

5. 二叉树

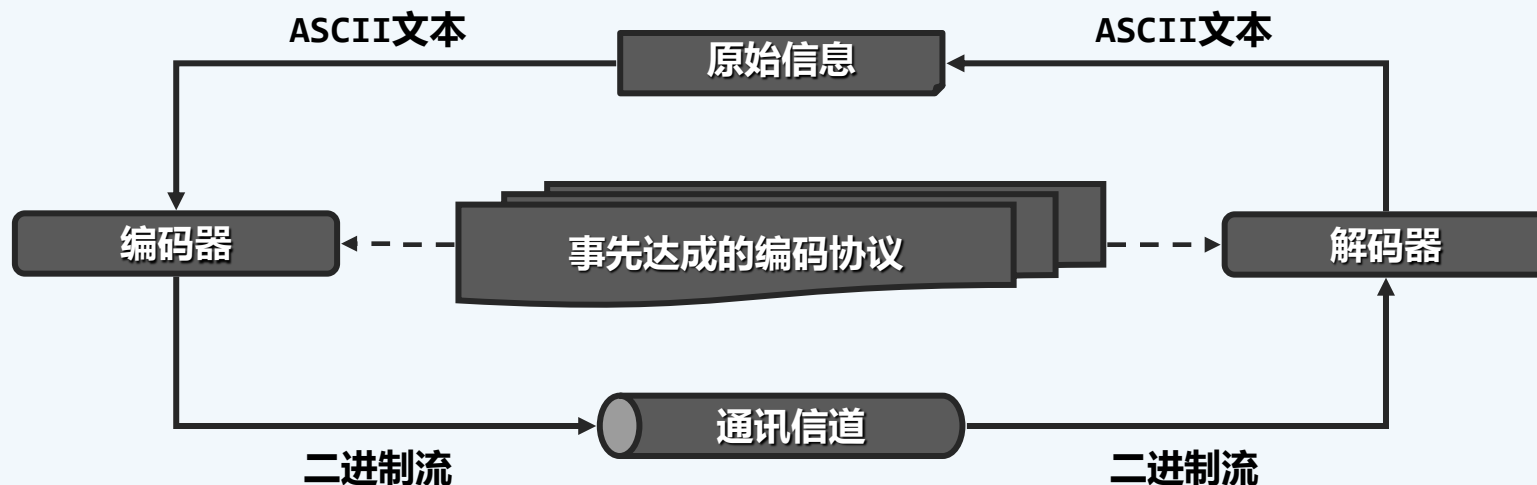
(f) PFC编码

邓俊辉

deng@tsinghua.edu.cn

问题

❖ 通讯 / 编码 / 译码



❖ 二进制编码

组成数据文件的字符来自字符集 Σ

字符被赋予互异的二进制串

❖ 文件的大小取决于

字符的数量 \times 各字符编码的长短

❖ 通讯带宽有限时

如何对各字符编码，使文件最小？

"1010₀₁₁00" = "MAIN"

M	A	I	N
1	010	011	00

二进制编码树

❖ 将 Σ 中的字符组织成一棵二叉树

以0、1表示左、右孩子

各字符 x 分别存放于对应的叶子 $v(x)$ 中

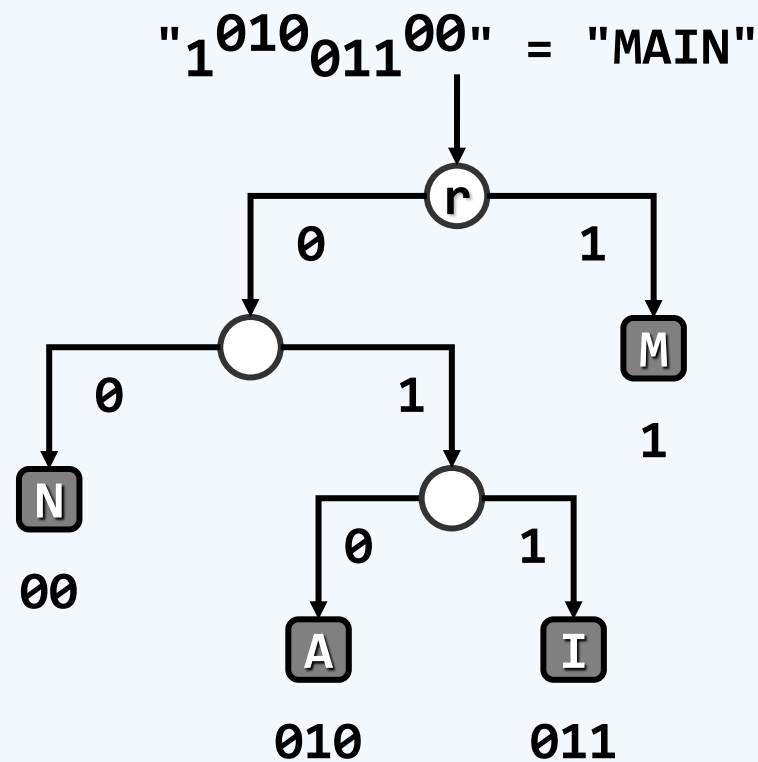
❖ 字符 x 的编码串 $\text{rps}(v(x)) = \text{rps}(x)$

由根到 $v(x)$ 的通路 (root path) 确定

❖ 优点：字符编码不必等长，且
不致出现解码歧义

❖ 这属于“前缀无歧义”编码 (prefix-free code)
不同字符的编码互不为前缀，故不致歧义

❖ 缺点：你能发现吗？



编码长度 vs. 叶节点平均深度

❖ $|rps(x)| = \text{depth}(v(x))$

❖ 编码总长 = $\sum_x \text{depth}(v(x))$

平均编码长度

$$= \sum_x \text{depth}(v(x)) / |\Sigma|$$

$$= \text{叶节点平均深度 } \text{ald}(T)$$

❖ 对于特定的 Σ

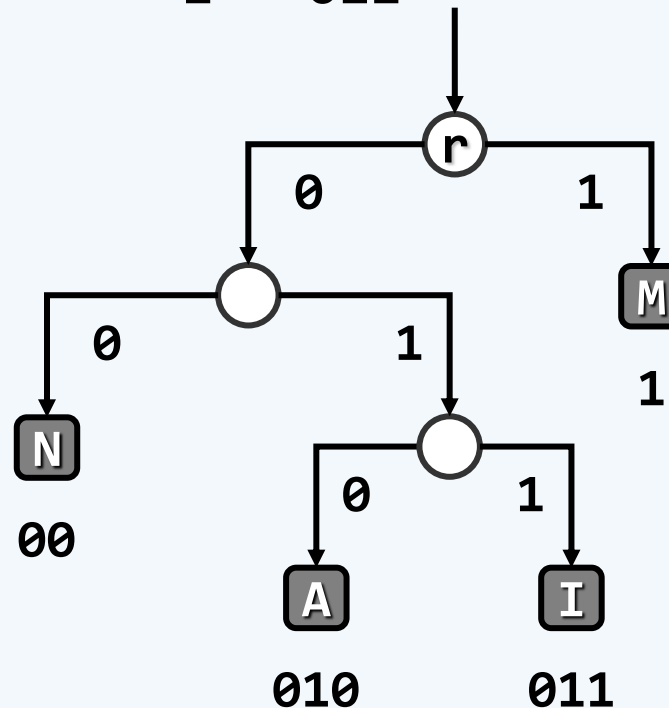
$\text{ald}()$ 最小者即为最优编码树 T_{opt}

❖ 最优编码树必然存在，但不见得唯一

它们具有哪些特征？

$$\text{ald}(T) * 4 = 2 + 3 + 3 + 1 = 9$$

$$"1^010_{011}00" = \text{"MAIN"}$$

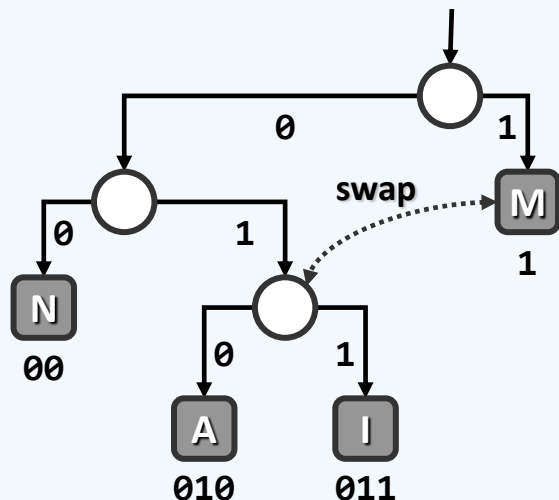


最优编码树

- ❖ $\forall v \in T_{\text{opt}}, \deg(v) = 0$ only if $\text{depth}(v) \geq \text{depth}(T_{\text{opt}}) - 1$
亦即，叶子只能出现在**倒数两层内**——否则，通过节点**交换**可以...
- ❖ 特别地，**真完全树**即是最优编码树
- ❖ 实际上，字符的**出现频率**不尽相同，例如 $w('E') \gg w('Z')$

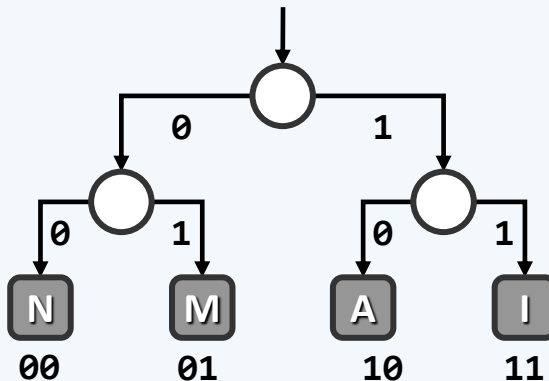
$$\text{ald} * 4 = 2 + 3 + 3 + 1 = 9$$

"101001100" = "MAIN"



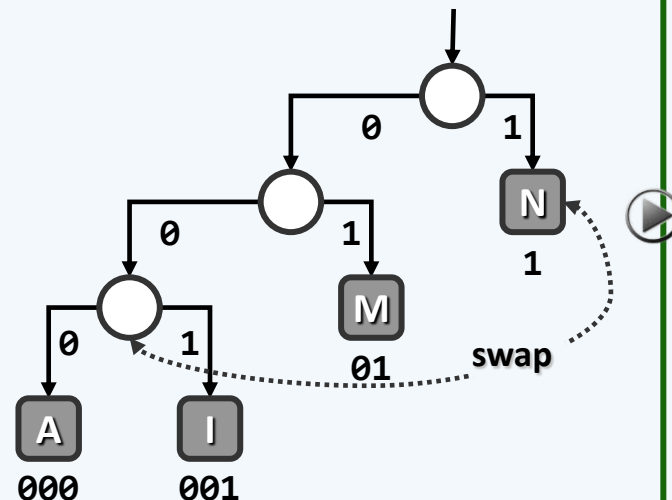
$$\text{ald} * 4 = 2 + 2 + 2 + 2 = 8$$

"01101100" = "MAIN"



$$\text{ald} * 4 = 2 + 3 + 3 + 1 = 9$$

"010000011" = "MAIN"



带权编码长度 vs. 叶节点平均带权深度

❖ 已知各字符的期望频率

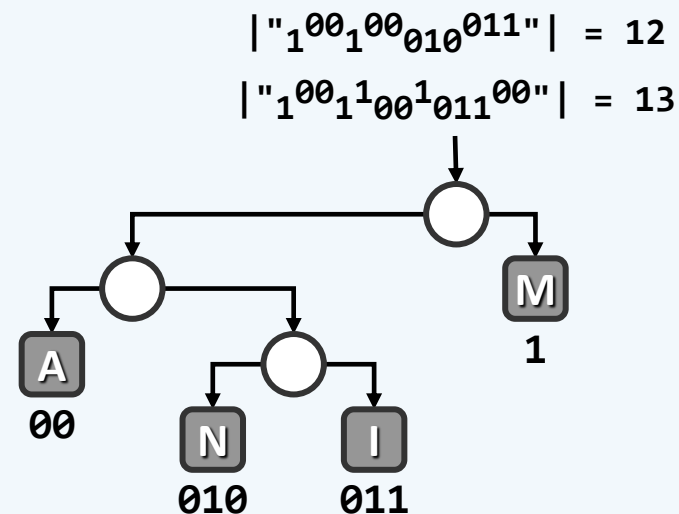
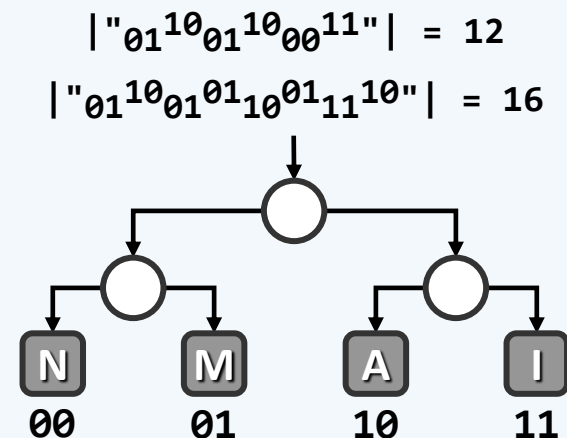
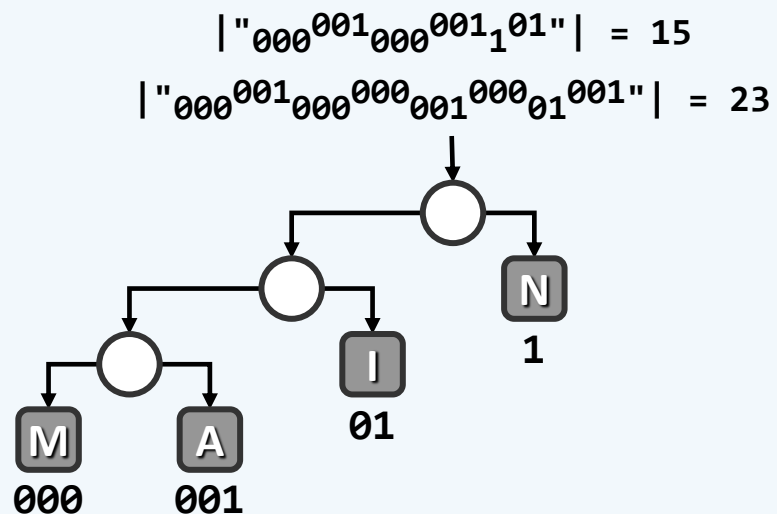
如何构造最优编码树？

❖ 文件长度 \propto 平均带权深度

$$= \text{wald}(T) = \sum_x \text{rps}(x) \times w(x)$$

❖ 此时，完全树**不见得**是最优编码树

比如，考查"mamani"和"mammamia"...



最优编码树

❖ 同样，频率高/低的（超）字符，应尽可能放在高/低处

❖ 故此，通过交换，同样可以缩短 $wald(T)$

