Fixed Length code & Variable Length Code:

If we have a finite set of symbols x_i, $i = 1, 2, 3 \ldots n$, Then the number of bits N required for unique coding when n is a power of 2,

$$N = \log_2 n \quad \text{and}$$

when n is not a power of 2,

$$N = (\log_2 n) + 1.$$

In generally ; $N \geq \log_2 n$.

eg. To encode the letters of the English alphabet, we use

$$N \geq \log_2 26$$

$$N \geq 4.7$$

$$N = 5 \text{ bits needed}$$

In a FLC all the symbols are equally important or equiprobable and hence each one requires 5 bits for representation.

If we consider English alphabet, some letters are less common (x, q, z etc) and some are frequently used (s, t, e etc). Represent more frequently occuring letters by fewer number of bits and lesser frequently occuring ones by larger number of bits. This method is called Variable length coding. When source symbols are not equally probable, more efficient method is VLC.

eg:

Source symbol	code F	Code G	Code H.
A	000	00	0
B	001	010	1
C	010	011	00
D	011	100	01
E	100	101	10
F	101	110	11
G	110	1110	000
H	111	1111	111

Code the sentence : "A BAD CAB"

FLC : code F : '000 001000 011 010 000 001' Total bits : 21

VLC : { code G : '00 010 00 100 011 00 010' " : 18

Code H : '0 1 0 01 00 01' " : 9

Problem with code H : For decoding we are able to regroup it using any manner i.e it is not uniquely decodable.

(01) (00) (10) (00) (1)

D C E C B

or (0)(1) (0)(0)(1)(0) (0) (0) (1)

A B A A B A A A B.

Kraft Inequality (Kraft - McMillan Inequality):

A necessary and sufficient condition for the existance of an instantaneous code with code word lengths $l_1, l_2, \ldots l_q$ is that

$$\sum_{i=1}^{q} r^{-l_i} \leq 1 \quad \text{when}$$

$r \rightarrow$ Number of different symbols used.

$l_i \rightarrow$ word length in binary digits of the codeword corresponding to ith source symbol.

$q \rightarrow$ number of source symbols.

For binary codes, we have $r = 2$, So Kraft inequality for binary code,

$$\sum_{i=1}^{q} 2^{-l_i} \leq 1.$$

Proof :

The word lengths $l_1, l_2 \ldots l_q$ are arranged in ascending order $l_1 \leq l_2 \leq l_3 \ldots \leq l_q$.

If we choose "one-length" codewords for all q source symbols, satisfying prefix property, then

$$q \leq r.$$

If $q > r$, go for combinations of 'r' symbols to form instantaneous codewords.

Let 'n_i' represents the number of messages encoded into codewords of length 'i'.

for $i = 1$; $n_1 \leq r$.

for $i = 2$; for getting an instantaneous code, we must start encoding using $(r - n_1)$ symbols only as the first digits and the second digit can be any of 'r' symbols of the code alphabet.

$$n_2 \leq (r - n_1) r$$

$$n_2 \leq r^2 - n_1 r.$$

for $i = 3$, The number of msgs to be encoded to 3 digits which has to be different from code-words corresponding to n_1 and n_2 number of msgs must satisfy.

$$n_3 \leq \left[(r^2 - n_1 r) - n_2\right] \cdot r$$

$$n_3 \leq r^3 - n_1 r^2 - n_2 r$$

Proceeding this way, we can write.

for ith codeword ;

$$n_i \leq r^i - n_1 r^{(i-1)} - n_2 r^{(i-2)} \ldots \ldots - n_{(i-1)} r.$$

Then

$$n_i + n_1 r^{(i-1)} + n_2 r^{(i-2)} + \ldots \ldots + n_{(i-1)} r \leq r^i$$