The Minimum Distance of the [137, 69] Binary Quadratic Residue Code

Nigel Boston

Abstract—We find that the minimum distance of the binary [137, 69] quadratic residue code is 21.

Index Terms—Quadratic residue.

The quadratic residue code $C_p$ of length $p \equiv \pm 1 \pmod{8}$ over GF(2) is a cyclic $[p, (p + 1)/2]$ code with generator polynomial $\prod_{\alpha \in Q}(x - \alpha^r)$, where $\alpha$ is a primitive $p$th root of 1 in some extension of GF(2) and $Q$ is the set of nonzero quadratic residues modulo $p$. The behavior of the minimum distance $d_p$ of $C_p$ for large $p$ is poorly understood [3, Ch. 16]. This correspondence makes a small contribution to our understanding by finding $d_{137} = 21$, which was previously unknown ($d_{113}$ was computed in [2]). It is interesting to note that the answer, $d_{137} = 21$, lies at the upper end of the theoretical range given in [3, p. 483]. This suggests that binary quadratic residue codes might in general be good codes. The next unknown case ($d_{167}$) is, however, well beyond current computational capabilities.

The method was to have the software package MAGMA [1] run the following program:

```python
print(MinimumDistance(QRCode(FiniteField(2),137)));
```

in the background for about 10 days. Note that MAGMA actually gives the dual code of $Q_{137}$ and thus the answer 22, but $d_{137} = 21$ then follows.

ACKNOWLEDGMENT

The author wishes to thank Richard Blahut for introducing him to this problem.

REFERENCES


The Minimum Distance of the [83, 42] Ternary Quadratic Residue Code

Doug Kuhlman

Abstract—We find the minimum distance of the nonextended [83, 42] ternary quadratic residue code to be 20.

Index Terms—Ternary quadratic residue code.

A quadratic residue code $C_p$ of length $p \equiv \pm 1 \pmod{12}$ over GF(3) is a cyclic $[p, (p + 1)/2]$ code with generator polynomial $\prod_{\alpha \in Q}(x - \alpha^r)$, where $\alpha$ is a primitive $p$th root of 1 in some extension of GF(3) and $Q$ is the set of nonzero quadratic residues modulo $p$. Although these codes are often very good, their full nature is not well understood [2, Ch. 16]. Specifically, the minimum distance $d_p$ of $C_p$ for large $p$ is not known. While this correspondence does not make any theoretical gain in understanding the general problem, it does provide another data point by finding $d_{83}$, which was previously unknown. It is interesting to note that the answer, $d_{83} = 20$, is significantly larger than the lower bound $\lceil \sqrt{p} \rceil = 10$ given in [2, p. 483].

Finding the answer involved having the software package MAGMA [1] run the following program:

```python
print(MinimumDistance(QRCode(GF(3),83)));
```

in the background for about 12 days (about 100 000 s of CPU time). In general, MAGMA uses the dual code for $p \equiv +1 \pmod{12}$, and the code itself for $p \equiv -1 \pmod{12}$. Therefore, MAGMA’s given result 20 is the correct value for $d_{83}$.

This method does not lend itself to generalization, nor even to the next logical step, namely, finding $d_{107}$. The program appears to run in $O(3^n)$ seconds, so an attempt to find $d_{107}$ would take on the order of 150 thousand years with current technology.

ACKNOWLEDGMENT

The author wishes to thank R. Blahut for introducing him to this problem and N. Boston for suggesting the mechanism for solving the problem.

REFERENCES