Digital Communication Systems
EES 452

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

2. Source Coding

2.2 Optimal Source Coding:
Huffman Coding: Origin, Recipe, MATLAB Implementation

Examples of Prefix Codes

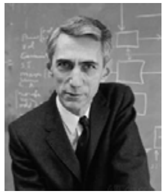
- Nonsingular Fixed-Length Code
- Shannon–Fano code
- Huffman Code

Prof. Robert Fano (1917-2016)
Shannon Award (1976)

Shannon–Fano Code

- Proposed in Shannon's "A Mathematical Theory of Communication" in 1948
- The method was attributed to Fano, who later published it as a technical report.
  - Fano, R.M. (1949). "The transmission of information". Technical Report No. 65. Cambridge (Mass.), USA: Research Laboratory of Electronics at MIT.
- Should not be confused with
  - Shannon coding, the coding method used to prove Shannon's noiseless coding theorem, or with
  - Shannon–Fano–Elias coding (also known as Elias coding), the precursor to arithmetic coding.

Claude E. Shannon Award



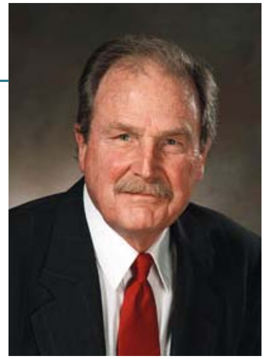
Claude E. **Shannon** (1972)
David S. Slepian (1974)
Robert M. Fano (1976)
Peter Elias (1977)
Mark S. Pinsker (1978)
Jacob Wolfowitz (1979)
W. Wesley Peterson (1981)
Irving S. Reed (1982)
Robert G. Gallager (1983)
Solomon W. Golomb (1985)
William L. Root (1986)
James L. Massey (1988)
Thomas M. Cover (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)
Gottfried Ungerboeck (2018)
Erdal Arıkan (2019)
Charles Bennett (2020)

[http://www.itsoc.org/honors/claude-e-shannon-award]
[https://en.wikipedia.org/wiki/Claude_E._Shannon_Award]

Huffman Code



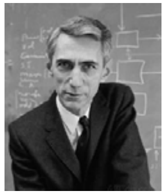
David Huffman (1925–1999)
Hamming Medal (1999)

- MIT, 1951
- Information theory class taught by Professor Fano.
- Huffman and his classmates were given the choice of
  - a term paper on the problem of finding the most efficient binary code.

  or
  - a final exam.
- Huffman, unable to prove any codes were the most efficient, was about to give up and start studying for the final when he hit upon the idea of using a frequency-sorted binary tree and quickly proved this method the most efficient.
- Huffman avoided the major flaw of the suboptimal Shannon-Fano coding by building the tree from the bottom up instead of from the top down.

[Encoding the "Neatness" of Ones and Zeroes from the September 1991 issue of Scientific American, pp. 54, 58. https://www.jstor.org/stable/24938714]

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)
Gottfried Ungerboeck (2018)
Erdal Arıkan (2019)
Charles Bennett (2020)

[http://www.itsoc.org/honors/claude-e-shannon-award]
[https://en.wikipedia.org/wiki/Claude_E._Shannon_Award]

https://www.ieee.org/content/dam/ieee-org/ieee/web/org/about/hamming_rl.pdf

IEEE Richard W. Hamming Medal

1988 - Richard W. **Hamming**
1989 - Irving S. Reed
1990 - Dennis M. Ritchie and Kenneth L. Thompson
1991 - Elwyn R. Berlekamp
1992 - Lotfi A. Zadeh
1993 - Jorma J. Rissanen
1994 - Gottfried Ungerboeck
1995 - Jacob Ziv
1996 - Mark S. Pinsker
1997 - Thomas M. Cover
1998 - David D. Clark
1999 - David A. Huffman
2000 - Solomon W. Golomb
2001 - A. G. Fraser
2002 - Peter Elias
2003 - Claude Berrou and Alain Glavieux
2004 - Jack K. Wolf
2005 - Neil J.A. Sloane

2006 - Vladimir I. Levenshtein
2007 - Abraham Lempel
2008 - Sergio Verdú
2009 - Peter Franaszek
2010 - Whitfield Diffie,
Martin Hellman,
and Ralph Merkle
2011 - Toby **Berger**
2012 - Michael Luby, Amin Shokrollahi
2013 - Arthur Robert Calderbank
2014 - Thomas Richardson
and Rüdiger L. Urbanke
2015 - Imre Csiszar
2016 - Abbas El Gamal
2017 - Shlomo Shamai
2018 - Erdal Arikan
2019 - David Tse
2020 - Cynthia Dwork

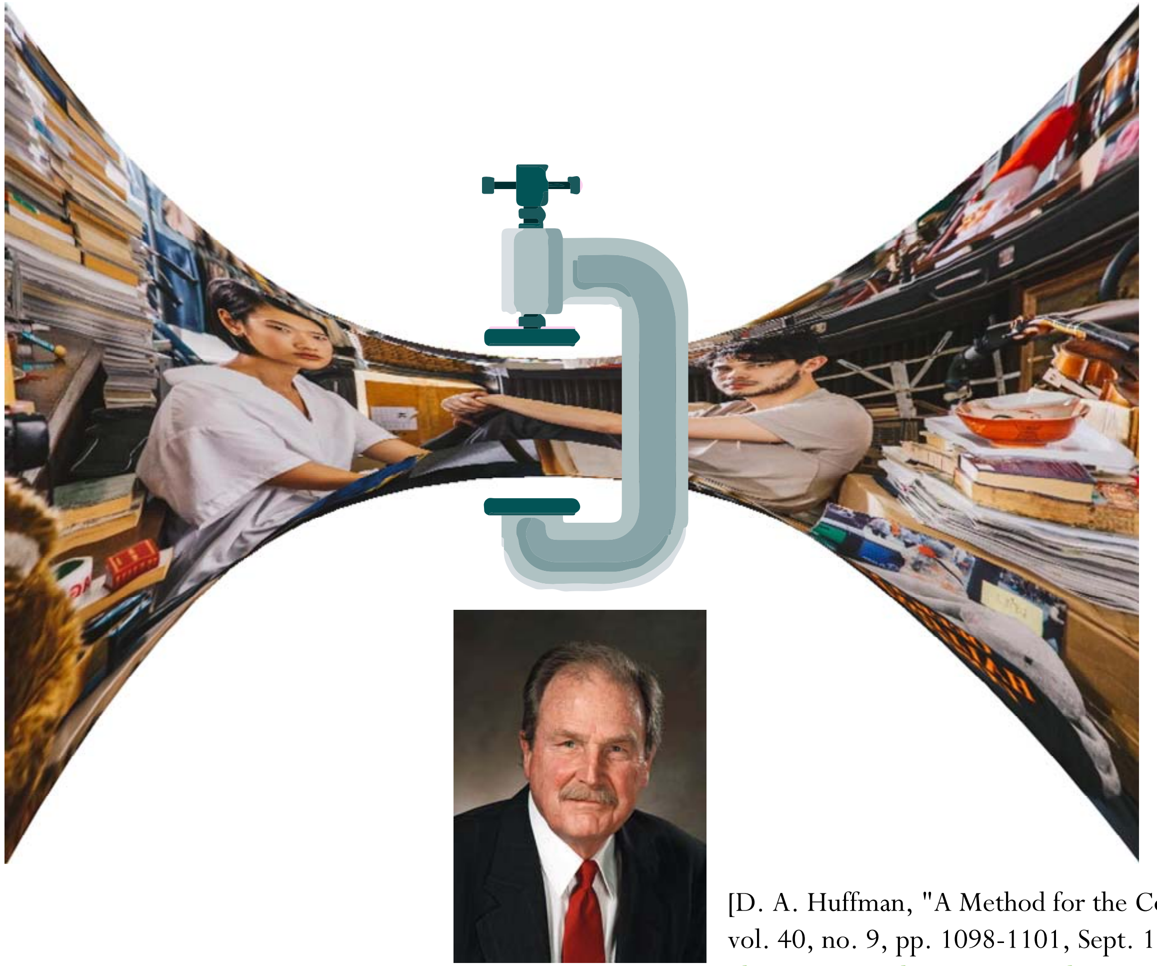
"For contributions to Information Theory, including **source coding** and its applications."

[http://www.cvaiee.org/html/toby_berger.html]

ฮาวทู compress

compress อย่างไร ไม่ให้เหลือ redundancy

[ฮาวทูทิ้ง ทิ้งอย่างไรไม่ให้เหลือเธอ. กำกับโดย นวพล ธำรงรัตนฤทธิ์, GDH, 2019. ภาพยนตร์]



1098 PROCEEDINGS OF THE I.R.E. September

A Method for the Construction of Minimum-Redundancy Codes*

DAVID A. HUFFMAN†, ASSOCIATE, IRE

Summary—An optimum method of coding an ensemble of messages consisting of a finite number of members is developed. A minimum-redundancy code is one constructed in such a way that the average number of coding digits per message is minimized.

INTRODUCTION

ONE IMPORTANT METHOD of transmitting messages is to transmit in their place sequences of symbols. If there are more messages which might be sent than there are kinds of symbols available, then some of the messages must use more than one symbol. If it is assumed that each symbol requires the same time for transmission, then the time for transmission (length) of a message is directly proportional to the number of symbols associated with it. In this paper, the symbol or sequence of symbols associated with a given message will be called the "message code." The entire number of messages which might be transmitted will be called the "message ensemble." The mutual agreement between the transmitter and the receiver about the meaning of the code for each message of the ensemble will be called the "ensemble code."

Probably the most familiar ensemble code was stated in the phrase "one if by land and two if by sea." In this case, the message ensemble consisted of the two individual messages "by land" and "by sea", and the message codes were "one" and "two."

In order to formalize the requirements of an ensemble code, the coding symbols will be represented by numbers. Thus, if there are D different types of symbols to be used in coding, they will be represented by the digits $0, 1, 2, \cdots, (D-1)$. For example, a ternary code will be constructed using the three digits 0, 1, and 2 as coding symbols.

The number of messages in the ensemble will be called N. Let $P(i)$ be the probability of the ith message. Then

$$\sum_{i=1}^{N} P(i) = 1. \tag{1}$$

The length of a message, $L(i)$, is the number of coding digits assigned to it. Therefore, the average message length is

$$L_{av} = \sum_{i=1}^{N} P(i)L(i). \tag{2}$$

The term "redundancy" has been defined by Shannon[1] as a property of codes. A "minimum-redundancy code" will be defined here as an ensemble code which, for a message ensemble consisting of a finite number of members, N, and for a given number of coding digits, D, yields the lowest possible average message length. In order to avoid the use of the lengthy term "minimum-redundancy," this term will be replaced here by "optimum." It will be understood then that, in this paper, "optimum code" means "minimum-redundancy code."

The following basic restrictions will be imposed on an ensemble code:

(a) No two messages will consist of identical arrangements of coding digits.

(b) The message codes will be constructed in such a way that no additional indication is necessary to specify where a message code begins and ends once the starting point of a sequence of messages is known.

Restriction (b) necessitates that no message be coded in such a way that its code appears, digit for digit, as the first part of any message code of greater length. Thus, 01, 102, 111, and 202 are valid message codes for an ensemble of four members. For instance, a sequence of these messages 1111022020101111102 can be broken up into the individual messages 111-102-202-01-01-111-102. All the receiver need know is the ensemble code. However, if the ensemble has individual message codes including 11, 111, 102, and 02, then when a message sequence starts with the digits 11, it is not immediately certain whether the message 11 has been received or whether it is only the first two digits of the message 111. Moreover, even if the sequence turns out to be 11102, it is still not certain whether 111-02 or 11-102 was transmitted. In this example, change of one of the two message codes 111 or 11 is indicated.

C. E. Shannon[1] and R. M. Fano[2] have developed ensemble coding procedures for the purpose of proving that the average number of binary digits required per message approaches from above the average amount of information per message. Their coding procedures are not optimum, but approach the optimum behavior when N approaches infinity. Some work has been done by Kraft[3] toward deriving a coding method which gives an average code length as close as possible to the ideal when the ensemble contains a finite number of members. However, up to the present time, no definite procedure has been suggested for the construction of such a code

* Decimal classification: R531.1. Original manuscript received by the Institute, December 6, 1951.

† Massachusetts Institute of Technology, Cambridge, Mass.

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Sys. Tech. Jour.*, vol. 27, pp. 398–403; July, 1948.

[2] R. M. Fano, "The Transmission of Information," Technical Report No. 65, Research Laboratory of Electronics, M.I.T., Cambridge, Mass.; 1949.

[3] L. G. Kraft, "A Device for Quantizing, Grouping, and Coding Amplitude-modulated Pulses," Electrical Engineering Thesis, M.I.T., Cambridge, Mass.; 1949.

[D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," in *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098-1101, Sept. 1952.]
[http://ieeexplore.ieee.org/document/4051119/]

https://drive.google.com/file/d/15UcoQP_zhLZeDWpahvfpgNY2wp-VeavE/view?fbclid=IwAR27aS0vw1ulOvV3cIMs-6l79QHRGy3v9KJZF5Idg1Cu4wwYF2drGtT75eQ

Huffman's paper (1952)

1098 *PROCEEDINGS OF THE I.R.E.* *September*

A Method for the Construction of Minimum-Redundancy Codes*

DAVID A. HUFFMAN†, ASSOCIATE, IRE

Summary—**An optimum method of coding an ensemble of messages consisting of a finite number of members is developed. A minimum-redundancy code is one constructed in such a way that the average number of coding digits per message is minimized.**

INTRODUCTION

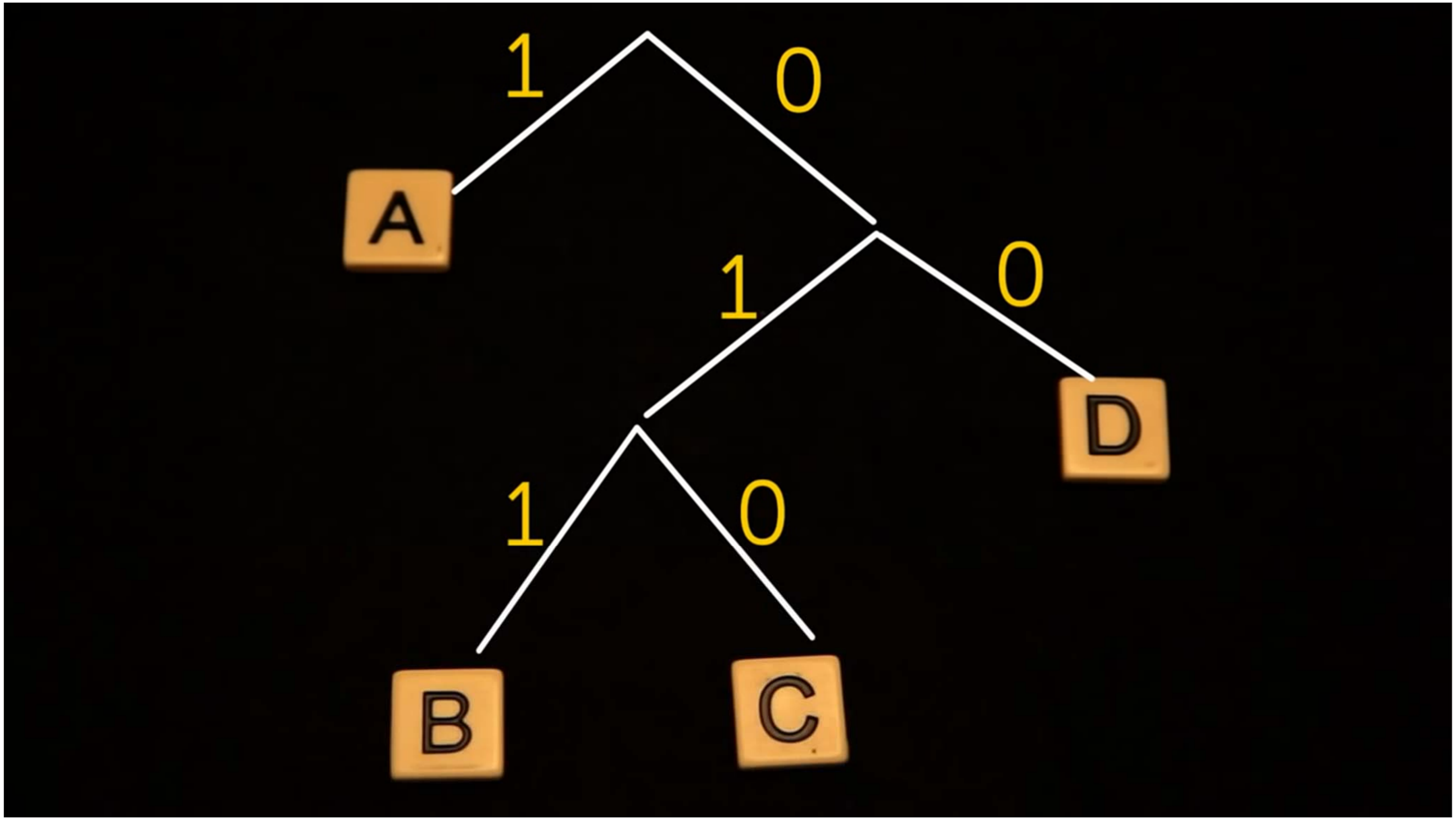
ONE IMPORTANT METHOD of transmitting messages is to transmit in their place sequences of symbols. If there are more messages which might be sent than there are kinds of symbols available, then some of the messages must use more than one symbol. If it is assumed that each symbol requires the same time for transmission, then the time for transmission will be defined here as an ensemble code which, for a message ensemble consisting of a finite number of members, N, and for a given number of coding digits, D, yields the lowest possible average message length. In order to avoid the use of the lengthy term "minimum-redundancy," this term will be replaced here by "optimum." It will be understood then that, in this paper, "optimum code" means "minimum-redundancy code."

The following basic restrictions will be imposed on an ensemble code:

(a) No two messages will consist of identical arrangements of coding digits.

[D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," in *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098-1101, Sept. 1952.]
[http://ieeexplore.ieee.org/document/4051119/]

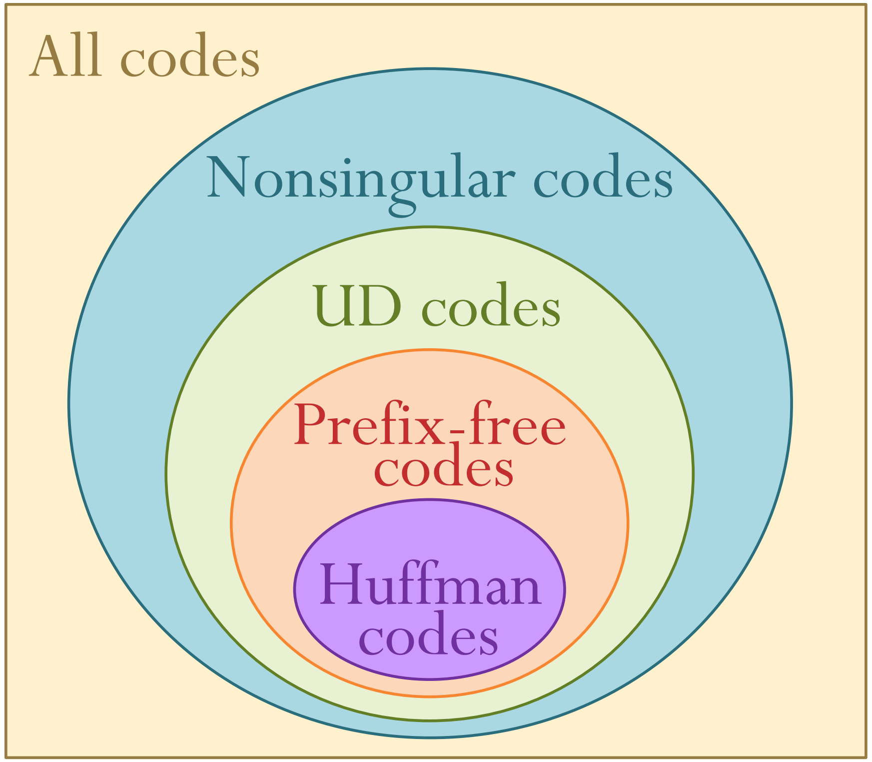
Huffman coding



[https://www.khanacademy.org/computing/computer-science/informationtheory/moderninfotheory/v/compressioncodes]

Summary:

- [2.16-17] A good code must be **uniquely decodable (UD)**. [Defn 2.18]
  - Difficult to check.
- [2.24] Consider a special family of codes: **prefix(-free) code**.
  - Always UD.
  - Same as being **instantaneous**.
- [Defn 2.30] **Huffman**'s recipe
  - Repeatedly combine the two least-likely (combined) symbols.
  - Automatically give prefix-free code.



No codeword is a prefix of any other codeword.

Each source symbol can be decoded as soon as we come to the end of the codeword corresponding to it

[Ex. 2.31] Huffman Coding in MATLAB

Observe that MATLAB automatically give the **expected length** of the codewords

```
pX = [0.5 0.25 0.125 0.125];       % pmf of X
SX = [1:length(pX)];               % Source Alphabet
[dict,EL] = huffmandict(SX,pX);    % Create codebook
```

```
%% Pretty print the codebook.
codebook = dict;
for i = 1:length(codebook)
    codebook{i,2} = num2str(codebook{i,2});
end
codebook
```

```
%% Try to encode some random source string
sourceString = randsrc(1,10,[SX; pX]) % Create data using pX
encodedString = huffmanenco(sourceString,dict) % Encode the data
```

[Huffman_Demo_Ex1]

[Ex. 2.31] Huffman Coding in MATLAB

```
codebook =

    [1]    '0'
    [2]    '1  0'
    [3]    '1  1  1'
    [4]    '1  1  0'
```

Remark: The codewords can be different from our answers found in the lecture notes.

```
sourceString =

     1     4     4     1     3     1     1     4     3     4


encodedString =

     0  1  1  0  1  1  0  0  1  1  1  0  0  1  1  0  1  1  1  1  1  0
```


[Huffman_Demo_Ex1]

[Ex. 2.31] Huffman Coding in MATLAB

```
codebook =

    [1]    '0'
    [2]    '1  0'
    [3]    '1  1  1'
    [4]    '1  1  0'
```

Remark: The codewords can be different from our answers found in the lecture notes.

Example 2.31.

x	$p_X(x)$	Codeword $c(x)$	$\ell(x)$
A	0.5	0	1
B	0.25	10	2
C	0.125	110	3
D	0.125	111	3

$\mathbb{E}[\ell(X)] = 1\times 0.5 + 2\times 0.25 + 3\times 0.125 + 3\times 0.125 = 1.75$ bits/symbol

[Huffman_Demo_Ex1]

Digital Communication Systems
EES 452

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

2. Source Coding

2.2 Optimal Source Coding:
Huffman Coding: More Examples and Its Optimal Property

[Ex. 2.32] Huffman Coding in MATLAB

```
pX = [0.4 0.3 0.1 0.1 0.06 0.04]; % pmf of X
SX = [1:length(pX)];              % Source Alphabet
[dict,EL] = huffmandict(SX,pX);   % Create codebook

%% Pretty print the codebook.
codebook = dict;
for i = 1:length(codebook)
    codebook{i,2} = num2str(codebook{i,2});
end
codebook

EL
```

```
>> Huffman_Demo_Ex2

codebook =

    [1]    '1'
    [2]    '0  1'
    [3]    '0  0  0  0'
    [4]    '0  0  1'
    [5]    '0  0  0  1  0'
    [6]    '0  0  0  1  1'


EL =

    2.2000
```


[Huffman_Demo_Ex2]

[Ex. 2.32] Huffman Coding in MATLAB

Example 2.32.

x	$p_X(x)$	Codeword $c(x)$	$\ell(x)$
'a'	0.4	0	1
'b'	0.3	10	2
'c'	0.1	110	3
'd'	0.1	1110	4
'e'	0.06	11110	5
'f'	0.04	11111	5

$\mathbb{E}[\ell(X)] = 1\times0.4 + 2\times0.3 + 3\times0.1 + 4\times0.1 + 5\times0.06$

$= 2.2$ bits/symbol

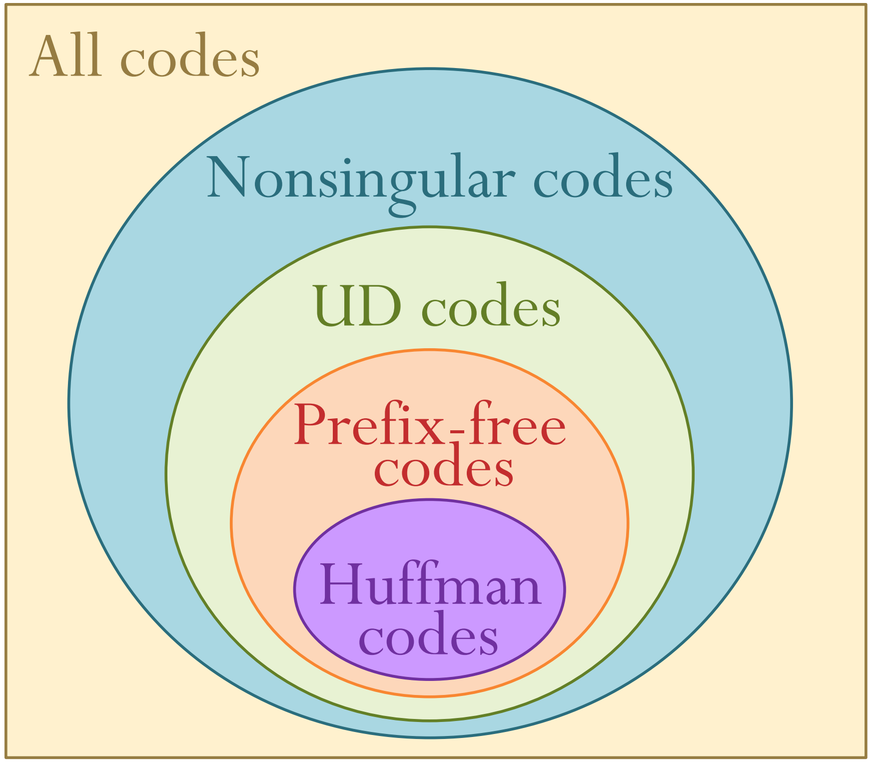
```
>> Huffman_Demo_Ex2

codebook =

    [1]    '1'
    [2]    '0   1'
    [3]    '0   0   0   0'
    [4]    '0   0   1'
    [5]    '0   0   0   1   0'
    [6]    '0   0   0   1   1'


EL =

    2.2000
```

The codewords can be different from our answers found earlier.

The expected length is the same.

[Huffman_Demo_Ex2]

Summary:



- [2.16-17] A good code must be **uniquely decodable (UD)**. [Defn 2.18]
  - Difficult to check.
- [2.24] Consider a special family of codes: **prefix(-free) code**.
  - Always UD.
  - Same as being **instantaneous**.

No codeword is a prefix of any other codeword.

Each source symbol can be decoded as soon as we come to the end of the codeword corresponding to it

- [Defn 2.30] **Huffman**'s recipe
  - Repeatedly combine the two least-likely (combined) symbols.
  - Automatically give prefix-free code.
- [2.37] For a given source's pmf, Huffman codes are ***optimal*** among all UD codes for that source. [Defn 2.36]

Summary: Optimality of Huffman Codes

Consider a given DMS with known pmf $p_X(x)$…

- [Defn 2.36] A code is **optimal** if it is UD and its corresponding expected length is the shortest among all possible UD codes for that source.
- [2.37] **Huffman codes** are **optimal**.