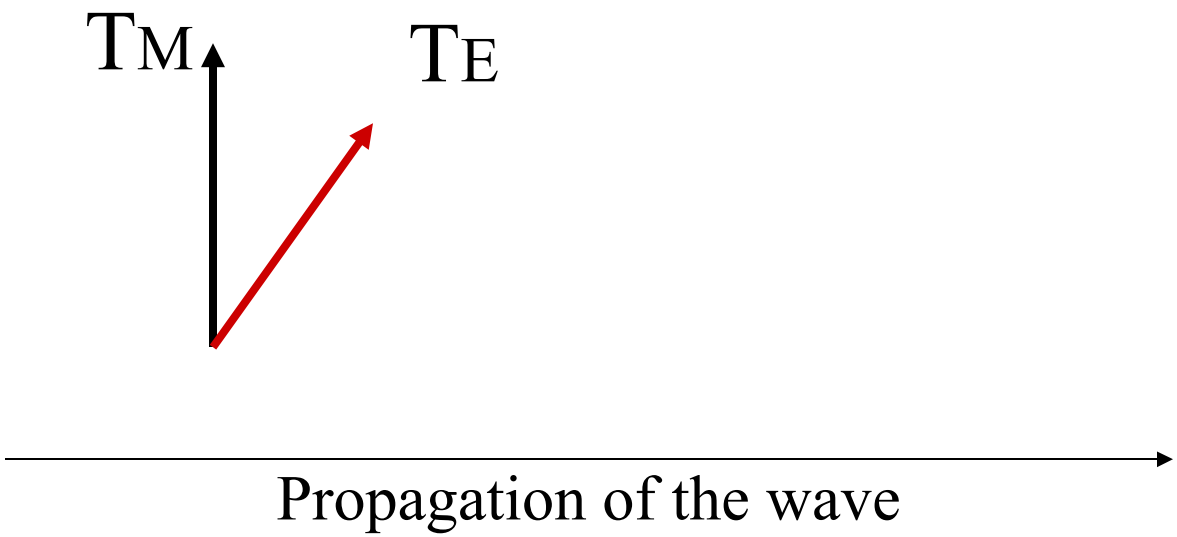
Compression and Separation

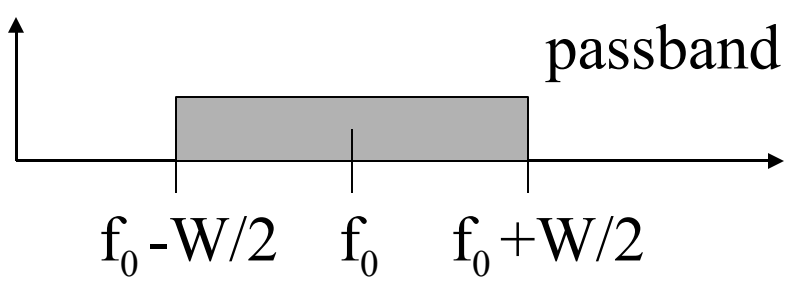
Signals

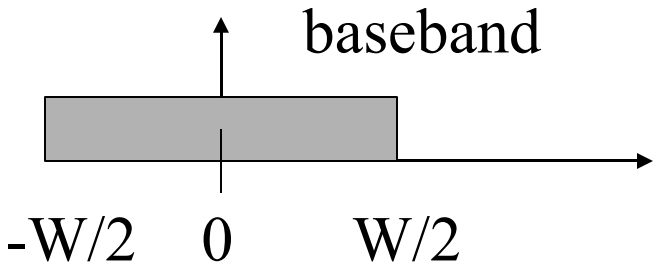
- Channel has a physical signal traversing a physical medium
- Electromagnetic signal
- Polarization of the wave



Signals

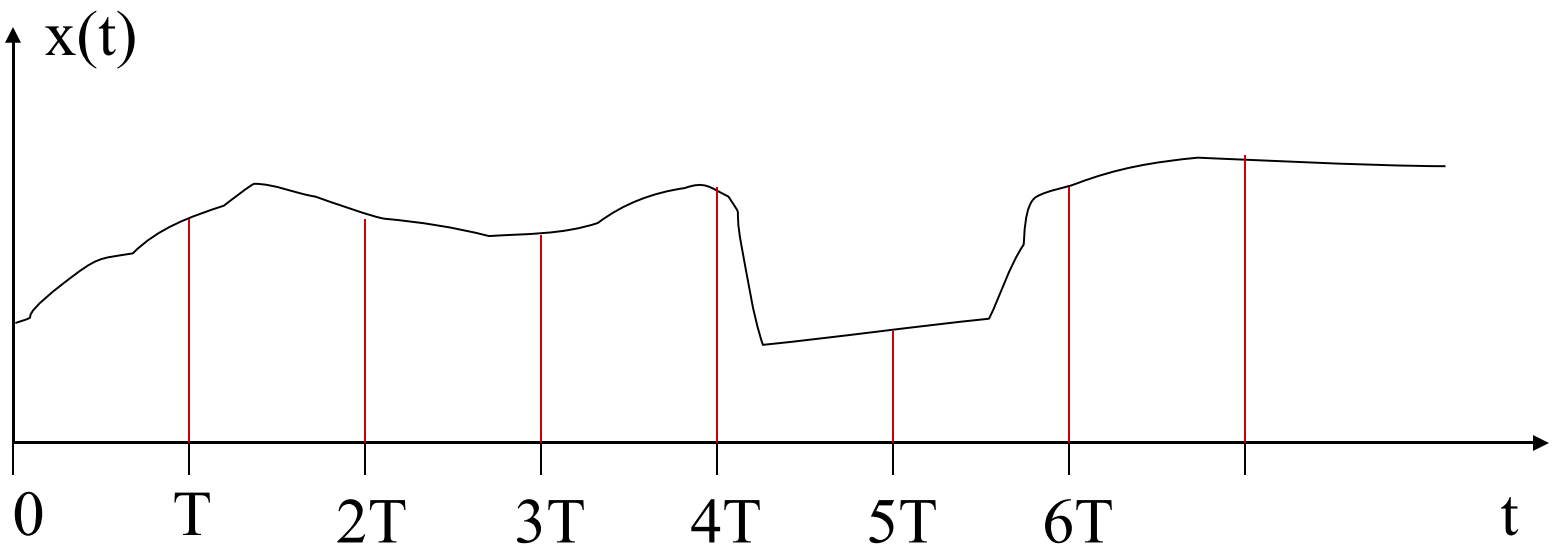
- What do we mean by frequency?
- We can express a signal by a superposition of sinusoids, each of them with its own frequency
- A continuous signal is bandlimited to a bandwidth W at carrier frequency f_0 if it can be expressed in terms of sinusoids in a range of frequencies of width W around a frequency f_0
- If we multiply the signal by a frequency shift, then we operate around 0, the baseband representation





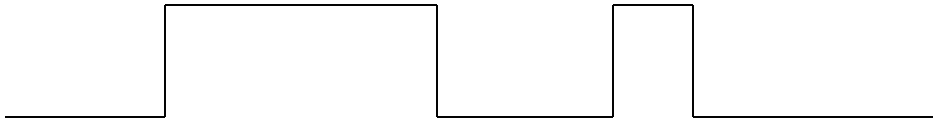
Nyquist sampling theorem

- We can wholly reconstruct a signal from its periodic samples as long as those samples are within 1/W of each other
- Continuous signal x(t), discrete time signal is x[n] = x(n T) where T = 1/W

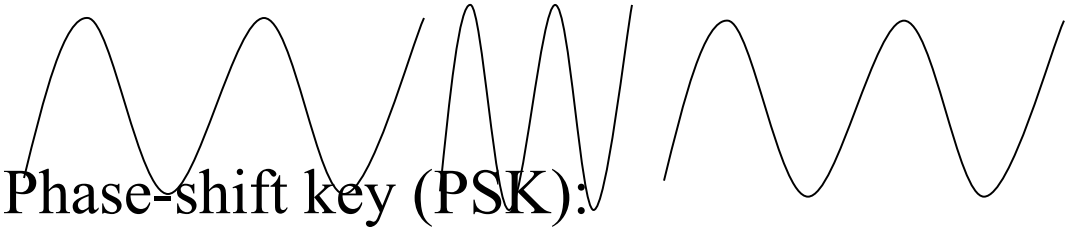


Modulation: creating signals

- How we generate signals, by varying amplitude, phase, frequency
- On-off keyed (OOK): send 1, 0, a special case of amplitude modulation

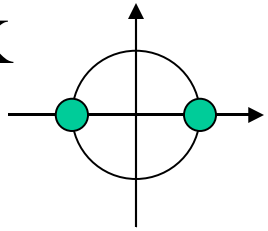


- Frequency-shift keyed (FSK), change frequencies:

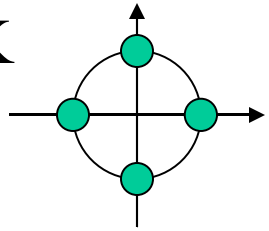


- Phase-shift key (PSK):

2 PSK



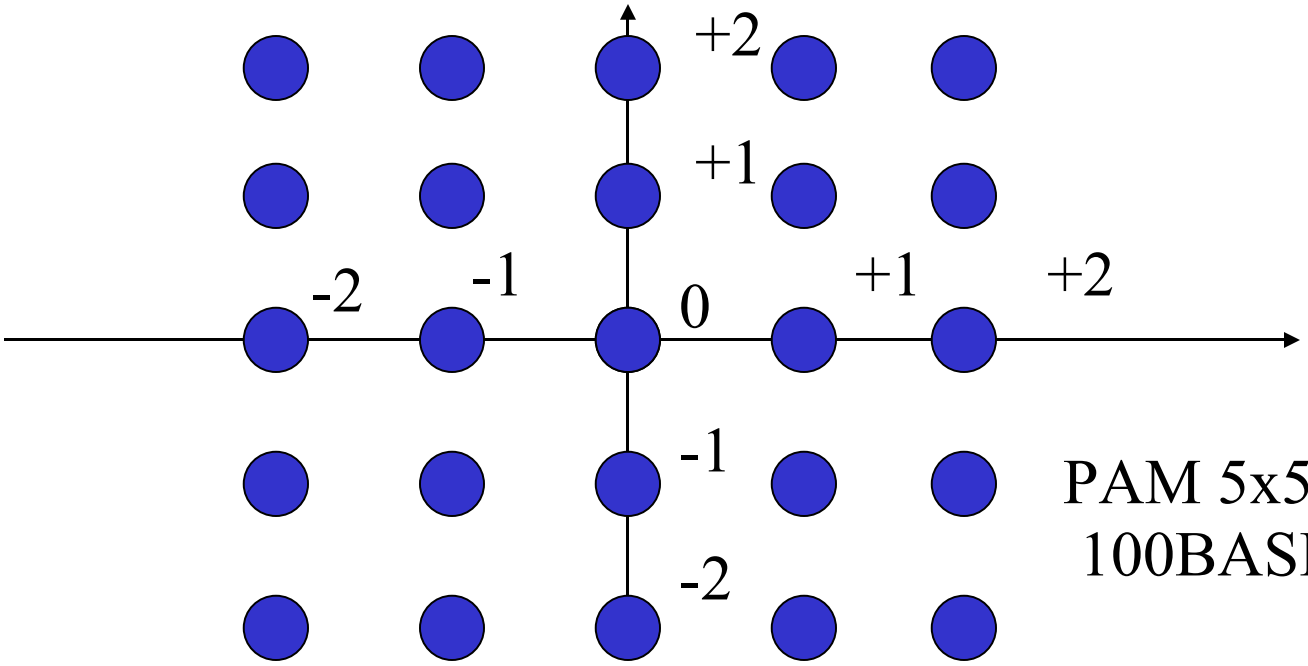
4 PSK



Modulation

- Pulse position modulation (PPM)

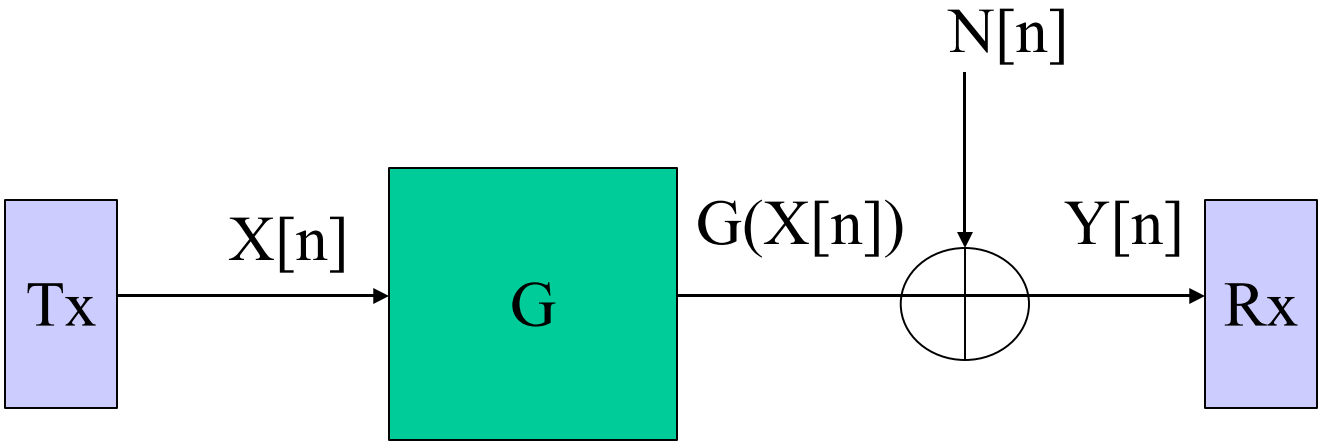
- Combining phase and amplitude, for instance:



PAM 5x5 symbol code for 100BASE-T2 PHY

Channels

- Canonical channel model: signal goes into channel, there is a non-additive effect and addition of noise



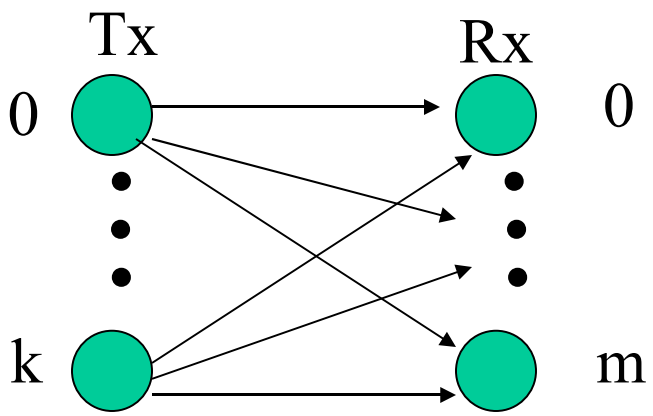
- Very commonly, G is a multiplicative effect:
  - $Y[n] = g\, X[n] + N[n]$
  - $Y[n] = g_{-1}\, X[n-1] + g_0\, X[n] + N[n]$
    INTERSYMBOL-INTERFERENCE (ISI)

Effect of the channel

- If we put in a certain string of input signals X[1], X[2], …, then we have a certain probability of having a certain output string Y[1], Y[2], …
- The net effect is what is the probability of having a certain output when a given input is given

Channel model

- The channel is described by an input alphabet, an output alphabet, a set of probabilities on the input alphabet and a set of channel transitions, for a memoryless channel (on every bit the channel behaves independently of other times):



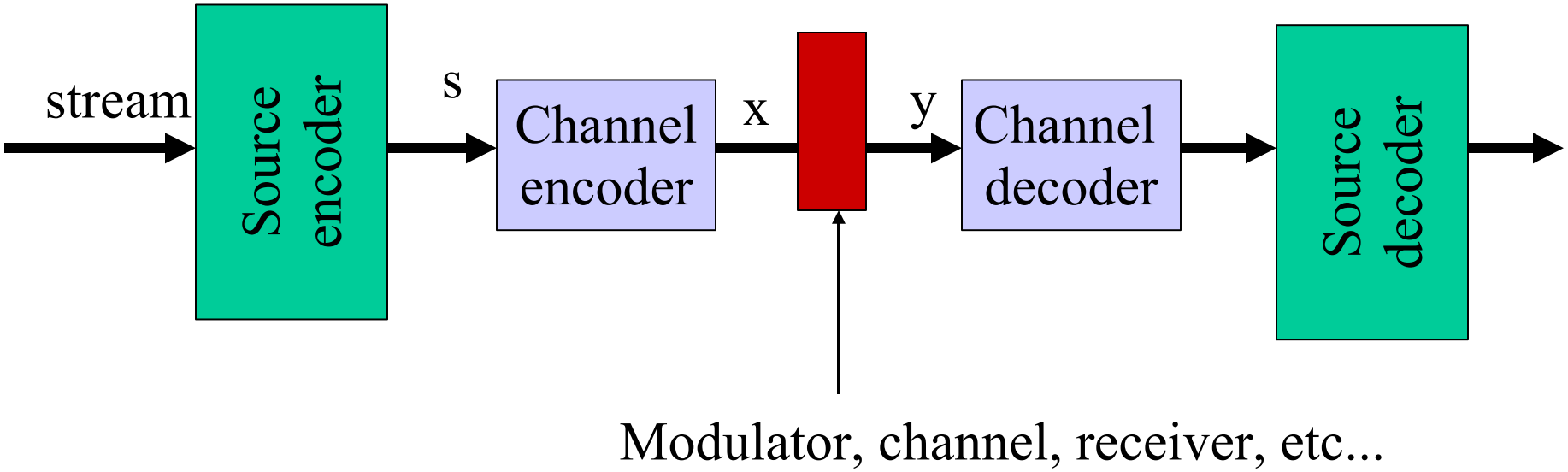
- Channel with input alphabet of size k+1 and output alphabet of size m+1
- p(y | x) is transition probability, probability of getting output y for input x

How do we determine the probabilities?

- We want to get to $p(x \mid y)$
- Use Bayes's rule: $p(x \mid y) = p(x, y) / p(y)$
- Also $p(x, y) = p(y \mid x)\, p(x)$
- $p(y) = \sum_i p(y \mid x_i)\, p(x_i)$
- For the case where all the inputs have the same probability, then $p(x \mid y) = 1/n\, p(y \mid x) / \sum_i p(y \mid x_i)$
- **How can we try to make up for the effect of the channel?**

When do we use codes

- Two different types of codes:
  - source codes: compression
  - channel codes: error-correction
  - Source-channel separation theorem says the two can be done independently for a large family of channels



Source codes for random variables

Notation: The set of all strings over a finite alphabet $\mathcal{D}$ is denoted by $\mathcal{D}^*$. W.l.o.g. assume $\mathcal{D} = 0, 1, \ldots, D-1$ where $D = |\mathcal{D}|$.

Definition: A source code for a random variable X is a map

$$\begin{aligned} C: \qquad & \mathcal{X} \mapsto \mathcal{D}^* \\ & x \to C(x) \end{aligned}$$

where $C(x)$ is the codeword associated with x

$l(x)$ is the length of $C(x)$

The length of a code C is

$$L(C) = E_X[l(X)]$$

Source codes for random variables

A code is instantaneous (or prefix code) iff no codeword of C is a prefix of any other codeword C

Visually: Construct a tree whose leaves are codewords

Kraft inequality

Any instantaneous code C with code lengths $l_1, l_2, \ldots, l_m$ must satisfy

$$\sum_{i=1}^{m} D^{-l_i} \leq 1$$

Conversely, given lengths $l_1, l_2, \ldots, l_m$ that satisfy the above inequality, there exists an instantaneous code with these codeword lengths

Proof: Construct a D-ary tree T (codewords are leaves)

Extend tree T to D-ary tree T' with depth l_{MAX}, total number of leaves is $D^{l_{MAX}}$

Kraft inequality

Each leaf of T' is a descendant of at most one leaf of T

Leaf in T corresponding to codeword $C(i)$ has exactly $D^{l_{MAX}-l_i}$ descendants in T' (1 if $l_i = l_{MAX}$)

Summing over all leaves of T gives

$$\sum_{i=1}^{m} D^{l_{MAX}-l_i} \leq D^{l_{MAX}}$$

$$\Rightarrow \sum_{i=1}^{m} D^{-l_i} \leq 1$$

Kraft inequality

Given lengths $l_1, l_2, \ldots, l_m$ satisfying Kraft's inequality, we can construct a tree by assigning $C(i)$ to first available node at depth l_i

Optimal codes

Optimal code is defined as code with smallest possible $L(C)$ with respect to P_X

Optimization:

minimize $\sum_{x \in \mathcal{X}} P_X(x) l(x)$

subject to $\sum_{x \in \mathcal{X}} D^{-l(x)} \leq 1$

and $l(x)$s are integers

Optimal codes

Let us relax the second constraint and replace the first with equality to obtain a lower bound

$$J = \sum_{x \in \mathcal{X}} P_X(x) l(x) + \lambda \left(\sum_{x \in \mathcal{X}} D^{-l(x)} - 1 \right)$$

Use Lagrange multipliers and set $\frac{\partial J}{\partial l(i)} = 0$

$$P_X(i) - \lambda \log(D) D^{-l(i)} = 0$$

equivalently $D^{-l(i)} = \frac{P_X(i)}{\lambda \log(D)}$

Using Kraft inequality (now relaxed to equality) yields

$$1 = \sum_{i \in \mathcal{X}} D^{-l(x)} = \sum_{i \in \mathcal{X}} \frac{P_X(i)}{\lambda \log(D)}$$

So $\lambda = \frac{1}{\log(D)}$, yielding $l(i) = -\log_D(P_X(i))$

Optimal codes

Thus a bound on the optimal code length is

$$-\sum_{i \in \mathcal{X}} P_X(i) \log_D(P_X(i)) = H_D(X)$$

This is lower bound, equality holds iff P_X is D-adic, $P_X(i) = D^{-l(i)}$ for integer $l(i)$

Optimal codes

The optimal codelength L^* satisfies

$$H_D(X) \leq L^* \leq H_D(X) + 1$$

Upper bound: Take $l(i) = \lceil \log_D(P_X(i)) \rceil$

$$\sum_{i \in \mathcal{X}} D^{\lceil -\log_D(P_X(i)) \rceil} \leq \sum P_X(i) = 1$$

Thus these lengths satisfy Kraft's inequality and we can create a prefix-free code with these lengths

$$\begin{aligned} L^* &\leq \sum_{i \in \mathcal{X}} P_X(i) \lceil -\log_D(P_X(i)) \rceil \\ &\leq \sum_{i \in \mathcal{X}} P_X(i)(-\log_D(P_X(i)) + 1) \\ &= H_D(X) + 1 \end{aligned}$$

We call these types of codes Shannon codes

Optimal codes

Is this as tight as it gets?

Consider coding several symbols together

$C : \mathcal{X}^n \mapsto \mathcal{D}^*$

Expected codeword length is $\sum_{\underline{x}^n \in \mathcal{X}^n} P_{\underline{X}^n}(\underline{x}^n) l(\underline{x}^n)$

Optimum satisfies $H_D(\underline{X}^n) \leq L^* \leq H_D(\underline{X}^n) + 1$

If $\underline{X}^n$ is an i.i.d. sequence, then

$$H_D(\underline{X}^n) = nH_D(X)$$

So per symbol codeword length is $H_D(X) \leq \frac{L^*}{n} \leq H_D(X) + \frac{1}{n}$

Hence $\lim_{n\to\infty} \frac{L^*}{n} = H_D(X)$