Digital Communication Systems

ECS 452

Asst. Prof. Dr. Prapun Suksompong

prapun@siit.tu.ac.th

4. Mutual Information and Channel Capacity

Office Hours:

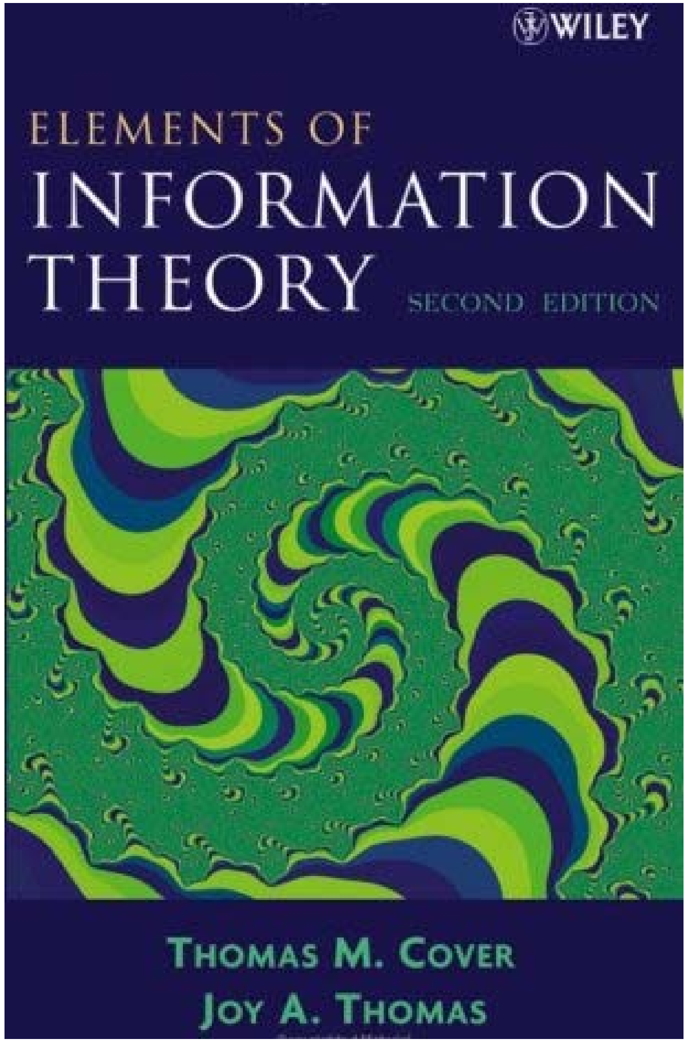
Check Google Calendar on the course website.

Dr.Prapun's Office:

6th floor of Sirindhralai building, BKD

Reference for this chapter

- Elements of Information Theory
- By Thomas M. **Cover** and Joy A. **Thomas**
- 2nd Edition (Wiley)
- Chapters 2, 7, and 8
- 1st Edition available at SIIT library: Q360 C68 1991



Recall: Entropy

4.29. Reminder:

(a) Some definitions involving entropy

(i) Binary entropy function: $h(p) = -p \log_2 p - (1-p) \log_2 (1-p)$

(ii) $H(X) = -\sum_x p(x) \log_2 p(x)$

(iii) $H(\underline{\mathbf{p}}) = -\sum_i p_i \log_2 (p_i)$

(b) A key entropy property that will be used frequently in this section is that for any random variable X,

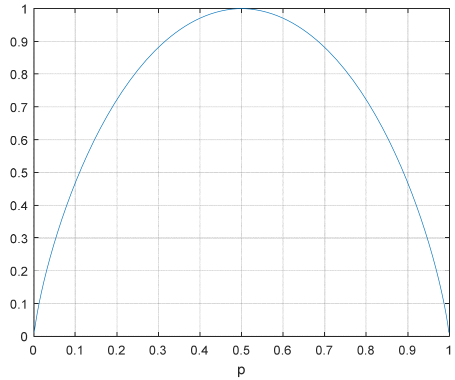
$$H(X) \leq \log_2 |\mathcal{X}| \text{ with equality iff } X \text{ is uniform.}$$

[Page 70]

Recall: Entropy

- **Entropy** measures the amount of uncertainty (randomness) in a RV.
- Three formulas for calculating entropy:
  - [Defn 2.41] Given a pmf $p_X(x)$ of a RV X,
    - $\boldsymbol{H(X)} \equiv -\sum_x p_X(x) \log_2 p_X(x).$ Set $0\log_2 0 = 0$.
  - [2.44] Given a probability vector $\underline{\mathbf{p}}$,
    - $\boldsymbol{H(\underline{\mathbf{p}})} \equiv -\sum_i p_i \log_2 p_i.$
  - [Defn 2.47] Given a number $p \in [0,1]$,
    - $\boldsymbol{H(p)} \equiv \boldsymbol{h_b(p)} = -p\log_2 p - (1-p)\log_2(1-p)$

binary entropy function



- [2.56] Operational meaning: Entropy of a random variable is the average length of its shortest description.

Recall: Entropy

- Important Bounds

$$\underset{\text{deterministic}}{0} \leq H(X) \leq \underset{\text{uniform}}{\log_2 |S_X|}$$

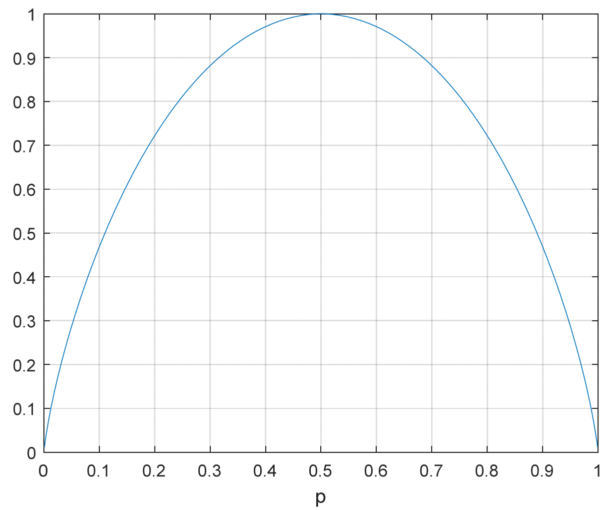
- The entropy of a uniform (discrete) random variable:

$$H(X) = \log_2 |S_X|$$

- The entropy of a Bernoulli random variable:

$$\boldsymbol{H(p) \equiv h_b(p)} = -p\log_2 p - (1-p)\log_2(1-p)$$

  - **binary entropy function**



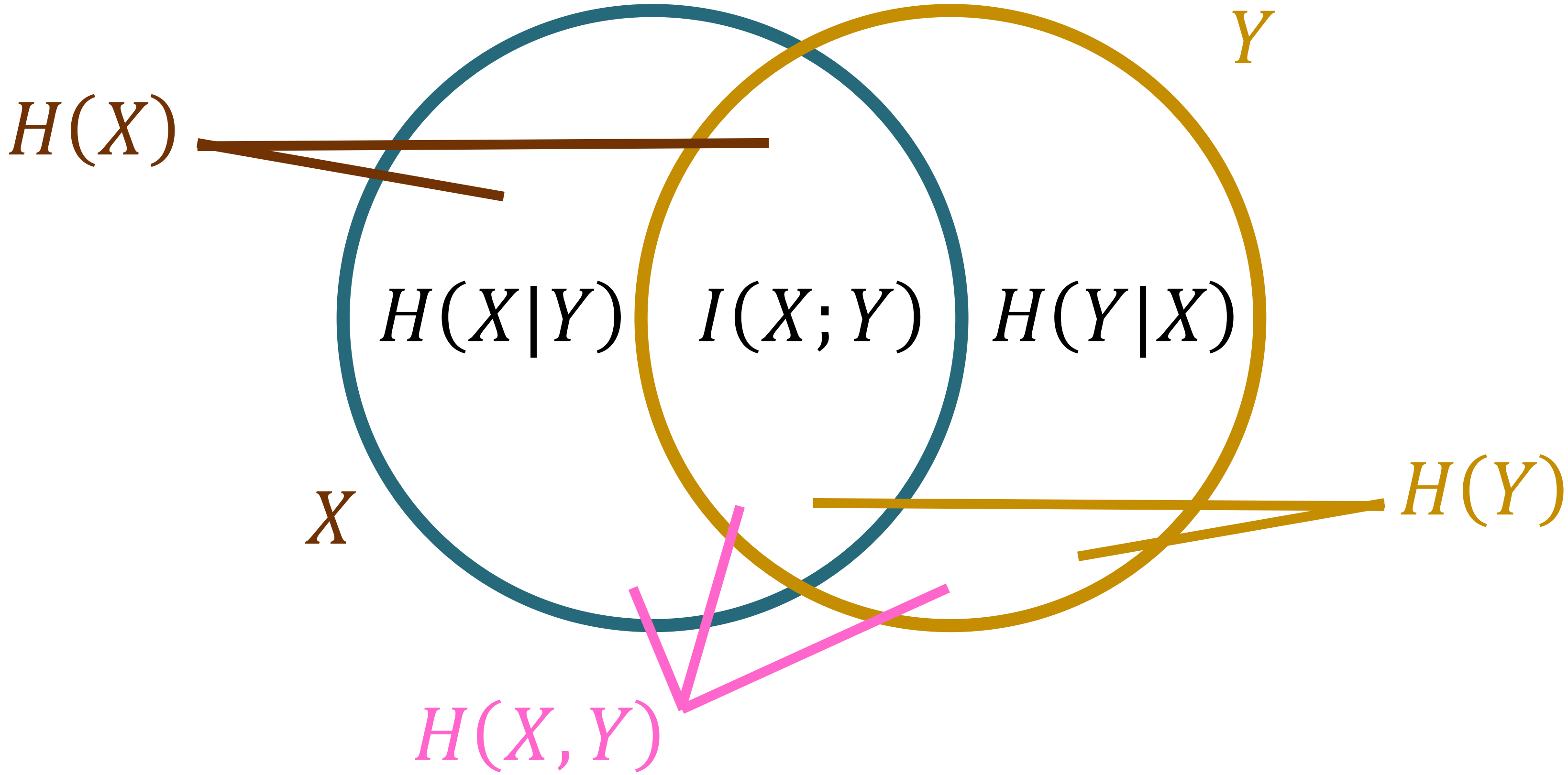
Digital Communication Systems
ECS 452

Asst. Prof. Dr. Prapun Suksompong
prapun@siit.tu.ac.th

Information-Theoretic Quantities

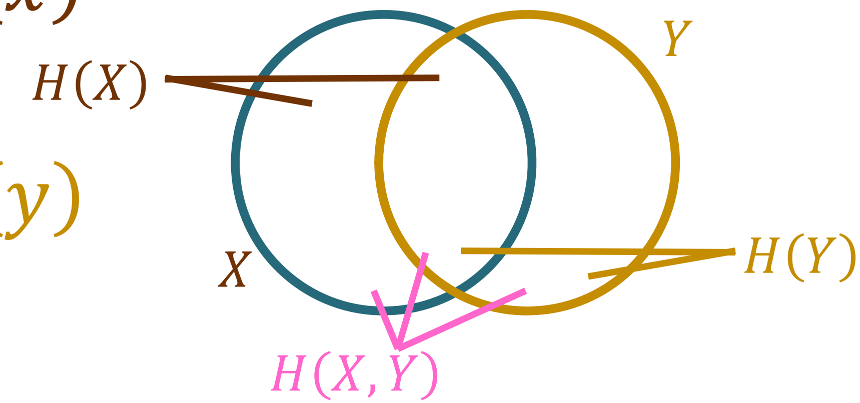
Information-Theoretic Quantities

Information Diagram



Entropy and Joint Entropy

- **Entropy**
  - $H(X) = -\sum_x p(x) \log_2 p(x)$
    - Amount of randomness in X
  - $H(Y) = -\sum_y q(y) \log_2 q(y)$
    - Amount of randomness in Y



- **Joint Entropy**
  - $H(X,Y) = -\sum_{(x,y)} p(x,y) \log_2 p(x,y)$
    - Amount of (combined) randomness in (X, Y) pair
  - In general, $H(X,Y) \neq H(X) + H(Y)$
    - There might be some shared randomness between X and Y.

Conditional Entropies

Amount of randomness in Y

$$H(Y) \equiv -\sum_{y \in \mathcal{Y}} q(y) \log_2 \overbrace{q(y)}^{P[Y=y]} \equiv H(\underline{\mathbf{q}})$$

Amount of randomness still remained in Y when we know that $X = x$.

$$\overbrace{H(Y|X=x)}^{\text{given a particular value } x} \equiv H(Y|x) \equiv -\sum_{y \in \mathcal{Y}} Q(y|x) \log_2 \overbrace{Q(y|x)}^{P[Y=y|X=x]}$$

Apply the entropy calculation to a row from the $\mathbf{Q}$ matrix

The **average** amount of randomness still remained in Y when we know X

$$H(Y|X) \equiv \overbrace{\sum_{x \in \mathcal{X}} p(x) H(Y|x)}^{\textbf{average} \text{ of } H(Y|x)}$$

$$= H(X,Y) - H(X)$$

Conditional Entropies

Amount of randomness in Y

$$H(Y) \equiv -\sum_{y \in \mathcal{Y}} q(y) \log_2 \overbrace{q(y)}^{P[Y=y]} \equiv H\left(\underline{\mathbf{q}}\right)$$

Amount of randomness still remained in Y when we know that $X = x$.

$$H(Y|\overbrace{X=x}^{\text{given a particular value } x}) \equiv H(Y|x) \equiv -\sum_{y \in \mathcal{Y}} Q(y|x) \log_2 \overbrace{Q(y|x)}^{P[Y=y|X=x]}$$

Apply the entropy calculation to a row from the $\mathbf{Q}$ matrix

The **average** amount of randomness still remained in Y when we know X

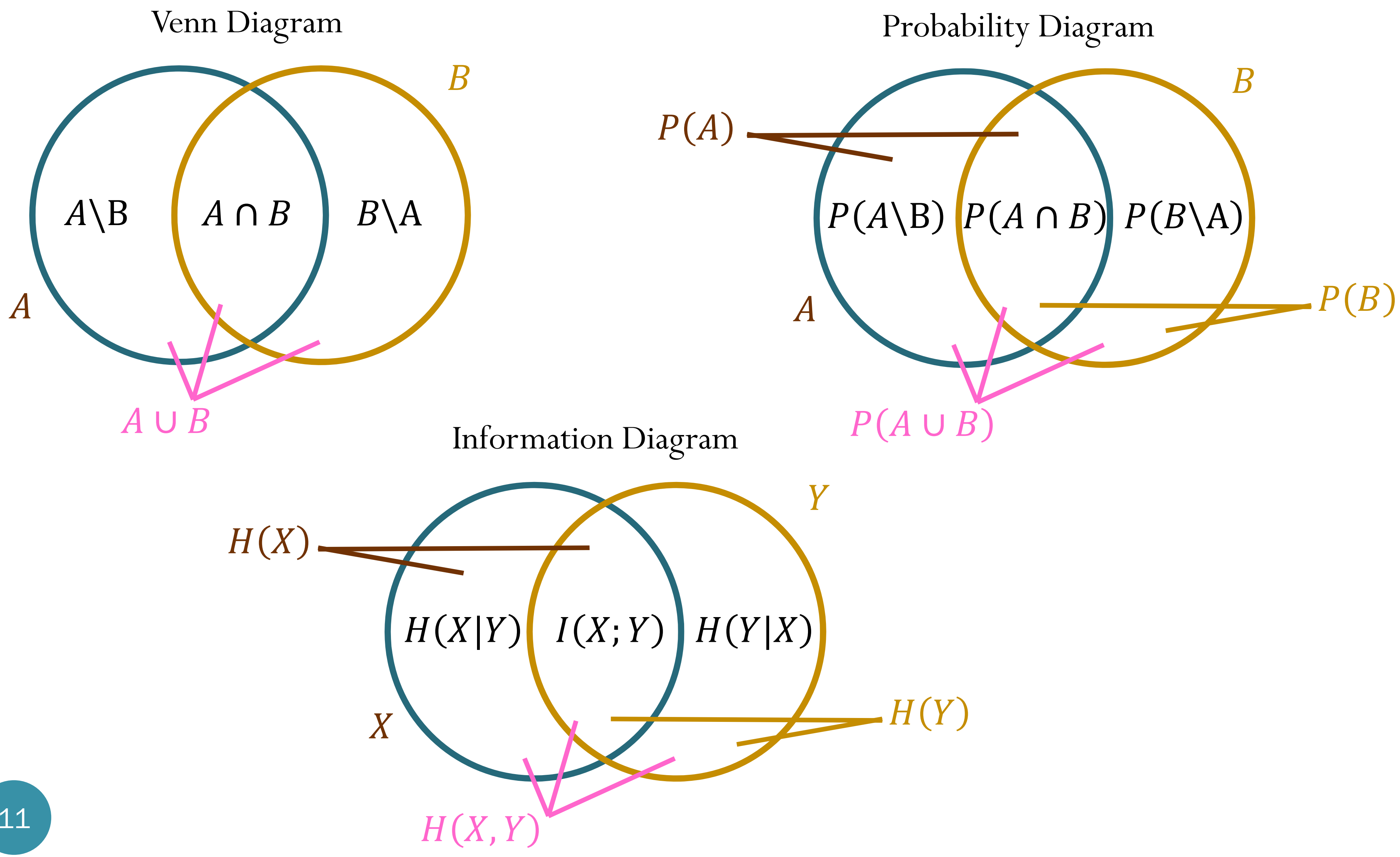
$$H(Y|X) \equiv \underbrace{\sum_{x \in \mathcal{X}} p(x) H(Y|x)}_{}$$

(**average** of $H(Y|x)$)

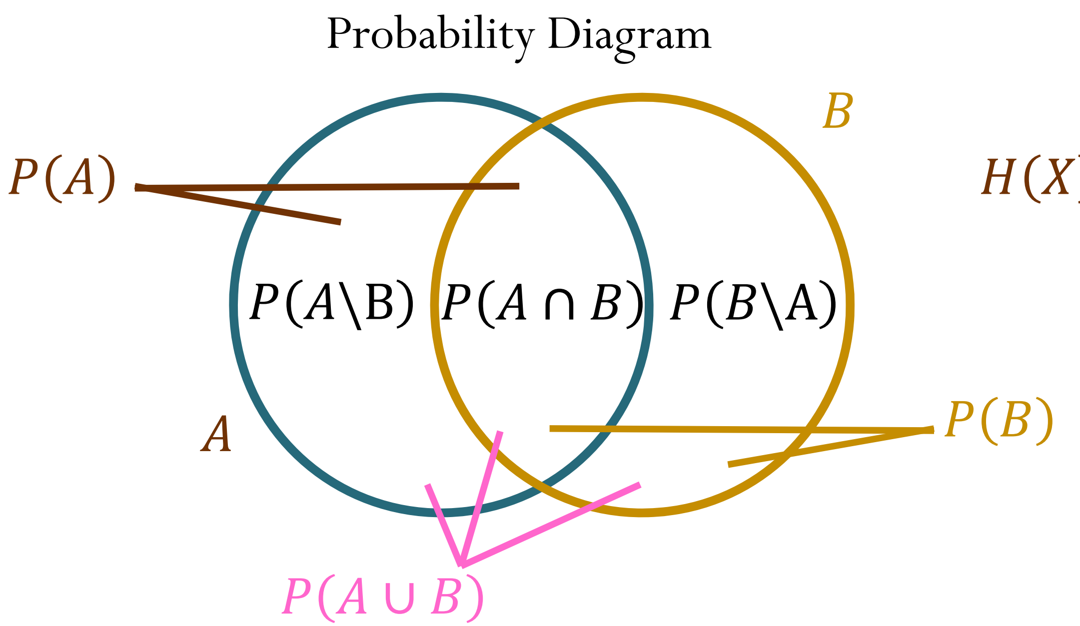
$$= H(X,Y) - H(X)$$

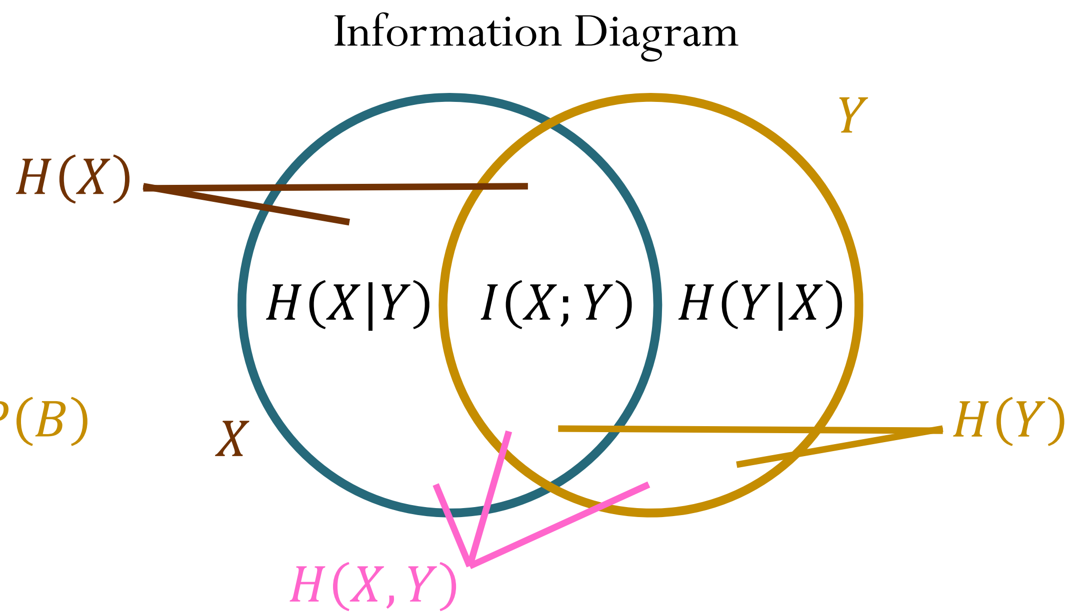
$$= H(Y) - I(X;Y)$$

Diagrams [Figure 16]



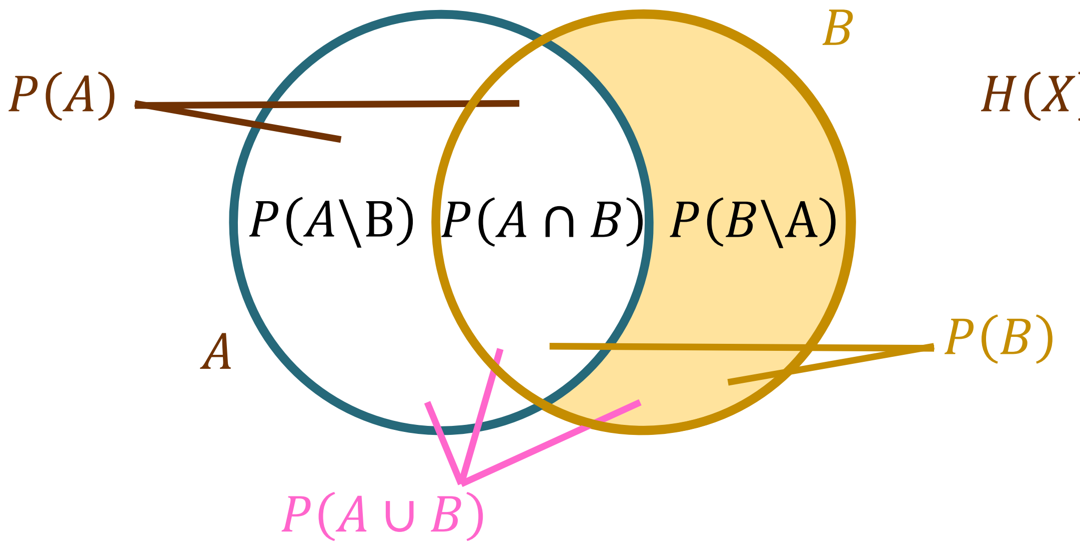
Diagrams [Figure 16]



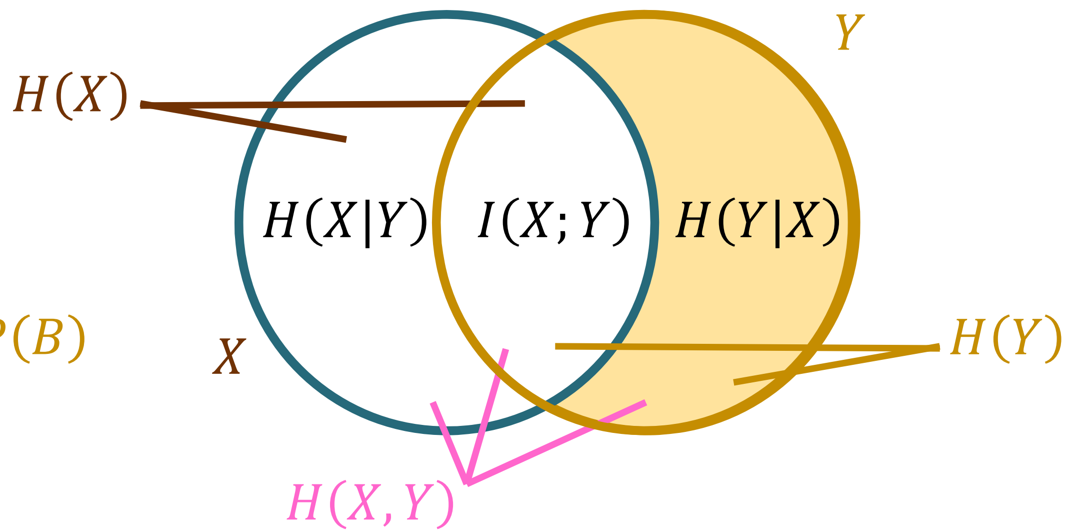


Diagrams

Probability Diagram



Information Diagram



$$P(B \setminus A) = P(A \cup B) - P(A)$$

$$H(Y|X) = H(X,Y) - H(X)$$

Digital Communication Systems ECS 452

Asst. Prof. Dr. Prapun Suksompong

prapun@siit.tu.ac.th

Operational Channel Capacity

Channel Capacity

[Section 4.2]

"**Operational**": max rate at which **reliable** communication is possible

→ Arbitrarily small error probability can be achieved.

Channel Capacity

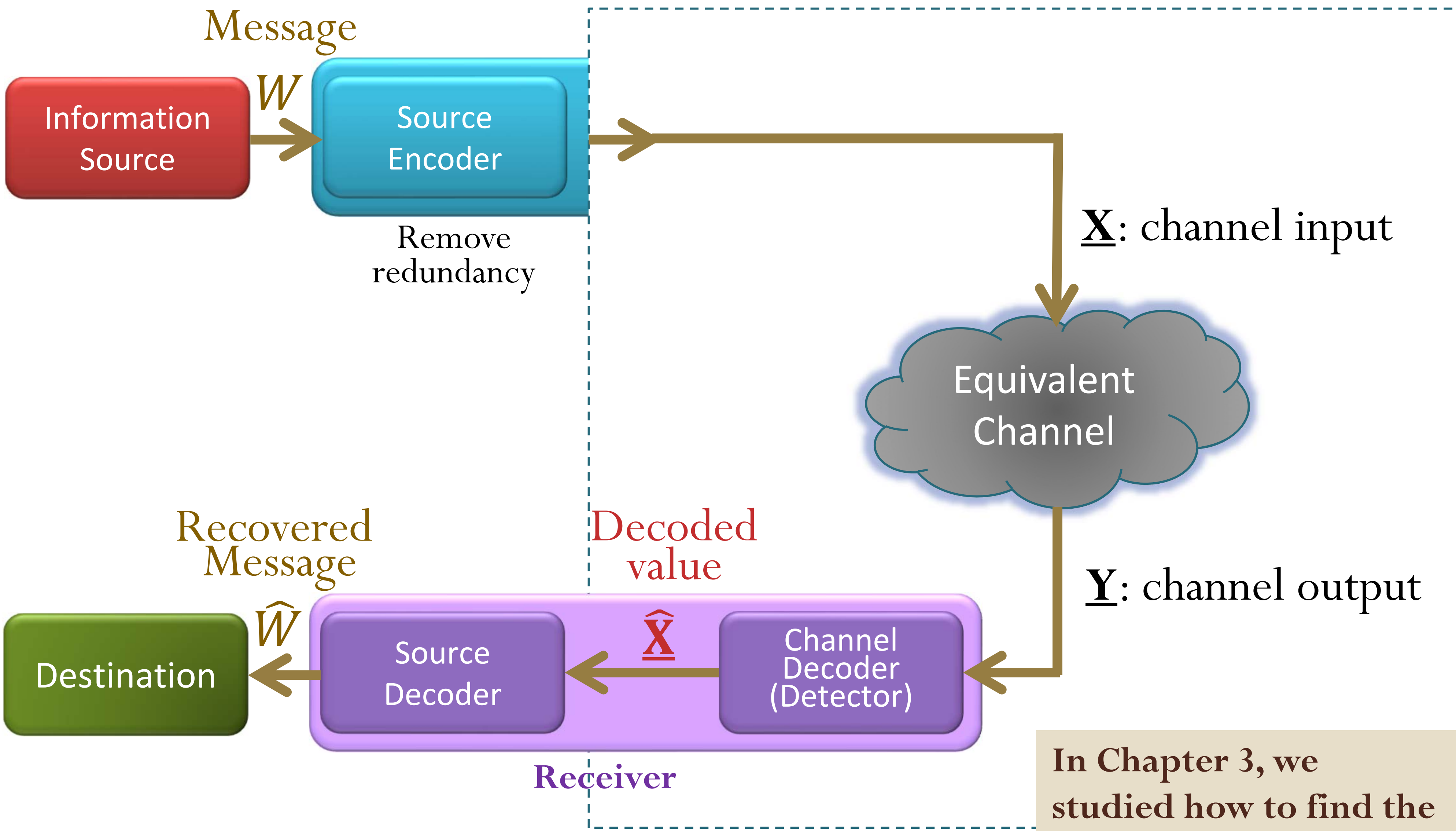
"**Information**": $\max_{\underline{\mathbf{p}}} I(X;Y)$ [bpcu]

[Section 4.3]

Shannon [1948] shows that these two quantities are actually the same.

Review

System Model for Section 3.4



In Chapter 3, we studied how to find the optimal decoder.

Review

Some results from Section 3.3-3.4

Example 3.66.

(1) MAP decoder is optimal. (It minimizes $P(\mathcal{E})$).

(2) ML decoder is suboptimal. However, it can be optimal (same $P(\mathcal{E})$ as the MAP decoder) when the codewords are equally-likely.

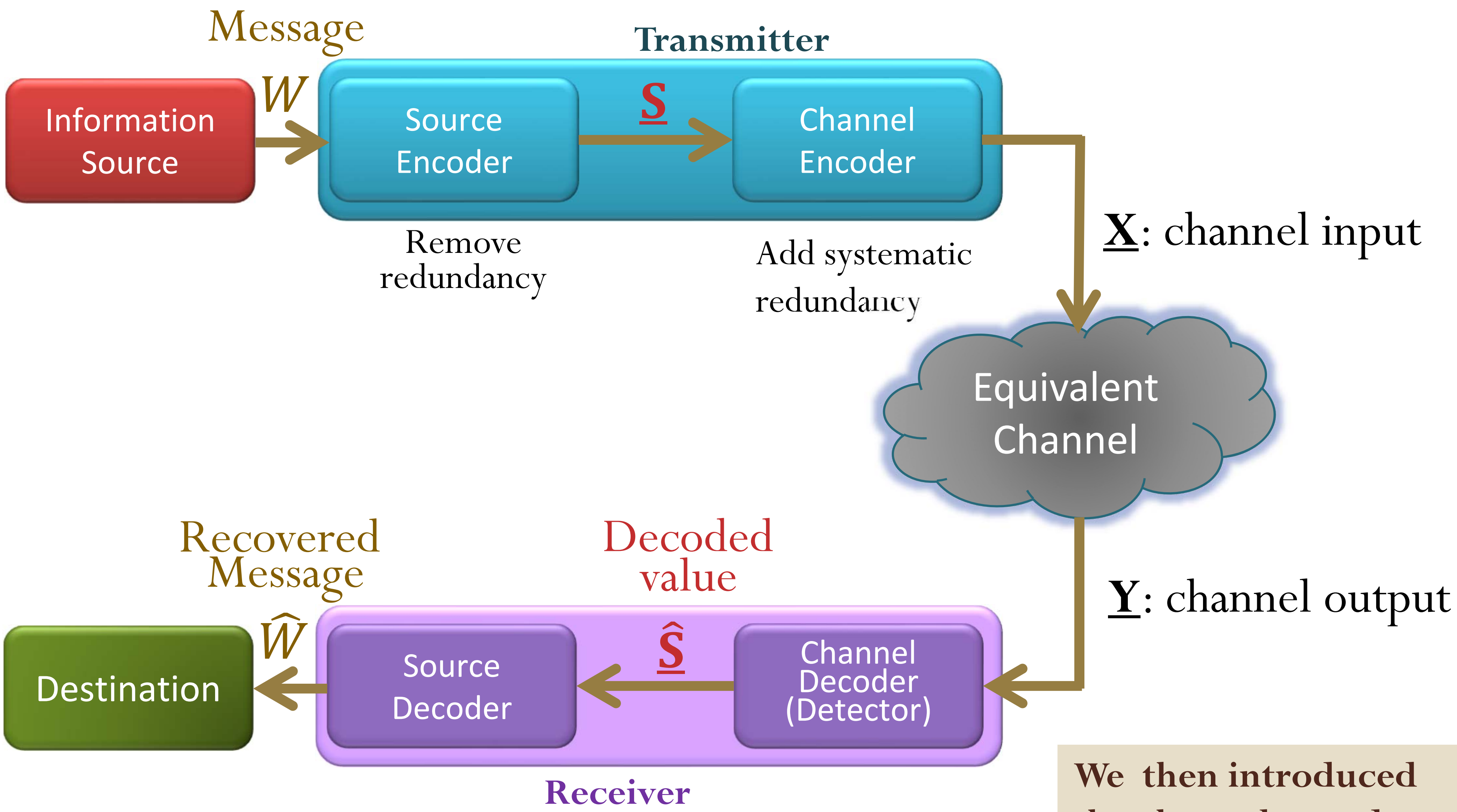
(3) ML decoder is the same as the minimum distance decoder when the crossover probability of the BSC p is < 0.5 (which is usually the case).

53

Under appropriate assumptions, minimum distance decoder is optimal.

System Model for Section 3.5

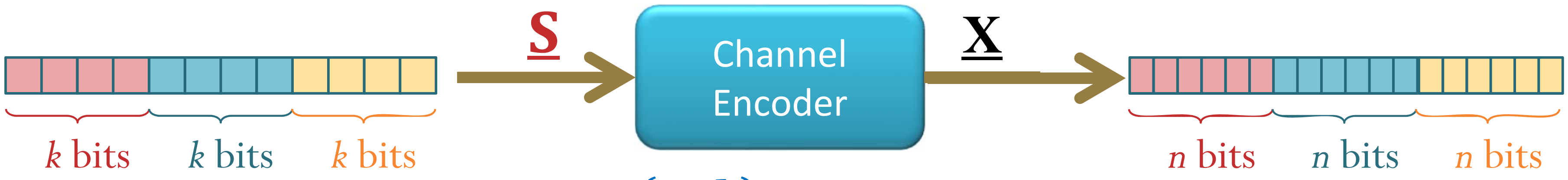
Review



We then introduced the channel encoder box.

Review

[3.62] Block Encoding

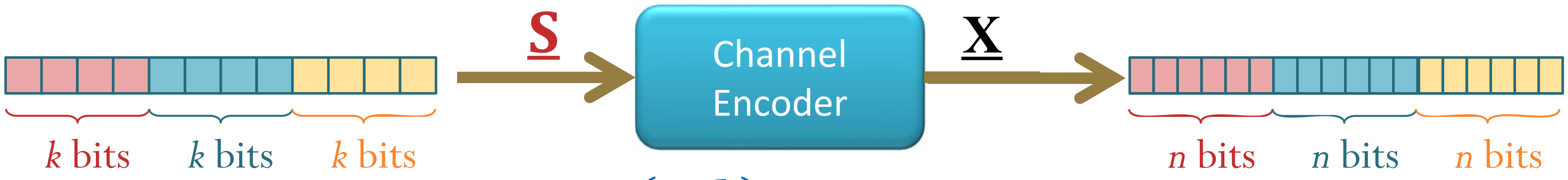


(n, k) code

Code rate $= \frac{k}{n}$

Review

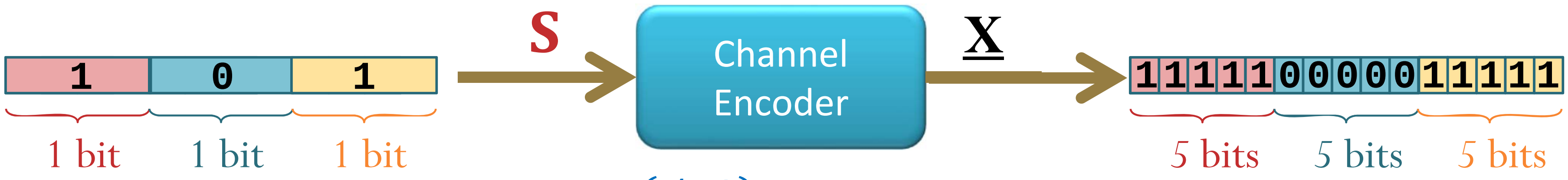
[3.62] Block Encoding



(n, k) code

$\text{Code rate} = \frac{k}{n}$

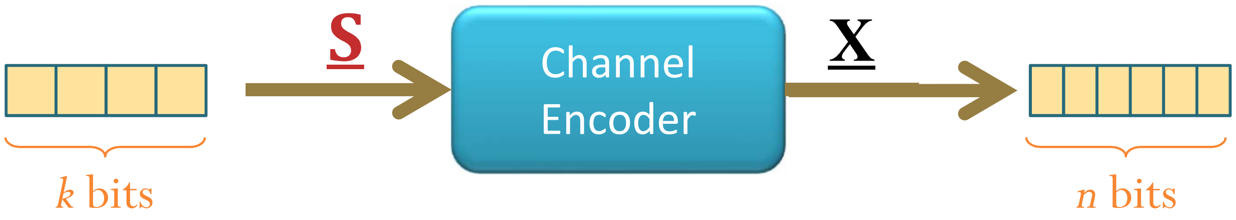
Example: Repetition Code



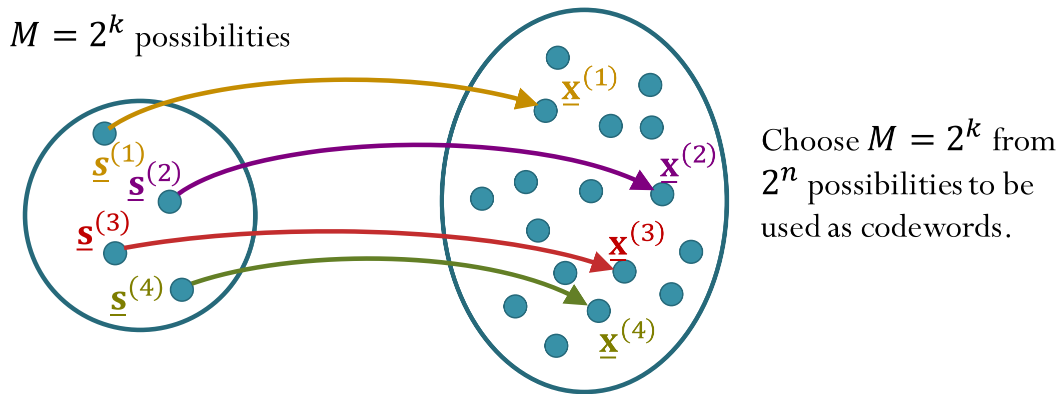
$(5, 1)$ code

$\text{Code rate} = \frac{1}{5}$

[3.62] Block Encoding



[Figure 13]



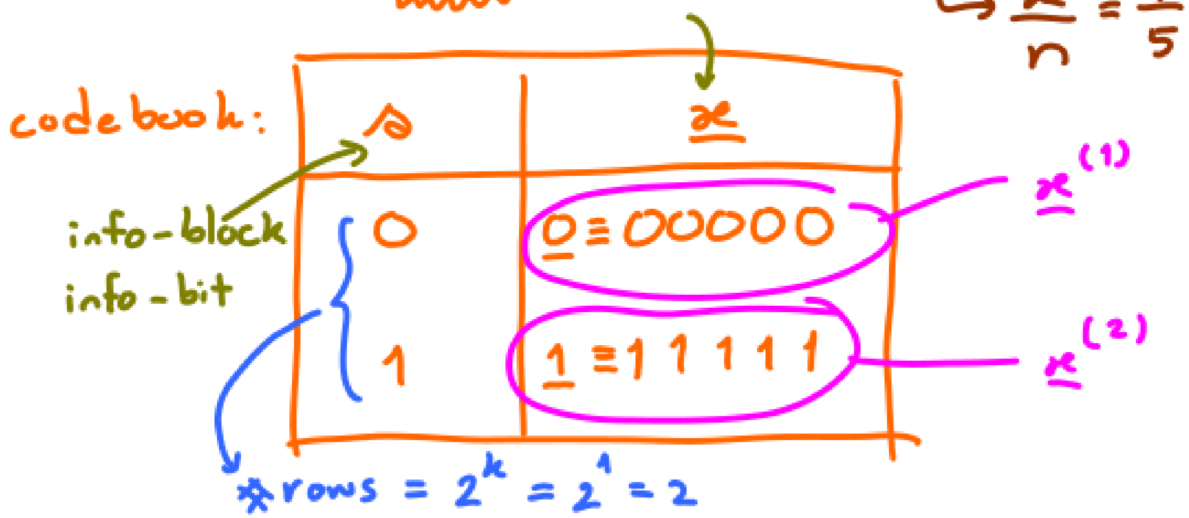
Codebook

index i	info-block $\underline{\mathbf{s}}$	codeword $\underline{\mathbf{x}}$
1	$\underline{\mathbf{s}}^{(1)} = 000\ldots0$	$\underline{\mathbf{x}}^{(1)} =$
2	$\underline{\mathbf{s}}^{(2)} = 000\ldots1$	$\underline{\mathbf{x}}^{(2)} =$
$\vdots$	$\vdots$	$\vdots$
M	$\underline{\mathbf{s}}^{(M)} = 111\ldots1$	$\underline{\mathbf{x}}^{(M)} =$

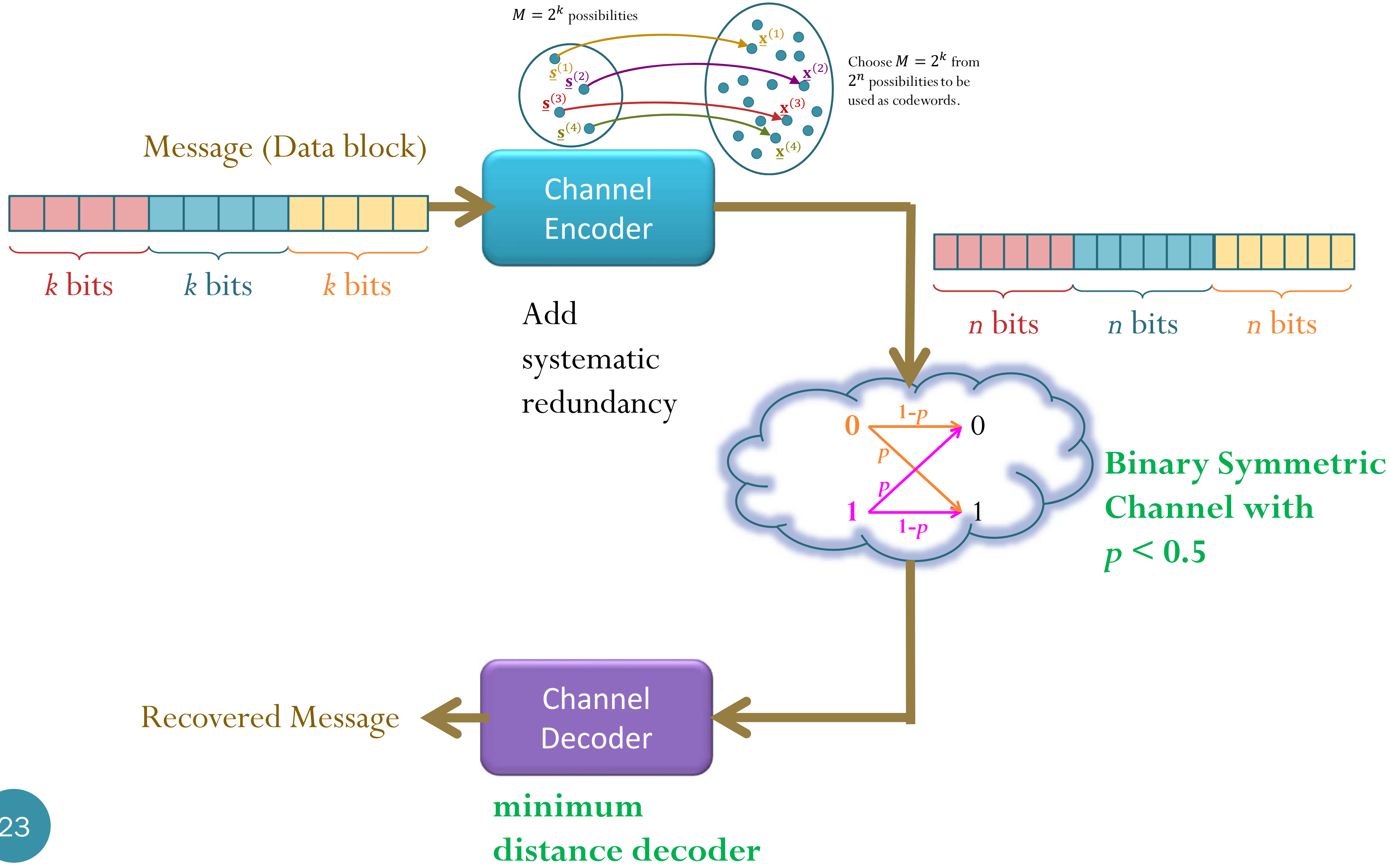
Review

Repetition Code

Example 3.63. Find the codebook and code rate for the encoder which uses repetition code with $n = 5$.

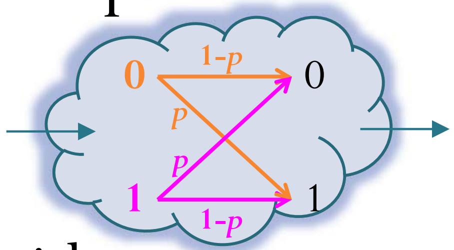


Review: Channel Encoder and Decoder



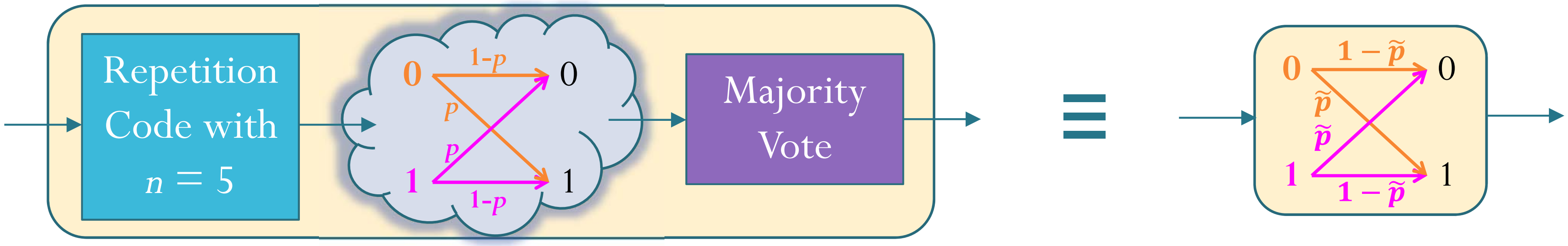
Example: Repetition Code [Figure 14]

- Original Equivalent Channel:



  - BSC with crossover probability $p = 0.01$

- New (and Better) Equivalent Channel:

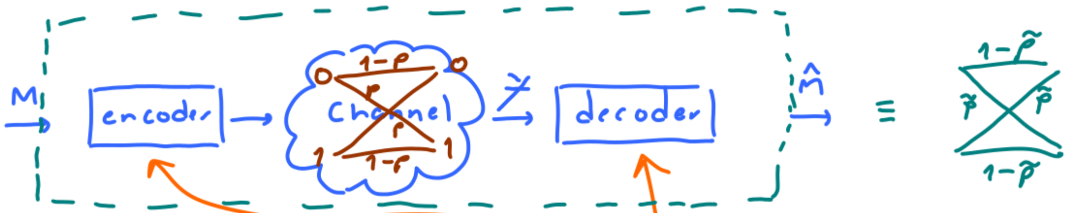


  - Use repetition code with $n = 5$ at the transmitter
  - Use majority vote at the receiver
  - New BSC with $\tilde{p} = \binom{5}{3} p^3(1-p)^2 + \binom{5}{4} p^4(1-p)^1 + \binom{5}{5} p^5(1-p)^0 \approx 10^{-5}$

Review

[From ECS315]

One method of reducing the error rate is to use error-correcting codes:



A simple error-correcting code is the *repetition code.* Example of such code is described below:

Two ways to calculate the probability of error:

(a) (transmission) error occurs if and only if the number of bits in error are ≥ 3.

exactly 3 error bits

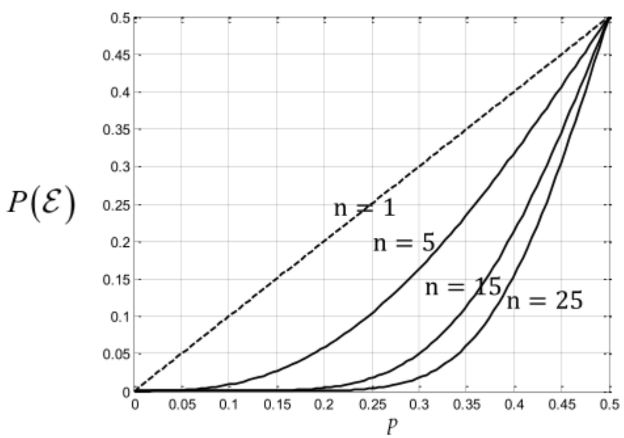
$$\tilde{p} \equiv P(\mathcal{E}) = \binom{5}{3}p^3(1-p)^2 + \binom{5}{4}p^4(1-p) + \binom{5}{5}p^5(1-p)^0$$

with $p = 0.01$

$P(\mathcal{E}) \approx 10^{-5}$

(b) (transmission) error occurs if and only if the number of bits *not* in error are ≤ 2. → 0, 1, 2

$$P(\mathcal{E}) = \binom{5}{0}(1-p)^0 p^5 + \binom{5}{1}(1-p)^1 p^4 + \binom{5}{2}(1-p)^2 p^3$$



MATLAB

```
close all; clear all;

% ECS315 Example 6.58
% ECS452 Example 3.66
C = [0 0 0 0 0; 1 1 1 1 1]; % repetition code

p = (1/100);
PE_minDist(C,p)
```

Code C is defined by putting all its (valid) codewords as its rows. For repetition code, there are two codewords: 00..0 and 11..1.

Crossover probability of the binary symmetric channel.

```
>> PE_minDist_demo1

ans =
   9.8506e-06
```

MATLAB

```
function PE = PE_minDist(C,p)
% Function PE_minDist_3 computes the error probability P(E) when code C
% is used for transmission over BSC with crossover probability p.
% Code C is defined by putting all its (valid) codewords as its rows.
M = size(C,1);
k = log2(M);
n = size(C,2);

% Generate all possible received vectors
Y = dec2bin(0:2^n-1)-'0';

% Normally, we need to construct an extended Q matrix. However, because
% each conditional probability in there is a decreasing function of the
% Hamming distance, we can work with the distances instead of the
% conditional probability. In particular, instead of selecting the max in
% each column of the Q matrix, we consider min distance in each column.
dmin = zeros(1,2^n);
for j = 1:(2^n)
    % for each received vector y,
    y = Y(j,:);
    % find the minimum distance (the distance from y to the closest
    % codeword)
    d = sum(mod(bsxfun(@plus,y,C),2),2);
    dmin(j) = min(d);
end

% From the distances, calculate the conditional probabilities.
% Note that we compute only the values that are to be selected (instead of
% calculating the whole Q first).
n1 = dmin; n0 = n-dmin;
Qmax = (p.^n1).*((1-p).^n0);
% Scale the conditional probabilities by the input probabilities and add
% the values. Note that we assume equally likely input.
PC = sum((1/M)*Qmax);
PE = 1-PC;
end
```

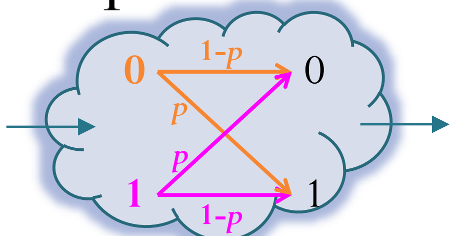
MATLAB

- Annotated version [Posted @ 3PM on Feb 21; Updated @ 5PM on Feb 28]
- Slides [Posted @ 9PM on Feb 8; Updated @ 4:30PM on Feb 14, @ 3PM on Feb 21, and @ 5PM on Feb 28]
- Exercise 5 Solution [Posted @ 10AM on Feb 25]
- Exercise 6 Solution [Posted @ 10AM on Feb 25]
- Exercise 7 Solution [Posted @ 10AM on Feb 25]
- MATLAB: BSC_demo.m, BAC_demo.m, DMC_demo.m, DMC_Analysis_demo.m, DMC_Channel_sim.m, BSC_decoder_ALL_demo.m, DMC_decoder_DIY_demo.m, DMC_decoder_ALL_demo.m, DMC_decoder_MAP_demo.m, DMC_decoder_ML_demo.m
- MATLAB: PE_minDist.m, PE_minDist_demo1 .m, PE_minDist_demo2.m

- Chapter 4: Mutual Information and Channel Capacity [Posted @ 11AM on Feb 20]
  - Annotated version [Posted @ 5PM on Feb 28; Updated @ 5PM on Mar 7 and @ 3PM on Mar 8]
  - MATLAB: capacity_blahut.m
  - Exercise 8 Solution [Posted @ 9AM on Mar 6]
  - Exercise 9 Solution [Posted @ 5PM on Mar 7]
  - Exercise 10 Solution [Posted @ 3PM on Mar 19]
  - Slides [Posted @ 5PM on Mar 7; Updated @ 3PM on Mar 8]
- Chapter 5: Channel Coding

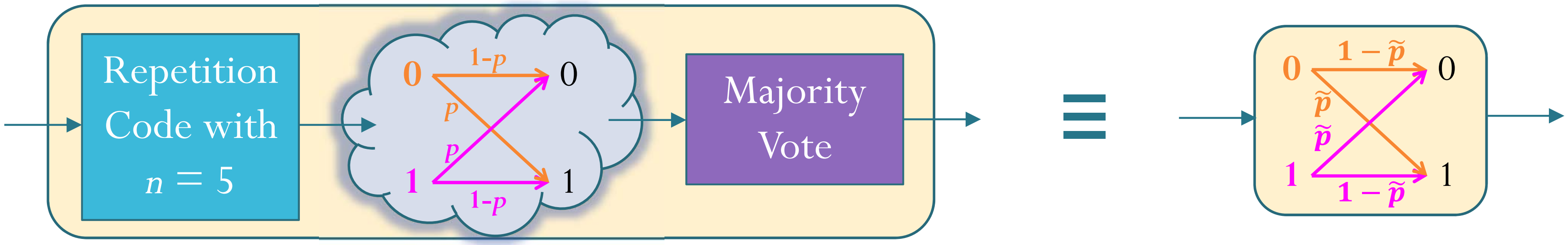
Example: Repetition Code

- Original Equivalent Channel:



  - BSC with crossover probability p

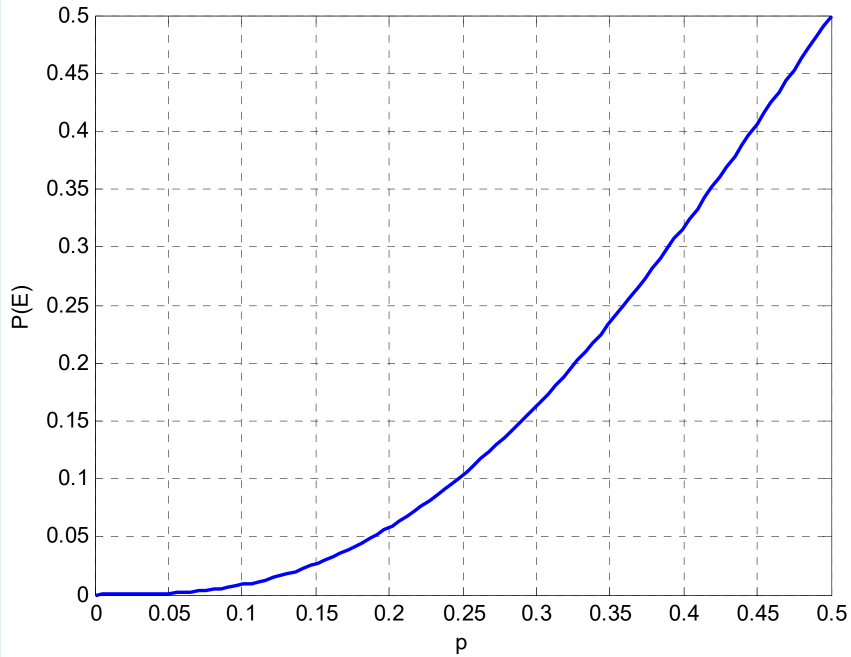
- New (and Better) Equivalent Channel:



  - Use repetition code at the transmitter
  - Use majority vote at the receiver
  - New BSC with new crossover probability $\tilde{p}$

MATLAB

```
close all; clear all;

% ECS315 Example 6.58
% ECS452 Example 3.66
C = [0 0 0 0 0; 1 1 1 1 1];

syms p;
PE = PE_minDist(C,p)
pp = linspace(0,0.5,100);
PE = subs(PE,p,pp);
plot(pp,PE,'LineWidth',1.5)
xlabel('p')
ylabel('P(E)')
grid on
```




```
>> PE_minDist_demo2

PE =
(p - 1)^5 + 10*p^2*(p - 1)^3 - 5*p*(p - 1)^4 + 1
```

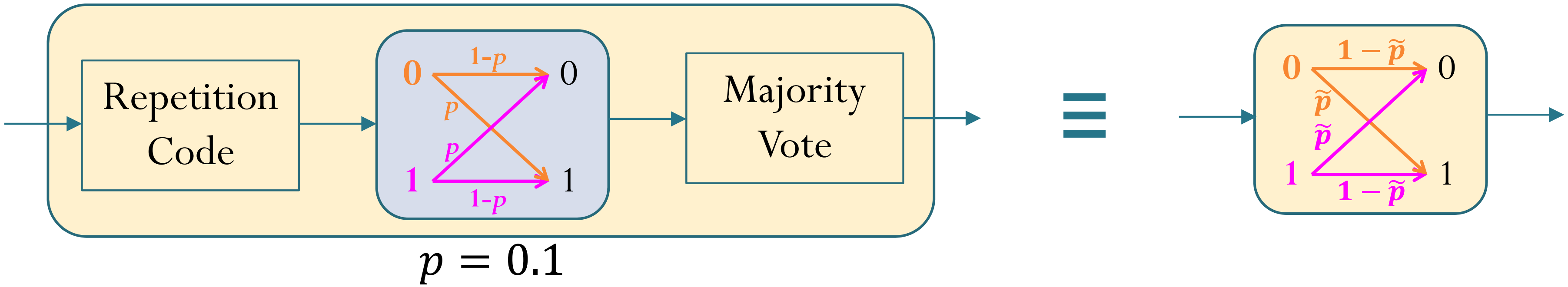
Searching for the best encoder

- Now that we have MATLAB function **PE_minDist**, for specific values of n, k, we can try to search for the encoder that minimizes the error probability.
- Recall that, from Example 3.64, there are

$$\binom{2^n}{M} = \binom{2^n}{2^k} \text{ reasonable encoders.}$$

- Even for small n and k, this is a large space to look at every possible cases.

Example: Repetition Code



n	$\tilde{p}$
1	$p = 0.1$
3	$\binom{3}{2} p^2 (1-p) + \binom{3}{3} p^3 \approx 0.0280$
5	$\binom{5}{3} p^3 (1-p)^2 + \binom{5}{4} p^4 (1-p)^1 + \binom{5}{5} p^5 \approx 0.0086$
7	≈ 0.0027
9	$\approx 8.9092 \times 10^{-4}$
11	$\approx 2.9571 \times 10^{-4}$

Channel Capacity

[Section 4.2]

"**Operational**": max rate at which **reliable** communication is possible

Arbitrarily small error probability can be achieved.

Channel Capacity

"**Information**": $\max_{\underline{\mathbf{p}}} I(X;Y)$ [bpcu]

[Section 4.3]

Shannon [1948] shows that these two quantities are actually the same.

Digital Communication Systems ECS 452

Asst. Prof. Dr. Prapun Suksompong
prapun@siit.tu.ac.th

Information Channel Capacity

Channel Capacity

Channel Capacity

"**Operational**": max rate at which **reliable** communication is possible

→ Arbitrarily small error probability can be achieved.

"**Information**": $\max_{\underline{\mathbf{p}}} I(X;Y)$ [bpcu]

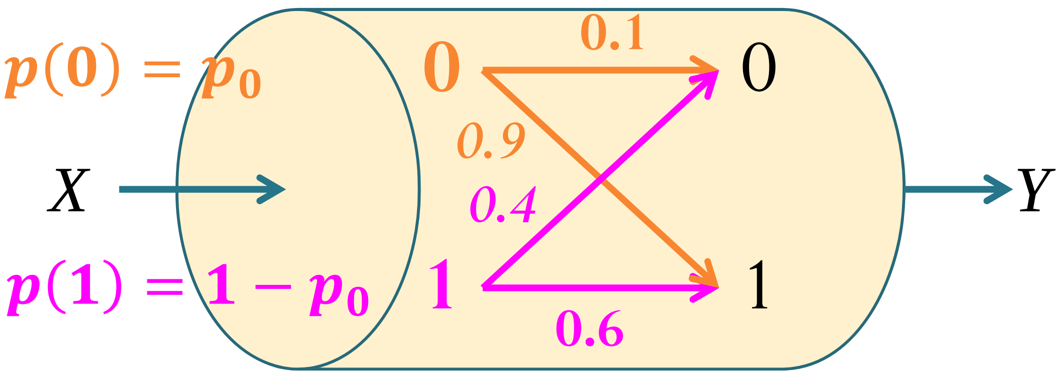
Shannon [1948] shows that these two quantities are actually the same.

MATLAB

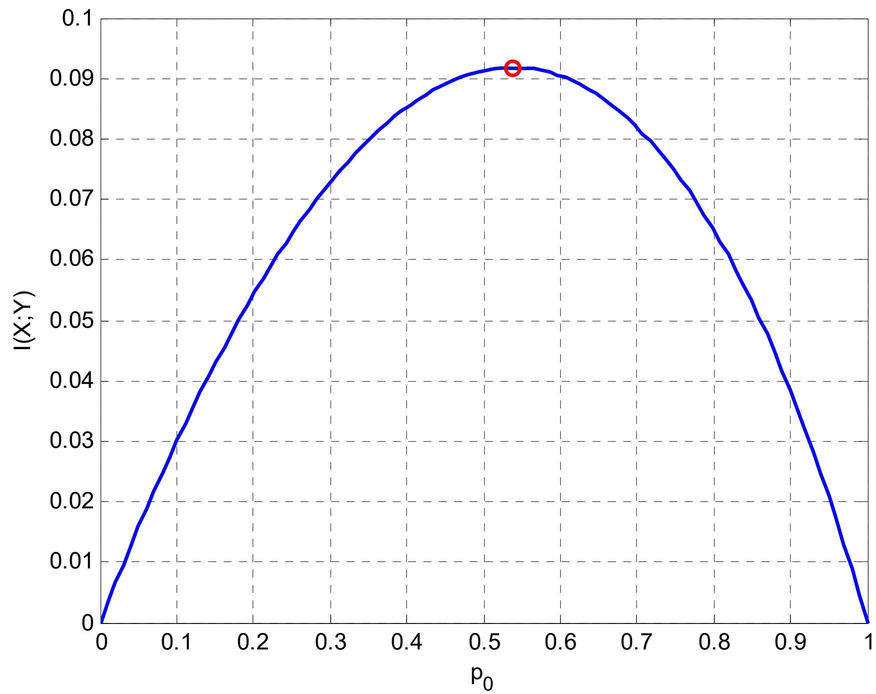
```
function H = entropy2s(p)
% ENTROPY2 accepts probability mass function
% as a row vector, calculate the corresponding
% entropy in bits.
p=p(find(abs(sort(p)-1)>1e-8)); % Eliminate 1
p=p(find(abs(p)>1e-8)); % Eliminate 0
if length(p)==0
    H = 0;
else
    H = simplify(-sum(p.*log(p))/log(sym(2)));
end
```

```
function I = informations(p,Q)
X = length(p);
q = p*Q;
HY = entropy2s(q);
temp = [];
for i = 1:X
    temp = [temp entropy2s(Q(i,:))];
end
HYgX = sum(p.*temp);
I = HY-HYgX;
```

Capacity calculation for BAC

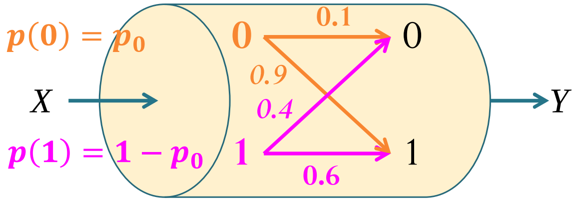


$$Q = \begin{bmatrix} 0.1 & 0.9 \\ 0.4 & 0.6 \end{bmatrix}$$



Capacity of 0.0918 bits is achieved by $\underline{p} = [0.5380,\ 0.4620]$

Capacity calculation for BAC

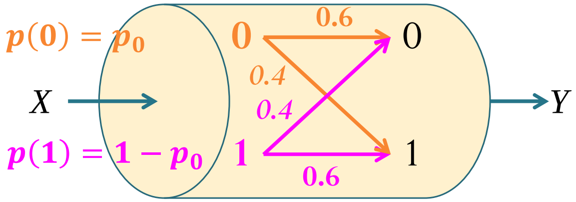


$$Q = \begin{bmatrix} 0.1 & 0.9 \\ 0.4 & 0.6 \end{bmatrix}$$

```
close all; clear all;
syms p0
p = [p0 1-p0];
Q = [1 9; 4 6]/sym(10);

I = simplify(informations(p,Q))

p0o = simplify(solve(diff(I)==0))

po = eval([p0o 1-p0o])

C = simplify(subs(I,p0,p0o))

eval(C)
```

```
>> Capacity_Ex_BAC
I =
(log(2/5 - (3*p0)/10)*((3*p0)/10 - 2/5) - log((3*p0)/10 + 3/5)*((3*p0)/10 + 3/5))/log(2) + (log((5*2^(3/5)*3^(2/5))/6)*(p0 - 1))/log(2) + (p0*log((3*3^(4/5))/10))/log(2)
p0o =
(27648*2^(1/3))/109565 - (69984*2^(2/3))/109565 + 135164/109565
po =
   0.5376   0.4624
C =
(log((3*3^(4/5))/10)*((27648*2^(1/3))/109565 - (69984*2^(2/3))/109565 + 135164/109565))/log(2) - (log((104976*2^(2/3))/547825 - (41472*2^(1/3))/547825 + 16384/547825)*((104976*2^(2/3))/547825 - (41472*2^(1/3))/547825 + 16384/547825) + log((41472*2^(1/3))/547825 - (104976*2^(2/3))/547825 + 531441/547825)*((41472*2^(1/3))/547825 - (104976*2^(2/3))/547825 + 531441/547825))/log(2) + (log((5*2^(3/5)*3^(2/5))/6)*((27648*2^(1/3))/109565 - (69984*2^(2/3))/109565 + 25599/109565))/log(2)
ans =
   0.0918
```

Same procedure applied to BSC



$$Q = \begin{bmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{bmatrix}$$

```
close all; clear all;
syms p0
p = [p0 1-p0];
Q = [6 4; 4 6]/sym(10);

I = simplify(informations(p,Q))

p0o = simplify(solve(diff(I)==0))

po = eval([p0o 1-p0o])

C = simplify(subs(I,p0,p0o))

eval(C)
```

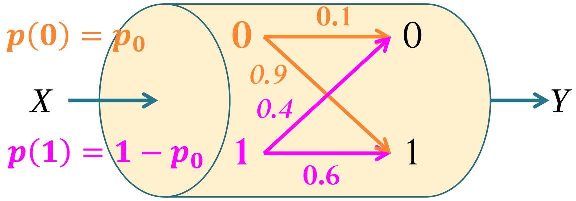
```
>> Capacity_Ex_BSC
I =
(log((5*2^(3/5)*3^(2/5))/6)*(p0 - 1))/log(2) -
(p0*log((5*2^(3/5)*3^(2/5))/6))/log(2) - (log(p0/5 +
2/5)*(p0/5 + 2/5) - log(3/5 - p0/5)*(p0/5 -
3/5))/log(2)
p0o =
1/2
po =
   0.5000    0.5000
C =
log((2*2^(2/5)*3^(3/5))/5)/log(2)
ans =
   0.0290
```

Blahut–Arimoto algorithm

```
function [ps C] = capacity_blahut(Q)
% Input:     Q  = channel transition probability matrix
% Output:    C  = channel capacity
%            ps = row vector containing pmf that achieves capacity

tl = 1e-8; % tolerance (for the stopping condition)
n = 1000; % max number of iterations (in case the stopping condition
          % is "never" reached")
nx = size(Q,1); pT = ones(1,nx)/nx; % First, guess uniform X.
for k = 1:n
    qT = pT*Q;
    % Eliminate the case with 0
    % Column-division by qT
    temp = Q.*(ones(nx,1)*(1./qT));
    %Eliminate the case of 0/0
    l2 = log2(temp);
    l2(find(isnan(l2) | (l2==-inf) | (l2==inf)))=0;
    logc = (sum(Q.*(l2),2))';
    CT = 2.^(logc);
    A = log2(sum(pT.*CT)); B = log2(max(CT));
    if((B-A)<tl)
        break
    end
    % For the next loop
    pT = pT.*CT;     % un-normalized
    pT = pT/sum(pT); % normalized
    if(k == n)
        fprintf('\nNot converge within n loops\n')
    end
end
ps = pT;
C = (A+B)/2;
```

[capacity_blahut.m]

Capacity calculation for BAC: a revisit



$$Q = \begin{bmatrix} 0.1 & 0.9 \\ 0.4 & 0.6 \end{bmatrix}$$

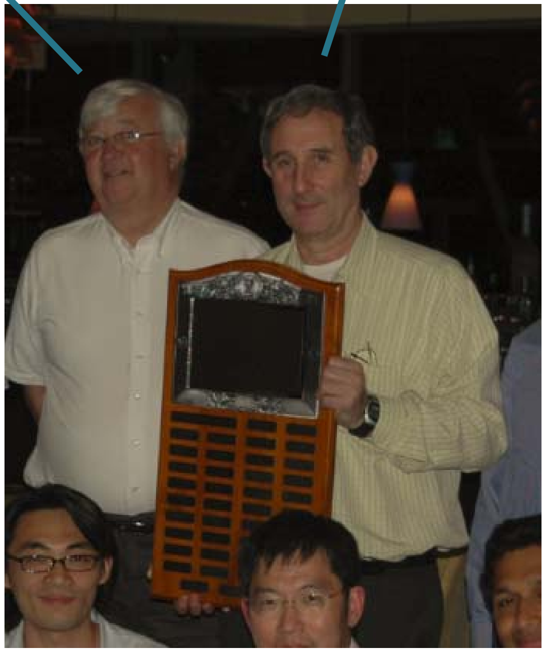
```matlab
close all; clear all;

Q = [1 9; 4 6]/10;

[ps C] = capacity_blahut(Q)
```

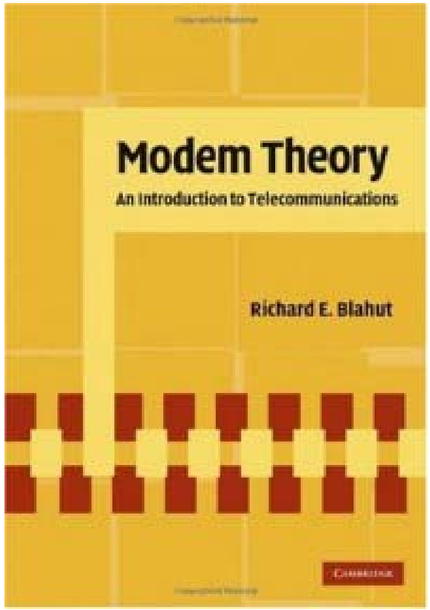
```
>> Capacity_Ex_BAC_blahut
ps =
   0.5376   0.4624
C =
   0.0918
```

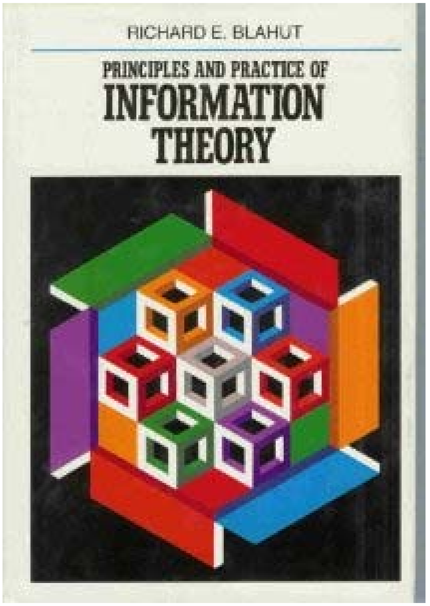
Richard Blahut

- Former chair of the Electrical and Computer Engineering Department at the University of Illinois at Urbana-Champaign
- Best known for **Blahut–Arimoto algorithm (Iterative Calculation of C)**

Toby Berger







Claude E. Shannon Award

Claude E. **Shannon** (1972)

David S. Slepian (1974)

Robert M. Fano (1976)

Peter Elias (1977)

Mark S. Pinsker (1978)

Jacob Wolfowitz (1979)

W. Wesley Peterson (1981)

Irving S. Reed (1982)

Robert G. Gallager (1983)

Solomon W. Golomb (1985)

William L. Root (1986)

James L. Massey (1988)

Thomas M. Cover (1990)

Andrew J. Viterbi (1991)

Elwyn R. Berlekamp (1993)

Aaron D. Wyner (1994)

G. David Forney, Jr. (1995)

Imre Csiszár (1996)

Jacob Ziv (1997)

Neil J. A. Sloane (1998)

Tadao Kasami (1999)

Thomas Kailath (2000)

Jack Keil Wolf (2001)

Toby **Berger** (2002)

Lloyd R. Welch (2003)

Robert J. McEliece (2004)

Richard Blahut (2005)

Rudolf Ahlswede (2006)

Sergio Verdu (2007)

Robert M. Gray (2008)

Jorma Rissanen (2009)

Te Sun Han (2010)

Shlomo Shamai (Shitz) (2011)

Abbas El Gamal (2012)

Katalin Marton (2013)

János Körner (2014)

Arthur Robert Calderbank (2015)

Alexander S. Holevo (2016)

David Tse (2017)

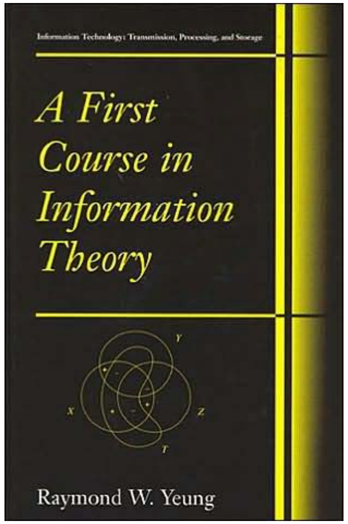
[http://www.itsoc.org/honors/claude-e-shannon-award]

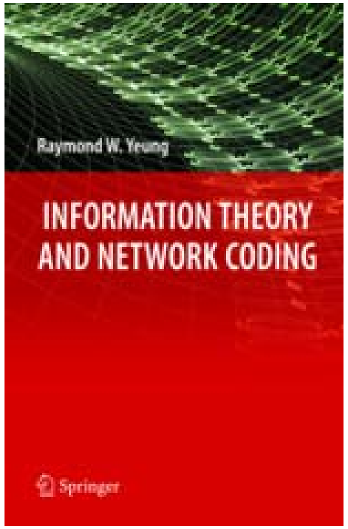
Berger plaque

เรย์มอนด์ ยึง

Raymond Yeung

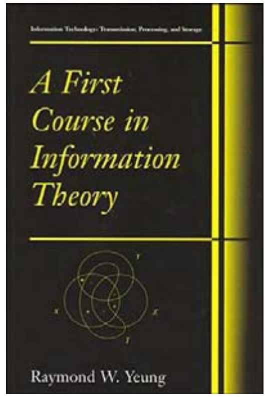
- BS, MEng and PhD degrees in electrical engineering from **Cornell** University in 1984, 1985, and 1988, respectively.





Raymond Yeung

- Introduce, for the first time in a textbook,
  - analytical theory of I-Measure and
  - geometrically intuitive information diagrams
  - Establish a one-to-one correspondence between Shannon's information measures and set theory.
- Rooted in works by G. D. Hu, by H. Dyckman, and by R. Yeung et al.



Chapter 6

THE *I*-MEASURE

In Chapter 2, we have shown the relationship between Shannon's information measures for two random variables by the diagram in Figure 2.2. For convenience, Figure 2.2 is reproduced in Figure 6.1 with the random variables X and Y replaced by X_1 and X_2, respectively. This diagram suggests that Shannon's information measures for any $n \geq 2$ random variables may have a set-theoretic structure.

In this chapter, we develop a theory which establishes a one-to-one correspondence between Shannon's information measures and set theory in full generality. With this correspondence, manipulations of Shannon's information measures can be viewed as set operations, thus allowing the rich suite of tools in set theory to be used in information theory. Moreover, the structure of Shannon's information measures can easily be visualized by means of an

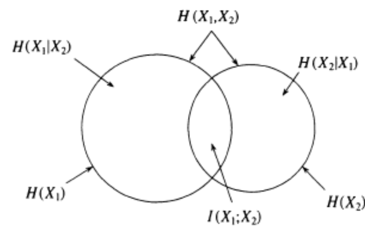


Figure 6.1. Relationship between entropies and mutual information for two random variables.

95

Toby Berger with Berger plaque

Douglas Chan and 802.11n

Contributions to this amendment was received from the follo

Bill Abbott
Santosh Abraham
Tomoko Adachi
Dmitry Akhmetov
Carlos Aldana
Dave Andrus
Micha Anholt
Tsuguhide Aoki
Yusuke Asai
Geert Awater
David Bagby
Raja Banerjea
Kaberi Banerjee
Amit Bansal
Gal Basson
Anuj Batra
John Benko
Mathilde Benveniste
Bjorn Bjerke
Yufei Blakenship
Daniel Borges
Douglas Chan
Jerry Chang

Vinko Erceg
Mustafa Eroz
Stefan Fechtel
Paul Feinberg
Matthew Fischer
Guido Frederiks
Takashi Fukagawa
Patrick Fung
Edoardo Gallizio

Yuh-Ren Jauh

Hui-Ling Lou
Adina Matache
Laurent Mazet

802.11n™

IEEE

IEEE Standard for
Information technology—
Telecommunications and information
exchange between systems—
Local and metropolitan area networks—
Specific requirements

Part 11: Wireless LAN Medium Access Control (MAC)
and Physical Layer (PHY) Specifications

Amendment 5: Enhancements for Higher
Throughput

IEEE Computer Society

Sponsored by the
LAN/MAN Standards Committee

IEEE
3 Park Avenue
New York, NY 10016-5997, USA

29 October 2009

IEEE Std 802.11n™-2009
(Amendment to IEEE Std 802.11™-2007
as amended by IEEE Std 802.11k™-2008,
IEEE Std 802.11r™-2008, IEEE Std 802.11y™-2008,
and IEEE Std 802.11w™-2009)

Douglas Chan and 802.11n

September 2005 doc.: IEEE 802.11-05/0946r0

Improving IEEE 802.11 Performance with Cross-Layer Design and Multipacket Reception via Multiuser Iterative Decoding

Date: 2005-09-20

Authors:

Name	Company	Address	Phone	email
Douglas S. Chan	Cornell University	School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853	607-254-8818	dsc29@cornell.edu
Prapun Suksompong				ps92@cornell.edu
Jun Chen				jc353@cornell.edu
Toby Berger			607-255-1447	berger@ece.cornell.edu

Digital Communication Systems ECS 452

Asst. Prof. Dr. Prapun Suksompong

prapun@siit.tu.ac.th

Special Cases for Calculation of Channel Capacity

Channel Capacity: Special Cases

- **Channel with Nonoverlapping Outputs (NO^2)**
  - There is only one non-zero element in each column of its **Q** matrix.
  - $C = \log_2 |\mathcal{X}|$ [4.30]
    is achieved by uniform input probabilities.
  - Ex. Noiseless Binary Channel: $C = 1$ [bpcu] [Ex. 4.27]
- **Weakly Symmetric Channel**
  - (1) all the rows of **Q** are permutations of each other and (2) all the column sums are equal. [Defn 4.36]
  - $C = \log_2 |\mathcal{Y}| - H(\underline{\mathbf{r}})$ where $\underline{\mathbf{r}}$ is any row from the **Q** matrix. [4.37]
    is achieved by uniform input probabilities.
  - Ex. Binary Symmetric Channel: $C = 1 - H(p)$ [bpcu]