A Simple Deterministic Reduction for the Gap Minimum Distance of Code Problem

Per Austrin
University of Toronto

Subhash Khot
New York University

May 16, 2012

Presented by: Pavel Rytíř

1 Handout

Let q equals 2.

Definition 1.1. A linear code C over a field $\mathbb{F}_q$ is a linear subspace of $\mathbb{F}_q^n$, where n is the block-length of the code and dimension of the subspace C is the dimension of the code. The distance of the code $d(C)$ is the minimum Hamming weight of any non-zero vector in C.

Definition 1.2. MIN DIST(q) is the problem of determining the distance $d(C)$ of a linear code $C \subseteq \mathbb{F}_q^n$. The code may be given by the basis vectors for the subspace C or by the linear forms defining the subspace.

Definition 1.3. NCP(q) is the problem of determining the minimum distance from a given point $p \in \mathbb{F}_q^n$ to any codeword in a given code $C \subseteq \mathbb{F}_q^n$. Equivalently, it is the problem of determining the minimum Hamming weight of any point z in a given affine subspace of $\mathbb{F}_q^n$ (which would be $C - p$).

Definition 1.4. Let $C_1, C_2 \subseteq \mathbb{F}_q^n$ be linear codes. Then the linear code $C_1 \otimes C_2 \subseteq \mathbb{F}_q^{n^2}$ is defined as the set of all $n \times n$ matrices over $\mathbb{F}_q$ such that each of its columns is a codeword in C_1 and each of its rows is a codeword in C_2.

Fact 1.5. Let $C_1, C_2 \subseteq \mathbb{F}_q^n$ be linear codes. Then the linear code $C_1 \otimes C_2 \subseteq \mathbb{F}_q^{n^2}$ has distance $d(C_1 \otimes C_2) = d(C_1)d(C_2)$.

Lemma 1.6. *Let $C \subseteq \mathbb{F}_q^n$ be a linear code of distance $d = d(C)$, and let $Y \in C \otimes C$ be a non-zero codeword with the additional properties that*

1. *The diagonal of Y is zero.*
2. *Y is symmetric.*

Then Y has at least $d^2(1 + 1/q)$ non-zero entries.

Fact 1.7. Let $C \subseteq \mathbb{F}_q^n$ be a linear code of distance $d = d(C)$. Then for any two linearly independent codewords $x, y \in \mathbb{F}_q^n$, the number of coordinates $i \in [n]$ for which either $x_i \neq 0$ or $y_i \neq 0$ is at least $d(1 + 1/q)$.

1.1 Hardness of Constraint Satisfaction

Definition 1.8. An instance Ψ of the MAX NAND problem consists of a set of quadratic equations over $\mathbb{F}_2$, each of the form $x_k = \text{NAND}(x_i, x_j) = 1 + x_i \cdot x_j$ for some variables x_i, x_j, x_k. The objective is to find an assignment to the variables such that as many equations as possible are satisfied. We denote by $\mathsf{Opt}(\Psi) \in [0, 1]$ the maximum fraction of satisfied equations over all possible assignments to the variables.

Theorem 1.9. *There is a universal constant $\delta > 0$ such that given a* MAX NAND *instance Ψ it is NP-hard to determine whether $\mathsf{Opt}(\Psi) = 1$ or $\mathsf{Opt}(\Psi) \leq 1 - \delta$.*

1.2 Reduction to Nearest Codeword

Given a MAX NAND instance Ψ with n variables and m constraints, we shall construct an affine subspace $\mathcal{S}$ of $\mathbb{F}_2^{4m}$ such that:

(i) If Ψ is satisfiable then $\mathcal{S}$ has a vector of Hamming weight at most m.

(ii) If $\mathsf{Opt}(\Psi) \leq 1 - 2\delta$ then $\mathcal{S}$ has no vector of Hamming weight less than $(1 + 2\delta)m$.

This proves, according to Definition 1.3, that NCP(2) is NP-hard to approximate within a factor $1 + 2\delta$.

Every constraint $x_k = 1 + x_i x_j$ in Ψ gives rise to four new variables, as follows. We think of the four variables as a function $S_{ijk} : \mathbb{F}_2^2 \to \mathbb{F}_2$. The intent is that this function should be the indicator function of the values of x_i and x_j, in other words, that

$$S_{ijk}(a, b) = \begin{cases} 1 & \text{if } x_i = a \text{ and } x_j = b \\ 0 & \text{otherwise} \end{cases} .$$

With this interpretation in mind, each function S_{ijk} has to satisfy the following linear constraints over $\mathbb{F}_2$:

$$\begin{aligned} S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) + S_{ijk}(1,1) &= 1 & (1) \\ S_{ijk}(1,0) + S_{ijk}(1,1) &= x_i & (2) \\ S_{ijk}(0,1) + S_{ijk}(1,1) &= x_j & (3) \\ S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) &= x_k. & (4) \end{aligned}$$

1.3 Reduction to Minimum Distance

$$S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) + S_{ijk}(1,1) = x_0 \qquad (1')$$

A first observation is that the system of constraints relating S_{ijk} to (x_0, x_i, x_j, x_k) is invertible. Namely, we have Equations (1')-(4), and inversely, that

$$\begin{aligned} S_{ijk}(0,0) &= x_i + x_j + x_k & \qquad S_{ijk}(0,1) &= x_0 + x_j + x_k \\ S_{ijk}(1,0) &= x_0 + x_i + x_k & \qquad S_{ijk}(1,1) &= x_0 + x_k. \end{aligned}$$

Analogously to the S_{ijk} functions intended to check the NAND constraints of Ψ, we now introduce for every $i, j \in [N]$ a function $Z_{ij} : \mathbb{F}_2^2 \to \mathbb{F}_2$ that is intended to check the constraint $Y_{ij} = y_i \cdot y_j$, and that is supposed to be the indicator of the assignment to the variables (y_i, y_j). We then impose the analogues of the constraints (1')-(4), viz.

$$\begin{aligned} Z_{ij}(0,0) + Z_{ij}(0,1) + Z_{ij}(1,0) + Z_{ij}(1,1) &= x_0 && (5) \\ Z_{ij}(1,0) + Z_{ij}(1,1) &= y_i && (6) \\ Z_{ij}(0,1) + Z_{ij}(1,1) &= y_j && (7) \\ Z_{ij}(1,1) &= Y_{ij}. && (8) \end{aligned}$$

Theorem 1.10. *For any finite field $\mathbb{F}_q$, there exists a constant $\gamma > 0$ such that it is NP-hard (via a deterministic reduction) to approximate the* Min Dist(q) *problem to within a factor $1 + \gamma$.*