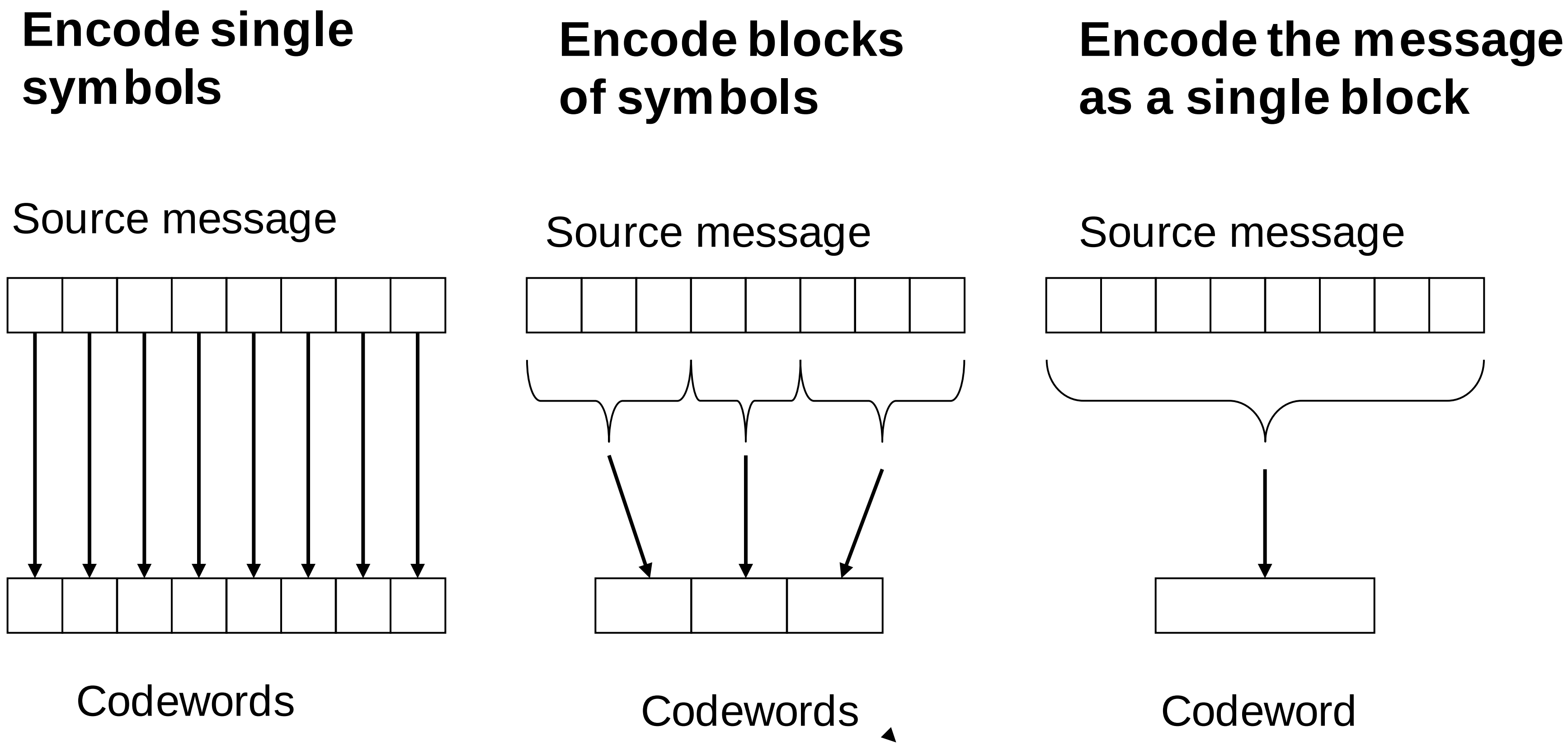
2. Coding-Theoretic Foundations

- **Source alphabet *S* $\rightarrow$ Target alphabet {0, 1}**
- **Categories of source encoding:**

1. Codebook methods: Symbols (*S*) $\rightarrow$ Codewords (*W*)
- Implicit / explicit codebook
- Static / dynamic codebook
- Original/extended alphabet

2. Global methods:
- The whole message is transformed into a single computed codeword.
- No explicit correspondence between source symbols and code bits .

Illustration of coding approaches



Requirements of a codebook

- Uniqueness: $s_i \neq s_j \;\Rightarrow\; C(s_i) \neq C\,(s_j)$
  Sufficient for fixed-length codes
- Length indication: required for variable-length codes. Alternatives:
  - Length prefixes the actual code (but length must also be coded …)
  - ‘Comma’ = special bit combination indicating the end
  - Carefully selected ‘self-punctuative’ codewords
- Example of an ill-designed codebook:

  ‘a’ = 0
  ‘b’ = 01
  ‘c’ = 11
  ‘d’ = 00

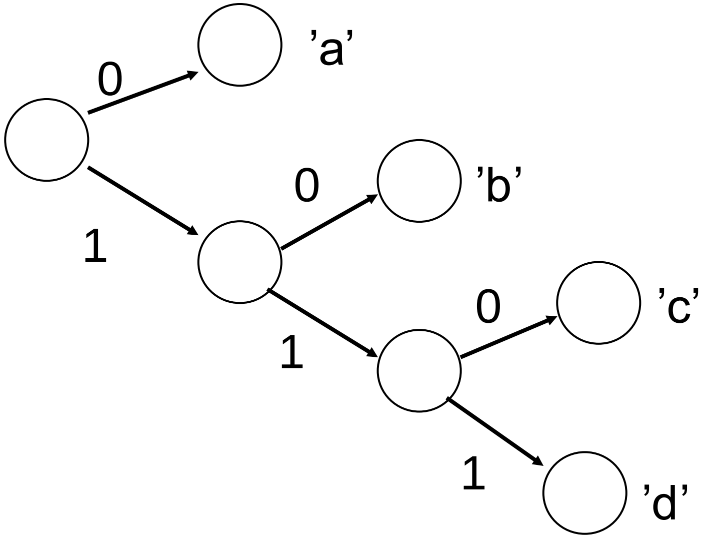
  Code string 00110 results from either ‘dca’ or ‘aaca’

Graphical representation of a codebook

Decoding tree (decision tree)

- Left son = bit 0, right son = bit 1
- Prefix-free code: Binary tree (usually complete); each symbol is represented by a path from the root to a leaf, e.g.:

Code table:
’a’ = 0
’b’ = 10
’c’ = 110
’d’ = 111



Properties of codes

- **Some general codebook categories:**
  - *Uniquely decodable* (decipherable) code
  - *Prefix-free* code (= prefix code)
  - *Instantaneous* code

- **Kraft inequality:** An instantaneous codebook *W* exists if and only if the lengths $\{l_1, ..., l_q\}$ of codewords satisfy

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

- **MacMillan inequality:** Same as Kraft inequality for any uniquely decodable code.
- **Important:** Instantaneous codes are sufficient.

Infinite, countable alphabet

- E.g. Natural numbers 1, 2, 3, … without limit
- No explicit codebook
- Codes must be determined algorithmically
- Requirements:
  - *Effectiveness*: There is an effective procedure to decide, whether a given codeword belongs to the codebook or not.
  - *Completeness*: Adding a new code would make the codebook not uniquely decipherable.

Application of coding arbitrary numbers

- ***Run-length coding*** (Finnish: ‘välimatkakoodaus’):
  Binary string, 0’s and 1’s *clustered*, cf. black&white images
- Two possible numberings:
  - Alternating runs of 0 and 1

$$\underbrace{0\ 0\ 0}_{3}\ \underbrace{1\ 1}_{2}\ \underbrace{0\ 0\ 0\ 0}_{4}\ \underbrace{1}_{1}\ \underbrace{0}_{1}\ \underbrace{1}_{1}\ \underbrace{0\ 0}_{2}\ \underbrace{1\ 1\ 1}_{3}$$

  - Runs of 0’s ending at 1:

$$\underbrace{0\ 0\ 0\ 1}_{3}\ \underbrace{1}_{0}\ \underbrace{0\ 0\ 0\ 0\ 1}_{4}\ \underbrace{0\ 1}_{1}\ \underbrace{0\ 0\ 1}_{2}\ \underbrace{1}_{1}\ \underbrace{1}_{1}$$

Encoding of natural numbers (P. Elias)

- α-*coding*: Unary code; not efficient enough (not *universal*).
- β-*coding*: Normal positional representation + end symbol (*ternary* alphabet).
- γ-*coding*: Positional representation with α-coded length as prefix.
- δ-*coding*: Positional representation with γ-coded length as prefix.
- Ω-coding: Recursive representation of lengths

Example codings of natural numbers

Number	α-code	β-code	γ-code	δ-code
1	1	11	1	1
2	01	011	010	0100
3	001	1011	011	0101
4	0001	0011	00100	01100
5	00001	01011	00101	01101
6	000001	10011	00110	01110
7	0000001	101011	00111	01111
8	00000001	00011	0001000	00100000
…	…	…	…	…

Example of robust universal coding

- *Zeckendorf* coding (called also *Fibonacci* coding)
- Number representation using a Fibonacci 'base'

Number	**Weights for**					**Code**
	8	**5**	**3**	**2**	**1**	
1	0	0	0	0	1	11
2	0	0	0	1	0	011
3	0	0	1	0	0	0011
4	0	0	1	0	1	1011
5	0	1	0	0	0	00011
6	0	1	0	0	1	10011
7	0	1	0	1	0	01011
8	1	0	0	0	0	000011
9	1	0	0	0	1	100011
10	1	0	0	1	0	010011
11	1	0	1	0	0	001011
12	1	0	1	0	1	101011

Semi-fixed-length codes for numbers x

- Called also reduced block coding

x	$q=2$	$q=3$	$q=4$	$q=5$	$q=6$	$q=7$	$q=8$
0	0	0	00	00	00	00	000
1	1	10	01	01	01	010	001
2		11	10	10	100	011	010
3			11	110	101	100	011
4				111	110	101	100
5					111	110	101
6						111	110
7							111

Golomb and Rice codes

- Examples of parametric codes for numbers

x	Golomb $m = 3$	Rice $m = 2$
0	0 0	00
1	0 10	01
2	0 11	100
3	10 0	101
4	10 10	1100
5	10 11	1101
6	110 0	11100
7	110 10	11101
8	110 11	111100
9	1110 0	111101