ECE462 – Lecture 11

Lossless Predictive Coding

- Autocorrelation Function

Given a sequence of samples

$$x[n], n = 0,1, \dots, N-1$$

The sample autocorrelation function is defined as follows

$$R_x(k) = \frac{1}{N} \sum_{n=0}^{N-k-1} x[n]x[n+k] \quad where\ 0 \le p \le N, k = 0,1,2 \dots, P$$

Lossless Predictive Coding

- Assume that x[n] is a stationary sequence with zero mean
- The $R_x(k)$ is an expression of the correlation among samples of x[n]
- If $R_x(k) = 0, k \neq 0$ and x[n] is zero mean then the samples x[n] are uncorrelated
- $R_x(0) = Power\ of\ x[n]$
- If $R_x(k) \neq 0, k \neq 0$ then there is correlation between samples of x[n] and thus, any sample of x[n] can be partially predicted from other samples of x[n].
- $R_x(0) \geq R_x(k), k \neq 0$
- If $R_x(k), k \neq 0$ is relatively significant then, sudden changes in neighboring samples of x[n] are highly non-probable.
- For real signals $R_x(-k) = R_x(k)$

Linear Prediction

Consider "predicting" or estimating the sample x[n] at time n based on a linear combination of p past samples.

$$\hat{x}[n] = \sum_{i=1}^{P} a_i x[n-i]$$

Where $\{a_i\}$, i=1,…,p are the "prediction" coefficients to be determined. One way to calculate $\{a_i\}$ is to minimize the Mean Square Error between the "true" and "predicted value", i.e.,

$$Minimize_{w.r.t.\{a_i\}} \sum_{n=1}^{N} (x[n] - \hat{x}[n])^2$$

Linear Prediction

Solution:

$$\sum_{i=1}^{P} a_i R_x(j-i) = R_x(j)\,, j = 1,2,\ldots,p$$

- In practice p is chosen arbitrarily.
- Notice that if $R_x(k) = 0, \quad k \neq 0$ then prediction is not possible
- To find $\{a_i\}$, repeat the last equation for j=1,...,p, to form a linear system of equations

Linear Prediction

$$
\underbrace{\begin{bmatrix} R_x(0) & R_x(1) & R_x(2) & \cdots & R_x(P-1) \\ R_x(1) & R_x(0) & R_x(1) & \cdots & R_x(P-2) \\ \vdots & & & & \\ R_x(P-1) & \cdots & \cdots & \cdots & R_x(0) \end{bmatrix}}_{R_x}
\underbrace{\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix}}_{a}
=
\underbrace{\begin{bmatrix} R_x(1) \\ R_x(2) \\ \vdots \\ R_x(P) \end{bmatrix}}_{= r_x}
$$

and $a = R_x^{-1} \cdot r_x$

Linear Prediction

- Ex: Let $R_x(k) = 0.5^k, k = 0,1 \dots.$ $\quad R_x(-K) = R_x(K)$

Let us choose p = 2

$$\begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.25 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.25 \end{bmatrix}$$

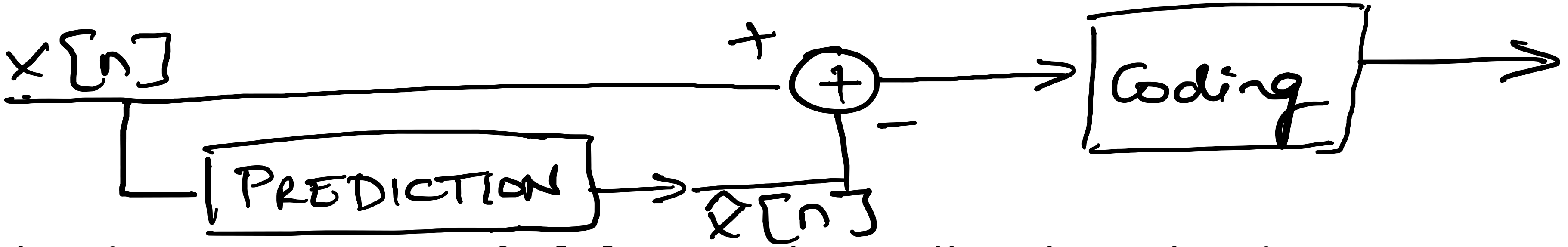
$$\Rightarrow \quad a_1 = 0.566, \quad a_2 = 0$$

Lossless Predictive Coding

Given a sequence x[n], n=0,1,…

- Predictive encoder

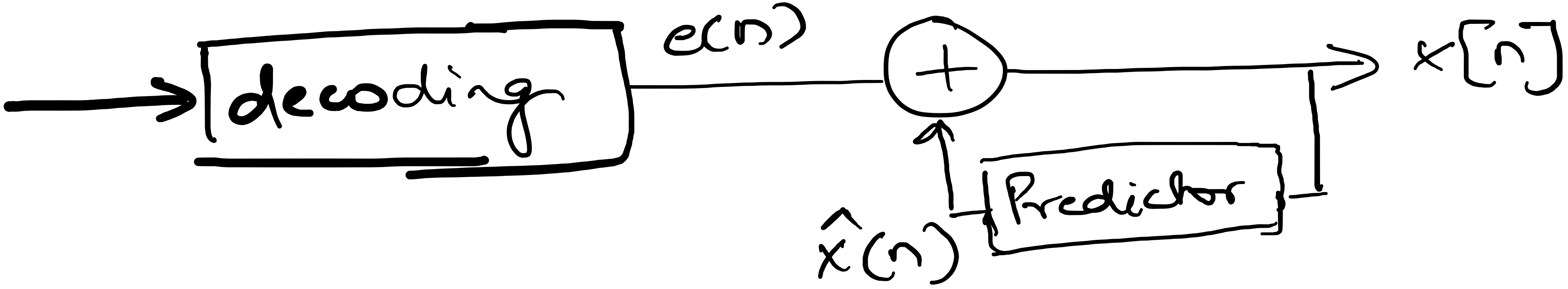
$e[n] = x[n] - \hat{x}[n]$



- The dynamic range of e[n] is much smaller than the dynamic range of x[n]. Therefore, coding can be more efficient (lower bit-rate).
- If Quantization is need prior to coding then the operation becomes lossy.

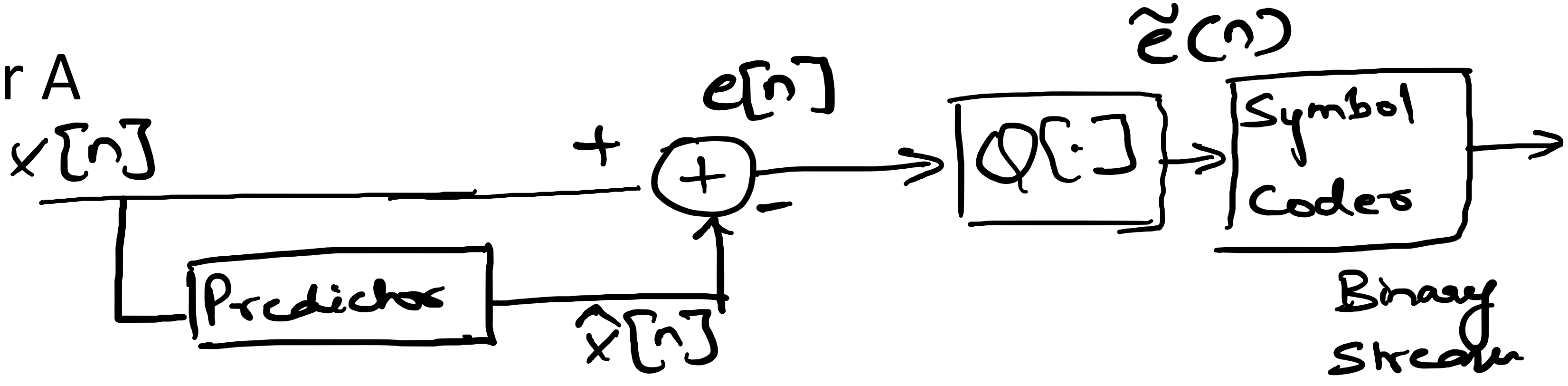
Lossless Predictive Coding

- Predictive decoder

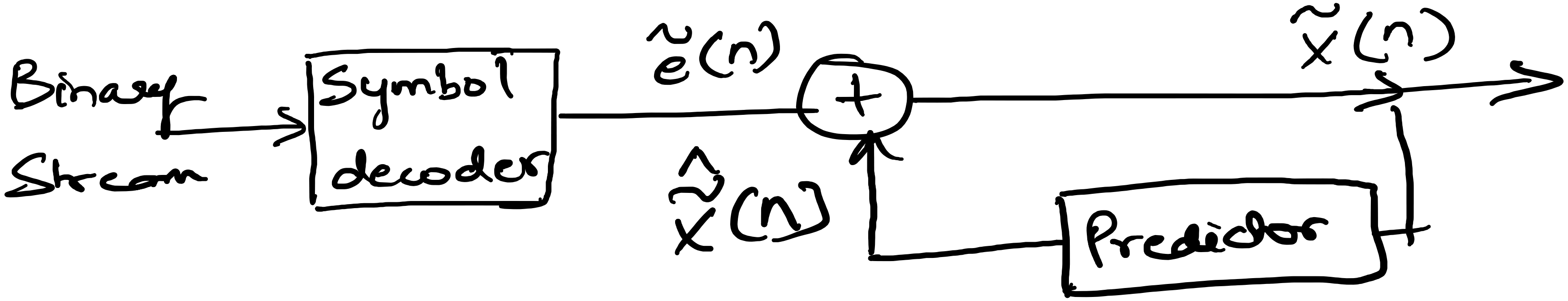


Differential pulse code modulation (DPCM)

- Encoder A



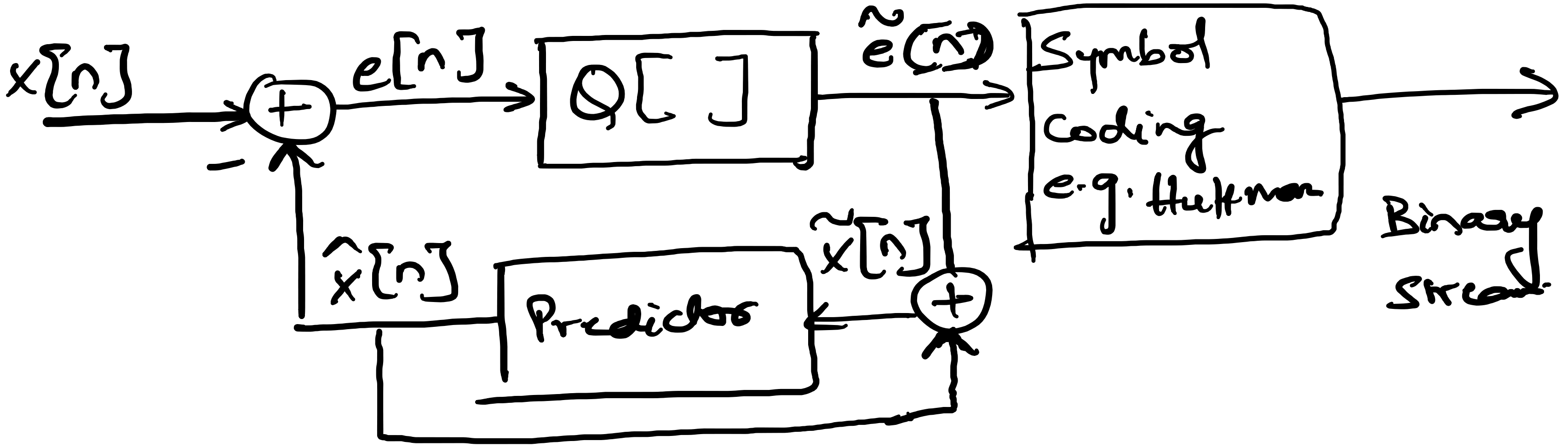
- Decoder



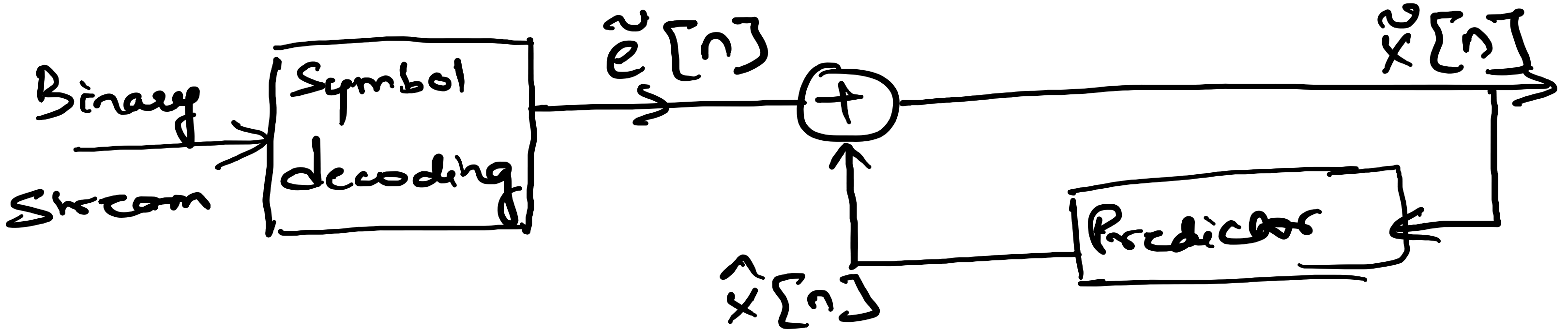
- However, this arrangement does not guarantee that encoder and decoder see the same quantization error. Actually, quantization errors propagate at decoder.

Differential pulse code modulation (DPCM)

- Encoder B



- Decoder



Note: Quantization error $q[n] = e[n] - \tilde{e}[n] = x[n] - \tilde{x}[n]$

Differential pulse code modulation (DPCM)

- Using feedback prediction at the encoder side prevents propagation of the quantization error at the decoder side (Both sides see the same quantization error)
- Example:

Suppose we use DPCM where:

Prediction: $\hat{x}[n] = x[n-1], n = 0,1,2 \ldots$

Quantization: $Q[x] = truncation[x]$

Input signal:

$n =$	0	1	2	3	4	5	6	7
$x[n]:$	20,	38.2,	56.4,	74.6,	92.7,	110.9,	129,	147.2

Differential pulse code modulation (DPCM)

- Calculate the sequence $\tilde{e}[n]$ as well the reconstructed $\tilde{x}[n]$ for the two cases of using Encoder A (forward prediction) and Encoder B (feedback prediction).
- Observe and comment on the effect of quantization noise in the two cases.

Answer

ENCODER A

n	0	1	2	3	4	5	6	7
$x[n]$	20	38.2	56.4	74.6	92.7	110.9	129	147.2
$\hat{x}[n]=x[n-1]$	0	20	38.2	56.4	74.6	92.7	110.9	129
$e[n]=x[n]-x[n-1]$	20	18.2	18.2	18.2	18.1	18.2	18.1	18.2
$\tilde{e}[n]=Q(e[n])$	20	18	18	18	18	18	18	18

Differential pulse code modulation (DPCM)

- Decoder A

$\tilde{x}[n] = \tilde{e}[n] + \tilde{x}[n-1]$: 20 38 56 74 92 110 128 146

Reconstruction: 0 0.2 0.4 0.6 0.7 0.9 1 1.2

error $(x[n] - \tilde{x}[n])$

- So the error increases due to the propagation of quantization errors.

- Encoder B

n	0	1	2	3	4	5	6	7
$x[n]$	20	38.2	56.4	74.6	92.7	110.9	129	147.2
$\hat{x}[n] = \tilde{e}[n-1] + \hat{x}[n-1]$	0	20	38	56	74	92	110	129
$e[n] = x[n] - \hat{x}[n]$	20	18.2	18.4	18.6	18.7	18.9	19	18.2
$\tilde{e}[n] = Q[e[n]]$	20	18	18	18	18	18	19	18

- Decoder B

$\tilde{x}[n] = \tilde{e}[n] + \tilde{x}[n-1]$:	20	38	56	74	92	110	129	147
Reconstruction error $x[n] - \tilde{x}[n]$:	0	0.2	0.4	0.6	0.7	0.9	0	0.2

- So clearly in the second case where feedback prediction is employed during encoding and decoding the effect of quantization errors is less.