

On Compression Encrypted Data – part 1

Prof. Ja-Ling Wu
The Graduate Institute of
Networking and Multimedia
National Taiwan University

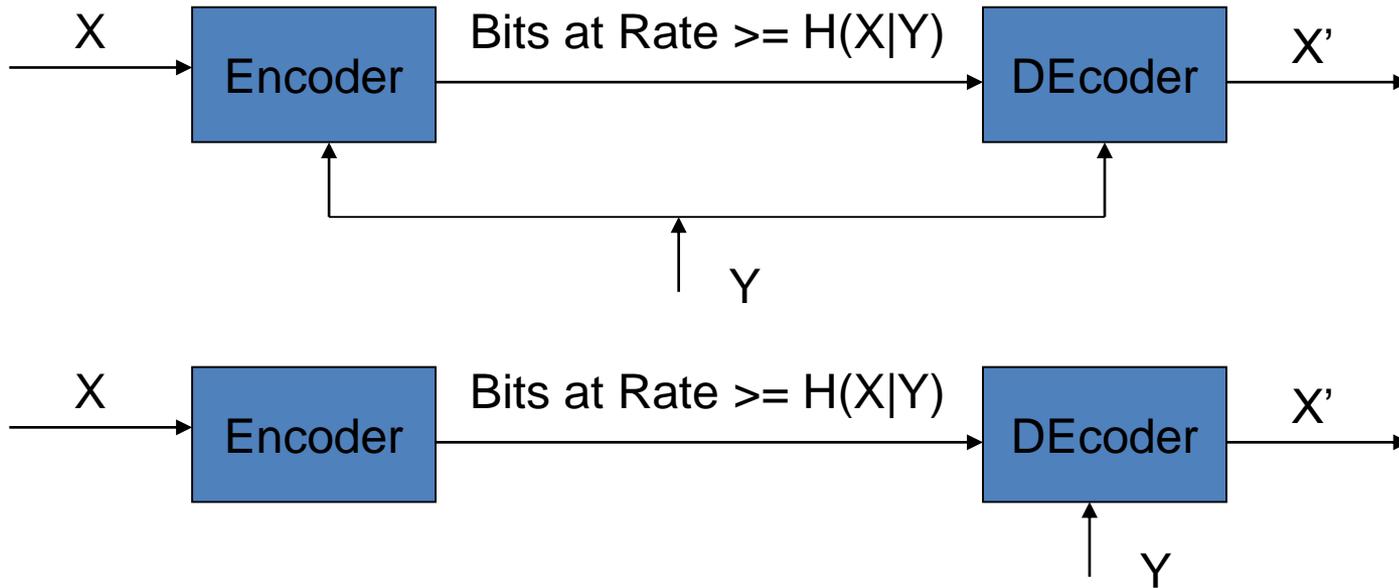


- Cited from: On Compression encrypted Data, IEEE Transactions on Signal Processing, Vol. 52, No. 5, pp. 2992-3006, Oct. 2004 ---

co-authored by:

M. Johnson, P. Ishwar, V. M.,
Prabhakaran, D. Schonberg, and
[K. Ramchandran \(U.C. Berkeley\)](#)





Problem:

Assume X and Y are **correlated** discrete alphabet memoryless sources, and we have to **compress X losslessly**, **with Y** (referred to as **side information**) being known at the decoder but not at the encoder.



If Y were known at both sides, then the problem of compressing X is well-understood : one can compress X at the theoretical rate of its **condition Entropy** given Y , $H(X|Y)$.

But what if Y were **known only at the decoder** for X and not at the encoder?



The answer is that one can still compress X using only $H(X|Y)$ bits, the same as the case where the encoder does know Y .



- By just knowing the joint distribution of X and Y , without explicitly knowing Y , the encoder of X can perform as well as an encoder which explicitly knows Y .
- This is known as the **Slepian-Wolf coding theorem**.



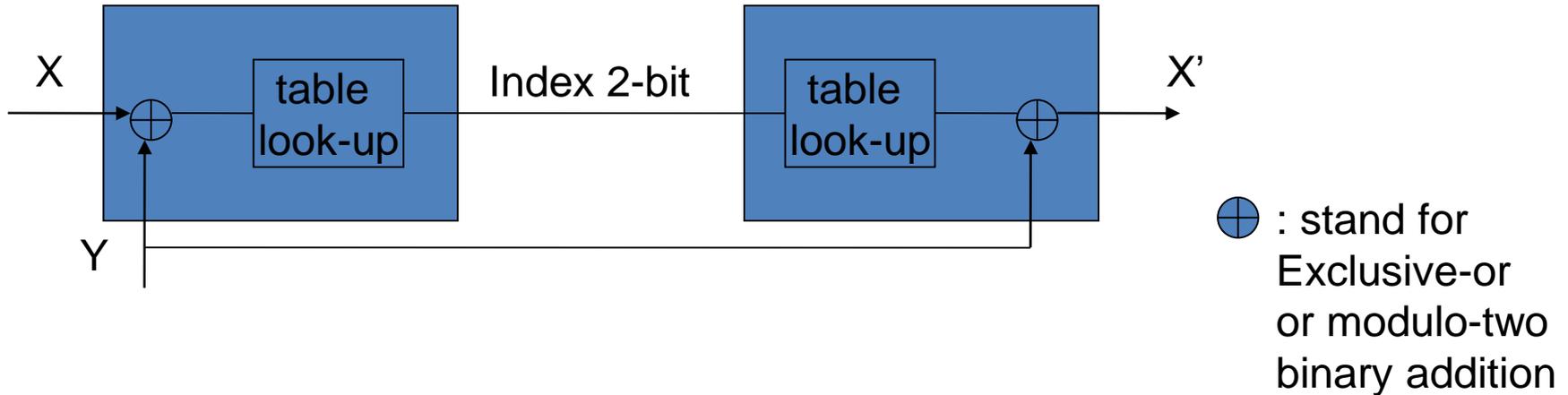
An illustrative example:

Suppose X and Y are equiprobable 3-bits binary words that are correlated in the following sense:

The Hamming distance between X and Y is at most 1 --- correlation between X and Y .

If Y (side information) is available to both the encoder and the decoder, we can describe X using 2 bits (there are only 4 possibilities for the modulo-two binary sum of X and Y : $\{(000), (001), (010), (100)\}$ index $(00), (01), (10), (11)$).





Now what if Y were revealed only to the decoder but not the encoder : Could X still be described using only 2 bits?



Since the decoder knows Y , it is wasteful for X to spend any bits in differentiating between $\{X=(000)$ and $X=(111)\}$, since **the Hamming** distance between these two codewords **is 3 --- do not follow the correlation constraint.**

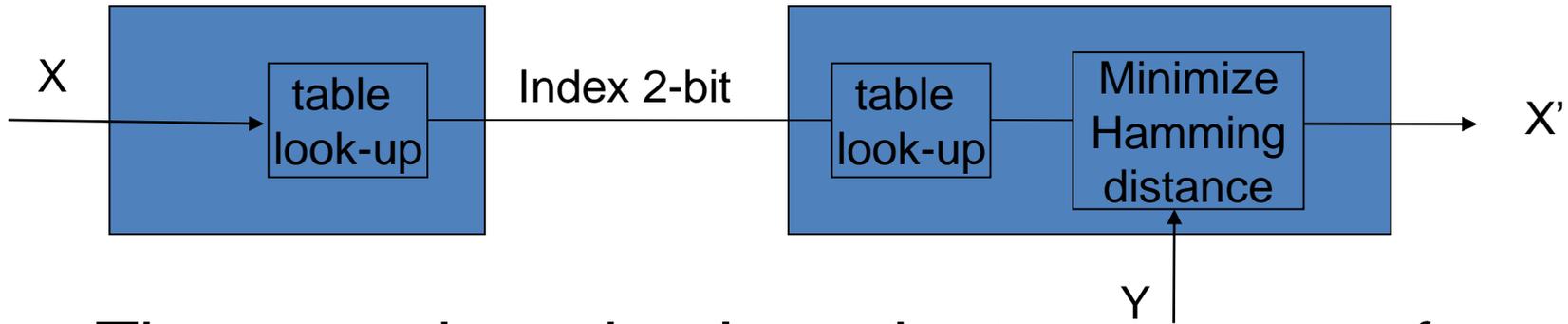
Thus, if the decoder knows that either $X=(000)$ or $X=(111)$, it can resolve this uncertainty by **checking which of them is closer in Hamming distance to Y** , and declaring that as the value of X .



Likewise, in addition to the set $\{(000), (111)\}$, each of the following three sets for X : $\{(100), (011)\}$, $\{(010), (101)\}$, and $\{(001), (110)\}$ is composed of pair of words whose Hamming distance is 3.

X : $\{(000), (111)\}$, $\{(001), (110)\}$, $\{(010), (101)\}$, $\{(100), (011)\}$
Index (00) (01) (10) (11)





These are just simple variants or cosets of 3-bit repetition code, and they cover the space of all binary 3-tuples that X can assume.

3-bit repetition code

error pattern	codewords
0 0 0	0 0 0 1 1 1
0 0 1	0 0 1 1 1 0
0 1 0	0 1 0 1 0 1
1 0 0	1 0 0 0 1 1



Thus, instead of describing X by its 3-bit value, we encode which coset X belongs to, incurring a cost of 2 bits, just as in the case where Y is known to both encoder and decoder.

Definition : **one time pad**

$$c_i = p_i \oplus k_i$$

where p_i = i -th binary digit of plaintext

k_i = i -th binary digit of a random key

c_i = i -th binary digit of ciphertext



If X_0 can only take on the values $\{000, 001, 010, 100\}$, then the Hamming distance between X and Y is at most 1. We can use the prescribed construction to compress X (an encrypted data) to 2 bits, and a decoder which has access to Y will be able to correctly decoder X . The decoder can then recover the original data X_0 (coset leader/error pattern) by computing

$$X_0 = X \oplus Y$$

If we treat X as P and Y as K

then the above scheme behaves like a one-time pad encryption scheme.



The above construction can be extended beyond the simple example considered here. The space of all possible words is **partitioned into cosets**, which are associated with the **syndromes (coset leader)** of the principal underlying **channel code**.

The encoding procedure is to compute the syndrome of X w.r.t the appropriate channel code and **transmit this syndrome to the decoder**.

The choice of channel code depends on the **correlation structure between X and Y** .



- If X and Y are more correlated, then the required strength (length) of the code is less (shorter).
- The decoding procedure is to identify the closest codeword to Y in the coset associated with the transmitted syndrome, and declare that codeword to be X .



Encoding with a Fidelity Criterion

- A. Problem formulation

Here we consider the continuous-valued source X and side-information Y . Specifically, X and Y are **correlated** memoryless processes **characterized** by **independent and identically distributed (i.i.d)** sequences $\{X_i\}_{i=1}^{\infty}$ and $\{Y_i\}_{i=1}^{\infty}$, respectively.



- We consider the special case where **Y is a noise version of X**: i.e., $Y_i = X_i + N_i$, where $\{N_i\}_{i=1}^{\infty}$ is also continuous valued (defined on the real line \mathfrak{R}), i.i.d, and independent of the X_i 's.
- As before, the **decoder alone** has access to the **Y** process (side information), and the task is to **optimally compress the X process** .



- For the rest of our discussion , we will confine ourselves to the case where the X_i 's and N_i 's are **zero-mean Gaussian** random variables with **known variances**, so as to benchmark the performance against the theoretical performance bound.



- The goal is to form **the best approximation** \hat{X} to X given an encoding rate **R** bits per sample. We assume encoding in blocks of length L . Let $\rho(\cdot)$ be a function $\rho: \mathcal{R} \times \mathcal{R} \rightarrow \mathcal{R}^+$. Let the **distortion measure be** $\rho(\cdot)$ over the L -sequence, and we assume on additive distortion measure

$$\rho(\bar{x}, \hat{\hat{x}}) = \frac{1}{L} \sum_{i=1}^L \rho(x_i, \hat{x}_i)$$

MSE distortion: $\rho(x, \hat{x}) = (x - \hat{x})^2$

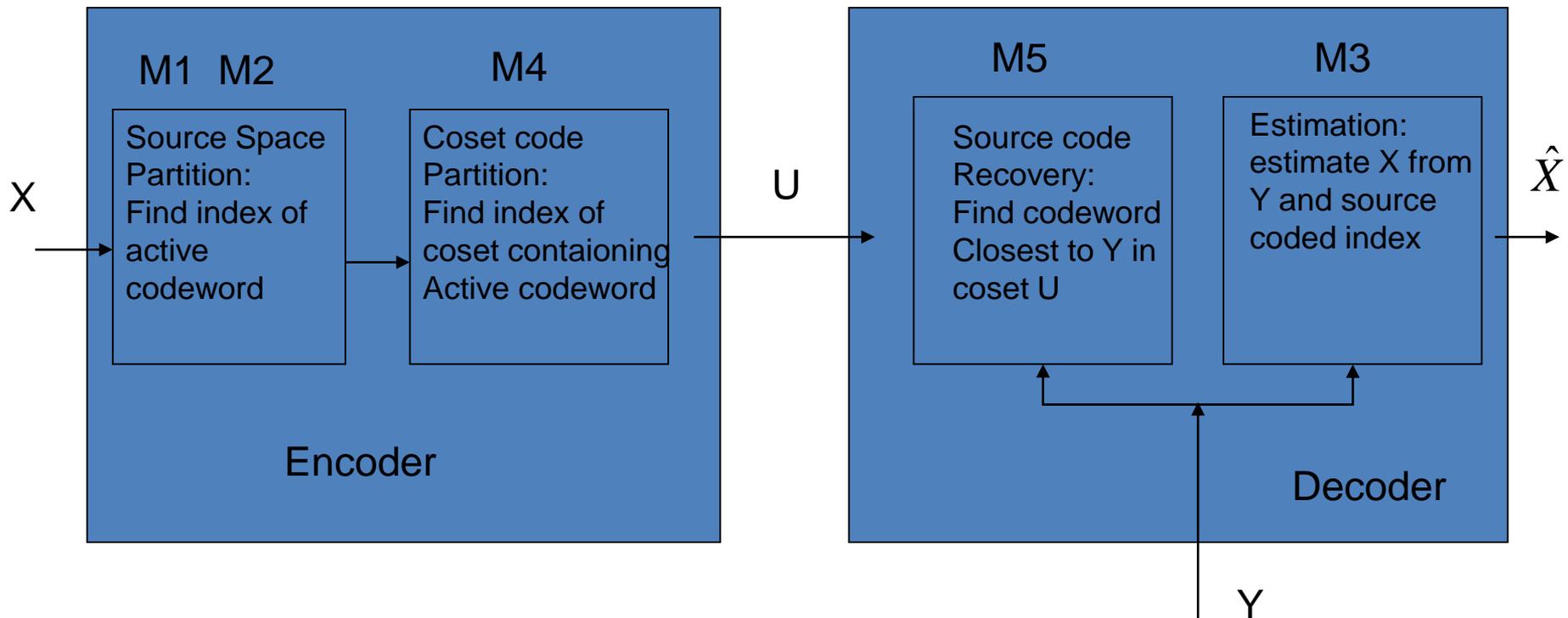


- This problem can be posed as minimizing the rate of transmission \mathfrak{R} such that the reconstruction fidelity $E\{\rho(X,Y)\}$ is less than a given value D , where $E(\cdot)$ is the expectation operator.
- The encoder is a mapping from the input space to the index set: $\mathfrak{R}^L \rightarrow \{1,2,\dots,2^{LR}\}$, and the decoder is a mapping from the product space of the encoded index set and the correlated L-sequence Y to the L-sequence reconstruction:

$$\{1,2,\dots,2^{LR}\} \times \mathfrak{R}^L \rightarrow \mathfrak{R}^L$$



- B. Design Algorithm



$\{M_i\}_{i=1}^5$ are five different mappings.



- Source coding (M_1, M_2) : Due to the finite rate constraint on the information transmitted, the source X has to be **quantized** .
- A source codebook is constructed for a give fidelity. The **source space** \mathfrak{R}^L is **partitioned** into 2^{LR_s} **disjoint regions**, where R_s is defined as the **source rate**. It is a mapping

$$M_1 : \mathfrak{R}^L \rightarrow \{1, 2, \dots, 2^{LR_s}\}$$



- Let $T = \{T_1, T_2, \dots, T_{2^{LR_s}}\}$ denote the set of 2^{LR_s} disjoint regions. Each region in the above partition is associated with a representation codeword. The set of representation codewords is referred to as the **source codebook (S)**. This is a mapping

$$M_2 : \{1, 2, \dots, 2^{LR_s}\} \rightarrow \mathfrak{R}^L$$

The objective is to design these mappings M_1 and M_2

The source is quantized to one of the codewords in **S** and the index of the quantized codeword is made available to the decoder error free, by transmitting at a rate R_s bps.



- We refer to the representation codeword to which X is quantized as the **active codeword**. Let the random variable characterizing the active codeword be denoted by W . Note, unlike traditional source coding, **the active codeword is not used as the reconstruction for the source (an extra ECC process will be involved)**.
- Rather, the **decoding** further involves “**estimation of the source**” based on the **available information about the source**, the result of estimation is used as the final reconstruction.



We consider a design of the partition T of the source space based on the marginal distribution of X . This can be done, for example, by generalizing the Lloyd algorithm (used in vector quantization).

- Estimation (M_3): The decoder gets the best estimate of X (minimizing the distortion) conditioned on the outcome of the side information and the element in T containing X . This is given by

$$\hat{x} = \arg \min_{\bar{a} \in \hat{A}^L} E[r(x, \bar{a}) | x \in T_i, Y = \bar{y}]$$

for the received message \bar{c} and the side-information outcome \bar{y} .



- It can be interpreted as a mapping

$$M_3 : \mathfrak{R}^L \times \{1, 2, \dots, 2^{LR_s}\} \rightarrow \mathfrak{R}^L$$

The estimation error is a function of R_s , which is chosen to keep this error within the given fidelity criterion.



- Channel coding : (M_4, M_5)

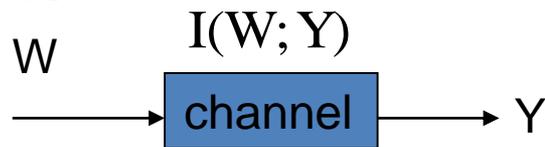
At this stage, the system requires a transmission of R_s bits per sample to guarantee the given fidelity criterion. By exploiting the correlation between X and Y , we make the decoder recover (with an arbitrarily small probability of error) the index of the active codeword with a transmission over an **error-free channel (with the help of proper channel codes) at rate lower than R_s .**



- This is done by noting that the random variable W characterizing the quantized source is correlated to X , and this in turn induces a correlation between W and the side information Y .
- This correlation can be characterized by a conditional distribution $P(Y | W)$ of the side information given W . With this conditional distribution we can associate a fictitious (virtual) channel with W as input which is observed at the encoder, and Y as output which is observed at the decoder, whose information channel capacity is greater than zero (due to this correlation).



- To actually communicate W to the decoder in the absence of side information requires a transmission rate R_s bps. With the presence of Y at the decoder, we have this fictitious “helper” channel carrying an amount of information $I(W; Y)$ about W .



- The remaining uncertainty in W after observing the side information Y is
$$H(W|Y) = H(W) - I(W;Y)$$
and this is **the desired final rate of transmission**.
The **rebate**(折扣) in the rate of transmission is $I(W;Y)$.
- Using this intuition, our goal is to get a rebate as close to $I(W;Y)$ as possible by building a practical structured “channel code” (C') for this fictitious channel on the space of W .



- Let 2^{LRc} denote the number of codewords in the designed channel code where Rc is defined as the **channel rate**.
- Suppose for a given realization, the **active codeword** (say W) belongs to this **channel code** (such as Turbo-code, LDPC, LDPCA) and this is **known at the decoder** (say communicated by a genie), then we do not need to send any information to the receiver, as it **can recover the intended codeword with a small probability of error by decoding Y with the aid of the channel code C'** . (this can be interpreted as transmitting W over the fictitious channel and observing the output of this channel Y as **side information** at the decoder).



- Since, in general, any codeword in the source codebook can be a quantization outcome with a nonzero probability, we partition the source codebook space into cosets of this channel code. The channel code is designed in such a way that “Each of Its Cosets is also an Equally Good Channel code for the channel $P(Y|W)$ ”.
- Thus, each quantization outcome belongs to a coset of this channel code, and this along has to be converged to the decoder, which can then proceed to use this coset of channel code for finding the intended active codeword.



- The encoder computes the index of the coset of the channel code containing the active codeword using a mapping

$$M_4 : \{1, 2, \dots, 2^{LR_s}\} \rightarrow \{1, 2, \dots, 2^{LR_c}\}$$

and transmit this information with rate $R = R_s - R_c$ bits per sample to the decoder.



- The decoder recovers the active codeword in the signaled coset by finding (channel coding) the **most likely codeword** given the observed **side information**. This is characterized by a mapping

$$M_5 : \mathfrak{R}^L \times \{1, 2, \dots, 2^{LR}\} \rightarrow \{1, 2, \dots, 2^{LR_s}\}$$



- In this approach, there is always a nonzero probability of decoding error, where the side information is decoded to a wrong codeword, and this can be made arbitrarily small by designing **efficient channel codes**.
- For a given region T , the choice of the representation codeword determines $I(W;Y)$, and hence the constructive frame work is referred to as “**Distributed Source coding using Syndromes (DISCUS)**”.



Summary of Design Algorithm :

- The rate of transmission and the distortion performance depend on the source and the channel codebooks. Thus, for a given fidelity criterion, our objective is to minimize R_s and maximize R_c . This is summarized as follows :
- (1) M_1 and M_3 : Minimizing R_s such that the reconstruction distortion is within the given criterion.



- (2) M_2 : The idea goal is to maximize the mutual information between W and Y , $I(W;Y)$. But as an approximation to it, we take the representation codeword of T_i as that vector for which T_i is its Voronoi region (which also amounts to a construction on M_1).
- (3) M_4 : Maximize the rate R_c of the channel code \mathbf{C} (and each of its cosets) such that probability of decoding error meets a desired tolerance level.
- (4) M_5 : Minimize computational complexity of the decoding rule.



- The final reconstruction \hat{X} is given by

$$\hat{X} = M_3[Y, M_5\{Y, M_4(M_1(X))\}]$$

for the source and the side-information vectors X , Y , respectively.

- The system has two separate entities in the design of the encoder and the decoder.
- The quantizer and the estimator are designed to minimize R_s , and the channel code is designed to maximize R_c .
- The fidelity during correct decoding is guaranteed by source coding design, and the probability of decoding error is controlled by the channel coding design. The distortion during decoding error can be bounded.



A Video Coding Architecture Based on Distributed compression Principles



- In the near future, **multiple video input** and output streams are expected to be used to enhance user experience. These streams need to be captured using a network of distributed devices and transmitted over a bandwidth constrained, noisy wireless transmission medium, **to a central location for processing**, with the goal for example, of creating high-resolution video using inexpensive cameras.



Uplink rich media application V.S.

Downlink video delivery model (traditional approach)

New demands:

- low-power and light-footprint encoding due to limited power and/or device memory.
- high compression efficiency due to both bandwidth and transmission power limitation.
- robustness to packet/frame loss caused by channel transmission errors.



H.26x/MPEG and HEVC video coding standards:

Computationally heavy at the encoder (primarily due to motion-search) and very fragile to packet loss



they achieve state-of-the-art compression efficiency but fail to meet the other two criteria.



Alternatively, **intra-frame** video coding methods (motion-JPEG) have low compression complexity, and are **relatively robust to packet drops due to lack of dependencies among frames**, but they take a relatively high bitrate in compression efficiency and resulting transmission power.

Question:

Is it possible to achieve full-motion-MPEG-like compression efficiency at motion-JPEG-like encoding complexity and robustness?



Characteristics of today's **Inter-frame Motion-compensated predictive coding** framework:

computational complexity:

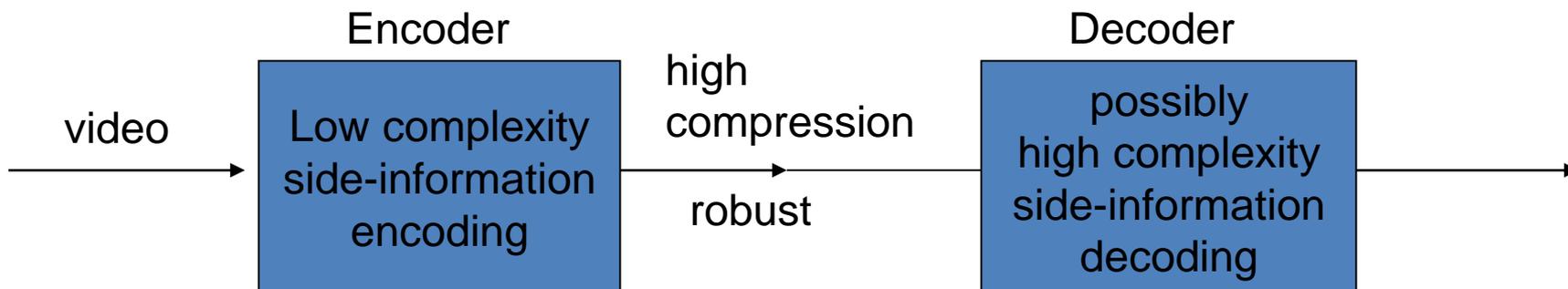
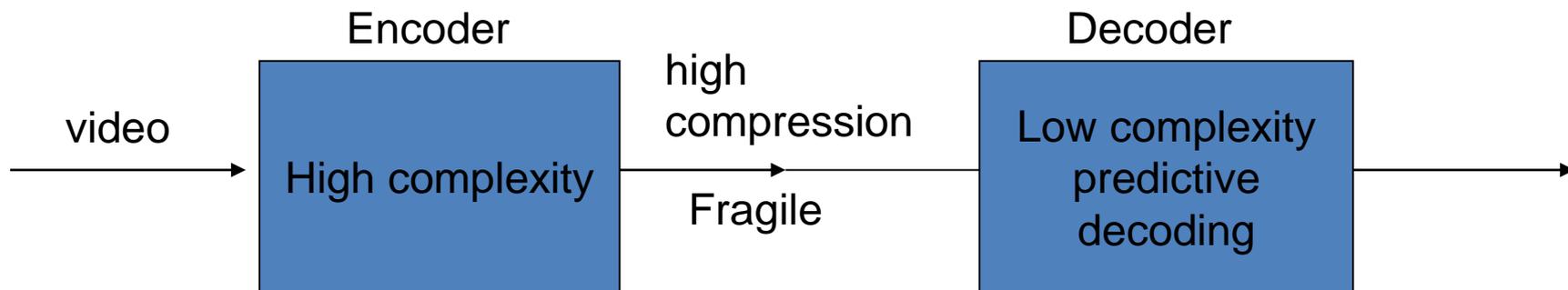
- encoder (heavy): dominate by the **motion search operation** needed to strip temporal redundancy from video frames.
- **decoder** (light): a relatively **lightweight** device operating in a “**slave**” mode to the encoder.



- fragility to synchronization or “drift” between encoder and decoder in the face of prediction mismatch e.g. due to channel loss, leading to well-known annoying motion-streak artifacts:
- The drift problem in video coding is an artifact of the predictive coding framework. When, for some reason, the frame memories at the encoder and the decoder are not identical, then the residue error is encoded at the encoder off some predictor and decoded at the decoder off some other predictor.
- Scenarios: transmission loss; unequal machine precision at the encoder and the decoder.
- Drift between the encoder and the decoder can be corrected when they are synchronized by an intra-coded frame.



A new look of the video codecs :



Characteristics:

- To have flexibility in the distribution of computational complexity between encoder and decoder without compromising compression efficiency.
- To have **inbuilt robustness** to “drift” caused by loss of synchronization between encoder and decoder.
- To have a **light yet rich encoder syntax** that can be standardized while leaving far greater room for creative growth and unencumbered (不受妨礙的) innovating within the standard than is possible today.



The investigation of new video codec discussed in the following is found on the principles of

- Distributed Source Coding (DSC)
- Channel coding
- Video Transcoding (DSC-to-HEVC)



- In **uplink-rich multimedia** application, it is desirable to share the complexity burden between encoder and decoder more equally or in any desirable ratio as demanded by a specific application scenario.
- Target : **a maximally thin encoder**
moving the expensive motion estimation component of the video codec from the encoder to the decoder without loss of compression efficiency in theory and with acceptable loss of efficiency in practice.



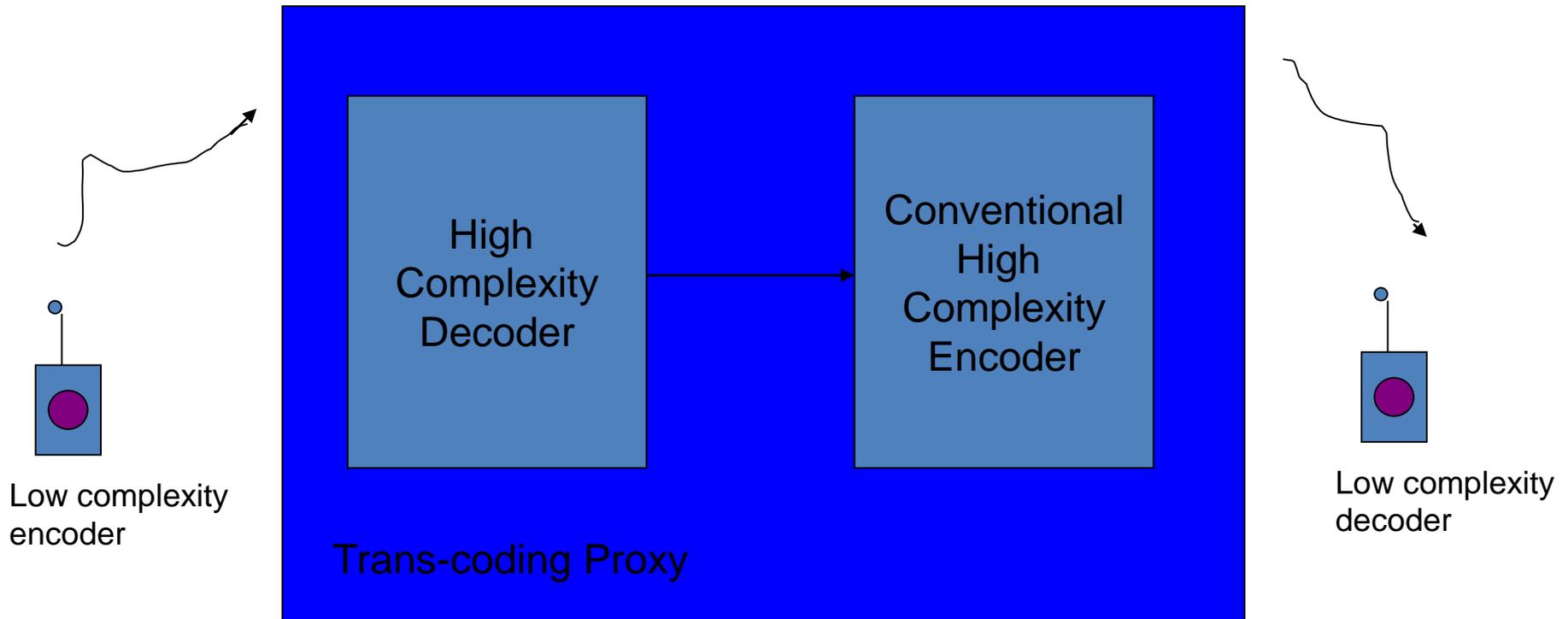
- For robustness, we have to dispense with the predictive framework and change it by a universally robust side-information based coding framework.
- Side-information source coding inherently consists of “good” channel codes, and therefore, has naturally inbuilt robustness to the synchronization loss issues.



- Traditional multimedia coding standards **constrain the bit-stream syntax at the encoder**
=> There is relatively little room for creativity within the standard.
- If all the **sophisticated signal processing task are performed at the decoder**, the syntax of the encoder becomes rich enough to accommodate a variety of decoders that are all “**Syntax-compatible**”. => this opens up the opportunity of a whole new set of creative algorithms/techniques for motion-estimation, **postprocessing** and other critical signal processing tasks.



Video codecs in a network configuration



System level diagram for a network scenario with low complexity encoding and decoding devices.



- Under this architecture, **the entire computational burden has been absorbed into the network device (such as Base Station or Cloud)**. Both the end devices, which are battery-constrained, run power efficient and light encoding and decoding algorithms. =>

Match the developing trend of

Mobile-First !



Summary

- Under the introduced construction, the relation used to **decompose** the source (such as **inter-frame-relationship**) behaves like the **correlation between the source**, while the subsets or associated **representation codewords** (**noisy versions of estimated input**) behave like the **side-information** Y .



- The **number of syndromes** determines the **bandwidth** between the **encoder and decoder** and therefore determines the **data reduction** (or **compression ratio**) of the system.
- Since the **mapping between the subsets** and **ECC** can be done in an **encryption manner**, therefore, one can compress an encrypted data.

