Sum Covers in Steganography

Petr Lisoněk
Simon Fraser University
Vancouver, BC, Canada

Algebraic and Combinatorial Coding Theory • ACCT2008
Pamporovo
20 June 2008

Steganography = Information Hiding Theory

- The *sender* embeds a hidden message into a *cover object* (eg. a digital multimedia file) by slightly distorting it.
- The *recipient* retrieves the hidden message from the distorted cover object.
- The existence of the message is *impossible to detect* by any third party.

Detecting the hidden message by a third party

- by naked eye/ear/...
- by powerful statistical methods

The amount of noise naturally present in the cover object determines the amount of distortion that can be introduced.

Examples: lossy compression (image/audio/...)

Is the third party an **adversary** (enemy)?

information hiding/embedding

Example: cover object & stego file

JPEG cover object size: 271,560 bits (JPEG compression 1:23)

payload: 10,000 random bits embedded

Representation of cover objects

- The cover object is a sequence of integers from $D = \{0, \ldots, 2^e - 1\}$. Typically $e \in \{8, 12, 16\}$.
- Example:
  $D =$ set of color intensities (grayscale or RGB)
- For simplicity we'll call the elements of D **pixel values** (could be also "audio pixels" etc.) and **assume** $D = \mathbb{Z}$.

Pixels $\longrightarrow$ Message Symbols

S ... set of *message symbols*

Retrieving information from pixel values:

$$s : \mathbb{Z} \to S$$

To embed a given symbol $z \in S$ into a given pixel value $x \in \mathbb{Z}$, the sender **modifies** $x \rightsquigarrow x'$ so that:

- $s(x') = z$, and
- $|x' - x|$ is minimized.

Example of Symbol-assignment function

$$s : \mathbb{Z} \to \mathbb{Z}_3$$

$$s(x) := x \bmod 3$$

This requires only $\pm$**1 changes,** whose **number** will be the **measure of distortion**.

Covering Codes

We need to manage the trade-off between
the amount of communicated **information** and
the amount of introduced **distortion.**

Galand & Kabatiansky, Steganography via covering codes. (ISIT 2003)

Syndrome coding: The hidden message is the syndrome of the vector of message symbols w.r.t. a fixed $r \times n$ parity check matrix.

The number of changes performed by the sender

upper bounded by

$$R(C) := \max_{x \in \mathbb{F}_q^n} d(x, C)$$

measured by

$$R_a(C) := q^{-n} \sum_{x \in \mathbb{F}_q^n} d(x, C)$$

... new invariant: **"average distance to code"**

Distortion rate & information rate

distortion rate:

$$\rho := \frac{R(C)}{n} \quad \text{or} \quad \rho := \frac{R_a(C)}{n}$$

is (an upper bound on) the probability that a given pixel will be subjected to a change

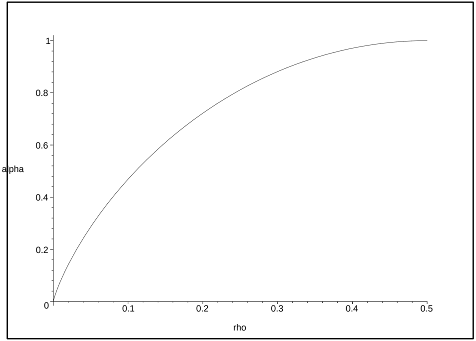
information rate:

$$\alpha := \frac{r}{n} \log_2 q$$

is the number of message bits per pixel

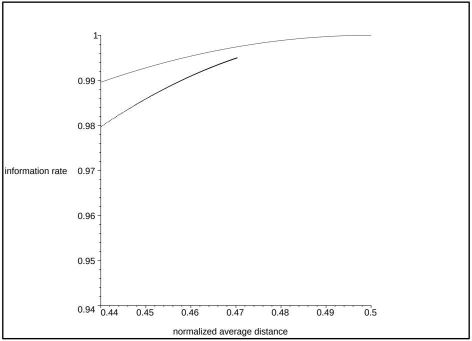
$q = 2$: α versus ρ trade-off

$$\alpha(\rho) \leq H_2(\rho) = -\rho\log(\rho) - (1-\rho)\log(1-\rho)$$



Binary case: optimal codes with $r = n - 2$

... completely classified under the $R_a(C)$ measure:
Khatirinejad & PL *(Discrete Appl. Math., in press)*



Restricting the embedding positions

During the JPEG compression of the raw image, DCT coefficients have to be rounded to integers.

The sender may employ *"dishonest rounding"* to embed information.

The sender would like to utilize **only** those values where the dishonest rounding is hard to detect. ($17.502 \to 17$ hurts less than $17.813 \to 17$.)

The receiver **(and the attacker)** do **not** have access to this **side information.**

Restricting the embedding positions: "Wet Paper Codes"

Fridrich, Goljan, PL & Soukal, "Writing on wet paper"
(IEEE Trans. Signal Process. 2005)

Theorem. Suppose that we use random binary linear codes of length n, and suppose that the sender can change k positions prescribed to him (and not known to the receiver), where $n >> k$. The expected number of bits that the sender can communicate is $k + \epsilon(k)$, where $|\epsilon(k)| < k^2 2^{8-k/4}$.

Wet Paper Codes - proof of the theorem

We use *variable rate codes:* The sender will keep adding rows to H (pseudo-randomly generated) as long as the system $\bar{H}c^T = m^T$ is solvable, where $c \in \mathbb{F}_2^k$ is the vector corresponding to the k changeable positions, $\bar{H}$ are the columns of H corresponding to c, and m is a part of the message to be communicated.

The probability that the $\mathbb{F}_2$-rank of a random $r \times k$ binary matrix is equal to s is

$$P_{r,k}(s) = 2^{s(r+k-s)-rk} \prod_{i=0}^{s-1} \frac{\left(1-2^{i-r}\right)\left(1-2^{i-k}\right)}{1-2^{i-s}}.$$

Using this we compute for each $b \geq 0$ the probability that the sender can communicate exactly b bits.

Cells - definition

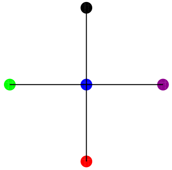
We partition the cover object into disjoint segments, each of which consists of d pixels.

cell ... an element of $\mathbb{Z}^d$

An example: Pooling pixels into pairs

Colours denote message symbols.

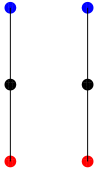
pixel pooling



embedding efficiency

$$= \frac{\log_2 5}{1} \approx 2.3$$

no pooling



embedding efficiency

$$= \frac{\log_2 3^2}{2} \approx 1.6$$

One change per cell: Symbol-assignment function

$$s : \mathbb{Z}^d \to \mathbb{Z}_{2d+1}$$

$$s(x_1, \ldots, x_d) := \left(\sum_{i=1}^{d} i x_i \right) \bmod (2d+1). \qquad (1)$$

In order to embed any symbol in $\mathbb{Z}_{2d+1}$ into any cell in $\mathbb{Z}^d$ using (1), at most one ± 1-change is required.

One change per cell: Theorem

Fridrich & PL *(IEEE Trans. Inf. Th. 2007)*

Theorem. The scheme that uses the symbol-assignment function (1) and then applies some $(2d+1)$-ary Hamming code **is never worse than** the scheme that changes individual pixels independently (without pooling) at the very same distortion rate, applying ternary Hamming codes.

Strict Sum Sets - definitions

Let $C \subseteq \mathbb{Z}_n$.

$$C + C := \{x + y : x, y \in C,\ x \neq y\}$$

$$-C := \{-x : x \in C\}$$

Symmetric strict sum cover of $\mathbb{Z}_n$

A subset $S \subseteq \mathbb{Z}_n$ is an **SSSC**(n) if

- $S + S = \mathbb{Z}_n$
- $0 \in S$
- $-S = S$.

Lemma. If $A = \{0, \pm a_1, \ldots, \pm a_d\}$ is an SSSC(n), then

$$s(x_1, \ldots, x_d) = \left(\sum_{i=1}^{d} a_i x_i \right) \bmod n$$

is a symbol-assignment function that allows the sender to embed any symbol in $\mathbb{Z}_n$ into any cell in $\mathbb{Z}^d$ by at most two ± 1-changes.

Maximizing the number of message symbols

$n_\gamma(k) :=$ the largest n s.t. $\exists$ SSC(n) of size k.
(Graham & Sloane 1980, Haanpää 2004)

$\hat{n}_\gamma(k) :=$ the largest n s.t. $\exists$ SSSC(n) of size k.

Proposition. For $3 \leq k \leq 13$, k odd, we have $\hat{n}_\gamma(k) = n_\gamma(k)$.

Proposition. Let $k = 2d + 1$. Then $\hat{n}_\gamma(k) \geq d^2 + 3d - 1$.
(This beats the one-change-per-cell scheme slightly.)

Please see the paper for proofs.

Open Problems

- The equality $\hat{n}_\gamma(k) = n_\gamma(k)$
  may hold for a larger set of values k.
- The bound $n^2 + 3d - 1$ is not tight, improve it.
- It appears that the optimal covers often possess a lot of symmetry. (Similarity with multiplier theorems for difference sets?)