UNIVERSITÄT
LEIPZIG

# Asymmetric Numeral System

December 30, 2024

Huo Jiang

Fakultät für Mathematik und Informatik
Universität Leipzig

# Outline

1. Introduction
2. Entropy Coding
3. Asymmetric Numeral System
4. rANS
5. Streaming-rANS
6. Summary
7. Reference

# Introduction

# Introduction

- Hi there, this is a self inspired latex beamer template for Leipzig university.

# Entropy Coding

## Entropy Coding

- The theoretical limit of compression is given by Shannon entropy, which measures the minimum average number of bits needed to encode a symbol from a data source.
- Lossless compression algorithms aim to approach this limit but cannot go below it.
- The entropy depends on the probability distribution of the symbols, with more skewed distributions being more compressible.
- Shannon Entropy [1]

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$$

- Shannon's Source Coding Theorem [1]

$$N \cdot H(X)$$

# Asymmetric Numeral System

## Motivation

ANS combines the compression ratio of arithmetic coding (which uses a nearly accurate probability distribution), with a processing cost similar to that of Huffman coding [4].

## Basic Idea

- Symbol spread function

## Basic Idea

- Symbol spread function
- Representation of information
  (represents an entire message as a single number, similar to Arithmetic Coding) –> example: representation of {2,0,2,5,1,8}

## Basic Idea

- Symbol spread function
- Representation of information
  (represents an entire message as a single number, similar to Arithmetic Coding) –> example: representation of {2,0,2,5,1,8}
- Symmetric or Asymmetric

## Basic Idea

- Symbol spread function
- Representation of information
  (represents an entire message as a single number, similar to Arithmetic Coding) –> example: representation of {2,0,2,5,1,8}
- Symmetric or Asymmetric
- rANS and tANS

## Symbol spread function

- the function that determines the split of the $\mathbb{N}$ subsets corresponding to different symbols [1]

## Symbol spread function

- the function that determines the split of the $\mathbb{N}$ subsets corresponding to different symbols [1]
- $\bar{s} : \mathbb{N} \rightarrow \mathcal{A}, \bar{s}(x) = \text{mod}(x, b) = s$

## Symbol spread function

- the function that determines the split of the $\mathbb{N}$ subsets corresponding to different symbols [1]
- $\bar{s} : \mathbb{N} \to \mathcal{A}, \bar{s}(x) = \text{mod}(x, b) = s$
- In the standard binary system example: $\bar{s}(x) = \text{mod}(x, 2)$
  e.g. splits natural numbers into subsets of even/odd numbers [2]

## Symbol spread function (Exa. 1)

$S = \{0, 1\}$

### Exa. 1: Standard Binary System

| N | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| s=0 | | | | | | | | | | | | | | | | | | | |
| s=1 | | | | | | | | | | | | | | | | | | | |

**Fig.1**

## Symbol spread function (Exa. 2)

$S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$x' = 10x + s$

| $\mathbb{N}$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | | 202 | | 2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $s = 0$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 1$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 2$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 3$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 4$ | | | | | | | | | | | | | | | | | | | | | | ... | | ... | |
| $s = 5$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 6$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 7$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 8$ | | | | | | | | | | | | | | | | | | | | | | | | | |
| $s = 9$ | | | | | | | | | | | | | | | | | | | | | | | | | |

**Fig.2**

## Representation of information (Exa. 2)

- a finite sequence of symbols/digits:

$$\mathcal{A} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

## Representation of information (Exa. 2)

- a finite sequence of symbols/digits:

$$\mathcal{A} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

- the sequence of symbols/digits to be encoded:

$$S_{in} = (2, 0, 2, 5)$$

## Representation of information (Exa. 2)

- a finite sequence of symbols/digits:

$$\mathcal{A} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

- the sequence of symbols/digits to be encoded:

$$S_{in} = (2, 0, 2, 5)$$

- How can we represent this sequence as a single number?

## Representation of information (Exa. 2)

- a finite sequence of symbols/digits:

$$\mathcal{A} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

- the sequence of symbols/digits to be encoded:

$$S_{in} = (2, 0, 2, 5)$$

- How can we represent this sequence as a single number?
- The easiest way is to concatenate the symbols/digits:

$$x_{out} = 2025$$

## Representation of Information (Exa. 2)

- encoding the sequence of digits (2,0,2,5,1,8)

$$
\begin{aligned}
x_{\text{init}} &= 0 \\
x_1 &= x_{\text{init}} \times 10 + 2 = 2 \\
x_2 &= x_1 \times 10 + 0 = 20 \\
x_3 &= x_2 \times 10 + 2 = 202 \\
x_{\text{out}} &= x_3 \times 10 + 5 = 2025
\end{aligned}
$$

## Representation of Information (Exa. 2)

– What is the encoding function?

## Representation of Information (Exa. 2)

- What is the encoding function?
- Encoding function

$$x_{i+1} = x_i \times 10 + s_i$$

## Representation of Information (Exa. 2)

- What is the encoding function?
- Encoding function

$$x_{i+1} = x_i \times 10 + s_i$$

- x' is the x-th appearance of symbol s

$$x' = C(s, x) = x \cdot 10 + s$$

# Representation of Information (Exa. 2)

$x' = 10x + s$

$s = 0$ $s = 2$ $s = 2$ $s = 5$

| $\mathbb{N}$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | | 202 | | 2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $s = 0$ | 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | |
| $s = 1$ | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | | |
| $s = 2$ | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | |
| $s = 3$ | | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | |
| $s = 4$ | | | | | 0 | | | | | | | | | | 1 | | | | | | | ... | | ... | |
| $s = 5$ | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | | |
| $s = 6$ | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | |
| $s = 7$ | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | |
| $s = 8$ | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | |
| $s = 9$ | | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | |

**Fig.3**

# Representation of Information (Exa. 1)

**Exa. 1: Standard Binary System**

**X' = 2X + s**

s=1 s=0 s=0 s=1

| X' | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X s=0 | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 |
| X s=1 | | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | |

**Fig.4**

# **Asymmetric Numeral System** | Representation of information |

**Exa. 1: Standard Binary System**

**X' = 2X + s**

s=1 s=0 s=0 s=1

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X' | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| x s=0 | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 |
| x s=1 | | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | |

**Exa. 2: Standard Decimal System**

**X' = 10X + s**

s=2 s=0 s=2 s=5

| N | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | | 202 | | 2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| s=0 | 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | |
| s=1 | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | | |
| s=2 | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | |
| s=3 | | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | |
| s=4 | | | | | 0 | | | | | | | | | | 1 | | | | | | | ... | | ... | |
| s=5 | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | | |
| s=6 | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | |
| s=7 | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | |
| s=8 | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | |
| s=9 | | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | |

**Fig.5**

# **Asymmetric Numeral System** | Symmetric or Asymmetric |

**Exa. 1: Standard Binary System**

**X' = 2X + s**

s=1 s=0 s=0 s=1

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X' | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| X s=0 | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 |
| X s=1 | | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | |

**Exa. 2: Standard Decimal System**

**X' = 10X + s**

s=2 s=0 s=2 s=5

| N | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | | 202 | | 2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| s=0 | 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | |
| s=1 | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | | |
| s=2 | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | | |
| s=3 | | | | 0 | | | | | | | | | | 1 | | | | | | | | | | | |
| s=4 | | | | | 0 | | | | | | | | | | 1 | | | | | | | ... | | ... | |
| s=5 | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | | |
| s=6 | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | | |
| s=7 | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | | |
| s=8 | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | | |
| s=9 | | | | | | | | | | 0 | | | | | | | | | | 1 | | | | | |

**Fig.6**

## Exa. 3: Asymmetric Binary System

- Let $\mathcal{A} = \{0, 1\}$
- $Pr(0) = 1/4, Pr(1) = 3/4$

## Exa. 3: Asymmetric Binary System

- Let $\mathcal{A} = \{0, 1\}$
- $Pr(0) = 1/4, Pr(1) = 3/4$
- a cycle of length 4, which contains 1 zero and 3 ones

## Exa. 3: Asymmetric Binary System

- Let $\mathcal{A} = \{0, 1\}$
- $Pr(0) = 1/4, Pr(1) = 3/4$
- a cycle of length 4, which contains 1 zero and 3 ones
- $x' = C(s, x) \approx x/p_s$

# Exa. 3: Asymmetric Binary System

**Exa. 3: Asymmetric Binary System**

$x' \approx x/Pr(s)$ | $Pr(0) = 1/4, Pr(1) = 3/4$

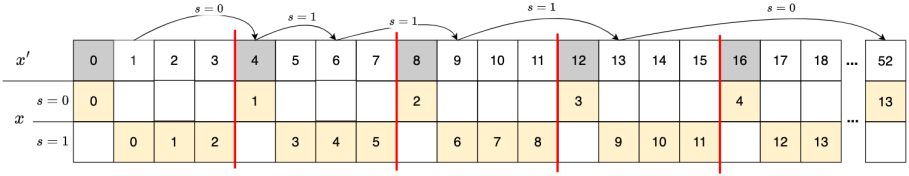



**Fig.7**

We refine even/odd number by changing their densities

## Symmetric or Asymmetric

- Symmetric: uniform distribution
- Asymmetric: non-uniform distribution

## tANS

- d

## Performance

- f

# rANS

## rANS

- ”r” = ”range”: similar to range coding (RC)
- symbol spread function
- representation of information
- asymmetric

## Exa.3

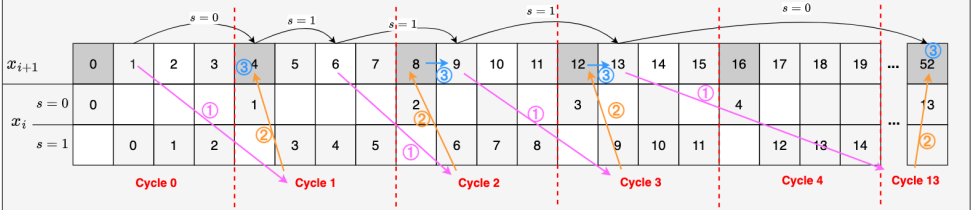
How to encode 101110? What is the encoding function $C(s, x)$?

$x' = C(s, x) \approx x/p_s$

## Some notations

- $\mathcal{A} = \{s_1, s_2, \ldots, s_k\}$ is the alphabet set
- $s$ is the symbol, if it is used in subscript: $s$ is the current symbol, $s-1$ is the previous symbol, $s+1$ is the next symbol
- $F_s$ is the absolute frequency of symbol $s$
- $m = \sum_{s \in \mathcal{A}} F_s$ is the total frequency
- $c_s = \sum_{s \in \mathcal{A}} F_{s-1}$ is the cumulative frequency ($c_{s+1}$ is the cumulative frequency of the next symbol)
- $p_s = F_s / m$ is the probability of symbol $s$
- $x_i$ is the current state (next state: $x_{i+1}$, previous state: $x_{i-1}$)

$x_{i+1} \approx x_i / Pr(s)$ | $Pr(0) = 1/4, Pr(1) = 3/4$




Given current state $x_i$ and next symbol $s$. How to find next state $x_{i+1}$, which is the $x_i$-th appearance of symbol $s$ ?

| STEPS / EXAMPLE | $Pr(0) = 1/4, Pr(1) = 3/4$ | | General |
|---|---|---|---|
| | $s = 0$ | $s = 1$ | |
| Step ①: Find cycle number $k$ | $k = \lfloor x_i/1 \rfloor$ | $k = \lfloor x_i/3 \rfloor$ | $k = \lfloor x_i/F_s \rfloor$ |
| Step ②: Navigate to the beginning (first number) of that cycle | $4 \cdot k$ | $4 \cdot k$ | $m \cdot k$ |
| Step ③: Find the position of the symbol $s$ in that cycle | part 1: first of one<br>+ 0 | part 1:<br>+ mod$(x_i, 3)$ | part 1:<br>+ mod$(x_i, F_s)$ |
| - part 1: position of the symbol $s$ in the sequence of the same symbol<br>- part 2: how many other symbols before the symbol $s$ | part 2: no other symbols<br>+ 0 | part 2: only one other symbol (0) and with one of that symbol in one cycle<br>+ 1 | part 2: only one other symbol (0) and with one of that symbol in one cycle<br>$+c_s$ |
| | $x_{i+1} = m \cdot k$ | $x_{i+1} = 4 \cdot \lfloor x_i/3 \rfloor + \text{mod}(x_i, 3) + 1$ | $x_{i+1} = m \cdot \lfloor x/F_s \rfloor + \text{mod}(x, F_s) + c_s$ |

$$x_{i+1} = C(s, x_i) = m \cdot \lfloor x_i/F_s \rfloor + \text{mod}(x_i, F_s) + c_s$$

**Fig.8**

## Encoding Function [2] [5]:

$$x_{i+1} = m \cdot \lfloor x_i / F_s \rfloor + \mathrm{mod}(x_i, F_s) + c_s$$

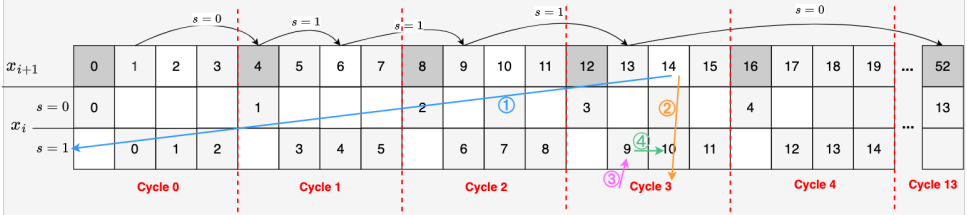
## implementation with python (dash app)

```python
def rANS_encoder(symbol_counts, s_input):
    cumul_freq = [] # c_s
    sum_freq = 0 # m
    x = 0 # initial state
    output = [] # list to store encoded (symbols, state)

    for count in symbol_counts:
        cumul_freq.append(sum_freq)
        sum_freq += count
    for s in s_input:
        F_s = symbol_counts[s]
        c_s = cumul_freq[s]
        x = sum_freq * (x // F_s) + (x % F_s) + c_s
        output.append((s, x))

    L_avg = math.ceil(math.log(x) / math.log(2.0)) / len(s_input)
    return output, x, L_avg, entropy(symbol_counts)
```

$x_{i+1} \approx x_i / Pr(s)$ $\boxed{Pr(0) = 1/4, Pr(1) = 3/4}$




Given current state $x_i$. How to find previous state $x_{i-1}$, where $x_i$ is the $x_{i-1}$-th appearance of symbol $s$ ?

| STEPS / EXAMPLE | $Pr(0) = 1/4, Pr(1) = 3/4$ | | General |
|---|---|---|---|
| | $s = 0$ | $s = 1$ | |
| Step ① : Find the symbol $s$ for the state $x_i$ | if $x_i \bmod 4 = 0$ | otherwise or if $x_i \bmod 4 = 1, 2, 3$ | $s$ such that $c_s \leq \mathrm{mod}(x_i, m) < c_{s+1}$ |
| Step ②: Find the cycle $k$ of the current state $x_i$ | $k = x_i/4$ | $k = \lfloor x_I/4 \rfloor$ | $k = \lfloor x_i/m \rfloor$ |
| Step ③: Find the first number of that symbol $s$ in that cycle $k$ | $x_i/4 \cdot 1$ | $\lfloor x_i/4 \rfloor \cdot 3$ | $\lfloor x_i/m \rfloor \cdot F_s$ |
| Step ④: Allocate to the exact position | $x_{i-1} = x_i/4 \cdot 1 + 0 - 0$ | $x_{i-1} = \lfloor x_i/4 \rfloor \cdot 3 + 2 - 1$ | $x_{i+1} = \lfloor x_i/m \rfloor \cdot F_s + \mathrm{mod}(x_i, F_s) - c_s$ |

1. $s$ such that $c_s \leq \mathrm{mod}(x_1, m) < c_{s+1}$
2. $x_{i-1} = D(x_i) = \lfloor x_i/m \rfloor \cdot F_s + \mathrm{mod}(x_i, F_s) - c_s$

**Fig.9**

## Decoding Function [2] [5]:

- $s$

$$s \text{ such that } c_s \leq \text{mod}(x_i, m) < c_{s+1}$$

- $x_{i-1}$

$$x_{i-1} = D(x_i) = \lfloor x_i / m \rfloor \cdot F_s + \text{mod}(x_i, F_s) - c_s$$

## implementation with python (dash app)

```python
def rANS_decoder(symbol_counts, x):
    cumul_freq = []  # c_s
    sum_freq = 0 # m
    output = []  # List to store decoded symbols and states
    # compute m and c_s
    for count in symbol_counts:
        cumul_freq.append(sum_freq)
        sum_freq += count
    while x > 0:
        slot = x % sum_freq  # Find the slot in the cumulative frequency range
        s = bisect_right(cumul_freq, slot) - 1 # Binary search to find the symbol s (slot).
        F_s = symbol_counts[s]          # Frequency of the symbol
        c_s = cumul_freq[s]             # Cumulative count of the symbol
        output.append((s,x))            # Append decoded symbol to the output
        x = (x // sum_freq) * F_s + x % sum_freq - c_s  # Update the state
    return output, state
```

# Streaming-rANS

## Streaming-rANS

To avoid using large number arithmetic, in practice stream variants are used: which enforce by renormalization: sending the least significant bits of x to or from the bitstream (usually L and b are powers of 2). [4]

Example of the problem //TODO: add example

- ANS (Asymmetric Numeral Systems) uses a single integer x, known as the "state," to represent the encoded data.
- Over time, as more symbols are encoded, the state x can grow arbitrarily large. This can lead to:
  - Overflow: If the value exceeds the precision limits of your system (e.g., a 32-bit or 64-bit register).
  - Inefficient Encoding: A larger x means you need more memory or bits to represent it.
- Using the C function from the previous section to encode succeeding symbols, we can encode a sequence into a very large natural number $x$, which can be stored as a length $\lfloor lg(x) \rfloor + 1$ bit sequence. [2]

## Renormalization in Range Coding

- Range Coding
  - 1
  - 2

## Renormalization in Range Coding

- Range Coding
  - 1
  - 2
- we use the similar idea for rANS
  - range $[L, H]$
  - Encoding: shrink the current state $x$ until it is in the range $[L, H]$
  - Decoding: expand the current state $x$ until it is in the range $[L, H]$

## Methods from Jarek Duda: Method 1 - Decoding [2]

**ANS decoding step from state *x***

$(s, x) = D(x)$ *# decode the current state x*

useSymbol($s$)

**while** $x < L$ **do** *# if x is too small that not in the range [L,H]*

*# expand the x by reading the outstreamed digits back from the stream (an array)*

expand$(x, m) = m \cdot x +$ readDigit$() \rightarrow x$

**end while**

*# return the new state*

## What is $L, H$ ?

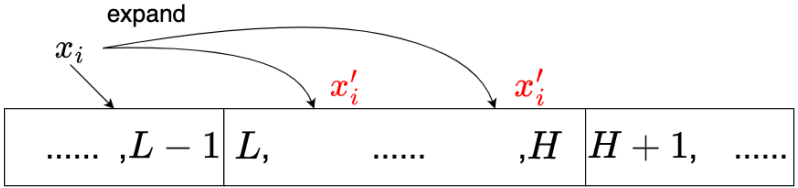



**Fig 11**

## What is $L, H$ ?

- $x_i = m \cdot x_{i-1} +$ bit
- Either $x_i$ in $[L, H]$ or $x_{i-1}$ in $[L, H]$. Assume $x_{i-1}$ in $[L, H]$.
  - $x_{i-1} >= L$
  - $x_i >= m \cdot L + bit >= m \cdot L$
  - $x_i$ can't be in $[L, H]$
  - derive: $H <= m \cdot L - 1$
- $[L, H]$, where $H <= m \cdot L - 1$ (Constraint 1)
- $I = \{L, ..., b \cdot L - 1\}$, which is defined by Jarek Duda $b$-unique interval and $L \in \mathbb{N}$
- This will ensure that the loop in Method 1 will always return to the original value

## Methods from Jarek Duda: Method 2 - Encoding [2]

shrink the current state $x$ by stream out the least significant bit ($x$ will be stored in binary and we take out the bit from right side)

**ANS encoding step of symbol sfrom state $x$**

*# once the current state x is greater than max value that we define later*

**while** $x > maxX[s]$ **do**

*# write the least significant digit to the stream (stream could be an array in implementaion )*

writeDigit(mod($x, m$))

shrink$(x, m) = \lfloor x/m \rfloor \rightarrow x$ *# shrink the x*

**end while**

$x = C(s, x)$ *# encode with the shrunk x*

## What is $maxX[s]$?

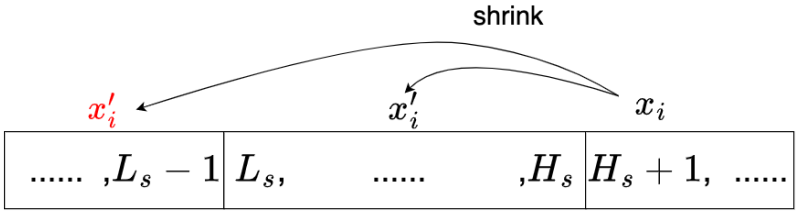



**Fig 10**

## What is $maxX[s]$?

- to make sure not to jump over the range
- Assume that $x_i = H_s + 1$ and it will be shrinked to $x'_i = \lfloor x_i/m \rfloor$
- $x'_i = \lfloor x_i/m \rfloor = \lfloor (H_s + 1)/m \rfloor >= L_s$
- $L_s <= (H_s + 1)/m <= \lfloor (H_s + 1)/m \rfloor < (H_s + 1)/b + 1$
- $H_s >= m \cdot L_s - 1, \forall s \in \mathcal{A}$ (Constraint 2)

## Final Form of $[L, H]$ in practice

- Contraints

$$[L, H], \text{ where } H <= m \cdot L - 1$$
$$H_s >= m \cdot L_s - 1, \forall s \in \mathcal{A}$$

- In practice

$$\mathcal{I} = [l \cdot m, 2l \cdot m - 1]$$
$$\mathcal{I}_s = [l \cdot F_s, 2l \cdot F_s - 1] \text{ for } \forall s \in \mathcal{A}$$

```
for i in range(10):
    print("Hello, World!")
```

# Summary

## Summary

- To summarize, this may help you to make a Leipzig university beamer presentation.

# Reference

# Reference


1. C. Shannon, “A mathematical theory of communication,” The Bell Sys- tem Technical Journal, vol. 27, pp. 379–423, 623–656, July, October
2. J. Duda, K. Tahboub, N. J. Gadgil and E. J. Delp, ”The use of asymmetric numeral systems as an accurate replacement for Huffman coding,” 2015 Picture Coding Symposium (PCS), Cairns, QLD, Australia, 2015, pp. 65-69, doi: 10.1109/PCS.2015.7170048.
3. J. Duda, “Asymmetric numeral systems: entropy coding combining speed of huffman coding with compression rate of arithmetic coding,” arXiv:1311.2540.
4. Asymmetric numeral systems - Wikipedia
5. What is Asymmetric Numeral Systems? - Kedar Tatwawadi 1948.