

# Research Progress Report

March 27, 2025

## 1. Simulating Data to Enhance VSD with AI

Data has been simulated where decoding by the traditional Vector Symbol Decoding (VSD) algorithm fails algebraically, with the objective of training an AI model to improve success rates in these non-decodable cases. An error-locating vector  $\sigma \in GF(2^n)$  is considered where  $\sigma[i] = 0$  indicates an unverified (erroneous) row position  $i \in \{1, \dots, n\}$ :

$$\sigma = [1 \quad 1 \quad \dots \quad \sigma_i = 0 \quad \dots \quad 1], \quad \exists i \in \{1, \dots, n\}.$$

A random message matrix  $M \in \mathbb{M}_{k \times r}(GF(2))$  of an  $(n, k)$  code, where  $k$  represents the number of information symbols,  $n$  the codeword length in symbols, and  $r$  the symbol length in bits, is used. The codeword matrix is generated via a generator matrix  $G$ :

$$G : M \mapsto Y$$

Errors are introduced in the received matrix  $\tilde{Y}$ : for  $\sigma_i = 0$ , the  $i$ -th row is given by  $\tilde{Y}_i = Y_i + e_i$ , where  $e_i \in GF(2^r)$ . The syndrome matrix is computed as:

$$S = H \cdot \tilde{Y},$$

where  $H$  denotes the  $(n - k) \times n$  parity-check matrix over  $GF(2)$ . If  $S_{ij} \neq 0$ , column operations are applied to  $S$ :

$$S = \begin{bmatrix} 0 & S_{ij} \neq 0 & \dots & 0 \\ & \ddots & & \\ 0 & & \ddots & \\ 0 & & & 0 \end{bmatrix} \xrightarrow{\text{column operation}} \begin{bmatrix} 0 \\ S_{i:} \neq \vec{0} \\ 0 \\ 0 \end{bmatrix}.$$

To be complete..

## 2. Literature Review on Kolmogorov-Arnold Networks (KANs)

A literature review has been conducted on Kolmogorov-Arnold Networks (KANs) as an alternative to Multi-Layer Perceptrons (MLPs), which are underpinned by the Universal Approximation Theorem. Functions in KANs are decomposed as:

$$f(x_1, \dots, x_n) = \sum_{q=1}^{2n+1} \Phi_q \left( \sum_{p=1}^n \phi_{q,p}(x_p) \right),$$

where  $\phi_{q,p}$  and  $\Phi_q$  are learned univariate functions. This structure is compared to MLPs' dense layers, potentially offering efficiency for decoding tasks like VSD, where  $S \in \mathbb{M}_{(n-k) \times r}(GF(2))$  maps to  $\sigma \in GF(2^n)$ . Insights into channel coding's frontier models are sought, with KANs' ability to capture non-linear error patterns being explored. Implementation and testing of a KAN-based decoder are planned.

### 3. Code Optimization and Simulation Speed-Up

The VSD algorithm and simulation framework have been optimized to accelerate experimentation. For  $Y \in \mathbb{M}_{n \times r}(GF(2))$ , the computation of  $S = H \cdot \tilde{Y} \bmod 2$  and Gaussian elimination have been improved. The random number generator has been replaced with a pseudorandom number generator (e.g., Mersenne Twister), and options like cryptographically secure PRNGs and quantum RNGs have been explored for generating  $e_i \in GF(2^r)$ . Matrix operations have been vectorized, and redundant steps minimized, enabling rapid simulation of large datasets.

### 4. Ongoing Literature Review on Cryptography and Key Distribution

A literature review on cryptography, emphasizing key distribution, is being conducted. For a codeword matrix  $Y$ , errors are modeled as:

$$\tilde{Y} = Y + E, \quad E \in \mathbb{M}_{n \times r}(GF(2)),$$

and their correction is considered alongside secure key distribution methods. Integration of AI-enhanced VSD with these systems is being investigated.

### 5. Challenges with the Error-Locating Vector

The explicit reason for the algebraic VSD's failure to compute the correct error-locating vector remains unresolved. The true vector is observed as:

$$\sigma_{\text{real}} = \sigma_{\text{AI}} = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, \dots],$$

indicating an error at position 14, while traditional error locating used in VSD yields:

$$\sigma_{\text{VSD}} = [1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, \dots],$$

flagging positions 7 and 14. The computation  $\sigma_{\text{VSD}} =_{i \in Id} h_i$  suggests a modulo-2 sum of rows  $h_{i:}$  of  $H$ . For  $\tilde{Y}_{14:} = Y_{14:} + e_{14}$ ,  $S = H_{14:}$ , but  $\sigma_{\text{VSD}}$  implies  $\sigma_{\text{VSD}} = H_{7:} + e \oplus H_{14:}$ . Possible errors in Gaussian elimination, dependencies in  $H$ , or symbol-bit mismatches are being analyzed. Simulation output, such as:

$$\text{tensor}([[\text{True}, \text{True}, \text{True}, \text{True}, \text{True}, \text{True}, \text{False}, \text{True}, \dots]]),$$

suggests VSD overestimates errors.

### Next Steps

Improvement of AI over VSD will be quantified, a KAN-based decoder will be implemented, simulation optimizations will be finalized, and  $\sigma_{\text{VSD}}$  failures will be resolved.