Recognition of a code in a noisy environment

Christophe Chabot

Université de Limoges - XLIM - DMI
INRIA Rocquencourt - Projet CODES

ISIT Nice - June 29, 2007
Outline

1. Introduction

2. Recognition of a code
   - Problem
   - Theoretical results
   - Experimental results

3. Conclusion
Introduction

Context:
- Data transmission.
- Several transformations applied to the data.
- Transmitted data is corrupted by noise.

General Problem: Finding some information on the transformations used only from an intercepted noisy sequence.
Reconstruction of code

Recovering the first layer i.e. the error correcting code used.
→ Problem of reconstruction of code.

State of art:
- convolutional codes, turbo codes: [Fil],[Bar],[Din]
- linear block codes: [Val],[Clu],[Bar]
  This problem is NP-Hard [Val].
  Complexity exponential in the error rate.

→ Easier problem : recognition of a code.
Problem

Let be given a binary sequence $S$ and a binary linear code $C$.

Is this sequence $S$ composed of noisy codewords of $C$?
**Binary linear code**

Let $C$ be a binary linear code of length $n$ and dimension $k$. i.e. $C$ is a vector subspace of $\mathbb{F}_2^n$ of dimension $k$.

It can be represented by:

- its generator matrix (whose lines span the vector subspace $C$).
- its parity check matrix (whose lines span $C^\perp$).

We will use a basis of the dual code of $C$, $H = (h_1, h_2, \ldots, h_{n-k})$, then, for all $j \in \{1, \ldots, n-k\}$, for all $c \in C$,

$$< h_j, c >= \sum_{i=1}^{n} h_{j,i} \ast c_i = 0.$$  

We will note, for $x \in \mathbb{F}_2^n$, $\text{supp}(x) = \{ i \in \{1, \ldots, n\} | x_i \neq 0 \}$, and $w(x) = \#(\text{supp}(x))$ its Hamming weight.
We will consider a binary symmetric channel with error rate $p$. 

$$S = (s_j)_{1 \leq j \leq M}$$ is a sequence taken at the output of the channel, $s_j \in \mathbb{F}_2^n$.

Note: if $p = 1/2$, $S$ is random (one-time pad).
Main idea

- If $S$ comes from $C$:
  - if $p = 0$, $<h_i, s_j> = 0$ for all $i,j$.
  - if $0 < p << 1/2$, $<h_i, s_j> = 1$ for a few number of $i,j$.
- Else: $<h_i, s_j> = 1$ half of the time.
Different cases

Let $h$ be a non-zero word of $C^\perp$.

- If $S$ comes from a random sequence:
  \[
  \dim(h^\perp) = n - 1, \quad \frac{\#(h^\perp)}{\#(\mathbb{F}_2^n)} = \frac{1}{2},
  \]
  \[
  \rightarrow P[< h, s_j > = 1] = 1/2.
  \]

- If $S$ comes from words of $C$:
  For all $j$, $s_j = c_j + e_j$, with $c_j \in C$ and $e_j$ the error vector.
  \[
  < h, s_j > = 1 \iff \#(\text{supp}(h) \cap \text{supp}(e_j)) \text{ is odd}.
  \]
  \[
  \rightarrow P[< h, s_j > = 1] = \frac{1 - (1 - 2p)^{w(h)}}{2}.
  \]
Different cases

- If $S$ comes from a sequence of words of $C' \not\in C$: There exists at least one $h_0$ in a basis of $C^\perp$ such that $h_0 \not\in C'^\perp$.
  \[ P[< h_0, s_j > = 1] = 1/2. \]

- Other cases: heuristic result but confirmed by experiments.
  \[ P[< h, s_j > = 1] = 1/2. \]
Statistical test

Summary:

- If $S$ does not come from $C$:
  There exists at least one $h \in \{h_1, \ldots, h_{n-k}\}$, s.t.
  $$P[< h, s_j > = 1] = 1/2.$$

- If $S$ comes from $C$:
  For all $h \in \{h_1, \ldots, h_{n-k}\}$,
  $$P[< h, s_j > = 1] = \frac{1}{2} - \frac{1}{2} (1 - 2p)^{w(h)}.$$

Idea:

$$\sum_{j=1}^{M} < h, s_j > \text{ (sum in } \mathbb{Z}) \text{ “close” to } \frac{M}{2} - \frac{M}{2} (1 - 2p)^{w(h)}?$$
Theorem

Theorem: The statistical test consisting in deciding that $S = (s_j)_{1 \leq j \leq M}$ comes from a sequence of $M$ words of $h^\perp$ if and only if

$$\sum_{j=1}^{M} < h, s_j > \leq T$$

with

$$M = \left( \frac{b\sqrt{1-(1-2p)^2w(h)}-a}{(1-2p)^w(h)} \right)^2, \quad T = \frac{1}{2}(M + a\sqrt{M})$$

and $a = \phi^{-1}(\alpha)$, $b = \phi^{-1}(1 - \beta)$ verifies:

$$P[(\sum_{j=1}^{M} < h, s_j >) \leq T | S \text{ is random}] = \alpha \text{ (false alarm)},$$

$$P[(\sum_{j=1}^{M} < h, s_j >) \geq T | S \text{ comes from } h^\perp] = \beta \text{ (non detection)}.$$
Input:
- $C$, a $[n, k]$-linear binary code,
- a binary symmetric channel with error rate $p$,
- $S = (s_j)_{1 \leq j \leq M}$, a sequence taken at the output of the channel,
- $\alpha, \beta$, false alarm and non detection probabilities.

Initialization:
- Compute $(h_1, \ldots, h_{n-k})$ a basis of $C^\perp$ with low weight words.
- Compute $M_i$ and $T_i$ for each $i \in \{1, \ldots, n-k\}$.

Algorithm:
- $N_i = \sum_{j=1}^{M_i} < h_i, s_j >$ (sum in $\mathbb{Z}$) for $i \in \{1, \ldots, n-k\}$.

Output:
- If $N_i \leq T_i$ for all $i \in \{1, \ldots, n-k\}$, say that $S$ comes from a sequence of words of $C$.
- If $N_i \geq T_i$ for at least one $i \in \{1, \ldots, n-k\}$, say that $S$ does not come from a sequence of words of $C$. 
### Computing results

<table>
<thead>
<tr>
<th>Code used</th>
<th>( n )</th>
<th>( w(h) ) max</th>
<th>Error rate ( p )</th>
<th>( M )</th>
<th>( T )</th>
<th>Time is s</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCH 511</td>
<td>511</td>
<td>140</td>
<td>0.005</td>
<td>1462</td>
<td>640</td>
<td>( \leq 1 )</td>
</tr>
<tr>
<td>BCH 511</td>
<td>511</td>
<td>140</td>
<td>0.01</td>
<td>25823</td>
<td>12529</td>
<td>17</td>
</tr>
<tr>
<td>RM 1024</td>
<td>1024</td>
<td>136</td>
<td>0.005</td>
<td>1345</td>
<td>585</td>
<td>4</td>
</tr>
<tr>
<td>RM 1024</td>
<td>1024</td>
<td>136</td>
<td>0.01</td>
<td>21963</td>
<td>10629</td>
<td>28</td>
</tr>
<tr>
<td>Random 2000</td>
<td>2000</td>
<td>535</td>
<td>0.001</td>
<td>724</td>
<td>298</td>
<td>9</td>
</tr>
<tr>
<td>Random 2000</td>
<td>2000</td>
<td>535</td>
<td>0.002</td>
<td>6540</td>
<td>3078</td>
<td>62</td>
</tr>
</tbody>
</table>

Here, \( \alpha = \beta = 10^{-6} \).
Synchronization

**Input:**

\[ C, p, S, \alpha \text{ and } \beta. \]

**Initialization:**

- Compute \[ S^{(l)} = (s_{j}^{(l)})_{1 \leq j \leq M}, \] \[ 0 \leq l \leq n - 1. \]
- Compute \( (h_1, \ldots, h_r) \) some words of a basis of \( C^\perp \).
- Compute \( M_i \) and \( T_i \) for each \( i \in \{1, \ldots, r\} \).

**Algorithm:**

For \( l \) from 0 to \( n - 1 \) do

\[ N_i^{(l)} = \sum_{j=1}^{M_i} < h_i, s_{j}^{(l)} > \text{ (sum in } \mathbb{Z} \text{) for } i \in \{1, \ldots, r\}. \]

**Output:**

If \( N_i^{(l)} \leq T_i \) for all \( i \in \{1, \ldots, r\} \),

say that \( S \) seems to come from a sequence of words of \( C \) with synchronization \( l \rightarrow \text{check it!} \).
Conclusion

- Easy and fast algorithm to recognize a code.
- Can reach high error rate (compared to reconstruction).
- Application to synchronization.
Thanks for your attention.