ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
School of Computer and Communication Sciences

Handout 13 Information Theory and Coding
Solutions to Homework 6 November 1, 2011

PROBLEM 1. Upon noticing $0.9^6 > 0.1$, we obtain $\{1, 01, 001, 0001, 00001, 000001, 0000001, 0000000\}$ as the dictionary entries.

PROBLEM 2.

(a) Note that with $l(u) = \lceil \log_2(1/q(u)) \rceil$ we have $2^{-l(u)} \leq q(u)$, and thus

$$\sum_u 2^{-l(u)} \leq \sum_u q(u)$$

As $q(u) = \sum_{u=1}^K \alpha_k p_k(u)$, we see that $\sum_u q(u) = \sum_k \alpha_k = 1$. Thus $l(u)$ satisfies Kraft's inequality and so a prefix-free code C with codewords lengths $l(u)$ exists.

(b) Since C is a prefix free code, its expected codeword length L_k is at least H_k and we can get $0 \leq L_k - H_k$. To upper bound $L_k - H_k$, note that since $\lceil x \rceil < x + 1$,

$$\begin{aligned} L_k(C) &= \sum_u p_k(u)\text{length}(C(u)) \\ &< \sum_u p_k(u)[1 + \log(1/q(u))]) \\ &= 1 + \sum_u p_k(u)\log(1/q(u)) \end{aligned}$$

Thus $L_k - H_k < 1 + \sum_u p_k(u)\log[(p_k(u)/q(u))]$. Observe now that $q(u) \geq \alpha_k p_k(u)$, thus $p_k(u)/q(u) \leq 1/\alpha_k$, and

$$L_k - H_k < 1 + \sum_u p_k(u)\log(1/\alpha_k) = 1 + log(1/\alpha_k).$$

(c) Choosing $\alpha_k = 1/K$ for each k, we get the desired conclusion.

(d) We can view the source as producing a sequence of 'supersymbols' each consisting of a block of L letters. Applying part (c) to this 'supersource', and noticing that the entropy of the supersymbols is $H(U_1, ..., U_L) = LH(U)$, we see that there is a prefix-free code for which

$$E_k[\text{number of bits to describe a supersymbol}] - LH_k \leq 1 + \log_2 K.$$

for each k. Dividing the above by L we get the desired conclusion.

PROBLEM 3.

(a) The intermediate nodes of a tree have the property that if w is an intermediate node, then so are its ancestors. Conversely, as we remark on the notes on Tunstall coding, if a set of nodes has this property, it is the intermediate nodes of some tree. Thus, all we need to show is that $w \in S$ implies that its prefixes are also in S.

Suppose v is a prefix of w, and $v \neq w$. Then $p_j(v) > p_j(w)$. Thus, $\hat{p}(v) > \hat{p}(w)$. Since S is constructed by picking nodes with highest possible values of $\hat{p}$, we see that if $w \in S$, then $v \in S$.

From class, we know that if a K-ary tree has α intermediate nodes, the tree has $1 + (K-1)\alpha$ leaves.

(b) Since S contains the α nodes with the highest value of $\hat{p}$, no node outside of S can have strictly larger $\hat{p}$ than any node in S. Thus, $\hat{p}(w) \leq Q$.

(c) From part (b) $p_j(w) \leq \hat{p}(w) \leq Q$. Thus, $\log(1/p_j(w)) \geq \log(1/Q)$. Multiplying both sides by $p_j(w)$ and summing over all W we get

$$H_j(W) \geq \log(1/Q).$$

(d) For any leaf w in W we have

$$\begin{aligned} p_1(w) &= p_1(\text{parent of } w) p_1(\text{last letter of } w) \\ &\geq p_1(\text{parent of } w) p_{1,\text{min}} \end{aligned}$$

For a leaf w in W_1 , $p_1(\text{parent of } w) = \hat{p}(\text{parent of } w) \geq Q$. Thus, all leaves in W_1 have $p_1(w) \geq Q p_{1,\text{min}}$. Now

$$1 = \sum_{w \in W} p_1(w) \geq \sum_{w \in W_1} p_1(w) \geq |W_1| Q p_{1,\text{min}}.$$

(e) The same argument as in (d) establishes that $|W_2| Q p_{2,\text{min}} \leq 1$. Thus

$$|W| = |W_1 \cup W_2| \leq |W_1| + |W_2| \leq \frac{1}{Q}[1/p_{1,\text{min}} + 1/p_{2,\text{min}}].$$

(f) By part (e) $\log(W) \leq \log(\frac{1}{Q}) + \log(1/p_{1,\text{min}} + 1/p_{2,\text{min}})$. By part (c) $\log(1/Q) \leq H_j(W)$, we also know $H_j(W) = H_j(U) E_j[\text{length(W)}]$.
Thus using $\lceil x \rceil < x + 1$,

$$\begin{aligned} \rho_j &= \frac{\lceil \log(|W|) \rceil}{E_j[\text{length(W)}]} \\ &< \frac{1 + H_j(U) E_j[\text{length(W)}] + \log(1/p_{1,\text{min}} + 1/p_{2,\text{min}})}{E_j[\text{length(W)}]} \\ &= H_j(U) + \frac{1 + \log(1/p_{1,\text{min}} + 1/p_{2,\text{min}})}{E_j[\text{length(W)}]}. \end{aligned} \tag{1}$$

(g) A α gets larger, since $|W| = 1 + (K-1)\alpha, \log(|W|)$ gets larger. As we saw in part (f) $H_j(W)$ is lower bounded by $\log(|W|) - \log(1/p_{1,\min} + 1/p_{2,\min})$, so $H_j(W)$ gets larger too. Furthermore, $E_j[\text{length(W)}] = H_j(W)/H_j(U)$, and thus so as α gets large $E_j[\text{length(W)}]$ gets larger also. Thus, as α gets large, we see that the right hand side of (1) approaches $H_j(U)$.

PROBLEM 4. Let $s(m) = 0 + 1 + \cdots + (m-1) = m(m-1)/2$. Suppose we have a string of length $n = s(m)$. Then, we can certainly parse it into m words of lengths 0, 1, ..., $(m-1)$, and since these words have different lengths, we are guaranteed to have a distinct parsing. Since a parsing with the maximal number of distinct words will have at least as many words as this particular parsing, we conclude that whenever $n = m(m-1)/2$, $c \geq m$.

Now, given n, we can find m such that $s(m-1) \leq n < s(m)$. A string with n letters can be parsed into $m-1$ distinct words by parsing its initial segment of $s(m-1)$ letters with the above procedure, and concatenating the leftover letters to the last word. Thus, if a string can be parsed into $m-1$ distinct words, then $n < s(m)$, and in particular, $n < s(c+1) = c(c+1)/2$.

From above, it is clear that no sequence will meet the bound with equality. On the other hand, an all zero string of length $s(m)$ can be parsed into at most m words: in this case distinct words must have distinct lengths.