Algorithms for the MDS Coding Conjecture

By: Safal Bora, University of Michigan. Dr. Steven B. Damelin, Mathematical Reviews, The American Mathematical Society.

Abstract

This research focuses on different conjectures for MDS (Maximum Distance Separable) codes, error correcting codes, and the algorithms associated with those conjectures. The objective of this research is to specifically focus on one of the conjectures for MDS codes and numerically verify this conjecture through algorithms using Extended Reed Solomon codes. This research is important because it allows for a different approach to decoding MDS codes as most algorithms created to verify different conjectures of MDS codes have only used Reed Solomon Codes. Another research paper "On a Condition Equivalent to the MDS Conjecture" mathematically proved a specific MDS conjecture and was first used to understand why the MDS conjecture was theoretically true. The next step was to use MATLAB, Finite Field packages, and books such as the Handbook of Finite Fields to create algorithms that numerically verify this MDS conjecture. This step is still ongoing and so is the next step of testing the algorithms to check whether they work for both small and large cases of numbers. The current results show that the algorithms hold for small cases of this MDS conjecture, yet for the large cases, it is still unclear whether the algorithms hold. Extensive research will be conducted to generalize this MDS conjecture for more numerical cases and eventually expand this MDS conjecture to Extended Reed Solomon codes with prime powers. Although the algorithms are currently being revised, there has not yet been a counterexample found from a compiler that disproves this MDS conjecture.

Methods

In order to check the theorem numerically to verify its correctness, we used the gf(Galois Field) class in MATLAB. By using this class we were able to set up random polynomials within each finite field for B and check for linear independence using the in-built function in the class that gave a matrix's rank. If the matrix rank equaled the smaller dimension of the matrix, then the matrix was linearly independent. Below is an example of a Reed-Solomon Code in the finite field of 7 where linear independence was checked:

```
A = [
      1 1 1 1 1 1 1
      0 1 2 3 4 5 6
      0 1 4 2 2 4 1
      0 1 1 6 1 6 6
      ]
B = rank(A)
```

Possible q	Can be checked?
2	Not in bounds
4	Yes
7	No
8	Yes
16	Yes
32	Yes
42	No
64	Yes
100	No
128	Possibly in Future
256	Possibly in Future

Background

The research mainly deals with finite fields and Extended Reed-Solomon Codes, which are both used in image and signal processing. Although the research is mainly focused on the mathematical and algorithmic side of these codes, our results can lead to a more comprehensive understanding of error correction in both signal and image processing. It is also important to note that MATLAB was used as the programming language because it is compatible with matrices and linear algebra, both of which are used quite frequently in the research.

Hypothesis

The following theorem(Sun, Damelin, Bora, Kaiser, Yu) below was verified by our research:

- Suppose q is odd or, if q is even, k≠ 3 or q-1. There is no integer s with $q \geq s > k$ such that the Reed-Solomon code R over F_q of dimension s can have s-k+2 columns B = {b1,.....,b_{s-k+2}} added to it, such that:
  - Any s x s submatrix of R ∪ B containing the first s-k columns of B is independent.
  - B ∪ {[0,0,….,0,1]} is independent. The MDS conjecture is true.

Results

As MATLAB only supports finite fields where $q = 2^p$ where p is a prime, we currently have only been able to verify q = 4,8,16,32, and 64 as shown in the figure on the right. Within these values of q, we have checked for every value of s and k to verify that there is no such B for which the two statements in the theorem hold.

Next Steps

As part of future work, we plan to try to test for higher values of q such as 128,256, and beyond. As the theorem above needs to be mathematically verified for all q, we will create our own finite field so that there will be no limits or specifications on q.