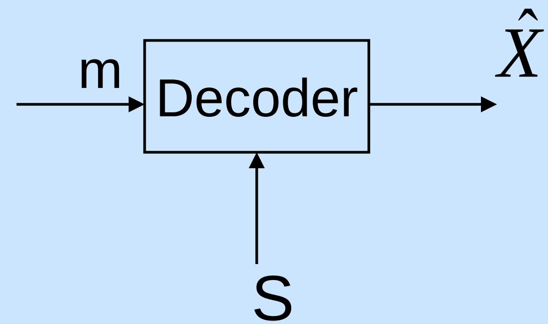
- Fundamental duality between source coding and channel coding with side-information
  - Media security: data-hiding, watermarking, steganography
- Digital upgrade of legacy analog systems
- M-channel Multiple Description codes
- Robust rate-constrained distributed estimation (CEO problem)
- Media broadcast using hybrid analog/digital techniques

- Distributed compression in sensor networks
- Compression of encrypted data
- Distributed Video Coding

Duality b/w source & channel coding with SI

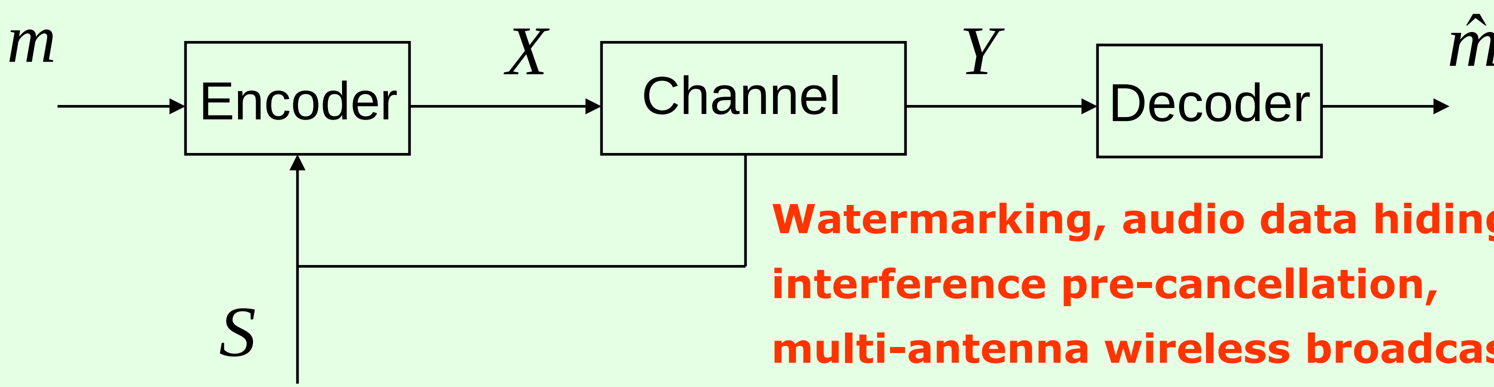
- Source coding with side information (SCSI)





Sensor networks, video-over-wireless, multiple description, secure compression

- Channel coding with side information (CCSI)

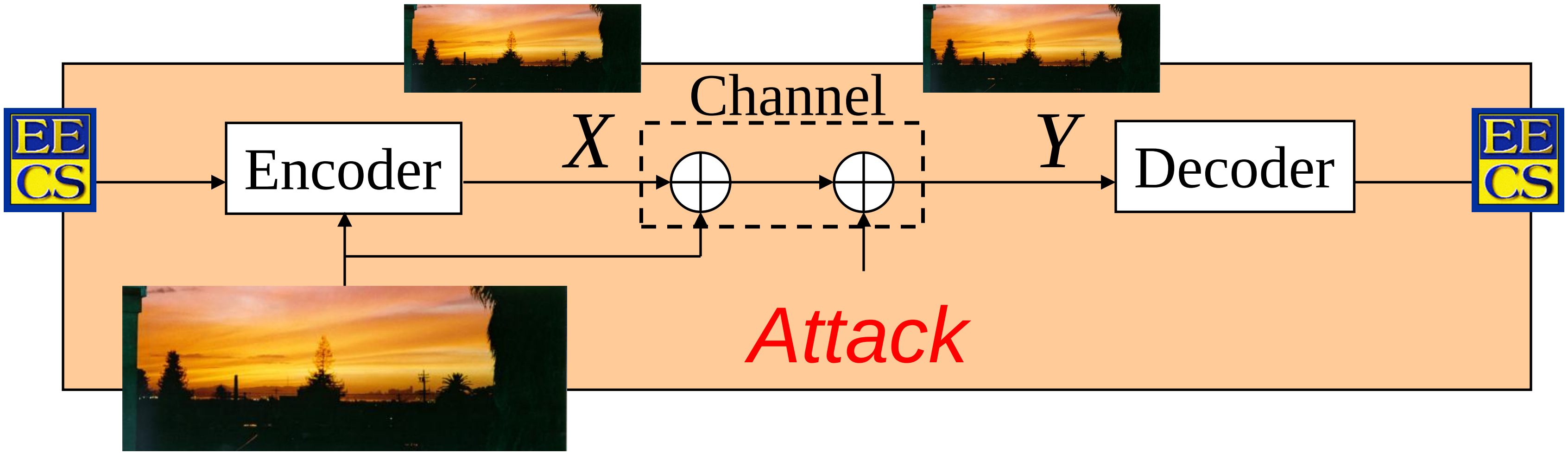


Watermarking, audio data hiding, interference pre-cancellation, multi-antenna wireless broadcast.

SP, J. Chou and KR, Trans. on IT, May 2003

Multimedia Watermarking and Data Hiding

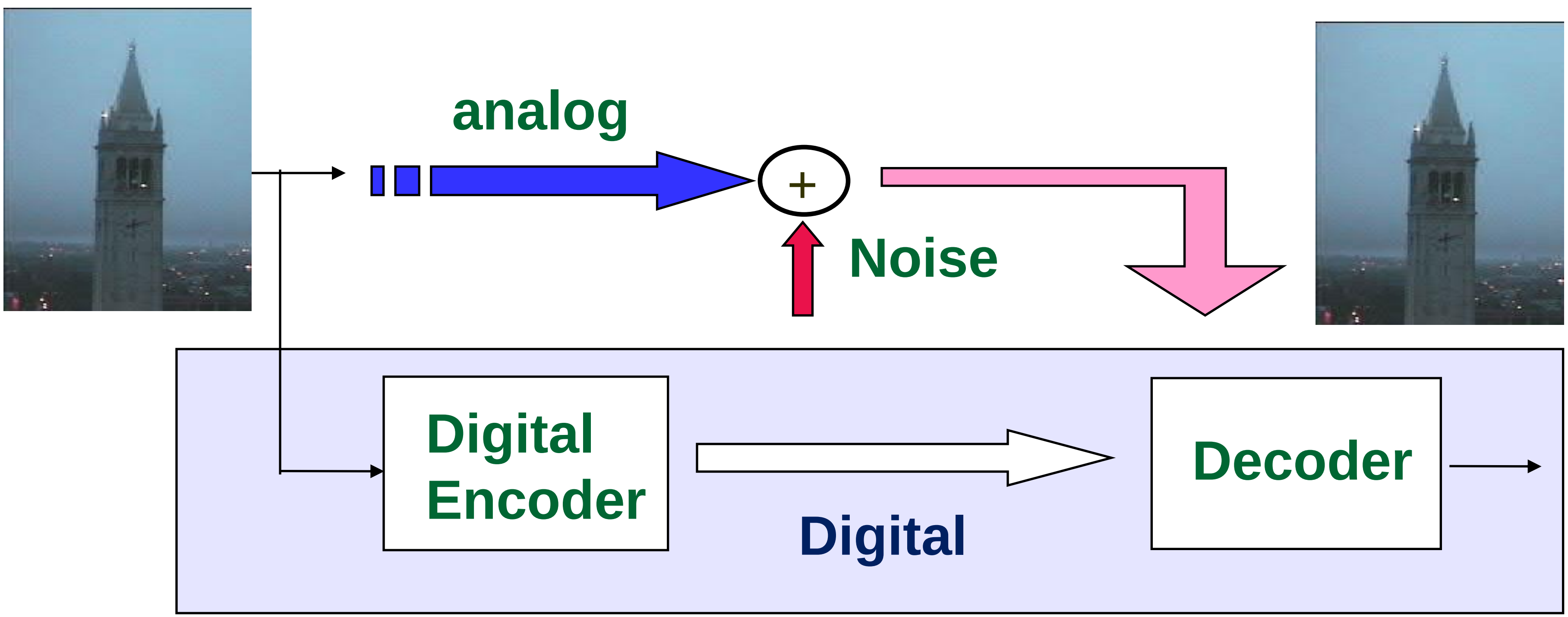
Embed (authentication) signature that is robust



- The encoder sends watermarked image X
- Attacker distorts X to Y
- Decoder extracts watermark from Y

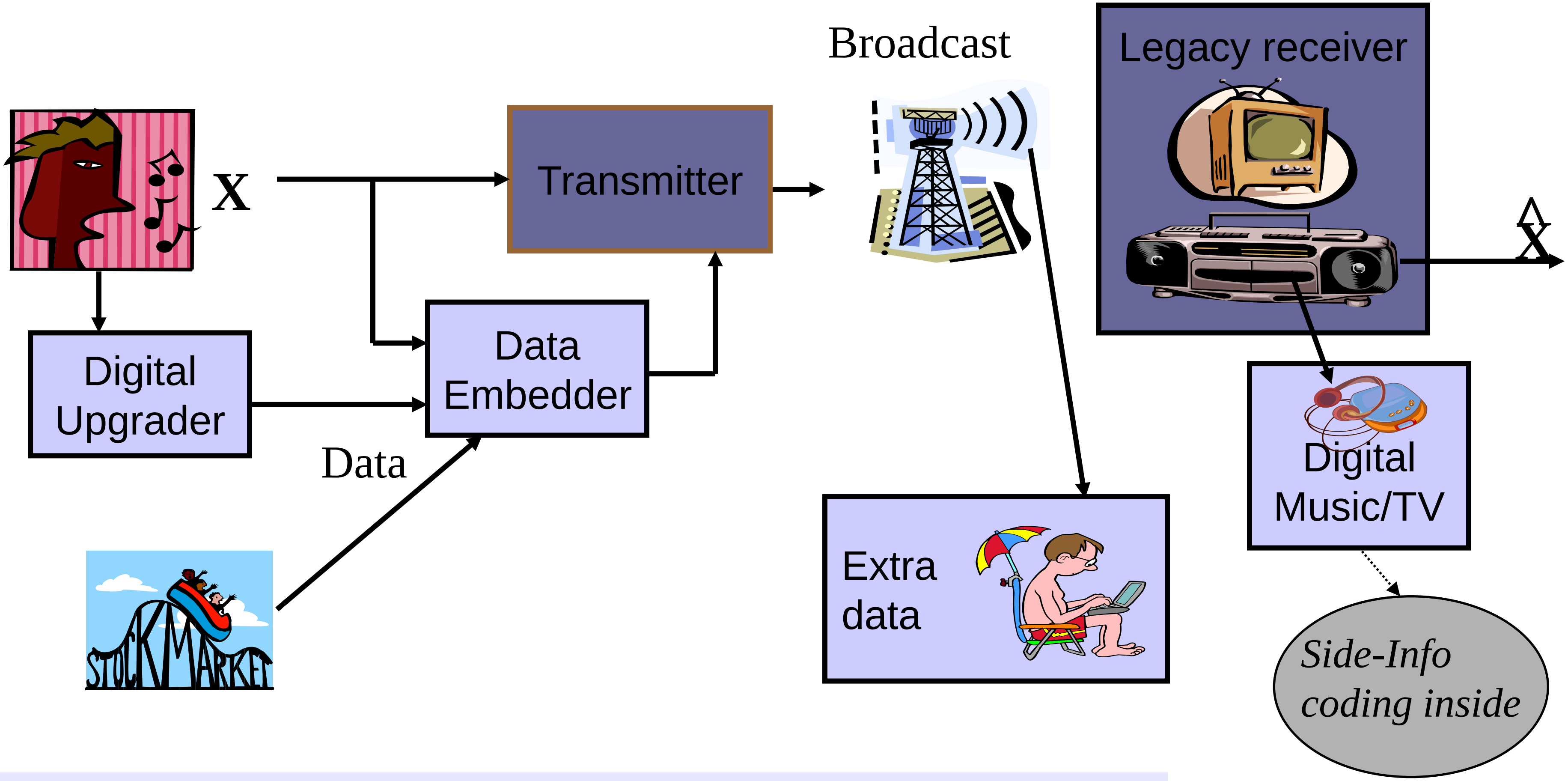
Application: digital audio/video simulcast

Ex.: Upgrading NTSC to HDTV with a digital side-channel. (also PAL+)



SP & KR,, DCC '01

Application: spectrum "recycling"

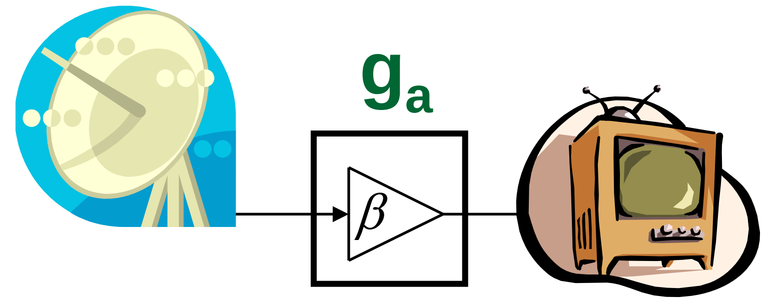


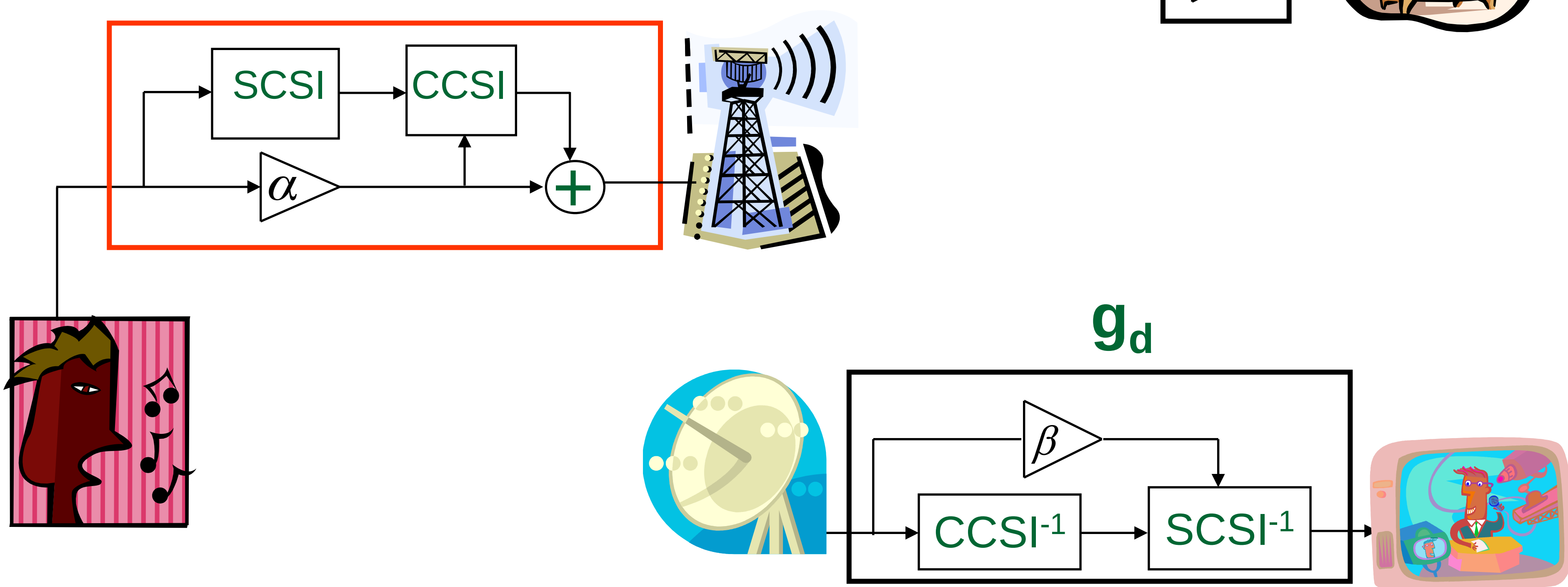
What is the optimal tradeoff between simultaneously delivered analog an digital quality? ➔ Need a combination of SCSI/CCSI

R.Puri, V. Prabhakran & KR, Trans. On Info Theory, April '08

Hybrid Analog-Digital Simulcast

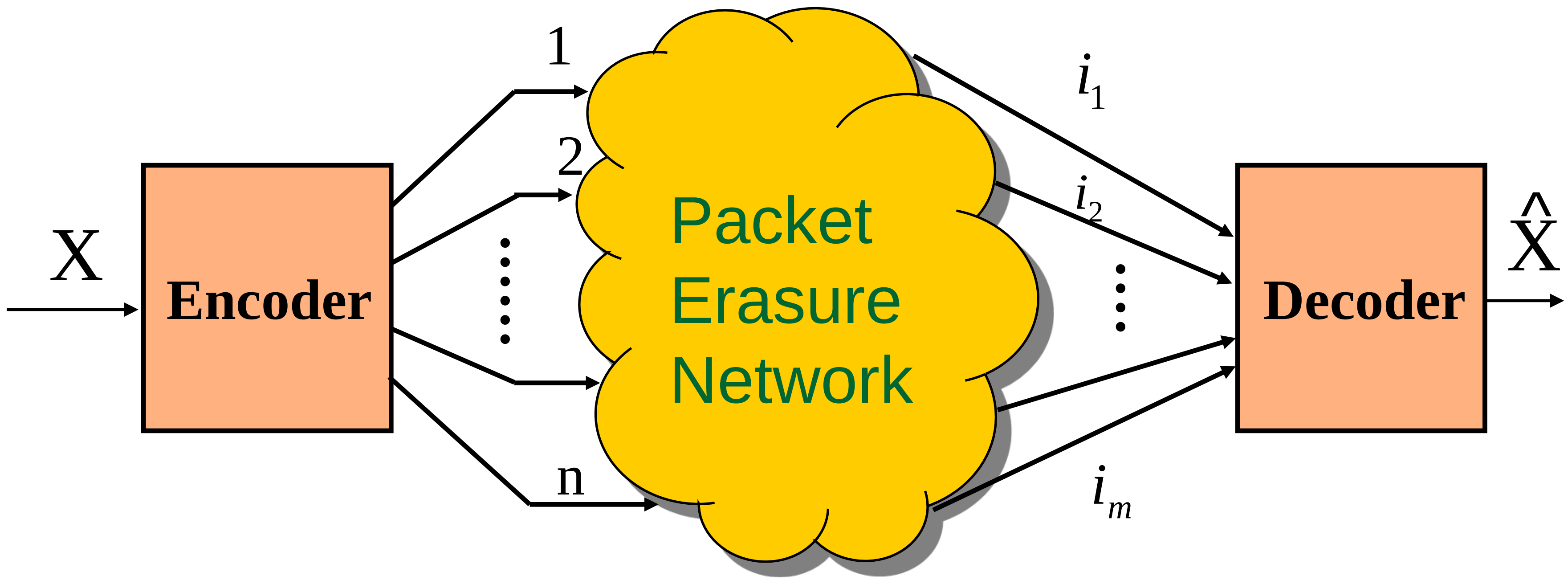
What is noise for analog receiver
is music for digital receiver!





R.Puri, V. Prabhakran & KR, Trans. On Info Theory, April '08

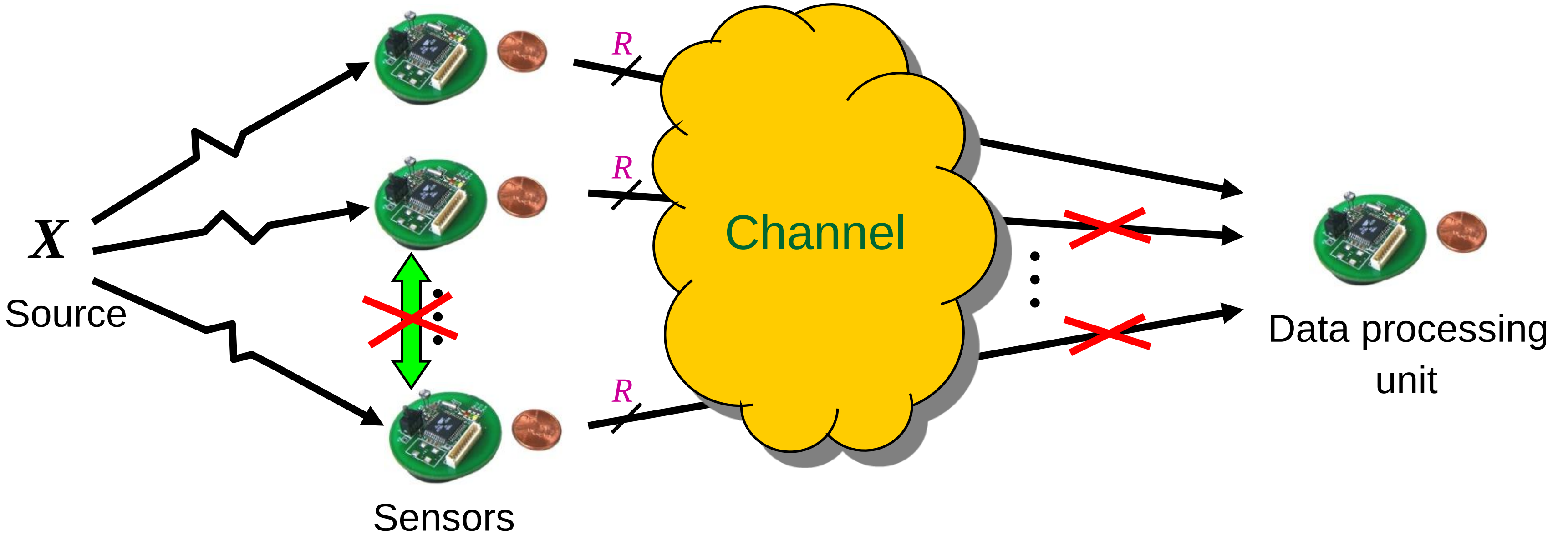
Multiple Description (MD) coding



- Packet erasure model: ***some*** subset of packets reach decoder.
- Connection to DSC: Uncertainty re. ***which*** packets reach decoder?
- Fundamental connection between MD coding and distributed source coding: leads to **new achievable rate results**!

R.Puri, SP & KR (Trans. on IT- Jan 04, Apr 05)

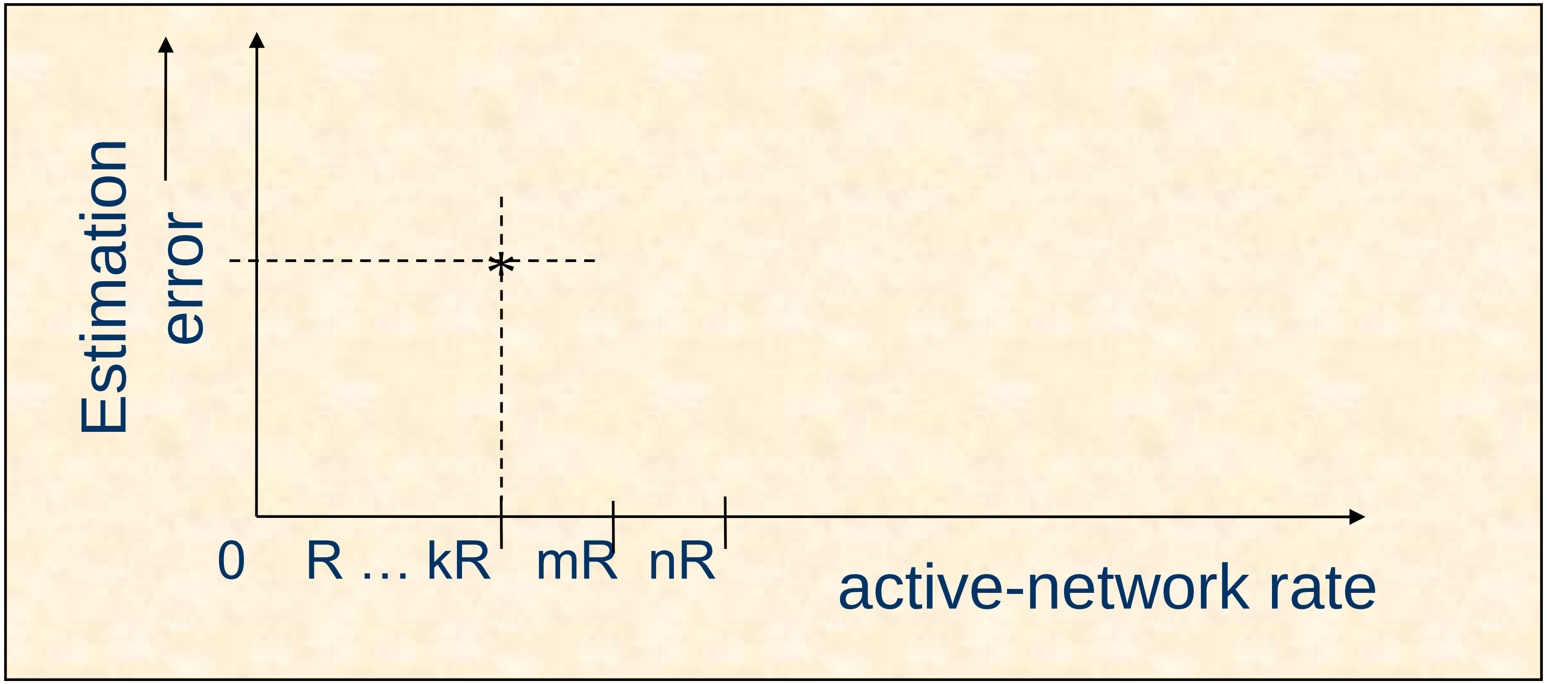
Distributed rate-constrained estimation



- Sensors make noisy (correlated) measurements of a physical quantity ***X*****,** e.g., temperature, pressure, seismic waves, audio data, video signals, etc.
- Central decoder needs to estimate ***X*** (many-one topology).
- Power conservation, node failures, communication failure => need **robustness**.

Robust distributed rate-constrained estimation

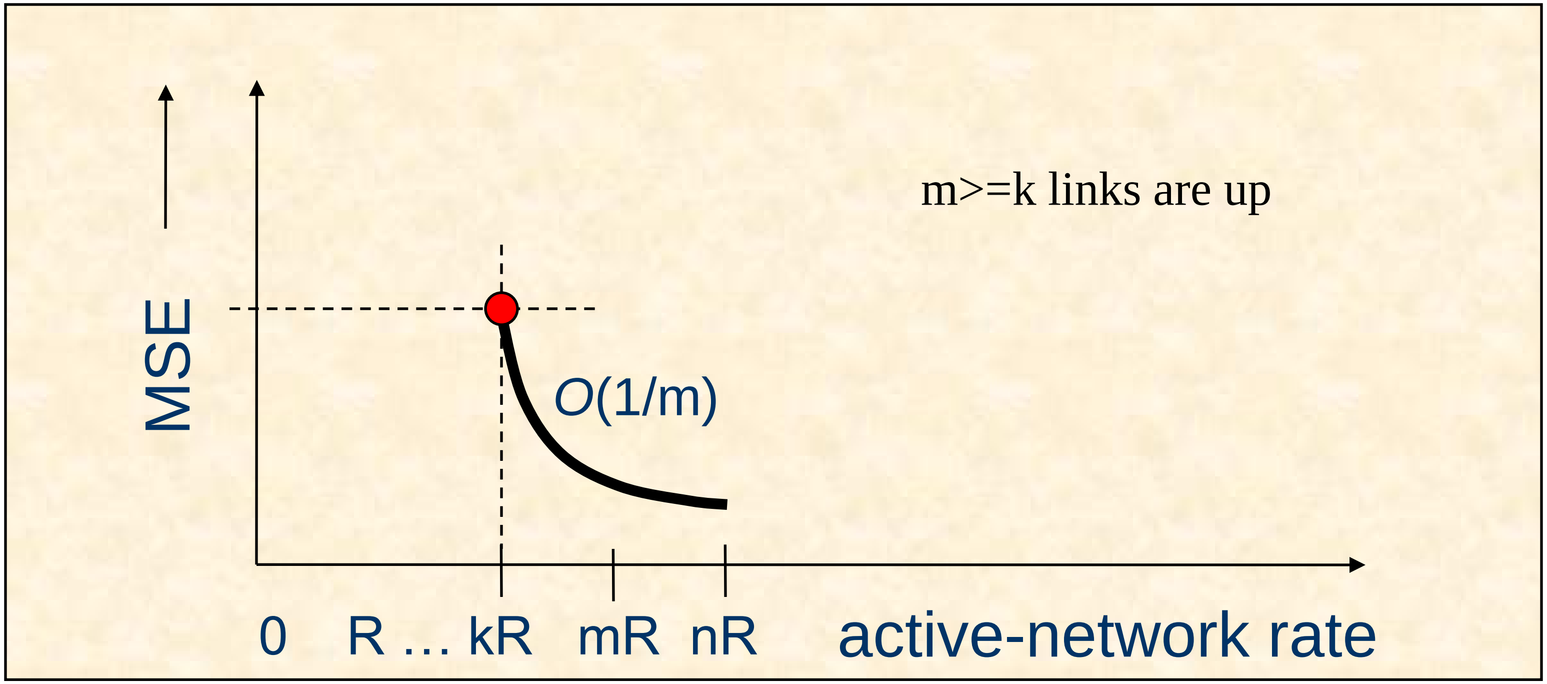
- For a (***k***, ***k***) **reliable Gaussian** network, full range of rate-MSE tradeoff: [Oohama, IT 1998].



Robust distributed rate-constrained estimation

- For a (***k***, ***k***) **reliable Gaussian** network, full range of rate-MSE tradeoff: [*Oohama, IT 1998, Prabhakaran, Tse & KR ISIT 2001*].

- For an (***n***, ***k***) **unreliable** Gaussian network, can match above performance for the reception of **any *k*** packets and get better quality upon receiving more packets!
- => Robustness without loss in performance.



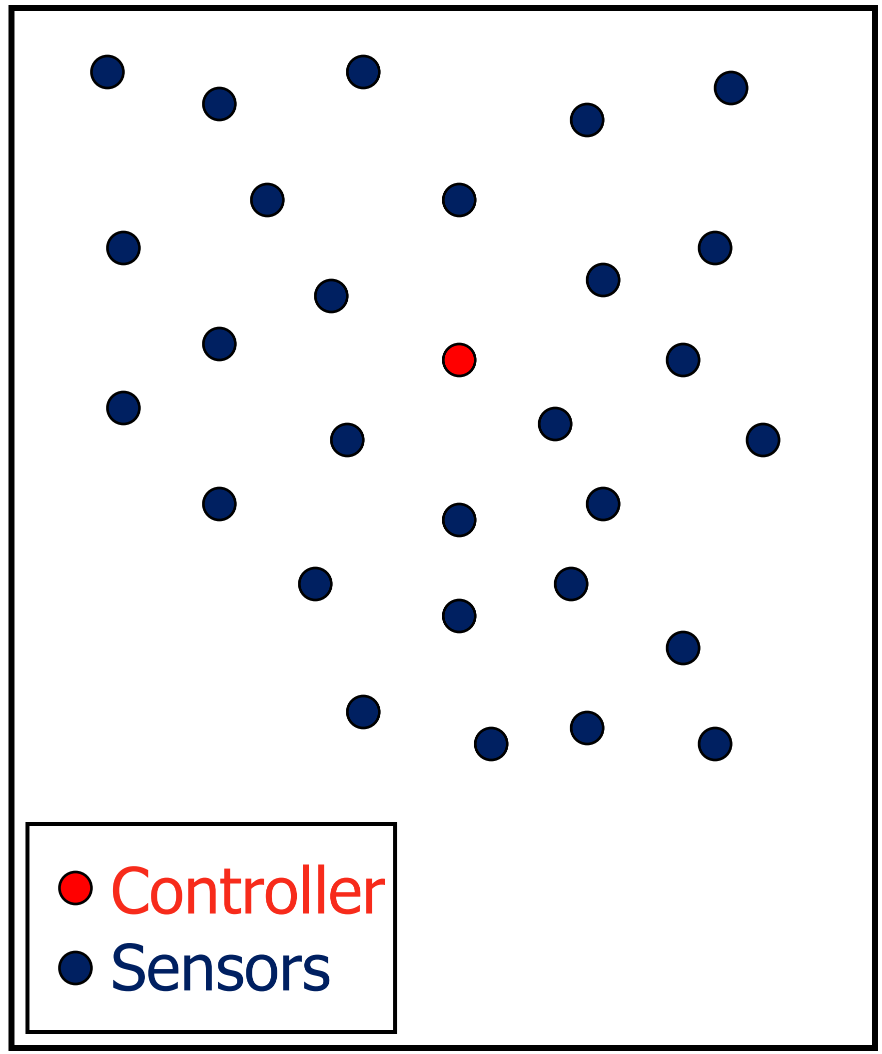
P.Ishwar, R. Puri, SP & KR, IPSN'03.

Adaptive filtering for distributed compression in sensor networks

J. Chou, D. Petrovic & KR: "A distributed and adaptive signal processing approach to exploiting correlation in sensor networks." Ad Hoc Networks 2(4): 387-403 (2004).

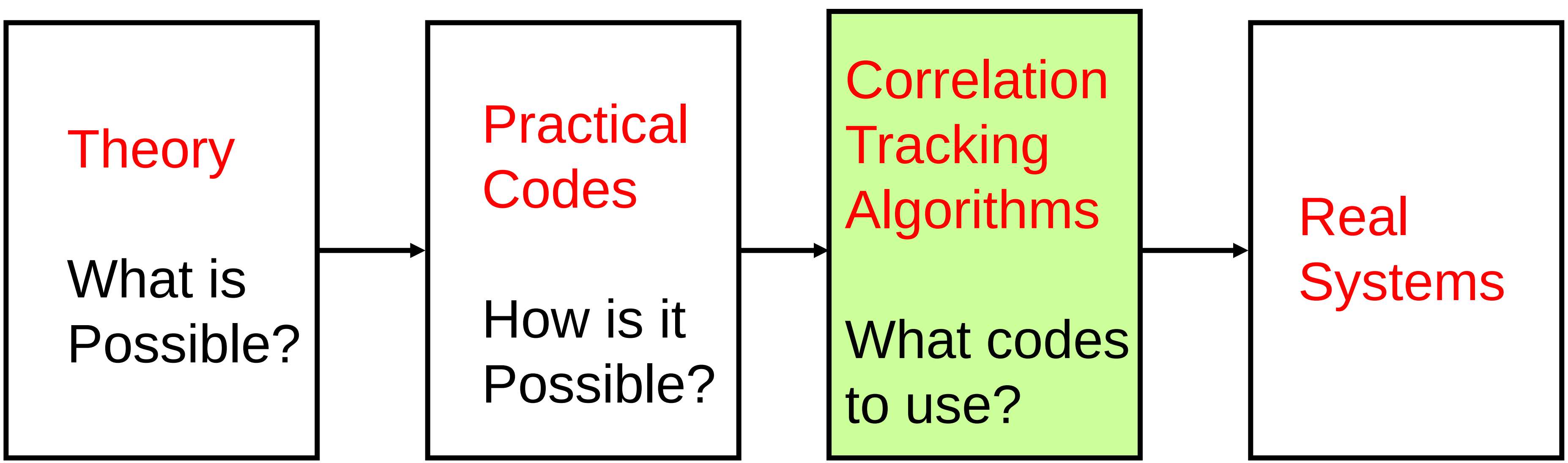
Deployment setup

- Network consists of many sensors, a few controllers, and a few actuators
- Sensors give their measurements to controllers, which process them and make decisions
- Many sensors have highly correlated data
- It would be beneficial to exploit this correlation to compress sensor readings



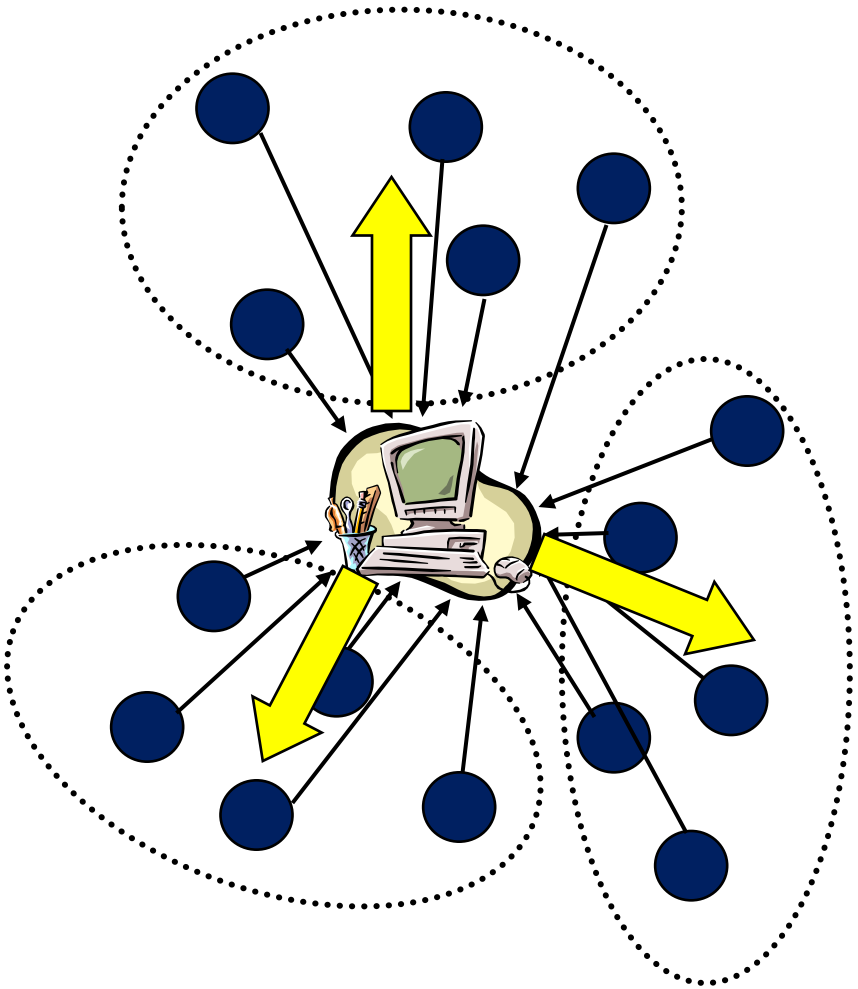
Challenges of Real World

- Theory says what is possible given the correlation.
- Codes exist which achieve bounds when correlation is known.
- How does one find the correlation?



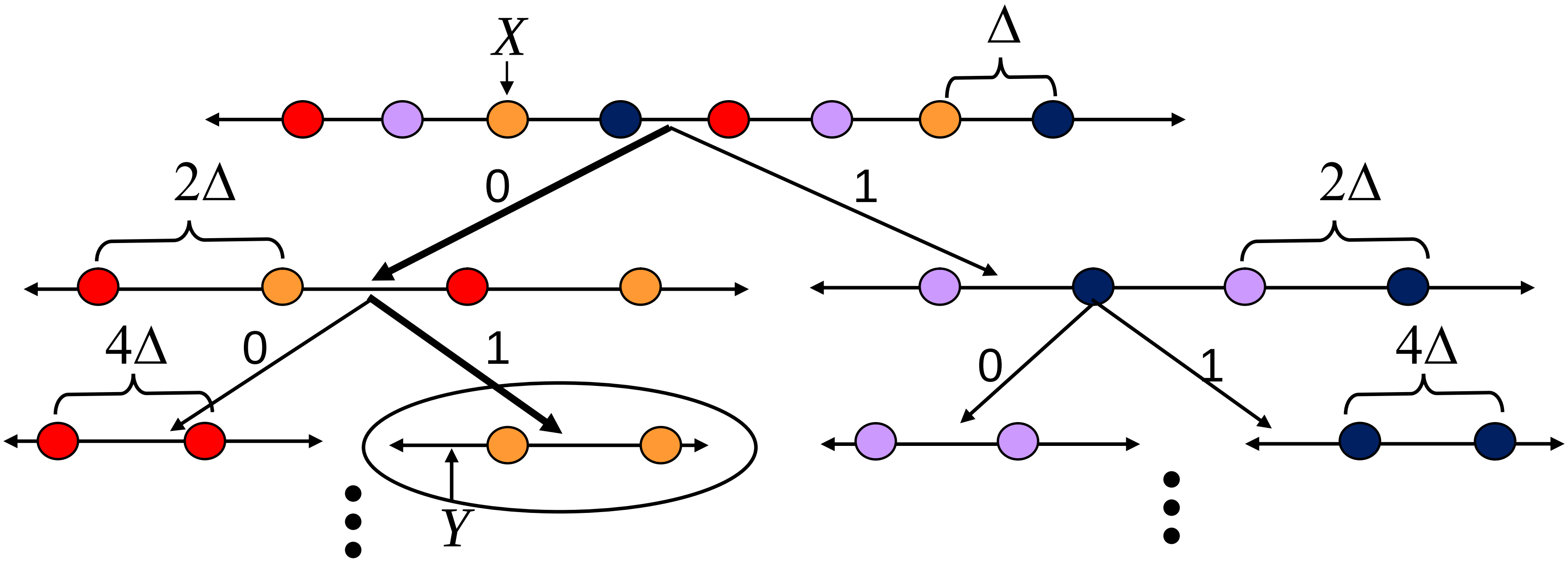
Setup

1. Controller receives uncoded data from sensors
2. Breaks them up into clusters s.t. nodes within cluster are highly correlated
3. Tells each cluster what code-book to use



Tree-Structured Code

- *Depth of tree specifies number of bits used for encoding*



- Path in the tree specifies the encoded value.
- Can tolerate $2^{i-1}\Delta$ of correlation noise using an i^{th} level codebook

How Much Compression?

- Sensor nodes measure X, data controller node has Y

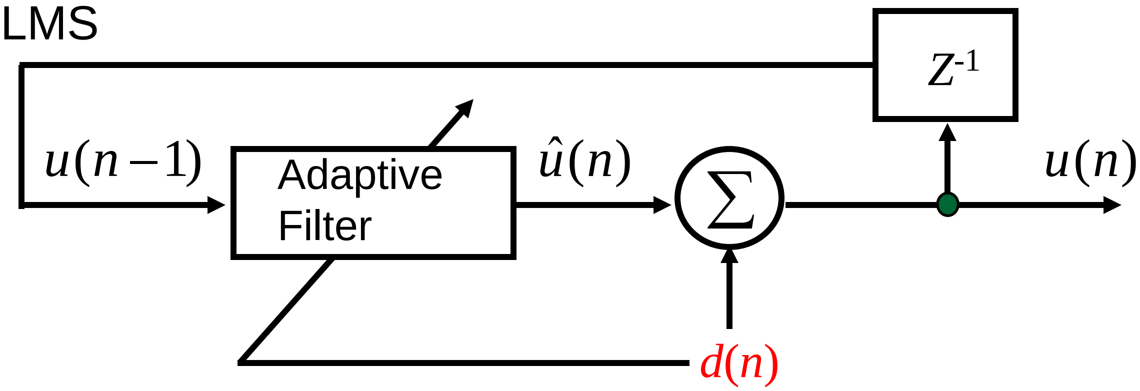
- Controller needs to estimate number of bits, i, it needs from sensor nodes for X.

- $X = Y + N$; N = correlation noise

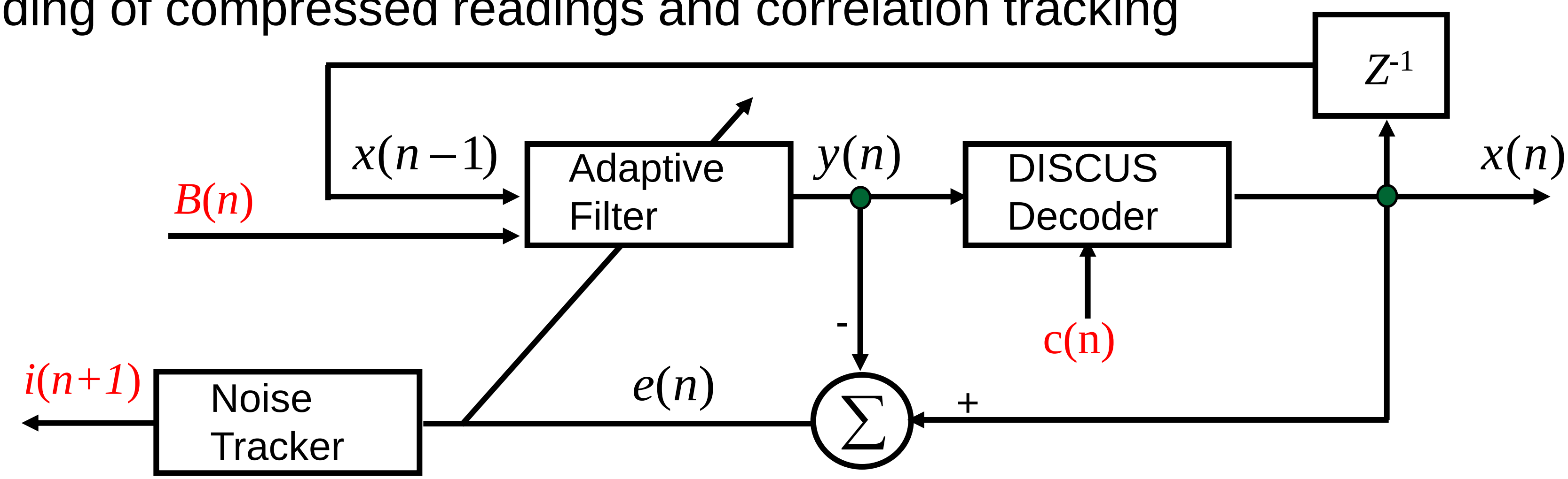
- $$P[|N| > 2^{i-1}\Delta] \leq \frac{\sigma_N^2}{\left(2^{i-1}\Delta\right)^2} \Rightarrow \boxed{i \geq \frac{1}{2}\log_2\left(\frac{\sigma_N^2}{\Delta^2 P_e}\right) + 1}$$

Decoding and Correlation Tracking

Standard LMS



Decoding of compressed readings and correlation tracking



$B(n)$ = decoded readings of all other sensors

$c(n)$ = coset index of $x(n)$, sent by encoder of x

$i(n+1)$ = number of bits to use in encoding x at time $n+1$, fed back to encoder of x

Adaptation Algorithm

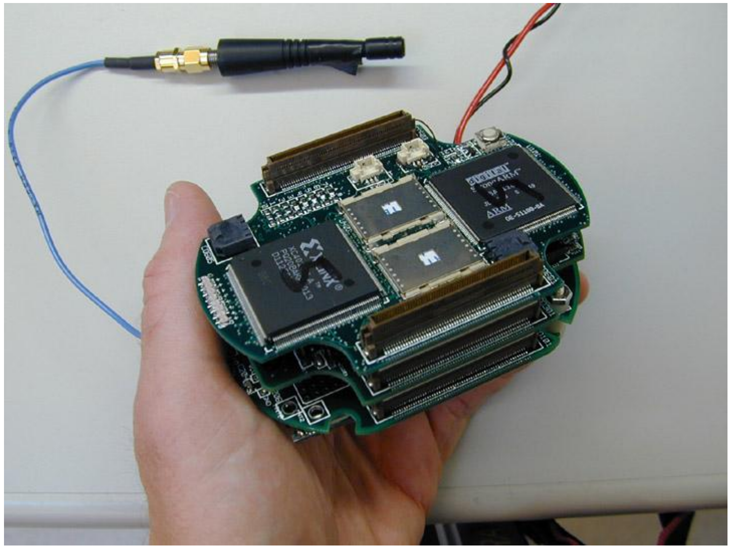
- $U(n)$ = Mx1 input at time n
- $y(n) = W(n)' * U(n)$
- Use DISCUS decoding to find $x(n)$
- $e(n) = x(n) - y(n)$
- $W(n+1) = W(n) + \mu * e(n) * u(n)$

Experimental Setup

- Collected data from PicoRadio test-bed nodes
- 5 light,

  5 temperature,

  5 humidity sensors
- Data was collected and used for testing real-time algorithms

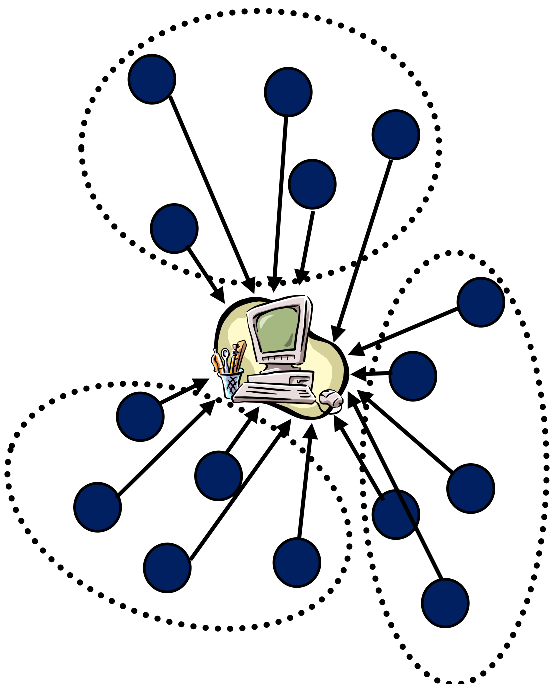


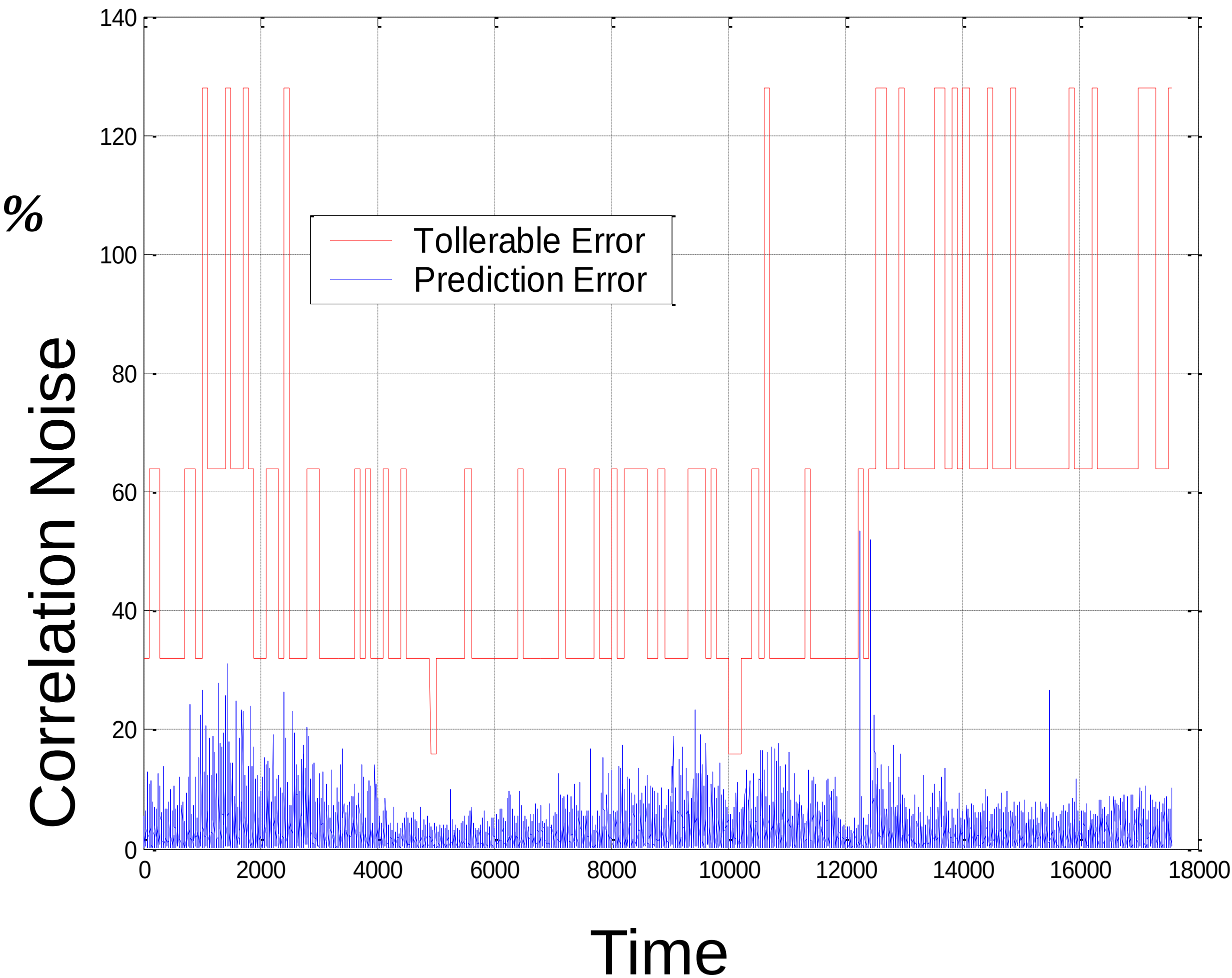
Simulations (correlation tracking)

•***Avg. Temp Savings = 66.6%***

•***Avg. Humidity Savings = 44.9%***

•***Avg. Light Savings = 11.7%***

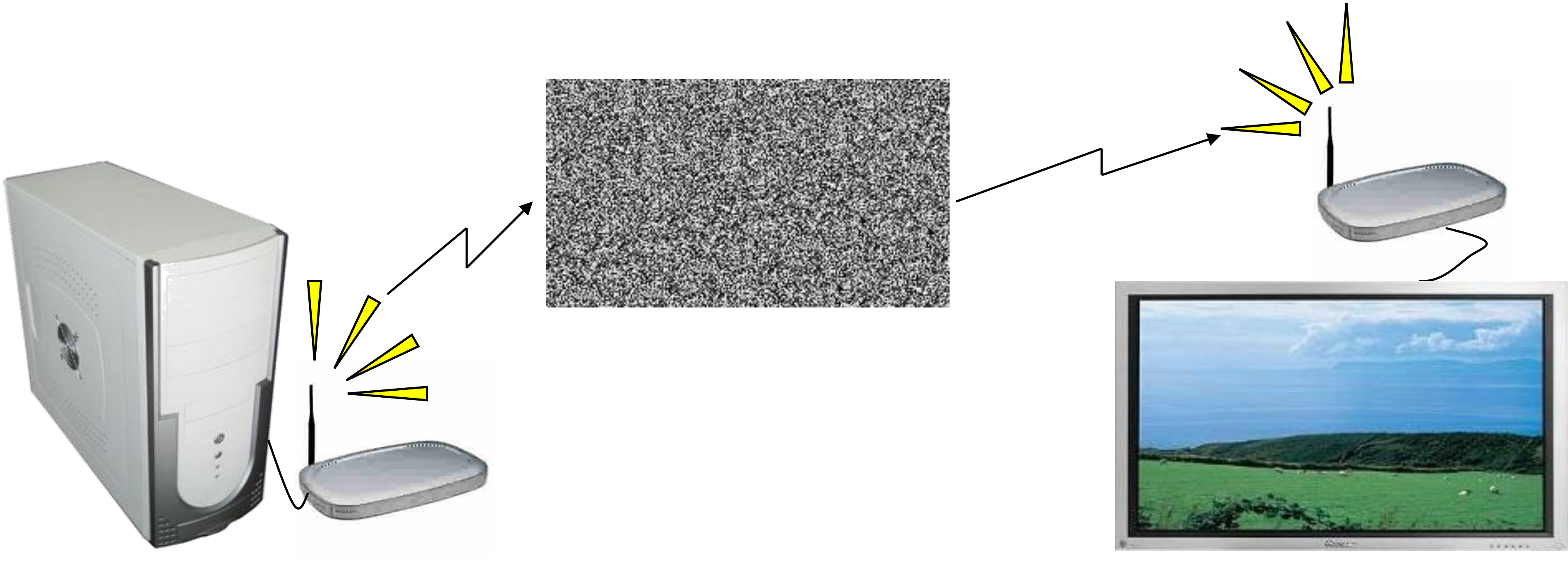




Compressing encrypted content without the cryptographic key

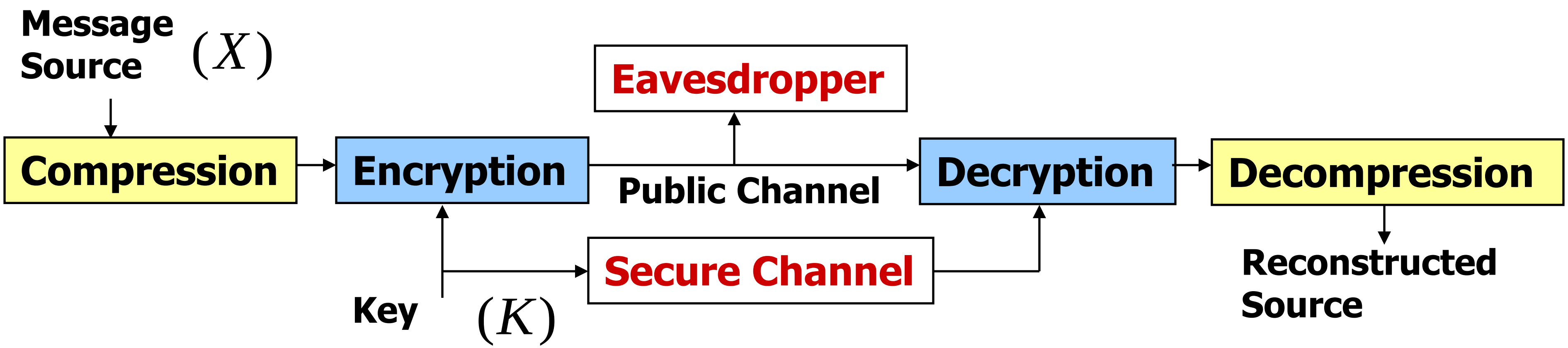
Secure multimedia for home networks

- Uncompressed encrypted video (HDCP protocol)
  - Can increase wireless range with lower data rate
  - But how to compress encrypted video **without access to crytpographic key**?

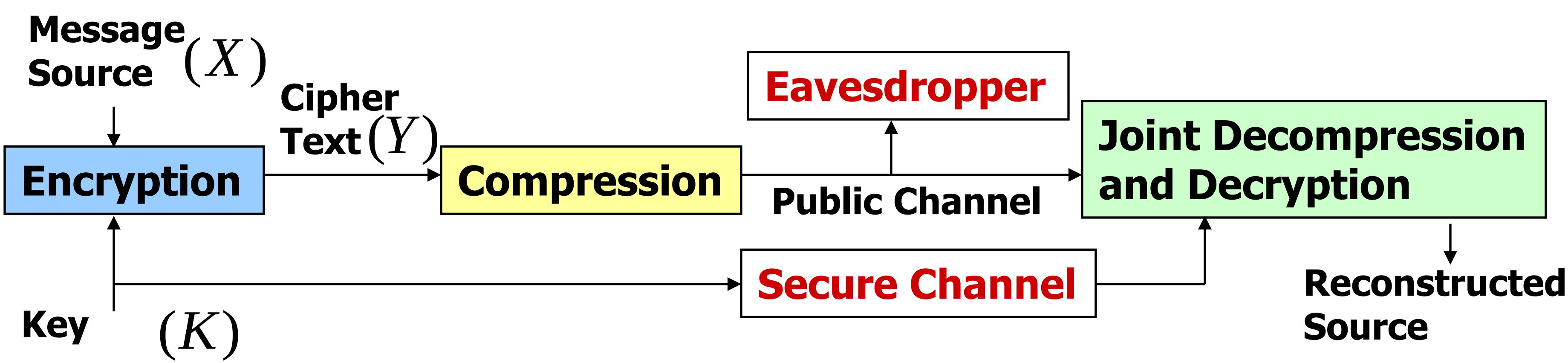


Application: Compressing Encrypted Data

Traditional/Best Practice:



Novel Structure:

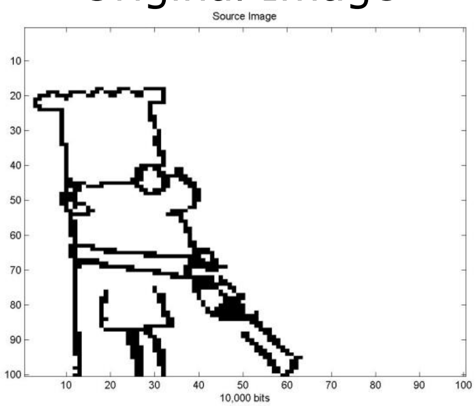


Johnson & Ramchandran (ICIP 2003),
Johnson et. al (Trans. on SP, Oct. 2004)

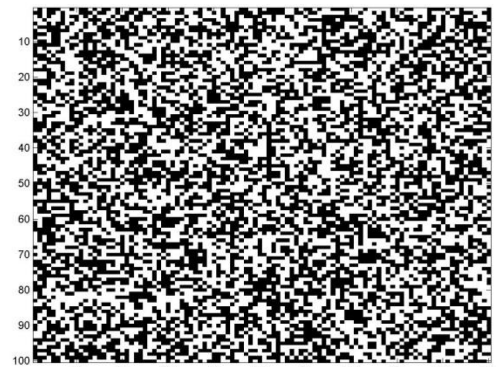
Example

10,000 bits

Original Image



Encrypted Image

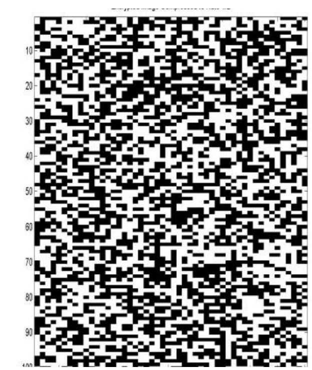


Compressed

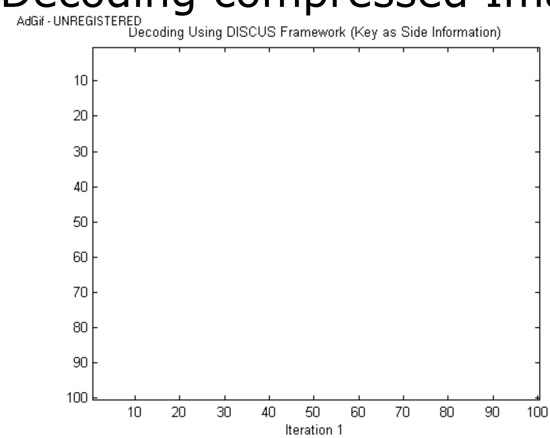
Encrypted Image

5,000

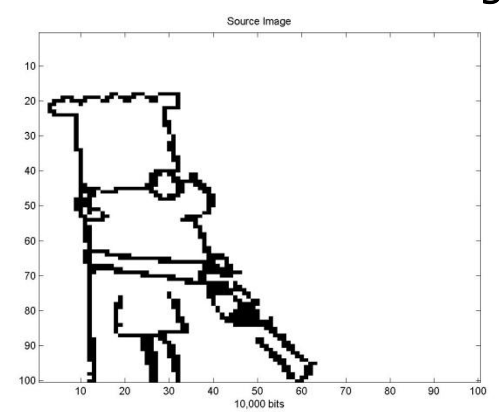
bits



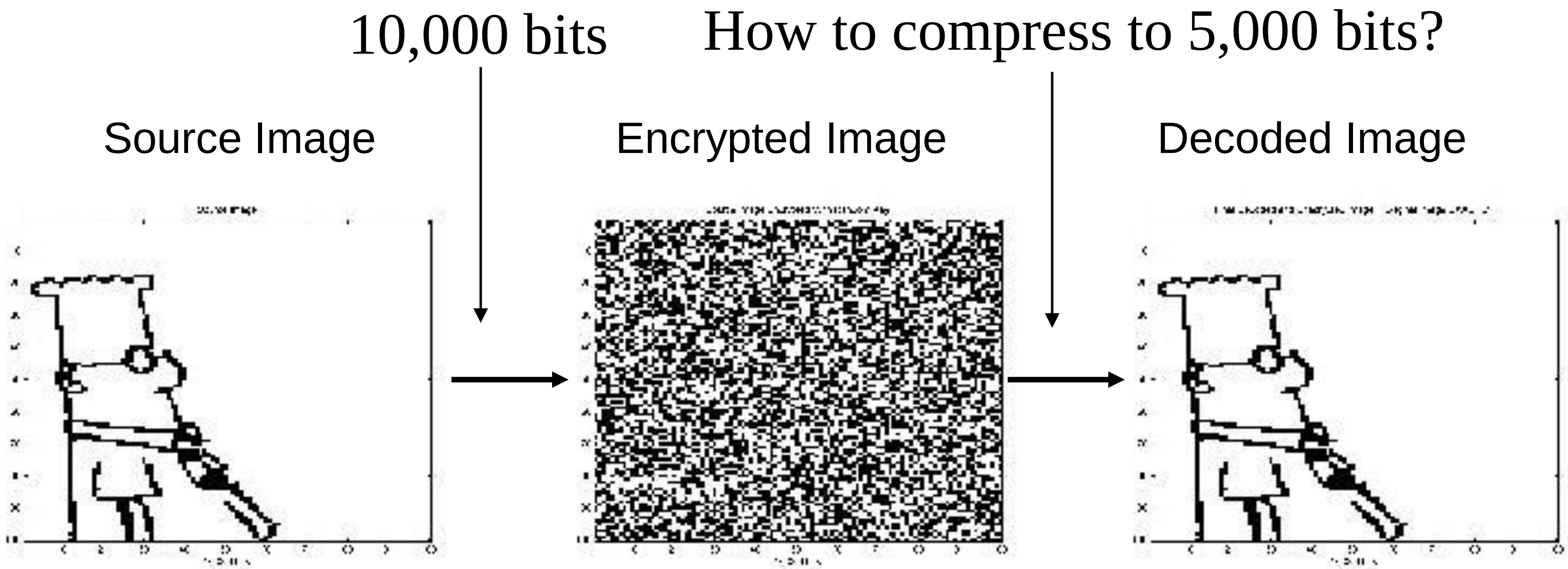
Decoding compressed Image



Final Reconstructed Image



Application: compressing encrypted data



Key Insight!

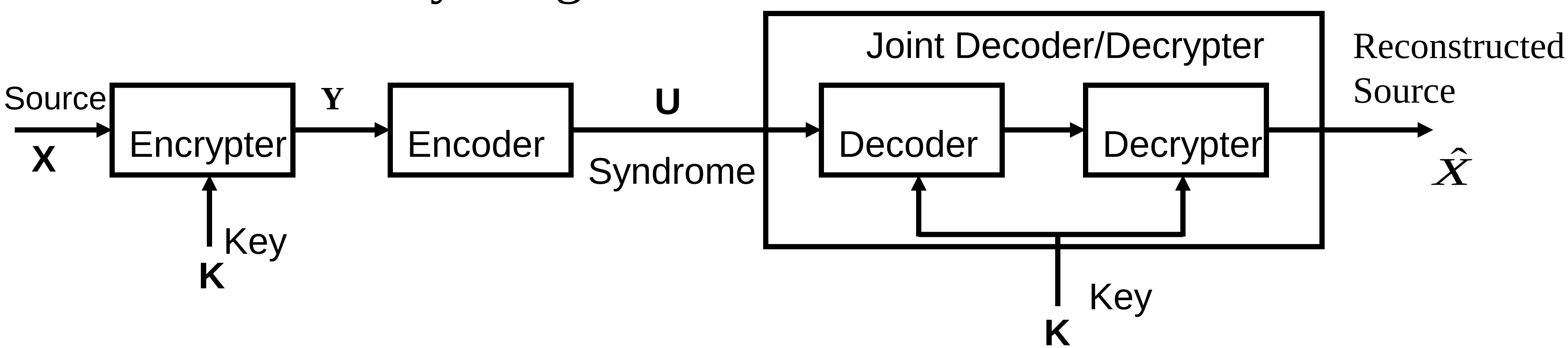
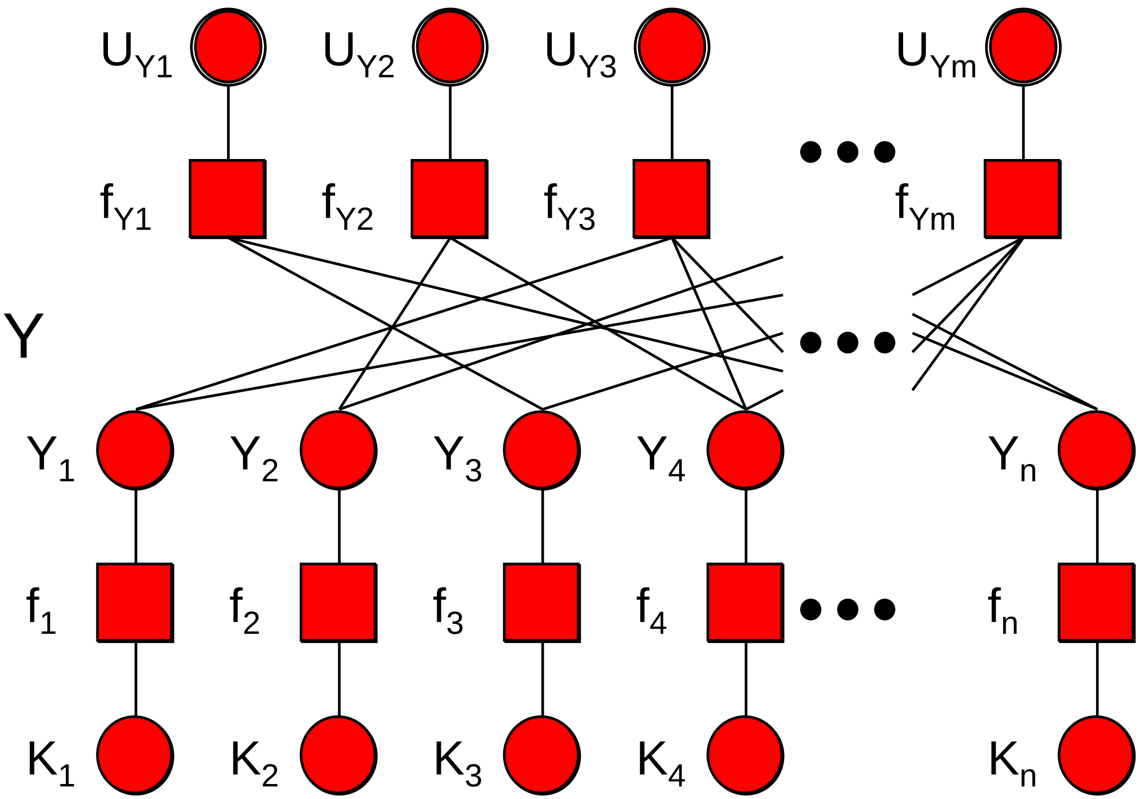
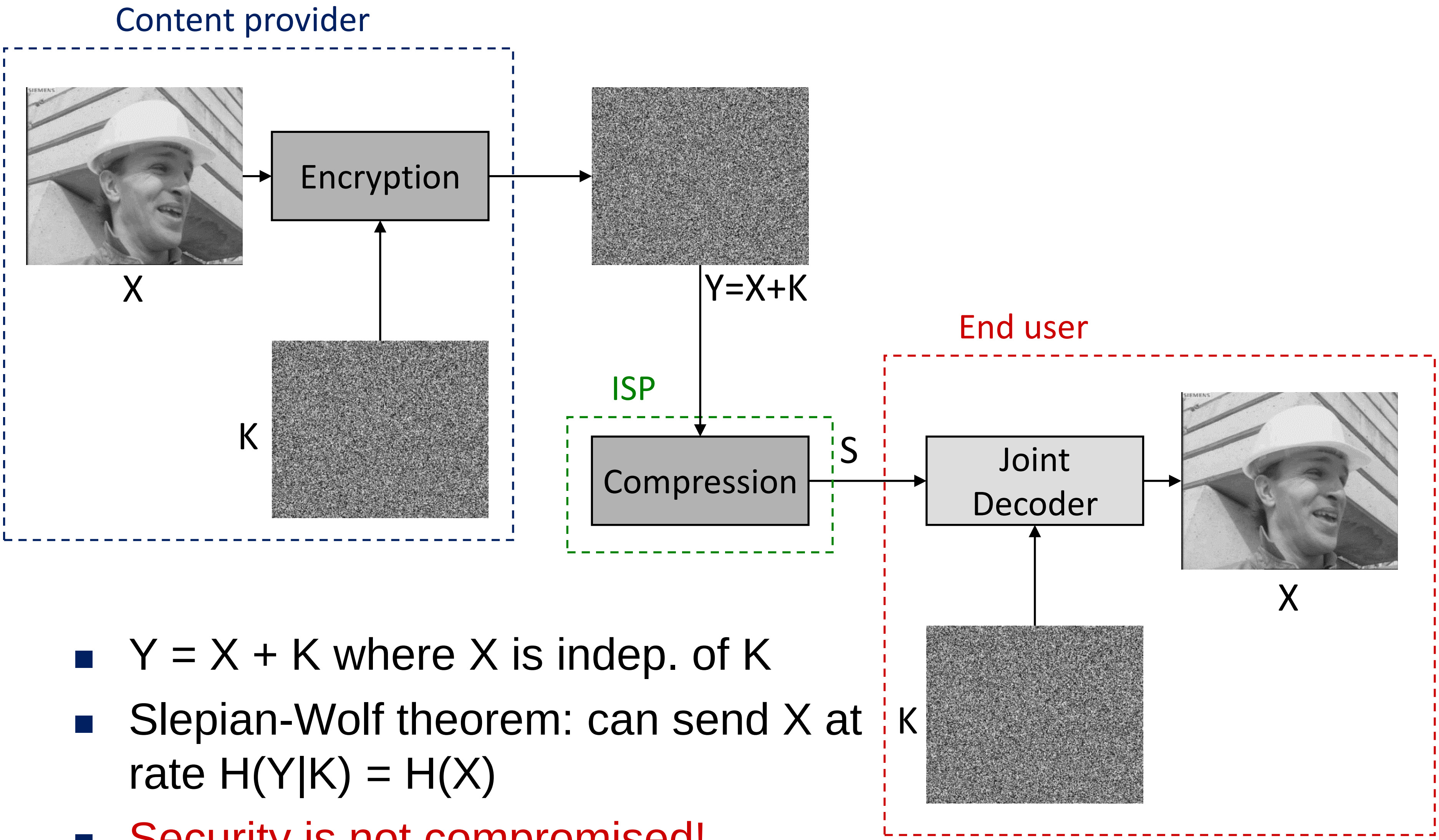


Illustration: coding in action

- Bits of Source Y
- Bits of Source K
- Y,K correlation
- LDPC code applied to Y
- Compressed bits of Y



Overview



- Y = X + K where X is indep. of K
- Slepian-Wolf theorem: can send X at rate H(Y|K) = H(X)
- Security is not compromised!

Johnson, Ishwar, Prabhakaran & KR (Trans. on SP, Oct. 2004)

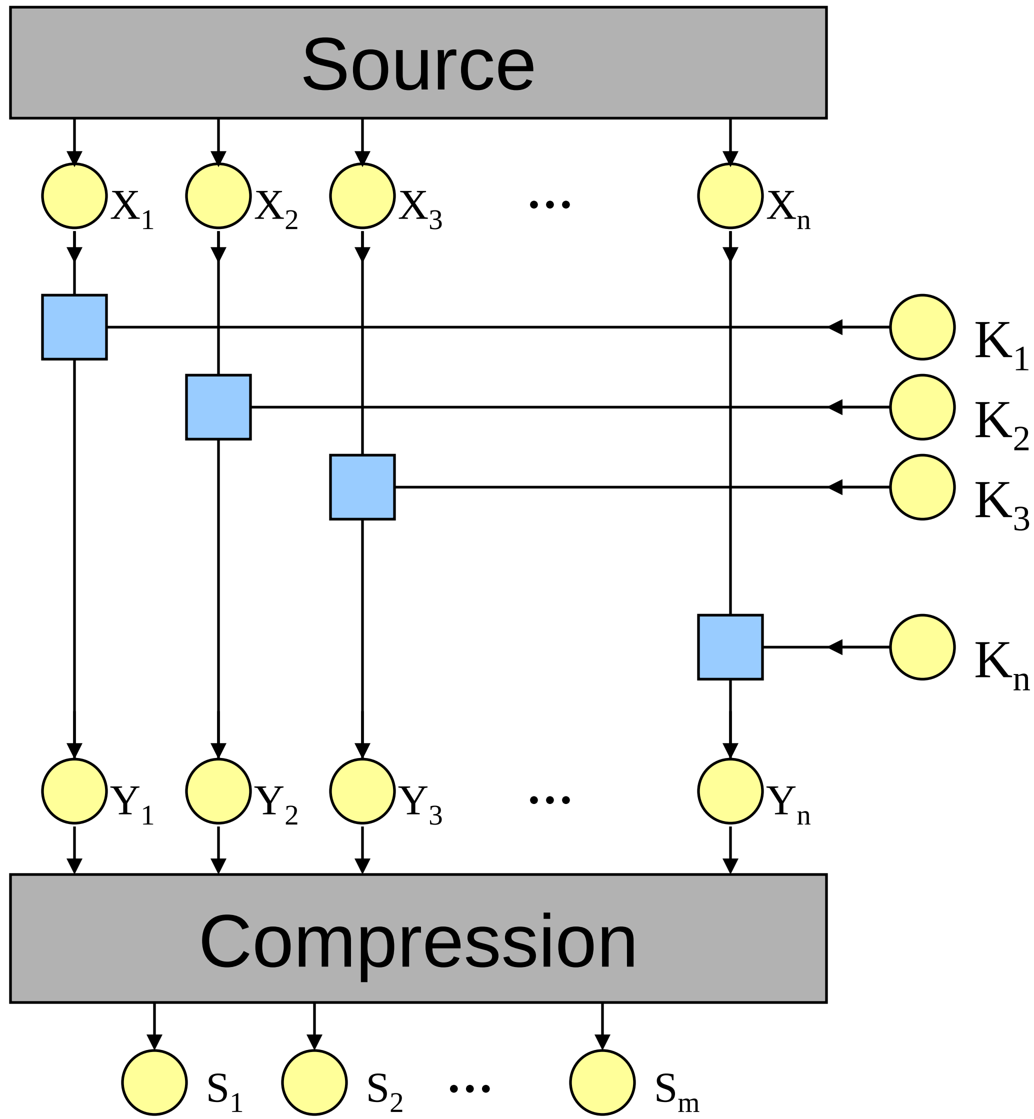
Framework: Encryption

Encryption:

- Stream cipher

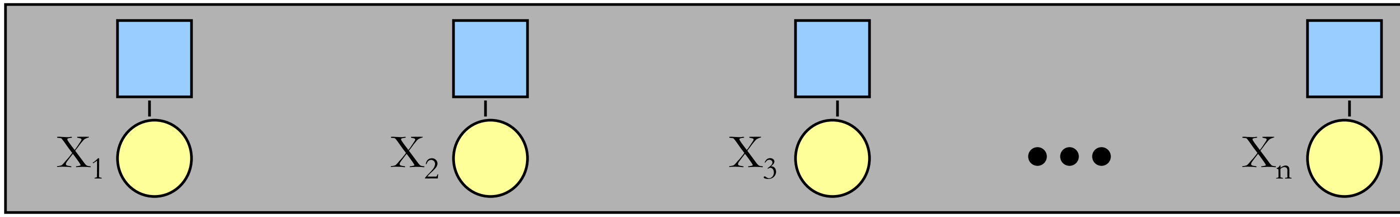
$$y_i = x_i \oplus k_i$$

- Graphical model captures exact encryption relationship

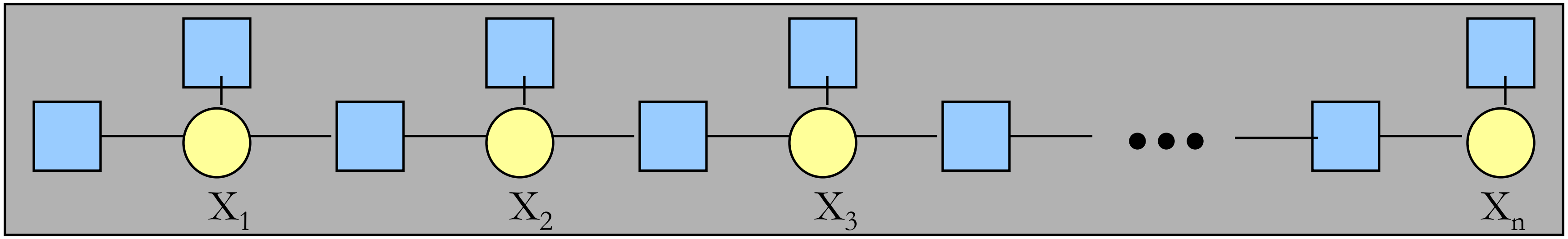


Source Models

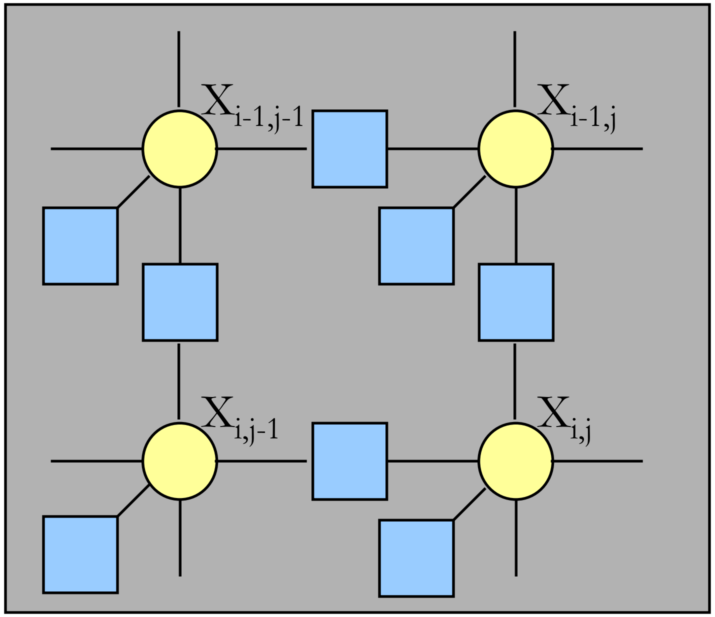
- IID Model



- 1-D Markov Model



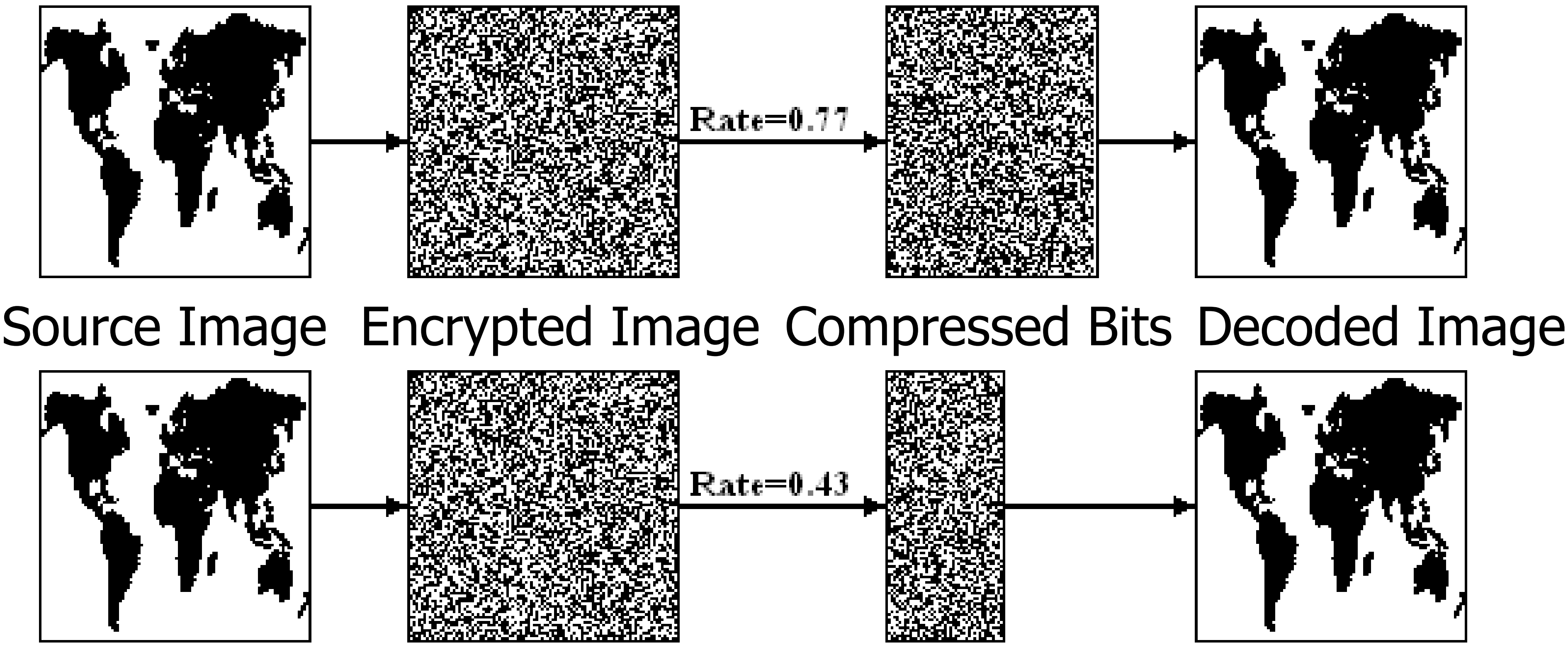
- 2-D Markov Model



Encrypted image compression results

- 100 x 100 pixel image (10,000 bits)
- No compression possible with IID model

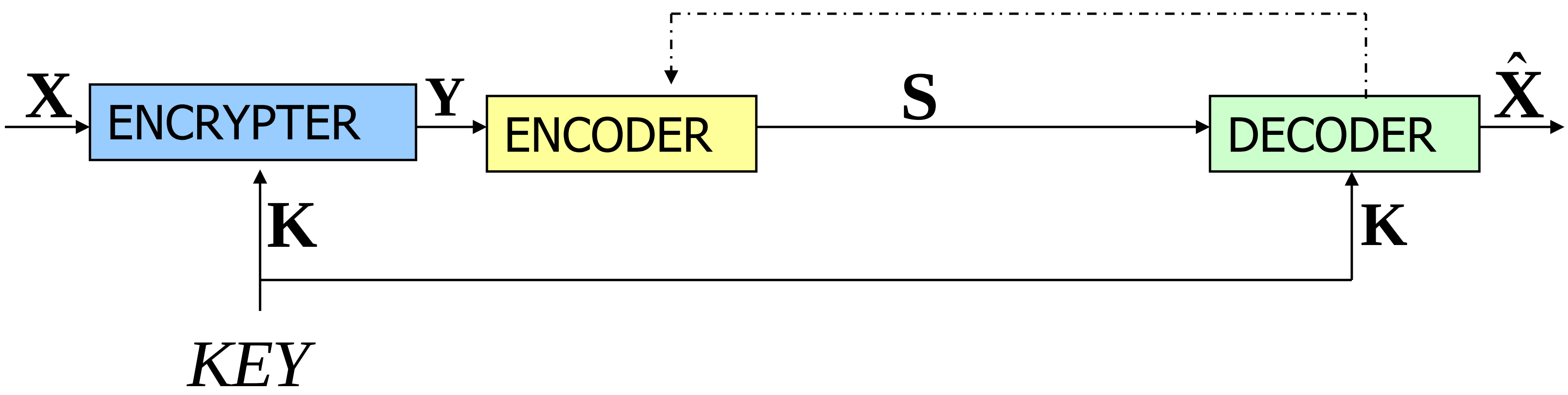
1-D Markov Source Model



2-D Markov Source Model

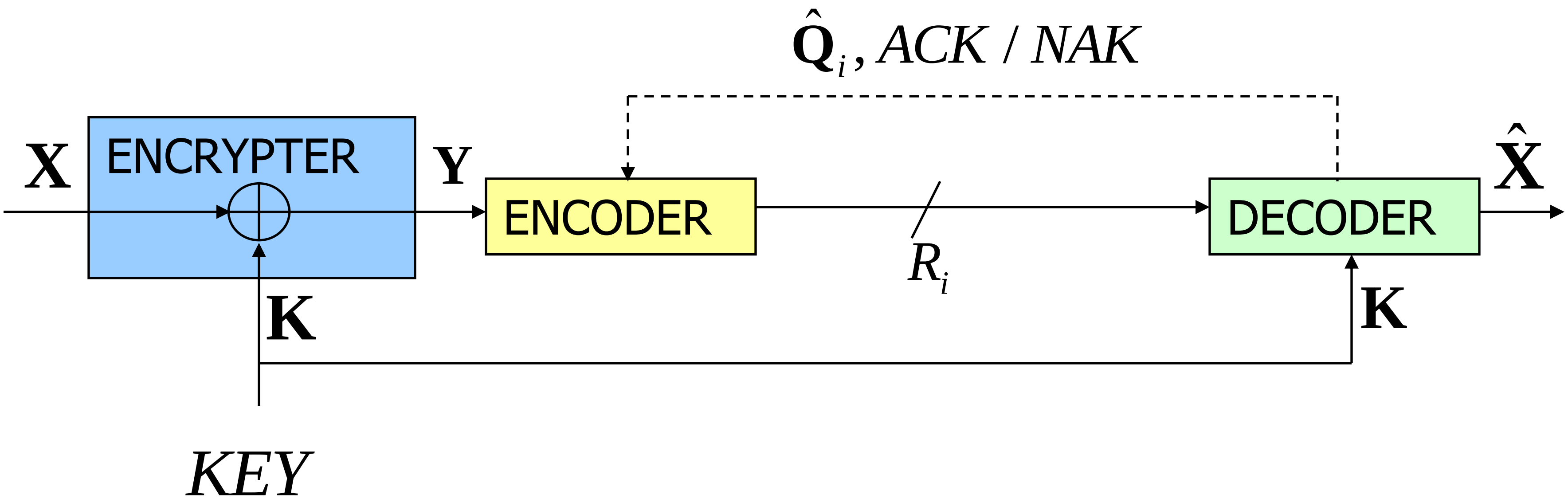
Key problems

- Data $\mathbf{X} \sim p(\mathbf{X})$
- When source statistics $p(\mathbf{X})$ are unknown
  - How to learn how much to compress?
  - How fast can limits be learned?
- When source statistics $p(\mathbf{X})$ are known
  - How to develop practical compression codes?
  - How well can they perform?

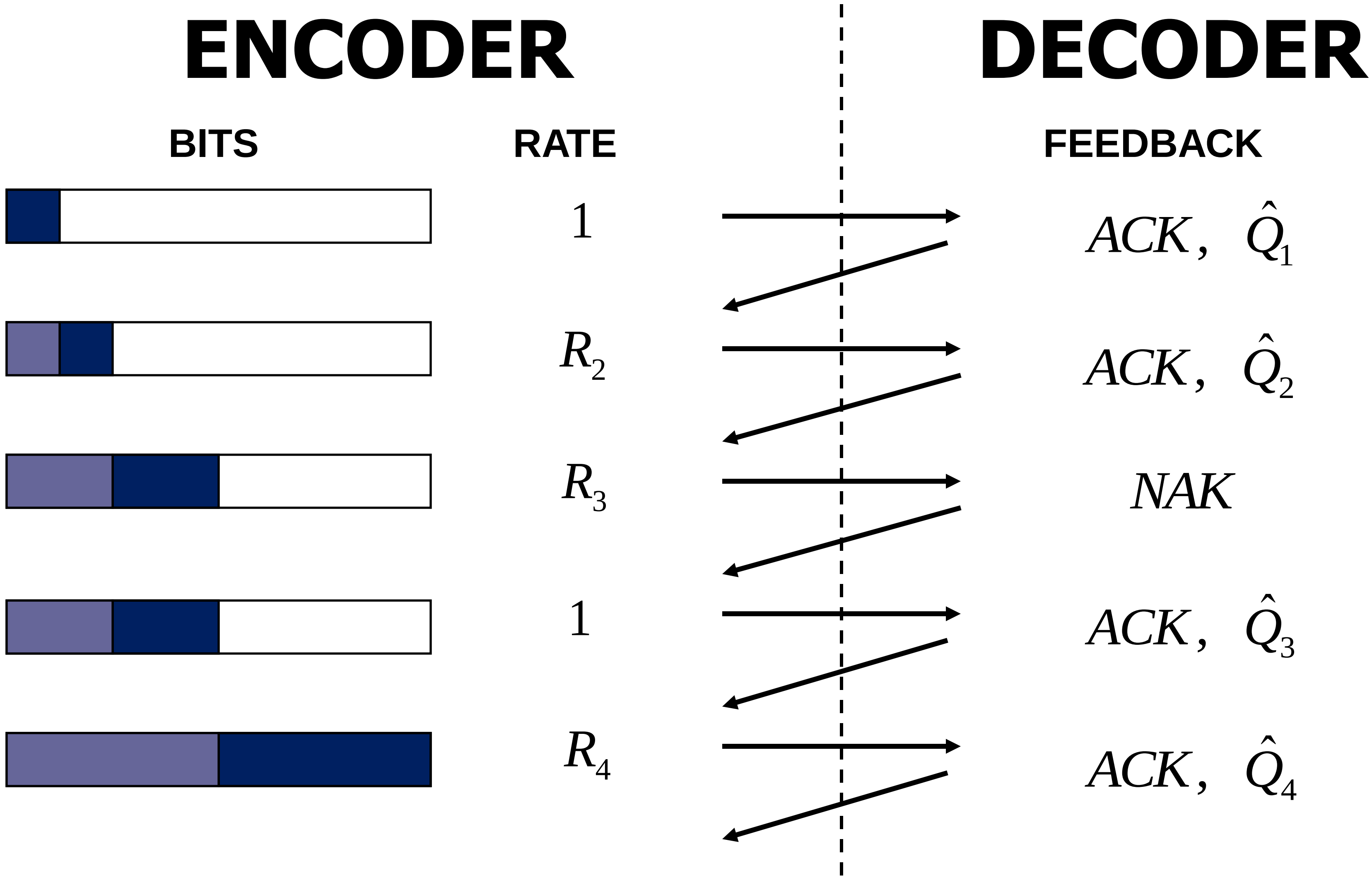


"Blind" compression protocol

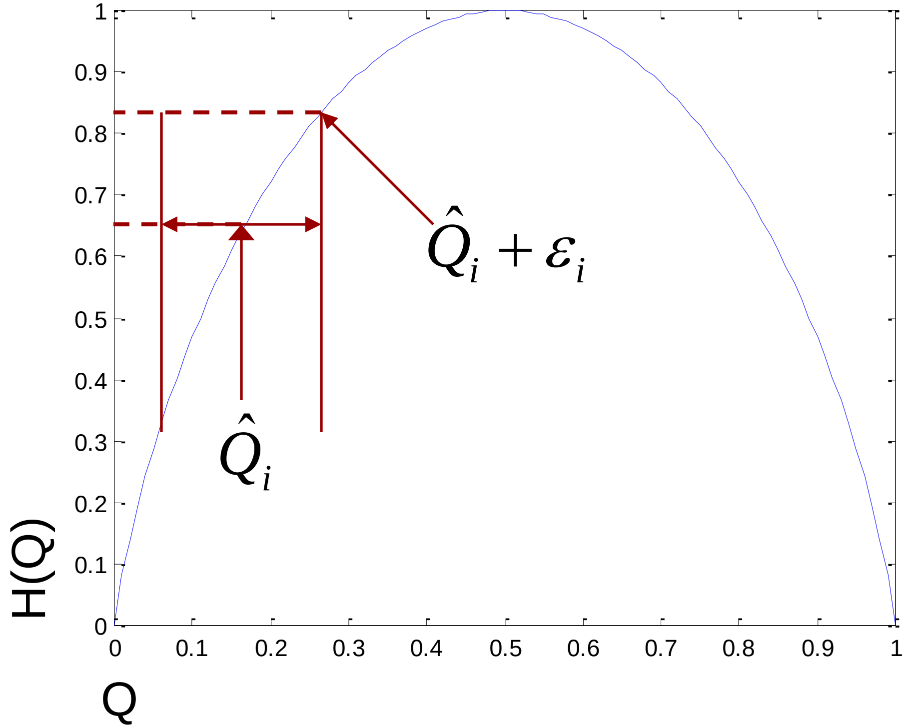
- For development: X is IID, $X \sim Bernoulli(Q)$
- Blocks indexed by i
- Encoder uses source estimate $\hat{Q}_i, \quad \hat{Q}_0 = 0.5$
- Compressible: $R = H(X) = H(Q)$



Sample run


ENCODER
DECODER
BITS
RATE
FEEDBACK
1
$ACK, \hat{Q}_1$
R_2
$ACK, \hat{Q}_2$
R_3
NAK
1
$ACK, \hat{Q}_3$
R_4
$ACK, \hat{Q}_4$

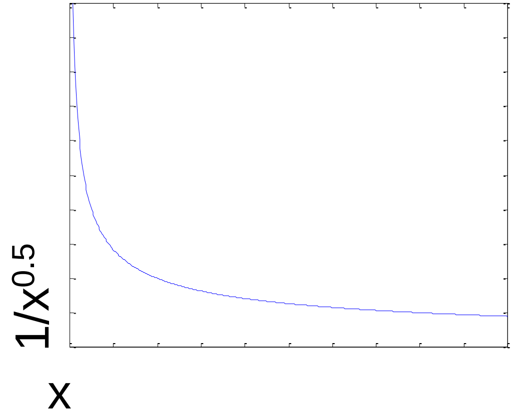
Rate selection



- Use ε redundancy to account for error in Q estimate

Redundancy ε_i

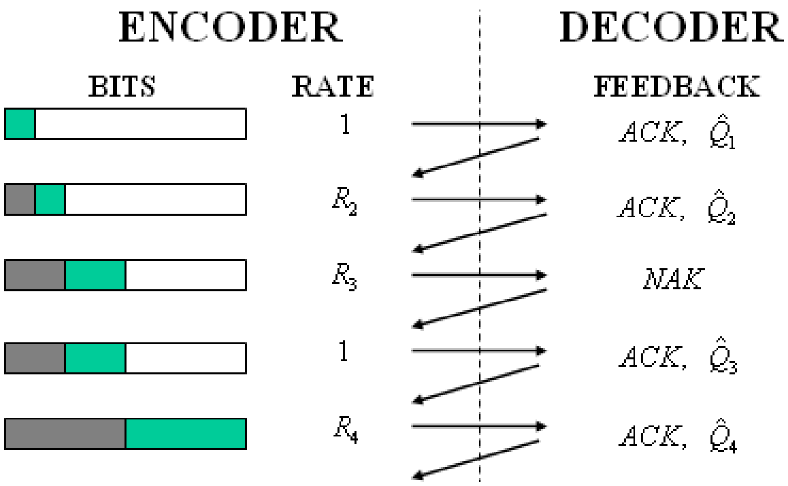
- Choose ε_i parameter to minimize redundancy.
  - Large $\rightarrow$ Low $\Pr(error)$, High R_i
  - Small $\rightarrow$ High $\Pr(error)$, Low R_i
- Define c_i as the transmission rate of block i.
- Minimize expected total rate: $E[c_i] = \Pr[e_i]\ln(2) + E[R_i]$
- Intuition – ε_i must go down $\sim 1/\sqrt{l}$
  - $$\hat{Q}_i = \sum_{j=1}^{n_i} x_j \Big/ n_i$$
  - $$E[\hat{Q}_i] = Q,\ \sigma_{\hat{Q}_i} = \sqrt{Q(1-Q)}\Big/\sqrt{n_i}$$



Analysis assumptions and bound

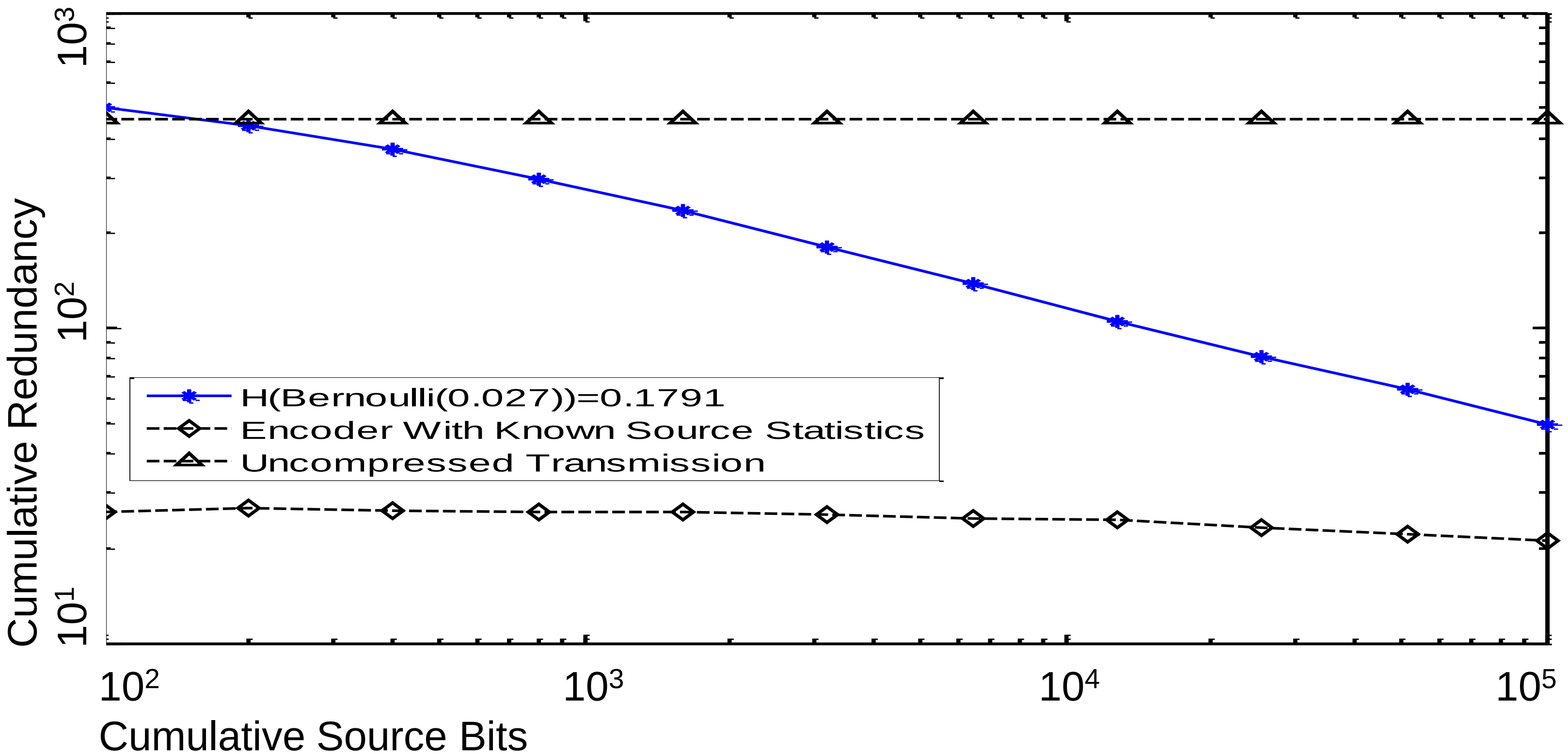
- Decoding errors are detected
- Decoding errors occur when empirical entropy exceed code rate (perfect codes)

$$H\left(x_{n_{i-1}+1}^{n_i}\right) > l_i R_i$$



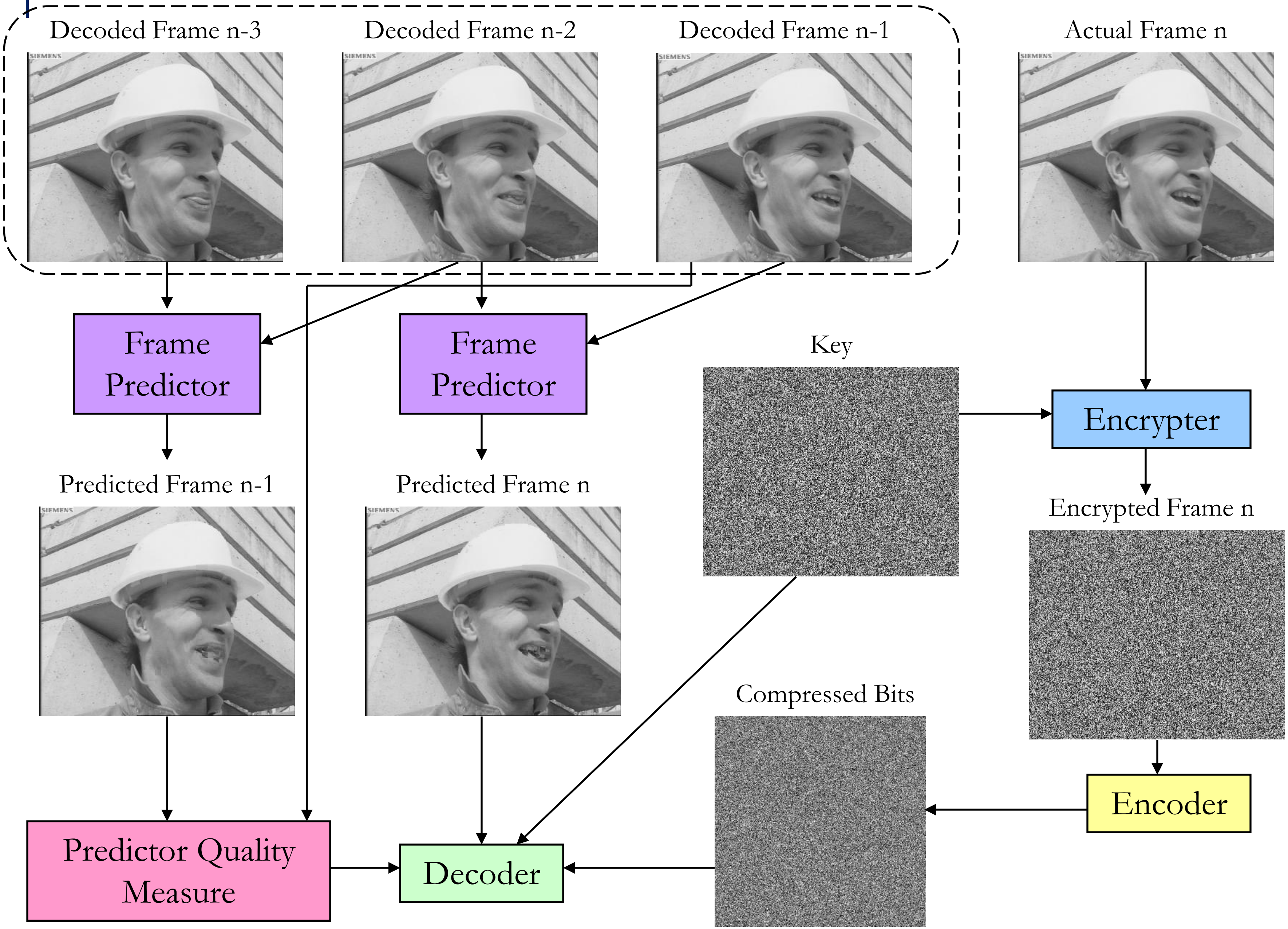
- Resulting Bound Form: $E[c_i] \le H(Q) + C_1\varepsilon_i + C_2 \exp\left\{-C_3 l_i \varepsilon_i^2\right\}$
  - Linear term → Make ε small
  - Exponential term → ε decays slower than $1/\sqrt{l_i}$

Results – Memoryless source



- $H(Q_1=.027)=0.17912$
- Uncompressed transmission (top line)
- With knowledge of source statistic Q (bottom line)

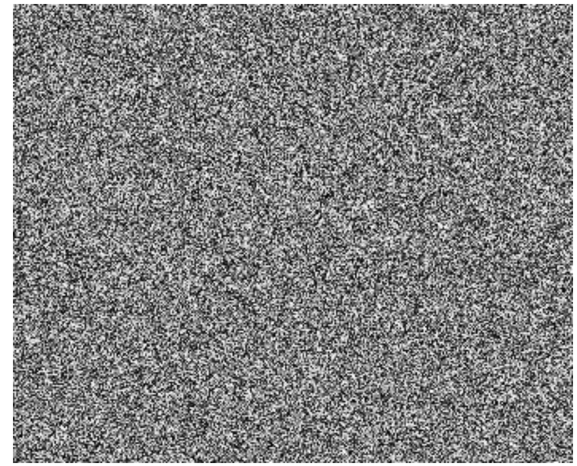
Temporal Prediction Scheme



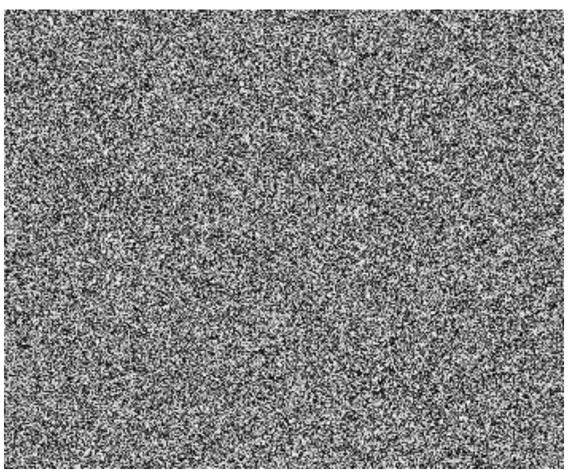
Compression of encrypted video

- Video offers both temporal and spatial prediction
  - Decoder has access to unencrypted prior frames

$\oplus$



=



	Blind approach (encoder has no access to key)
Foreman	**Saves 33.00%**
Garden	**Saves 17.64%**
Football	**Saves 7.17%**

Schonberg, Yeo, Draper & Ramchandran, DCC '07

Distributed Video Compression

Active research area

- Puri and KR: Allerton'02, ICASSP'03, ICIP'03.
- Aaron, Zhang and Girod: Asilomar'02
- Aaron, Rane, Zhang and Girod: DCC'03
- Aaron, Setton and Girod: ICIP'03
- Sehgal, Jagmohan and Ahuja: DCC'03, ICIP'03.
- Wang, Majumdar and KR: ACM MM'04
- Yaman and AlRegib: ICASSP'04
- Xu & Xiong: VCIP '04
- Wang & Ortega: ICIP '04

First side-information-coding based video coding idea was however in 1978!!

Application scope

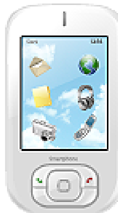
- Motivation: Uncertainty in the side information
  - Low complexity encoding
  - Transmission packet drops
  - Multicast & scalable video coding
  - Flexible decoding

- Physically distributed sources
  - Multi-camera setups

- Other interesting applications?

Low complexity encoding

- Motivation



Low-Complexity Encoder

Trans-coding proxy: High-Complexity Decoder → High-Complexity Encoder

Low-Complexity Decoder

current frame

DSC Encoder

Low-complexity
- No motion search
- Various channel codes

reference frame

S-I Generator

DSC Decoder

current frame

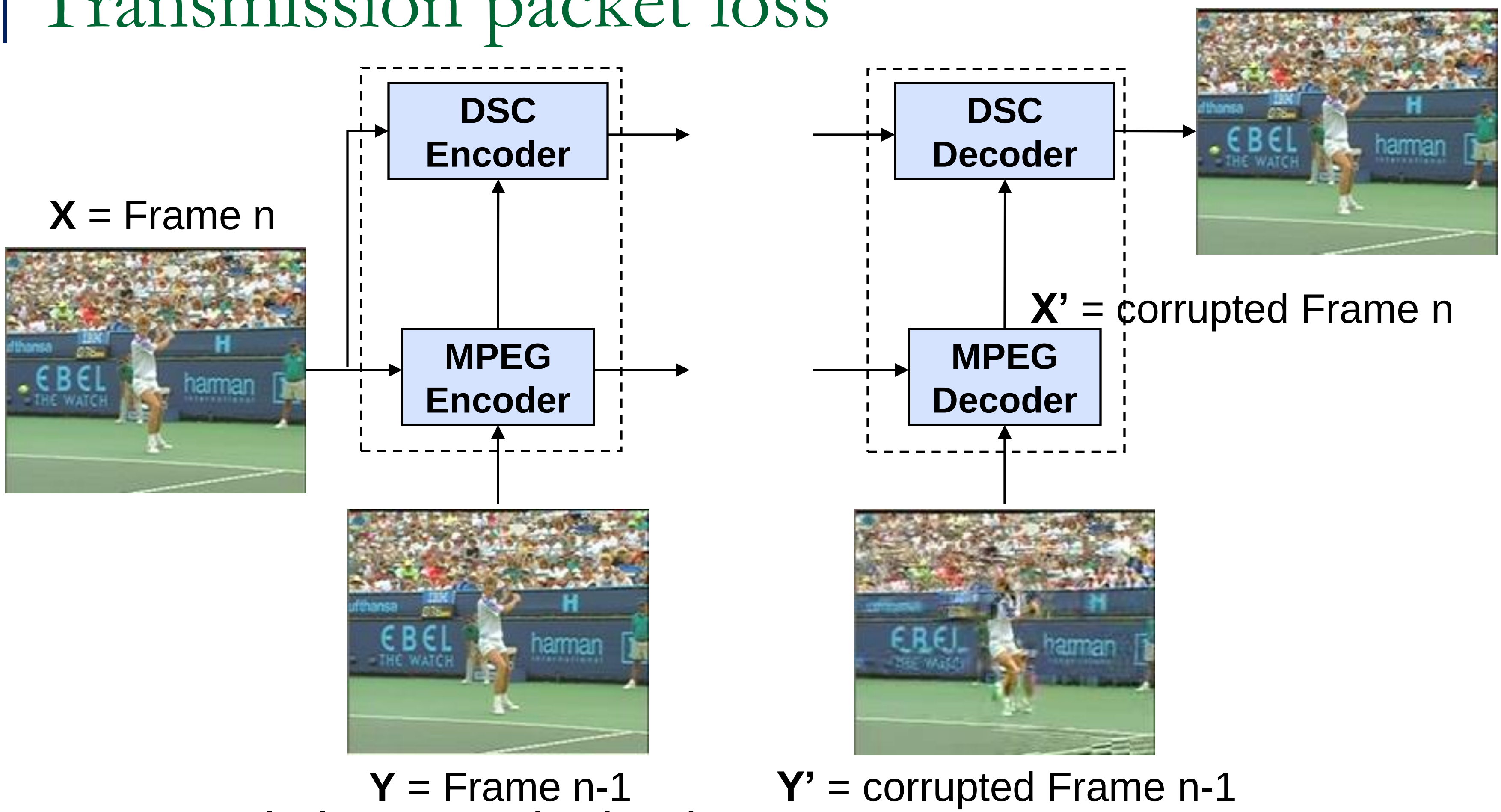
High-complexity
- Block or frame-level
- Interpolated or compensated motion

Puri & Ramchandran, Allerton 2002

Aaron, Zhang & Girod, Asilomar 2002

Artigas, Ascenso, Dalai, Klomp, Kubasov & Ouaret, PCS 2007

Transmission packet loss



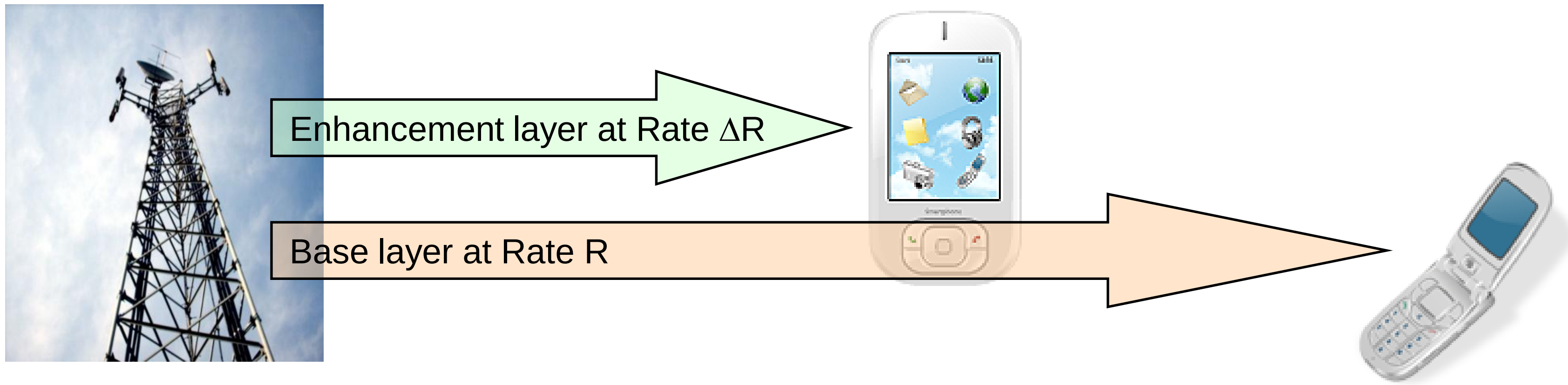
- FEC solutions may be inadequate
- Can be made compatible with existing codec
- Corrupted current frame is S-I at DSC robustness decoder

A. Aaron, S. Rane, D. Rebollo-Monedero & B. Girod: DCC’03, ICIP’04, ICIP’05

A. *Sehgal, A. Jagmohan & N. Ahuja: Trans’04*

B. *J. Wang, A. Majumdar, K. Ramchandran & H. Garudadri: PCS’04, ICASSP’05*

Multicast & scalable video coding

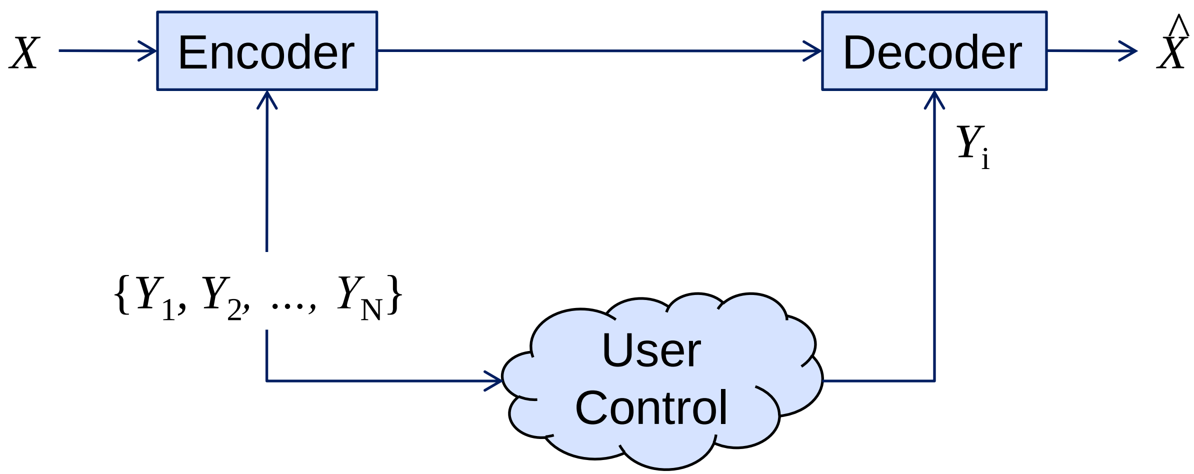


- Multicast
  - Accommodate heterogeneous users
    - Different channel conditions
    - Different video qualities (spatial, temporal, PSNR)

Majumdar & Ramchandran, ICIP 2004
Tagliasacchi, Majumdar & Ramchandran, PCS 2004
Sehgal, Jagmohan & Ahuja, PCS 2004
Wang, Cheung & Ortega, EURASIP 2006
Xu & Xiong, Trans. Image Processing 2006

Flexible decoding

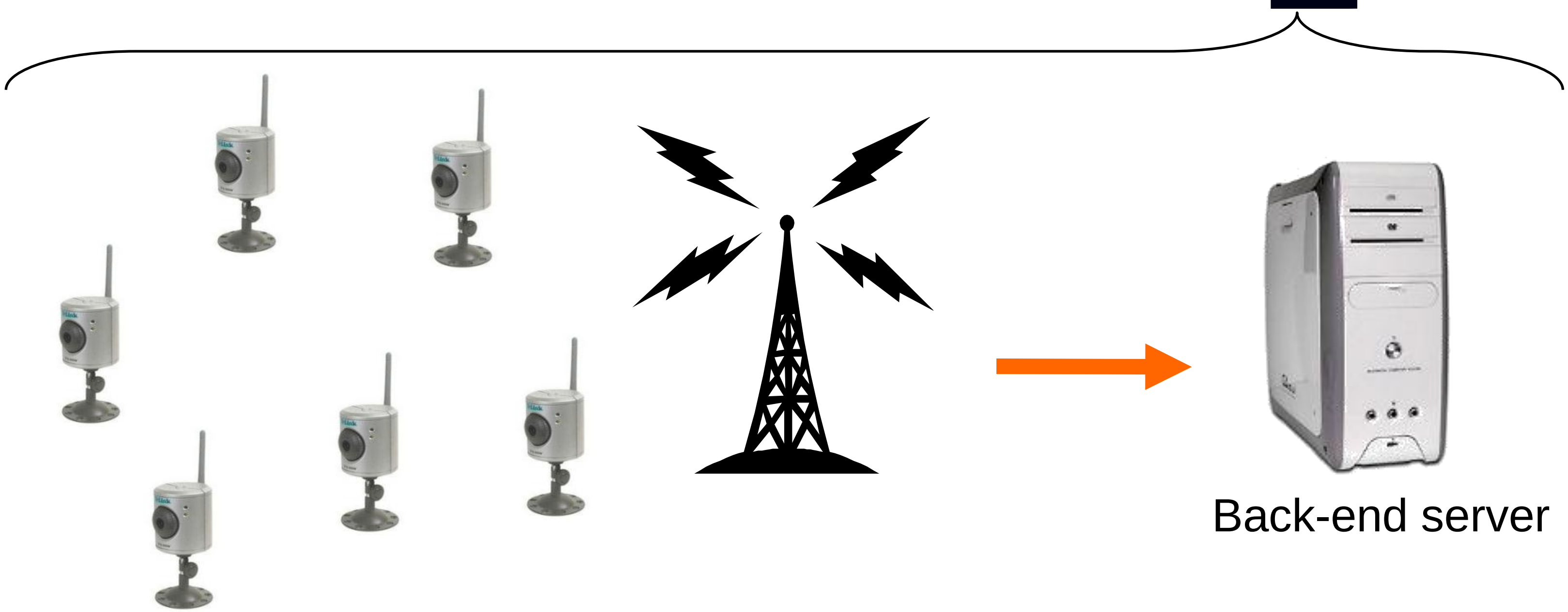
- $\{Y_1, Y_2, \ldots, Y_N\}$ could be
  - Neighboring frames in time
    → forward/backward playback without buffering
  - Neighboring frames in space
    → random access to frame in multi-view setup
  - …



Cheung, Wang & Ortega, VCIP 2006, PCS 2007
Draper & Martinian, ISIT 2007

Multi-camera setups

- Dense placement of low-end video sensors
- Sophisticated back-end processing
  - 3-D view reconstruction
  - Object tracking
  - Super-resolution
- Multi-view coding and transmission

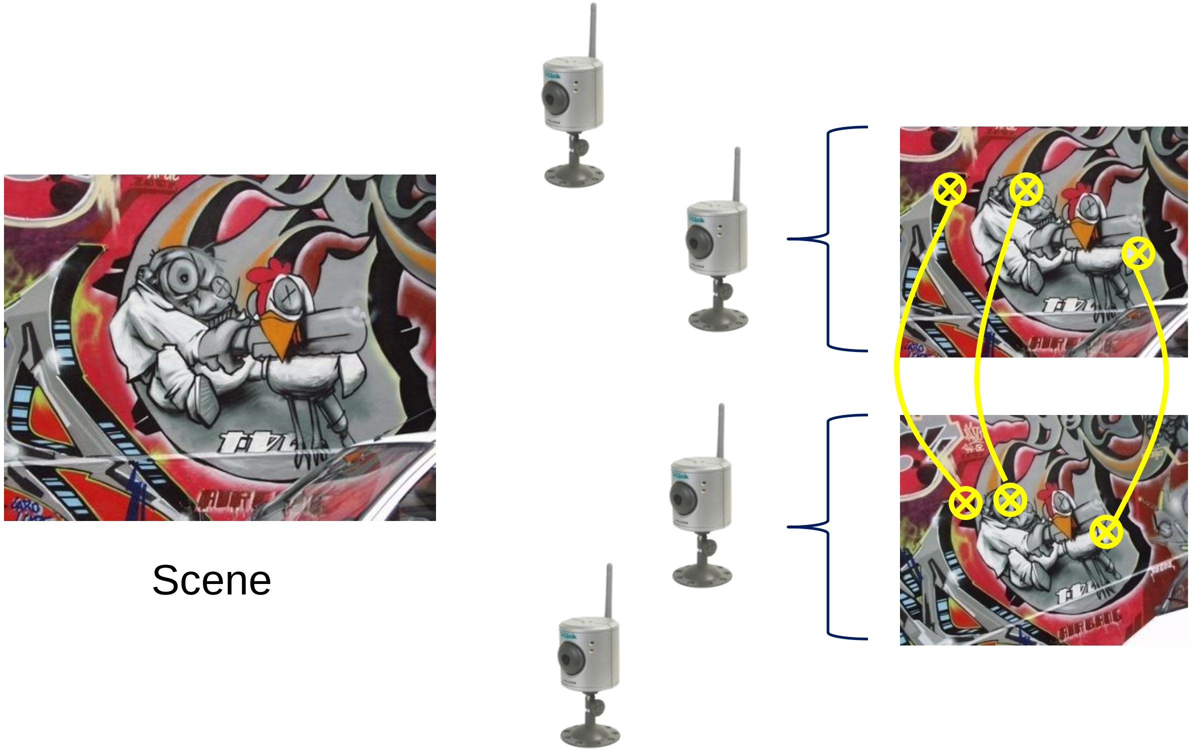


Other applications?

- Rate-efficient camera calibration
  - Visual correspondence determination

Tosic & Frossard, EUSIPCO 2007

Yeo, Ahammad & Ramchandran, VCIP 2008

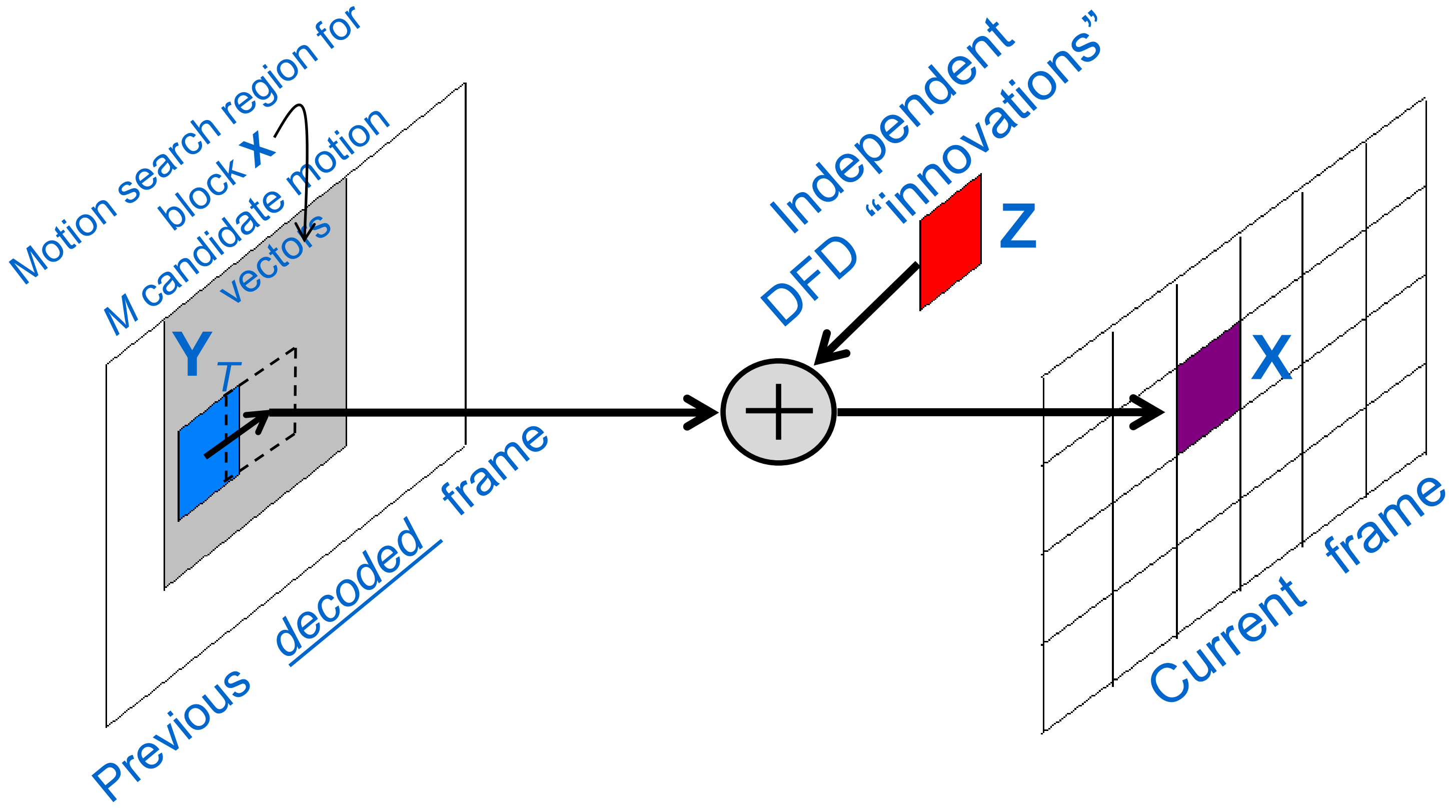


Scene

PRISM: DSC based video compression

- Motivation:
  - Low encoding complexity
  - Robustness under low latency

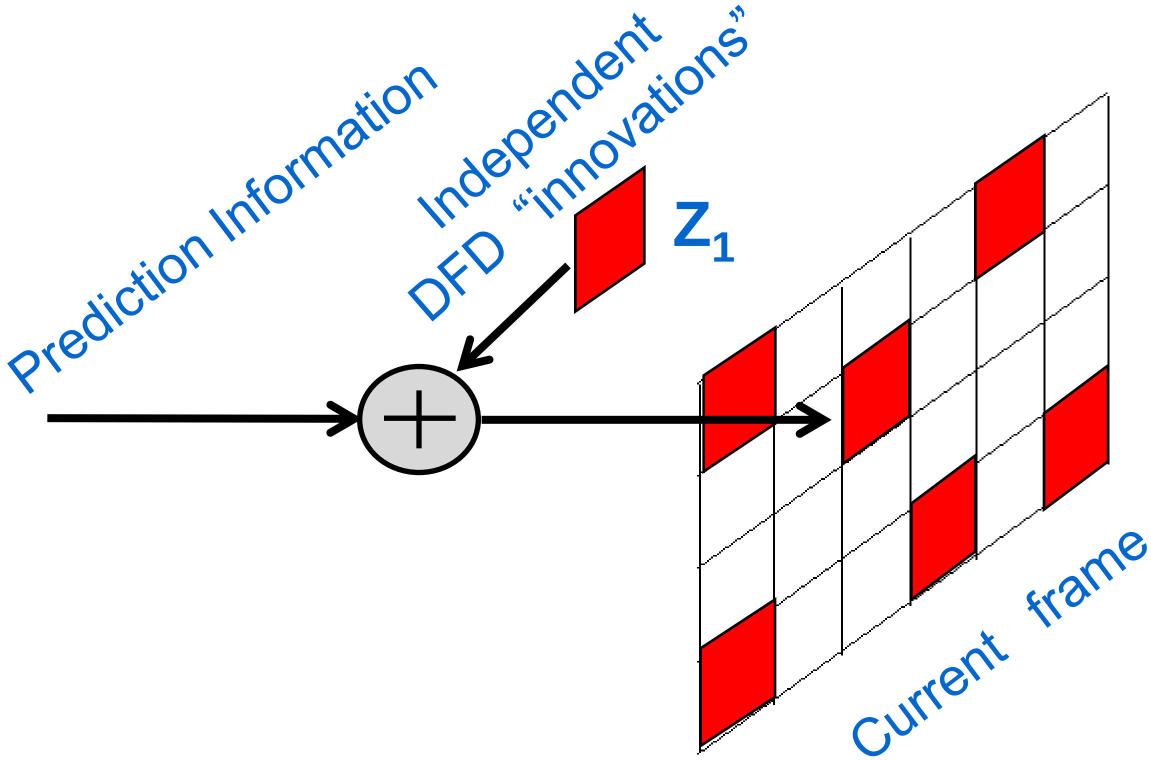
A closer look at temporal aspects of video



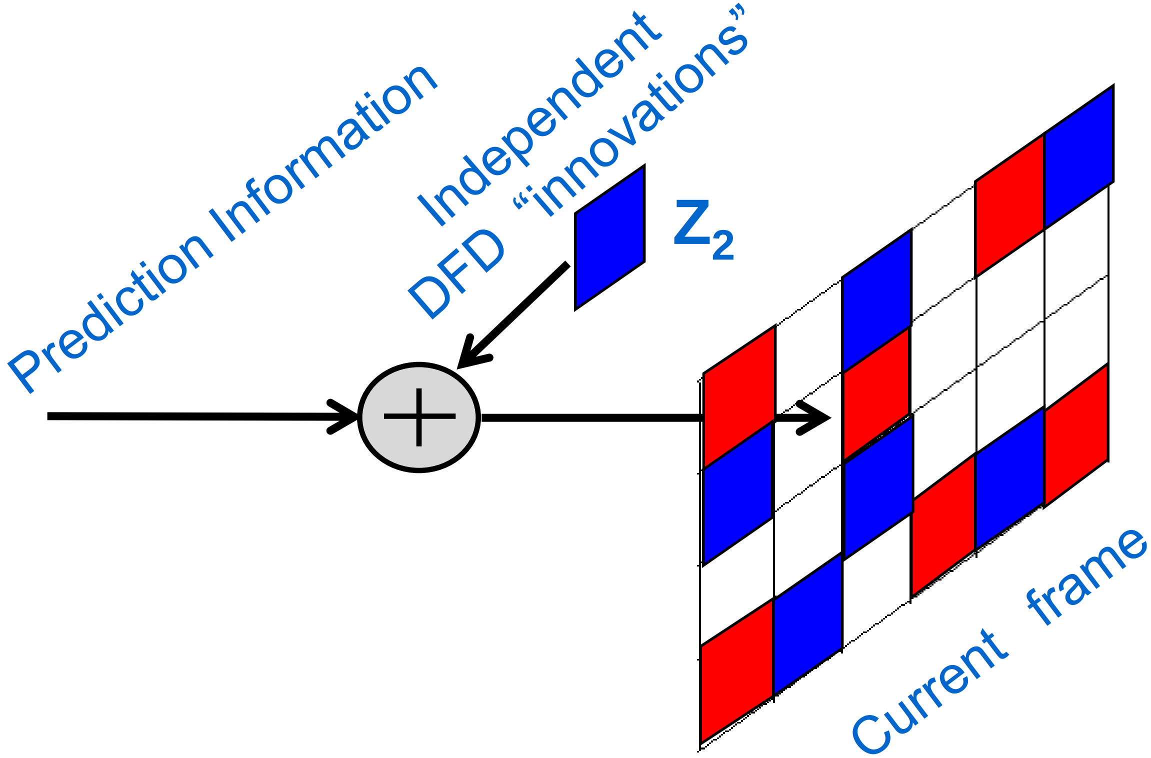
- Motion is a "local" phenomenon

Block-motion estimation is key to success

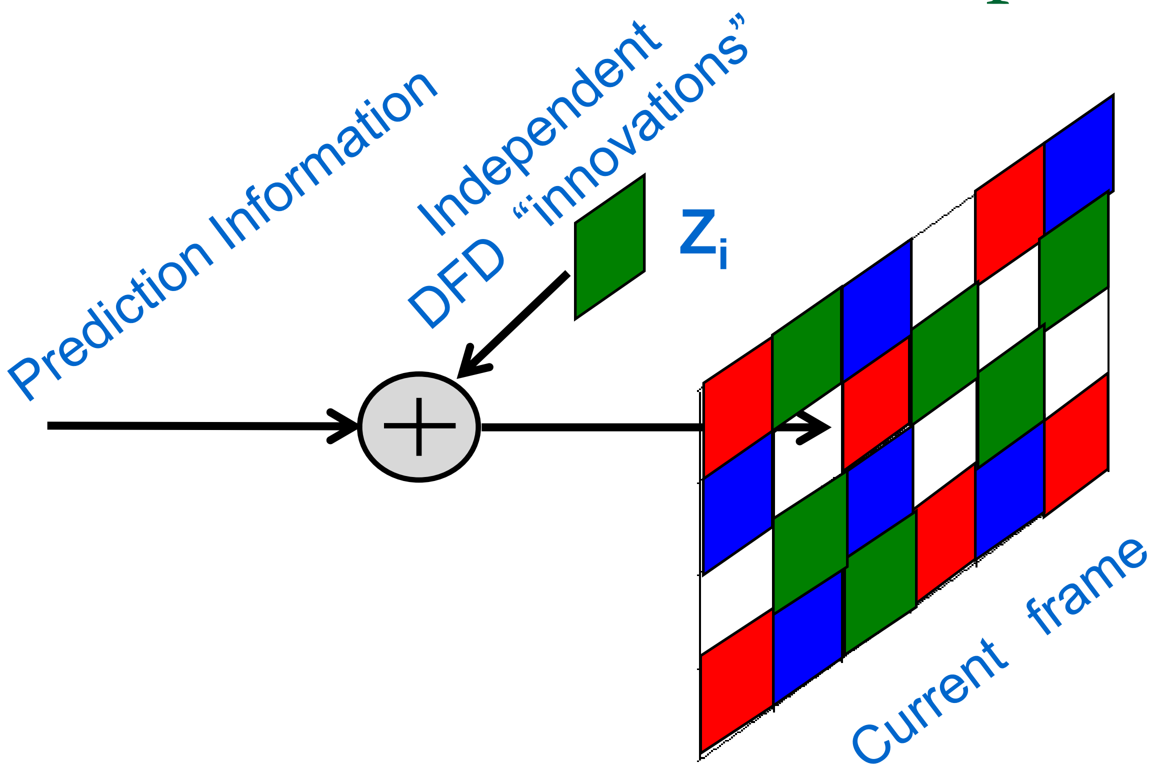
DFD Statistics: mixture process



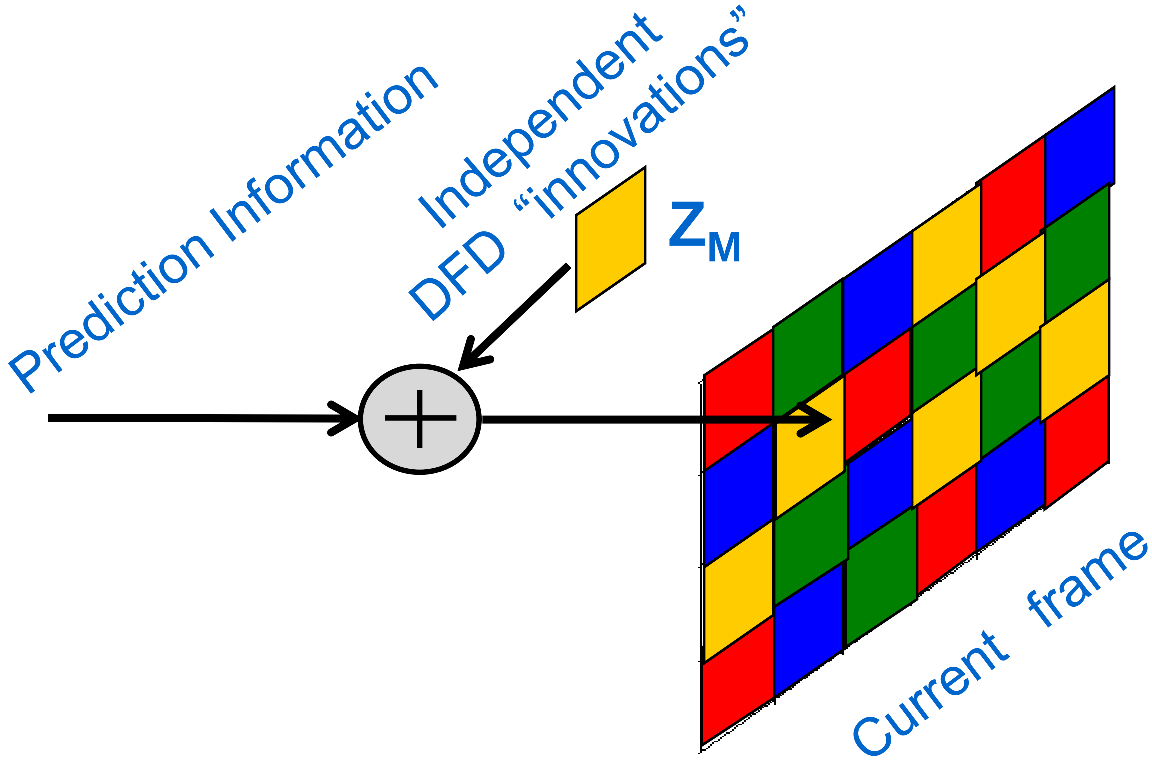
DFD Statistics: mixture process



DFD Statistics: mixture process

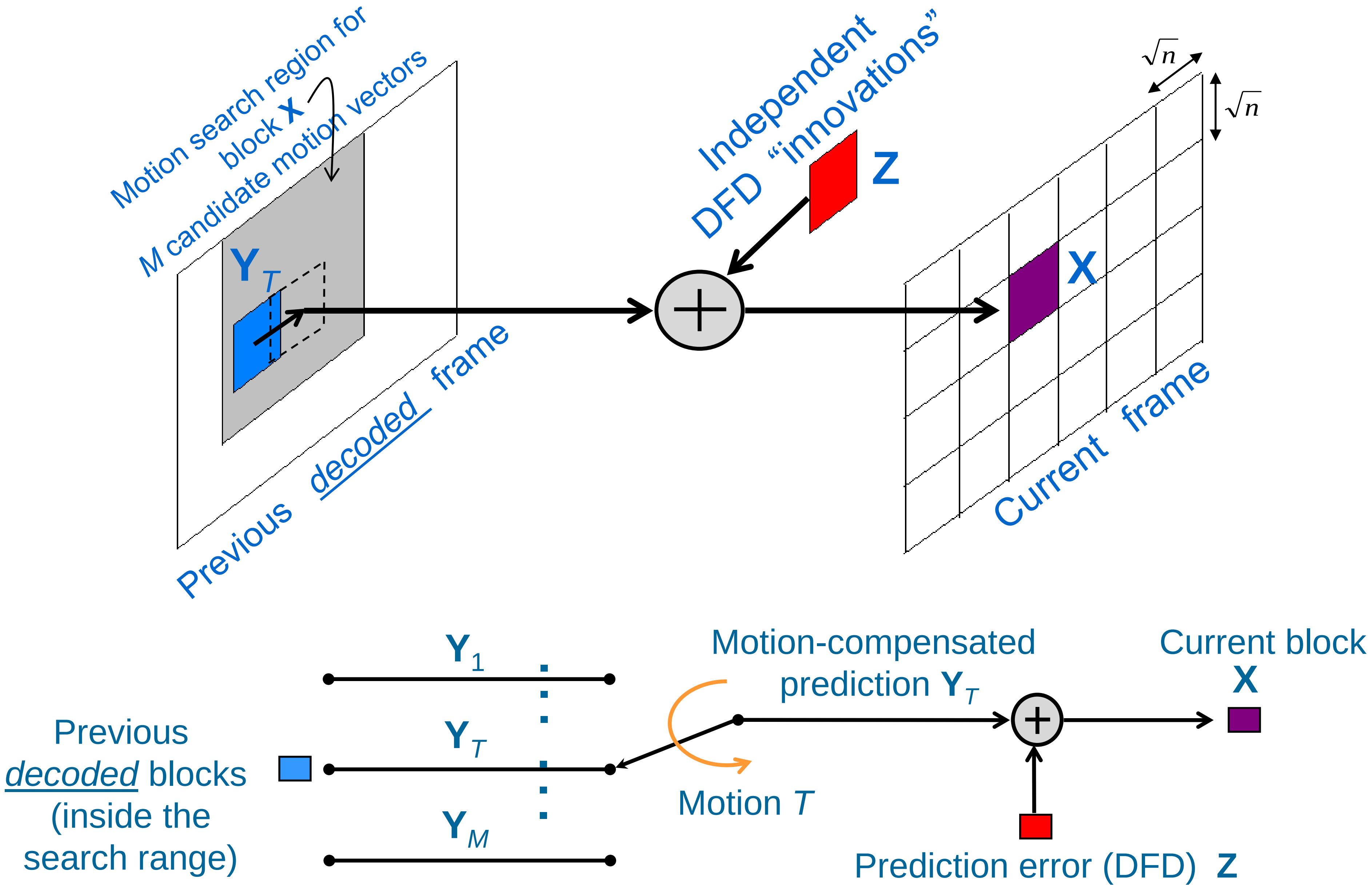


DFD statistics: mixture process

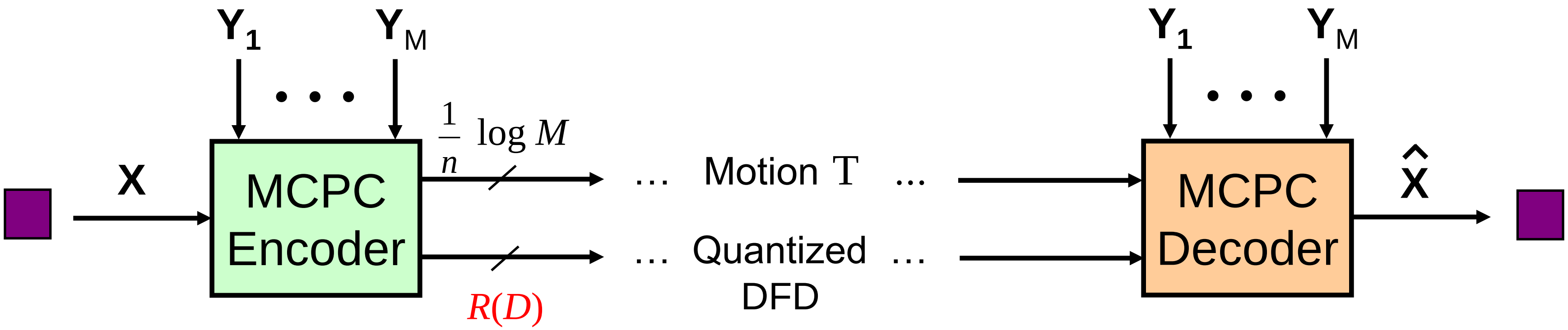


- Frame-level treatment of DFD ignores block-level statistical variations.
- Suggests block-level study of side-information coding problem
- Challenge: How to approach MCPC-like performance with/without doing motion search at the encoder?

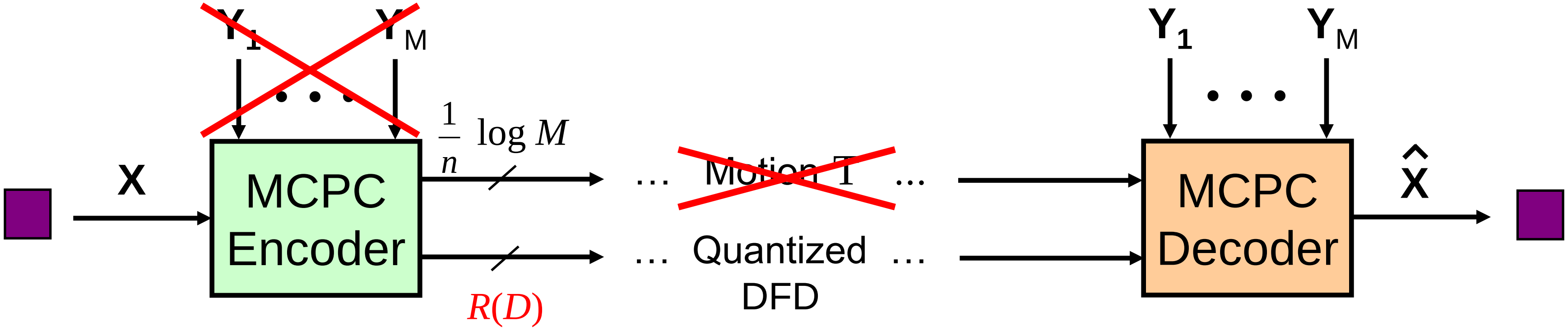
MCPC: a closer look



Motion-free encoding?

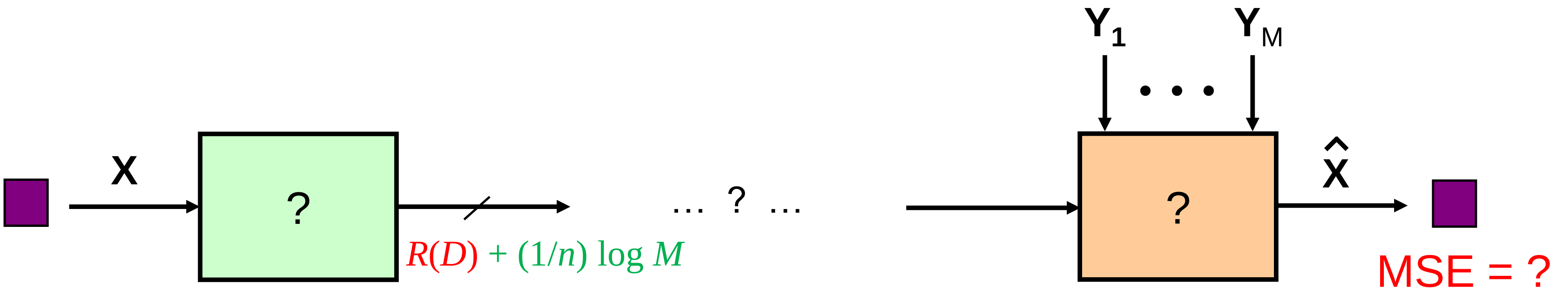


Motion-free encoding?



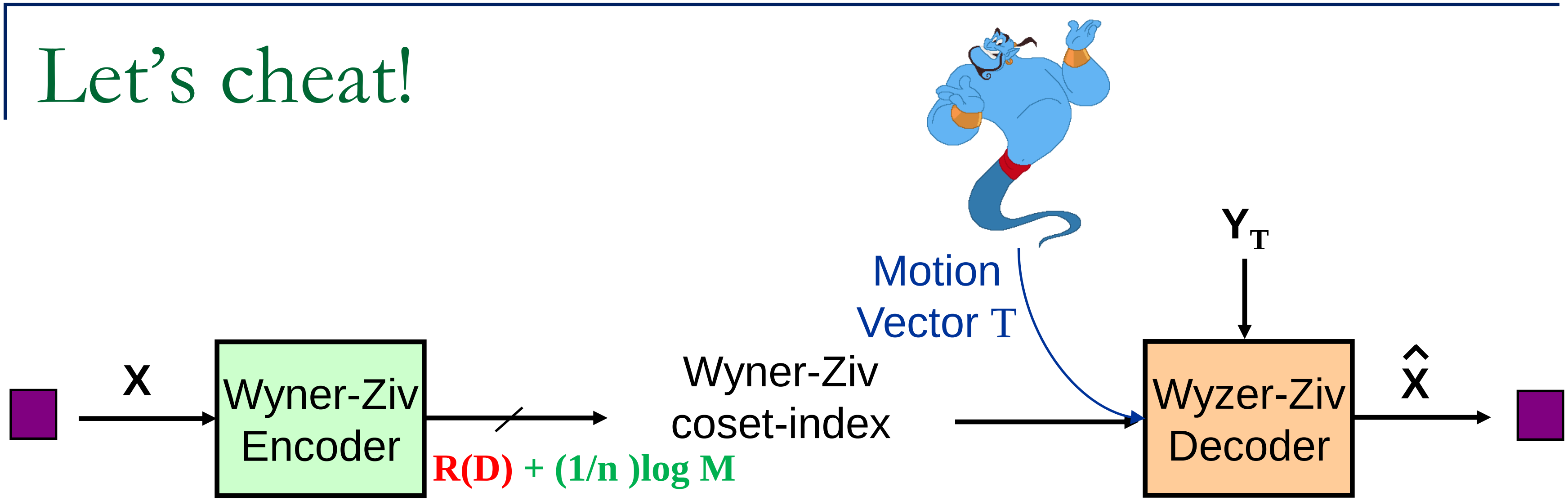
- The encoder does not have or cannot use $Y_1, \ldots, Y_M$
- The decoder does not know T

Motion-free encoding?



- The encoder does not have or cannot use $Y_1, \ldots, Y_M$
- The decoder does not know T
- The encoder may work at rate: $R(D) + (1/n)\log M$ bits per pixel.
- How to decode and what is the performance?

Let’s cheat!



- Let’s cheat and let the decoder have the motion vector T $\rightarrow$ “classical” Wyner-Ziv problem
- The encoder works at same rate as predictive coder

What if there is no genie?

- Can decoding work without a genie?
  - Yes
- Can we match the performance of predictive coding?
  - Yes (when DFD statistics are Gaussian)

Ishwar, Prabhakaran, and KR, ICIP '03.

Theorem

- Source Encoding with side-information under Ambiguous State Of Nature (SEASON)
  - The encoder does not have $Y_1, \ldots, Y_M$
  - Neither the decoder nor the encoder knows T
  - The MSE is still the same as for the MCPC codec

  - **Theorem:** SEASON codecs have the same rate-distortion performance as the MCPC codec for Gaussian DFD, i.e.

$$R(D) = ½ \log(\sigma_Z^2/D)$$

Theorem (cont'd)

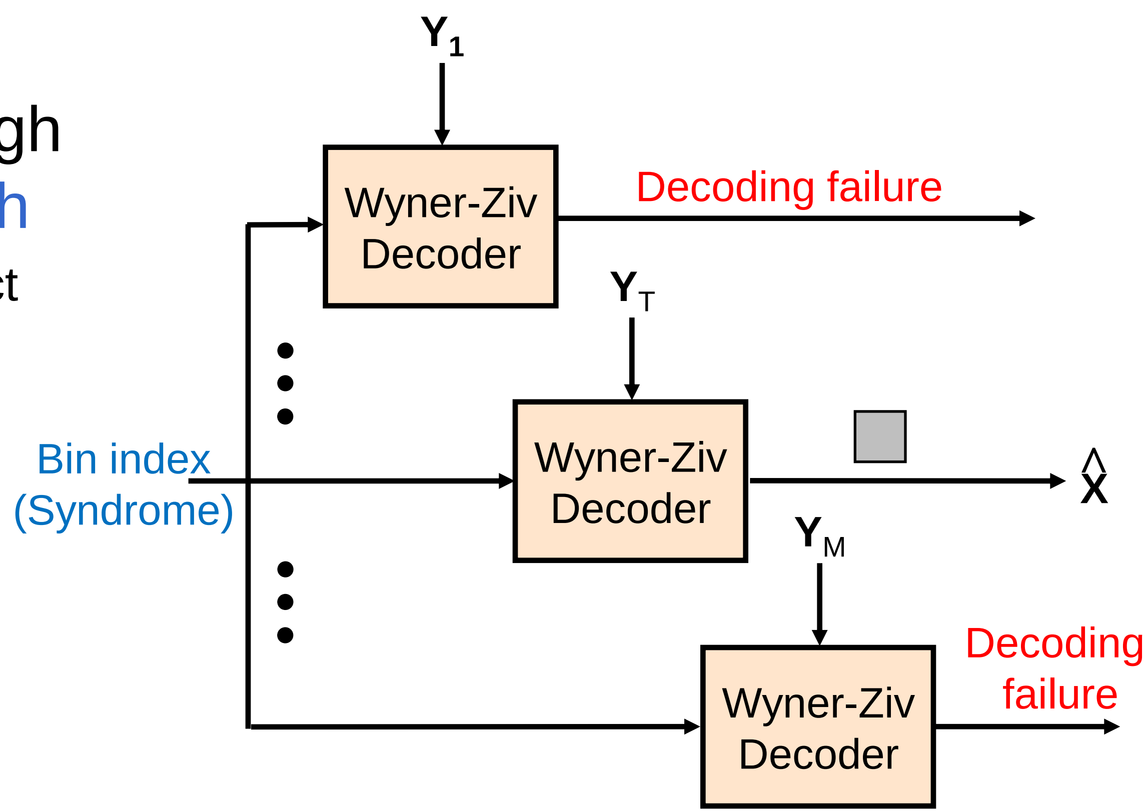
- Source Encoding with side-information under Ambiguous State Of Nature (SEASON)
  - **Theorem:** SEASON codecs have the same rate-distortion performance as the genie-aided codecs.
    This performance is (in general)

$$R(D) = \min_U I(U;X) - I(U;Y_T)$$

subj. $X \leftrightarrow Y_T \leftrightarrow U$ is Markov,
and $f(\cdot, \cdot)$ such that $E[(X-f(Y_T, U))^2] = D$

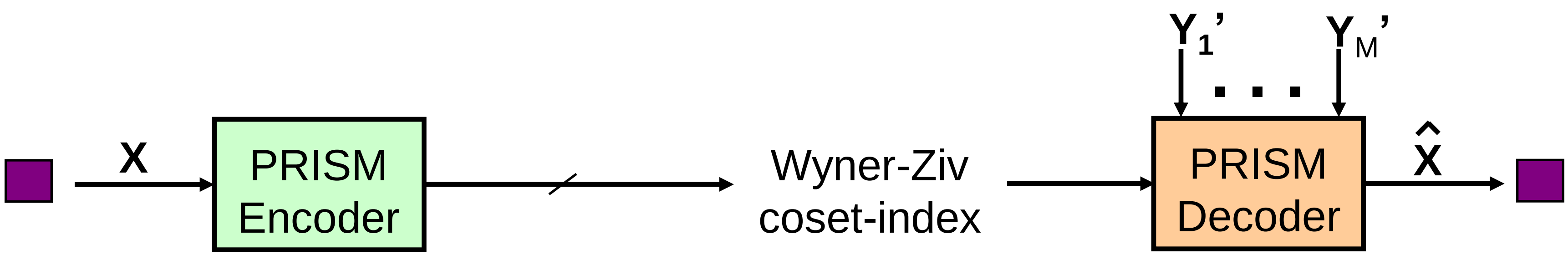
Practical implementation

- Low-complexity motion-free encoder
- Can be realized through decoder motion search
  - Need mechanism to detect decoding failure
  - In theory: joint typicality (statistical consistency)
  - In practice: Use CRC



Noisy channel: drift analysis

MCPC: Channel errors lead to prediction mismatch and drift.



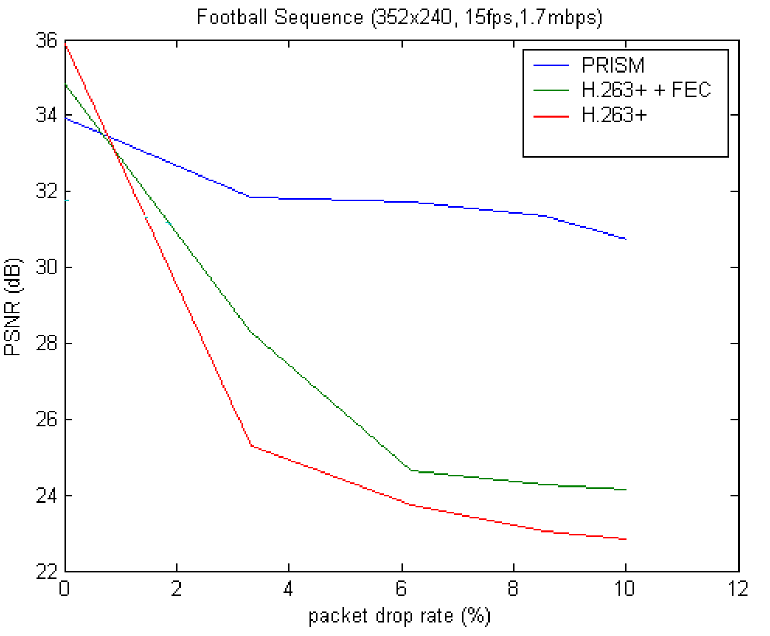
PRISM: Drift stopped if syndrome code is “strong enough”

- All that matters:

  Targeted syndrome code noise ≥
  Video innovation + Effect of Channel + Quantization Noise

Results

- Qualcomm's channel simulator for CDMA 2000 1X wireless networks



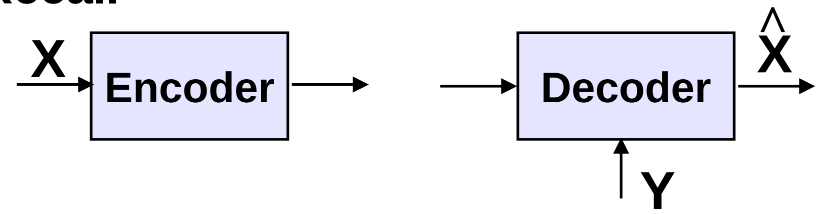
- Stefan
(SIF, 2.2 Mbps, 5% error)

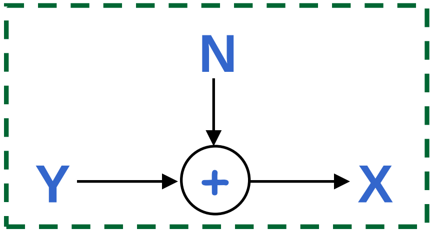
PRISM vs. H.263+

PRISM vs. H.263+ FEC

Challenges: correlation estimation

Recall





- PRISM:
  - Superior performance over lossy channels.
  - But compression efficiency inferior to predictive codecs.

- Challenge: correlation estimation, i.e. finding H(X|Y) = H(N)

- N = Video innovation + Effect of channel + Quantization noise

  Hard to model without motion search

- Without accurate estimate of the total noise statistics, need to over-design → compression inefficiency.

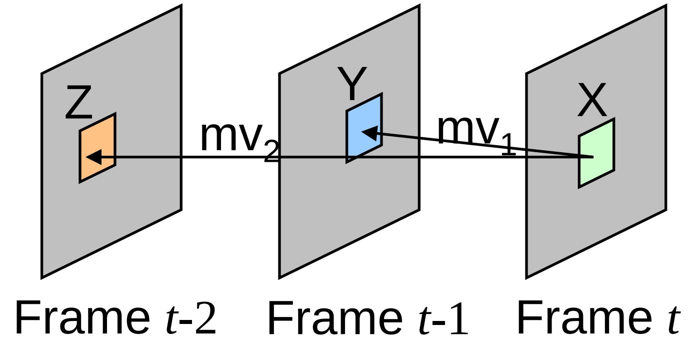
- What if complexity were less of a constraint and we allow motion search at the encoder?

What if complexity were less of a constraint?

- Allow motion search at encoder → can model video innovation
- Distributed Video Coding can approach the performance of predictive coders when it estimates the correlation structure accurately
- How to enhance robustness by considering effect of channel?

Modeling effect of channel at enc.: finding H(X|Y')

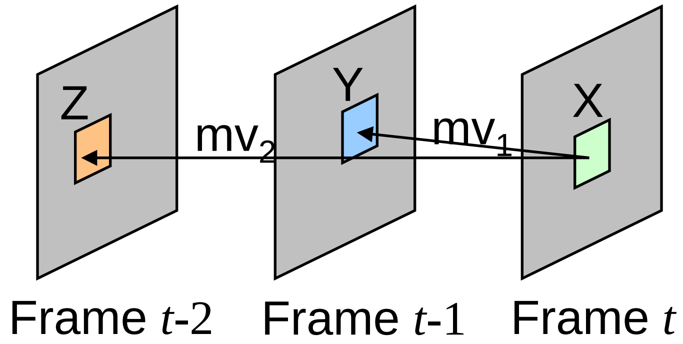
- Efficient strategy to exploit natural diversity in video data



  - Encoder has access to both Y and Z
- Fact: there is natural **diversity** in video data
  - An ***intact*** second best predictor (P_2) is typically a better predictor than a ***corrupted*** best predictor (P_1)
  - Can be viewed as motion search with two candidates.
- The decoder ***knows*** to use the better of P_1 or P_2 as SI.
- We have control over uncertainty set at decoder

J. Wang, V. Prabhakaran & KR: ICIP'06

Finding H(X|Y')

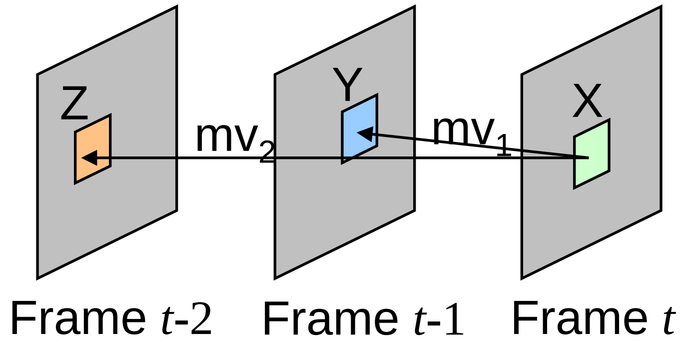


- If we have some knowledge about the channel:

$$Y' = \begin{cases} Y \text{ if } Y \text{ is intact} & \text{with probability } (1-p) \\ Z \text{ if } Y \text{ is corrupted} & \text{with probability } p \end{cases}$$

- We obtain
  $H(X|Y', \text{decoder state}) = (1-p)*H(X|Y) + p*H(X|Z)$

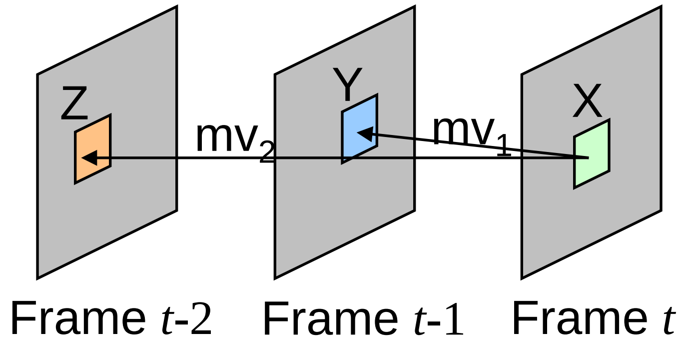
Another way to think about it



- $H(X|Y', \text{decoder state}) = (1-p)*H(X|Y) + p*H(X|Z)$

$$= \underbrace{\boxed{p*[H(X|Z) - H(X|Y)]}}_{\text{Effect of channel}} + \underbrace{\boxed{H(X|Y)}}_{\text{Video innovation}}$$

Yet another way to think about it



- H(X|Y', decoder state) = (1-p)*H(X|Y) + p*H(X|Z)

$$= \boxed{p} * \boxed{[H(X|Z) - H(X|Y)]} + \boxed{H(X|Y)}$$

Can be achieved by applying channel code to sub-bin indices

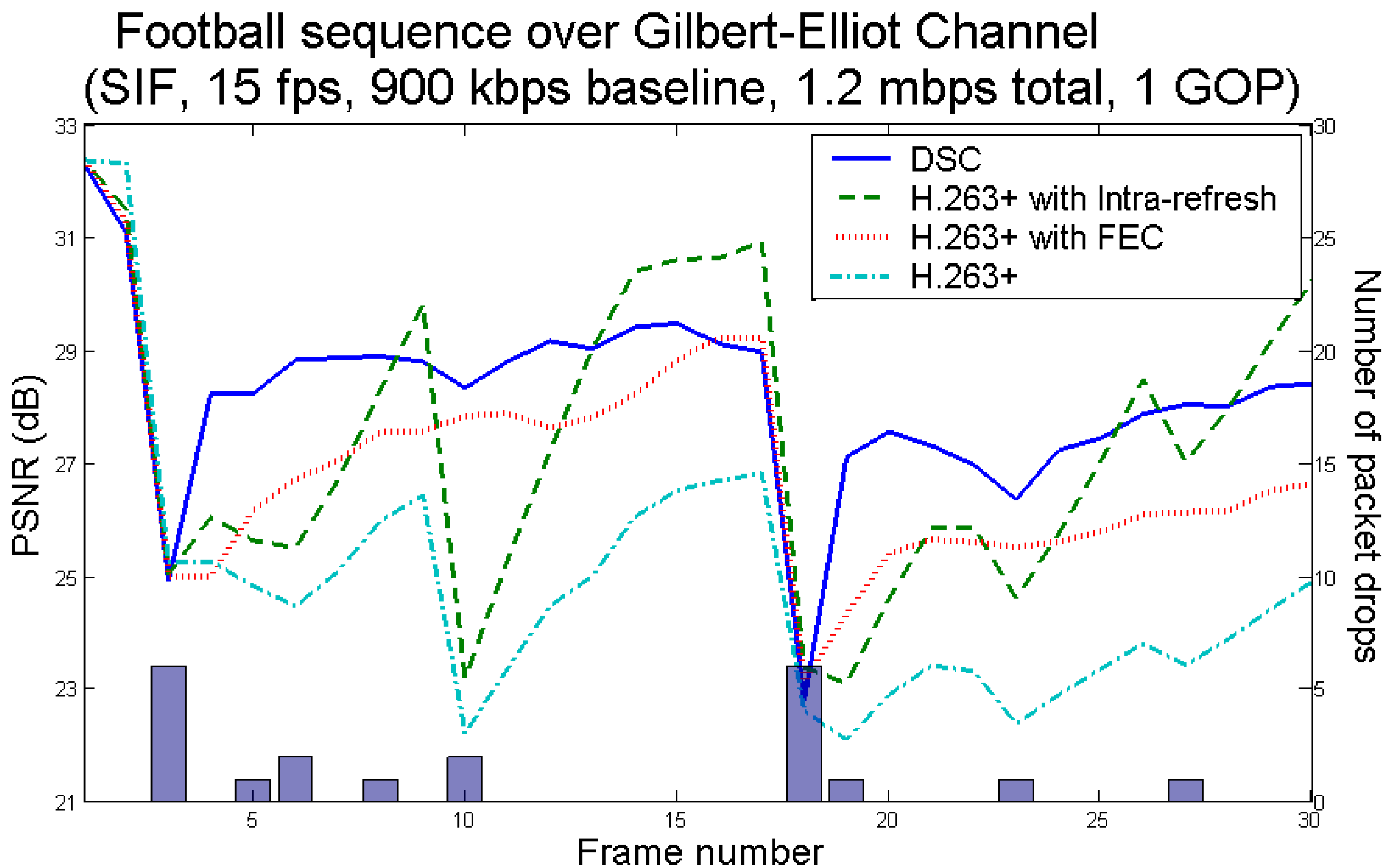
Additional syndrome (sub-bin index) for drift correction

Bare minimum syndrome (bin index) needed when channel is clean

Robustness result

Setup:

- Channel:
  - Simulated Gilbert-Elliot channel with $p_g = 0.03$ and $p_b = 0.3$



Robustness result

Setup:

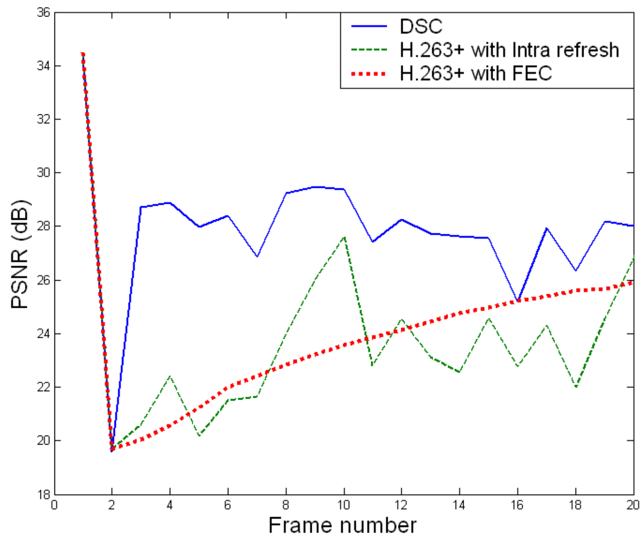
- Channel:
  - Simulated CDMA 2000 1x channel

Stefan (SIF) sequence

1 GOP = 20 frames

1 mbps baseline, 1.3 mbps total (15 fps)

7.1% average packet drop rate

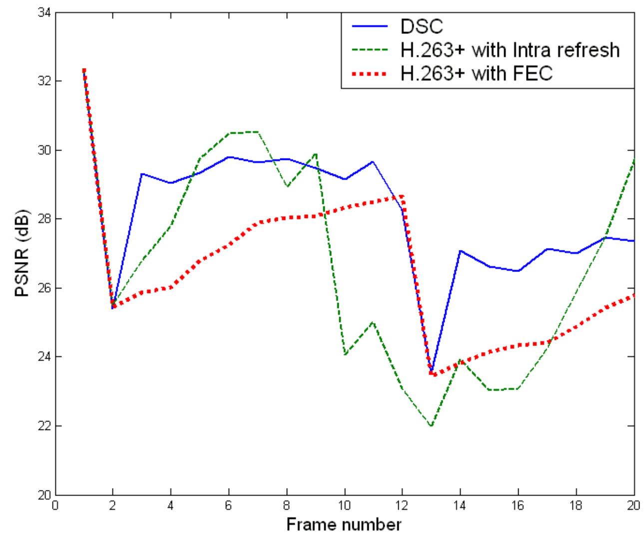


Football (SIF) sequence

1 GOP = 20 frames

900 kbps baseline, 1.12 mbps total (15 fps)

7.4% average packet drop rate



Videos

- Garden

352x240, 1.4 mbps, 15 fps, gop size 15, 4% error
(Gilbert Elliot channel with 3% error rate in good state and 30% in bad state)

DSC
vs.
H.263+ FEC

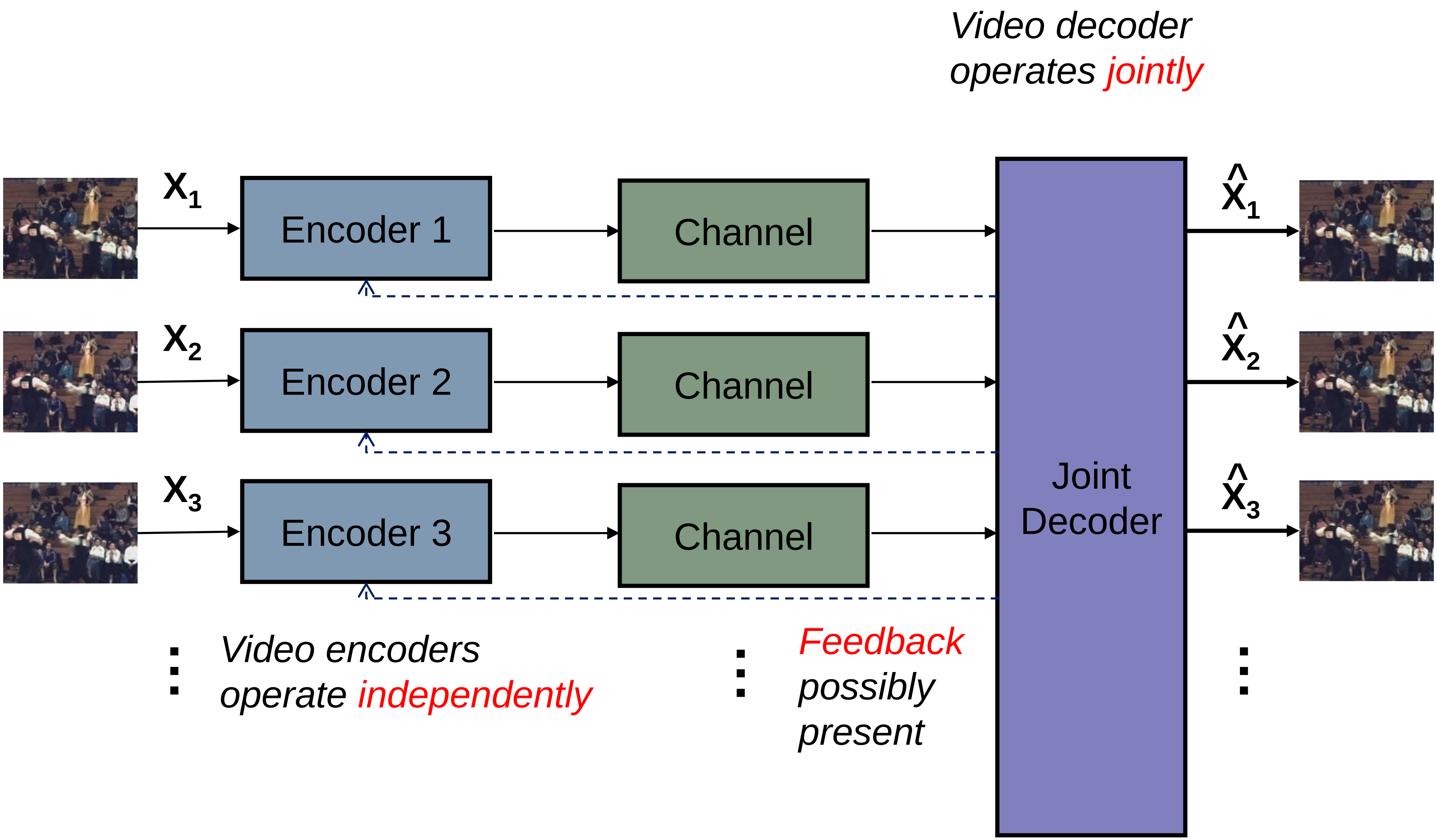
- Football

352x240, 1.12 mbps, 15 fps, gop 15, simulated CDMA channel with 5% error

DSC
vs.
H.263+ FEC

DSC for multi-camera video transmission

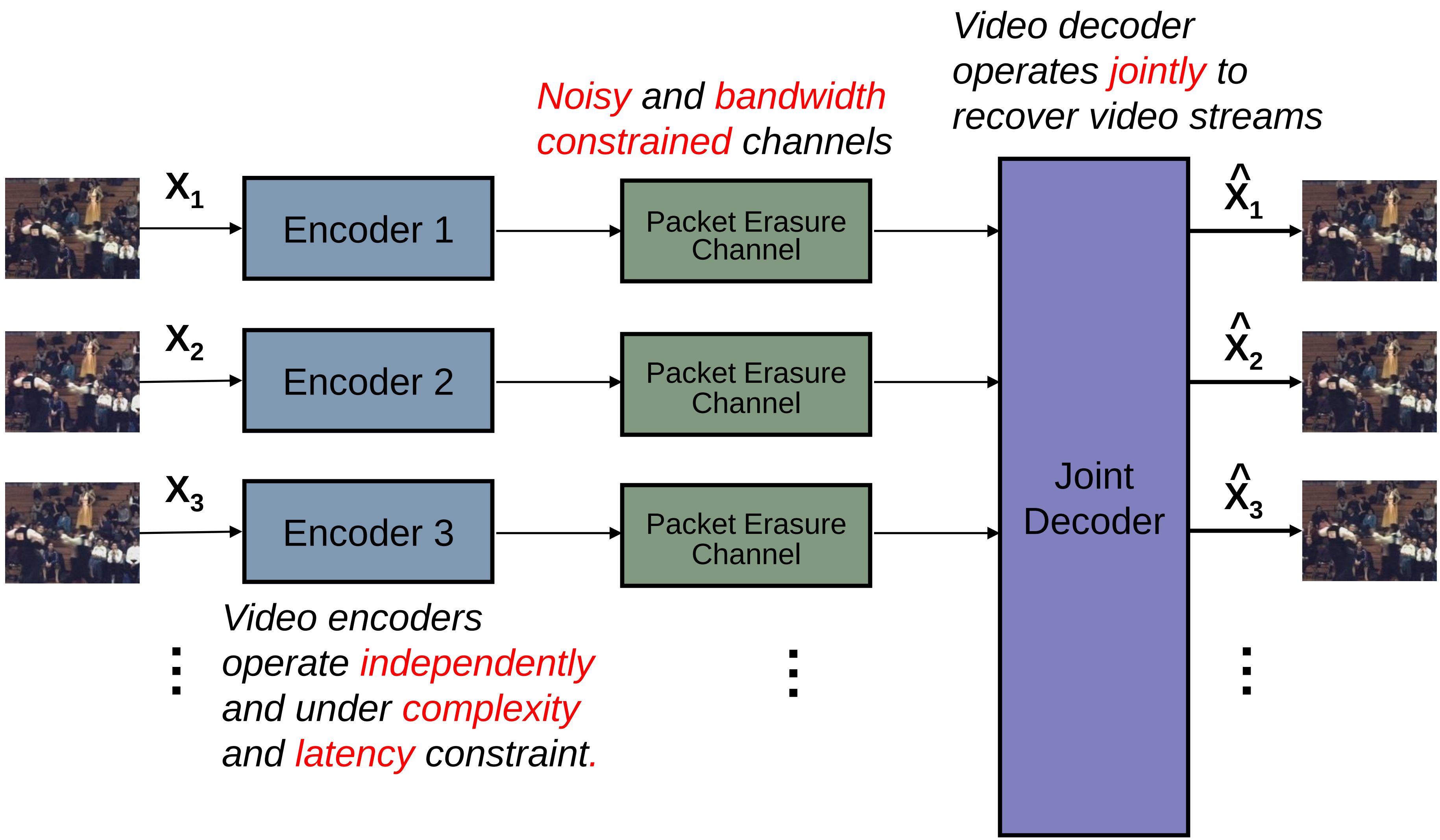
Distributed multi-view coding



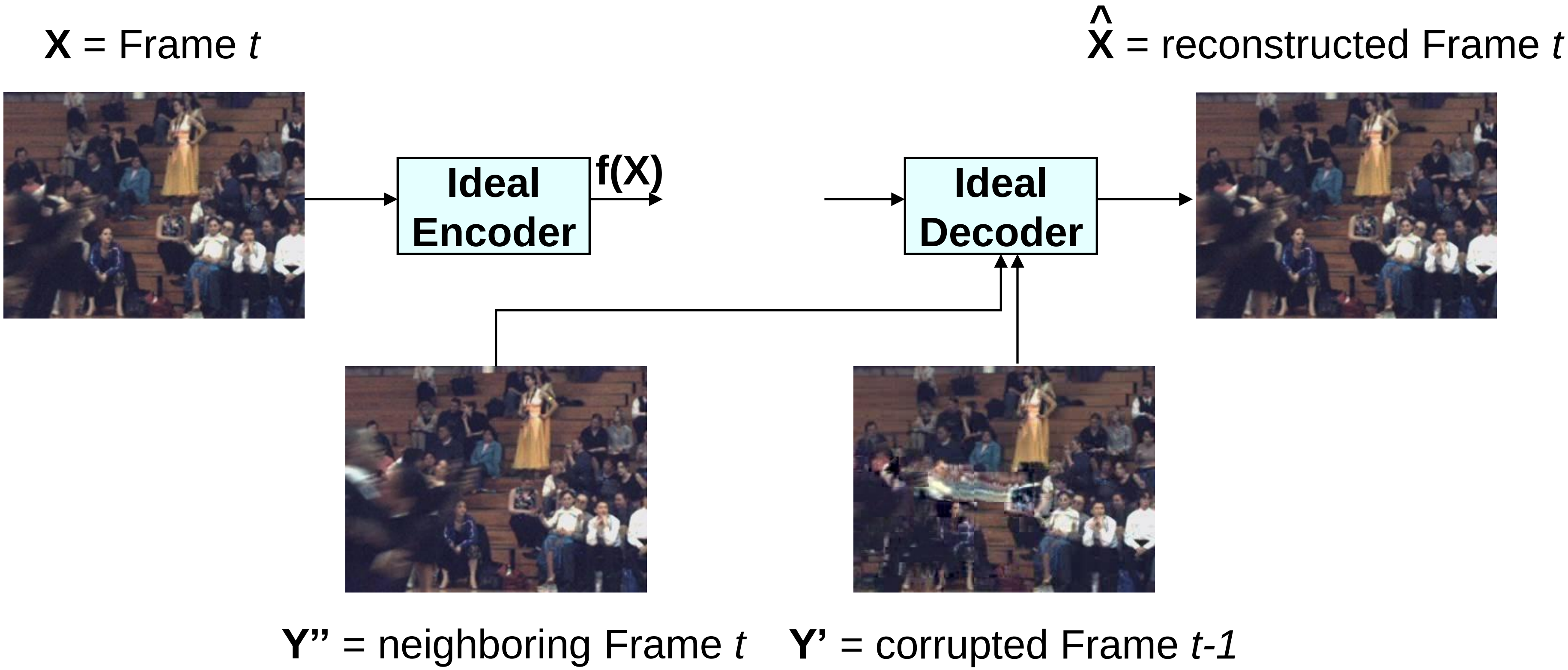
Active area of research

- Distributed multi-view image compression
  - Down-sample + Super-resolution *[Wagner, Nowak & Baraniuk, ICIP 2003]*
  - Geometry estimation + rendering *[Zhu, Aaron & Girod, SSP 2003]*
  - Direct coding of scene structure *[Gehrig & Dragotti, ICIP 2005] [Tosic & Frossard, ICIP 2007]*
  - Unsupervised learning of geometry *[Varodayan, Lin, Mavlankar, Flierl & Girod, PCS 2007]*
  - *…*
- Distributed multi-view video compression
  - Geometric constraints on motion vectors in multiple views *[Song, Bursalioglu, Roy-Chowdhury & Tuncel, ICASSP 2006] [Yang, Stankovic, Zhao & Xiong, ICIP 2007]*
  - Fusion of temporal and inter-view side-information *[Ouaret, Dufaux & Ebrahimi, VSSN 2006] [Guo, Lu, Wu, Gao & Li, VCIP 2006]*
  - MCTF followed by disparity compensation *[Flierl & Girod, ICIP 2006]*
  - *…*
- *Robust* distributed multi-view video compression
  - Disparity search / View synthesis search *[Yeo, Wang & Ramchandran, ICIP 2007]*

Robust distributed multi-view video transmission



Side information from other camera views

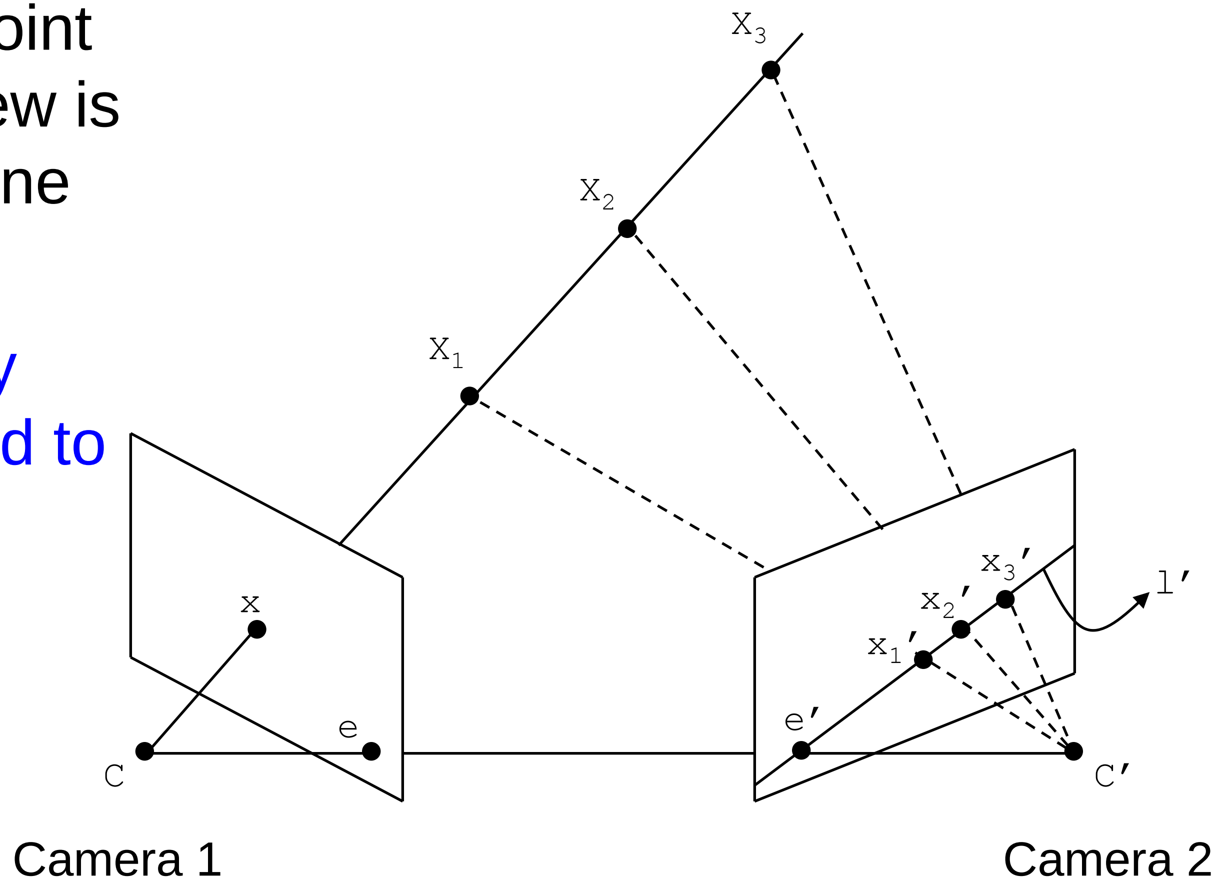


- How should we look in other camera views?
  - Naïve approach of looking everywhere can be extremely rate-inefficient
- Possible approaches
  - View synthesis search
  - Disparity search

Epipolar geometry

- Given an image point in one view, corresponding point in the second view is on the epipolar line

- Upshot: Disparity search is reduced to a 1-D search



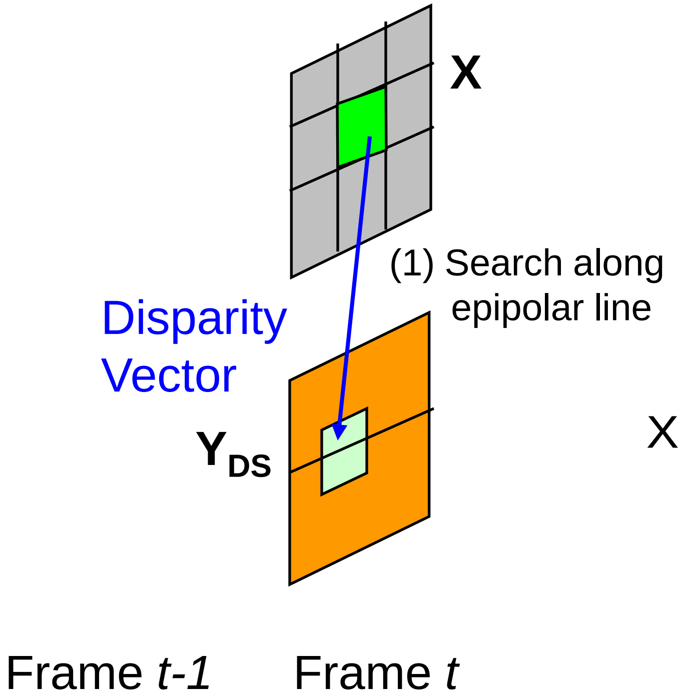
Decoder disparity search

Camera 1

Temporal – Poor reference

Camera 2

Spatial – Good reference



$X = Y_{DS} + N_{DS}$

- Extension of decoder motion search using epipolar geometry

[Yeo & Ramchandran, VCIP 2007]

PRISM-DS vs MPEG with FEC

- "Ballroom" sequence (from MERL)
  - 320x240, 960 Kbps, 30fps, GOP size 25, 8% average packet loss

Original

MPEG+FEC

PRISM-DS

- Drift is reduced in PRISM-DS

[Yeo & Ramchandran, VCIP 2007]