Volume 24 Number 4

ACTA CYBERNETICA

Szeged, 2020

ACTA CYBERNETICA

Information for authors. Acta Cybernetica publishes only original papers in the field of Computer Science. Manuscripts must be written in good English. Contributions are accepted for review with the understanding that the same work has not been published elsewhere. Papers previously published in conference proceedings, digests, preprints are eligible for consideration provided that the author informs the Editor at the time of submission and that the papers have undergone substantial revision. If authors have used their own previously published material as a basis for a new submission, they are required to cite the previous work(s) and very clearly indicate how the new submission offers substantively novel or different contributions beyond those of the previously published work(s). There are no page charges. An electronic version of the published paper is provided for the authors in PDF format.

Manuscript Formatting Requirements. All submissions must include a title page with the following elements: title of the paper; author name(s) and affiliation; name, address and email of the corresponding author; an abstract clearly stating the nature and significance of the paper. Abstracts must not include mathematical expressions or bibliographic references.

References should appear in a separate bibliography at the end of the paper, with items in alphabetical order referred to by numerals in square brackets. Please prepare your submission as one single PostScript or PDF file including all elements of the manuscript (title page, main text, illustrations, bibliography, etc.).

When your paper is accepted for publication, you will be asked to upload the complete electronic version of your manuscript. For technical reasons we can only accept files in LaTeX format. It is advisable to prepare the manuscript following the guidelines described in the author kit available at `http://www.inf.u-szeged.hu/kutatas/acta-cybernetica/information-for-authors#AuthorKit` even at an early stage.

Submission and Review. Manuscripts must be submitted online using the editorial management system at `http://cyber.bibl.u-szeged.hu/index.php/actcybern/submission/wizard`. Each submission is peer-reviewed by at least two referees. The length of the review process depends on many factors such as the availability of an Editor and the time it takes to locate qualified reviewers. Usually, a review process takes 6 months to be completed.

Subscription Information. Acta Cybernetica is published by the Institute of Informatics, University of Szeged, Hungary. Each volume consists of four issues, two issues are published in a calendar year. Subscription rates for one issue are as follows: 5000 Ft within Hungary, €40 outside Hungary. Special rates for distributors and bulk orders are available upon request from the publisher. Printed issues are delivered by surface mail in Europe, and by air mail to overseas countries. Claims for missing issues are accepted within six months from the publication date. Please address all requests to:

Acta Cybernetica, Institute of Informatics, University of Szeged
P.O. Box 652, H-6701 Szeged, Hungary
Tel: +36 62 546 396, Fax: +36 62 546 397, Email: `acta@inf.u-szeged.hu`

Web access. The above information along with the contents of past and current issues are available at the Acta Cybernetica homepage `https://www.inf.u-szeged.hu/en/kutatas/acta-cybernetica` .

Zoltan Kato
Department of Image Processing
and Computer Graphics
University of Szeged
Szeged, Hungary
kato@inf.u-szeged.hu

Dragan Kukolj
RT-RK Institute of Computer Based
Systems
Novi Sad, Serbia
dragan.kukolj@rt-rk.com

László Lovász
Department of Computer Science
Eötvös Loránd University
Budapest, Hungary
lovasz@cs.elte.hu

Dana Petcu
Department of Computer Science
West University of Timisoara, Romania
petcu@info.uvt.ro

Heiko Vogler
Department of Computer Science
Dresden University of Technology
Dresden, Germany
Heiko.Vogler@tu-dresden.de

Gerhard J. Woeginger
Department of Mathematics and
Computer Science
Eindhoven University of Technology
Eindhoven, The Netherlands
gwoegi@win.tue.nl

Acta Cybernetica **24** (2020) 593–612.

An Information Theoretic Image Steganalysis for LSB Steganography

Sonam Chhikara[a] and Rajeev Kumar[b]

Abstract

Steganography hides the data within a media file in a subtle way while steganalysis exposes steganography by using detection measures. Traditionally, steganalysis reveals steganography by targeting perceptible and statistical properties of the image for improving the security of steganography methods. In this work, we target LSB image steganography methods by using information theoretic metrics for steganalysis. Our technique works in two phases. First, the features of embedded image are extracted, and then this image is analyzed on the basis of entropy and joint entropy features corresponding to the original image. Second, from these extracted features of the image, we train SVM and ensemble classifiers. The classifiers discriminate cover image from the stego image. For evaluating the robustness of our method, several attacks over the stego images are applied. These attacked stego images are then analyzed for the detection reliability of our method. Original images and LSB embedded images of the dataset are analyzed by comparing information gain from entropy and joint entropy metrics. Results suggest that entropy of the stego images are more preserving than that of joint entropy. Experimental results show that before histogram attack, detection rate with entropy and joint entropy are 70% and 98%, respectively while after the attack entropy metric gives 30% detection rate and joint entropy gives 93% detection rate. Therefore, joint entropy proves to be a good steganalysis measure having fewer false alarms for across a range of hiding ratios.

Keywords: data embedding, steganography, information theory, steganalysis, classifier

1 Introduction

Steganography aims to support secret communication in an innocuous looking media file [1]. The core objective of steganography is to maximize secret information (embedding capacity) with least distortion in original cover media (imperceptibility). Cover media and secret data both can be audio, video, image, and a text

[a] Deceased on Dec. 16, 2019. This paper is dedicated to her memory.

[b] School of Computer and Systems Sciences, Jawaharlal Nehru University, India. E-mail: `RajeevKumar.cse@gmail.com`, ORCID: `https://orcid.org/0000-0003-0233-6563`.

DOI: 10.14232/actacyb.279174

file. Imperceptibility indicates that embedding should not be visually recognizable. However, due to the addition of secret information, steganography subtly degrades the embedding media quality. Higher embedding capacity results in more modifications in original media that affects imperceptibility. Therefore, to overcome this trade-off a good steganography technique focuses on both the requirements in a balanced way. Image steganography methods are generally classified into spatial and transform based techniques.

Spatial based steganography hides data directly in image pixels e.g., LSB embedding. LSB embedding is the simplest method for hiding the secret data by replacing the least significant bit of each cover image pixel with secret data bit imperceptibly. In addition to being less suspicious for human eyes, LSB steganography is easy to implement. There exist different versions of LSB embedding to make steganography more secure. LSB plus-minus (LSBPM) [16] also known as LSB matching, is one of the advancements in standard LSB replacement steganography. LSBPM adds or subtracts '1' from the least significant bit of cover image pixel according to secret message bit. Spatial steganography is sensitive to filtering, scaling, translation, rotation and cropping on stego image. For increasing the security, LSB method can be combined with other steganography methods. Spatial steganography is further classified into sequential and random embedding. Sequential embedding starts from the first bit of the cover image and sequentially proceeds until no secret bit is left. On the contrary, random embedding embeds the secret message bits randomly in the cover image, hence it increases security by providing no fixed pattern for detection.

Frequency transformation based steganography overcomes the limitations of spatial based techniques. In frequency based techniques, the cover image is transformed from spatial to frequency domain and the coefficients of transform are modified to embed secret data. Hence, transform domain based steganography is less perceptible with less capacity, though it complicates the implementation. Some examples of frequency based steganography are DCT (Discrete Cosine Transformation), DWT (Discrete Wavelet Transformation), and DFT (Discrete Fourier Transformation) based steganography. In DCT based techniques, the image is divided into 8×8 blocks followed by quantization of each block. Later, quantized components are used for embedding, finally inverse DCT is applied to get back the image. Different types of steganography and steganalysis techniques are illustrated in Fig. 1. In this paper, LSB steganography is discussed for its security with attacks.

Steganography affects perceptional and statistical properties of the image. Embedding can be noticed by comparing pixel values and the file size of the original and stego images. Statistical properties consist of the mean value, standard deviation, histogram modification, etc. Due to the addition of information, steganography affects statistical properties internally. So, steganalysis uses the embedding side-effects to detect their existence.

Steganalysis is orthogonal to steganography which targets affected image properties to enhance steganography security. Steganalysis reveals the secret communication by inspecting suspected image [2, 6, 9, 10]. Advancement in steganography also results in a new steganalysis technique, hence both the fields are explored

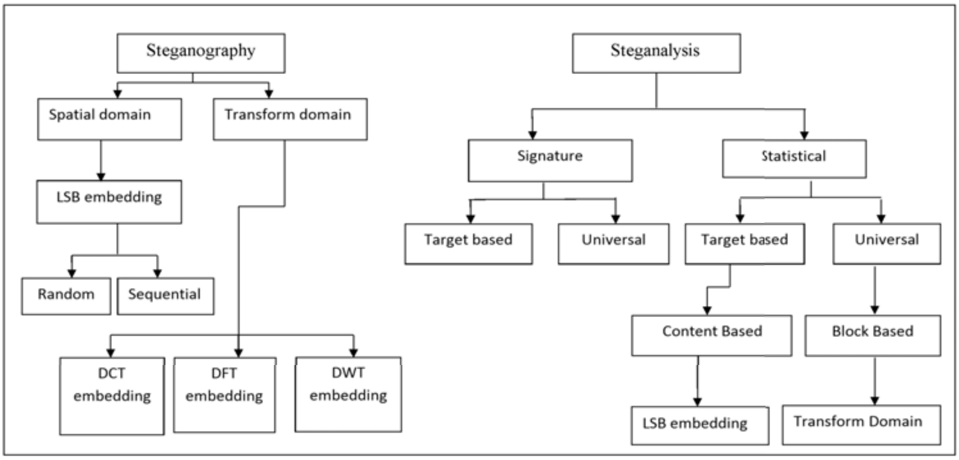


Figure 1: Steganography and Steganalysis classification [18]

by researchers equally. Based on information about steganography technique and the original image, steganalysis is categorized into two classes: blind steganalysis and targeted steganalysis. In blind steganalysis, no information of the cover image and steganography technique is available which makes it comparatively more challenging and realistic than that of targeted steganalysis [3, 13, 15].

In targeted steganalysis, embedding method with original image is provided along with given stego image. It depicts better detection with no random assumptions [8, 9, 17]. On the basis of applied steganography technique, targeted steganalysis is further divided into two categories: content based steganalysis and block based steganalysis [18]. Content based steganalysis produces a detector for LSB based steganography [22]. Block based steganalysis develops the method to detect transform based steganography. Here, blocks can vary in size and each block can adopt a different method for secrecy detection [4, 7, 14, 21].

In content based steganalysis, spatial domain based steganography techniques are targeted. Since LSB steganography modifies the direct pixel values, detection method inspects the modified pixels and related properties. Fridrich [19] used first and second order Markov chains for imitating the difference between adjacent pixels before and after LSB embedding. Resultant difference matrix acted as a feature set for classification using SVM. In this paper, we work on content based steganalysis for imperceptibility and embedding capacity.

Steganography methods can be considered as statistically affected and non-statistically affected methods. Non-statistically affected steganography techniques embed data without affecting the statistical properties of the image, while statistically affected steganography modifies the statistical properties of the original image. In this paper, we are targeting statistically affected LSB embedding methods by considering information theoretic features as steganalysis measures. First, the stego image is processed through the feature extraction phase. Next, entropy and joint entropy features are selected for discriminating original image from the stego image. The proposed method is also analyzed for reliability by attacking the stego image. We use a dataset of grayscale images for all of our experiments. Experimental results show comparison of entropy and joint entropy for steganalysis. For classification purpose, we use entropy and joint entropy features to train multiple classifiers. Based on detection accuracy, we choose SVM and ensemble classifiers for final results.

Rest of the paper is organized as follows. A survey of the related work is included in Section 2. Section 3 gives an introduction of LSB embedding and information theoretic metrics. Section 4 explains the information theoretic measures for image steganalysis and their effects as used in this work. Section 5 presents the performance evaluation and experimental results. Finally, the work is concluded in Section 6.

2 Literature Survey

Existing research in targeted steganalysis has extracted features from the suspected and original images individually. In recent years, the number of features has increased for advanced and accurate detection. Pevny et al. [19] proposed first and second order Markov chains for imitating the difference between adjacent pixels after and before LSB embedding. Resultant difference matrix acted as a feature set to classify cover image from stego image using SVM.

Fillatre & Lionel [5] added on by introducing an asymptotically and uniformly more powerful test to find out the hidden message bits irrespective of the hiding ratio. This technique explored the parametric model of the natural images where physical dependency between pixels is exploited. Generally, it sets the upper bound of hidden message bits for LSB embedding. Lerch-Hostalot & Megías [12] proposed the LSB detection idea by comparing the correlation of modified and adjacent pixels prior and post-embedding. The pattern generated after analysis of stego file is compared with the pattern of pixels in the original media. Results depicted that there exists a strong dependency in pixels of natural image that gets affected after embedding.

Kodovsky & Fridrich [11] worked on detection accuracy of steganalysis scheme by preprocessing the cover image. In her proposal, downsampling of the cover image is done before applying LSB steganography technique. Observations showed that processing of cover image affects the detectability. It draws attention toward the processing of cover image before analysis by comparing directly cover and stego images.

Sadat et al. [20] introduced entropy for motion vector steganalysis. He deployed block entropy to determine the texture and precision of the motion vector to differentiate between high and low textured blocks. High textured blocks were used as effective features and used in re-estimation of the motion equation. It shows that block-entropy classifies the video into cover and stego files with more precision.

In this paper, the developed steganalysis methods exploit features of stego and cover images for comparison. An enhanced technique can consider joint information from stego and cover images. In the proposed work, both original and suspected images are processed to extract a common measure for detection by using information theory. Here, LSB steganography is targeted with entropy and joint entropy for extracting information. Reliability of both metrics is analyzed by further attacking the image. It is shown that joint entropy is an appropriate steganalysis measure in comparison to entropy. Further extracted information is fed into classifiers. From experiments, we infer that Support Vector Machine (SVM) and ensemble classifiers give better detection accuracy. Finally, We conclude that joint entropy improves detection by using information from both the original and the stego images.

3 Background

3.1 LSB Embedding

LSB embedding is the most straightforward and standard steganography method that influences the content of an image by modifying the least significant bits of each pixel. Consider an 8-bit grayscale image I_c with $M \times N$ pixels and a secret message S with n bits to be communicated secretly by I_c, over a local channel in an imperceptible way. Let the last m bits of the cover image be replaced according to the message length and the quality requirements of the stego image, where

$$I_c = \{x_{ij} \mid \quad 0 \leq i < M, \quad 0 \leq j < N\}, \tag{1}$$

$$x_{ij} \in \quad \{0, 1, \ldots, 255\}, \quad and$$

$$S = \{s_k \mid \quad 0 \leq k < n, \ s_k \in \{0, 1\}\}. \tag{2}$$

First, S should be compatible to a cover image so that the quality of the image can be maintained after embedding. Therefore, the rightmost bit of the pixels should be used for embedding. However, the quality and embedding capacity requirements during steganography need more embedding bits with less perceptibility. Thus for embedding, we consider m rightmost bits from each selected pixels, and the value of m depends on the requirement of the method. Generally, the preferred value of m is less than 4, as up to there, quality gets affected in imperceptible range. For this purpose, S is rearranged to form a virtual image I' compatible to the original image, such that,

$$I' = \{y_i \mid \quad 0 \leq i < n'\}, \tag{3}$$

where,

$$y_i \in \{0, 1, \ldots, 2^m - 1\},$$

$n' < M \times N$, and I' contains non binary values for $m > 1$. Binary message S is mapped to non binary message I' by following equation:

$$y_i = \sum_{k=0}^{m-1} S_{i \times m + k} \ \cdot \ 2^{m-1-k}. \tag{4}$$

Now, embedding of I' is done by selecting x_l from I_c and replacing m least significant bits with secret message bits to form z_i as:

$$z_i = x_l \ - \ x_l \ mod \ 2^m, \tag{5}$$

where z_i is a pixel of the stego image that keeps secret bits without revealing their existence and I_s is the corresponding stego image. At the receiving end, the same process is reversed to get the secret message bits, ss_i back with the image as:

$$ss_i = z_l \ mod \ 2^m. \tag{6}$$

In stego image, z_l pixels are selected that are modified during steganography. From the selected pixels of the stego image, m LSB bits are extracted and arranged in an informative form similar to a secret message.

3.2 Information Theory

Information entropy defines the uncertainty and predictability of a discrete system. Let X be a discrete random variable with D as its domain with probability mass function as:

$$P(X) = \sum_{x \in D} p(x), \tag{7}$$

then, entropy of the system X can be defined mathematically as:

$$H(X) = -\sum_{x \in D} p(x) \ log \ p(x). \tag{8}$$

Here, logarithm of base 2 is taken as information is revealed in binary form. Entropy represents information in bits. Mathematical representation of entropy shows that it is inversely proportional to probability mass function of the events in the system. For an instance, if an event has maximum probability then it will reveal less information with more certainty. Hence, entropy depends on probability of the event. In terms of expectation $E(X)$ of the system, entropy is defined as:

$$H(X) = -E(X) \ log \ P(X). \tag{9}$$

Uncertainty associated with a set of random variables is termed as joint entropy. Joint entropy reveals information by considering more than one random variable at the same time. Let X and Y are two discrete random variables with U and V

as their respective domains having joint distribution $P(X, Y)$. Then, joint entropy $H(X, Y)$ for pair (X, Y) is defined as:

$$H(X,Y) = -\sum_{x \in U} \sum_{y \in V} p(x,y)\ log\ p(x,y). \tag{10}$$

Also, when we have expectation value by observing X and Y together, then joint entropy can be defined as:

$$H(X,Y) = -E(X,Y)\ log\ P(X,Y). \tag{11}$$

The joint entropy can also be formulated in terms of individual entropies of X and Y, i.e., $H(X)$ and $H(Y)$ respectively, with conditional entropy. The conditional entropy, $H(X|Y)$ of X and Y is the amount of information about X with prior information of Y. Then, joint entropy is defined as:

$$H(X,Y) = H(Y) + H(X|Y) = H(X) + H(Y|X). \tag{12}$$

In case of independent random variables, $H(X|Y) = H(Y|X) = 0$, as there is no benefit of prior information from another variable. Therefore,

$$H(X,Y) \leq H(X) + H(Y) \geq 0, \tag{13}$$

and,

$$H(X,Y) \geq max(H(X), H(Y)) \tag{14}$$

In this paper, random variables are related to an image system in order to compute entropy and joint entropy of the system. An image is a bundle of pixels in a fixed pattern to deliver related information. Image entropy extracts information of its uniformity and randomness. For an individual image, entropy reveals the information needed while joint entropy is used for knowing the information about the images at the same time. Here, entropy and joint entropy of the images are computed to get related information for detecting LSB steganography. Joint entropy needs joint probability distribution for its computation that requires joint histogram of the participating images. The joint histogram is a 2-dimensional representation of 1-dimensional histograms, where, the first dimension is for the intensity of one image and second dimension for another image. So, instead of evaluating detection measures separately for the original and stego images, joint entropy extracts combined information from both the images. Hence, we speculate that entropy and joint entropy can be used to discriminate the stego image from the cover image.

4 Proposed Information Theoretic Steganalysis

In this section, steganalysis for LSB steganography is proposed with information theoretic metrics. The proposed system works in two phases: first is embedding & training phase, followed by the second phase of verification. These two phases are described in subsequent subsections. Block diagram of the proposed information theoretic image steganalysis is shown in Fig 2.

- Embedding & Training Phase: For communicating secretly, a secret file is needed, which is to be embedded in a media file. Initially, an image and a text file as a secret message are given as input for LSB steganography system. After embedding, steganalysis scheme extracts entropy and joint entropy of the embedded image. Next, according to the behavior of the extracted features, we train the classifier for discriminating stego image from the cover image. We consider the SVM and ensemble classifiers for the proposed scheme. We conduct experiments with suspected image for entropy and joint entropy-based measures for detection of the existence of the steganography.

  Empirical results show the increase in original joint entropy during the training phase that indicates the existence of steganography. On the other hand entropy showed no appreciable change in its original value. Thus, joint entropy is an effective measure for steganalysis in comparison to entropy.

- Verification Phase. In this phase, system inspects the suspected image by scheme A and scheme B. In scheme A, entropy and joint entropy are used to detect steganography by trained classifiers. Results show that joint entropy can easily detect embedding while entropy shows no significant participation in steganalysis. In scheme B, the histogram of the given image is attacked. After that resultant image is analyzed by training classifiers for entropy and joint entropy measures thus making a final decision whether steganography is still detectable or not. According to the experimental results of scheme B, joint entropy is proved to be an effective measure for detection also with the attack on stego image, while entropy shows no change in its behavior.

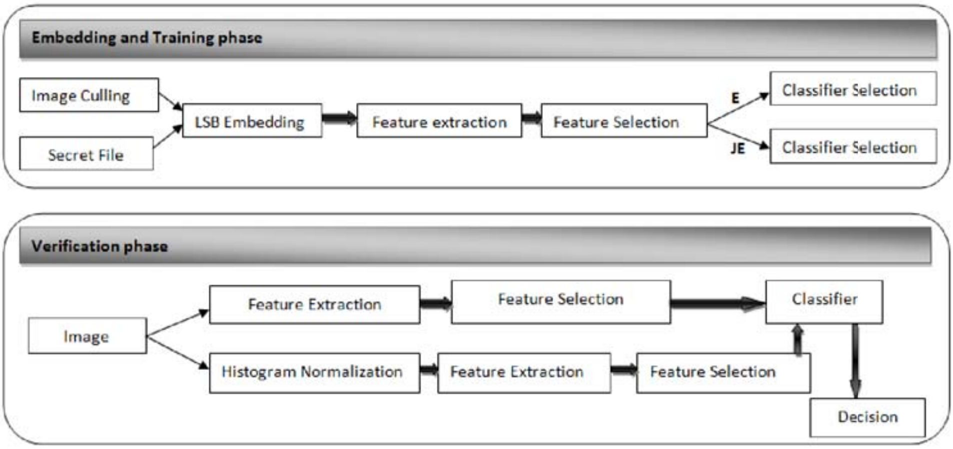


Figure 2: The proposed system framework

It is worthwhile to compare information theoretic measures for image steganalysis with individual feature based steganalysis. Traditional steganalysis focuses on image size, statistical properties, energy and contrast of the image for detection of steganography, while we introduce entropy and joint entropy for detection purpose. For analyzing the robustness of proposed metrics, histogram of the stego image is modified and then entropy and joint entropy features of the image are used for steganalysis. Further, classifiers are trained and tested with the proposed measures for efficient detection.

4.1 Embedding & Training Phase

Adaptation of steganography technique ensures the embedding of secret message in the cover image. Here, the cover image file is embedded with secret file by using LSB steganography method. In training phase, the proposed steganalysis targets applied steganography for its detection. First, features of the suspected (stego) image are extracted followed by relevant feature selection for detection. Next, we train the classifiers according to the decided discrimination function. For the proposed method, entropy and joint entropy metrics are selected during feature selection.

- *Entropy*: Entropy is one of the features that may get affected by embedding. Since information is added into the cover image by modifying its original contents, so randomness may either increase or decrease. However, steganography produces stego image perceptibly similar to cover image, and entropy is the factor that may affect during the embedding phase. Thus, we apply entropy metric over steganography to detect its existence.

- *Joint Entropy*: While entropy of the image can be extracted for steganalysis, joint entropy of stego image with the cover image can be a reliable metric. Since joint entropy considers stego and cover images together for the joint information, this may distinguish stego and cover images more clearly. After analysis with entropy and joint entropy, experimental results show that joint entropy gives a fixed pattern for different hiding ratio. Hence, joint entropy performed better than entropy, as joint entropy gives results with a fewer false alarms. Joint entropy does not change its behavior after the attack.

Above, feature selection process is followed by a classifier selection and training stage. We experimented with Fischer linear discriminate (FLD), logistic, support vector machine (SVM) and ensemble classifiers, the last two gave better detection accuracy.

4.2 Verification Phase

For a given steganography method, a secret message is embedded into a cover image to get cover/stego set. In the proposed steganalysis technique, a cover/stego pair is processed through two different schemes as described below. Here, the verification phase includes testing of the given image with and without attack. The verification phase aims to check the given image pair with entropy and joint entropy metrics of the proposed schemes.

- *Scheme A*: Verification with entropy and joint entropy. One perceptive way is to feed the image pair into a trained classifier using the selected features. First, choose a cover/stego pair and extract entropy and joint entropy as features. Next, based on information entropy (IE), a discrimination function D_e is formulated for the classifier to discriminate between cover image I_c and stego image I_s as:

  $$D_e = \|IE(I_c) - IE(I_s)\|. \tag{15}$$

  Instead of evaluating individual entropy of cover and stego images, joint entropy of both the images is calculated. Joint entropy (JE) of cover and stego images is proposed here for another discrimination function D_j for classifier as:

  $$D_j = \|JE(I_c, I_c) - JE(I_c, I_s)\|. \tag{16}$$

  The given image pair is checked with both the discrimination functions, entropy and joint entropy. It is observed that the joint entropy metric gives better detection.

- *Scheme B*: Here, we verify reliability of entropy and joint entropy by attacking the image histogram. The stego image is modified by histogram normalization, and then reliability is inspected by a respective discrimination function. Histogram normalization works as:

  $$I_{s\_norm} = N(I_s) = \sum_{j=0}^{k} pr(n_j), \quad k = 0 \ldots L, \tag{17}$$

  where,

  $$n_j = \frac{j}{L},$$

  and k is the maximum intensity level used in the image. L is the upper bound of intensity level. I_{s_norm} is a normalized image, n_j is a normalized intensity level of the input image, $pr(n_j)$ is probability density function of the image with normalized levels. For example, as shown in Fig. 3, stego image is normalized by the given equations that result in Fig. 3(c-d). As in scheme A, the modified image is analyzed by entropy and joint entropy metrics, respectively, as given in the following two equations:

  $$D_{he} = \|IE(I_c) - IE(I_{s\_norm})\|, \tag{18}$$

  and,

  $$D_{hj} = \|JE(I_c, I_c) - JE(I_c, I_{s\_norm})\|. \tag{19}$$

Verification phase provides discrimination functions for classifiers to decide whether the given image is stego or original. According to the proposed discrimination functions after histogram attack, D_{he} for entropy metric and D_{hj} for joint entropy metric results show that joint entropy is still able to distinguish cover and stego images.

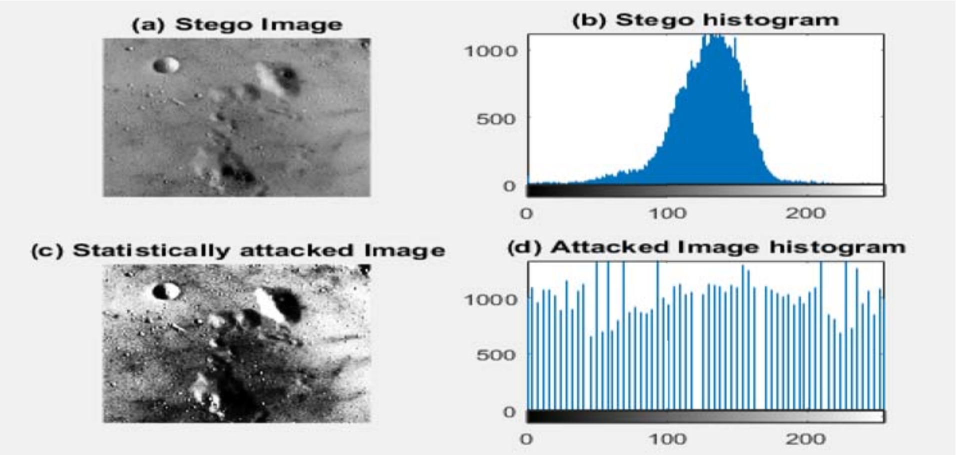


Figure 3: Example of a sample stego image and its normalized image with histogram: (a) Original stego image, (b) Histogram of the original stego image, (c) Stego image after histogram normalization, i.e., normalized stego image, a.k.a. the stego image after histogram attack, and (d) Histogram of the normalized stego image.

5 Performance Evaluation

The performance of information theoretic steganalysis is studied in this section. We compare both the proposed metrics – entropy and joint entropy – and then select joint entropy as a better metric for steganalysis. Therefore, we include initially results for both the entropy and joint entropy, and later we include detailed results for joint entropy only. We carry out experimental results using different secret files and binary classifiers for detection.

The performance of the steganalysis method is measured by detection accuracy rate with respect to the specific steganography:

$$P_{detect} = 1 - D_{error}, \tag{20}$$

where D_{error} is the probability of error difference. When we target a steganography method, the error difference between actual embedding and detected embedding acts as a steganalysis measure. Error difference is defined as a difference between original embedding features, o_{embed} and detected embedding features, d_{embed} :

$$D_{error} = o_{embed} - d_{embed}. \tag{21}$$

An ideal steganalysis technique has zero error difference with 100% accuracy. Detected embedding decision process includes false positives and false negatives. Targeted steganalysis tries to maximize detection accuracy by observing these two aspects. False positives give the false alarm by placing the cover image into stego image class, while false negatives contribute to false alarm by escaping stego image undetectable. So, extracted embedding during steganalysis scheme d_{embed} can be defined as:

$$d_{embed} = \frac{1}{2}(fp + fn), \tag{22}$$

where fp and fn are respective false positives and false negatives of steganalysis scheme. Therefore detection accuracy rate is:

$$P_{detect} = 1 - [o_{embed} - (\frac{1}{2}(fp + fn))]. \tag{23}$$

5.1 Experimental Setup

For experiments, we have used MATLAB 2017a Windows 7, CPU core i7 with 10 GB of RAM. We have examined 1000 grayscale images of dimensions 256 × 256, 384 × 512, 512 × 512 and 1024 × 1024 for training and testing. The dataset includes benchmark images taken from nature and some random captures, which are used in image processing and data hiding. A few images used from this image dataset, are shown in Fig. 4. For analyzing the effects of secret file size, we have taken 1000, 5000, 10000 and 20000 KB text files. Each image of the dataset is processed by LSB embedding methods with various secret text files.

After obtaining the stego and cover images, all the possible features are extracted, including entropy and joint entropy. Further, with extracted information theoretic features, every stego image is inspected for maximum embedding detection. We have used binary classifiers for decision making between the cover and stego images. For the initial training phase, we have used half of the images of the dataset and rest half for testing process. For further experiments, we have used a different ratio of training and testing sets from the dataset to examine the accuracy of the corresponding classifier. During the training and testing phase, the extracted features are compared with the original image while the verification phase checks the reliability of the proposed metrics by attacking the histogram of the image.

5.2 Performance Analysis of Proposed Metrics

For performance analysis of the proposed steganalysis features, we gather 1000 gray-scale images of different dimensions and use them as cover images. The spatial domain based steganography methods, namely, standard LSB embedding and LSB plus-minus (LSBPM) [16] are used to create corresponding stego image datasets. Furthermore, the produced stego images are inspected with entropy and joint entropy based features.

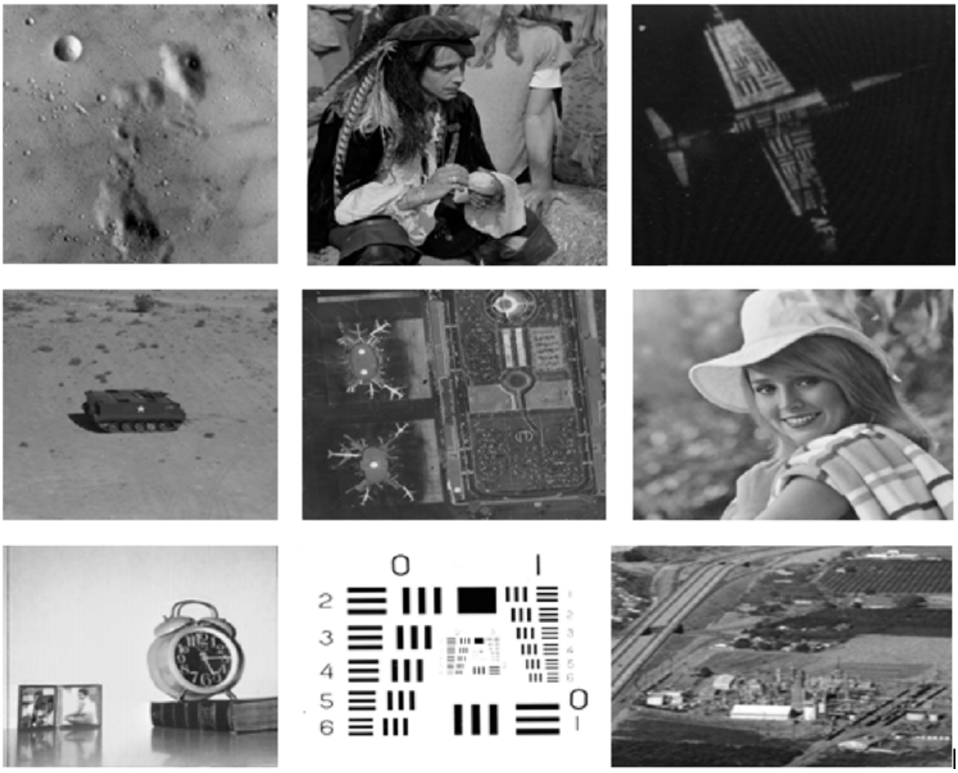


Figure 4: A few gray scale images taken from the dataset.

5.2.1 Entropy and joint entropy with varying hiding ratio

Both entropy and joint entropy are information theoretic metrics, however, the information gain during steganalysis makes the difference. From equations (15), (16), (18) and (19), we compare the performance of these metrics as:

$$P_{compare} = D_j - D_e, \tag{24}$$

or

$$P_{compare} = D_{hj} - D_{he}. \tag{25}$$

Here, the accuracy and efficiency of the proposed metrics are analyzed for detecting LSB and LSBPM steganography methods. First, the stego image generated by steganography methods is experimented for detection purpose by using entropy and joint entropy metrics. We repeated experiments over a set of twenty images selected randomly. Since the behavior remains the same throughout our experiments, hence we represent the results of one representative sample of our experiments in Table 1 due to space consideration. Table 1 represents the results obtained for entropy and joint entropy based features for detecting standard LSB embedding and LSBPM steganography methods.

Table 1: Entropy and joint entropy for LSB and LSBPM steganography methods over a randomly selected image.

steganograpy	Bit Rate	Entropy	Joint entropy
LSB	0.20	6.709312	6.709312
	0.45	6.709350	6.749539
	0.73	6.709482	7.700813
	0.99	6.709482	7.700813
LSBPM	0.20	6.756524	6.701335
	0.45	6.761320	6.790011
	0.73	6.774975	7.713119
	0.99	6.786145	7.895221

From Table 1, we observe that the standard LSB embedding method does not show any significant variation in entropy values, while joint entropy increases with increase in bit rate. Almost similar is the case with LSBPM steganography, in which the entropy values are increased very marginally, while joint entropy increases significantly over increase in bit-rate. We also got the similar pattern of variations of entropy and joint entropy with varying bit-rates after histogram attack. This is discussed in the next sub-section. Therefore, we conclude from these results that joint entropy is an effective measure over entropy for detection.

5.2.2 Effect of histogram attack

Further, performance of entropy and joint entropy is explored by attacking histograms of the stego images. Here, the stego images of used dataset with histogram attack are experimented to check the metrics efficiency. Table 2 shows a representative sample result for showing the reliability of proposed features for detecting LSB steganography methods after histogram attack.

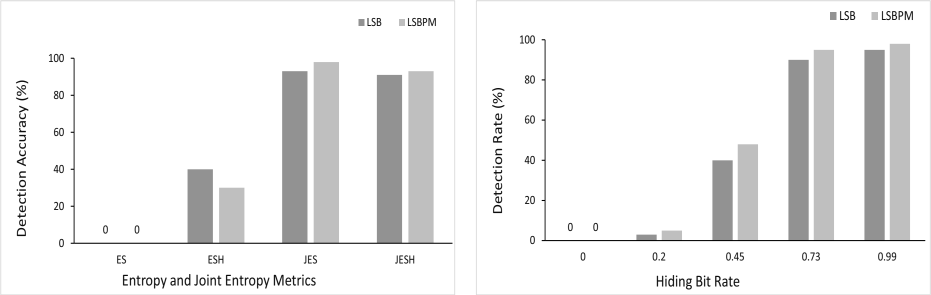
Table 2: Entropy and joint entropy for LSB and LSBPM steganography methods after histogram attack over a randomly selected image.

steganograpy	Bit Rate	Entropy	Joint entropy
LSB	0.20	5.921757	6.709312
	0.45	5.921710	6.743879
	0.73	5.918913	7.475977
	0.99	5.918913	7.475977
LSBPM	0.20	5.935647	6.712546
	0.45	5.938745	6.735481
	0.73	5.949965	7.328165
	0.99	5.949965	7.452096

From Table 2, we observe the similar behaviour as in Table 1, that there is no appreciable change in entropy values with varying bit-rates, while joint entropy increases with increase in hiding bit-rates. From these experimental results, taken over a range of grey-level images with varying hiding rates and histogram attacks, we conclude that (i) entropy is not a discriminative feature, while (ii) joint entropy can be used reliably for both the LSB and LSBPM steganography methods with varying hiding ratio and with histogram attacks.

Next, we experiment both the metrics for detection accuracy before and after the statistical, i.e., histogram attack. We calculate detection accuracy with entropy and joint entropy, before and after histogram attack, for both LSB and LSBPM methods. The entropy metric before and after histogram attack does not indicate any pattern for detecting steganography. Thus, the detection accuracy with entropy for both the methods, i.e., before and after attacks, is quite low.

However, the joint entropy metric deflects after embedding in both cases, i.e., before and after attack. The stego image has marginally higher joint entropy with respect to the original image. The detection accuracy with joint entropy is found to be above 90% for all the cases across all the randomly selected images. The results are depicted in Fig. 5(a) for both entropy and joint entropy before and after histogram attack.



(a) (b)

Figure 5: (a) Detection accuracy with entropy and joint entropy (before and after attack): ES (Entropy of stego image), ESH (Entropy of stego image after histogram attack), JES (Joint entropy of stego image), JESH (Joint entropy of stego image after histogram attack), and (b) Detection rate by joint entropy with varying hiding rate for LSB and LSBPM steganography methods.

All the results presented herein above in Tables 1 & 2 and Fig. 5(a), discard use of entropy as an information theoretic feature for steganalysis. Next, we calculate joint entropy with different hiding ratios for both LSB and LSBPM methods, and plot in Fig. 5(b). These plots show that joint entropy varies significantly with different hiding ratios. Thus, joint entropy is shown to be an effective metric for detection of steganography with histogram attack. All the above results computed for joint entropy, finally, establish joint entropy as a distinctive feature for steganalysis.

Table 3: Joint entropy with different classifiers.

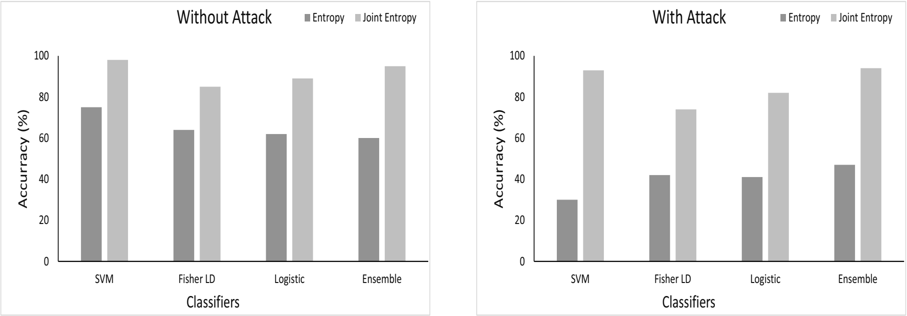
Classifier	Detection			
	Correct	Incorrect	Total	Accuracy
SVM	245	5	250	0.98
Fisher LD	454	78	535	0.85
Logistic	135	17	152	0.89
Ensemble	570	32	602	0.95

5.2.3 Detection accuracy with classifiers

We use two class classifiers for discrimination between cover and stego images. For classification, we selected four classifiers, namely, Support Vector Machine (SVM), Fisher linear discriminant (Fisher LD), logistic, and ensemble classifiers. Table 3 includes the results of joint entropy with these four classifiers before the attack. The SVM and ensemble classifiers have shown higher accuracy over the Fisher LD and logistic classifiers. The corresponding classification results of joint entropy, after histogram attack, are shown in Table 4. In analogy with the classification results before attack, the results of SVM and ensemble classifiers, after attack, have higher accuracy over the other two classifiers.

Table 4: Joint entropy after histogram attack with different classifiers.

Classifier	Detection			
	Correct	Incorrect	Total	Accuracy
SVM	232	18	250	0.93
Fisher LD	396	139	535	0.74
Logistic	124	28	152	0.82
Ensemble	566	36	602	0.94

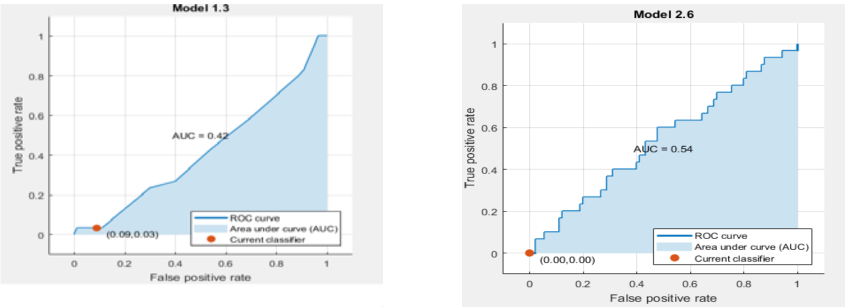


(a) (b)

Figure 6: Classification results using entropy and joint entropy: (a) before attack, and (b) after attack.

The corresponding classification results of Tables 3 & 4 are plotted in Fig. 6; the plots in Fig. 6(a) represent classification accuracy before attack, and plots in Fig. 6(b) represent the results after attack. In addition, the plots in Fig. 6 also include classification accuracy with entropy. It can be observed that the detection accuracy is much higher with joint entropy metric over entropy. Moreover, the detection accuracy with entropy decreases very significantly after the statistical histogram attack. For all classifiers, the classification accuracy with joint entropy, before and after attack, remains above 80%, this is above 90% with the state of art classifiers, namely, SVM and ensemble classifiers.

Finally, we plot ROC curves for joint entropy with respect to the steganography detection before and after the attack in Fig. 7. These experiments are done by averaging observations from LSB and LSBPM steganography.



(a) (b)

Figure 7: ROC curve with joint entropy: (a) without attack, and (b) with attack.

In summary, the steganalysis performance with information theoretic metrics, namely, entropy and joint entropy, for detecting LSB class of steganography, is shown by Table 1 to Table 4 and Fig. 5 to Fig. 7. As the embedding rate increases, joint entropy is shown to be an effective detection metric with a better accuracy rate in comparison to entropy. The detection rate with entropy metric, before the attack, is 70%, and after histogram attack, it decreases to 30%. In contrast, the same detection rate with joint entropy metric before the attack is 98%, and remains almost the same after the attack, it reduces very marginally to 93%. Therefore, the detection of LSB steganography methods by joint entropy metric is much superior than that of entropy metric.

6 Conclusion

In this work, information theoretic measures for steganalysis are proposed for detecting LSB steganography methods. Existing steganalysis schemes distinguish the cover and stego images based on their individual information. In the proposed scheme, images with their original versions are used to extract entropy and joint entropy based features. Extracted features were used to distinguish stego and cover images by using SVM and ensemble classifiers. For checking the efficiency of the method, images were attacked statistically and detection accuracy was then measured. Experimental results showed that entropy is not a reliable measure for the detection of LSB steganography, while joint entropy is shown to be quite discriminative.

In order to analyze the proposed detection measures for different hiding ratio, we have examined entropy and joint entropy over different secret files. Detection accuracy by joint entropy is observed to be better than the detection accuracy by entropy. For classification between cover and stego images, a few classifiers are compared in terms of false alarms, correct prediction and accuracy; the SVM and ensemble classifiers show maximum accuracy.

Future research augmentation can be to use the proposed metrics for detecting frequency based steganography. Further analysis can be done with other attacks and observing their effects on detection accuracy. In order to define a standard steganalysis measure, one may work to derive a well-formed metric using joint entropy and compare its performance with other well-known steganalysis methods.

Acknowledgements

The authors thank Managing Editor Boglárka G.-Tóth, Associate Editor Zoltan Kato, and anonymous reviewers for their help and bringing clarity in presentation of this work. The author, Rajeev Kumar, also acknowledges help of Roopam Sadh, Neeraj Pathak, and Priti Kumari for carrying-out some of the experimental results for major revision of this manuscript.

References

[1] Amin, Muhalim Mohamed, Salleh, Mazleena, Ibrahim, Subariah, Katmin, Mohd Rozi, and Shamsuddin, MZI. Information hiding using steganography. In *Proc. 4th Nat. Conf. Telecommunication Technology (NCTT)*, pages 21–25. IEEE, 2003. DOI: 10.1109/NCTT.2003.1188294.

[2] Chandramouli, Rajarathnam, Kharrazi, Mehdi, and Memon, Nasir. Image steganography and steganalysis: Concepts and practice. In *Proc. Int. Workshop Digital Watermarking (IWDW), LNCS, vol. 2939*, pages 35–49. Springer, 2003. DOI: 10.1007/978-3-540-24624-4_3.

[3] Chen, Xiaochuan, Wang, Yunhong, Tan, Tieniu, and Guo, Lei. Blind image steganalysis based on statistical analysis of empirical matrix. In *Proc. 18th Int. Conf. Pattern Recognition (ICPR)*, volume 3, pages 1107–1110. IEEE, 2006. DOI: 10.1109/ICPR.2006.332.

[4] Do, Minh N and Vetterli, Martin. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. *IEEE Trans. Image Processing*, 11(2):146–158, 2002. DOI: `10.1109/83.982822`.

[5] Fillatre, Lionel. Adaptive steganalysis of least significant bit replacement in grayscale natural images. *IEEE Trans. Signal Processing*, 60(2):556–569, 2011. DOI: `10.1109/TSP.2011.2174231`.

[6] Fridrich, Jessica, Goljan, Miroslav, and Du, Rui. Detecting LSB steganography in color, and gray-scale images. *IEEE Multimedia*, 8(4):22–28, 2001. DOI: `10.1109/93.959097`.

[7] Fu, Dongdong, Shi, Yun Q, Zou, Dekun, and Xuan, Guorong. Jpeg steganalysis using empirical transition matrix in block DCT domain. In *Proc. IEEE Workshop Multimedia Signal Processing*, pages 310–313. IEEE, 2006. DOI: `10.1109/MMSP.2006.285320`.

[8] Hawi, Tariq Al, Qutayri, MA, and Barada, Hassan. Steganalysis attacks on stego-images using stego-signatures and statistical image properties. In *Proc. IEEE Region 10 Conf. TENCON*, pages 104–107. IEEE, 2004. DOI: `10.1109/TENCON.2004.1414542`.

[9] Johnson, Neil F and Jajodia, Sushil. Steganalysis of images created using current steganography software. In *Proc. Int. Workshop Information Hiding (IWIH), LNCS, vol. 1525*, pages 273–289. Springer, 1998. DOI: `10.1007/3-540-49380-8_19`.

[10] Johnson, Neil F and Jajodia, Sushil. Steganalysis: The investigation of hidden information. In *Proc. IEEE Information Technology Conf. Information Environment for the Future (Cat. No. 98EX228)*, pages 113–116. IEEE, 1998. DOI: `10.1109/IT.1998.713394`.

[11] Kodovsky, Jan and Fridrich, Jessica. Effect of image downsampling on steganographic security. *IEEE Trans. Information Forensics & Security*, 9(5):752–762, 2014. DOI: `10.1109/TIFS.2014.2309054`.

[12] Lerch-Hostalot, Daniel and Megías, David. LSB matching steganalysis based on patterns of pixel differences and random embedding. *Computers & Security*, 32:192–206, 2013. DOI: `10.1016/j.cose.2012.11.005`.

[13] Lie, Wen-Nung and Lin, Guo-Shiang. A feature-based classification technique for blind image steganalysis. *IEEE Trans. Multimedia*, 7(6):1007–1020, 2005. DOI: `10.1109/TMM.2005.858377`.

[14] Liu, Shaohui, Yao, Hongxun, and Gao, Wen. Steganalysis of data hiding techniques in wavelet domain. In *Proc. Int. Conf. Information Technology: Coding and Computing (ITCC)*, volume 1, pages 751–754. IEEE, 2004. DOI: `10.1109/ITCC.2004.1286558`.

[15] McBride, Brent, Peterson, Gilbert, and Gustafson, Steven. A new blind method for detecting novel steganography. *Digital Investigation*, 2:50–70, 02 2005. DOI: `10.1016/j.diin.2005.01.003`.

[16] Mielikainen, Jarno. LSB matching revisited. *IEEE Signal Processing Letters*, 13(5):285–287, 2006. DOI: `10.1109/LSP.2006.870357`.

[17] Niimi, Michiharu, Eason, Richard O, Noda, Hideki, and Kawaguchi, Eiji. Intensity histogram steganalysis in BPCS-steganography. In *IS&T/SPIE Electronic Imaging*. SPIE, 2001. DOI: `10.1117/12.435440`.

[18] Nissar, Arooj and Mir, Ajaz Hussain. Classification of steganalysis techniques: A study. *Digital Signal Processing*, 20(6):1758–1770, 2010. DOI: `10.1016/j.dsp.2010.02.003`.

[19] Pevny, Tomáš, Bas, Patrick, and Fridrich, Jessica. Steganalysis by subtractive pixel adjacency matrix. *IEEE Trans. Information Forensics & Security*, 5(2):215–224, 2010. DOI: `10.1109/TIFS.2010.2045842`.

[20] Sadat, Elaheh, Faez, Karim, and Saffari Pour, Mohsen. Entropy-based video steganalysis of motion vectors. *Entropy*, 20(4):244, 2018. DOI: `10.3390/e20040244`.

[21] Sullivan, Kenneth, Bi, Zhiqiang, Madhow, Upamanyu, Chandrasekaran, Shivkumar, and Manjunath, BS. Steganalysis of quantization index modulation data hiding. In *Proc. Int. Conf. Image Processing (ICIP).*, volume 2, pages 1165–1168. IEEE, 2004. DOI: `10.1109/ICIP.2004.1419511`.

[22] Trivedi, Shalin and Chandramouli, Rajarathnam. Active steganalysis of sequential steganography. In *Proc. Security and Watermarking of Multimedia Contents V*, volume 5020, pages 123–130. Int. Society for Optics & Photonics, 2003. DOI: `10.1117/12.473115`.

Received 4th January 2019

Acta Cybernetica **24** (2020) 613–623.

On the Steps of Emil Post: from Normal Systems to the Correspondence Decision Problem*

Vesa Halava[a] and Tero Harju[b]

Abstract

In 1946 Emil Leon Post (*Bull. Amer. Math. Soc.* **52** (1946), 264–268) introduced his famous correspondence decision problem, nowadays known as the Post Correspondence Problem (PCP). Post proved the undecidability of the PCP by a reduction from his normal systems. In the present article we follow the steps of Post, and give another, somewhat simpler and more straightforward proof of the undecidability of the problem by using the same source of reductions as Post did. We investigate these, very different, techniques, and point out out some peculiarities in the approach taken by Post.

Keywords: normal systems, Post correspondence problem, undecidability, assertion problem

1 Introduction

The original formulation of the *Post correspondence problem* (or, as Post called it, the *correspondence decision problem* [8]), PCP for short, is stated as follows:

Problem 1 (Post Correspondence Problem). *Let $A = \{a, b\}$ be a binary alphabet, and denote by A^* the set of all finite words over A. Given a finite set of pairs of words in A^*,*

$$W = \{(u_i, v_i) \mid u_i, v_i \in A^*,\ i = 1, 2, \dots, n\},$$

does there exist a nonempty sequence $i_1, i_2, \dots, i_k$ of indices, where $i_j \in \{1, 2, \dots, n\}$ for $1 \le j \le k$, such that

$$u_{i_1} u_{i_2} \cdots u_{i_k} = v_{i_1} v_{i_2} \cdots v_{i_k}\,? \tag{1}$$

[*] The first author was supported by emmy.network foundation under the aegis of the Fondation de Luxembourg.

[a] Complex Systems Research Group, Department of Mathematics and Statistics, University of Turku, Turku, Finland. E-mail: vesa.halava@utu.fi, ORCID: https://orcid.org/0000-0003-3633-4902.

[b] Department of Mathematics and Statistics, University of Turku, Turku, Finland. E-mail: harju@utu.fi, ORCID: https://orcid.org/0000-0002-9640-6309.

DOI: 10.14232/actacyb.284625

In the history of computability, the Post correspondence problem and its many variants have played an important role as simply defined algorithmically undecidable problems that can be used to prove other undecidability results. Here we concentrate on the undecidability proofs of the PCP itself.

A standard textbook proof of the undecidability of the PCP employs the *halting problem of the Turing machines* as the base of the reduction; see e.g. [9], or the construction by Claus [2] from the *word problem of the semi-Thue systems*, which gives better undecidability bounds on the number of pairs in the sets. The integer $n = |W|$ in Problem 1 is said to be the *size* of the set W. The set W is called an *instance* of the PCP. Recently, Neary [5] showed that the PCP is undecidable for $|W| = 5$ using the (Post) *tag systems* that form a special class of Post normal systems; see [6].

In his article [8], Post proved that the PCP is unsolvable, i.e., undecidable, by a technical and nontrivial reduction from the *assertion problem of the Post normal systems.* We shall give another proof by utilizing the same source.

There are several proofs of the PCP. The standard reductions from the Turing machines, semi-Thue systems and tag systems to the PCP have a common leading idea: An instance of the PCP is constructed so that any solution to the instance is a (encoded) concatenation of the configurations required in the computation or derivation of the original machine or system. This is *not the case* in Post's original proof. Indeed, he relies simply on the words in the rules of a derivation in a normal system. A sequence of these words imply a required derivation in the normal system if and only if the sequence is a solution of a particular instance of the PCP. The new proof presented in this article is based on the idea in the standard type – a solution exists to the constructed instance of the PCP if and only if the solution is a concatenation of the full configurations required of the given Post normal system.

We note that, in Post's definition, the PCP is defined for binary words. Actually, the cardinality of the alphabet A is not relevant, since every instance of the PCP with any alphabet size has an equivalent one in terms of binary words using an injective encoding into binary alphabet $\{a, b\}^*$ from A^*. For example, if $A = \{a_1, a_2, \ldots, a_k\}$, then φ defined by $\varphi(a_i) = a^i b$ is such an encoding. Note, however, that the PCP is decidable for sets of pairs W over unary alphabet.

The structure of this article is the following: In Section 2, we present the basic notions, notations needed in this article. Especially, we introduce the normal systems and present preliminary results on them. In Section 3 we present Post's construction, following Post's original article, but also give some explanatory steps for readability. In Section 4 we present our main contribution: another proof for the undecidability of the Post Correspondence Problem using the same source of undecidability as in Section 3 in the Post's construction.

Short preliminary version of this article can be found in [3].

2 Normal systems

Let A be a finite alphabet and denote by A^* the set of all finite words over A including the empty word ε. The length of a word u, i.e., the number of occurrences of letters in u, is denoted by $|u|$. For words u and w, the word u is a *prefix* of w if there exists a word v such that $w = uv$. If u is a prefix of w with $w = uv$, we denote the *suffix* v also by $u^{-1}w$.

For a word $w \in A^*$, if $w = a_1 \cdots a_{n-1}a_n$ where $a_i \in A$ for all $i = 1, \dots, n$, then the *reverse of* w is defined to be $w^{\mathrm{R}} = a_n a_{n-1} \cdots a_1$.

The words u and v are *(cyclic) conjugates* if there exist words x and y such that $u = xy$ and $v = yx$.

We give a formal definition of a normal system instead of the bit informal one used by Post in [8].

Let $A = \{a, b\}$ be a binary alphabet, and let X be a variable ranging over the words in A^*. A *normal system* $S = (w, P)$ consists of an *initial word* $w \in A^+$ and a finite set P of *rules* of the form $\alpha X \mapsto X\beta$, where $\alpha, \beta \in A^*$. We say that a word v is a *successor* of a word u, if there is a rule $\alpha X \mapsto X\beta$ in P such that $u = \alpha u'$ and $v = u'\beta$. We denote this by $u \to v$. Let $\to^*$ be the reflexive and transitive closure of $\to$. Then $u \to^* v$ holds if and only if $u = v$ or there is a finite sequence of words $u = v_1, v_2, \dots, v_n = v$ such that $v_i \to v_{i+1}$ for $i = 1, 2, \dots, n-1$. A normal system is a special case of the *Post canonical system* for which Post proved in 1943 the Normal-Form Theorem; see [6]. On the other hand, the tag systems mentioned in the introduction are a special class of the normal systems that have a constant length left for rule words α; see [6].

The *assertion* of a normal system $S = (w, P)$ is the set

$$\mathcal{A}_S = \{v \in A^* \mid w \to^* v\} \ . \tag{2}$$

Problem 2 (Assertion Problem). *Given a normal system $S = (w, P)$ and a word u, does $u \in \mathcal{A}_S$ hold?*

The following result is crucial for the construction presented in this article, but the reference for it is a bit peculiar: in footnote 2 of [8], Post gives citation to his paper [7] for an informal proof and to Church [1] for a formal proof, but with a comment that a verification of the recursiveness of the reduction is needed and then he gives guidelines for missing details on the footnote.

Proposition 1. *The assertion problem for normal systems is undecidable.*

Actually, the problem remains undecidable even if we assume that in each rule $\alpha X \mapsto X\beta$ in P the words α and β are non-empty; see Post [8], footnote 3. A normal system with non-empty rule words is called a *standard* normal system in the literature. Therefore, we can assume in the following that the normal systems are standard. This assumption is indeed crucial when we construct instances of the PCP from the normal systems in Sections 3 and 4.

3 Undecidability of the PCP by Emil Post

In this section we present the original proof and construction of Emil Post in [8]. Occasionally we use more modern terminology instead of Post's original terms.

Let $u \in \mathcal{A}_S$, where $S = (w, P)$. As the assertion problem is trivial for the case $u = w$, we assume that $u \neq w$. Therefore, there exists a sequence

$$w = \alpha_1 x_1,\ x_1\beta_1 = \alpha_2 x_2, \ldots, x_{k-1}\beta_{k-1} = \alpha_k x_k,\ x_k\beta_k = u\,, \tag{3}$$

where $\alpha_i X \to X\beta_i$ is a rule for each i with $x_j \in A^*$ for all j. The idea in Post's proof is to present the set of equations in (3) as a single equation in the form of a word equality (1).

Consider the word $w\beta_1\beta_2\cdots\beta_k$ obtained from (3). Using the equations in (3), we obtain, for each $j = 1, 2, \ldots, k$,

$$w\beta_1\beta_2\cdots\beta_{j-1} = \alpha_1\alpha_2\cdots\alpha_j x_j\,, \tag{4}$$

and finally

$$w\beta_1\beta_2\cdots\beta_k = \alpha_1\alpha_2\cdots\alpha_k u\,. \tag{5}$$

By (4), we have, for each $j = 1, 2, \ldots, k$,

$$|w\beta_1\beta_2\cdots\beta_{j-1}| \geq |\alpha_1\alpha_2\cdots\alpha_j|\,. \tag{6}$$

So we have shown that the derivation sequence (3) implies (4), which further implies (5) together with the inequalities (6). Actually, $u \in \mathcal{A}_S$, that is, existence of a sequence (3) is indeed equivalent to the join of (5) and (6). For this we need to prove the above implications in the opposite direction.

First, we show that the join of the equalities (5) and (6) imply the equations in (4). For this it suffices to choose

$$x_j = (\alpha_1\alpha_2\cdots\alpha_j)^{-1}(w\beta_1\beta_2\cdots\beta_{j-1})$$

for all j.

Furthermore, for the equations in (3), we obtain, for all $1 \leq j \leq k$,

$$\begin{aligned}\alpha_{j+1}x_{j+1} &= \alpha_{j+1}(\alpha_1\alpha_2\cdots\alpha_{j+1})^{-1}(w\beta_1\beta_2\cdots\beta_j)\\ &= (\alpha_1\alpha_2\cdots\alpha_j)^{-1}(w\beta_1\beta_2\cdots\beta_{j-1})\beta_j = x_j\beta_j\,,\end{aligned}$$

and we have the equations in (3) except the first and the last ones. The first one is obtained directly from (4) by setting $j = 1$. Also, the last one will follow, since

$$\alpha_1\alpha_2\cdots\alpha_k x_k\beta_k = w\beta_1\beta_2\cdots\beta_{k-1}\beta_k = \alpha_1\alpha_2\cdots\alpha_k u\,.$$

We have proved that (5) and (6) are satisfied if and only if the equations in (3) are satisfied, i.e., (5) and (6) are equivalent to the condition $u \in \mathcal{A}_S$.

Finally, we need to get rid of the extra condition (6). This is done by constructing a new normal system S_1, where (5) implies (6), and $uc \in \mathcal{A}_{S_1}$ if and only if $u^{\mathrm{R}} \in \mathcal{A}_S$ holds, where c is a new letter introduced below.

For this, let first $S' = (w^{\mathrm{R}}, P')$, where

$$P' = \left\{ X\alpha^{\mathrm{R}} \mapsto \beta^{\mathrm{R}} X \;\mid\; \alpha X \mapsto X\beta \in P \right\}.$$

Strictly speaking the system S' is not normal. It is a 'dual' of a normal system. However, we can still write $u \in \mathcal{A}_s$ if and only if $u^{\mathrm{R}} \in \mathcal{A}_{S'}$. Next we design a system $S'' = (w^{\mathrm{R}}c, P'')$, where c is a new letter. Let

$$P'' = \left\{ X\alpha^{\mathrm{R}}c \mapsto \beta^{\mathrm{R}} Xc \;\mid\; \alpha X \mapsto X\beta \in P \right\}.$$

It is immediate that $u^{\mathrm{R}} \in \mathcal{A}_{S'}$ if and only if $u^{\mathrm{R}}c \in \mathcal{A}_{S''}$. Obviously, S'' is even less normal as the letter c is kept constantly in the end of the words of the derivations.

Finally, let $S_1 = (w^{\mathrm{R}}c, P_1)$ be the normal system, where

$$P_1 = \left\{ \alpha^{\mathrm{R}}cX \mapsto Xc\beta^{\mathrm{R}} \mid \alpha X \mapsto X\beta \in P \right\} \cup \{ yX \mapsto Xy \mid y \in \{a,b,c\} \}.$$

Notice that the rules $yX \mapsto Xy$ for $y \in \{a,b,c\}$ imply that any sequence can be transformed to its conjugates. Therefore, if a rule is applied in S'', then the corresponding rule can be applied in S_1, since in the rules in P_1 the left hand sides are conjugates of the left hand sides of the rules in P'' and the right hand sides are conjugates of the right hand sides of the corresponding rules in P''. Let

$$\mathcal{C}_v = \{ w \mid w \text{ is a conjugate of } v \}$$

called the *conjugacy class* of the word v.

Then we have

$$\mathcal{A}_{S_1} = \mathcal{A}_{S''} \cup \bigcup_{v \in \mathcal{A}_{S''}} \mathcal{C}_v = (\mathcal{A}_{S'})^{\mathrm{R}} c \cup \bigcup_{v \in \mathcal{A}_{S'}} \mathcal{C}_{v^{\mathrm{R}}c} = (\mathcal{A}_S)^{\mathrm{R}} c \cup \bigcup_{v \in \mathcal{A}_S} \mathcal{C}_{v^{\mathrm{R}}c} = \bigcup_{v \in \mathcal{A}_S} \mathcal{C}_{v^{\mathrm{R}}c}.$$

To verify the above equality of sets, assume $u \in \mathcal{A}_{S_1}$. Denote by $x \overset{\mathcal{C}}{\to} y$ if $y \in \mathcal{C}_x$. For u, there exist a sequence of words and successors

$$\begin{gathered} w^R c \overset{\mathcal{C}}{\to} \alpha_1^R c x_1, \quad x_1 c \beta_1^R \overset{\mathcal{C}}{\to} \alpha_2^R c x_2, \quad x_2 c \beta_2^R \overset{\mathcal{C}}{\to} \alpha_3^R c x_3, \ldots, \\ x_{n-1} c \beta_{n-1}^R \overset{\mathcal{C}}{\to} \alpha_n^R c x_n, \quad x_n c \beta_n^R \overset{\mathcal{C}}{\to} u, \end{gathered} \tag{7}$$

where $\alpha^R cX \mapsto Xc\beta_n^R$ are in P_1 and $\overset{\mathcal{C}}{\to}$ part are done with rules $yX \mapsto Xy$ in S_1. Using cyclic shifts for all words in sequence (7) so that the special symbol c is the right most symbol of the word makes $\overset{\mathcal{C}}{\to}$ to be equality, and we get that there exists a sequence

$$\begin{gathered} w^R c = x_1 \alpha_1^R c, \quad \beta_1^R x_1 c = x_2 \alpha_2^R c, \quad \beta_2^R x_2 c = x_3 \alpha_3^R c, \ldots, \\ \beta_{n-1}^R x_{n-1} c = x_n \alpha_n^R c, \quad \beta_n^R x_n c \overset{\mathcal{C}}{\to} u. \end{gathered}$$

Now cancelling the letters c, taking reverse of all words and using the equality $(xy)^R = y^R x^R$, we have the sequence

$$\begin{gathered} w = (w^R)^R = (x_1 \alpha_1^R)^R = \alpha_1 x_1^R, \quad x_1^R \beta_1 = (\beta_1^R x_1)^R = (x_2 \alpha_2^R)^R = \alpha_2 x_2^R, \quad \ldots, \\ x_{n-1}^R \beta_{n-1} = (\beta_{n-1}^R x_{n-1})^R = (x_n \alpha_n^R)^R = \alpha_n x_n^R, \quad x_n^R \beta_n = v, \end{gathered}$$

for a word v with $vc \xrightarrow{\mathcal{C}} u^R$. We have proved that $u \in \mathcal{A}_{S_1}$ implies $u^R \in \mathcal{C}_{vc}$ for a word $v \in \mathcal{A}_S$. We note that $u^R \in \mathcal{C}_{vc}$ is equivalent to $u \in \mathcal{C}_{v^Rc}$. As the above verification works also in the other direction, we have proved that

$$\mathcal{A}_{S_1} = \bigcup_{v \in \mathcal{A}_S} \mathcal{C}_{v^{\mathrm{R}}c}.$$

Denote the rules of P_1 in the form $\gamma X \mapsto X\delta$. As in the above for the system S, we obtain that if $u^{\mathrm{R}}c \in \mathcal{A}_{S_1}$ then

$$w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_k = \gamma_1\gamma_2\cdots\gamma_k u^{\mathrm{R}}c\,, \tag{8}$$

where $\gamma_i X \mapsto X\delta_i \in P_1$ for each i. We shall prove that in S_1 the condition (8) implies the condition

$$|w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_{j-1}| \geq |\gamma_1\gamma_2\cdots\gamma_j|\,, \tag{9}$$

for all $j = 1, 2, \ldots, k$.

Assume contrary to the claim that there is a solution to (8) such that for some s,

$$|w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_{s-1}| < |\gamma_1\gamma_2\cdots\gamma_s|\,,$$

and let v be a nonempty word in $\{a,b,c\}^*$ such that

$$w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_{s-1}v = \gamma_1\gamma_2\cdots\gamma_s\,. \tag{10}$$

Now for each rule $\gamma_i X \mapsto X\delta_i$ of P_1, either both sides contain one c or neither of them contains c. Therefore if γ_s contains no occurrences of c then the left hand side of (10) would have at least one more occurrence of c than the right hand side; a contradiction. If c occurs in γ_s, then c is necessarily the last letter of γ_s and v would also end with c, and again the left hand side has more occurrences of the letter c than the right hand side; again a contradiction.

Therefore we have shown that $u \in \mathcal{A}_s$ if and only if $u^{\mathrm{R}}c \in \mathcal{A}_{S_1}$, which holds if and only if there exist rules $\gamma_i X \mapsto X\delta_i \in P_1$ for $i = 1, \ldots, k$ with

$$w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_k = \gamma_1\gamma_2\cdots\gamma_k u^{\mathrm{R}}c\,. \tag{11}$$

We then begin by the equation (11), and use the technique which is nowadays called desynchronization. Let d be a new symbol absent in S_1 and define a mapping $\ell_d : \{a,b,c\} \to \{a,b,c,d\}$ which writes the letter d before (to the left hand side of) every letter in a word, and define r_d similarly writing d to the right hand side of every letter. The mappings ℓ_d and r_d extend to morphisms in the natural manner. They are called desynchronizing morphisms. Now from equation (11), we obtain

$$d\ell_d\big(w^{\mathrm{R}}c\delta_1\delta_2\cdots\delta_k\big)dd = ddr_d\big(\gamma_1\gamma_2\cdots\gamma_k u^{\mathrm{R}}c\big)d\,, \tag{12}$$

where both sides begin and end with a double dd, and elsewhere d is between all pairs of letters from $\{a,b,c\}$. We let

$$W = \{(\ell_d(\gamma), r_d(\delta)) \mid \gamma X \mapsto X\delta \in P_1\} \cup \big\{(d\ell_d(w^{\mathrm{R}}c), dd), (dd, r_d(u^{\mathrm{R}}c)d)\big\} \tag{13}$$

be an instance of the PCP. It is straightforward that if $u \in \mathcal{A}_S$, then the instance W has a solution. We note that the assumption that the normal system S is standard, i.e., the rule words are non-empty, is needed at this point to guarantee that the desynchronization works properly.

What is left is to show is the converse: if W has a solution, then $u \in \mathcal{A}_S$. For this, assume that W has a solution, and choose a minimal solution, i.e. a solution that does not contain any solutions as a proper prefix. It is immediate that if W has a solution, it has a minimal solution.

Obviously, a solution must begin with the pair $(d\ell_d(w^{\mathrm{R}}c), dd)$ as that is the only pair having a common nonempty prefix. Similarly, a solution must end with the pair $(dd, r_d(u^{\mathrm{R}}c)d)$, since that is the only pair in W with a common nonempty suffix. On the other hand, these two special pairs with occurrences of the word dd cannot appear in the middle of any minimal solution as dd can be covered only by these two pair. Therefore, if $i_1, \ldots, i_k$ is a minimal solution to the instance W, then

$$u_{i_1}u_{i_2}\cdots u_{i_k} = v_{i_1}v_{i_2}\cdots v_{i_k} \text{ with } (u_{i_j}, v_{i_j}) \in W \text{ for } j = 1, \ldots, k\,,$$

then $(u_{i_1}, v_{i_1}) = (d\ell_d(w^{\mathrm{R}}c), dd)$, $(u_{i_k}, v_{i_k}) = (dd, r_d(u^{\mathrm{R}}c)d)$ and

$$(u_{i_j}, v_{i_j}) = (\ell_d(\gamma_j), r_d(\delta_j)) \text{ and } \gamma_j X \mapsto X\delta_j \in P_1\,,$$

for $j = 2, \ldots, k-1$. It follows that the minimal solution corresponds to the equation (12) which implies that $u \in \mathcal{A}_s$.

By Proposition 1, we have

Theorem 3. *The PCP is undecidable.*

Recall that $\{a, b, c, d\}^*$ can be embedded into $\{a, b\}^*$ by an injective morphisms. In this way we obtain instances in the binary alphabet as originally considered by Post.

Actually, Post proved the undecidability of a special form of the PCP, called the *generalized PCP* in the literature; see for example [4].

Theorem 4. *It is undecidable for given set of pairs $\{(u_i, v_i) \mid 1 \leq i \leq n\}$ of words whether or not there exist a sequence $i_1, i_2, \ldots, i_k$ such that*

$$u_1 u_{i_1} \cdots u_{i_k k} u_n = v_1 v_{i_1} \cdots v_{i_k} v_n\,. \tag{14}$$

Note that in Theorem 4 the first pair and the last pair of the required solution are fixed in (14) to be (u_1, v_1) and (u_n, v_n), respectively. In W constructed in (13), $(u_1, v_1) = (d\ell_d(w^{\mathrm{R}}c), dd)$ and $(u_n, v_n) = (dd, r_d(u^{\mathrm{R}}c)d)$. Note also that in (14) we could assume that, $i_j \notin \{1, n\}$ for all $j = 1, \ldots, k$.

4 Another proof for the PCP from the normal systems

In this section we give a new proof for the undecidability of the PCP by a reduction to the assertion problem of the normal systems, i.e., we show that if the PCP is decidable, then the assertion problem of the normal systems is also decidable, contradicting Proposition 1. For this, we take an arbitrary (non-trivial) instance of the assertion problem, that is, normal system $S = (w, P)$ and word u with $u \neq w$, and construct an instance $W_{S,u}$ of the PCP such that $W_{S,u}$ has a solution if and only if $u \in \mathcal{A}_S$.

The present proof takes a modern approach of connecting the configurations of a derivation in the normal systems to a solution of the PCP – instead of the rule words used by PCP as was done in the original proof by Post in Section 3.

Let $S = (w, P)$ be a normal system over the binary alphabet $\{a, b\}$ where $P = \{p_1, \dots, p_t\}$ and $p_j = \alpha_j X \mapsto X\beta_j$ for $j = 1, \dots, t$. As Post did, we begin with the sequence (3), but use different indices: we assume that there exists a sequence of equalities

$$w = \alpha_{i_1} x_1,\ x_1\beta_{i_1} = \alpha_{i_2} x_2, \dots, x_{k-1}\beta_{i_{k-1}} = \alpha_{i_k} x_k,\ x_k\beta_{i_k} = u\,, \tag{15}$$

for the input word u where $\alpha_{i_j} X \mapsto X\beta_{i_j} \in P$ for $j = 1, \dots, k$. Instead of the equations (5) and (6), we take

$$w x_1 \beta_{i_1} x_2 \beta_{i_2} \cdots x_k \beta_{i_k} = \alpha_{i_1} x_1 \alpha_{i_2} x_2 \cdots \alpha_{i_k} x_k u, \tag{16}$$

where the configurations in (15) are concatenated – left hand sides on the left and right hand sides on the right.

Let c and f be new letters. We split each rule $p_j \in P$ to two pairs p_j^α and p_j^β as follows:

$$p_j^\alpha = (\ell_d(c^j f), r_d(f\alpha_j)) \quad \text{and} \quad p_j^\beta = (\ell_d(\beta_j), r_d(c^j)),$$

where r_d and ℓ_d are the desynchronizing mappings for the letter d. The word $c^j f$ is a marker word that forces a solution of the (the below) instance of the PCP to choose the pairs jointly. Consider the following instance of the PCP:

$$\begin{aligned} W = &\{(d\ell_d(fw), dd), (dd, r_d(fu)d), (da, ad), (db, bd)\} \\ &\cup \{p_j^\alpha, p_j^\beta \mid j = 1, \dots, t\}. \end{aligned} \tag{17}$$

To see the idea encoded in W, let us first assume that W has a solution. A solution must necessarily begin with the pair $(d\ell_d(fw), dd)$ that we now write in the form

L: $d\ell_d(fw)$
R: dd

In order to produce $r_d(fw)$ to the right hand side, we need to use a pair which has f as a first symbol on the right hand side. As the pair $(dd, r_d(fu)d)$ produces dd to the end of left hand side, which then has to match with dd produced by the right hand side, we must have

$$d\ell_d(fw) \cdot dd = dd \cdot r_d(fu)d,$$

implying $w = u$. In a non-trivial case of the assertion problem, $u \neq w$, for producing $r_d(fw)$ to the right hand side, there must exist a pair $p^{\alpha}_{i_1} = (\ell_d(c^{i_1}f), r_d(f\alpha_{i_1}))$ in W with $w = \alpha_{i_1}x_1$. After this we have

$$\begin{aligned}&\mathrm{L}\colon d\ell_d(fwc^{i_1}f)\\&\mathrm{R}\colon ddr_d(f\alpha_{i_1})\end{aligned}$$

Now the first occurrence of the letter c forces the use of the pairs $(da, ad), (db, bd) \in W$ in order to have x_1 and cover the start word w of the left hand side. So now we have

$$\begin{aligned}&\mathrm{L}\colon d\ell_d(fwc^{i_1}fx_1)\\&\mathrm{R}\colon ddr_d(f\alpha_{i_1}x_1)\end{aligned}$$

Next to match c^{i_1}, we must chose the other half of the rule P_{i_1}, i.e., the pair $p^{\beta}_{i_1} = (\ell_d(\beta_{i_1}), r_d(c^{i_1}))$. We then have

$$\begin{aligned}&\mathrm{L}\colon d\ell_d(fwc^{i_1}fx_1\beta_{i_1})\\&\mathrm{R}\colon ddr_d(f\alpha_{i_1}x_1c^{i_1}).\end{aligned}$$

In other words, after forgetting the synchronizing letters d, the left hand side has the overflow $fx_1\beta_{i_1}$. As above, the occurrence of f forces to chose the rule $p^{\alpha}_{i_2}$, then write x_{i_2} and the other half of the rule p_{i_2}, the pair $p^{\beta}_{i_2}$ etc. Therefore, at some point we must have

$$\begin{aligned}&\mathrm{L}\colon d\ell_d(fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots fx_{t-1}\beta_{i_{t-1}}c^{i_t}fx_t\beta_{i_t})\\&\mathrm{R}\colon ddr_d(f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots f\alpha_{i_t}x_tc^{i_t})\end{aligned} \tag{18}$$

with each w, u, x_{i_j}, α_{i_j} and β_{i_j} satisfying (15) up to index $t \geq 2$. Note that

$$d\ell_d(fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots fx_{t-1}\beta_{i_{t-1}}c^{i_t})d = ddr_d(f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots f\alpha_{i_t}x_tc^{i_t}).$$

A minimal solution in W must end with pair $(dd, r_d(fu)d)$ in order to match the d's in the solution. As $r_d(fu)d$ begin with f and has no c's, and left hand side has one more f than the right hand side in (18), the pair $(dd, r_d(fu)d)$ has to match $fx_t\beta_{i_t}$ in (18). Therefore, at some point $t = k$ in (15) and (16) with $x_k\beta_{i_k} = u$ and

$$\begin{aligned}&d\ell_d(fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots c^{i_k}fx_k\beta_{i_k})dd\\=&ddr_d(f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots\alpha_{i_k}x_kc^{i_k}fu)d.\end{aligned}$$

The other direction is clear: Suppose $u \in \mathcal{A}_S$. Then there exists a sequence of equations (15) satisfying (16). We start with (16), and place symbols f and words c^i accordingly using the equations in (15). Then we get

$$fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots c^{i_k}fx_k\beta_{i_k} = f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots\alpha_{i_k}x_kc^{i_k}fu.$$

Now placing dd to both ends and d between the letters a, b, c, f, we obtain

$$\begin{aligned} &d\ell_d(fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots c^{i_k}fx_k\beta_{i_k})dd \\ &\quad = ddr_d(f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots\alpha_{i_k}x_kc^{i_k}fu)d\,. \end{aligned} \tag{19}$$

What is left is to show that the words in (19) can be build with pairs of W, correspondingly. For this, for a word $x \in \{a,b\}^*$, $x = e_1\cdots e_n$ and $e_i \in \{a,b\}$, denote by $\bar{x}$ the sequence of pairs $(de_1, e_1d), \ldots, (de_n, e_nd)$ from W. The idea is that the sequence $\bar{x}$ writes $\ell_d(x)$ to the left hand side and $r_d(x)$ to the right hand side in a solution. Let Z be the following sequence of pairs of W,

$$(d\ell_d(fw), dd), p^{\alpha}_{i_1}, \bar{x_1}, p^{\beta}_{i_1}, p^{\alpha}_{i_2}, \bar{x_2}, p^{\beta}_{i_2}, \ldots, p^{\alpha}_{i_k}, \bar{x_k}, p^{\beta}_{i_k}, (dd, r_d(fu)d).$$

Now, Z is a solution of the PCP, as

$$\begin{aligned} &d\ell_d(fw)\ell_d(c^{i_1}f)\ell_d(x_1)\ell_d(\beta_{i_1})\ell_d(c^{i_2}f)\cdots\ell_d(c^{i_k}f)\ell_d(x_k)\ell_d(\beta_{i_k})dd \\ &= d\ell_d(fwc^{i_1}fx_1\beta_{i_1}c^{i_2}f\cdots c^{i_k}fx_k\beta_{i_k})dd \\ &= ddr_d(f\alpha_{i_1}x_1c^{i_1}f\alpha_{i_2}x_2c^{i_2}f\cdots\alpha_{i_k}x_kc^{i_k}fu)d \\ &= ddr_d(f\alpha_{i_1})r_d(x_1)r_d(c^{i_1})r_d(f\alpha_{i_2})r_d(x_2)\cdots r_d(f\alpha_{i_k})r_d(x_k)r_d(c^{i_k})r_d(fu)d\,, \end{aligned} \tag{20}$$

where the first and the last words are the left hand and the right hand sides (respectively) of words in pairs in Z catenated correspondingly. This implies the existence of a solution of the PCP for the set W when $u \in \mathcal{A}_S$. Therefore, the PCP is undecidable.

5 Conclusion

A shorter and bit simpler proof for the undecidability of the PCP was given using the same source of undecidability, the Post normal systems, as in the original proof by Post. We are in no doubt that the present proof could have been found by Emil Post as well, but as a true pioneer of the field of computability he immediately would have noticed the following deficiency of the construction: when considering the size of an instance as constructed in the proof, Post's original construction gives an instance of size $|P| + 5$, but our new construction gives an instance of size $2|P| + 4$. As the undecidable problem in the normal system, the cardinality of P must be at least two, we realize that Post's proof gives a better bound for the undecidability.

Acknowledgement

The authors thank the anonymous referees whose comments helped to improve the presentation of the article.

References

[1] Church, Alonzo. Review of [6]. *The Journal of Symbolic Logic*, 8(1):50–52, 1943. DOI: 10.2307/2268006.

[2] Claus, V. Some remarks on PCP(k) and related problems. *Bull. EATCS*, 12:54–61, 1980.

[3] Halava, Vesa. Another proof of undecidability for the correspondence decision problem - Had I been Emil Post. *CoRR*, abs/1411.5197, 2014.

[4] Harju, Tero and Karhumäki, Juhani. *Morphisms*, pages 439–510. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997. DOI: 10.1007/978-3-642-59136-5_7.

[5] Neary, Turlough. Undecidability in binary tag systems and the Post correspondence problem for five pairs of words. In *32nd International Symposium on Theoretical Aspects of Computer Science*, volume 30 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages 649–661. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2015.

[6] Post, Emil Leon. Formal reductions of the general combinatorial decision problem. *Amer. J. Math.*, 65:197–215, 1943. DOI: 10.2307/2371809.

[7] Post, Emil Leon. Recursively enumerable sets of positive integers and their decision problems. *Bull. Amer. Math. Soc.*, 50:284–316, 1944. DOI: 10.1090/S0002-9904-1944-08111-1.

[8] Post, Emil Leon. A variant of a recursively unsolvable problem. *Bull. Amer. Math. Soc.*, 52:264–268, 1946. DOI: 10.1090/S0002-9904-1946-08555-9.

[9] Sipser, Michael. *Introduction to the theory of computation. 3rd. ed.* Cengage Learning, 3rd. ed. edition, 2013.

Received 23rd October 2019

Acta Cybernetica **24** (2020) 625–641.

Estimating the Dimension of the Subfield Subcodes of Hermitian Codes

Sabira El Khalfaoui[a] and Gábor P. Nagy[b]

Abstract

In this paper, we study the behavior of the true dimension of the subfield subcodes of Hermitian codes. Our motivation is to use these classes of linear codes to improve the parameters of the McEliece cryptosystem, such as key size and security level. The McEliece scheme is one of the promising alternative cryptographic schemes to the current public key schemes since in the last four decades, they resisted all known quantum computing attacks. By computing and analyzing a data collection of true dimensions of subfield subcodes, we concluded that they can be estimated by the extreme value distribution function.

Keywords: AG code, Hermitian code, subfield subcode, extreme value distribution

1 Introduction

Recently, there has been a big amount of research addressed to quantum computers that use quantum mechanical techniques to solve hard computational problems in mathematics [2]. The existence of these powerful machines threaten many of the public-key cryptosystem that are widely in use. Combined with Shor's algorithms [38], this would risk the confidentiality and integrity of today's digital communications. Post-quantum cryptography aims to construct and develop cryptosystems that resist against quantum computing attacks.

McEliece [28] introduced the first code-based public-key cryptosystem in 1978, where he employed error correcting codes to generate the public and private key with security relying on two aspects: NP-completeness of decoding linear codes and the distinguishing of the chosen codes. The original McEliece scheme was constructed with binary Goppa codes which are subfield subcodes of generalized

[a] Bolyai Institute, University of Szeged, Aradi vértanúk tere 1, H-6720 Szeged, Hungary. E-mail: `sabira@math.u-szeged.hu`, ORCID: `https://orcid.org/0000-0002-1792-2947`.

[b] Department of Algebra, Budapest University of Technology and Economics, Egry József utca 1, H-1111 Budapest, Hungary and Bolyai Institute, University of Szeged, Aradi vértanúk tere 1, H-6720 Szeged, Hungary. E-mail: `nagyg@math.u-szeged.hu`, ORCID: `https://orcid.org/0000-0002-9558-4197`.

DOI: 10.14232/actacyb.285453

Reed-Solomon codes. Even today, this proposal represents a good candidate for post-quantum cryptography [1]. There have been several attempts to find appropriate classes of codes and their parameters, which give rise to a secure and effective cryptosystem, for more details see [31, 27]. In this paper, we study the possibility of the application of subfield subcodes of Hermitian codes in the McEliece scheme. More precisely, we do the first step by investigating the true dimension of these codes for a broad spectrum of parameters, for partial results see [13, 34]. Our main observation is that the true dimension of subfield subcodes of Hermitian codes can be estimated by the extreme value distribution function.

In the literature, several attacks have been proposed against McEliece cryptosystem in general, and against McEliece systems based on AG codes, see [3, 27, 6]. Attacks can be divided into two classes: structural, or key recovery attacks, aimed at recovering the secret code, and decoding, or message recovery attacks, aimed at decrypting the transmitted ciphertext. The generic decoding attack against the McEliece scheme is the information set decoding (ISD) algorithm. The most recent and most effective structural attack against AG code based McEliece systems is the Schur product distinguisher.

The structure of this paper is as follows. In section 2, we review the necessary background to define subfield subcodes, algebraic geometry codes and Hermitian codes. In section 3, we introduce some tools borrowed from statistics in order to handle our computed data on the true dimension of subfield subcodes of Hermitian codes, the latter being presented in section 4. Our main result is Proposition 1 in section 5 which shows the excellent fitting properties of the extreme value distribution to our measurements. In section 6, we applied this estimate to study the development of the key size of Hermitian subfield subcodes.

2 Backgrounds, formulas

In this section, we give an overview on subfield subcodes, AG codes and some of their properties, for more details the reader is refereed to the monographs [17, 40, 41]. Our terminology on coding theory is standard, see [40, 18]. In particular, by an $\mathbb{F}_q$-linear code of length n, we mean a linear subspace of $\mathbb{F}_q^n$.

2.1 Subfield subcodes

Let h be a positive integer and r, q be prime powers with $q = r^h$. Then $\mathbb{F}_r$ is a subfield of $\mathbb{F}_q$ and the field extension $\mathbb{F}_q/\mathbb{F}_r$ has degree h. Let C be an $\mathbb{F}_q$-linear code of length n and dimension k. The $\mathbb{F}_q/\mathbb{F}_r$ subfield subcode of C is defined by

$$C|_{\mathbb{F}_r} = C \cap \mathbb{F}_r^n.$$

The trace polynomial $\mathrm{Tr}(x) = x + x^r + \cdots + x^{r^{h-1}}$ defines a map $\mathbb{F}_q \to \mathbb{F}_r$, which can be extended to a map $\mathbb{F}_q^n \to \mathbb{F}_r^n$ component wise. The trace code of the linear code C is

$$\mathrm{Tr}(C) = \{\mathrm{Tr}(c) \mid c \in C\}.$$

Clearly, both the subfield subcode and the trace code are $\mathbb{F}_r$-linear codes of length n. However, it is in general very hard to determine the true dimension of these new codes. The fascinating result given by Delsarte [8] in 1975 plays a key role for studying the class of the subfield subcodes of linear codes. It established a closed link between subfield subcodes and trace codes:

$$(C|_{\mathbb{F}_r})^{\perp} = \mathrm{Tr}(C^{\perp}).$$

Véron [44] used this equation to give the exact dimension formula

$$\dim_{\mathbb{F}_r}(C|_{\mathbb{F}_r}) = n - h(n-k) + \dim_{\mathbb{F}_r} \ker(\mathrm{Tr}). \tag{1}$$

In particular, we have the trace bound

$$\dim_{\mathbb{F}_r}(C|_{\mathbb{F}_r}) \geq n - h(n-k). \tag{2}$$

2.2 Algebraic geometry codes

In this section, we give an overview on the construction of algebraic geometry (AG) codes, which is a version of V.D. Goppa's original construction. We note that there are many ways to produce linear codes from algebraic curves. Also we give some details on the properties, parameters and duality of AG codes. AG codes are linear codes that use algebraic curves and finite fields for their construction. The construction can be done by evaluating functions (elements of the function field) or by computing residues of differentials. Our notation and terminology on algebraic plane curves over finite fields, their function fields, divisors and Riemann-Roch spaces are standard, see for instance [17, 29, 41].

Let q be a prime power and $\mathbb{F}_q$ be the finite field of order q. Let $\mathcal{X}$ be an algebraic curve i.e. an affine or projective variety of dimension one, which is absolutely irreducible and nonsingular and whose defining equations are (homogeneous) polynomials with coefficients in $\mathbb{F}_q$. Let g be the genus of $\mathcal{X}$ and denote by $\mathbb{F}_q(\mathcal{X})$ the function field of $\mathcal{X}$. For a divisor of D of $\mathbb{F}_q(\mathcal{X})$, the Riemann-Roch space is

$$\mathscr{L}(D) = \{f \in \mathbb{F}_q(\mathcal{X}) \mid (f) \succcurlyeq -D\} \cup \{0\},$$

where (f) is the principal divisor of f. The dimension $\ell(D)$ of $\mathscr{L}(D)$ is given by the Riemann-Roch Theorem [41]*Theorem 1.1.15:

$$\ell(D) = \ell(W - D) + \deg D - g + 1, \tag{3}$$

where W is a canonical divisor of $\mathbb{F}_q(\mathcal{X})$.

Let G and D be two divisors of $\mathbb{F}_q(\mathcal{X})$ such that $D = P_1 + \cdots + P_n$ is the sum of n distinct rational places of $\mathbb{F}_q(\mathcal{X})$ and $P_i \notin \mathrm{supp}(G)$ for any i. With these data, two types of algebraic geometry codes can be constructed:

$$\begin{aligned} C_L(D,G) &= \{(f(P_1), \cdots, f(P_n)) \mid f \in \mathscr{L}(G)\}, \\ C_\Omega(D,G) &= \{(res_{P_1}(\omega), \cdots, res_{P_n}(\omega)) \mid \omega \in \Omega(G-D)\}. \end{aligned}$$

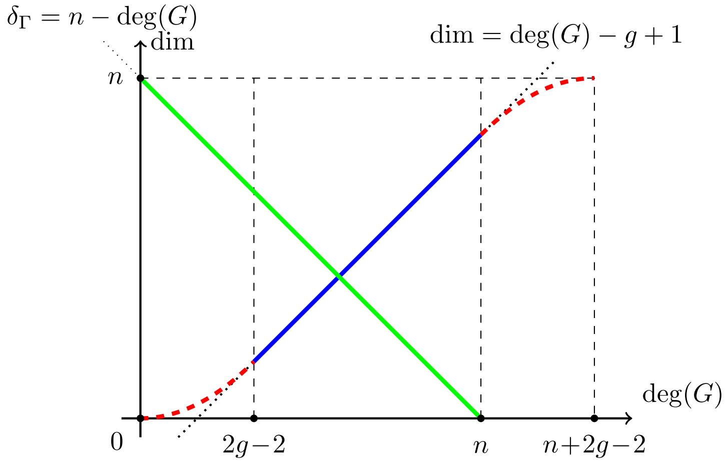


Figure 1: Dimension and designed minimum distance of AG codes

The codes $C_L(D,G)$ and $C_\Omega(D,G)$ are called the *functional* and the *differential codes,* respectively. These two codes are dual to each other. Moreover, the differential code $C_\Omega(D,G)$ is equivalent with the functional code $C_L(D,W+D-G)$. In particular, they have the same dimension and minimum distance, even though this equivalence does not preserve all important properties of the code. The formula

$$k = \ell(G) - \ell(G-D)$$

for the dimension k of $C_L(D,G)$ follows from the Riemann-Roch Theorem, which also provides a lower bound $\delta_\Gamma = n - \deg(G)$ for its minimum distance. The integer δ_Γ is called the *Goppa designed minimum distance* of the AG code.

We illustrate the behavior of the dimension k of $C_L(D,G)$ depending on the degree of the divisor G by Figure 1. In fact, (3) implies the exact value $k = \deg(G) - g + 1$ provided $2g-2 < \deg(G) < n$. Furthermore, if $\deg(G) > n + 2g - 2$, then $k = n$. In the intervals $[0, 2g-2]$, and $[n, n+2g-2]$, the dimension depends on the specific structure of the divisor G.

2.3 On the decoding of AG codes

Algebraic geometry codes are a generalization of Reed-Solomon codes, then it is not extraordinary that they benefit from similar decoding algorithms. The work on the decoding of AG codes seems to begin in 1986 when Driencourt gave a first decoding algorithm for codes on elliptic curves of characteristic 2 [9] correcting $\lfloor(\delta_\Gamma - 1)/2\rfloor$ errors. By generalizing the work of Arimoto and Peterson [33] on employing a locator polynomial to decode Reed-Solomon codes, Justesen, Larsen, Jensen, Havemose and Høhold published [21] in 1989 a decoding algorithm for a larger class of AG codes, which can correct up to $\lfloor(\delta_\Gamma - g - 1)/2\rfloor$ errors, moreover in improved version [20] the error capability is increased to $\lfloor (\delta_\Gamma - g/2 - 1)/2 \rfloor$. This method was generalized to arbitrary curves by Skorobogatov and Vladut [39], and independently by Krachkovskii [26], then extended by Duursma [10, 12] to correct $\lfloor (\delta_\Gamma - 1)/2 \rfloor - \sigma$ errors, where σ is the Clifford defect of the curve [12]*Definition 3.7 (is approximately $g/4$). In 1993, Feng and Rao [15] gave a majority voting scheme allowing a decoding up to $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors. Duursma generalized this result to all AG codes [11]. An efficient algorithm was described by Sakata, Justesen, Madelung, Jensen and Høhold in [35] using a multidimensional generalization of Massey-Berlekamp algorithm done by Sakata [36]. Kirfel and Pellikaan [22] noticed that one can decode beyond $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors for 1–point AG codes by studying the Weierstrass semigroup. The reader can refer to [18, 19, 32] for more details on decoding methods.

2.4 Hermitian codes

The classes of AG codes we study in this paper are defined over the Hermitian curve [41]*VI.3.6 and VI.4.3. Let $\mathbb{F}_q$ be a finite field and define the Hermitian curve $\mathscr{H}_q$ by the affine equation $Y^q + Y = X^{q+1}$. Notice that $\mathscr{H}_q$ is defined over $\mathbb{F}_{q^2}$, that is, its rational points are points of the projective plane $PG(2, q^2)$, satisfying the homogeneous equation $Y^q Z + Y Z^q = X^{q+1}$. With respect to the line $Z = 0$ at infinity, $\mathscr{H}_q$ has one infinite point $P_\infty = (0 : 1 : 0)$ and q^3 affine rational points $P_1, \dots, P_{q^3}$. As usual, we also look at the curve $\mathscr{H}_q$ as the smooth curve defined over the algebraic closure $\bar{\mathbb{F}}_{q^2}$. Then, there is a one-to-one correspondence between the points of $\mathscr{H}_q$ and the places of the function field $\bar{\mathbb{F}}_{q^2}(\mathscr{H}_q)$ of $\mathscr{H}_q$.

With a Hermitian code we mean a functional AG code of the form $C_L(D, G)$, where the divisor D is defined as the sum $P_1 + \cdots + P_{q^3}$ affine rational points of $\mathscr{H}_q$. In our investigations, the divisor G can take two forms. In the *1-point case,* we set $G = sP_\infty$ with integer s. In the *degree 3 case,* we put $G = sP$, where P is a place of degree 3. Let P_1, P_2, P_3 be the extensions of P in the constant field extension of $\mathbb{F}_{q^2}(\mathscr{H}_q)$ of degree 3. Then P_1, P_2, P_3 are degree one places of $\mathbb{F}_{q^6}(\mathscr{H}_q)$ and, up to labeling the indices, $P_{j+1} = \mathrm{Frob}(P_j)$ where Frob is the q^2-th Frobenius map and the indices are taken modulo 3. Also, P may be identified with the $\mathbb{F}_{q^2}$-rational divisor $P_1 + P_2 + P_3$ of $\mathbb{F}_{q^6}(\mathscr{H}_q)$. Functional AG codes of the form $C_L(D, sP_\infty)$ and $C_L(D, sP)$ will be called 1-point Hermitian codes, and Hermitian codes over a degree 3 place, respectively. In the 1-point case, the basis of the Riemann-Roch space $\mathscr{L}(sP_\infty)$ can be given explicitly by [40]:

$$\mathcal{M}(s) := \left\{ x^i y^j \mid 0 \le i \le q^2 - 1, 0 \le j \le q - 1, qi + (q+1)j \le s \right\}.$$

In the degree 3 case, the basis of

$$\mathscr{L}(sP) = \left\{ \frac{f}{(\ell_1 \ell_2 \ell_3)^u} \mid f \in \mathbb{F}_{q^2}[X, Y], \deg f \le 3u, v_{P_i}(f) \ge v \right\} \cup \{0\}.$$

can be computed, see [24]. In this formula, $\ell_i = 0$ is the equation of the tangent line of $\mathscr{H}_q$ at P_i, and $s = u(q+1) - v$, $0 \le v \le q$.

The group $\mathrm{Aut}(\mathscr{H}_q)$ of all automorphisms of $\mathscr{H}_q$ is defined over $\mathbb{F}_{q^2}$. It is a group of projective linear transformations of $PG(2, q^2)$, isomorphic to the projective unitary group $PGU(3, q)$. Furthermore, $\mathrm{Aut}(\mathscr{H}_q)$ acts doubly transitively on the set $\{P_\infty, P_1, \ldots, P_{q^3}\}$ of $\mathbb{F}_{q^2}$-rational points. As it was pointed out in [24], the automorphism group of $\mathscr{H}_q$ acts transitively on the set of degree 3 places of $\mathbb{F}_{q^2}(\mathscr{H}_q)$, as well. Hence, the geometry of a degree 3 place is independent on the choice of P. However, the stabilizer G_P of P in $\mathrm{Aut}(\mathscr{H}_q)$ is not transitive on the set of $q^3 + 1$ rational points. In fact, G_P is a cyclic group of order $q^2 - q + 1$ and the number of G_P-orbits on the set of rational points is $q + 1$. (See [5, 24], where [5]*Section 4.2 holds for any characteristic.)

3 Moments of the extended rate of subfield subcodes

In order to make our notation consistent, we make the following conventions. Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_q$ and D, G effective divisors such that the AG code $C_L(D, G)$ is well defined. Assume that the objects δ and γ determine the curve $\mathcal{X}$ and the divisors D, G in a unique way. Let s be an integer and $\mathbb{F}_r$ be a subfield of $\mathbb{F}_q$. Then,

$$C^\gamma_{\delta,r}(s) = C_L(D, sG)|_{\mathbb{F}_r}$$

denotes the $\mathbb{F}_q/\mathbb{F}_r$ subfield subcode of the AG code $C_L(D, sG)$. The length of $C^\gamma_{\delta,r}(s)$ is $n = \deg(D)$.

For the integer s, let

$$R(s) = R^\gamma_{\delta,r}(s) = \frac{\dim_{\mathbb{F}_r} C^\gamma_{\delta,r}(s)}{n}$$

denote the rate of the subfield subcode $C^\gamma_{\delta,r}(s)$. We extend $R^\gamma_{\delta,r}$ to $\mathbb{R}$ in the usual way: $R^\gamma_{\delta,r}(x) = R^\gamma_{\delta,r}(\lfloor x \rfloor)$.

Lemma 1. *Let g be the genus of $\mathcal{X}$ and define*

$$\alpha = \left\lceil \frac{n + 2g - 2}{\deg(G)} \right\rceil .$$

Then $R(x) = R^\gamma_{\delta,r}(\lfloor x \rfloor)$ is a monotone increasing function, with

$$R(x) = \begin{cases} 0 & \text{for } x < 0, \\ 1 & \text{for } x \geq \alpha. \end{cases}$$

Proof. If $s \deg(G) > n + 2g - 2$, then $\deg(D + W - G) < 0$, and

$$C_\Omega(D, G) \cong C_L(D, D + W - G) = \{0\}.$$

Hence, if $s \geq \alpha$, then $C_L(D, sG) = \mathbb{F}_q^n$ and $C_L(D, sG)|_{\mathbb{F}_r} = \mathbb{F}_r^n$. □

The following observation has been made in [13]*Theorem 5.1 for the special case of a one point divisor of a Hermitian curve.

Lemma 2. *For $0 \leq x < n/(r\deg(G))$, we have $R(x) = 1/n$.*

Proof. Let s be an integer with $0 \leq s < \frac{n}{r\deg(G)}$. As the divisor sG is positive for $s > 0$, the constant vectors are in $C_L(D, sG)|_{\mathbb{F}_r}$ and $R(s) \geq 1/n$ holds. Assume $R(s) > 1/n$, that is, the subfield subcode contains a non constant element $\boldsymbol{v} = (f(P_1), \ldots, f(P_n))$ with $f \in \mathscr{L}(sG)$. Since a function of the form $f+c$ cannot have more than $\deg(sG)$ zeros, $\boldsymbol{v}$ cannot have the same entry more than $s\deg(G)$ times. This implies $r\deg(sG) \geq n$. □

Lemma 1 implies that we can consider $R(x)$ as the distribution function of some random variable ξ, cf. [37]*Definition 1, Section 2.3.

Lemma 3. *Let $R(x)$ be the extended rate function of a class of subfield subcodes $C_L(D, sG)|_{\mathbb{F}_r}$. Define the integer α as in Lemma 1. Let ξ be a random variable with distribution function $R(x)$. Then*

$$\mathsf{E}(\xi) = \sum_{s=0}^{\alpha} 1 - R(s), \qquad \mathsf{E}(\xi^2) = \sum_{s=0}^{\alpha} (2s+1)(1 - R(s)).$$

Proof. This follows from [37]*Section 2.6, Corollary 2. □

Remark 1. Considered as a distribution function, $R^{\gamma}_{\delta,r}(s)$ has an expectation $\mathsf{E}^{\gamma}_{\delta,r}$, a variance $\mathsf{Var}^{\gamma}_{\delta,r}$ and a standard deviation $\mathsf{D}^{\gamma}_{\delta,r}$. These constants can be computed from the true dimensions of the subfield subcodes using Lemma 3 and the well known formulas for random variables.

4 Computed true dimensions of Hermitian subfield subcodes

Let q be a prime power. We say that the object $\delta = q$ determines the Hermitian curve $\mathscr{H}_q$ over $\mathbb{F}_{q^2}$, together with the divisor D which is the sum of affine rational points of $\mathscr{H}_q$. The objects $\gamma = \text{1-pt}$ or $\gamma = \text{deg-3}$ determine the divisor G to be equal either to the rational infinite place P_∞, or the degree 3 Hermitian place P, respectively. That being said, for any integer s and subfield $\mathbb{F}_r$ of $\mathbb{F}_{q^2}$, the Hermitian subfield subcodes

$$C^{\text{1-pt}}_{q,r}(s) = C_L(D, sP_\infty)|_{\mathbb{F}_r}, \qquad C^{\text{deg-3}}_{q,r}(s) = C_L(D, sP)|_{\mathbb{F}_r}$$

are well defined and consistent with the notation of section 3. These codes are $\mathbb{F}_r$-linear codes of length $n = q^3$.

Let $R^{\text{1-pt}}_{q,r}(s)$ and $R^{\text{deg-3}}_{q,r}(s)$ be the true rates of the codes $C^{\text{1-pt}}_{q,r}(s)$ and $C^{\text{deg-3}}_{q,r}(s)$. Using the GAP [16] package `HERmitian` [30], we have been able to compute the true dimension values of the codes $C^{\text{1-pt}}_{q,q}(s)$, $C^{\text{deg-3}}_{q,q}(s)$ for

$$q \in \{2, 3, 4, 5, 7, 8, 9, 11, 13\}$$

and the binary codes $C^{\text{1-pt}}_{q,2}(s)$, $C^{\text{deg-3}}_{q,2}(s)$ for

$$q \in \{2, 4, 8, 16\}.$$

(Cf. [13] for preliminary results on explicit computation of subfield subcodes of Hermitian 1-point codes.)

As given in Lemma 3, we computed the expectations $\mathsf{E}^{\text{1-pt}}_{q,q}$, $\mathsf{E}^{\text{1-pt}}_{q,2}$, $\mathsf{E}^{\text{deg-3}}_{q,q}$, $\mathsf{E}^{\text{deg-3}}_{q,2}$, the variances $\mathsf{Var}^{\text{1-pt}}_{q,q}$, $\mathsf{Var}^{\text{1-pt}}_{q,2}$, $\mathsf{Var}^{\text{deg-3}}_{q,q}$, $\mathsf{Var}^{\text{deg-3}}_{q,2}$, and the standard deviations $\mathsf{D}^{\textbf{1-pt}}_{q,r}$, $\mathsf{D}^{\text{1-pt}}_{q,2}$, $\mathsf{D}^{\text{deg-3}}_{q,q}$, $\mathsf{D}^{\text{deg-3}}_{q,2}$ for these true rates. The numerical results are shown in Table 1 for $q = 3, 4, 5, 7, 8, 9, 11, 13$ and $r = q$, and in Table 2 for $q = 2, 4, 8, 16$ and $r = 2$. In Figure 2, we present the ratios $\mathsf{E}^{\gamma}_{q,r} \deg(G)/n$ and $\mathsf{D}^{\gamma}_{q,r} \deg(G)/n$, where $\gamma \in \{\text{1-pt}, \text{deg-3}\}$. While our data sets are small, these figures motivate the following open problem.

Problem 1. *Are there constants* $c_1, c_2 > 0$ *such that*

$$\mathsf{E}^{\textit{1-pt}}_{q,q} \approx \mathsf{E}^{\textit{deg-3}}_{q,q} \approx c_1 q^3 / \deg(G), \qquad \mathsf{D}^{\textit{1-pt}}_{q,q} \approx \mathsf{D}^{\textit{deg-3}}_{q,q} \approx c_2 q^3 / \deg(G),$$

where $a \approx b$ *means* $a/b \to 1$ *with* $q \to \infty$*.*

Remark 2. Our data suggests that for small q, the choice $c_1 = 0.75$ and $c_2 = 0.2$ is sound.

Table 1: Expectations and variances for Hermitian $\mathbb{F}_{q^2}/\mathbb{F}_q$ subfield subcodes

q	1-point codes		Codes over a degree 3 place	
	Expectation	Variance	Expectation	Variance
3	20.15	53.46	7.63	4.09
4	48.66	246.79	17.77	16.02
5	95.04	841.16	33.37	60.18
7	259.10	5 553.32	88.99	503.78
8	385.49	11 862.84	131.61	1 106.63
9	546.30	23 541.65	186.22	2 206.21
11	992.73	74 679.83	336.49	7 262.13
13	1 631.29	197 675.07	550.94	19 807.94

Table 2: Expectations and variances for Hermitian $\mathbb{F}_{q^2}/\mathbb{F}_2$ subfield subcodes

q	1-point codes		Codes over a degree 3 place	
	Expectation	Variance	Expectation	Variance
2	5.38	6.48	2.12	0.86
4	54.86	164.96	20.38	10.52
8	458.22	4 838.52	162.50	216.32
16	3 698.92	195 390.48	1 303.40	6 029.44

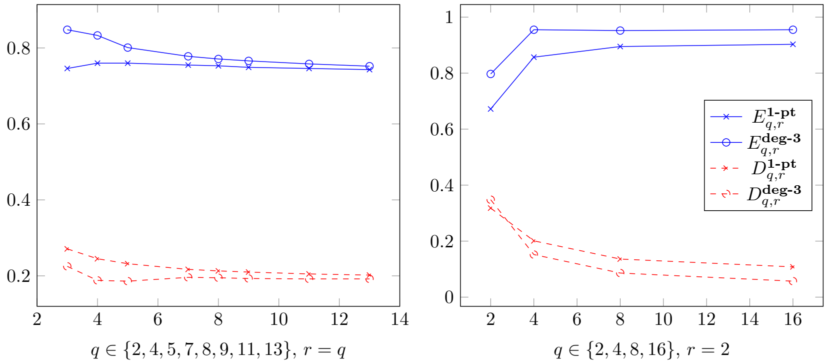


Figure 2: The ratios of expectations and standard deviations to $n/\deg(G)$

5 Distribution fitting

In general, no explicit formula is known for the true dimension of subfield subcodes of AG codes. Our goal was to use the method of distribution fitting in order to study the behavior of these true dimensions in the case of subfield subcodes of Hermitian codes.

As in the previous sections, we used the notation $\mathscr{H}_q$ for the Hermitian curve over $\mathbb{F}_{q^2}$, P_∞, P for the places of degree 1 and 3, D and $G \in \{P_\infty, P\}$ for the divisors, and $C^\gamma_{q,r}(s)$, $\gamma \in \{\text{1-pt, deg-3}\}$, for the $\mathbb{F}_{q^2}/\mathbb{F}_r$ subfield subcodes $C_L(D, sG)|_{\mathbb{F}_r}$. Then, with fixed q, r and $\gamma \in \{\text{1-pt, deg-3}\}$ the dimensions of the subfield subcodes are given by the extended rate function $R^\gamma_{q,r}(x)$.

$$R^{\text{1-pt}}_{q,q}(x), \quad R^{\text{1-pt}}_{q,2}(x), \quad R^{\text{deg-3}}_{q,q}(x), \quad R^{\text{deg-3}}_{q,2}(x).$$

Our goal was to consider these functions as distribution functions and fit some well known probability distribution functions to our experimental rate function $R(x)$.

We obtained numerical results by using the distribution fitting methods offered by MATLAB's Statistics and Machine Learning Toolbox [43]. The technique MLE (Maximum Likelihood Estimation) is a method for estimating the parameters of a probability distribution from a data set. The method finds the parameter values maximizing the logarithm of the likelihood function [14]. In order to compare different distributions for a given data set, one can use the log-likelihood values for a ranking. This is implemented MATLAB's `fitmethis` function [7]. Notice that `fitmethis` also computes the AIC value for each distribution, which stands for Akaike Information Criterion, that measures the quality of a model (distribution) versus the other models. It has the formula

$$AIC = 2l - 2\log(\hat{L})$$

where l is the number of parameters and $\hat{L}$ is the maximum values of the likelihood function. In the case of AIC, smaller values correspond to better fitting distributions (see [23]).

In our comparisons, we restricted ourselves to parametric distributions having at most two parameters, that is, we used MATLAB's `fitmethis` to compare the log-likelihood values of the following distributions: normal, exponential, gamma, logistic, uniform, extreme value, Rayleigh, beta, Nakagami, Rician, inverse Gaussian, Birnbaum-Saunders, log-logistic, log-normal and Weibull. We can summarize the results as follows:

Proposition 1. *1. The best fitting distribution was the extreme value distribution for $R^{\textit{1-pt}}_{q,q}(x)$, $q \in \{4,5,7,8,9,11,13\}$, for $R^{\textit{deg-3}}_{q,q}(x)$, $q \in \{7,8,9,11,13\}$, and for $R^{\textit{1-pt}}_{8,2}(x)$, $R^{\textit{1-pt}}_{16,2}(x)$, $R^{\textit{deg-3}}_{4,2}(x)$, $R^{\textit{deg-3}}_{8,2}(x)$, and $R^{\textit{deg-3}}_{16,2}(x)$.*

2. For the missing cases $R^{\textit{1-pt}}_{2,2}(x)$, $R^{\textit{1-pt}}_{3,3}(x)$, $R^{\textit{deg-3}}_{2,2}(x)$, $R^{\textit{deg-3}}_{3,3}(x)$, $R^{\textit{deg-3}}_{4,4}(x)$, and $R^{\textit{deg-3}}_{5,5}(x)$, the best fitting distribution was the gamma distribution.

3. The second best fitting distribution was the extreme value distribution for $R^{\textit{1-pt}}_{3,3}(x)$, $R^{\textit{deg-3}}_{3,3}(x)$, $R^{\textit{deg-3}}_{4,4}(x)$, $R^{\textit{deg-3}}_{5,5}(x)$.

Our results show that for $q \geq 3$, among the two-parameter distributions considered, the extreme value distribution function is a good estimation of the rate function of subfield subcodes of Hermitian codes.

The extreme value distribution is also referred to as Gumbel or type 1 Fisher-Tippet distribution. In probability theory, these are the limiting distributions of the minimum of a large number of unbounded identically distributed random variables. The extreme value distribution function is

$$F(x;\alpha,\beta) = 1 - \exp\left(-\exp\left(\frac{x-\alpha}{\beta}\right)\right),$$

with location parameter $\alpha \in \mathbb{R}$ and a scale parameter $\beta > 0$. The mean μ and the variance σ^2 are

$$\mu = \alpha - \beta\gamma, \qquad \sigma^2 = \frac{\pi^2}{6}\beta^2,$$

where

$$\gamma = \int_1^\infty \left(-\frac{1}{x} + \frac{1}{\lfloor x \rfloor}\right) dx \approx 0.57721566490153$$

is the Euler-Mascheroni constant, see [25]*Section 1.4. With given empirical mean and variance of the data series, the parameters can be computed by

$$\alpha = \mu + \frac{\sqrt{6}\gamma}{\pi}\sigma, \qquad \beta = \frac{\sqrt{6}}{\pi}\sigma.$$

In Figure 3 we visualized the fitting of the extreme value distribution function to our experimental results on the true dimension of subfield subcodes of Hermitian codes.

The occurrence of the extreme value distribution in the context of subfield subcodes of AG codes may be somewhat surprising and we cannot give a simple mathematical explanation for this. However, the rank of random matrices over finite fields is known to be related to the class of Gumbel type distributions, see Cooper's result [4]*Theorem 2 for the theoretical background. This theory has been applied to parameter estimates of random erasure codes by Studholme and Blake [42].

6 Application: Estimating the key size of McEliece Cryptosystem

The largest (but not the only) part of the public key of the McEliece cryptosystem is the matrix A which defines the underlying error correction code. A is either the $n \times k$ generator matrix, or the $n \times (n-k)$ parity check matrix. In either case, A may be assumed to be in standard form, which means that the public key is given by $k(n-k)$ elements of $\mathbb{F}_r$. Hence, the key size is

$$\log_2(r)k(n-k).$$

Hence, for a fixed field $\mathbb{F}_r$ and length n, the key size is propotional to $R(1-R)$, see [31]. The true values of $R^{\gamma}_{q,r}(s)(1 - R^{\gamma}_{q,r}(s))$ can be estimated by $F(x)(1 - F(x))$, where $F(x)$ is the extreme value distribution function, see Figure 4.

7 Conclusion and future work

The main goal of this study was to establish an approximating formula of the true dimension of the subfield subcodes of Hermitian codes. We conducted an experimental study to analyze the datasets of the true dimension of the $\mathbb{F}_r$-linear codes $C^{1-pt}_{q,r}(s)$, $C^{deg-3}_{q,r}(s)$ for $q \in \{2, 3, 4, 5, 7, 8, 9, 11, 13, 16\}$, $r = 2$ or $r = q$, and s is an integer parameter running from 0 to $q^3 + (q+1)(q-2)$. This analysis helped us to derive new properties of their structure and led to an approach that might be useful for further research and applications.

Theoretically, the main contribution of this work is a collection of formulas of statistical flavour, such as moments of the extended rate function for subcodes of Hermitian codes.

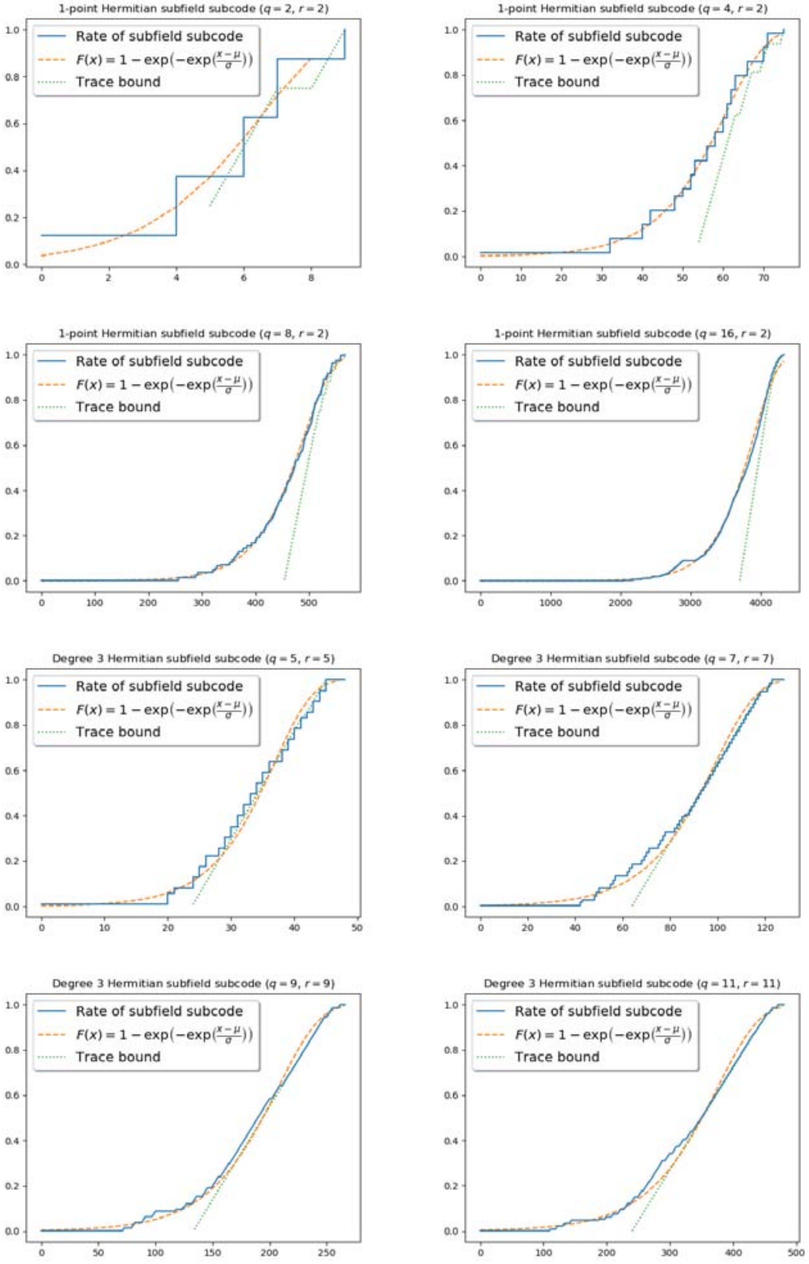


Figure 3: Estimating the extended rate function by extreme value distribution for subfield subcodes Hermitian codes

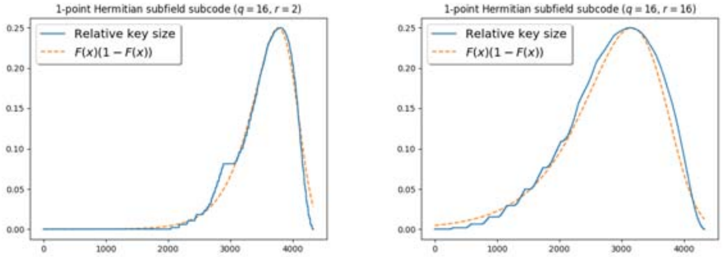


Figure 4: Estimating the key size $n^2R(1 - R)$

From a statistical perspective, the main result is the comparison of the fitting of our datasets of true dimensions to well known distribution functions of MATLAB's Statistics and Machine Learning Toolbox, using the method of `fitmethis`.

We found that the extreme value distribution is the best fitting one for $q > 5$ and the second best fitting distribution for smaller values of q. Also the gamma and the normal distributions have good fitting properties. Our proposal is to use the extreme value distribution function to estimate the true dimension of subfield subcodes of Hermitian codes. In the last section of this paper, we applied this formula to give an approximation for the key size of the McEliece scheme, depending on the parameter s.

In the future, we aim to replace Goppa codes in McEliece's original version with a family of codes that permit to reduce the public key size and to increase the code rate by maintaining a given level of security. Therefore, we intend to analyze the McEliece cryptosystem based on subclasses of subfield subcodes of Hermitian codes. Our future work will include experiments, simulations, and security and cryptanalysis of the McEliece scheme in terms of its public key size and other parameters. The measurements are based on attacks with supposed lowest complexity, e.g. information set decoding or the Schur product distinguisher.

Acknowledgment

The presented work was carried out within the project "Security Enhancing Technologies for the Internet of Things" 2018-1.2.1-NKP-2018-00004, supported by the National Research, Development and Innovation Fund of Hungary, financed under the 2018-1.2.1-NKP funding scheme. Partially supported by NKFIH-OTKA Grants 119687 and 115288.

The authors would like to thank Levente Buttyán (Budapest University of Technology, Hungary) for motivating discussions and Mátyás Barczy (University of Szeged, Hungary) for his help to deal successfully with the concepts from probability theory and statistics.

References

[1] Alagic, Gorjan, Alperin-Sheriff, Jacob, Apon, Daniel, Cooper, David, Dang, Quynh, Liu, Yi-Kai, Miller, Carl, Moody, Dustin, et al. *Status report on the first round of the NIST post-quantum cryptography standardization process*. US Department of Commerce, National Institute of Standards and Technology, 2019. DOI: 10.6028/NIST.IR.8240.

[2] Arute, Frank, Arya, Kunal, Babbush, Ryan, and et. al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. DOI: 10.1038/s41586-019-1666-5.

[3] Baldi, Marco, Bianchi, Marco, and Chiaraluce, Franco. Security and complexity of the mceliece cryptosystem based on quasi-cyclic low-density parity-check codes. *IET Information Security*, 7(3):212–220, 2013. DOI: 10.1049/iet-ifs.2012.0127.

[4] Cooper, C. On the distribution of rank of a random matrix over a finite field. In *Proceedings of the Ninth International Conference "Random Structures and Algorithms" (Poznan, 1999)*, volume 17, pages 197–212, 2000. DOI: 10.1002/1098-2418(200010/12)17:3/4<197::AID-RSA2>3.0.CO;2-K.

[5] Cossidente, Antonio, Korchmáros, Gabor, and Torres, Fernando. On curves covered by the Hermitian curve. *J. Algebra*, 216(1):56–76, 1999. DOI: 10.1006/jabr.1998.7768.

[6] Couvreur, Alain, Márquez-Corbella, Irene, and Pellikaan, Ruud. Cryptanalysis of McEliece cryptosystem based on algebraic geometry codes and their subcodes. *IEEE Trans. Inform. Theory*, 63(8):5404–5418, 2017. DOI: 10.1109/TIT.2017.2712636.

[7] de Castro, Francisco. fitmethis, Version 1.3.0.0, Jan 2020. MATLAB Central File Exchange.

[8] Delsarte, Philippe. On subfield subcodes of modified Reed-Solomon codes. *IEEE Trans. Inform. Theory*, IT-21(5):575–576, 1975. DOI: 10.1109/tit.1975.1055435.

[9] Driencourt, Yves. Some properties of elliptic codes over a field of characteristic 2. In Calmet, Jacques, editor, *Algebraic Algorithms and Error-Correcting Codes*, pages 185–193, Berlin, Heidelberg, 1986. Springer Berlin Heidelberg. DOI: 10.1007/3-540-16776-5{\textunderscore}721.

[10] Duursma, Iwan M. Algebraic decoding using special divisors. *IEEE Trans. Inform. Theory*, 39(2):694–698, 1993. DOI: 10.1109/18.212305.

[11] Duursma, Iwan M. Majority coset decoding. *IEEE Trans. Inform. Theory*, 39(3):1067–1070, 1993. DOI: 10.1109/18.256518.

[12] Duursma, Iwan Maynard. *Decoding codes from curves and cyclic codes.* Technische Universiteit Eindhoven, Eindhoven, 1993. Dissertation, Technische Universiteit Eindhoven, Eindhoven, 1993.

[13] El Khalfaoui, Sabira and Nagy, Gábor P. On the dimension of the subfield subcodes of 1-point Hermitian codes. *Advances in Mathematics of Communications*, 0(0):0, 2019. DOI: `10.3934/amc.2020054`.

[14] Eliason, Scott R. *Maximum likelihood estimation: Logic and practice.* SAGE Publications, Inc, 1993. DOI: `10.4135/9781412984928`.

[15] Feng, Gui Liang and Rao, T. R. N. Decoding algebraic-geometric codes up to the designed minimum distance. *IEEE Trans. Inform. Theory*, 39(1):37–45, 1993. DOI: `10.1109/18.179340`.

[16] GAP – Groups, Algorithms, and Programming, Version 4.10.2, Jun 2019.

[17] Hirschfeld, J. W. P., Korchmáros, G., and Torres, F. *Algebraic curves over a finite field.* Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2008. DOI: `10.1515/9781400847419`.

[18] Høholdt, Tom and Pellikaan, Ruud. On the decoding of algebraic-geometric codes. *IEEE Trans. Inform. Theory*, 41(6, part 1):1589–1614, 1995. DOI: `10.1109/18.476214`, Special issue on algebraic geometry codes.

[19] Høholdt, Tom, van Lint, Jacobus H., and Pellikaan, Ruud. Algebraic geometry codes. In *Handbook of coding theory, Vol. I, II*, pages 871–961. North-Holland, Amsterdam, 1998.

[20] Justesen, J., Larsen, K. J., Jensen, H. Elbrønd, and Høholdt, T. Fast decoding of codes from algebraic plane curves. *IEEE Trans. Inform. Theory*, 38(1):111–119, 1992. DOI: `10.1109/18.108255`.

[21] Justesen, Jørn, Larsen, Knud J., Jensen, H. Elbrønd, Havemose, Allan, and Høholdt, Tom. Construction and decoding of a class of algebraic geometry codes. *IEEE Trans. Inform. Theory*, 35(4):811–821, 1989. DOI: `10.1109/18.32157`.

[22] Kirfel, Christoph and Pellikaan, Ruud. The minimum distance of codes in an array coming from telescopic semigroups. *IEEE Trans. Inform. Theory*, 41(6, part 1):1720–1732, 1995. DOI: `10.1109/18.476245`, Special issue on algebraic geometry codes.

[23] Konishi, Sadanori and Kitagawa, Genshiro. *Information criteria and statistical modeling.* Springer Science & Business Media, 2008. DOI: `10.1007/978-0-387-71887-3`.

[24] Korchmáros, Gábor and Nagy, Gábor P. Hermitian codes from higher degree places. *J. Pure Appl. Algebra*, 217(12):2371–2381, 2013. DOI: `10.1016/j.jpaa.2013.04.002`.

[25] Kotz, Samuel and Nadarajah, Saralees. *Extreme value distributions.* Imperial College Press, London, 2000. DOI: `10.1142/9781860944024`, Theory and applications.

[26] Krachkovskii, V. Yu. Decoding of codes on algebraic curves. In *Conference Odessa*, 1988.

[27] Loidreau, Pierre. Strengthening McEliece cryptosystem. In *Advances in cryptology—ASIACRYPT 2000 (Kyoto)*, volume 1976 of *Lecture Notes in Comput. Sci.*, pages 585–598. Springer, Berlin, 2000. DOI: `10.1007/3-540-44448-3_45`.

[28] McEliece, Robert J. A public-key cryptosystem based on algebraic coding theory. *Coding Thv*, 42–44:114–116, 1978.

[29] Menezes, Alfred J., Blake, Ian F., Gao, XuHong, Mullin, Ronald C., Vanstone, Scott A., and Yaghoobian, Tomik. *Applications of finite fields*, volume 199 of *The Kluwer International Series in Engineering and Computer Science.* Kluwer Academic Publishers, Boston, MA, 1993. DOI: `10.1007/978-1-4757-2226-0`.

[30] Nagy, Gábor P. and El Khalfaoui, Sabira. HERmitian, Computing with divisors, Riemann-Roch spaces and AG-odes of Hermitian curves, Version 0.1, Mar 2019. GAP package.

[31] Niebuhr, Robert, Meziani, Mohammed, Bulygin, Stanislav, and Buchmann, Johannes. Selecting parameters for secure mceliece-based cryptosystems. *International Journal of Information Security*, 11(3):137–147, Jun 2012. DOI: `10.1007/s10207-011-0153-2`.

[32] Pellikaan, R. On the efficient decoding of algebraic-geometric codes. In *Eurocode '92 (Udine, 1992)*, volume 339 of *CISM Courses and Lect.*, pages 231–253. Springer, Vienna, 1993. DOI: `10.1007/978-3-7091-2786-5_20`.

[33] Peterson, W. W. Encoding and error-correction procedures for the Bose-Chaudhuri codes. *Trans. IRE*, IT-6:459–470, 1960. DOI: `10.1109/tit.1960.1057586`.

[34] Piñero, Fernando and Janwa, Heeralal. On the subfield subcodes of Hermitian codes. *Des. Codes Cryptogr.*, 70(1-2):157–173, 2014. DOI: `10.1007/s10623-012-9736-9`.

[35] Sakata, Shajiro, Justesen, Jørn, Madelung, Y., Jensen, Helhe Elbrønd, and Høholdt, Tom. Fast decoding of algebraic-geometric codes up to the designed minimum distance. *IEEE Trans. Inform. Theory*, 41(6, part 1):1672–1677, 1995. DOI: `10.1109/18.476240`, Special issue on algebraic geometry codes.

[36] Sakata, Shojiro. Extension of the Berlekamp-Massey algorithm to N dimensions. *Inform. and Comput.*, 84(2):207–239, 1990. DOI: `10.1016/0890-5401(90)90039-K`.

[37] Shiryaev, Albert N. *Probability. 1*, volume 95 of *Graduate Texts in Mathematics.* Springer, New York, 3 edition, 2016. DOI: `10.1007/978-0-387-72206-1`, Translated from the fourth (2007) Russian edition by R. P. Boas and D. M. Chibisov.

[38] Shor, Peter W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Review*, 41(2):303–332, 1999. DOI: `10.1137/S0036144598347011`.

[39] Skorobogatov, Alexei N. and Vlăduţ, Sergei G. On the decoding of algebraic-geometric codes. *IEEE Trans. Inform. Theory*, 36(5):1051–1060, 1990. DOI: `10.1109/18.57204`.

[40] Stepanov, Serguei A. *Codes on algebraic curves.* Kluwer Academic/Plenum Publishers, New York, 1999. DOI: `10.1007/978-1-4615-4785-3`.

[41] Stichtenoth, Henning. *Algebraic function fields and codes*, volume 254 of *Graduate Texts in Mathematics.* Springer-Verlag, Berlin, second edition, 2009. DOI: `10.1007/978-3-540-76878-4`.

[42] Studholme, Chris and Blake, Ian F. Random matrices and codes for the erasure channel. *Algorithmica*, 56(4):605–620, 2010. DOI: `10.1007/s00453-008-9192-0`.

[43] The MathWorks, Inc. *Statistics and Machine Learning Toolbox.* Natick, Massachusetts, United State, 2019.

[44] Véron, P. Proof of conjectures on the true dimension of some binary Goppa codes. *Des. Codes Cryptogr.*, 36(3):317–325, 2005. DOI: `10.1007/s10623-004-1722-4`.

Received 10th February 2020

Acta Cybernetica **24** (2020) 643–661.

Graph Coloring based Heuristic for Crew Rostering

László Hajdu[a], Attila Tóth[b], and Miklós Krész[c]

Abstract

In the last years personnel cost became a huge factor in the financial management of many companies and institutions.The firms are obligated to employ their workers in accordance with the law prescribing labour rules. The companies can save costs with minimizing the differences between the real and the expected worktimes. Crew rostering is assigning the workers to the previously determined shifts, which has been widely studied in the literature. In this paper, a mathematical model of the problem is presented and a two-phase graph coloring method for the crew rostering problem is introduced. Our method has been tested on artificially generated and real life input data. The results of the new algorithm have been compared to the solutions of the integer programming model for moderate-sized problems instances.

Keywords: crew rostering, graph coloring, tabu search

1 Introduction

In certain areas of the industry, the workers' work is performed not in a fixed order. The work activities are organized into shifts, which may vary in duration, time of the day and other properties. In generally, every worker has a contract with a defined expected worktime and a base salary for that, hence the overtime or the undertime is a large-scale extra cost for the company. In the life of the companies, the human cost is a significant part of the complete budget, hence they want to employ the workers in the most efficient way. In most cases, the scheduling of the workers has two different steps (see Figure 1). The first is the crew scheduling which means that the daily tasks are catenated into shifts so that each shift must meet the law constraints. The second step is the crew rostering. In this step the question is that how to assign the crew members to shifts for a work period called planning period which is typically being 1-3 month long. This study concentrates

[a] University of Szeged, InnoRenew CoE and University of Primorska. E-mail: hajdul@inf.u-szeged.hu, ORCID: https://orcid.org/0000-0002-1832-6944.

[b] University of Szeged, Hungary. E-mail: attila@jgypk.u-szeged.hu, ORCID: https://orcid.org/0000-0003-1014-7965.

[c] University of Szeged, InnoRenew CoE and University of Primorska. E-mail: kresz@jgypk.u-szeged.hu, ORCID: https://orcid.org/0000-0002-7547-1128.

DOI: 10.14232/actacyb.281106

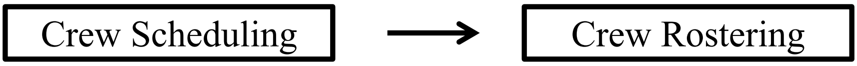


Figure 1: Scheduling of workers

on the rostering phase. It is important to note, that our solution is a proof of concept, even though it is possible to extend the core of the method with more complicated regulations. The objective of this paper is to show the efficiency of the core algorithm with basic regulations which are mostly used internationally as we did not intend to give a country specific solution. In real life applications very specific regulations are applied in several cases, such as the constraint of the travel time from home, or fair distribution of popular and not popular shifts can be also taken into account[2]. In summary, in real life environment the solution for crew rostering must meet some European Union and local regulations as well as the preferences of the workers.

There are two different cases concerned in the workers-shifts assignment. In the first case the companies assign an optimal number of crew members to the predefined daily shifts minimizing the total cost. In the second case there is a given set of workers and the firm assigns the shifts to them in a most efficient way. In this paper we deal with the first case i.e. we have shifts and the main goal is to minimize the overall cost.

The crew rostering problem is based on the generalized set covering model. Dantzig was the first who dealt with its mathematical application [7]. The crew rostering problem is NP-hard therefore it is generally considered that exact solution is not realistic to produce in the case of real life size problems. [12, 21, 18, 20, 25] By the above reason as a consensus of theory and practice, heuristic algorithms are used. The crew rostering solution methods have a quite rich literature, several overviews are available [4, 9, 22, 24, 10, 27]. The literature provides numerous examples of the issue, among which the most significant ones are the airline crew rostering [15], railway crew scheduling [17] and the driver scheduling [1, 19]. In the literature, the studies are generally grouped around the related application areas. These solutions need to correspond to the regulations of the company as well as to the "national norms". In most cases the regulations are different in each area, for instance the qualifications of the workers are handled in different ways in airline crew rostering, while in driver rostering it is usually ignored. These regulations are formalized as constraints which usually have two different types. A hard constraint must not be violated, while in a case of a soft constraint it is allowed with being penalized by some extra cost. For example single workday in a long days-off period or work in a requested days-off can be handled by soft constraints. In this paper, we only deal with hard constraints but the model and methodology can be easily extended to soft constraints with using appropriate penalties in the objective function.

Taking into consideration the basic regulations of all the significant areas (typically regulated in national laws) we have applied the following constraints:

1. A worker can have maximum one shift in one day.

2. There is a minimum rest time between two shifts.

3. There is a maximum total worktime during one week.

4. There is a minimum number of days-off in one month.

5. There is a maximum number of consecutive workdays.

Every worker has a contract which determines the expected worktime. The difference between this defined worktime and the length of the shifts may produce overtime or undertime. For overtime the worker receives extra payment above the basic salary. However, the basic salary must be paid even in the case of undertime. Therefore the human resource extra cost on the schedules shifts can be optimised through minimizing the overtime.

In this paper, we give a new heuristic solution method based on graph coloring which fits to many application area. Our algorithm has two main steps. In the initial rostering, the algorithm estimates the number of workers, builds a conflict graph from the shifts, and generates days-off pattern for every worker. One of the key innovations of the method is to generate days-off patterns which meet the basic regulations and create a frame for the problem, and to get the additional free days indirectly from the tabu search method. Afterwards the graph is colored, so an initial rostering is created by a modified greedy algorithm. When an initial solution is generated the graph is iteratively recolored with a tabu search method to reduce the cost. An important part of the method is to balance the shifts among workers to create a solution where workers are close enough to their expected workload. The results of the algorithm have been compared to the results of the integer programming model with moderate problem size. These results show that our algorithm is efficient and robust. Our solution is a wireframe for the general crew rostering problem, however we tested our method in a real-life application area of the driver rostering case. In the next two sections the crew rostering problem is defined and the mathematical model of the problem is introduced. Finally, in Section 4 and 5 our heuristic method and the test results will be presented.

2 Problem formulation

The crew rostering problem is formally defined as the following. Let C be the set of workers. The set of shifts denoted by S which needs to be carried out. A shift is composed of a series of daily tasks. A shift is defined by its date, starting and ending time in the day, duty time (i.e. the time between the starting and ending time), and the working time. The working time might differ from the duty time since a shift may contain idle periods when the worker does not work. The aim is to assign the crew members to the shifts with minimal cost. Consequently, let $f = S \rightarrow C$ be an assignment where the shifts are covered by the workers where exactly one worker assigned to each shift.

Let $ct(i)$ be the contract type of worker i. The type of the contract defines the prescribed worktime in average for a single workday. Let $aw(ct)$ denote the expected daily working hours by contract type ct. Based on the contract it is possible to calculate the expected worktime in the planning period. For example in our case, every crew member has a contract which defines eight working hours per day. In our model, the type of the contract is a parameter, therefore it is possible to deal with different contract types also. The expected worktime can be defined in the following way:

$$expected\ worktime(i) = number\ of\ workdays * aw(ct(i)) \quad i \in C \tag{1}$$

The basic employment cost is based on workers' expected worktime defined by their contract. So in optimal case every worker will work according to her/his expected worktime. However, in case of working over the expected worktime, employers have to provide extra salary for this overtime. Therefore, the cost is the following:

$$cost = \alpha * overtime + \beta * employment\ cost \tag{2}$$

where α and β are pre-defined weights. In a real problem these multipliers can adjust the different costs to the preferences of the company. Following the practice we suppose that the employment cost is proportional to the working hours prescribed by the contract type. Hence, the objective function will consider in the minimization both the overtime and the number of workers. We also assume that the planning period P is fixed (typically 1-3 months) and all the rules having no specified time period (e.g. the average working time) are considered with respect to P; with this approach we follow the practice as well.

3 Mathematical model

Let D be the set of the days, $Week$ the set of the weeks and Mon the set of the months in the given planning period. Consequently D_w^{Week} denotes the days of the week w and D_m^{Mon} is the days of the month m. Meanwhile the length of the planning period (the number of days) is defined by l, and let l_m be the number of days in the month m. The set of the workers is denoted by C, and the set of shifts by S where S_p is the set of shifts on day p. Let SS_{jk} is a compatibility relation between the shifts, where its value is 1 if shift j and shift k can be assigned to the same worker, and 0 otherwise (can be used to define Rule 2). Let WT be the maximum worktime on a week (for Rule 3), WD be the maximum number of consecutive workdays (for Rule 5) and D_p^{WD} be such number of the consecutive days beginning from day p ($WD + 1$ *days in a row*). The minimum number of days-off is denoted by RD (for Rule 4). Let the expected worktime in a month be $aw(ct(i, m))$, where $m \in Mon$ and $i \in C$. In order to minimize the number of workers let $cc(i)$ be the operational cost of worker i, i.e the base salary. Let $wt(j,p)$ be the worktime of shift j on day p. Finally let α be the weight of the overtime and β be the weight of the employment cost. We need the following variables to define the model:

Driver i assigned to shift j:

$$x_{ij} \in \{0,1\} \quad \forall i \in C,\ \forall j \in S \tag{3}$$

Driver i works on day p:

$$z_{ip} \in \{0,1\} \quad \forall i \in C,\ \forall p \in D \tag{4}$$

Driver i works in the planning period:

$$y_i \in \{0,1\} \quad \forall i \in C \tag{5}$$

Overtime of worker i:

$$\pi_i \geq 0 \quad \forall i \in C \tag{6}$$

The constraints of the integer programming model are the followings:

Exactly one worker is assigned to each shift.

$$\sum_{i \in C} x_{ij} = 1 \quad \forall j \in S \tag{7}$$

There is at most one shift assigned to worker on a given day.

$$\sum_{j \in S_p} x_{ij} = z_{ip} \quad \forall i \in C,\ \forall p \in D \tag{8}$$

An employee works in the planning period if he/she has at least one shift.

$$\sum_{p \in D} z_{ip} \leq l y_i \quad \forall i \in C \tag{9}$$

The following constraint excludes the possibility of assigning two incompatible shifts to a worker.

$$x_{ij} + x_{ik} \leq SS_{jk} + 1 \quad \forall i \in C, \quad \forall j,k \in S \tag{10}$$

The worktime must not exceed the maximum working time in any week.

$$\sum_{p \in D_w^{Week}} \sum_{j \in S_p} x_{ij} wt(j,p) \leq WT \quad \forall i \in C,\ \forall w \in Week \tag{11}$$

A worker can have at most the maximum number of consecutive shifts.

$$\sum_{p \in D_q^{WD}} z_{ip} \leq WD \quad \forall i \in C,\ \forall q \in D \tag{12}$$

At least the required number of days-off have to be given for each worker in a month.

$$\sum_{p \in D_m^{Mon}} z_{ip} \leq l_m - RD \quad \forall i \in C,\ \forall m \in Mon \tag{13}$$

Let us define the overtime.

$$\sum_{m \in Mon} \Big(\sum_{p \in D_m^{Mon}} \sum_{j \in S_p} x_{ij} wt(j,p) - aw(ct(i,m)) \Big) \leq \pi_i \quad \forall i \in C \tag{14}$$

The objective function minimizes the sum of the weighted overtime and the operational cost.

$$\min \sum_{i \in C} (\alpha \pi_i + \beta y_i cc(i)) \tag{15}$$

The mathematical model is a good approach if the problem is relatively small because it gives an exact solution for the problem. Unfortunately in real life there are often too big problems for these solution methods therefore in most cases the companies use heuristics.

4 Heuristic method

Our algorithm is a two-phase graph coloring method (TPC), where in the first step an initial rostering is produced with a graph coloring procedure and in the second step this rostering is improved with the over- and undertime being minimized by tabu search. M. Gamache et. al [11] developed a method, where graph coloring algorithm based tabu search was used for scheduling pilots. These pilots have qualification and the objective is to find a feasible solution in a pre-specified workload interval. Nevertheless, in the literature tabu search method is a widely used technique especially in bus driver scheduling or rostering. Some example papers using tabu search to solve the scheduling problem (defining the shifts) are [6, 26], but the method was also successfully applied for laboratory personnel scheduling problems [3]. However, in our case both the problem and the solution method are different by two reasons: the regulations are different and here days-off patterns are used. Furthermore, we will give a general solution method for the crew rostering, serving as a framework to be specialized to several fields. In our approach by its universality and flexibility, days-off patterns will be applied with a k-coloring algorithm.

The TPC has the following steps:

1. Initial rostering.

a Estimate the number of workers.

b Generate days-off patterns.

c Build a conflict graph.

d Color the graph.

2. Tabu Search to improve the solution.

The initial rostering determines the number of workers, the days-off patterns and the initially colored graph. First, the algorithm estimates the optimal number of workers and for each worker generates a days-off pattern. These patterns will define whether a worker can work on a day or not. In the last two steps of the initial phase the conflict graph is built and colored. In most cases the cost of the initial rostering can be improved, therefore in the second phase the method switches the shifts between the workers with a local search method in order to minimize the cost of the rostering.

It is important to note that in our heuristic the objective function is divided into two parts (see Formula 2). The algorithm tries to minimize the number of workers in the initial phase, while the recoloring step is to minimize the overtime. A feature of the heuristic that we get better solution if we also minimize the undertime: we will see below that it is an equivalent approach with respect to Formula 2 and it will provide a solution in which the workload will be balanced among the employees.

Nevertheless, the general cost function of the method is based on the overtime and undertime only. Since the employment cost is proportional to the working hours by contract (expected worktime), we will see that Formula (2) is equivalent to the weighted linear combination of overtime and undertime. Formally, the overtime and the undertime definition and the cost of the heuristic are the following.

$$overtime = \sum_{i \in C} \max\left(0, worktime(i) - expected\ worktime(i)\right) \tag{16}$$

$$undertime = \sum_{i \in C} abs(\min\left(0, worktime(i) - expected\ worktime(i)\right)) \tag{17}$$

$$cost = \alpha' * overtime + \beta' * undertime \tag{18}$$

In order to clarify the equivalence of Formulas (2) and (18), notice that the total worktime required by the shifts (the sum of the worktime of all the shifts) in the whole planning period is a constant. Therefore it is easy to see that in any solution the expected worktime is equal to ($total\ worktime - overtime + undertime$). By this we obtain for Formula (2):

$$cost = \alpha * overtime + \beta * (total\ worktime - overtime + undertime) \tag{19}$$

Then ($\beta * total\ worktime$) is a constant and provides the basic theoretical lower bound in the cost. Therefore $\alpha' = \alpha - \beta$ and $\beta' = \beta$ will make Formulas (2) and (18) equivalent from optimization point of view with (18) expressing the extra cost above the basic theoretical lower bound.

4.1 Initial rostering

4.1.1 Estimate the number of workers

At first, the initial set of workers (C^*) needs to be defined wherewith the shifts can be effectively covered. The number of these workers can be estimated for the planning period using the total worktime of the shifts and the contracts of the workers. In our case we concern "uniform workers" meaning that they have the same type of contracts, i.e. their expected worktime is the same. Basically, we suppose that the planning period is one or a few months (typically 1-3 months). This assumption is general in real life situations, but it can be easily relaxed to any length of the planning period. Therefore, the number of the crew members for a given contract type *ct* is given by the following way:

$$|C^*| = round\left(\frac{total\ worktime}{aw(ct) * number\ of\ workdays}\right)$$

Practically, an initial set of workers is given in such a way that the expected worktime is assigned to all of the workers determined by the contract type in every day of working. In this way, the estimated number will be the cardinality where the difference of the total worktime and the expected worktime for this set is minimal.

A trivial lower bound for the number of workers can also be given by the number of the shifts on the busiest days, since one crew member can have at most one shift a day. Furthermore, if the estimated number of workers is known, a theoretical lower bound (LB^{ot}) for the overtime can be given by the following way:

$$LB^{ot} = \max\left(0, (total\ worktime - |C^*| * aw(ct) * number\ of\ workdays)\right)$$

It is clear that with a given set of workers LB^{ot} is correct since the overtime is minimal in this case if every worker has workload at least according to his/her contract.

4.1.2 Generate days-off patterns

In some areas, it may be necessary to pre-specify days for a worker in advance when he or she does not work (called days-off) in the planning period (called days-off pattern). Some example for the usage of days-off patterns can be found in the literature [23, 8]. It is a widely accepted and used methodology to generate a days-off for the workers in the crew rostering. The advantage of the days-off patterns is that the days-off requested previously by the workers can be taken into account, as well as the 4th (minimum number of days-off in one month) and 5th (maximum number of consecutive workdays) rules are fulfilled here, so we do not have to concern these in the latter phases.

Days-off patterns define different fixed free days for each worker. We tried 3 different methods to generate the days-off patterns. At first we generated random patterns as presented in Table 1. The days-off are generated by weighted random generator which consider the number of the shifts on the days and Rule 4 and

Rule 5. On days with higher shift load the workers received a day off with lower probabilities than those days where lower number of shifts are. To formalize the probability let $|C*|$ be the estimated number of workers and S_j denote the set of the shifts on the day j, the probability of the days-off on the day j is given by $1 - (|S_j|/|C*|)$; if the generated days-off pattern doesn't meet with the rules, we insert additional free days into the pattern randomly. The pattern vector is generated for every worker one by one.

Table 1: Random days-off patterns

Worker	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1.	0	1	0	0	1	0	1
2.	1	0	1	1	0	0	0
3.	1	0	0	0	0	1	0
4.	0	1	0	0	1	0	1
5.	0	0	1	0	0	1	0
6.	0	0	1	0	1	0	1
7.	0	0	1	0	0	1	0

The second tested pattern type was the so called 5-2 pattern (see Table 2). Here, the workers get two days-off after each consecutive five workdays. These days-off patterns repeat a seven days long sub-pattern along the planning period such that if the sub-pattern starts at day $j*$ then from day $j*$ to day $j*+4$ the days are workdays and days $j*+5$ and $j*+6$ are days-off. The next pattern shifts the sub-pattern forward with one day. Therefore, seven different patterns are generated from this days-off pattern type.The 5-2 pattern seemed to be appropriate, since a week usually contains 5 workdays and 2 days off. Hence, if everyone will work by this 5-2 pattern, then their working hours are likely close to their expected worktime. It is clear that the 5-2 days-off patterns also meet the defined hard constraints.

We found that the random and the 5-2 patterns are too rigid and greedy with producing too much exclusion in the first phase. This means that during the graph coloring the days-off patterns exclude too many workers on days where they could possibly work.

To overcome this problem the third pattern type defines minimal fixed days-off considering the rules. Thus, we propose a 6-1 days-off pattern which means that the workers get one fixed days-off after every six consecutive workdays. These patterns are generated with the same method as 5-2 patterns, but in the sub-pattern from days $j*$ to $j*+5$ are workdays and day $j*+6$ is days-off (see in Table 3). This pattern meets both the minimal free days rule (Rule 4) and the maximal consecutive workdays (Rule 5) based on the European regulations (see in Section 5). It causes only a few exclusion during the graph coloring. Also, the tabu search will have a relatively big state space. Naturally, the pattern defines only fixed days-off and in the last phase we may get automatically some additional days-off in the

Table 2: 5-2 days-off patterns

Worker	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1.	0	0	0	0	0	1	1
2.	0	0	0	0	1	1	0
3.	0	0	0	1	1	0	0
4.	0	0	1	1	0	0	0
5.	0	1	1	0	0	0	0
6.	1	1	0	0	0	0	0
7.	1	0	0	0	0	0	1

final solution: when the workers are not assigned to a shift they automatically have a day off. Because of the days-off patterns, the 4th and 5th rules does not need to be taken into account later in step 2. In the Table 2 and Table 3 it is enough to give the day-off patterns for the first seven workers, since if i and j are the indexes of the workers, and positive integers, furthermore $i \equiv j\ mod(7)$ then the $i-th$ and $j-th$ workers have the same days-off pattern.

Table 3: 6-1 days-off patterns

Worker	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1.	0	0	0	0	0	0	1
2.	0	0	0	0	0	1	0
3.	0	0	0	0	1	0	0
4.	0	0	0	1	0	0	0
5.	0	0	1	0	0	0	0
6.	0	1	0	0	0	0	0
7.	0	0	0	0	0	0	1

4.1.3 Build a conflict graph

Let $G = (V, E)$ be a graph so called conflict graph, where every shift in the planning period is a vertex of the graph. There is an edge between two vertices if the corresponding shifts must not be performed by the same worker. For example if the time between two shifts is less than defined by the 2nd rule or they are on the same day.

To understand the graph building let us see an example. Let the initial input of two days with 8 shifts is prescribed in Table 4. The shifts of each day compose cliques. There are 3 additional edges between them (1-5,1-6,3-5) because the time period between these shifts is less than the required one by Rule 2 (general value is 12 hours what we use here too). The generated graph is presented by Figure 2.

Table 4: Example shifts with their date,startin and ending time in minutes (0-1440 a day), and the contained working time in minutes.

Id	Date	Begin	End	Worktime
1	2016.01.01	889	1411	522
2	2016.01.01	540	984	444
3	2016.01.01	714	1135	421
4	2016.01.01	237	713	476
5	2016.01.02	396	850	454
6	2016.01.02	454	992	538
7	2016.01.02	702	1138	436
8	2016.01.02	814	1327	394

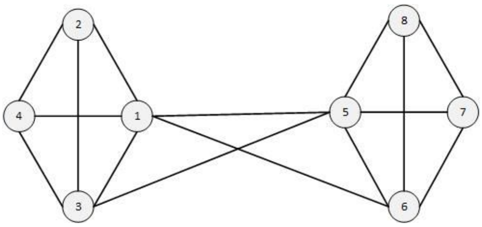


Figure 2: Example graph

4.1.4 Color the graph

Node coloring of a graph means assigning a color to each node in such a way that every neighboured node has different color. A coloring of the conflict graph gives a rostering for the problem if one color corresponds one worker. This rostering is correct since it fulfills the defined rules: Rule 1 and Rule 2 are guaranteed by the structure of the conflict graph, Rule 4 and 5 are fulfilled by the days-off pattern and Rule 3 is handled in the coloring. If we set the ending time of the vertices to the maximum of the increased value by the time of the 2nd rule and that of the end of the day, then we will obtain an interval graph. [16] There is efficient algorithm to color interval graph [14], but due to the extra restrictions of the crew rostering problem it becomes NP-hard. A good overview of the problem written by Kolen et. al [28]. Such a k-coloring algorithm is needed while adding a new color is possible. The initial estimation of the number of workers is usually correct, but some extraordinary inputs, e.g shifts with too short worktime, result such a graph that could not be colored with the estimated number of colors.Therefore we used the DSATUR algorithm which is effective and can be easily adopted to our needs (see Brelaz [5]) with coloring the nodes by their saturation values (saturation of a vertex represents the number of different color classes in its neighbourhood). Each coloring iteration chooses the node with the highest saturation value and chooses the color where the corresponding worker is in maximum undertime. It starts with the estimated number of colors (i.e worker) and adds new color if it is necessary (i.e no available worker for a shift).

Algorithm 1 Graph coloring.

```
While there is an uncolored vertex
Let $v$ be the next uncolored vertex with the maximal saturation value
If there is an equal saturation we choose the vertex with maximal degree
$C \leftarrow$ available colors where the regulations are not violated
Choose available color $c \in C$ (i.e  worker) with the lowest worktime.
If $c$ does not exist let $c$ be a new color.
Assign $c$ to $v$.
for each $v_n \in neighbours\ of\ v$ do
    $v_n \leftarrow update\ saturation\ value$
end for
end while
```

The graph coloring gives an initial solution which meets the given constraints and assigns a worker to every shift. Since the real working time of each worker and their expected worktime is known, the initial cost can be calculated.

4.2 Tabu Search

The tabu search was introduced by Glover in 1986 and it is one of the most famous local search techniques [13]. The state space of the tabu search denoted by SL consists of all the feasible coloring patterns in the conflict graph. Each state is a coloring and the objective function has to be minimized on SL. The algorithm visits solutions $sl_0, sl_1, ...sl_n \in L$, where sl_0 is the initial solution produced by the graph coloring and $sl_{i+1} \in N(sl_i)$ with $N(sl)$ denoting the set of neighbours of sl. The next neighbour is chosen by a first fit method meaning that the first neighbouring solution is chosen being better than the actual solution and the neighbours are in a random order. Steps chosen in one step are stored in the tabu list denoted by TL. The steps in the list are forbidden to repeat. The neighbours of a solution sl are defined by using the following operations:

1. *Recoloring of a vertex:* Another color is given to a vertex of the graph. It means that a shift is taken away from a worker and given to another one permitted by the constraints.

2. *Swapping the colors of two vertices:* The colors of two vertices are switched, therefore shifts are swapped between two workers in a way that both of them receive a shift which they can carry out with keeping the constraints.

The algorithm tries to change or swap a shift between the colors of the most undertimed worker and the most overtimed worker. After carrying out the vertex recoloring or the vertex swapping, the step is added to the tabu list. So the moves are stored in the tabu list instead of the states. The elements spend a specified number of iterations in the tabu list, and if the length of the tabu list is greater than the max allowed iteration for an element on the list, then the oldest element which took the longest time on the TL, will be deleted from it. Pseudocode is given in Algorithm 2.

Algorithm 2 Tabu search.

```
 1: $s_0 \leftarrow initial\ solution$
 2: $TL \leftarrow \emptyset,\ s \leftarrow s_0,\ best \leftarrow s_0$
 3: while $stopping\ criteria$ do
 4:     $E \leftarrow workers\ ordered\ by\ worktime$
 5:     for each $e_1 \in E\ from\ undertimed\ to\ overtimed\ order$ do
 6:         for each $e_2 \in E\ from\ overtimed\ to\ undertimed\ order$ do
 7:             $s* \leftarrow the\ first\ recoloring\ or\ swap\ between\ e_1\ and\ e_2\ where\ cost(s*) <$
                $cost(s)\ and\ move(s, s*) \notin TL$ and goto line 10
 8:         end for
 9:     end for
10:     if $s * not\ exist$ then
11:         $s* \in N(s)\ the\ first\ solution\ where\ move(s, s*) \notin TL$
12:     end if
13:     if $cost(best) > cost(s*)$ then
14:         $best \leftarrow s*$
15:     end if
16:     $TL \leftarrow TL \cup move(s, s*)$
17:     if $TL.size > max\ size\ of\ tabu\ list$ then
18:         Remove the oldest element from $TL$ which took the longest time on the
            list
19:     end if
20:     $s \leftarrow s*$
21: end while
```

Multiple methods can be applied as stopping criteria. In most cases, the criteria is based on the iteration number or on the time bound. In our case, the algorithm stops if it can not find a better solution within a certain number of iteration.

5 Test results

In this section we will introduce the specific regulations for our case study, the test cases and the results of our algorithm. The algorithm was implemented in Java language, and IBM Cplex 12.4 software was used to solve the integer programming part (see Section 3).

The heuristic was run on the following computer:

- Processor: Intel Core I7 970 3.2Ghz
- Memory: 14Gb.
- Operation System: Microsoft Windows 7 Enterprise 64bit

Since the problem definition ignores the personal conditions i.e. illness or holiday, usually the rostering method is a part of a decision support system and the members of the crew are fictive workers. One roster is just the duty which can be performed by one worker following the rules. Therefore, the rostering problem deals with driver types. In this study, one driver type is used but it can be extended several different worker types. Although the presented method is general for any crew rostering problem, we tested it with rostering of bus drivers of a city public transport company. In our case every driver had a contract of 8 hours of worktime with respect to an average workday. The planning period (the timeframe with respect to which the average daily working time should be 8 hours) was 1 month. The following regulation constraints were used for our test cases, as being the general rules in the driver rostering area:

- The minimum rest time between two shifts lasts for 12 hours (Reg. 2)
- The worktime must be less than 48 hours in a week (Reg. 3.)
- Drivers must have at least 4 free days in a month (Reg. 4.)
- The number of the maximum consecutive workdays is 6 (Reg. 5.)

The stopping criteria of the algorithm is given in a way that it ends when no improvement in solutions is found after 200 iterations, or when drivers with under- or overtime left only. In the objective function of the mathematical model $cc(i)$ was defined by the expected worktime, and we set $\alpha = 2$ and $\beta = 1$. We evaluated the methods with Formula (18) during the testing, where $\alpha' = 1$ and $\beta' = 1$ by the notes following the definition of the formula. We have chosen this case as the typical example when overtime dominates the cost function. The use of parameters α and β makes our approach flexible: if we consider the above natural case ($cc(i)$ being the expected worktime), then practically $\beta = 1$ and $1 \leq \alpha \leq 2$. Since our goal is to show the applicability of the method as a proof of concept, we made our testing on the above basic case.

The algorithm was tested on real-life problems and on randomly generated data. The real-life instances came from the bus driver scheduling department of the local city bus company Szeged, a mid-size city of Hungary. The artificial problem instances are generated in such a way that the properties of the shifts reflect the characteristics of real life data. The shift distributions of the generated samples are the same with the following properties of the generated shifts:

- Worktime is between 7 and 9 hours.
- Duty time is between 6 and 10 hours.
- The division of the shifts on a day follows the normal distribution.

The tests on the generated inputs were running in five problem groups of different size and we also tested our algorithm on a real world input (25,50,75,100,200,real). In this context by the size we mean the number of shifts on the most loaded days of the planning period. The workdays are considered with the maximal number of shifts, while weekends are generated with approximately 80-90% of this maximum. The time limit of the CPLEX was 8 hours in each case.

Table 5: Test results on size (5, 50, 75, 100, 200, real)

Input	Lower bound	Cost heur	Time heur	Cost IP	Time IP
gen25_1	4807	4807	3.2	4900	18.77
gen25_2	4658	4658	3.6	4691	495.2
gen25_3	1825	1825	4.1	1855	154.8
gen25_4	4371	4371	3.1	4457	22.96
gen25_5	4815	4815	5.4	4911	18.11
gen50_1	3359	3359	5.4	3415	875.1
gen50_2	2581	2581	15.2	2581	844.8
gen50_3	3000	3000	10.4	3047	1102.34
gen50_4	1671	1671	4.8	1696	1252.78
gen50_5	1925	1925	10.3	1960	1529.56
gen75_1	2861	2861	50.9	-	-
gen75_2	4051	4051	65.1	-	-
gen75_3	3809	3809	88.9	-	-
gen75_4	90	100	79.4	-	-
gen75_5	4682	4682	65.3	-	-
gen100_1	3261	3361	124.4	-	-
gen100_2	987	2581	145.2	-	-
gen100_3	1663	3000	160.4	-	-
gen100_4	1671	1671	121.8	-	-
gen100_5	1925	1925	130.3	-	-
gen200_1	3564	3564	420.4	-	-
gen200_2	4895	4895	321.2	-	-
gen200_3	3771	3771	345.4	-	-
gen200_4	2930	3930	329.8	-	-
gen200_5	2029	2029	298.3	-	-
volan_real	4897	4897	9.87	5012.87	21418.56

The results of TPC are compared to the lower bound (see Section 4.1.1) and the solution of the IP, see Table 5. It can be stated that in most cases our algorithm produced low running time and good solutions for each test case. The time complexity is found much higher for the IP than the heuristic in all cases. Though heuristic can not guarantee optimal solution, but in practical situations the running time is also an important aspect. However, as the problem size increased the running time of IP became unacceptable.

As the test cases have shown, the iteration number was high enough to reach the lower bound in most cases. The IP was running with a relative gap of 2% in a case of the generated input and 4% in a case of the real input which means the IP stopped when the actual solution was maximum of 2% and 4% distance from the optimal solution. The above fact explains why the costs of the IP are slightly higher than that of the TGPM.

The real input consisted of approximately 4000 shifts in a one month planning period, i.e. approximately 120-140 shift per each workday. The number of the working drivers in the final solution was 150. The results of the real input were tested with the integer programming model, the results of the IP can be found in the last row of the table. With the TPC the problem could be solved with the estimated number of drivers in every case. We obtained that TPC is able to handle relatively large inputs producing good quality feasible solutions in reasonable running time. Furthermore, in the majority of the test cases, our method has reached the theoretical lower bound.

6 Conclusion and future works

In this paper important aspects of the crew rostering in the scheduling of bus drivers were introduced. First a mathematical model of the problem was defined. Then we proposed the TPC method for the crew rostering problem, which is a proof of concept dealing with some general international regulations. In the initial phase a preliminary rostering is constructed. First the number of workers is estimated, then days-off patterns are generated, and a conflict graph is built. Lastly, to produce an initial rostering, the graph is colored with the estimated number of colors, where each color refers to an worker. In the second phase, a tabu search method recolors the graph to reduce the cost by minimizing the total undertime and overtime. Our method has been tested with artificially generated and real life instances. The real instances belongs to the local public transport company of Szeged. For moderate sized problems, the results of the presented algorithm have been compared to the solutions of the appropriate integer programming model and to a lower bound. TPC produced a satisfactory running time, reached the lower bound in most cases and returned a feasible solution in all cases. In the future the running time could be improved with other methods such as paralell programming and additional regulations could be taken into account in the conflict graph with additional edges, or with penalties in the objective function.

7 Acknowledgement

László Hajdu was supported by the project "Integrated program for training new generation of scientists in the fields of computer science", no EFOP-3.6.3- VEKOP-16-2017-0002 (supported by the European Union and co-funded by the European Social Fund).

Attila Tóth acknowledges the support of the National Research, Development and Innovation Office - NKFIH Fund No. SNN-117879.

Miklós Krész was supported by the ARRS project N1-0093 (Public call for financing the Slovenian part of bilateral Hungarian and Slovenian Cooperation with NKFIH – National Research, Development and Innovation Office).

Miklós Krész and László Hajdu gratefully acknowledge the European Commission for funding the InnoRenew CoE project (Grant Agreement 739574) under the Horizon2020 Widespread-Teaming program, and the Republic of Slovenia (Investment funding of the Republic of Slovenia and the European Union of the European Regional Development Fund), and the support of the EU-funded Hungarian Grant EFOP-3.6.2-16-2017-00015.

The authors thank Tisza Volán for the test data and the technical consultation, and the anonymous referees whose detailed comments helped to improve the presentation of the paper.

References

[1] A. Wren, J. Rousseau. Bus driver scheduling: An overview. *Comput. Aided Sched. Pubilc Trahnsport.*, pages 173–187, 1995. DOI: 10.1007/978-3-642-57762-8_12.

[2] Abbink, Erwin, Albino, Luis, Dollevoet, Twan, Huisman, D., Roussado, Jorge, and Saldanha, Ricardo. Solving large scale crew scheduling problems in practice. *Public Transport*, 3:149–164, 06 2011. DOI: 10.1007/s12469-011-0045-x.

[3] Adamuthe, Amol and Bichkar, Rajan. Tabu search for solving personnel scheduling problem. In *Proceedings - 2012 International Conference on Communication, Information and Computing Technology, ICCICT 2012*, pages 1–6, 10 2012. DOI: 10.1109/ICCICT.2012.6398097.

[4] Alfares, Hesham K. Survey, categorization, and comparison of recent tour scheduling literature. *Annals of Operations Research*, 127(1):145–175, Mar 2004. DOI: 10.1023/B:ANOR.0000019088.98647.e2.

[5] Brélaz, Daniel. New methods to color the vertices of a graph. *Commun. ACM*, 22(4):251–256, April 1979. DOI: 10.1145/359094.359101.

[6] Chen, Mingming and Niu, Huimin. A model for bus crew scheduling problem with multiple duty types. *Discrete Dynamics in Nature and Society*, 2012, 09 2012. DOI: 10.1155/2012/649213.

[7] Dantzig, G.B. A comment on edies traffic delays at toll booths. *Operations Research*, 2:339–341, 1954. DOI: `10.1287/opre.2.3.339`.

[8] Elshafei, Moustafa and Alfares, Hesham K. A dynamic programming algorithm for days-off scheduling with sequence dependent labor costs. *J. of Scheduling*, 11(2):85–93, April 2008. DOI: `10.1007/s10951-007-0040-x`.

[9] Ernst, A. T., Jiang, H., Krishnamoorthy, M., and Sier, D. Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, 153(1):3–27, 2004. DOI: `10.1016/s0377-2217(03)00095-x`.

[10] Ernst, Andreas Tilman, Jiang, H, Krishnamoorthy, Mohan, and Sier, David. Staff scheduling and rostering: a review of applications, methods and models. *European Journal of Operational Research*, 153(1):3 – 27, 2004. DOI: `10.1016/S0377-2217(03)00095-X`.

[11] Gamache, Michel, Hertz, Alain, and Ouellet, Jérôme Olivier. A graph coloring model for a feasibility problem in monthly crew scheduling with preferential bidding. *Comput. Oper. Res.*, 34(8):2384–2395, August 2007. DOI: `10.1016/j.cor.2005.09.010`.

[12] Garey, Michael R. and Johnson, David S. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1979. DOI: `10.2307/2273574`.

[13] Glover, Fred and Laguna, Manuel. *Tabu Search.* Kluwer Academic Publishers, Norwell, MA, USA, 1997.

[14] Golumbic, M.C. Algorithmic graph theory and perfect graphs. *Academic Press, San Diego, California,*, 1980. DOI: `10.1016/c2013-0-10739-8`.

[15] H., Dowling D. Krishnamoorthy M. Mackenzie and D., Sier. Staff rostering at a large international airport. *Annals of Operations Research*, 72:125–147, 1997. DOI: `10.1023/A:1018992120116`.

[16] Hajos, G. Über eine art von graphen. *Int. Math Nachrichten*, 1957.

[17] Heil, Julia, Hoffmann, Kirsten, and Buscher, Udo. Railway crew scheduling: Models, methods and applications. *European Journal of Operational Research*, 2019. DOI: `https://doi.org/10.1016/j.ejor.2019.06.016`.

[18] John J. Bartholdi, III. A guaranteed-accuracy round-off algorithm for cyclic scheduling and set covering. *Operations Research*, 29(3):501–510, 1981. DOI: `10.1287/opre.29.3.501`.

[19] Kemeny, V. Argilan A. Toth B. David C. and Pongracz, G. Greedy heuristics for driver scheduling and rostering. *Proc. 2010 Mini-Conference Appl. Theor. Comput. Sci.*, pages 101–108, 2010.

[20] Lau, Hoong Chuin. On the complexity of manpower shift scheduling. *Computers & Operations Research*, 23(1):93–102, 1996. DOI: `10.1016/0305-0548(94)00094-o`.

[21] Marx, Daniel. Graph colouring problems and their applications in scheduling. *Periodica Polytechnica Electrical Engineering*, 48(1–2):11–16, 2004.

[22] Meisels, Amnon and Schaerf, Andrea. Modelling and solving employee timetabling problems. *Annals of Mathematics and Artificial Intelligence*, 39(1):41–59, Sep 2003. DOI: `10.1023/A:1024460714760`.

[23] Mesquita, Marta, Moz, Margarida, Paias, Ana, and Pato, Margarida. A decompose-and-fix heuristic based on multi-commodity flow models for driver rostering with days-off pattern. *European Journal of Operational Research*, 245(2):423 – 437, 2015. DOI: `10.1016/j.ejor.2015.03.030`.

[24] Nurmi, K., Kyngäs, J., and Post, G. Driver rostering for bus transit companies. *Engineering Letters*, 19(2):125–132, December 2011.

[25] O., Fukunaga A. Hamilton E. Fama J. Andre D. Matan and I., Nourbakhsh. Staff scheduling for inbound call and customer contact centers. *AI Magazine*, 23(4):30–40, 2002.

[26] Shen, Yindong and Kwan, Raymond S. K. Tabu search for driver scheduling. In Voß, Stefan and Daduna, Joachim R., editors, *Computer-Aided Scheduling of Public Transport*, pages 121–135, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. DOI: `10.1007/978-3-642-56423-9_7`.

[27] Van den Bergh, Jorne, Beliën, Jeroen, Bruecker, Philippe, Demeulemeester, Erik, and De Boeck, Liesje. Personnel scheduling: A literature review. *European Journal of Operational Research*, 226:367–385, 05 2013. DOI: `10.1016/j.ejor.2012.11.029`.

[28] W.J. Kolen, Antoon, Karel Lenstra, Jan, H. Papadimitriou, Christos, and Spieksma, Frits. Interval scheduling: A survey. *Naval Research Logistics*, 54:530 – 543, 08 2007. DOI: `10.1002/nav.20231`.

Received 26th March 2019

Acta Cybernetica **24** (2020) 663–678.

Pixel Grouping of Digital Images for Reversible Data Hiding

Sultan A Hasib[a b] and Hussain Nyeem[a c]

Abstract

Pixel Grouping (PG) of digital images has been a critical consideration in the recent development of the Reversible Data Hiding (RDH) schemes. While a PG kernel can define pixel-groups with the different neighborhoods for better embedding rate-distortion performance, only the group of horizontal neighborhood pixels of size 1×3 has so far been considered. In this paper, we, therefore, construct the PG kernels of sizes 3×1, 2×3 and 3×2, and investigate their potentials to improve both the embedding capacity and the embedded image quality for a PG-based RDH scheme. A kernel of size 3×2 (or 2×3) that creates a pair of pixel-triplets (*i.e.*, two L-shaped blocks) and offers a higher possible correlation among the pixels. These kernels thus can be better utilized for improving a PG-based RDH scheme. Considering this, we develop and present an improved PG-based RDH scheme and the computational model of its key processes. Experimental results demonstrated that our proposed RDH scheme offers reasonably better embedding rate-distortion performance than the original scheme.

Keywords: pixel value ordering, reversible embedding, data hiding, prediction and sorting

1 Introduction

Multimedia data have recently witnessed a tremendous growth that continues with a broader impact on today's life-hood, society, research, and industry. Their uses have shown great promises for the spectrum of emerging applications like different distant and cooperative systems and services in the areas of medical, space, military, security, and surveillance. However, with the advances in communication technologies, their exchange over the public communication network is also raising many security concerns, including forgery, copyright violation, and privacy invasion of multimedia data [6]. To addressing these problems, Reversible Data Hiding (RDH) is being widely investigated [15, 24].

[a] Department of Electrical, Electronic and Communication Engineering (EECE), Military Institute of Science and Technology (MIST), Mirpur Cantonment, Dhaka-1216, Bangladesh

[b] E-mail: `hasib_3635@hotmail.com`, ORCID: `https://orcid.org/0000-0003-1335-0053`

[c] E-mail: `h.nyeem@eece.mist.ac.bd`, ORCID: `https://orcid.org/0000-0003-4839-5059`

DOI: 10.14232/actacyb.277104

RDH is an evolving forensic and covert-communication technology for multimedia data like digital images. An RDH scheme embeds data into a *cover* image, and the embedded data later can be extracted on-demand basis. An RDH scheme thus has two main processes: *generation* and *embedding* [14, 16]. In the *generation*, the data to be embedded in the cover image are generated and processed as per the requirements of an intended application. The *embedding*, on the other hand, deals with how and where the data are to be embedded in the cover image. The generation process thus deals with the required security properties like integrity and confidentiality, and an embedding technique controls the embedding performance of the RDH scheme.

The embedding rate-distortion criteria mainly determine the embedding performance of an RDH scheme. The *embedding rate* or *embedding capacity* measures how much data can be embedded in a cover image, and the *distortion* measures how much visual quality of the cover is compromised for embedding. Much attention in the data hiding research thus can reasonably be tracked in the development of various embedding techniques with better embedding rate-distortion performance in the last two decades [1–5, 9–13, 17–23, 25, 27].

Among different types of RDH schemes, Pixel Grouping (PG), also called Pixel Value Ordering (PVO), has shown great promises for better embedding rate-distortion performance [10, 12, 19–22] (see Sec. 2). The PG-based schemes thus have the potential to offer a higher embedding rate and lower embedding distortion (*i.e.*, better-embedded image quality). In such schemes, while pixel values are grouped and arranged in a numerical order to better utilize their correlations for improving the embedded image quality, not much attention has been paid in the computation of PG with better pixel correlation.

In this paper, we report an improved PG-based RDH scheme with better utilization of pixels' correlation in pixel grouping. We call each pixel-group an *image-block* in this paper. As will be discussed in Sec. 2, Jung's scheme [10] showed the best possible embedding rate-distortion performance so far in a minimum image-block scenario. We have investigated the case of that scheme [10] that employed image-block of size 1 × 3 and analyzed the embedding rate-distortion performance of our proposed improvement with other possible block sizes to have better pixel correlation. Notably, in a mixed (*i.e.*, combination of horizontal, vertical, and diagonal) neighborhood, pixels in an image-block remain relatively more correlated. We, therefore, construct and analyze different image-blocks in modeling a PG-based RDH scheme. Thereby, a greater possible pixels' correlation in an image-block can be, utilized in embedding for a better rate-distortion performance.

The remainder of this paper is structured as follows. The current state of the PG-based RDH schemes is reviewed in Sec. 2. We develop and present a general computational model of a PG-based RDH scheme to construct different image-blocks and to examine their effect on the embedding performance in Sec. 3 and analyze the experimental results in Sec. 4. Conclusions are given in Sec. 5.

2 State of RDH schemes

Development of reversible embedding techniques has underpinned different RDH schemes; for example, Difference Expansion (DE) schemes [1,9], Histogram Shifting (HS) schemes [13, 23], Reversible Contrast Matching (RCM) schemes [2, 5], and Prediction Error Expansion (PEE) schemes [3, 4, 10–12, 19–22, 25]. Among them, PEE-based embedding combined the potential of HS and DE techniques to utilize the image redundancy better. Embedding distortion in PEE highly depends on the prediction-error histogram, where a sharp distribution of the histogram offers lower embedding distortion. A better *predictor* is thus always desirable in PEE to obtain a sharper histogram [4].

Additionally, the sorting of prediction errors has been another consideration for improving the performance of PEE-based embedding [23]. Of the sorted prediction errors, the lower values are used for embedding to minimize distortion in the embedded image. Li *et al.* [11] reported that a higher embedding rate with lower distortion is obtainable by embedding in the prediction-errors with lower complexities. Coatrieux *et al.* [3] proposed an adaptive embedding technique that determines the most suitable carrier-class according to its local specificity for data embedding. For better embedding rate-distortion performance, a PEE-based RDH scheme, therefore, aims to utilize correlations of the pixels in an image-block.

The PG technique has lately better utilized the image correlations in PEE-based embedding. Unlike the classical PEE, the PG-based PEE predicts a pixel that has a higher correlation to the original pixels in an image-block. Li *et al.* [12] introduced the PG-based RDH scheme that either increases (or decreases) or keeps unchanged the maximum (or minimum) pixel in a block for embedding 1-bit data. That scheme was later improved with the consideration of dual maximum (or minimum) pixels for prediction errors [21], adaptive prediction of maximum (or minimum) valued pixel [20], pixel-wise PVO [22] and 2D-PVO with pairs of prediction errors [19].

Recently, Jung [10] proposed a scheme that operates on the image-blocks of three pixels, where two successive blocks do not share any pixel. For embedding in each block, its pixel-values are sorted in ascending order to compute the maximum and minimum prediction errors from the maximum and minimum pixels in the block, respectively. That scheme offers better-embedded image quality with reasonably higher embedding capacity.

However, like the other aforementioned PG-based RDH schemes, computation of image-blocks with higher pixels correlation has not been considered. The better utilization of pixels correlation may lead to further improvement of the scheme with better rate-distortion performance. This consideration leads us to investigate different structures of image-blocks for PG-based embedding. Our preliminary results were presented in the conference proceedings [7, 8] that have been extended in this paper with the substantial revision of the model, analysis with more details, and new results.

3 An improved PG-based RDH scheme

In this section, we develop and present a computational model of the PG-based RDH scheme that generally captures the principle of both Jung's scheme [10] and our proposed modifications. The improved PG process is briefly introduced below, followed by our generalized embedding and extraction processes.

3.1 Proposed pixel grouping

A PG-based embedding utilizes image correlations to improve the embedding rate-distortion performance, as mentioned in Sec. 1. The embedding of the Jung's scheme computes the unit prediction error in an image-block of size 1 × 3, which restricts the block-pixels' correlations to only the horizontal context. Thus, redefining an image-block with both the horizontal and vertical contexts may further improve the embedding rate-distortion performance.

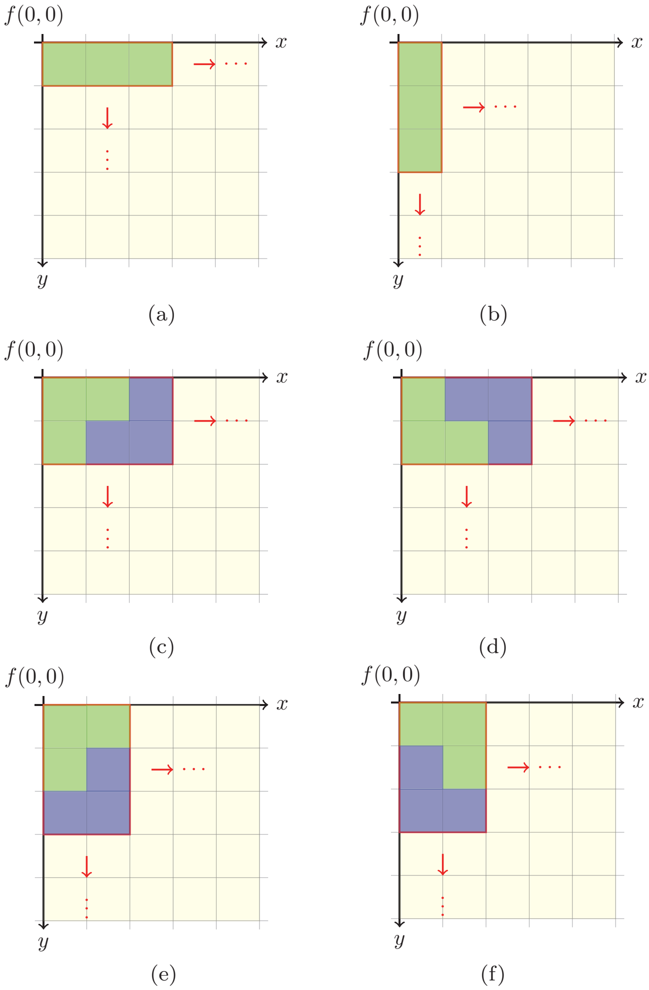
We thus have investigated the embedding performance of the PG-based RDH scheme for different structures of the image-blocks. Unlike the image-blocks used in the Jung's RDH scheme [10], we have employed the other possible structures of an image-block to determine the improvements in the embedding performance. The image-blocks of size 3 × 1 for vertical orientation and the blocks of size 2 × 3 and 3 × 2 for the mixed (*e.g.*, horizontal, vertical, and diagonal) orientations are considered. The image-blocks of both the sizes, 3 × 2 and 2 × 3 give a pair of pixel-triplets (*i.e.*, two L-shaped blocks). The construction of two L-shaped blocks illustrated in Fig. 1(*c–f*) (with the green and blue colors) from a block of 6-pixels means that each L-shaped block is of a fixed size of 3 pixels. These blocks of 3 pixels can be used as the other blocks of 3 pixels like in Fig. 1(*a* and *b*) for embedding. Note that Jung [10] used the structure in Fig. 1(*a*), and the others in Fig. 1(*b–f*) are studied for the proposed PG-based embedding.

Construction of the structures of an image-block shown in Fig. 1 can be abstracted with the *block* $(\cdot)$ and *de_block* $(\cdot)$ for the generalized PG-based embedding with an additional input argument σ (see Sec. 3.2). This means, for Jung's scheme, $\sigma = [1, 3]$ defines a block of size 1 × 3, and for the proposed embedding, $\sigma = [3, 1], [3, 2]$ and $[2, 3]$ define an image-block of size 3 × 1, 3 × 2 and 2 × 3, respectively. With a suitable σ, a PG-based embedding would have more correlated pixels in an image-block to offer better rate-distortion performance.

3.2 PG-based embedding

Let an image, I of size $M \times N$ is to be given as input (or *cover*) image and used for the embedding of secret-data D. The embedding process follows the following steps to output the embedded image I'. As in Algorithm 1, steps of the embedding are discussed below.

Step 1: A set of image-blocks, B is first obtained from an input image, I for a given block-size σ such that $B = \{B_n\}$, where B_n is a set three pixels of the n-th



(a) (b) (c) (d) (e) (f)

Figure 1: Structures of an image-block of 3-pixels for PG-based RDH scheme: (a) 3×1, (b) 1×3, (c, d) 2×3, and (e, f) 3×2.

block. This processing is abstracted with the function, *block* $(\cdot)$. That is, $B_n = \{b_n^i, b_n^{i+1}, b_n^{i+2}\}$ with $i \in \{1, 2, \cdots, M \times N\}$ for $n \in \{1, 2, \cdots, \frac{M \times N}{3}\}$.

Step 2: A set of sorted image-blocks, $P = \{P_n\}$ is obtained by sorting the pixel-values of each image-block, B_n. For example, a sorted image-block, P_n is obtained by applying the sorting function *sort* $(\cdot)$ block-wise for each B_n. That is, $P_n = \{p_n^i, p_n^{i+1}, p_n^{i+2}\}$, where $p_n^i \leq p_n^{i+1} \leq p_n^{i+2}$.

Step 3: A set of predicted errors E_n is obtained for each sorted block P_n using the function *predict* $(\cdot)$. That is, for each P_n, predicted error

Algorithm 1 PVO Embedding

```
Input: image $I$, block-size $\sigma$, and payload $D$
Output: embedded image $I'$
$\{B_n\} \leftarrow block\,(I, \sigma)$        ▷ $n$ is total no. of blocks
for all $B_n$ do
    $P_n \leftarrow sort\,(B_n)$
    $E_n \leftarrow predict\,(P_n)$
    $P'_n \leftarrow embed\,(P_n, E_n, \{d\})$
    $B'_n \leftarrow inverse\_sort\,(P'_n)$
end for
$I' \leftarrow de\_block\,(B'_n)$
```

$E_n = \{e_n^{max}, e_n^{min}\}$ of the n-th block is obtained using (1).

$$e_n^{max} = p_n^{i+2} - p_n^{i+1} \tag{1a}$$

$$e_n^{min} = p_n^{i} - p_n^{i+1} \tag{1b}$$

Step 4: A pair of predicted errors, e_n^{max} and e_n^{min} of an n-th block is expanded according to the secret bits, $\{d\} \in D$ or is shifted by unit value using (2) and (3) to obtain the modified errors, $\hat{e}_n^{max}$ and $\hat{e}_n^{min}$. These modified errors are then used to compute the set of estimated pixels, $P'_n = \{p_n^{'i}, p_n^{i+1}, p_n^{'i+2}\}$ using (4).

$$\hat{e}_n^{max} = \begin{cases} e_n^{max}, & \text{for } e_n^{max} = 0 \\ e_n^{max} + d, & \text{for } e_n^{max} = 1 \\ e_n^{max} + 1, & \text{for } e_n^{max} > 1 \end{cases} \tag{2}$$

$$\hat{e}_n^{min} = \begin{cases} e_n^{min}, & \text{for } e_n^{min} = 0 \\ e_n^{min} - d, & \text{for } e_n^{min} = -1 \\ e_n^{min} - 1, & \text{for } e_n^{min} < -1 \end{cases} \tag{3}$$

$$p_n^{'i+2} = p_n^{i+1} + \hat{e}_n^{max} \tag{4a}$$

$$p_n^{'i} = p_n^{i+1} + \hat{e}_n^{min} \tag{4b}$$

Step 5: The embedded pixels of each block are then relocated to their original locations using the inverse of $sort\,(\cdot)$ that we call here $inverse\_sort\,(\cdot)$.

Step 6: The embedded image-blocks are finally combined to return the complete embedded image, I'.

Algorithm 2 PVO Extraction

```
Input: embedded image $I'$
Output: original image $I$ and extracted payload $D$
1: Initialize: $D \leftarrow \varnothing$
2: $\sigma \leftarrow blocksize\,(I')$
3: $\{B'_n\} \leftarrow block\,(I', \sigma)$
   for all $B'_n$ do
4:     $P'_n \leftarrow sort\,(B'_n)$
5:     $E'_n \leftarrow predict\,(P'_n)$
6:     $(P_n, \{d\}) \leftarrow extract\,(P'_n, E'_n)$
7:     $D \leftarrow concat\,(D, \{d\})$
8:     $B_n \leftarrow inverse\_sort\,(P_n)$
   end for
9: $I \leftarrow de\_block\,(B_n)$
```

3.3 PG-based extraction

PG-based extraction follows the inverse processing of embedding (see Algorithm 2). This algorithm takes the embedded image, I' and block-size, σ as inputs to return the original image, I and extracted data, D. Key steps of this algorithm are briefly discussed below.

Step 1: The extracted payload, D is initialized with an empty array, $\varnothing$.

Step 2: The size of the embedded image-blocks, σ is extracted from I' using *blocksize* $(\cdot)$.

Step 3: A set of image-blocks, $B' = \{B'_n\}$ is obtained from the embedded image, I' using the same function, *block* $(\cdot)$, and σ used in embedding, where B'_n is the n-th image-block of three pixels.

Step 4: A set of sorted image-blocks, P' is obtained from B'. This means that the n-th embedded image-block, P'_n is obtained by the block-wise sorting function *sort* $(\cdot)$ for each B'_n such that $P'_n = \{p_n^{'i}, p_n^{'i+1}, p_n^{'i+2}\}$, where $p_n^{'i} \leq p_n^{'i+1} \leq p_n^{'i+2}$.

Step 5: For each sorted image-block, $P'_n \in P'$, the function *predict* $(\cdot)$ outputs a set of predicted errors, $E'_n = \{\hat{e}_n^{max}, \hat{e}_n^{min}\}$ using (5).

$$\hat{e}_n^{max} = p_n^{'i+2} - p_n^{'i+1} \tag{5a}$$

$$\hat{e}_n^{min} = p_n^{'i} - p_n^{'i+1} \tag{5b}$$

Step 6: From each embedded block, P'_n, the embedded bits, $\{d\}$ are extracted, and the pair of embedded/expanded predicted errors, $\hat{e}_n^{max}$ and $\hat{e}_n^{min}$ are computed using (6). These errors are then used to compute the originally sorted image-block, $P_n = \{p_n^i, p_n^{i+1}, p_n^{i+2}\}$ using (7). We note that this extraction function is computationally inverse of the embedding function such that $\mathit{extract}\,(\cdot) = \mathit{embed}^{-1}\,(\cdot)$.

$$d = \begin{cases} \hat{e}_n^{max} - 1, & \text{for } 1 \leq \hat{e}_n^{max} \leq 2 \\ -\hat{e}_n^{min} - 1, & \text{for } -2 \leq \hat{e}_n^{min} \leq -1 \end{cases} \tag{6}$$

$$p_n^{i+2} = \begin{cases} p_n^{'i+2}, & \text{for } \hat{e}_n^{max} = 0 \\ p_n^{'i+2} - d, & \text{for } 1 \leq \hat{e}_n^{max} \leq 2 \\ p_n^{'i+2} - 1, & \text{for } \hat{e}_n^{max} > 2 \end{cases} \tag{7a}$$

$$p_n^{i+1} = p_n^{'i+1} \tag{7b}$$

$$p_n^{i} = \begin{cases} p_n^{'i}, & \text{for } \hat{e}_n^{min} = 0 \\ p_n^{'i} + d, & \text{for } -2 \leq \hat{e}_n^{min} \leq -1 \\ p_n^{'i} + 1, & \text{for } \hat{e}_n^{min} < -2 \end{cases} \tag{7c}$$

Step 7: The extracted bits, $\{d\}$ from each embedded image-block is then concatenated with D, which was initialized as an empty array in Step 1.

Step 8: The pixels in each sorted image-block, P_n are relocated to their original locations to obtain the image-block, B_n.

Step 9: Each image-block, B_n is then combined using the function, *de_block* $(\cdot)$ to obtain the original image, I.

4 Experimental results

The performance of the proposed PG-based RDH scheme has been evaluated and compared with Jung's PG-based scheme [10]. The USC-SIPI test-images [26] of size $256 \times 256 \times 8$ have been used for this performance evaluation. The embedding-capacity and embedding-rate have been determined in terms of the total embedded bits and bit-per-pixels (*bpp*), respectively. For embedding, a set of pseudo-random bits is generated as D. The proposed scheme is implemented using MATLAB R2016b.

Additionally, the embedded image quality has been determined in terms of two popular objective visual quality metrics, *peak signal to noise ratio* (PSNR) defined in (8) and *structural similarity* (SSIM) [28] defined in (9). Here, $M \times N$ is the image size, and $I(i,j)$ and $I'(i,j)$ are the pixel-values of the location (i,j) in an original image and its embedded version, respectively. In (9), μ_x and μ'_x are the average-values of x and x', where $x \in I$ and $x' \in I'$ are the pixels of original and embedded images, respectively. Similarly, σ_x^2 and $\sigma_{x'}^2$ are the variances of x and

x', respectively; $\sigma_{xx'}$ is the covariance of x and x'; c_1 and c_2 are two regularization constants, and L is the dynamic range of the pixel values.

$$\text{MSE} = \frac{\sum_{j=1}^{N}\sum_{i=1}^{M}\left(I'(i,j) - I(i,j)\right)^2}{MN} \tag{8a}$$

$$\text{PSNR} = 10\log\frac{L^2}{\text{MSE}} \tag{8b}$$

$$\text{SSIM} = \frac{(2\mu_x\mu_{x'} + c_1)(2\sigma_{x,x'} + c_2)}{(\mu_x^2 + \mu_{x'}^2 + c_1)(\sigma_x^2 + \sigma_{x'}^2 + c_2)} \tag{9}$$

A better embedding rate-distortion performance has been observed for PVO embedding with L-shaped image-blocks. The pixel-correlations in an image-block thus can be better utilized in PG-based embedding with blocks of size 2×3 or 3×2, resulting in better embedding rate-distortion performance, as illustrated in Table 1. In other words, the room for embedding more bits with the complex image-blocks is mainly resulting from the increasing possibility of expanding the required predicted errors for data-bit embedding as defined with the middle-cases of (2) and (3) in Sec. 3.2, which is attained in the cases of L-shaped image-blocks. For example, the total embedding capacity of Jung's Scheme is 44992 bits (or 0.1716 bpp) for Airplane image, which is increased to 46547 *bits*, 46612 *bits*, and 46762 *bits* (or 0.1776 *bpp*, 0.1778 *bpp*, and 0.1784 *bpp*) for the image-blocks of sizes 3×1, 2×3, and 3×2 of the proposed schemes, respectively.

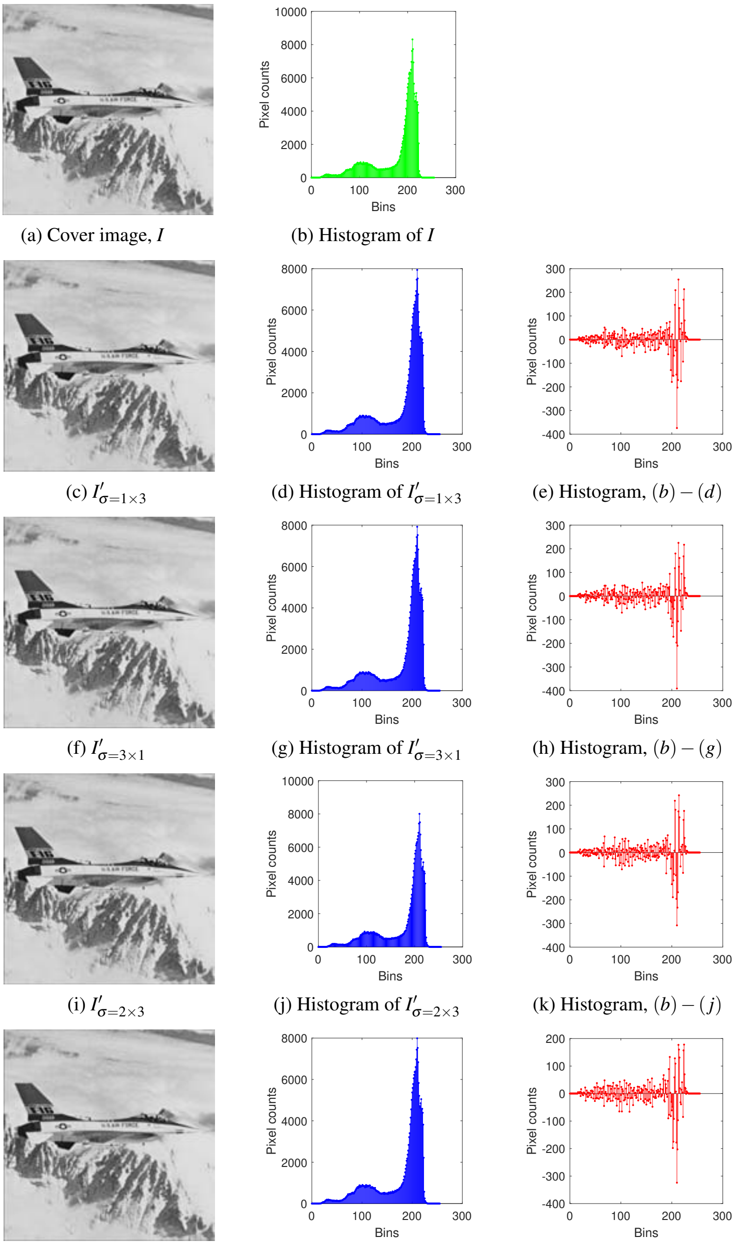
Additionally, the visual quality of the embedded images has remained at a similar level, as evident in Table 1 and Table 2.improved embedding capacity also For example, the PSNR and SSIM values of Airplane embedded images are 51.576 *dB* and 0.9759, respectively. In contrast, the proposed embedding with 3×1, 2×3, and 3×2 offered the PSNR and SSIM values of 51.617 *dB* and 0.9756, 51.639 *dB* and 0.9760, and 51.629 *dB* and 0.9759, respectively. We have observed that, while the performance of the proposed scheme with 3×1 block-size slightly improves over the Jung's scheme, this improvement becomes more noticeable for the other proposed block-sizes (*i.e.*, 2×3 and 3×2). This is because these image-blocks capture pixels in the horizontal, vertical, and diagonal directions to be more correlated than the image-block of size 3×1 (proposed) and 1×3 (Jung's)).

Despite the improvement in the embedding rate, the proposed scheme retains similar intensity distribution of the cover image. The histograms of the cover image and its embedded versions with different values of σ are illustrated in Fig. 2–3. The difference between the cover and any embedded image can hardly be perceived; however, the differences of respective histograms illustrate the changes made in the intensity distribution of the cover image (see the *third-column* from the *left* in Fig. 2–3). Such trivial visual changes remain unnoticeable, as also suggested by the absolute-difference images on the *right-most* columns in those figures.

The above trend of improvement also holds for the average performance of the proposed scheme. The average embedding capacity achieved with the 3×2 size

Table 1: Comparison of rate-distortion performance

Images	**Metric**	**Jung [10]**	**Ours**		
		$(\mathbf{1} \times \mathbf{3})$	$(\mathbf{3} \times \mathbf{1})$	$(\mathbf{2} \times \mathbf{3})$	$(\mathbf{3} \times \mathbf{2})$
Airfield	Capacity (bits)	27307	29414	30333	30104
	bpp	0.1042	0.1122	0.1157	0.1148
	PSNR (dB)	50.756	50.842	50.864	50.862
	SSIM	0.9941	0.9943	0.9943	0.9943
Airplane	Capacity (bits)	44992	46547	46612	46762
	bpp	0.1716	0.1776	0.1778	0.1784
	PSNR (dB)	51.576	51.617	51.639	51.629
	SSIM	0.9759	0.9756	0.9760	0.9759
Baboon	Capacity (bits)	13226	14046	14090	14087
	bpp	0.0505	0.0536	0.0537	0.0537
	PSNR (dB)	50.263	50.283	50.282	50.286
	SSIM	0.9977	0.9977	0.9977	0.9977
Boat	Capacity (bits)	26588	25521	26224	26338
	bpp	0.1014	0.0973	0.1000	0.1005
	PSNR (dB)	50.681	50.6485	50.660	50.666
	SSIM	0.9926	0.9926	0.9925	0.9925
Couple	Capacity (bits)	34494	34968	34882	34596
	bpp	0.1316	0.1334	0.1331	0.1320
	PSNR (dB)	51.016	51.008	50.996	50.985
	SSIM	0.9916	0.9915	0.9915	0.9915
Elaine	Capacity (bits)	23306	23997	24392	24304
	bpp	0.0889	0.0915	0.0930	0.0927
	PSNR (dB)	50.595	50.612	50.633	50.629
	SSIM	0.9929	0.9926	0.9928	0.9928
Goldhill	Capacity (bits)	27021	29365	28280	28573
	bpp	0.1031	0.1120	0.1079	0.1090
	PSNR (dB)	50.688	50.759	50.719	50.730
	SSIM	0.9922	0.9924	0.9923	0.9923
Peppers	Capacity (bits)	33483	31933	33423	33802
	bpp	0.1277	0.1218	0.1275	0.1289
	PSNR (dB)	50.923	50.869	50.914	50.916
	SSIM	0.9887	0.9885	0.9886	0.9886
Tiffany	Capacity (bits)	41750	38807	41864	41680
	bpp	0.1593	0.1480	0.1597	0.1590
	PSNR (dB)	51.316	51.183	51.305	51.303
	SSIM	0.9829	0.9826	0.9829	0.9829



(a) Cover image, I (b) Histogram of I

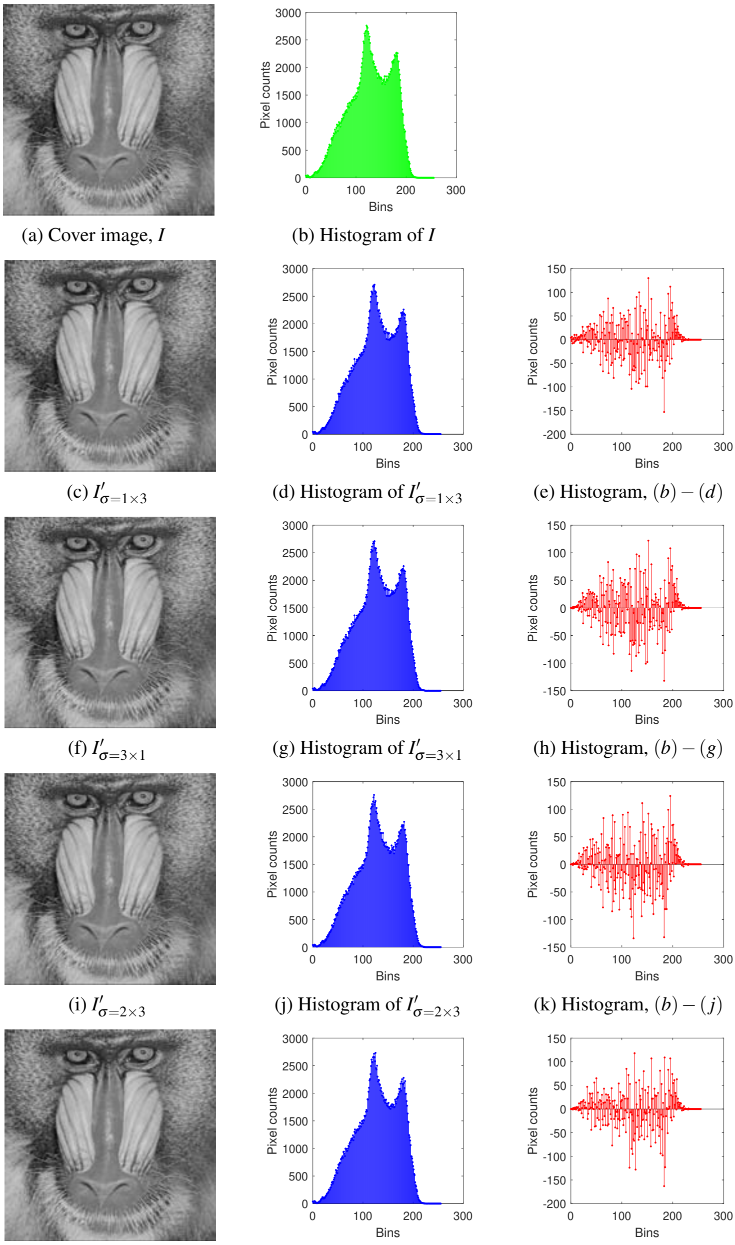
(c) $I'_{\sigma=1\times3}$ (d) Histogram of $I'_{\sigma=1\times3}$ (e) Histogram, $(b)-(d)$

(f) $I'_{\sigma=3\times1}$ (g) Histogram of $I'_{\sigma=3\times1}$ (h) Histogram, $(b)-(g)$

(i) $I'_{\sigma=2\times3}$ (j) Histogram of $I'_{\sigma=2\times3}$ (k) Histogram, $(b)-(j)$

(l) $I'_{\sigma=3\times2}$ (m) Histogram of $I'_{\sigma=3\times2}$ (n) Histogram, $(b)-(m)$

Figure 2: Comparison of the cover image and its histogram with different embedded versions and their histograms for the *Airplane* image.



(a) Cover image, I

(b) Histogram of I

(c) $I'_{\sigma=1\times3}$

(d) Histogram of $I'_{\sigma=1\times3}$

(e) Histogram, $(b)-(d)$

(f) $I'_{\sigma=3\times1}$

(g) Histogram of $I'_{\sigma=3\times1}$

(h) Histogram, $(b)-(g)$

(i) $I'_{\sigma=2\times3}$

(j) Histogram of $I'_{\sigma=2\times3}$

(k) Histogram, $(b)-(j)$

(l) $I'_{\sigma=3\times2}$

(m) Histogram of $I'_{\sigma=3\times2}$

(n) Histogram, $(b)-(m)$

Figure 3: Comparison of the cover image and its histogram with different embedded versions and their histograms for the *Baboon* image.

Table 2: Comparison of average rate-distortion performance

Metric	**Jung [10]**	**Ours**		
	(1 × 3)	**(3 × 1)**	**(2 × 3)**	**(3 × 2)**
Capacity (bits)	31223	31387	32228	32191
bpp	0.1191	0.1197	0.1229	0.1228
PSNR (dB)	50.921	50.916	50.948	50.944
SSIM	0.9891	0.9890	0.9891	0.9891

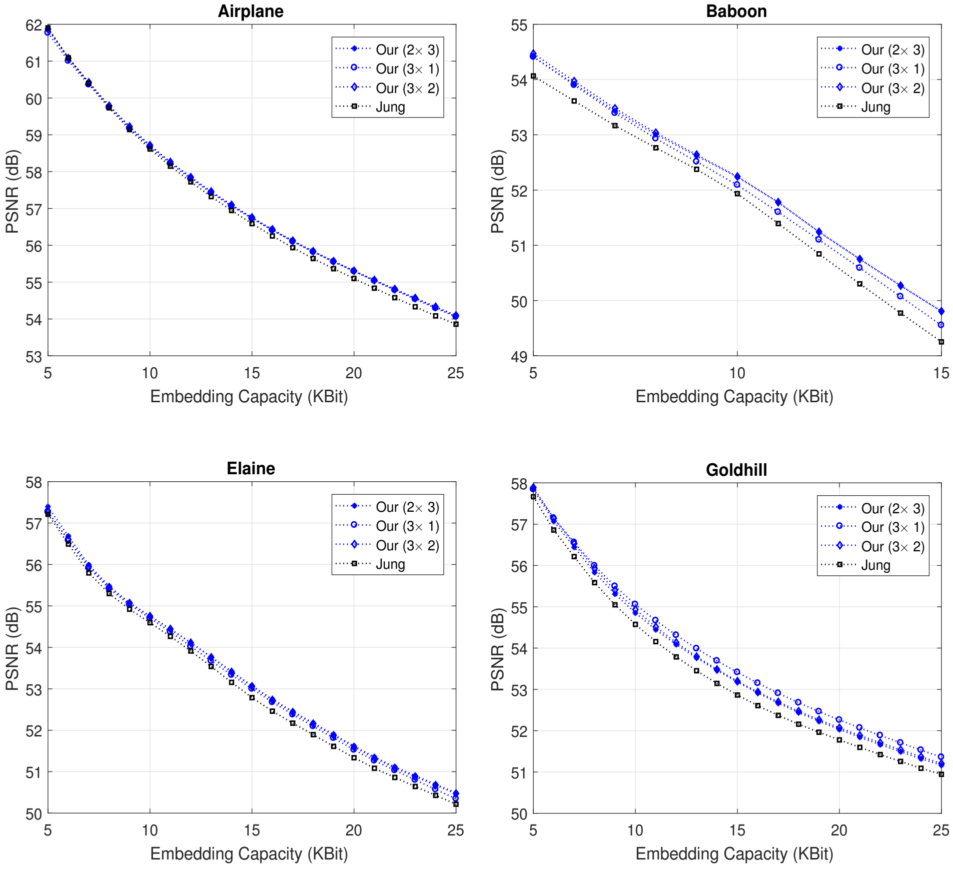


Figure 4: Embedding rate-distortion performance comparison

image-block is 32191 *bits*, and that with an image-block of size 2×3 is 32228 *bits*; whereas, the capacity is found of 31223 *bits* and 31387 *bits* for the image-blocks of size 1×3 and 3×1, respectively. This improved embedding capacity also maintains an improved average PSNR and similar SSIM values in case of the image-block of size 2×3. We note that similar improvements in the rate-distortion performance of the proposed RDH scheme also exist for the other test images we experimented with.

5 Conclusions

PG-based RDH is generalized for different image-blocks and its embedding rate-distortion performance is investigated for better utilization of block-pixels correlation. The image-blocks with different structures have been investigated for the PG-based embedding. The presented simulation and experimental results in this paper suggest that a better rate-distortion performance can be obtained with the embedding in an L-shaped image-block capturing pixels in the horizontal, vertical, and diagonal contexts. In other words, the PG-based embedding with 2×3 and 3×2 image-blocks would offer an improved rate-distortion performance compared to the other block-sizes and the Jung's scheme. This consideration of constructing image-block may also contribute to the development of PG-based RDH schemes in the future.

References

[1] Alattar, A. M. Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Transactions on Image Processing*, 13(8):1147–1156, 2004. DOI: `10.1109/TIP.2004.828418`.

[2] Chen, X., Li, X., Yang, B., and Tang, Y. Reversible image watermarking based on a generalized integer transform. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2382–2385, 2010. DOI: `10.1109/ICASSP.2010.5496175`.

[3] Coatrieux, G., Pan, W., Cuppens-Boulahia, N., Cuppens, F., and Roux, C. Reversible watermarking based on invariant image classification and dynamic histogram shifting. *IEEE Transactions on Information Forensics and Security*, 8(1):111–120, 2013. DOI: `10.1109/TIFS.2012.2224108`.

[4] Coltuc, D. Improved embedding for prediction-based reversible watermarking. *IEEE Transactions on Information Forensics and Security*, 6(3):873–882, 2011. DOI: `10.1109/TIFS.2011.2145372`.

[5] Coltuc, D. and Chassery, J. Very fast watermarking by reversible contrast mapping. *IEEE Signal Processing Letters*, 14(4):255–258, 2007. DOI: `10.1109/LSP.2006.884895`.

[6] Eskicioglu, A. M. Multimedia security in group communications: recent progress in key management, authentication, and watermarking. *Multimedia Systems*, 9(3):239–248, 2003. DOI: 10.1007/s00530-003-0095-2.

[7] Hasib, S. A. and Nyeem, H. Developing a pixel value ordering based reversible data hiding scheme. In *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*, pages 1–6, 2017. DOI: 10.1109/EICT.2017.8275160.

[8] Hasib, S. A. and Nyeem, H. Rate-distortion analysis of an improved pixel value ordering based reversible embedding. In *2017 2nd International Conference on Electrical Electronic Engineering (ICEEE)*, pages 1–4, 2017. DOI: 10.1109/CEEE.2017.8412855.

[9] Jun Tian. Reversible data embedding using a difference expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(8):890–896, 2003. DOI: 10.1109/TCSVT.2003.815962.

[10] Jung, Ki-Hyun. A high-capacity reversible data hiding scheme based on sorting and prediction in digital images. *Multimedia Tools and Applications*, 76(11):13127–13137, 2017. DOI: 10.1007/s11042-016-3739-x.

[11] Li, X., Yang, B., and Zeng, T. Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection. *IEEE Transactions on Image Processing*, 20(12):3524–3533, 2011. DOI: 10.1109/TIP.2011.2150233.

[12] Li, Xiaolong, Li, Jian, Li, Bin, and Yang, Bin. High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion. *Signal Processing*, 93(1):198 – 205, 2013. DOI: https://doi.org/10.1016/j.sigpro.2012.07.025.

[13] Ni, Zhicheng, Shi, Yun-Qing, Ansari, N., and Su, Wei. Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(3):354–362, 2006. DOI: 10.1109/TCSVT.2006.869964.

[14] Nyeem, H., Boles, W., and Boyd, C. Developing a digital image watermarking model. In *2011 Int. Conf. on Digital Image Computing: Techniques and Applications*, pages 468–473, 2011. DOI: 10.1109/DICTA.2011.85.

[15] Nyeem, H.A. *A digital watermarking framework with application to medical image security.* PhD thesis, Queensland University of Technology, 2014.

[16] Nyeem, Hussain, Boles, Wageeh, and Boyd, Colin. Digital image watermarking: its formal model, fundamental properties and possible attacks. *EURASIP Journal on Advances in Signal Processing*, 2014(1):135, 2014. DOI: 10.1186/1687-6180-2014-135.

[17] Nyeem, Hussain, Boles, Wageeh, and Boyd, Colin. Content-independent embedding scheme for multi-modal medical image watermarking. *Biomedical engineering online*, 14(1):1–19, 2015. DOI: 10.1186/1475-925X-14-7.

[18] Nyeem, Hussain, Boles, Wageeh, and Boyd, Colin. Watermarking capacity control for dynamic payload embedding. In *Recent Advances in Information and Communication Technology 2015*, pages 143–152. Springer, 2015. DOI: 10.1007/978-3-319-19024-2-15.

[19] Ou, Bo, Li, Xiaolong, and Wang, Jinwei. High-fidelity reversible data hiding based on pixel-value-ordering and pairwise prediction-error expansion. *Journal of Visual Communication and Image Representation*, 39:12–23, 2016. DOI: 10.1016/j.jvcir.2016.05.005.

[20] Ou, Bo, Li, Xiaolong, Zhao, Yao, and Ni, Rongrong. Reversible data hiding using invariant pixel-value-ordering and prediction-error expansion. *Signal processing: image communication*, 29(7):760–772, 2014. DOI: 10.1016/j.image.2014.05.003.

[21] Peng, F., Li, X., and Yang, B. Improved PVO-based reversible data hiding. *Digital Signal Processing*, 25:255–265, 2014. DOI: 10.1016/j.dsp.2013.11.002.

[22] Qu, Xiaochao and Kim, Hyoung Joong. Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding. *Signal Processing*, 111:249–260, 2015. DOI: 10.1016/j.sigpro.2015.01.002.

[23] Sachnev, V., Kim, H. J., Nam, J., Suresh, S., and Shi, Y. Q. Reversible watermarking algorithm using sorting and prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):989–999, 2009. DOI: 10.1109/TCSVT.2009.2020257.

[24] Shi, Y., Li, X., Zhang, X., Wu, H., and Ma, B. Reversible data hiding: Advances in the past two decades. *IEEE Access*, 4:3210–3237, 2016. DOI: 10.1109/ACCESS.2016.2573308.

[25] Thodi, D. M. and Rodriguez, J. J. Expansion embedding techniques for reversible watermarking. *IEEE Transactions on Image Processing*, 16(3):721–730, 2007. DOI: 10.1109/TIP.2006.891046.

[26] USC-SIPI. Image database. http://sipi.usc.edu/database/.

[27] Wahed, M. A. and Nyeem, H. Developing a block-wise interpolation based adaptive data embedding scheme. In *2016 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, pages 1–6, 2016. DOI: 10.1109/CEEICT.2016.7873152.

[28] Zhou Wang, Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. DOI: 10.1109/TIP.2003.819861.

Received 2nd October 2018

Acta Cybernetica **24** (2020) 679–697.

Semi Fragile Audio Crypto-Watermarking based on Sparse Sampling with Partially Decomposed Haar Matrix Structure

Electa Alice Jayarani Appadurai[a], Mahabaleswara Ram Bhatt[b], and Geetha D.D.[c]

Abstract

In the recent era the growth of technology is tremendous and at the same time, the misuse of the technology is also increasing with an equal scale. Thus, the owners have to protect the multimedia data from the malicious and piracy. This has led the researchers to the new era of cryptography and watermarking. In the traditional security algorithm for the audio, the algorithm is implemented on the digital data after the traditional analog to digital conversion. But in this article, we propose the crypto–watermarking algorithm based on sparse sampling to be implemented during the analog to digital conversion process only. The watermark is generated by exploiting the structure of Haar transform. The performance of the algorithm is tested on various audio signals and the obtained SNR is greater than 30dB and the algorithm results in good robustness against various signal attacks such as echo addition, noise addition, reverberation etc.

Keywords: audio, watermarking, cryptography, compressive sensing

1 Introduction

The most common and widely used security algorithm for the multimedia files is digital algorithms. The multimedia data can be the image, audio, video, text, etc. Mainly there are two ways to achieve the privacy in digital data, namely, watermarking and cryptography [11]. The digital watermarking is defined as embedding the highly decryptable watermark into the digital data without harming the content of the original host signal. Whereas in cryptography the data would be in

[a] Research Scholar, Department of Electronics and Communication, Reva University, Bangalore, India. E-mail: electalice@gmail.com, ORCID: https://orcid.org/0000-0002-5117-6917.

[b] Professor, Department of Medical Electronics, BMS College of Engineering, Bangalore, India. E-mail: bhatt.mr@rediffmail.com, ORCID: https://orcid.org/0000-0002-6921-036X.

[c] Professor, Department of Electronics and Communication, Reva University, Bangalore, India. E-mail: dgeetha@reva.edu.in, ORCID: http://orcid.org/0000-0002-7788-5615.

DOI: 10.14232/actacyb.280899

disguise form to protect its content. In other words, Cryptography converts the intelligible data into unintelligible data which appears as meaningless for attackers. By seeing the data one can tell the data is encrypted but cannot decrypt without the proper secret key. If the data is decrypted the data is no longer protected. Both the algorithm should maintain the robustness nature to protect the secret message. On the other hand, the privacy in watermarking is not strictly inevitable but in cryptography, it has to be private by definition. For example, the watermark presence on the rupee note can be easily seen by everyone against the light.

In this article, we propose the algorithm to protect the audio signals from the piracy. As the human Auditory System (HAS) is more sensitive than the Human Visual System (HVS) [11], the audio watermarking becomes a very tedious task. The audio data security has been under research for many years but still, it is falling short of safety requirements and it is vulnerable to attack, privacy and piracy. The natural audio signal that is audible by the Human ear originates from acoustic variation. These acoustic signals are converted to analog and subsequently digital data using Shannon sampling theory. The encrypted key or watermarking is carried out on the obtained digital data for protection. A large amount of research in watermarking is centered on digital techniques which are more prone to attack as shown in Fig. 1.

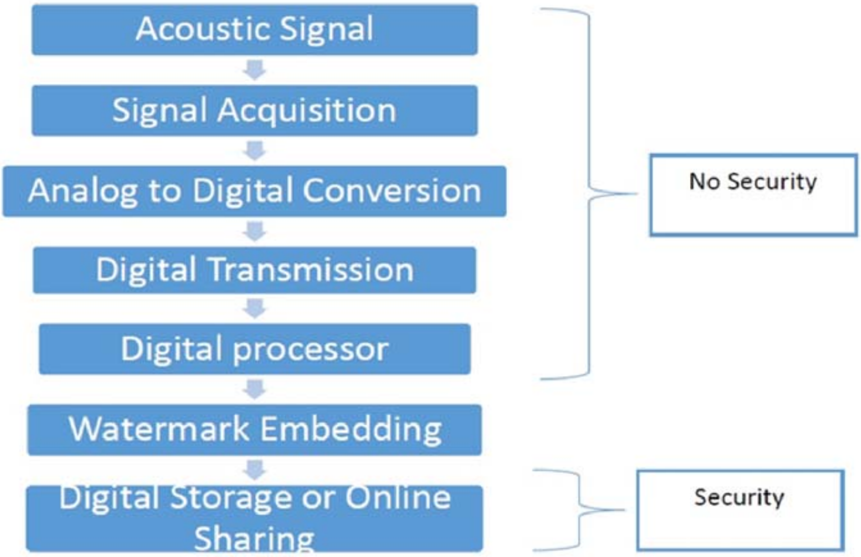


Figure 1: Existing Methods Flowchart

To overcome this problem, in this paper, embedding the crypto-watermark signature on the audio during the time of digital conversion as shown in Fig. 2 is studied and experimented.

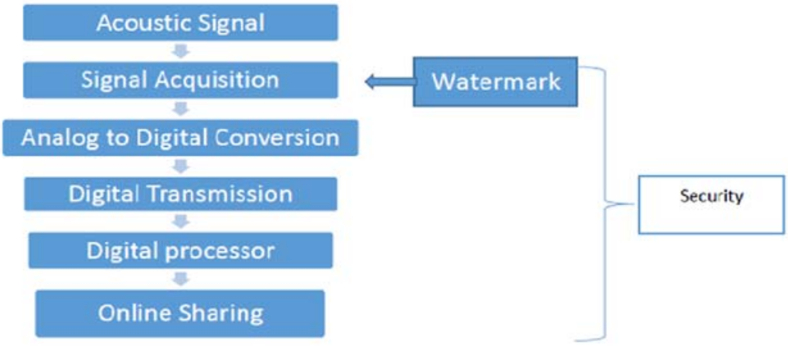


Figure 2: Proposed Algorithm Flowchart

In the past decade, there has been a paradigm shift in approaches to signal acquisition that explores and employs sparse coding or compressing sensing [3, 12, 13, 14]. By compressive sensing the audio with the watermark, the data is referred as 'digital information data' instead of typical digital audio data, which precludes from direct conversion to analog audio unless the audio can be recovered using mathematical programming techniques only. The advent of this technique is to embed the watermark with the secret key at the time of digital to analog conversion without altering the perceptual quality of the audio signal. By using only, the mathematical programming technique the audio can be converted and can be played using a transducer.

2 Existing methods

In the past years, various research had been undergone to protect the ownership of audio files and a various algorithm is developed based on Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Empirical Mode Decomposition (EMD), etc. In [6], Guo et al. propose a transform domain watermarking algorithm. By altering the DCT coefficient the watermark is embedded into the host and the algorithms average Signal to Noise Ratio (SNR) reaches up to 20dB. A novel audio watermarking algorithm based on the randon transformnumber and DWT was defined by Cairong Li et al. [10]. A new adaptive audio watermarking algorithm based on Empirical Mode Decomposition is introduced by Khaldi and Boudraa [9] and the average SNR reaches up to 25dB. In [1], the author attempts to implement a baseline audio watermarking system that embeds the information by modulating the phase in Weighted Overlap-Add Algorithm (WOLA). The algorithm gives SNR values from 0 to 25dB. In [7] blind audio watermarking is proposed based on a combination of Discrete Wavelet Packet Transformation (DWPT), Singular Value Decomposition (SVD) and Quantization Index modulation (QIM). The author Fallahpour and Megias [4] venture an innovative method of embedding the audio watermark. The Fibonacci series is used to select the FFT samples of the host signal to embed the watermark. In all the methods the acoustic signal is converted to digital data using traditional analog to digital conversion (ADC) and the algorithms are implemented on the digital data.

3 Block diagram

The general digital audio watermarking process is shown in Fig. 3. From the performer through the microphone the audio signal is transmitted to the processor where the signal is converted into digital and watermark embedding is done. The watermarked data can be transmitted or can be stored digitally. At the receiver side, the signal is converted into audio and played through the speaker. Thus, the algorithms cannot be used for the live audio concert.

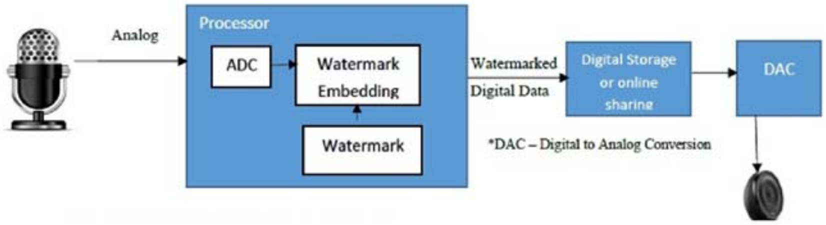


Figure 3: General Digital Audio Watermarking process

To overcome this problem in this paper we propose a compressive sensing based crypto–watermarking algorithm to be implemented during the process of ADC only. The general block diagram of the compressive sensing based crypto-watermarking algorithm for audio is shown in Fig. 4.

Here we propose a customized microphone where the watermark is embedded in the time of signal acquisition and the watermarked digital data can be transmitted or can be stored digitally. The analog data can be recovered only by the customized speaker where the security key and watermark are embedded. The traditional speaker cannot retrieve the data. The customized microphone and speaker block diagram are shown in Fig. 5.

4 Compressive sensing and its role for audio security

Essentially, the compressive sampling (CS) is a method of converting the analog signal into a digital information with sparse. This non-uniform sampling yields fewer sample data, which can be used to recover the signal using a mathematical

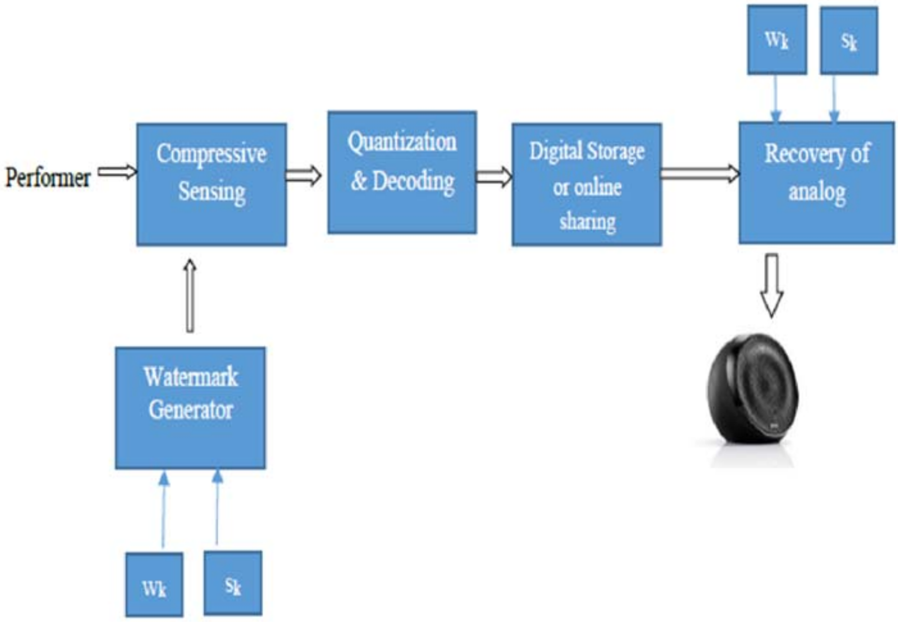


Figure 4: Crypto–Watermarking Block Diagram

convex programming. This is in contrast to the conventional analog to digital conversion technique which exploits digital filtering technique based on Shannon uniform sampling principle.

Let $x \in R^n$ be a one dimensional (1-D) original audio signal and the signal is considered as K-sparse or K non-zero entries. The transform matrix vector representation with the orthonormal basis matrix $\Psi \in R^{n \times n}$ is $X = \Psi x$ with x is a K-sparse signal.

The method of obtaining linear measurement data vector $y \in R^m$ from an incoherent sampling or sensing matrix $\phi \in R^{mxn}$ (m $\ll$ n) is expressed as $y = \phi \Psi x$.

On denoting matrix $\Theta = \phi \Psi$ as compressive sensing process we get

$$y = \Theta x \ . \tag{1}$$

By finding solutions to an underdetermined linear system of equation (1), the original signal can be reconstructed. In underdetermined linear system, the system has infinite number of solutions and more unknowns than the equations. Most common methods to solve the sparse approximation are Basis Pursuit and Orthogonal Matching Pursuit methods. In basis Pursuit method, the sparse approximation problem can be replaced as convex problem, hence the same is used for the recovery in the proposed method.

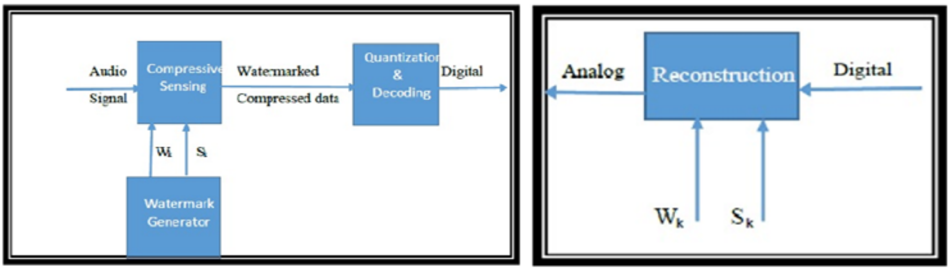


Figure 5: (a) Customized Microphone Block Diagram (left) (b) Customized Speaker (right)

The sparse problem in Basis Pursuit is given as

$$\min(\|x\|_0) \text{ subject to } y = \Theta x \ , \tag{2}$$

where $y \in R^m$ is the measured vector, ϕ is the $m \times n$ matrix and $x \in R^n$ is the vector to be recovered. In the equation (2), the norm-0, $||.||_0$ is non-convex and difficult to solve. It is an NP-hard (Non-deterministic Polynomial-time hardness) problem. Therefore, it is replaced with l_1-norm and it is given as

$$\min(\|x\|_1) \text{ subject to } y = \Theta x \ . \tag{3}$$

It can be recast as Linear Programming problem (LP) and is given as

$$\begin{aligned} \min f^T x \text{ subject to } & y = \Theta x \\ & x \geq 0 \ , \end{aligned}$$

where $f^T x$ is the objective function, $y = \Theta x$ is collection of equality constraint and $x \geq 0$ is set of bounds. By adding new variable, the nonlinearity is recast to the set of constraints and it is given as

$$\begin{gathered} \min \sum_{i-1}^{n} U_i \text{ subject to } -u \leq x \leq u \\ y = \Theta x \end{gathered}$$

Or it can be written as

$$\begin{gathered} \min \sum_{i-1}^{n} U_i \text{ subject to } -x_i - u_i \leq 0, \ i = 1, 2, \ldots, n \\ x_i - u_i \leq 0, i = 1, 2, \ldots, n \\ y = \Theta x \end{gathered} \tag{4}$$

There are many algorithms to solve the basis pursuit problem such as simplex method and primal–dual interior point method. For high accuracy, the primal dual method is used with Newton method combined with modified KKT (Karush-Kuhn-Tucker) condition for search criteria.

For example, consider $n = 2$,

$$\begin{aligned} f_{u1} &= x_1 - u_1 \\ f_{u2} &= -x_1 - u_1 \end{aligned}$$

And the corresponding dual variable is considered as λ_1 and λ_2 and given as

$$\begin{aligned} \lambda_1 &= -\frac{1}{f_{u1}} \\ \lambda_2 &= -\frac{1}{f_{u2}} \end{aligned}$$

The modified KKT condition for the residual $r_t = (x, \lambda, \upsilon)$ is given as

$$\begin{aligned} \nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \Theta^T \upsilon = 0 \\ -\lambda_i f_i(x) = \frac{1}{t} \quad i = 1, 2, \ldots, m \\ \Theta x = y \end{aligned}$$

For $t > 0$, it is given as

$$r_t\left(x, \lambda, \upsilon\right) = \begin{pmatrix} \triangledown f_0\left(x\right) + Df\left(x\right)^T \lambda + \Theta^T \upsilon \\ -\mathrm{diag}\left(\lambda\right) f\left(x\right) - \frac{1}{\tau} 1 \\ \Theta x - y \end{pmatrix} \tag{5}$$

where $f : R^n \to R^m$ and the matrix Df is its derivative

$$f\left(x\right) = \begin{pmatrix} f_1\left(x\right) \\ \vdots \\ f_m\left(x\right) \end{pmatrix} \text{ and } Df\left(x\right) = \begin{pmatrix} \triangledown f_1\left(x\right)^T \\ \vdots \\ \triangledown f_m\left(x\right)^T \end{pmatrix}.$$

If x, λ, υ and $r_t(x, \lambda, \upsilon) = 0$, then $x = x^*(t)$, $\lambda = \lambda^*(t)$ and $\upsilon = {}^*(t)$. x is primal feasible, and λ, ν are dual feasible. The duality gap is $\tau = \frac{m}{t}$.

The first term of equation (5) is called dual residual, 2nd term is called centrality residual and 3rd term is primal residual. For a fixed time t, at a point (x, λ, υ) that satisfies $f(x) < 0, \lambda > 0$ the Newton's step is used to solve $r_t(x, \lambda, \upsilon) = 0$.

$$y = (x, \lambda, \upsilon), \ \triangle y = (\triangle x, \triangle \lambda, \triangle \upsilon)$$

$$\begin{pmatrix} \triangledown^2 f_0\left(x\right) + \sum_{i=1}^{m} \lambda_i \triangledown^2 f_i\left(x\right) & Df\left(x\right)^T & \Theta^T \upsilon \\ -diag\left(\lambda\right) Df\left(x\right) & -diag\left(f\left(x\right)\right) & 0 \\ \Theta & 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta \lambda \\ \Delta \upsilon \end{pmatrix} = -\begin{pmatrix} r_{dual} \\ r_{cent} \\ r_{pri} \end{pmatrix} \tag{6}$$

The solution of equation (6) will be the primal dual search direction. For the primal-dual interior point method, we use the surrogate duality gap. For any x that satisfies $f(x) < 0$, $\lambda \geq 0$ it is defined as $\eta(x, \lambda) = -f(x)^T\lambda$.

If x is a primal feasible, and λ, υ are dual feasible, which means $r_{pri} = 0$ and $r_{dual} = 0$ then the surrogate gap will be the duality gap. In general, the steps to compute the optimal solution is as follows. The inputs are a point x that satisfies $f(x) < 0, \lambda > 0, \mu > 1\varepsilon_{feas} > 0, \varepsilon > 0$.

1. Set $t = \frac{\mu m}{\eta}$.
2. Compute primal dual search direction using equation (6).
3. We determine the step length $s > 0$ and compute $y = y + s\Delta y$ until $\|r_{pri}\|_2 \leq \varepsilon_{feas}$, $\|r_{dual}\|_2 \leq \varepsilon_{feas}$, and $\eta \leq \varepsilon$.
4. For the implementation, the step length is chosen in the range of $0 < s \leq 1$. The step length tracking is started with $s = 0.99 \cdot \min\{1, \min\frac{-\lambda_i}{\triangle\lambda_i} \mid \triangle\lambda_i < 0\}$, $i = 1, 2, \ldots, m$. Multiply the s by $\beta \in (0, 1)$ until we have $\|r_\tau(x + s \triangle x, \lambda + s \triangle \lambda, \upsilon + \triangle\upsilon\|_2 \leq (1 - \alpha s).\|r_\tau(x, \lambda, \upsilon)\|_2$ where α is set as 0.01.
5. Continue the steps until the optimal value of x is found.

5 Haar transform and its orthogonal property

Haar Transform is the simplest and the fastest wavelet transform. The Haar function is denoted as $h_k(x)$ and will fall in the closed interval of $[0, 1]$. Whereas the k is the order of the function and it is decomposed into two parameter such as $k = 2^p + q - 1$, $k = 0, 1, \ldots, N - 1$ where $N = 2^n$, $0 \leq p \leq n - 1$, $0 \leq q \leq 2^p$.

The Haar function is defined as

$$h_0(x) \equiv h_{00}(x) = \frac{1}{\sqrt{N}}, \quad x \in [0, 1]$$

and

$$h_k(x) \equiv h_{pq}(x) = \frac{1}{\sqrt{N}} \begin{cases} 2^{\frac{p}{z}} \frac{q-1}{2^p} \leq x < \frac{q-0.5}{2^p} \\ -2^{\frac{p}{z}} \frac{q-0.5}{2^p} \leq x < \frac{q}{2^p} \\ \qquad\qquad\qquad 0 \quad \text{otherwise.} \end{cases}$$

The amplitude and the width of the function which involves the value other than zero is given by p and position of the non-zero value is given by q. The Haar transform matrix for the $N = 2$ is given below:

$$H_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 2 \\ 1 & -1 \end{bmatrix}.$$

It is observed that $H = H^*$ and $H^{-1} = H^T$ therefore $H^T H = I$ where I is the identity matrix.

6 Privacy preserving crypto-watermarking technique

Typically, the intent of both cryptology and watermarking is to add a signature into the data to make it secure from an unintended audience and to maintain privacy and authenticity while communicating through the unsecured channels with robustness to attacks. But the significant difference is that in cryptology, both data and signature are invisible, whereas in watermarking the data could be visible but signature may or may not be visible. The current exploration has both the features which we refer to as crypto-watermarking technique.

For our algorithm, we create a matrix U which can be a unitary matrix or permutation matrix since both has very interesting properties. For example, let's consider U as a unitary matrix and considering the property of unitary matrix

$$UU^T = U^T U = I \ . \tag{7}$$

Applying the equation (7) to (1) we get

$$y = \Theta x = \phi \Psi x = (U\phi^T)^T (U\Psi)x \tag{8}$$

or

$$y = \Theta x = \phi \Psi x = (U^T \phi^T)^T (U^T \Psi)x \ \ . \tag{9}$$

Note here the matrix x is the segmented audio frame of the original host signal. Based on the above relationship we now formulate the sensing matrix and the transform matrix by using either equation (8) or (9). By calculating the scaling factor, the equation (7) can be rewritten as

$$UU^T = U^T U = \frac{1}{n} I \ \ . \tag{10}$$

Therefore, we can view y as

$$y = \Theta x = \phi \Psi x = (U\phi^T)^T (U\Psi)x$$

$$y = \Theta x = \phi \Psi x = (U^T \phi^T)^T (U^T \Psi)x \ \ .$$

7 Proposed algorithm

Generation of K-sparse signal

The original host signal is divided into frames and the input audio sequence from an audio frame is $x \in R^n$ (e.g., Figure 6) with K sparse.

Process of generating a watermark signature

1. Consider U=H, a Haar matrix.

2. Form $H = Q_1 Q_2 Q_3 \cdots Q_j R$ where Q_j is an orthogonal matrix and R is an upper triangular matrix which in turn is non-orthogonal matrix.

3. Perform the various signal function on Q to generate a watermark and is given as $W = \text{signalfunction}_i(Q)$ where the signal function can be circular shift, addition, etc. on the decomposed orthogonal matrix without affecting the orthogonal property. The signal function and "i" times is considered as extra security key (S_k).

4. The generated watermark is considered as watermark key (W_k).

Process of embedding watermark signature in compressive sensed data

Let us consider equation (8).

1. Decompose the Haar matrix and generate the watermark key and secret key.

2. Obtain the shuffled audio matrix as $X = (U\Psi)x$.

3. Obtain $A = (U\phi^T)^T$.

4. Obtain watermarked data matrix as $Y = AX$.

Process of Recovery of Signal from compressive signal

In order to recover the signal from equation (1), the primal dual interior method is used. The recovery algorithm explained in section 4 is implemented in MATLAB and the signal is recovered. Depends on the length of the audio signal, the number of iterations varies. Table 1 lists the number of iterations for the different audio files.

Table 1: Number of iterations

Audio file	**Number of iterations**	**Duration(sec)**
Guitar	9	16.52
Flute	14	37.47
Bass	21	46.53

8 Results and discussion

In this section, we concentrate on the audio quality aspects arising due to compressed sensing that exploits various k-sparse audio data. Subsequently, we demonstrate and highlight a few experimental results of the proposed semi-fragile audio crypto-watermarking based on compressed sensing while acquiring audio clips and also audio data recovery processes. The experiment involves schemes such as crypto-watermarking signature generation, embedding the watermark signature and l_1 recovery algorithm for the recovery of the signal. And we have compared the quality assessment of the audio recovery using the proposed algorithm with and without watermarking signatures. The proposed algorithm is implemented using MATLAB 2016 in Intel Core i5 processor.

Generating K-sparse data for experimentation

A set of 10 source audio clips are chosen for the experiment. All the clips are mono-channel with less than 60 seconds duration sampled with 44.1 KHz having audio data width as 8 bits. All the audio clips generated includes solo musical instruments like violin, guitar, piano, flute, equinox, bass, Handel, track, Mary Song, Backstreet boys song, Crazy Frog - Axel F, Emilie big world, and different frequency clips. Table 2 lists the different audio clips names, duration, length and the sampling frequency.

Table 2: Experimented audio file's details

Audio	**Length**	**Duration**	**Sampling Frequency (Hz)**
bass	525200	11 s	44100
Guitar	90309	2 s	44100
Piano	409101	9 s	44100
Handel	73113	8s	8192
violin	305172	6s	44100
flute	346724	7s	44100
tone	384000	8s	48000
Mary	319725	7s	44100
Backstreet boys	1323000	30s	44100
Emilie big world	1323000	30s	44100
Irish Whistel	1323000	30s	44100
100Hz	220500	5s	44100
250Hz	220500	5s	44100
440Hz	220500	5s	44100
1KHz	220500	5s	44100

For better implementation, the source signal is reduced to frames with the samples of 256 for each frame. Many natural signals are pithy when it is expressed in an appropriate basis. The example is shown in below Fig. 6(a) of the source signal and its transform in Fig. 6(b).

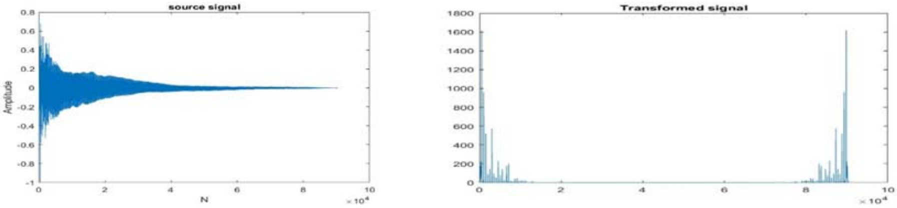


Figure 6: (a) Source Signal (b) Transformed signal

Based on observation, it is evident that the most coefficients are very small and negligible and at the same time only a few coefficients would comprise of a significant amount of information. Hence compressive sensing exploits this sparse nature of the signal. For simplicity, in this article, we would like to generate the sparse signals which are obtained by utilizing the transformed signal using pseudo-random sequence generator. The uniformly distributed random numbers are selected according to our frame size and using that the K-sparse signal is generated and only the nonzero K values are considered. Different K values are taken for the test and the results are quite similar to any value of K, whether it is less K or greater K.

Considering $x \in R^n$ and the transform coefficient is K-sparse then the measurement m of the basis matrix is selected by generating a random vector uniformly. It is shown in [5, 2] K-sparse vector x can be reconstructed from $y = Ax$ using l_1 minimization provided

$$m \geq CK\ln\frac{n}{K} \tag{11}$$

where $C > 0$ is a universal constant independent of $K, n, m.$ In equation (11), m is directly proportional to K and hence if the sparsity is considered small then the measurement m can also be chosen small in comparison with nso that the solution of an underdetermined system of linear equation is reasonable. Different sparse K signal and the corresponding m measurement by considering $C = 0$ are listed in Table 3.

Recovered signal

The reconstructing can be performed only by the customized speaker which is embedded with the secret key and the security key as shown in Fig. 5(b). The compressed watermarked signal reaches the speaker where the programming recovery takes place using the l_1 minimization with w_k and s_k and the optimum value is obtained by primal dual sparse approximation algorithm. The recovered signal is shown in Fig. 7.

Table 3: Different K and m

S.No	**K**	**m**
1.	3	$\geq$14
2.	5	$\geq$20
3.	7	$\geq$25
4.	10	$\geq$32
5.	13	$\geq$39
6.	15	$\geq$43
7.	20	$\geq$51

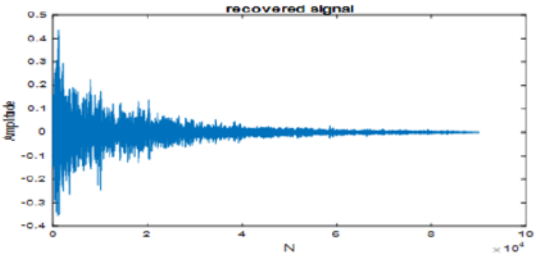


Figure 7: Recovered signal

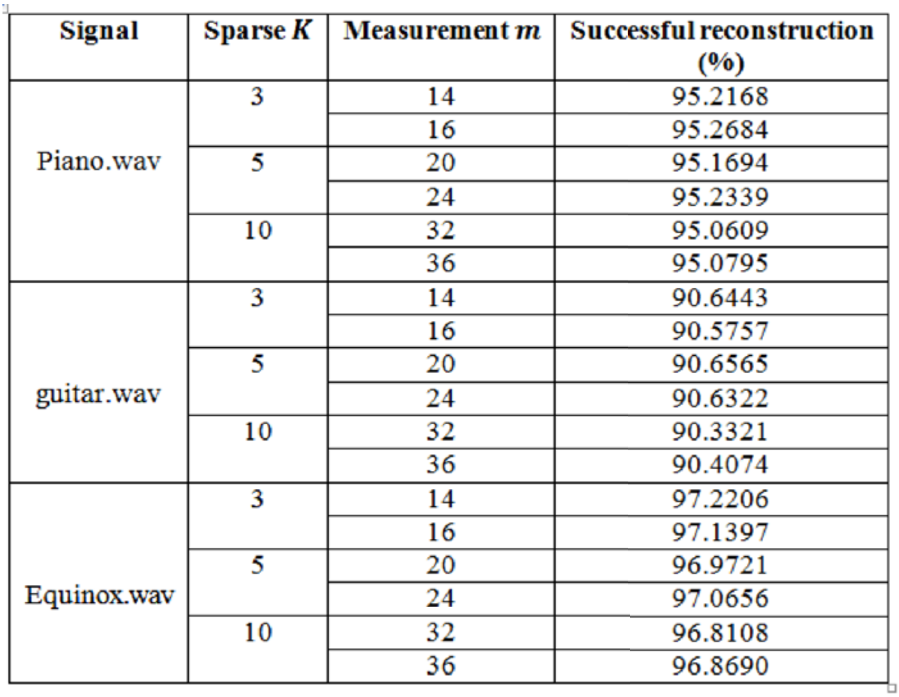
For our experiment we have tested different instrumental audio data such as piano, guitar etc. and the results are listed below in Table 4. The proposed algorithm takes approximately 2ms to perform a crypto watermarking on an audio of length of 256 samples and takes approximately 0.1s to reconstruct the host signal using the security key and watermark key. Further, the above proposed algorithm is tested on various audio album songs such as Backstreet boys, Emilie Big world and observed that the success rate is around 80%, which yields a good efficiency with a reduced delay for embedding and reconstructing the signal.

9 Imperceptibility

Imperceptibility is the parameter used to measure the perceptual quality of the original audio after embedding the watermark data into it. The objective parameter to measure the imperceptibility is Signal-to-Noise ratio (SNR) and Objective Difference Grade (ODG). The SNR is a measurement that compares the similarity between the undisturbed host signal and the watermarked host signal. The SNR is calculated as

$$SNR = -10\log_{10}\frac{\sum_{i=1}^{n}(Y-Y')^2}{\sum_{i=1}^{n}(Y)^2}dB \tag{12}$$

Table 4: Test Results



Signal	Sparse K	Measurement m	Successful reconstruction (%)
Piano.wav	3	14	95.2168
		16	95.2684
	5	20	95.1694
		24	95.2339
	10	32	95.0609
		36	95.0795
guitar.wav	3	14	90.6443
		16	90.5757
	5	20	90.6565
		24	90.6322
	10	32	90.3321
		36	90.4074
Equinox.wav	3	14	97.2206
		16	97.1397
	5	20	96.9721
		24	97.0656
	10	32	96.8108
		36	96.8690

where Y is the compressive sensed data without embedding a watermark signature and Y' is the compressive sensed data by embedding the watermark signature.

We have used the kabal [8], PEAQ Basic Model to evaluate the Perceptual Evaluation Audio Quality where $ODG = 0$ means no impairment whereas $ODG = -4$ means it's very annoying. It is observed that the obtained ODG is less than -1.9 which shows the fair perceptual quality of audio.

As the final judgment of the perceptual quality of audio has to be made by the HumanAuditory System (HAS) we have experimented with the subjective quality measurement test also. For the test, we have selected four participants and asked them to grade the dissimilarity between the original host and the recovered signal. The Subjective Difference Grade (SDG) is reported by the participants where $SDG = 5$ means no dissimilar and $SDG = 0$ means totally dissimilar. It is observed that the obtained SDG is greater than four which shows the good perceptual quality of the audio signal. Table 5 shows the SNR, ODG, and SDG of the different audio signals.

Table 5: Imperceptibility measurement

Audio	**SNR**	**ODG**	**SDG**
Piano	32.38	-1.131	> 4
Guitar	32.96	-1.126	> 4
Handel	31.2	-1.889	> 4.5
Bass	31.31	-1.9	> 4.5
440Hz	34.3	-1.32	> 4
1kHz	31.82	-1.2	> 4

10 Robustness

To verify the robustness of the proposed method the following attacks are performed.

a. Amplitude Modification
The amplitude of the watermarked signal is modified by ±6% whereas the positive and negative scale is boosting off the amplitude and cutting off the amplitude respectively.

b. Echo Addition
An echo with a delay of 350ms and echo level of 85% is added to the watermarked audio signal.

c. Filtering
Different filtering such as Low Pass Filter, High Pass Filter, Band Pass Filter and Band Stop Filter with different cut off frequency is applied to the watermarked audio signal.

d. Reverberation
Big room reverberation with a reverberation time of 1000ms is exerted on the watermarked audio signal.

e. Resampling
The watermarked audio is downsampled 22050 Hz and upsampled back to source sampling frequency of 44100Hz.

f. MP3 Compression
The watermarked audio signal is compressed to a bit rate of 16kbps and decompressed back to .wav format.

g. Noise addition
White Gaussian Noise is added to the watermarked audio signal.

To measure the robustness, the commonly used parameters are Normalized Correlation (NC) and Bit Error Rate (BER). The Normalized Correlation (NC) is defined as

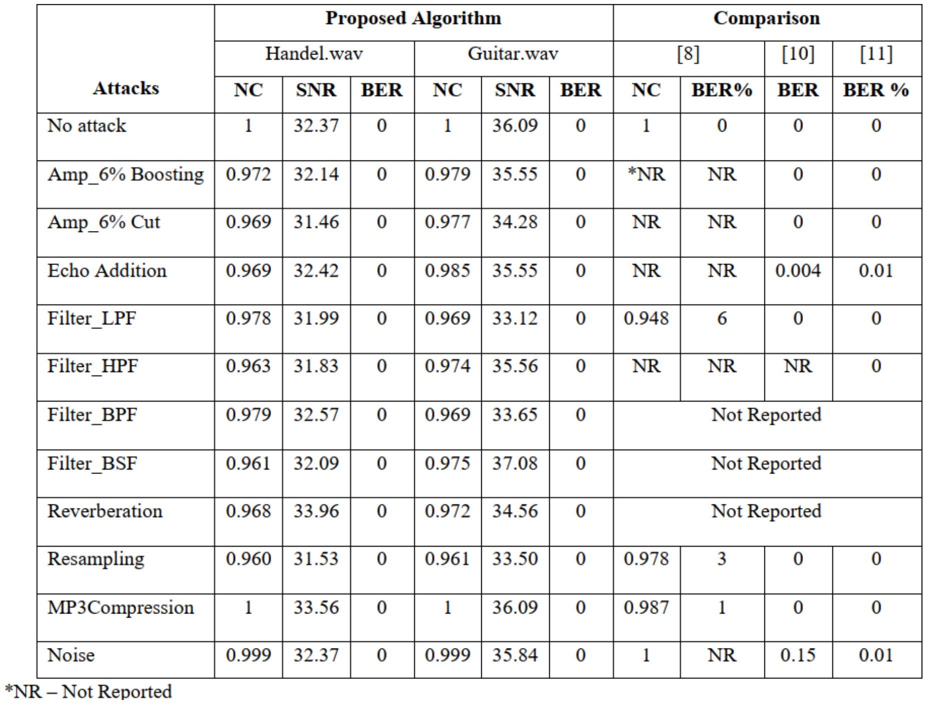
$$NC = \frac{x^*\tilde{x}}{\sqrt{x^2}\sqrt{\tilde{x}^2}} \ . \tag{13}$$

The Bit Error Rate (BER) is defined as

$$BER = \frac{x^*\tilde{x}}{n} \tag{14}$$

where x is the recovered signal without any attacks, $\tilde{x}$ is the recovered signal with an attack, and n is the length of the signal. Table 6 shows the NC and BER for the audio files of `Handel.wav` and `guitar.wav`. If $NC = 1$ means the algorithm is high robustness to attacks whereas if $NC = 0$ means the algorithm is fragile to attacks. It can be observed from the table, the proposed algorithm is possessing the nature of high robustness as NC is greater than 0.96 and BER of zero for all cases. The Robustness comparison of the proposed algorithm with the other existing watermarking algorithm is also shown in Table 6.

Table 6: Robustness Test Results of Proposed algorithm and Comparison of Robustness with other watermarking algorithms



Attacks	Proposed Algorithm						Comparison			
	Handel.wav			Guitar.wav			[8]		[10]	[11]
	NC	SNR	BER	NC	SNR	BER	NC	BER%	BER	BER %
No attack	1	32.37	0	1	36.09	0	1	0	0	0
Amp_6% Boosting	0.972	32.14	0	0.979	35.55	0	*NR	NR	0	0
Amp_6% Cut	0.969	31.46	0	0.977	34.28	0	NR	NR	0	0
Echo Addition	0.969	32.42	0	0.985	35.55	0	NR	NR	0.004	0.01
Filter_LPF	0.978	31.99	0	0.969	33.12	0	0.948	6	0	0
Filter_HPF	0.963	31.83	0	0.974	35.56	0	NR	NR	NR	0
Filter_BPF	0.979	32.57	0	0.969	33.65	0	Not Reported			
Filter_BSF	0.961	32.09	0	0.975	37.08	0	Not Reported			
Reverberation	0.968	33.96	0	0.972	34.56	0	Not Reported			
Resampling	0.960	31.53	0	0.961	33.50	0	0.978	3	0	0
MP3Compression	1	33.56	0	1	36.09	0	0.987	1	0	0
Noise	0.999	32.37	0	0.999	35.84	0	1	NR	0.15	0.01

*NR – Not Reported

11 Comparison

The proposed algorithm in this article is compared with the recent audio watermarking scheme. Each algorithm uses different properties and we have chosen SNR, ODG and SDG values as the comparisonparameter with our proposed algorithm. All the compared algorithms, embed the watermark in the digital data and reported SNR values is greater than 20 dB whereas the reported ODG is less than -2. Comparing with the other methods, our method proposes a high SNR which is greater than 31dB. As we use the crypto watermarking at the time of ADC, the ODG values observed is fair compared with the other method. We can make a convenient tradeoff in this case as the watermark is embedded at the time of signal acquisition. Table 7 shows the comparison of a different watermarking algorithm.

Table 7: Comparison with other Watermarking Algorithm

Algorithm	**SNR (dB)**	**ODG**	**SDG**
Guo et al. (2012)	20	Not reported	Not reported
Cairong Li et al. (2012)	22.35 to 27.35	Not reported	Not reported
Khaldi and Boudraa (2013)	24.12 to 26.38	0.4 to -0.6	Not reported
Arnold et al. (2014)	0 to 25	-0.42	-1.07
Hu et al. (2014)	20.889	-0.062	Not reported
Fallahpour, Megias (2015)	35 to 61	-0.3 to -1.1	> 3.5
Proposed Algorithm	31.2 to 34.3	-1.1 to -1.9	> 4

12 Conclusion

The proposed crypto-watermarking algorithm is based on compressive sensing and by exploiting a partially decomposed Haar matrix, the watermark is generated. The results show the SNR is above 30dB which shows that the perceptual quality of the audio is not degraded in the name of increasing the security. The security of the audio is more as the watermark and security key are embedded into the host audio signal at the time of signal acquisition only. Hence the proposed algorithm can be utilized for real-time application and can be used to protect the original audio from illegal copying. The results of the robustness shows that the NC is close to unity and BER is zero and therefore the algorithm is highly robust against various signal attacks such as noise addition, echo addition, reverberation, etc. Hence the proposed algorithm can be used to embed a watermark in a live concert and protect the data by providing the security key.

References

[1] Arnold, Michael, Chen, Xiao-Ming, Baum, Peter, Gries, Ulrich, and Doerr, Gwenael. A phase-based audio watermarking system robust to acoustic path propagation. *IEEE Transactions on Information Forensics and Security*, 9(3):411–425, 2014. DOI: `10.1109/TIFS.2013.2293952`.

[2] Candes, E.J. and Wakin, M.B. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, 2008. DOI: `10.1109/MSP.2007.914731`.

[3] Candes, Emmanuel, Romberg, Justin, and Tao, Terence. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006. DOI: `10.1002/cpa.20124`.

[4] Fallahpour, Mehdi and Megias, David. Audio watermarking based on Fibonacci numbers. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(8):1273–1282, 2015. DOI: `10.1109/TASLP.2015.2430818`.

[5] Foucart, Simon and Rauhut, Holger. *A Mathematical Introduction to Compressive Sensing*. Springer Science+Business Media, Birkhäuser, New York, NY, 2013. DOI: `10.1007/978-0-8176-4948-7`.

[6] Guo, Qijun, Zhao, Yanbin, Cheng, Pingpan, and Wang, Fengming. An audio digital watermarking algorithm against A/D and D/A conversions based on DCT domain. In *Proceedings of the 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pages 871–876, Yichang, China, 2012. IEEE. DOI: `10.1109/CECNet.2012.6201522`.

[7] Hu, Hwai-Tsu, Chou, Hsien-Hsin, Yu, Chu, and Hsu, Ling-Yuan. Incorporation of perceptually adaptive QIM with singular value decomposition for blind audio watermarking. *EURASIP Journal on Advances in Signal Processing*, 2014. DOI: `10.1186/1687-6180-2014-12`.

[8] Kabal, P. An examination and interpretation of ITU-R BS.1387: Perceptual evaluation of audio quality. Technical report, McGill University, 2002.

[9] Khaldi, Kais and Boudraa, Abdel-Ouahab. Audio watermarking via EMD. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(3):675–680, 2013. DOI: `10.1109/TASL.2012.2227733`.

[10] Li, Cairong, Hu, Ruimin, and Zeng, Wei. Radon transform and DWT based audio watermarking algorithm against DA/AD conversion. In *Proceedings of the International Conference on Audio, Language and Image Processing*, pages 282–286, Shanghai, China, 2012. IEEE. DOI: `10.1109/ICALIP.2012.6376626`.

[11] Lin, Yiqing and Abdulla, Waleed H. *Audio Watermark: A Comprehensive Foundation Using MATLAB*. Springer International Publishing, Switzerland, 2015. DOI: `10.1007/978-3-319-07974-5`.

[12] Mishali, M., Eldar, Y.C., Dounaevsky, O., and Shoshan, E. Xampling: Analog to digital at sub-Nyquist rates. *IET Circuits, Devices & Systems*, 5(1):8–20, 2010. DOI: `10.1049/iet-cds.2010.0147`.

[13] Qi, Jin, Hu, Xiaoxuan, Ma, Yun, and Sun, Yanfei. A hybrid security and compressive sensing-based sensor data gathering scheme. *IEEE Access*, 3:718–724, 2015. DOI: `10.1109/ACCESS.2015.2439034`.

[14] Selesnick, Ivan. Introduction to sparsity in signal processing, 2012. Connexions Web site, `http://cnx.org/content/m43545/`.

Received 5th March 2019

Contents

ISSN 0324—721 X (Print)
ISSN 2676—993 X (Online)

Editor-in-Chief: Tibor Csendes