1 Codes without long runs

We recall the well known Run Length Limited (RLL) constraint in which the lengths of the zero runs are limited to a fixed range of values.

**Definition 1.** Let $k$ and $n$ be two integers. We say that a vector $a \in \Sigma_q^n$ satisfies the $k$-run length limited (RLL) constraint, and is called a $k$-RLL vector, if $n < k$ or $\forall i \in [n - k + 1]: w_H(a_{i+k-1}) \geq 1$.

In other words, a vector is called a $k$-RLL vector if it does not contain a zero run of length $k$.

We denote by $A_q(n, k)$ the set of all $k$-RLL length-$n$ vectors over the alphabet $\Sigma_q$, and $a_q(n, k) = |A_q(n, k)|$. The redundancy of the set $A_q(n, k)$ is denoted as $\text{red}(A_q(n, k)) = n - \log_q a_q(n, k)$.

The definition of the $k$-RLL constraint is similar to the $(d, k)$-Run Length Limited (RLL) constraint that states that any zero run is of length at least $d$ and at most $k$. In other words, the $k$-RLL constraint is equivalent to the well studied $(0, k - 1)$-RLL constraint.

**Theorem 2.** Let $n, k$ be positive integers such that $k \leq n$, then

$$a_q(n, k) \geq q^n - q^{n-k}(n - k + 1).$$

In particular, for $k \geq \log_q n + 1$, we get that $\text{red}(A_q(n, k)) \leq \log_q \left(\frac{q}{q-1}\right)$.

**Proof.** There are $n - k + 1$ positions in which a zero run can start and for each position we have at most $q^{n-k}$ different vectors. From the union bound we have that the number of length-$n$ vectors with a zero run of length at least $k$ is upper bounded by $q^{n-k}(n - k + 1)$, and by excluding those vectors from the set of all length-$n$ vectors we get

$$a_q(n, k) \geq q^n - q^{n-k}(n - k + 1).$$

This bound is irrelevant for values of $k$ smaller than $\log_q n$ as it is less than zero. However, if we
choose \( k \geq \log_q n + 1 \) it becomes useful and we get

\[
a_q(n, k) \geq q^n - q^{n-\log_q n -1}(n - \log_q n) \\
= q^n \left( 1 - \frac{n - \log_q n}{qn} \right) \\
\geq q^n \left( 1 - \frac{n}{qn} \right) \\
= q^{n-1}(q-1).
\]

Thus,

\[
\text{red}(A_q(n, k)) = n - \log_q a_q(n, k) \leq n - \log_q (q^{n-1}(q-1)) = \log_q \left( \frac{q}{q-1} \right).
\]

2 Balanced Codes

A vector \( v \in \Sigma_q^n \) will be called:

- **symbol-balanced** if the number of times each symbol appears is the same \( \left( \frac{n}{q} \right) \) and \( n \) is a multiple of \( q \),
- **weight-balanced** if \( \sum_{i=1}^n v_i = \frac{n(q-1)}{2} \) and \( n \) is even.

We let \( A_{n,q}, B_{n,q} \) be the set of all symbol-balanced and weight-balanced vectors over \( \Sigma_q^n \), respectively.

(a) Calculate the size of each set \( A_{n,q}, B_{n,q} \). Conclude on the minimum redundancy of any code for symbol-balanced vectors (you don’t need to calculate the minimum redundancy for the weight-balanced vectors).

(b) Use Knuth’s algorithm in order to design codes for symbol-balanced and weight-balanced vectors. Prove correctness, analyze the number of redundancy symbols, and, for symbol-balanced vectors, compare with the lower bound on the redundancy. You can assume that the redundancy symbols do not need to satisfy the constraint in each case.