CNN-based steganalysis and parametric adversarial embedding:A game-theoretic framework

(Article begins on next page)

01 April 2023

CNN-based Steganalysis and Parametric Adversarial Embedding: a Game-Theoretic Framework☆

Xiaoyu Shi[a], Benedetta Tondi[b], Bin Li[a,c,*], Mauro Barni[b]

[a]*Guangdong Key Laboratory of Intelligent Information Processing and Shenzhen Key Laboratory of Media Security, Shenzhen University, Shenzhen 518060, China*
[b]*Department of Information Engineering and Mathematics, University of Siena, Siena 53100, Italy*
[c]*Peng Cheng Laboratory, Shenzhen 518052, China*

Abstract

CNN-based steganalysis has recently achieved very good performance in detecting content-adaptive steganography. At the same time, recent works have shown that, by adopting an approach similar to that used to build adversarial examples, a steganographer can adopt an adversarial embedding strategy to effectively counter a target CNN steganalyzer. In turn, the good performance of the steganalyzer can be restored by retraining the CNN with adversarial stego images. A problem with this model is that, arguably, at training time the steganalizer is not aware of the exact parameters used by the steganograher for adversarial embedding and, vice versa, the steganographer does not know how the images that will be used to train the steganalyzer are generated. In order to exit this apparent deadlock, we introduce a game theoretic framework wherein the problem of setting the parameters of the steganalyzer and the steganographer is solved in a strategic way. More specifically, a non-zero sum game is first formulated to model the problem, and then instantiated by considering a specific adversarial embedding scheme setting its operating parameters in a game-theoretic fashion. Our analysis shows that the equilibrium solution of the non zero-sum game can be conveniently found by solving an associated zero-sum game, thus reducing greatly the complexity of the problem. Then we run several experiments to derive the optimum strategies for the steganographer and the staganalyst in a game-theoretic sense, and to evaluate the performance of the game at the equilibrium, characterizing the loss with respect to the conventional non-adversarial case. Eventually, by leveraging on the analysis of the equilibrium point of the game, we introduce a new strategy to improve the reliability of the steganalysis, which shows the benefits of addressing the security issue in a game-theoretic perspective.

Keywords: Adversarial embedding, deep learning, steganography, steganalysis, game theory
2010 MSC: 00-01, 99-00

1. Introduction

As a popular technique in multimedia security, image steganography [1, 2] tries to conceal a secret message in a cover image by slightly modifying pixel values or DCT coefficients. Accompanying the development of image steganography, steganalysis aims at detecting the presence of hidden information within an image. The two techniques are applied by two competing players in a hunting and escaping game.

Nowadays, mainstream steganographic schemes are designed to be content-adaptive under the framework of distortion minimization [3]. Different schemes are designed by properly choosing the distortion function. For example, HUGO (Highly Undetectable steGO) [4] defines the distortion function according to the impact that data embedding has on SPAM (Subtractive Pixel Adjacency Matrix) [5] features. WOW (Wavelet Obtained Weights) [6] assigns distortion costs by using three wavelet directional filters. S-UNIWARD (Spatial Universal Wavelet Relative Distortion) [7] is a slightly modified version of WOW, which can be easily extended to work with JPEG images. HILL (High-pass, Low-pass, and Low-pass) [8] employs a high-pass and two low-pass filters to make sure that pixels within textured regions have relatively low costs. Other schemes are designed by minimizing the difference between cover and stego images characterized by a statistical model, such as MG (Multivariate Gaussian) [9], MVGG (Multivariate Generalized Gaussian) [10], MiPOD (Minimizing the Power of Optimal detector) [11], and MRG (Multivariate Gaussian for Residuals) [12]. Some useful methods such as MDS (Modification Direction Synchronization) [13, 14] and CPP (Controversial Pixels Prior) [15] can be used to further exploit non-additive distortion.

Steganalysis has also made substantial progress in this competition. The most common approach to counter content-adaptive steganographic schemes consists in the analysis of high dimensional feature vectors [16, 17, 18, 19, 20]. The use of SRM (Spatial Rich Model) features [16] exploiting high-order pixel dependency and resulting in a feature space with tens of thousands dimensions, is the most representative example of this approach. Other methods exploit the so called "selection-channel" information. For instance ASRM (Adaptive Spatial Rich Models)[17] and maxSRM [18], put more emphasis on the regions that are more likely modified by the steganographer.

Motivated by the recent success of deep-learning in image processing and computer vision applications, deep-learning-based steganalytic methods relying on Convolution Neural Networks (CNNs) have been explored. Tan-net [21] was the first deep learning method for steganalysis based on auto-encoders. Later, Qian et al. [22] proposed a CNN equipped with Gaussian activation functions and high-pass pre-processing filters. A breakthrough in this direction has recently been achieved by Xu et al.[23]. Their proposed CNN, called Xu-net, is a new structure explicitly designed for image steganalysis, which considered several advanced CNN methods for image classification tasks, such as batch normalization (BN) [24], Tanh activation function, and 1×1 convolution in deeper layer to enhance the strength of modeling. Ye-net [25] further improved the performance of CNN-based steganalysis by incorporating selection-channel information and applying truncated linear unit (TLU) in the first few layers to accelerate network-training convergence. Recently, an end-to-end deep residual architecture, called SR-net [26], has been proposed, which minimizes the use of heuristics and hand-crafted components and works well for steganalysis in both spatial and JPEG domains.

1.1. Motivation

CNN-based steganalysis can provide better performance with respect to standard machine-learning (ML) methods. However, as shown by many recent works in the general literature of deep learning, CNNs are vulnerable to so called adversarial examples [27]: slight, often imperceptible, perturbations of the input, which are sufficient to induce a wrong decision. The concept of adversarial examples has been already successfully exploited in steganography, to counter a target CNN steganalyzer [28, 29, 30]. In [30], in particular, an adversarial steganographic scheme is developed by adjusting the embedding costs based on the back-propagated gradient of the target CNN steganalyzer.[1]

Compared to [28, 29], the adversarial stego images generated through the scheme in [30] are less detectable by standard feature based steganalyzers, the rate of modification being only slightly higher than with conventional stego schemes. When the steganalyst is aware of the presence of the adversarial steganographer, robustness against attacks can be improved by adversary-aware training [31], that is, by re-training the steganalyzer also with adversarial stego samples. On the other hand, the steganographer may anticipate the countermeasures adopted by the aware steganalyst and refine the steganographic scheme to prevent steganalysis. A problem with this framework is that, arguably, while training the adversary-aware steganalyzer, the steganalyst does not know the exact internal parameters used by the steganograher for adversarial embedding. In the same way, the steganographer does not know the exact kind of images that are used by the steganalyst to train the steganalyzer and hence he can not tune the internal parameters of the steganographic scheme to maximize its deception capability. A pictorial representation of such a dilemma is given in Figure 1.

In this paper, we propose to exit this apparent deadlock by resorting to *game theory* (GT). In particular, we introduce a game-theoretic framework according to which each contender of this race of arms, sets the internal parameters of its algorithm trying to strategically anticipate the choice of the other player.

1.2. Prior Art on Game Theory in Related Security Areas

The use of game theory to model the interplay between steganography and stenaganalysis has been explored in several works. In [32], for instance, game theory is used to find the best strategy for a steganographer who can spread the secret message over several homogeneous cover media (batch steganography), and for a steganalyst who anticipates this and tries to detect the existence of at least one secret message (pooled steganalysis). Other interesting game-theoretical approaches have been proposed recently in the field of content-adaptive steganography. Content-adaptive steganographic schemes embed the stego-message in the locations of the cover medium where the changes are harder to detect [33]. Schöttle et al. [34] have drawn the attention to the fact that, if the steganalyzer behaves in a strategic manner (and then can recalculate the adaptivity criterion), adaptive embedding schemes risk to be less secure than random embedding. The authors provide a rigorous approach to secure content-adaptive steganography by means of a game-theoretic model: the defender and the attacker must decide in which position to hide and look for evidence of embedding, respectively, by taking into account the opponent's action. Using the notion of Nash equilibrium, an optimal adaptive embedding strategy which maximizes the security against a strategic detector is identified in a simple case. The model has been later extended in [35]. In [36], the same approach is applied to the case of a Gaussian cover and embedding changes based on LSB matching. Game theory has also been used in many other contiguous security-related fields, e.g., in watermarking [37] and multimedia forensics [38, 39]. A game-theoretic framework to account for the presence of adversaries in general binary detection problems has been studied in [40, 41]. The game-theoretic approach followed in this paper is similar to the one adopted in [42], where the problem of data fusion in the presence of malicious nodes is studied.

1.3. Method and Contribution

In this paper, we assume that the steganographer adopts the adversarial embedding strategy (ADV-EMB) adopted in [30], targeting the Xu-net spatial steganalyzer [23], while the steganalyst trains an adversary aware version of Xu-net. In doing so, the steganographer must set the parameter β of the ADV-EMB algorithm. Such a parameter states the fraction of *so called* adjustable elements, that is, those elements (be them pixels or DCT coefficients) that are modified by the algorithm in such a way that the targeted steganalyzer makes a wrong decision. In [30], this parameter is minimized in order to reduce the embedding distortion. However, for a given prescribed payload, the steganographer may want to optimize β in such a way to reduce

---

[1]Throughout the paper we refer to the person aimed at detecting stego signals as steganalyst. The term steganalyzer is used when we refer to the classifier/network implementing the staganalysis.
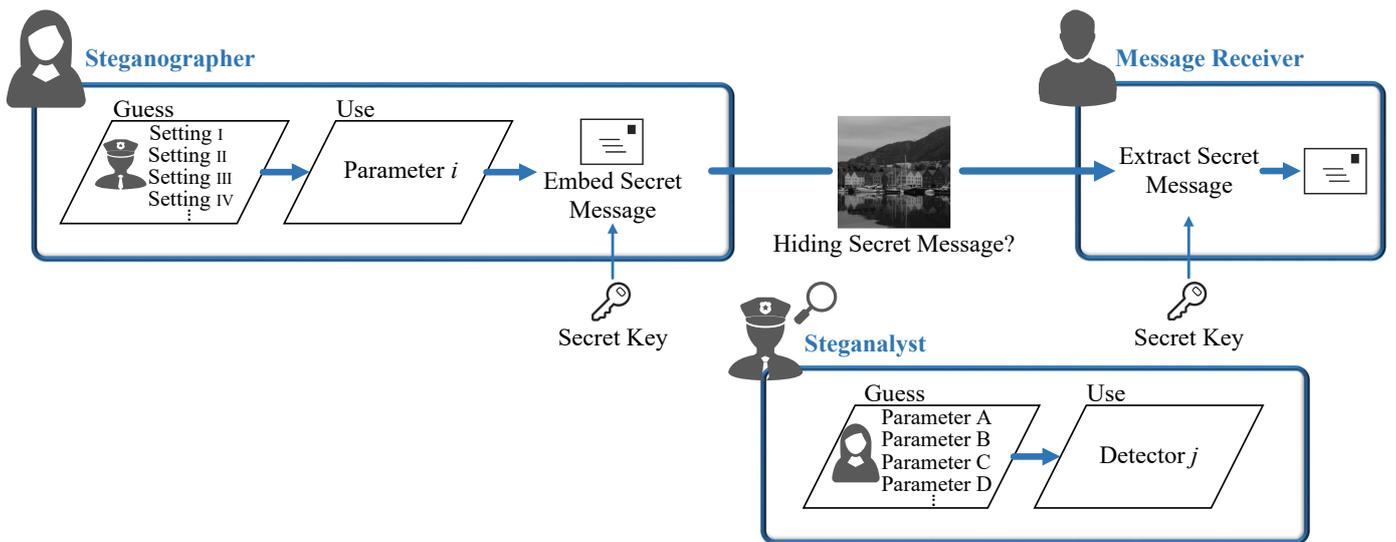
Figure 1: Scheme of the adversary-aware stego embedding/detection problem considered in this paper. The steganographer sets the internal parameters of its steganographic scheme by assuming that the steganalst will adopt an adversary aware training strategy, but without knowing the setting that he will use to do that. On the other side the steganalyst implements an adversary aware detector by trying to guess the internal parameters used by the steganographer.

the probability that the stego-image is detected by a CNN-based detector. In turn, the steganalyst knows the ADV-EMB scheme adopted by the steganographer and then he trains the detection network in an adversary-aware fashion. However, the steganalyst does not know the exact value of the parameter β adopted by the ADV-EMB algorithm and then he has to make an *educated guess*. On his side, the steganographer must determine the best β without knowing the value used by the CNN to train the adversary-aware steganalyzer. The core of our work is the proposal of a game-theoretic approach to strategically choose the values of β used by the steganographer (for embedding) and the steganalyst (for adversary-aware training) [43]. By adopting a game-theoretic terminology, the *optimum* choice for the two players, represents the equilibrium point of the game (usually, the Nash equilibrium solution is considered [44]), and the evaluation of the performance at the equilibrium permits to assess which contender will gain an advantage over the other in the steganographer vs steganalyst struggle. To the best of our knowledge, this work is the first one that investigates the interplay between CNN-based steganalysis and adversarial attacks from a GT perspective.

Throughout the paper, we will refer to the steganalyst as the defender (D) and to the steganographer as the attacker (A). The main contributions of our work are reported in the following.

- We formulate a *non zero-sum game* [45] that models the interplay between D, whose goal is to minimize the overall error probability of detecting correctly cover and stego images, and A, who wants to maximize the probability that the stego images are not detected as such, i.e maximize the missed detection error probability.[2]

- We prove that, in order to study the equilibria of the game, we can conveniently solve an associated zero-sum game, where the payoff of the game is defined as the overall error probability, thus greatly simplifying the resolution of the game [3].

- We apply the proposed game-theoretic framework to a specific instantiation of the game, when the plain embedding scheme ADV-EMB steganography relies on corresponds to S-UNIWARD [7]. We derive the optimum strategies for the steganographer and the steganalyst and evaluate the corresponding payoff. Eventually, we show experimentally that an improved solution for the staganalyst can be obtained by considering the distribution of β at the equilibrium of the game, and use it to perform aware training.

The behavior of the optimum strategy for the steganographer confirms the necessity to find a good trade-off between hiding the adversarial embedding on one hand (by using a low β), and trying to force the classifier towards a wrong decision on the other (by using a large β).

1.4. Organization

The rest of this paper is organized as follows. In Section 2, we introduce the notations and basic concepts used in this paper, and present our re-adaptation of the adversarial steganographic scheme where the embedding is controlled by a strength parameter β. In Section 3, we formulate the game between the CNN steganalyzer and the steganographer. The analysis of the equilibrium solution of the game is carried out in Section 4. The methodology of practical equilibrium assessment is detailed in Section 5. In Section 6, we experimentally

---

[2]Arguably, the steganographer does not care about the correct or incorrect classification of cover images

[3]Zero-sum games are generally easier to solve, thanks to the minimax theorem and its relationship with the linear programming duality [46, 47]

derive and discuss the optimum strategies for the steganalyser and the steganographer and the payoffs at the equilibrium under different settings. Finally, in Section 7, we draw some conclusions.

2. Technical Preliminaries

In this section, we introduce the main notations and the metrics used. We also provide a brief introduction to the ADV-EMB algorithm [30] and a description of the more general and re-adapted version considered in this paper, where the steganographer can adjust the strength of the attack by choosing the number of modifiable elements.

2.1. Notation and basic concepts

In the rest of the paper, we use bold capital letters for matrices and images, bold lowercase letters for vectors, and flourish letters for sets. We let $\mathbf{C} = (c_{i,j})^{H \times W}$, $\mathbf{S} = (s_{i,j})^{H \times W}$, and $\mathbf{Z} = (z_{i,j})^{H \times W}$ be the cover, conventional stego, and adversarial stego images, respectively, where H and W are the height and width of the image. The sets containing cover, stego, and adversarial stego images are denoted as $\mathcal{C}$, $\mathcal{S}$, and $\mathcal{Z}$, respectively.

Steganalysis can be regarded as a two-class classification problem, where we usually consider stego images as the positive class and cover images as the negative one. To build a CNN classifier, a number of data samples associated with their labels is fed for training in a supervised-learning fashion. In order to train a CNN classifier, typically, a loss function is defined and back-propagation is performed according to the gradients of the loss with respective to the learnable parameters/weights.

For the adversary-unaware scenario, we denote a CNN classifier trained on C and S as $\phi_{C,S}$. Let $\mathbf{X}$ be an input image and y be its groundtruth label, where $y = 0$ stands for cover and $y = 1$ for stego. The performance of the CNN are determined by the probability of two kinds of error, namely the false alarm probability (P_{fa}) and the missed detection probability (P_{md}), defined as:

$$P_{fa} = \Pr\{\phi_{C,S}(\mathbf{X}) = 1 | y = 0\}$$
$$= \Pr\{\phi_{C,S}(\mathbf{C}) = 1\}, \quad (1)$$

and

$$P_{md} = \Pr\{\phi_{C,S}(\mathbf{X}) = 0 | y = 1\}$$
$$= \Pr\{\phi_{C,S}(\mathbf{S}) = 0\}. \quad (2)$$

The overall performance can be evaluated by the total error probability (P_e), corresponding to (we assume that the a-priori probabilities of cover and stego images are equal):

$$P_e = \frac{P_{fa} + P_{md}}{2}. \quad (3)$$

Let $L(\mathbf{X}, y; \phi_{C,S}^{\mathbf{W}})$ be the loss function of $\phi_{C,S}$, where $\mathbf{W} = [w_{a,b}]_{(a,b)}$ indicates all the learnable parameters of the CNN. A typical cross-entropy loss function is defined as

$$L(\mathbf{X}, y; \phi_{C,S}^{\mathbf{W}}) = -y \log(\phi_{C,S}^{\mathbf{W}}(\mathbf{X})) - (1-y) \log(1 - \phi_{C,S}^{\mathbf{W}}(\mathbf{X})). \quad (4)$$

The parameter $w_{a,b}$ in the t-th iteration is updated according to the gradient of the loss function with respect to it, i.e., $\nabla_{w_{a,b}} L(\mathbf{X}, y; \phi_{C,S}^{\mathbf{W}})$, as follows:

$$w_{a,b}(t) = w_{a,b}(t-1) - \alpha \nabla_{w_{a,b}} L(\mathbf{X}, y; \phi_{C,S}^{\mathbf{W}}), \quad (5)$$

where α is the learning rate.

In a white-box scenario [27], where the target classifier is assumed to be known to the attacker, an adversarial attack can be launched by modifying the input according to the output loss function in (4) so that the target classifier makes a wrong decision.

One of the core ideas of the adversarial embedding scheme proposed in [30] is to modify $\mathbf{C}$ according to both message bits and the signs of the gradients of the loss function with respective to the input image elements, i.e., $\text{sign}(\nabla_{x_{i,j}} L(\mathbf{X}, \hat{y}; \phi_{C,S}))$, where $\hat{y}$ is the false target label. As the signs of modification and the signs of the gradients are identical, the output stego image is capable of misleading the target classifier. However, modifying all elements according to predetermined directions would reduce the actual payload, leading to a larger distortion compared to conventional embedding. In addition, in order to mislead the classifier, it may not be necessary to utilize all image elements for adversarial embedding. Therefore, the image elements are randomly divided into two groups, where one group is used for adversarial embedding (group of adjustable elements) and the other for conventional embedding. The fraction β, indicating the ratio of the adjustable elements over all image elements, is minimized so to reduce the artifacts introduced in the stego image (under the constraint that the target steganalyzer makes a decision error). Note that the ADV-EMB scheme can successfully fool the steganalyzer trained by stego images with conventional embedding. However, in an adversarial aware steganalysis scenario, the steganographer does not know the exact kind of images (whether conventional or adversarial stego images) that are used by the steganalyst to train the steganalyzer, therefore, he may not use ADV-EMB to maximize its deception capability.

2.2. Parametric ADV-EMB

We first describe in more detail the ADV-EMB scheme proposed in [30]. ADV-EMB works under the conventional framework of distortion minimization, in which embedding costs are firstly defined according to the impact they have on each individual image element, and then practical steganographic codes [48] are employed to minimize the total distortion associated with the embedding.

Assume k message bits must be embedded in $\mathbf{C}$. For a given value $\beta \in [0, 1]$, embedding consists of the following steps.

1. Use a conventional cost function to compute the initial embedding costs of a cover image. The resultant costs of positive and negative modifications are respectively denoted as $\rho_{i,j}^+$ and $\rho_{i,j}^-$. The cost of no modification $\rho_{i,j}^0$ is assumed to be zero.

2. Randomly select a number of $l_1 = [H \times W \times (1 - \beta)]$ elements in $\mathbf{C}$ to form a common group, where $[\cdot]$ is the

4

rounding operation. The remaining $l_2 = H \times W - l_1$ elements are called adjustable elements and form the adjustable group.

3. Embed $k_1 = [k \times (1 - \beta)]$ bits into the common group using the initial embedding costs $\{\rho_{i,j}^+, \rho_{i,j}^0, \rho_{i,j}^-\}$ with a distortion minimization coding scheme, such as [48]. The resulting intermediate image is denoted as $\mathbf{Z}_c$.

4. Compute the gradients $\nabla_{z_{i,j}} L(\mathbf{Z}_c, \hat{y}; \phi_{C,S})$ of the CNN steganalyzer with respective to image elements using the target label $\hat{y} = 0$. Update the embedding costs for the adjustable elements as follows:

$$\varrho_{i,j}^+ = \begin{cases} \rho_{i,j}^+/\lambda, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) > 0, \\ \rho_{i,j}^+, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) = 0, \quad (6) \\ \rho_{i,j}^+ \cdot \lambda, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) < 0, \end{cases}$$

$$\varrho_{i,j}^0 = \rho_{i,j}^0, \quad (7)$$

$$\varrho_{i,j}^- = \begin{cases} \rho_{i,j}^-/\lambda, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) < 0, \\ \rho_{i,j}^-, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) = 0, \quad (8) \\ \rho_{i,j}^- \cdot \lambda, & \text{if } -\nabla_{z_{i,j}} L(\mathbf{Z}_c, 0; \phi_{C,S}) > 0, \end{cases}$$

where $\lambda = 2$ is a scaling factor used to bias the costs towards the desired directions. Embed $k_2 = k - k_1$ bits into the adjustable elements by using the updated embedding costs $\{\varrho_{i,j}^+, \varrho_{i,j}^0, \varrho_{i,j}^-\}$ and the same distortion minimization coding scheme used for the common group.

The above process is applied iteratively, starting from $\beta = 0$ and increasing β of a small amount $\Delta\beta$ at each iteration. When $\phi_{C,S}(Z) = 0$, i.e., the adversarial stego image Z can fool the steganalyzer, Z is taken as the output and the embedding process ends, the resulting value of β corresponding to the minimum amount of adjustable elements.

As we said, ADV-EMB is effective in fooling the target steganalyzer, but its effectiveness is reduced when an adversary-aware version of the steganalyzer, trained with adversarial stego images, is considered [30]. In fact, if the steganalyst is aware of the adversarial embedding strategy adopted by the steganographer, he also knows the parameter β to generate the adversarial stego images (β is deterministically derived by solving a minimization problem), and then he can mitigate the effectiveness of adversarial embedding by training an adversary-aware version of the CNN steganalyzer.

In this work, we consider a parametric version of the original ADV-EMB scheme, referred to as P-ADV-EMD, where embedding is implemented without minimizing the amount of adjustable elements, but performing Steps 1 to 4, for a given β. Let us denote with $\mathbf{Z}^\beta$ the resultant image. The set containing the adversarial stego images attacked with embedding parameter β is referred to as $\mathcal{Z}^\beta$. Arguably, the steganalyst does not know the value of β adopted by the attacker and has to make an educated guess; then, he/she trains the steganalyzer by considering the adversarial embedding performed with the guessed β. As confirmed in our experiments (see Section 6), when the steganalyzer trained with a given value of β is used to detect adversarial stego images with a mismatched β, the detection

performance are impaired. The best choice of the parameter β to be used by the steganalyst and the steganographer can then be determined in a game-theoretic framework, as stated in the next section.

3. CNN-based Adversary-aware Stego Embedding/Detection game.

Before defining the Adversary-aware Stego Embedding/Detection game, namely the *ASED* game, in Section 3.2, we recall some basic concepts of game theory which are necessary to understand the rest of the paper. Then, we will investigate the behavior of the *ASED* game at the Nash equilibrium.

3.1. Game theory in a nutshell

A two-player game is defined by a 4-tuple $(\mathcal{S}_1, \mathcal{S}_2, u_1, u_2)$, where $\mathcal{S}_1 = \{z_{1,1} \dots z_{1,n_1}\}$ and $\mathcal{S}_2 = \{z_{2,1} \dots z_{2,n_2}\}$ are the set of actions, or strategies, the first and the second player can choose from, and $u_l(z_{1,i}, z_{2,j}), l \in \{1, 2\}$, is the payoff of the game for player l, when the first player chooses the strategy $z_{1,i}$ ($i \in \{1, \cdots, n_1\}$) and the second chooses $z_{2,j}$ ($j \in \{1, \cdots, n_2\}$). A pair of strategies $(z_{1,i}, z_{2,j})$ is called a profile. In a strategic game, $\mathcal{S}_1$, $\mathcal{S}_2$ and the payoff functions are assumed to be known to the two players, who choose their strategies before starting the game, without knowing the strategy chosen by the other player.

When $u_1(z_{1,i}, z_{2,j}) + u_2(z_{1,i}, z_{2,j}) = 0$, i.e., the players have opposite payoffs, the game is said to be competitive or *zero-sum*. In that case, the payoff of the game can be defined by adopting the perspective of one of the two players.

The goal of game analysis is to determine the existence of equilibrium points, i.e. profiles, that in some way represent a *satisfactory* choice for both players [44]. The most famous notion of equilibrium is due to John Nash. A profile is a Nash equilibrium if no one of the players has any interest in changing his strategy assuming the other does not change his own. For the particular case of a two-player game, a profile (z_{1,i^*}, z_{2,j^*}) is a Nash equilibrium if:

$$\begin{aligned} u_1(z_{1,i^*}, z_{2,j^*}) &\geq u_1(z_{1,i}, z_{2,j^*}) \quad \forall z_{1,i} \in \mathcal{S}_1, \\ u_2(z_{1,i^*}, z_{2,j^*}) &\geq u_2(z_{1,i^*}, z_{2,j}) \quad \forall z_{2,j} \in \mathcal{S}_2. \end{aligned} \quad (9)$$

For a zero-sum game, $u_2 = -u_1$.

The above definition assumes that the players deterministically choose one of the strategies in $\mathcal{S}_i$ (pure strategy). A more flexible approach consists in letting each player choose a strategy with a certain probability. In this way, we introduce a new game in which the strategies available to the players are probability distributions over $\mathcal{S}_1$ and $\mathcal{S}_2$. The payoffs are redefined in terms of expected payoffs under the probability distributions chosen by the players. A probability distribution $\boldsymbol{p}_l$ over $\mathcal{S}_l$, that is,

$$\boldsymbol{p}_l = \{p_l(z_{l,i}), z_{l,i} \in \mathcal{S}_l | \sum_i p_l(z_{l,i}) = 1, p_l(z_{l,i}) \geq 0, \forall i\}, \quad (10)$$

is said a mixed strategy for player l. The definition of Nash equilibrium in mixed strategies extends the one given in (9). Accordingly, a mixed strategy profile $(\boldsymbol{p}_1^*, \boldsymbol{p}_2^*)$ is a mixed strategy Nash equilibrium if it satisfies $\bar{u}_1(\boldsymbol{p}_1^*, \boldsymbol{p}_2^*) \geq \bar{u}_1(\boldsymbol{p}_1, \boldsymbol{p}_2^*)$ and $\bar{u}_2(\boldsymbol{p}_1^*, \boldsymbol{p}_2^*) \geq \bar{u}_2(\boldsymbol{p}_1^*, \boldsymbol{p}_2)$, for any mixed strategy profile $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$, where $\bar{u}_l$ denotes the expected payoff for player l under the distribution corresponding to the mixed strategy profile. A central result of game theory [49] states that if we allow mixed strategies, then every game with a finite number of players and with a finite number of pure strategies for every player has at least one Nash equilibrium.

3.2. The AS ED game

As we said, we assume that the P-ADV-EMB is implemented by the steganographer A, by choosing the parameter β in a strategic way[4]. In turn, the steganalyst D has to *guess* the value of β used by the steganographer to build the adversary-aware version of the CNN detector. As anticipated, we model the interplay between the value of β adopted by A and the one adopted by D as a game. For sake of clarity, in the following, we indicate with β_A the fraction of adjustable elements adopted by A and with β_D the value considered by D in the implementation of the adversary-aware CNN steganalyzer.

More formally, A selects $\phi_{C,S}$, i.e., the unaware version of the steganalyzer, as the target steganalyzer, and the adversarial stego images are generated by considering β_A. On the other hand, D selects β_D and considers $\phi_{C,Z^{\beta_D}}$ for detection. With these ideas in mind, we are now ready to define the CNN-based AS ED game.

Definition 1. *The AS ED$(\mathcal{S}_A, \mathcal{S}_D, u_A, u_D)$ game is a two-player, non-zero-sum, strategic game played by the steganalyst (D) and the steganographer (A), defined by the following strategies and payoffs.*

- *The sets of strategies the steganographer (A) and the steganalyst (D) can choose from are, respectively, the set of possible values of β_A and β_D:*

$$\begin{aligned} \mathcal{S}_A &= \{\beta_A \in [0, 1]\}, \\ \mathcal{S}_D &= \{\beta_D \in [0, 1]\}. \end{aligned} \tag{11}$$

- *The payoff of the steganalyst (D) is defined as the negative error probability of the CNN classifier; that is*

$$u_D(\beta_A, \beta_D) = -P_e(\beta_A, \beta_D) = -\frac{1}{2}(P_{fa}(\beta_D) + P_{md}(\beta_A, \beta_D)) \tag{12}$$

where

$$P_{fa}(\beta_A, \beta_D) = \Pr\{\phi_{C,Z^{\beta_D}}(\mathbf{C}) = 1\}, \tag{13}$$

and

$$P_{md}(\beta_A, \beta_D) = \Pr\{\phi_{C,Z^{\beta_D}}(\mathbf{Z}^{\beta_A}) = 0\}. \tag{14}$$

- *The payoff of the steganographer (A) is defined as the missed detection probability, i.e., $u_A(\beta_A, \beta_D) = P_{md}(\beta_A, \beta_D)$.*

We stress that the above non zero-sum game formulation is a novelty with respect to prior art in the field, where zero sum game formulations have been considered; moreover, it better models the general adversarial stego detection scenario, where the goal of the adversary is to conceal the presence of the message in the stego images, that is, to pass off a stego as a cover, and not to induce general misclassification error.

In the above definition, the sets of strategies available to A and D are continuous sets. However, to derive the equilibrium point for the AS ED game, we will consider discrete sets of strategies by properly quantizing the values of β_A and β_D. The quantized sets of strategies are indicated by $\mathcal{S}_A^q$ and $\mathcal{S}_D^q$. Then, we consider the pair of strategies $(\beta_A, \beta_D) \in \mathcal{S}_A^q \times \mathcal{S}_D^q$. We denote with $(-\mathbf{P}_e)$, res. $\mathbf{P}_{md}$, the payoff matrices of D, res. A, where $\mathbf{P}_e = [P_e(\beta_A, \beta_D)]_{\beta_A \in \mathcal{S}_A^q, \beta_D \in \mathcal{S}_D^q}$, $\mathbf{P}_{md} = [P_{md}(\beta_A, \beta_D)]_{\beta_A \in \mathcal{S}_A^q, \beta_D \in \mathcal{S}_D^q}$.

In the sequel, we will always consider the AS ED game with quantized sets of strategies, namely $AS ED(\mathcal{S}_A^q, \mathcal{S}_D^q, u_A, u_D)$, unless stated differently.

4. Equilibrium Point Analysis (of the AS ED Game)

The ultimate goal of our analysis is to determine the equilibrium point(s) of the quantized version of the AS ED game, which, as we will see, can be found in mixed strategies.

Let $\boldsymbol{p}_D$, res. $\boldsymbol{p}_A$, indicate the mixed strategies vectors, that is, the (column) vectors with the probability distribution over the possible values of β_D, res. β_A, in $\mathcal{S}_D^q$ and $\mathcal{S}_A^q$. For a given mixed strategy profile $(\boldsymbol{p}_A, \boldsymbol{p}_D)$, the expected payoffs of A and D can be computed as:

$$\bar{u}_A(\boldsymbol{p}_A, \boldsymbol{p}_D) = \sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) \sum_{\beta_D \in \mathcal{S}_D^q} P_{md}(\beta_A, \beta_D) p_D(\beta_D), \tag{15}$$

$$\bar{u}_D(\boldsymbol{p}_A, \boldsymbol{p}_D) = -\sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) \sum_{\beta_D \in \mathcal{S}_D^q} P_e(\beta_A, \beta_D) p_D(\beta_D). \tag{16}$$

A mixed strategies Nash equilibrium profile $(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ is a pair of mixed strategies for which we have:

$$(\boldsymbol{p}_A^*)^T \cdot \mathbf{P}_{md} \cdot \boldsymbol{p}_D^* = \max_{\boldsymbol{p}_A} \left\{ (\boldsymbol{p}_A)^T \cdot \mathbf{P}_{md} \cdot \boldsymbol{p}_D^* \middle| \sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) = 1, p_A(\beta_A) \geq 0 \right\},$$

$$(\boldsymbol{p}_A^*)^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D^* = \min_{\boldsymbol{p}_D} \left\{ (\boldsymbol{p}_A^*)^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D \middle| \sum_{\beta_D \in \mathcal{S}_D^q} p_D(\beta_D) = 1, p_D(\beta_D) \geq 0 \right\}. \tag{17}$$

In general, solving a non-zero-sum game, i.e. finding the Nash equilibrium (equilibria) of the game, is not easy [50]. In hindsight, in our case, the problem can be simplified by observing that the P_{fa} corresponding to the steganalyzer $\phi_{C,Z^{\beta_D}}$ does not depend on β_A. Specifically, we can write the following equivalence:

$$(\boldsymbol{p}_A)^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D = \frac{1}{2} \left\{ (\boldsymbol{p}_A)^T \cdot \mathbf{P}_{md} \cdot \boldsymbol{p}_D + \sum_{\beta_D \in \mathcal{S}_D^q} P_{fa}(\beta_D) p_D(\beta_D) \right\}. \tag{18}$$

---

[4]We assume that the steganographer has a perfect knowledge of the target unaware steganalyzer, i.e., the specific CNN architecture adopted and the training set.

The above relation follows immediately by observing that

$$(\boldsymbol{p}_A)^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D = \frac{1}{2} \left\{ (\boldsymbol{p}_A)^T \cdot \mathbf{P}_{md} \cdot \boldsymbol{p}_D + (\boldsymbol{p}_A)^T \mathbf{P}_{fa} \cdot \boldsymbol{p}_D \right\}$$

where $\mathbf{P}_{fa} = \left[P_{fa}(\beta_A, \beta_D) \right]_{\beta_A \in \mathcal{S}_A^q, \beta_D \in \mathcal{S}_D^q}$, which is constant over the rows. Then:

$$
\begin{aligned}
(\boldsymbol{p}_A)^T \mathbf{P}_{fa} \cdot \boldsymbol{p}_D &= \sum_{\beta_D \in \mathcal{S}_D^q} \left(\sum_{\beta_A \in \mathcal{S}_A^q} P_{fa}(\beta_A, \beta_D) \cdot p_A(\beta_A) \right) \cdot p_D(\beta_D) \\
&= \sum_{\beta_D \in \mathcal{S}_D^q} P_{fa}(\beta_D) \cdot \left(\sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) \right) \cdot p_D(\beta_D) \\
&= \sum_{\beta_D \in \mathcal{S}_D^q} P_{fa}(\beta_D) \cdot p_D(\beta_D).
\end{aligned}
$$

(19)

Accordingly, the mixed strategy $\boldsymbol{p}_A^*$ at the Nash equilibrium can be equivalently obtained by solving the maximization below:

$$\boldsymbol{p}_A^* = \arg \max_{\boldsymbol{p}_A} \left\{ (\boldsymbol{p}_A)^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D^* \,\middle|\, \sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) = 1, p_A > 0 \right\}, \quad (20)$$

where $\mathbf{P}_e$ is considered as payoff matrix for A.

Given the above derivation, the following property holds immediately.[5]

Property 1. *A profile $(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ is a Nash equilibrium of the non-zero-sum strategic game $ASED(\mathcal{S}_A^q, \mathcal{S}_D^q, u_A, u_D)$, if and only if it is a Nash equilibrium of the zero-sum strategic game $ASED'(\mathcal{S}_A^q, \mathcal{S}_D^q, u)$ where $u = \mathbf{P}_e$ is the payoff of the game, defined by adopting the steganalyst's perspective (i.e., A aims at maximizing u, while D aims at minimizing it).*

Thanks to Property 1, in order to find the equilibrium point of the $ASED$ game, we can conveniently solve the zero-sum game $ASED'$. This represents a great advantage, since finding the Nash equilibrium of a zero-sum game is significantly easier. In particular, since the set of strategies is discrete and finite, the equilibrium point can be derived by relying on the minimax theorem [44]. According to such a theorem, the mixed strategies Nash equilibrium $(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ can be obtained by solving separately the following max-min and min-max problems:

$$
\begin{aligned}
\boldsymbol{p}_A^* &= \arg \max_{\boldsymbol{p}_A(\mathcal{S}_A^q)} \min_{\boldsymbol{p}_D(\mathcal{S}_D^q)} \boldsymbol{p}_A^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D \\
\boldsymbol{p}_D^* &= \arg \min_{\boldsymbol{p}_D(\mathcal{S}_D^q)} \max_{\boldsymbol{p}_A(\mathcal{S}_A^q)} \boldsymbol{p}_A^T \cdot \mathbf{P}_e \cdot \boldsymbol{p}_D,
\end{aligned}
$$

(21)

where the max and min are restricted to the set of probability vectors, that is, the vectors for which $\{\boldsymbol{p}_A : \sum_{\beta_A \in \mathcal{S}_A^q} p_A(\beta_A) = 1, p_A(\beta_A) \geq 0, \forall \beta_A\}$ and $\{\boldsymbol{p}_D : \sum_{\beta_D \in \mathcal{S}_D^q} p_D(\beta_D) = 1, p_D(\beta_D) > 0, \forall \beta_D\}$. The above system can be reduced to the solution of a linear programming (LP) problem, see [51]. The expected payoffs of D and A at the equilibrium of the non-zero-sum game are $\bar{u}_D(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ and $\bar{u}_A(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$, computed as in Equation (15) and (16).

---

[5]The property can also be proved with reference to the continuous game $ASED(\mathcal{S}_A, \mathcal{S}_D, u_A, u_D)$ (the derivation is obtained by replaing the sums with integrals in equations (17) through (20)).

5. Equilibrium Assessment in Practice

Having established that solving the $ASED'$ game is equivalent to finding the equilibrium points of the $ASED$ game, our next step is to investigate the behavior of the $ASED$ game in a practical scenario and analyze the achievable performance of the steganalyst and the steganographer, when they adopt CNN aware training and the P-ADV-EMB scheme respectively, and tune the values of β_D and β_A strategically. Specifically, the goal of our research is to study the equilibrium point of the game in pure or mixed strategies, and analyze the behavior and the payoff of the steganographer and the steganalyst at the equilibrium. The performance at the equilibrium are then compared to those obtained by adopting a worst case approach and those achieved by training the steganalyzer with a proper mixture of adversarial stego samples obtained by adopting different values of β_D.

5.1. Experimental setting

The setup we have used to conduct our experiments is described in the following. We applied the P-ADV-EMB algorithm in the spatial domain by selecting S-UNIWARD [7] as the baseline scheme for conventional embedding. Xu-net [23] was used as the CNN classifier for its fast convergence under a moderate size of the training set. The Xu-net steganalyzer trained with conventional S-UNIWARD stego images, i.e., $\phi_{C_{trn}, S_{trn}}$ (or, equivalently, $\phi_{C_{trn}, \mathcal{Z}_{trn}^0}$), was used as the target steganalyzer used to generate the adversarial stego images $\mathbf{Z}^{\beta_A}$.

For the evaluation, we used the BOSSBase v1.01 [52] dataset, which contains 10,000 cover images of size 512×512. The payload was set to 0.4 bpp (bit per pixel). In the experiments, firstly, we quantized β based on the considerations we made in Section 5.3. Then, for a given β, we randomly split 10,000 pairs of cover images and their corresponding stego images into three disjoint subsets, i.e., training set $\{C_{trn}, \mathcal{Z}_{trn}^\beta\}$, validation set $\{C_{val}, \mathcal{Z}_{val}^\beta\}$, and test set $\{C_{tst}, \mathcal{Z}_{tst}^\beta\}$, with 4000, 1000, and 5000 pairs of images, respectively. The training set was employed to train the learnable parameters/weights in the CNN steganalyzer, while the validation set was used for the selection of the best-performing model, i.e., the best-performing parameters/weights in the CNN, to prevent over-fitting to the training set. The reported performance of the CNN steganalyzer was evaluated on the test sets. Finally, to build the payoff matrices $\mathbf{P}_e$ and $\mathbf{P}_{md}$, each CNN steganalyzer $\phi_{C_{trn}, \mathcal{Z}_{trn}^{\beta_D}}, \beta_D \in \mathcal{S}_D^q$, was run on the set $\{C_{tst}, \mathcal{Z}_{tst}^{\beta_A}\}$ for all $\beta_A \in \mathcal{S}_A^q$.

CNN training and testing was performed on TensorFlow with Python interface and a NVIDIA Tesla P100 GPU card. The weights of CNN convolutional filters and fully connected layers were initialized by using a normal distribution with zero mean and standard deviation equal to 0.01. With regard to learning, we used stochastic gradient descent, with momentum equal to 0.9 and initial learning rate 0.001. The learning rate decay was set to 90% every 5,000 training steps. The batch size in each iteration was set to 50 (25 cover/stego pairs). The training stage lasted 110,000 iterations, and validation was performed every 5,000 iterations.

5.2. Investigated solutions

5.2.1. Nash equilibrium

To stat with, we measured the performance at the equilibrium of the stego detection game by following the approach presented in Section 4. Specifically, by Property 1, we obtained the Nash equilibrium profile $(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ by solving the LP problem associated to the zero-sum game formulation $ASED'$, with payoff matrix $\mathbf{P}_e$ having entries $|\mathcal{S}_A^q| \times |\mathcal{S}_D^q|$. Then, we computed the payoffs of D and A according to Equations 15 and 16. We used the linear programming tools of Matlab Optimization Toolbox [53] to solve the LP problem by means of the simplex algorithm.

5.2.2. Worst case solution

As a second possibility, we considered the case in which A and B adopts a conservative worst case approach. In this case, A and D choose the strategy which maximizes their own payoff in the worst case with respect to the move of the opponent [6].

In our setup, the worst case strategy for A corresponds to the value of β_A that maximizes the missed detection probability when D plays the strategy that minimizes it, that is, given the payoff matrix $\mathbf{P}_{md}$,

$$\hat{\beta}_A = \arg \max_{\beta_A \in \mathcal{S}_A^q} \left\{ \min_{\beta_D \in \mathcal{S}_D^q} P_{md}(\beta_A, \beta_D) \right\}. \tag{22}$$

The corresponding worst case solution is $\min_{\beta_D} P_{md}(\hat{\beta}_A, \beta_D)$.

Likewise, given D's payoff matrix $-\mathbf{P}_e$, the worst case strategy for D is

$$\hat{\beta}_D = \arg \min_{\beta_D \in \mathcal{S}_D^q} \left\{ \max_{\beta_A \in \mathcal{S}_A^q} P_e(\beta_A, \beta_D) \right\}. \tag{23}$$

Usually, $\hat{\beta}_D$ (res. $\hat{\beta}_A$) is different from the value obtained from the inner minimization in (22) (res. in (23)). When the solution of Equations (22) and (23) is the same, the profile $(\hat{\beta}_A, \hat{\beta}_D)$ corresponds to a pure strategy Nash equilibrium.

In the next section, we compare the worst case solution to the Nash equilibrium of the game, to show the advantage provided by the game-theoretic analysis.

5.2.3. Training over a mixture of strategies

Eventually, we considered the solution obtained by training the CNN on a mixture of β values. In particular, we considered the mixture corresponding to the equilibrium mixed strategy for D, namely $\boldsymbol{p}_D^* = [p_D^*(\beta_D)]_{\beta_D \in \mathcal{S}_D^q}$, to build the adversarial stego image set $\mathcal{Z}^{\boldsymbol{p}_D^*}$, where adversarial embedding was performed with various $\beta_D \in \mathcal{S}_D^q$ according to the probability vector $\boldsymbol{p}_D^*$. The trained model is denoted with $\phi_{C_{trn}, \mathcal{Z}_{trn}^{\boldsymbol{p}_D^*}}$ in this case. In practice, we exploited the equilibrium solution of the game from the perspective of the steganalyst, thus going beyond a strictly game-theoretic analysis. More specifically, by training

the CNN with a mixture of β, we gave an advantage to the steganalyst, since the steganographer is assumed to keep playing at the equilibrium of the game, which was found without considering the possibility that the steganalyzer was trained on a mixture of β's.[7] We also considered the case in which adversarial embedding was performed by considering an uniform distribution of β_D across the training set $\mathcal{S}_D^q$. In this case, the trained steganalyzer is denoted by $\phi_{C_{trn}, \mathcal{Z}_{trn}^{uni}}$.

5.3. Quantization of β values

In order to approximate the behavior of the continuous game (see Definition 1), we should consider a fine enough quantization of β_A and β_D. However, considering a very fine quantization increases dramatically the computational burden of the tests, since a CNN model has to be trained for every value of $\beta_D \in \mathcal{S}_D^q$. In our experiments, we first quantized β_A and β_D with a uniform quantization step size of 0.05, then we considered non-uniform quantization steps with a smaller step size in the region of (β_A, β_D) values where the payoff varies more rapidly. We considered only the case of equal quantization strategies for D and A, that is $\mathcal{S}_A^q \equiv \mathcal{S}_D^q \equiv \mathcal{S}^q$.

6. Experimental Results

In this section, we report the results of our experiments and discuss their meaning.

6.1. Payoff matrices

In Table 1, we report the payoff matrix $\mathbf{P}_e$ obtained from our experiments when β_A and β_D are quantized with a uniform quantization step size equal to 0.05. We also report the payoff matrix $\mathbf{P}_{md}$ in Table 2. From the analysis carried out in Section 3 (and the result stated by Property 1), we know that only the matrix $\mathbf{P}_e$ is necessary to solve the game and derive the optimum strategies for D and A. Then, the values of P_{md} corresponding to the optimum parameters are used to compute the payoff of A at the equilibrium. In the tables, we highlight the results for the case $\beta_A = \beta_D$ (diagonal) with grey shades. The performance at $(\beta_A, \beta_D) = (0, 0)$ corresponds to the performance when conventional S-UNIWARD steganography and steganalysis are considered (this case provides the baseline performance and is highlighted with a wave line), in which case $P_e = 20.9\%$ and $P_{md} = 24.0\%$.

For a fixed β_A, one would expect that the best possible performance for D is achieved when $\beta_D = \beta_A$. Actually, this is not always the case. However, in all the cases where, for a given β_A, the steganalyzer $\phi_{C_{trn}, \mathcal{Z}_{trn}^{\beta_A}}$ is outperformed by $\phi_{C_{trn}, \mathcal{Z}_{trn}^{\beta_D}}$ with a $\beta_D \neq \beta_A$ (highlighted with dash underlines in the table), the difference in the corresponding value of the P_e is not much. It can also be observed that the region where $0.1 \leq \beta_D < \beta_A \leq 0.3$ is where the difference between the performance of the steganalyzer with matched and mismatched β is more relevant. By

---

[6]Note that, in a zero-sum game, this strategy may not correspond to the strategy that maximizes the opponent's payoff.

[7]Training on mixtures of β's was not included in the set of strategies of the game.

Table 1: P_e (in%) with a uniform quantization step size of 0.05 on β. (Wave: the case of conventional embedding and detection. Bold: the case when the steganalyst has a worse performance compared to the case of conventional embedding and detection. Shade: the case when the steganalyst uses a matched parameter for detection, i.e., $\beta_D = \beta_A$. Dash underline: the case when the steganalyst has a better performance with a mismatched parameter than with a matched parameter. Underline: worst case solution for steganalyst.)

$\beta_A\backslash\beta_D$	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
0.00	20.9	26.4	50.5	50.3	50.1	50.1	50.1	50.1	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
0.05	39.0	26.7	42.7	48.5	49.5	49.8	49.7	49.7	49.8	50.0	49.9	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
0.10	54.7	27.1	20.1	30.9	45.0	47.2	47.1	48.1	48.7	49.5	49.6	49.8	49.9	49.8	49.9	49.8	49.9	49.9	49.9	49.8	49.9
0.15	57.0	28.2	8.0	10.1	24.6	29.1	34.2	42.8	43.6	47.5	47.8	48.3	49.1	48.9	49.3	49.0	49.4	48.3	48.8	48.8	49.2
0.20	57.7	28.8	4.8	4.3	6.9	8.6	12.7	26.1	22.3	36.4	37.1	38.8	42.1	44.9	43.9	42.5	45.3	42.7	44.2	45.8	46.7
0.25	57.9	30.1	3.7	2.6	2.9	3.1	4.6	7.4	6.9	14.4	16.3	19.1	24.1	29.3	31.3	29.1	35.8	36.0	38.0	41.1	43.0
0.30	58.1	31.4	3.0	1.9	1.6	1.8	2.0	2.5	2.0	5.2	5.5	7.5	10.7	14.3	18.9	17.8	24.4	28.0	30.3	34.8	38.0
0.35	58.2	32.5	2.7	1.4	1.1	1.1	1.2	1.1	0.6	2.6	2.0	2.9	4.7	6.6	10.0	10.0	13.5	19.1	21.7	26.7	31.2
0.40	58.3	33.9	2.5	1.1	0.9	0.8	0.9	0.9	0.3	1.0	0.6	1.2	1.8	2.8	4.8	4.6	6.5	10.7	13.4	18.0	23.4
0.45	58.4	34.8	2.4	1.0	0.8	0.6	0.7	0.7	0.2	0.6	0.3	0.5	0.8	1.3	2.1	2.1	3.0	5.3	7.1	9.7	15.4
0.50	58.5	35.5	2.3	1.0	0.7	0.4	0.5	0.7	0.2	0.4	0.2	0.3	0.4	0.8	1.0	1.0	1.4	2.5	3.5	4.6	9.0
0.55	58.6	36.2	2.2	0.9	0.7	0.5	0.5	0.6	0.2	0.3	0.2	0.2	0.2	0.6	0.5	0.5	0.8	1.1	1.8	2.0	5.1
0.60	58.6	36.7	2.2	0.9	0.7	0.4	0.4	0.6	0.2	0.2	0.1	0.1	0.2	0.5	0.3	0.3	0.4	0.6	0.8	0.9	2.9
0.65	58.6	37.4	2.2	0.8	0.6	0.3	0.3	0.6	0.2	0.2	0.1	0.1	0.1	0.3	0.2	0.3	0.3	0.3	0.5	0.5	1.6
0.70	58.6	37.6	2.1	0.8	0.6	0.3	0.3	0.6	0.2	0.2	0.1	0.1	0.1	0.3	0.2	0.2	0.2	0.2	0.4	0.3	1.1
0.75	58.7	37.8	2.1	0.8	0.6	0.3	0.3	0.6	0.2	0.2	0.1	0.1	0.1	0.3	0.1	0.2	0.2	0.2	0.2	0.2	0.8
0.80	58.7	37.7	2.1	0.8	0.6	0.3	0.3	0.6	0.2	0.2	0.2	0.2	0.1	0.3	0.1	0.2	0.1	0.1	0.1	0.2	0.6
0.85	58.7	37.7	2.1	0.8	0.6	0.4	0.4	0.6	0.2	0.3	0.2	0.2	0.2	0.3	0.2	0.2	0.1	0.1	0.1	0.2	0.4
0.90	58.7	37.4	2.1	0.8	0.6	0.4	0.4	0.7	0.2	0.4	0.3	0.3	0.3	0.5	0.2	0.2	0.1	0.1	0.1	0.1	0.3
0.95	58.7	37.5	2.1	0.8	0.7	0.4	0.5	0.8	0.3	0.5	0.5	0.5	0.4	0.6	0.4	0.3	0.2	0.2	0.1	0.2	0.3
1.00	58.7	37.2	2.1	0.8	0.7	0.5	0.7	0.9	0.5	0.6	0.6	0.6	0.6	0.8	0.6	0.5	0.2	0.2	0.2	0.2	0.2

Table 2: P_{md} (in%) with a uniform quantization step size of 0.05 on β. (Wave: the case of conventional embedding and detection. Shade: the case when the steganalyst uses a matched parameter for detection, i.e., $\beta_D = \beta_A$. Bold: the case when the steganographer has a better performance compared to the case of conventional embedding and detection. Double underline: worst case solution for steganographer.)

$\beta_A\backslash\beta_D$	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
0.00	24.0	29.8	96.8	99.2	99.1	99.6	99.5	99.1	99.8	99.7	100.0	99.9	99.9	99.6	99.9	99.8	99.9	100.0	99.9	100.0	99.9
0.05	60.3	30.6	81.3	95.6	97.8	98.9	98.8	98.2	99.3	99.6	99.8	99.8	99.8	99.5	99.9	99.7	99.9	99.9	99.9	99.8	99.9
0.10	91.7	31.2	36.1	60.4	88.8	93.7	93.6	95.0	97.1	98.7	99.2	99.4	99.6	99.1	99.7	99.4	99.7	99.7	99.6	99.6	99.7
0.15	96.3	33.5	11.8	18.7	48.0	57.5	67.8	84.4	86.8	94.7	95.5	96.5	98.1	97.4	98.4	97.8	98.6	96.5	97.4	97.5	98.3
0.20	97.7	34.8	5.5	7.2	12.6	16.5	24.8	51.1	44.5	75.5	74.1	77.4	84.1	89.4	87.6	84.8	90.6	85.4	88.2	91.6	93.3
0.25	98.2	37.4	3.3	3.8	4.6	5.6	8.7	13.7	13.6	28.4	32.5	38.0	48.2	58.2	62.5	58.0	71.4	72.0	75.8	82.1	85.9
0.30	98.5	39.9	1.9	2.3	2.0	3.0	3.5	3.9	3.7	10.1	11.0	14.9	21.4	28.2	37.7	35.5	48.6	55.9	60.4	69.5	75.9
0.35	98.7	42.2	1.2	1.3	1.1	1.5	1.8	1.1	0.9	3.8	4.0	5.7	9.3	12.9	19.8	19.7	26.8	38.1	43.6	53.4	62.3
0.40	98.9	44.9	0.8	0.8	0.7	0.9	1.2	0.7	0.4	1.6	1.2	2.2	3.6	5.2	9.4	9.1	12.8	21.4	26.7	35.9	46.8
0.45	99.2	46.6	0.7	0.7	0.5	0.5	0.7	0.4	0.2	0.8	0.6	0.8	1.6	2.2	4.0	4.0	6.0	10.5	14.2	19.4	30.7
0.50	99.3	48.1	0.4	0.6	0.3	0.2	0.5	0.2	0.1	0.5	0.3	0.4	0.7	1.2	1.8	1.8	2.7	4.9	6.9	9.1	17.9
0.55	99.4	49.6	0.3	0.3	0.2	0.3	0.4	0.2	0.1	0.3	0.2	0.2	0.4	0.7	0.9	0.9	1.5	2.1	3.4	4.0	10.0
0.60	99.5	50.5	0.2	0.3	0.2	0.1	0.3	0.1	0.1	0.2	0.2	0.1	0.3	0.5	0.5	0.5	0.7	1.0	1.6	1.8	5.7
0.65	99.6	51.9	0.2	0.2	0.1	0.0	0.1	0.1	0.1	0.1	0.1	0.2	0.3	0.3	0.3	0.3	0.5	0.5	1.0	0.9	3.1
0.70	99.6	52.4	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.2	0.3	0.2	0.3	0.2	0.4	0.3	0.6	0.4	2.1
0.75	99.6	52.8	0.1	0.1	0.0	0.0	0.1	0.1	0.1	0.1	0.1	0.1	0.2	0.1	0.2	0.1	0.2	0.3	0.3	0.3	1.4
0.80	99.7	52.5	0.1	0.1	0.0	0.0	0.1	0.1	0.2	0.2	0.2	0.1	0.2	0.2	0.2	0.2	0.1	0.2	0.2	0.3	1.4
0.85	99.7	52.5	0.1	0.1	0.1	0.1	0.2	0.2	0.2	0.3	0.3	0.3	0.3	0.3	0.3	0.2	0.1	0.1	0.2	0.2	0.7
0.90	99.7	51.9	0.1	0.1	0.1	0.1	0.2	0.2	0.2	0.4	0.5	0.5	0.5	0.5	0.3	0.3	0.2	0.1	0.2	0.2	0.3
0.95	99.7	52.1	0.1	0.2	0.2	0.2	0.4	0.5	0.4	0.6	0.5	0.8	0.8	0.8	0.6	0.4	0.3	0.3	0.2	0.2	0.3
1.00	99.7	51.5	0.1	0.2	0.3	0.4	0.7	0.7	0.7	0.9	1.2	1.1	1.1	1.2	1.0	0.7	0.3	0.3	0.3	0.2	0.2

Table 3: P_e (in%) with a non-uniform quantization on β. (Wave: the case of conventional embedding and detection. Bold: the case when the steganalyst has a worse performance compared to the case of conventional embedding and detection. Shade: the case when the steganalyst uses a matched parameter for detection, i.e., $\beta_D = \beta_A$. Dash underline: the case when the steganalyst has a better performance with a mismatched parameter than with a matched parameter. Underline: worst case solution for steganalyst.)

$\beta_A\backslash\beta_D$	0.00	0.02	0.04	0.05	0.06	0.07	0.08	0.10	0.12	0.14	0.16	0.18	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.00	20.9	21.5	23.0	26.4	32.1	37.5	43.2	50.5	50.9	50.6	50.9	50.3	50.1	50.1	50.0	50.0	50.0	50.0	50.0	50.0	50.0
0.02	27.1	23.6	24.3	26.6	30.2	33.9	39.1	49.2	49.7	50.0	50.4	50.0	49.9	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
0.04	34.9	27.1	25.7	26.7	28.3	30.0	33.1	45.7	46.6	48.8	49.1	49.4	49.7	49.8	49.9	50.0	50.0	50.0	50.0	50.0	50.0
0.05	40.0	28.4	26.3	26.7	27.1	28.2	29.7	42.7	43.7	47.4	47.5	48.8	49.5	49.7	49.8	49.9	50.0	50.0	50.0	50.0	50.0
0.06	42.7	30.3	27.0	26.7	26.1	26.6	26.2	38.9	40.4	45.5	45.7	48.3	49.1	49.6	49.7	49.9	50.0	50.0	50.0	50.0	50.0
0.07	46.5	32.3	28.0	26.8	25.0	24.9	23.1	34.6	35.4	43.1	42.6	47.3	48.6	49.2	49.6	49.8	49.9	50.0	50.0	50.0	50.0
0.08	50.1	34.4	28.8	27.2	24.8	23.9	20.3	30.3	31.0	39.4	38.5	45.8	48.1	48.8	49.4	49.8	49.9	50.0	50.0	49.9	50.0
0.10	54.7	38.9	30.4	27.1	23.0	21.7	14.9	20.1	21.3	29.0	26.8	39.7	45.0	47.1	48.7	49.6	49.9	49.9	49.9	49.9	49.9
0.12	56.3	42.4	32.3	27.3	22.1	19.9	12.0	13.2	15.0	18.4	17.5	29.5	40.0	43.8	47.6	49.2	49.7	49.7	49.8	49.6	49.8
0.14	56.8	46.6	34.2	27.8	21.2	18.9	9.5	9.2	10.9	11.6	12.7	18.4	30.5	38.2	45.4	48.4	49.4	49.6	49.6	49.1	49.4
0.16	57.2	49.7	36.1	28.4	20.5	17.9	8.2	7.0	8.6	7.6	9.5	11.3	19.3	30.0	40.4	46.6	48.6	48.9	49.1	48.0	48.8
0.18	57.4	52.5	38.0	28.7	19.7	17.1	7.0	5.6	7.0	5.7	7.7	7.2	11.3	20.3	32.6	43.4	46.9	47.6	47.8	46.5	47.9
0.20	57.7	54.5	40.0	28.8	19.2	16.5	6.6	4.8	6.2	4.6	6.6	4.7	6.9	12.7	22.3	37.1	42.1	43.9	45.3	44.2	46.7
0.30	58.1	56.9	47.1	31.4	17.8	14.4	4.9	3.0	4.3	2.9	3.8	2.0	1.6	2.0	2.0	5.5	10.7	18.9	24.4	30.3	38.0
0.40	58.3	57.7	51.5	33.9	16.7	13.2	4.3	2.5	3.7	2.5	2.8	1.8	0.9	0.9	0.3	0.7	1.8	4.8	6.5	13.4	23.5
0.50	58.5	58.2	53.8	35.5	16.4	12.6	4.0	2.3	3.2	2.2	2.4	1.7	0.7	0.5	0.2	0.2	0.4	1.0	1.4	3.5	9.0
0.60	58.6	58.5	54.9	36.7	17.2	12.2	3.8	2.2	3.1	2.2	2.1	1.7	0.7	0.4	0.2	0.1	0.2	0.3	0.4	0.8	2.9
0.70	58.6	58.9	55.2	37.6	17.5	12.0	3.8	2.1	3.0	2.1	1.9	1.6	0.6	0.3	0.2	0.1	0.1	0.2	0.2	0.4	1.1
0.80	58.7	59.0	55.1	37.7	18.2	11.8	3.8	2.1	3.0	2.1	1.8	1.6	0.6	0.3	0.2	0.2	0.1	0.1	0.1	0.1	0.6
0.90	58.7	59.1	54.2	37.4	18.6	11.9	3.8	2.1	3.0	2.1	1.7	1.7	0.6	0.4	0.2	0.3	0.3	0.2	0.1	0.1	0.3
1.00	58.7	58.9	53.0	37.2	18.3	11.9	3.9	2.1	3.0	2.2	1.7	1.7	0.7	0.7	0.5	0.6	0.6	0.6	0.2	0.2	0.2

Table 4: P_{md} with a non-uniform quantization on β. (Wave: the case of conventional embedding and detection. Shade: the case when the steganalyst uses a matched parameter for detection, i.e., $\beta_D = \beta_A$. Bold: the case when the steganographer has a better performance compared to the case of conventional embedding and detection. Double underline: worst case solution for steganographer.)

$\beta_A\backslash\beta_D$	0.00	0.02	0.04	0.05	0.06	0.07	0.08	0.10	0.12	0.14	0.16	0.18	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.00	24.0	21.3	26.3	29.8	44.3	55.8	79.1	96.8	96.2	97.1	98.8	97.4	99.1	99.5	99.8	100.0	99.9	99.9	99.9	99.9	99.9
0.02	36.4	25.5	28.9	30.3	40.4	48.5	70.8	94.3	93.9	95.8	97.7	96.7	98.7	99.4	99.7	99.8	99.9	99.9	99.9	99.9	99.9
0.04	52.2	32.5	31.7	30.6	36.7	40.7	58.8	87.2	87.6	93.4	95.2	95.6	98.2	99.0	99.5	99.8	99.9	99.9	99.9	99.9	99.9
0.05	60.3	35.1	32.8	30.6	34.2	37.1	52.0	81.3	81.7	90.7	92.1	94.3	97.8	98.8	99.3	99.8	99.8	99.9	99.9	99.8	99.9
0.06	67.7	38.9	34.3	30.5	32.1	34.0	45.2	73.7	75.2	86.8	88.5	93.3	97.0	98.6	99.2	99.7	99.9	99.8	99.9	99.8	99.8
0.07	75.4	42.9	36.3	30.6	30.0	30.6	38.8	65.1	65.1	82.0	82.2	91.4	96.0	97.9	98.9	99.5	99.8	99.8	99.9	99.8	99.8
0.08	82.4	47.1	37.9	31.6	29.5	28.6	33.2	56.5	56.4	74.7	74.0	88.4	95.0	97.0	98.5	99.5	99.8	99.8	99.8	99.7	99.8
0.10	91.7	56.0	41.0	31.2	26.0	24.2	22.6	36.1	36.9	53.9	50.7	76.1	88.8	93.6	97.1	99.2	99.6	99.7	99.7	99.6	99.7
0.12	94.9	63.1	45.0	31.6	24.1	20.6	16.8	22.4	24.4	32.7	32.0	55.7	78.9	86.9	94.9	98.2	99.4	99.3	99.5	99.1	99.5
0.14	96.0	71.4	48.7	32.7	22.5	18.6	11.8	14.3	16.2	19.1	22.5	33.6	60.0	75.8	90.5	96.7	98.6	99.0	99.1	98.1	98.7
0.16	96.7	77.7	52.4	33.9	20.9	16.6	9.1	9.8	11.6	11.1	15.9	19.4	37.4	59.4	80.4	93.2	97.2	97.8	98.1	95.8	97.4
0.18	97.1	83.2	56.2	34.5	19.5	14.9	6.8	7.2	8.4	7.2	12.3	11.1	21.4	40.0	64.9	86.8	93.6	95.0	95.5	92.8	95.8
0.20	97.7	87.2	60.3	34.8	18.4	13.7	5.8	5.5	6.9	5.1	10.2	6.2	12.6	24.8	44.4	74.1	84.1	87.6	90.6	88.2	93.3
0.30	98.5	92.1	74.5	39.9	15.7	9.5	2.5	1.9	2.9	1.7	4.5	0.8	2.0	3.5	3.7	11.0	21.4	37.7	48.6	60.4	75.9
0.40	98.9	93.7	83.2	44.9	13.3	7.2	1.3	0.8	1.7	0.8	2.7	0.3	0.7	1.2	0.4	1.2	3.6	9.4	12.8	26.7	46.8
0.50	99.3	94.6	87.8	48.1	12.8	5.9	0.7	0.4	0.9	0.4	1.8	0.2	0.3	0.5	0.1	0.3	0.7	1.8	2.7	6.9	17.9
0.60	99.5	95.3	90.0	50.5	14.4	5.2	0.4	0.2	0.6	0.3	1.1	0.1	0.2	0.3	0.1	0.2	0.3	0.5	0.7	1.6	5.7
0.70	99.6	96.0	90.7	52.4	15.0	4.7	0.3	0.2	0.4	0.1	0.8	0.1	0.1	0.1	0.1	0.1	0.1	0.3	0.4	0.6	2.1
0.80	99.7	96.2	90.4	52.5	16.4	4.4	0.4	0.1	0.3	0.1	0.6	0.1	0.0	0.1	0.1	0.2	0.1	0.2	0.1	0.2	1.1
0.90	99.7	96.5	88.7	51.9	17.2	4.6	0.4	0.1	0.3	0.1	0.4	0.1	0.1	0.2	0.2	0.5	0.5	0.3	0.2	0.2	0.5
1.00	99.7	96.1	86.3	51.5	16.6	4.6	0.5	0.1	0.3	0.2	0.3	0.2	0.3	0.7	0.7	1.2	1.1	1.0	0.3	0.3	0.3

closer inspection of Table 1, we also observe that, in this region, the performance of the steganalyzer varies greatly, thus calling for a further investigation on a finer quantization of β. For this reason, we also considered a different quantization for the β values, which is finer for small β, and coarser for large β; in particular, we considered the following non-uniform set of values: $\mathcal{S}_{nu}^{q} = \{0, 0.02, 0.04, 0.05, 0.06, 0.07, 0.08, 0.10, 0.12, 0.14, 0.16, 0.18, 0.20, 0.30, 0.40, 0.50, 0.60, 0.70, 0.80, 0.90, 1.00\}$. The corresponding matrices $\mathbf{P}_e$ and $\mathbf{P}_{md}$ are shown in Table 3 and 4, respectively.

From both Table 1 and 3, we can observe the following general behavior:

- when D uses a steganalyzer with a very small β_D ($\beta_D \leq 0.05$), the attacker would better choose a large β_A;

- when D uses a steganalyzer with a large β_D, for the attacker it is better to choose a small β_A;

- for matched values of β_A and β_D (corresponding to the diagonal of the matrix), the attacker can outperform the baseline only when such values are relatively small.

A similar behavior also holds for the $\mathbf{P}_{md}$ matrix in Table 2 and 4.

6.2. Equilibrium of the game

In Table 5 we report the mixed strategy Nash equilibrium $(\boldsymbol{p}_A^*, \boldsymbol{p}_D^*)$ for the $ASED$ game when β is uniformly quantized (there are no equilibria in pure strategies). At the equilibrium, the optimal payoff for the defender is $P_e^* = 30.8\%$, while for the attacker the optimal payoff is $P_{md}^* = 42.9\%$. Compared to conventional embedding where $P_e(0,0) = 20.9\%$, the error probability is increased by almost 10%. This percentage characterizes the loss in the error probability with respect to the case of conventional non-adversarial steganography, due to adversarial embedding. With regard to the equilibrium strategies, interestingly, it can be observed that the optimum behavior of the steganalyst, and especially the steganographer, corresponds to alternate playing a small and a medium/large β. In particular, among the possible values of β, the best for the attacker is to perform conventional steganography ($\beta_A = 0$) and adversarial steganography with $\beta_A = 0.05$, with a probability of 0.476 and 0.140 respectively, and to consider a stronger adversarial embedding with $\beta_A = 0.75$ with a probability of 0.384. On the other hand, for the defender, the best is to consider the aware steganalyzers $\phi_{C_{trn}, \mathcal{Z}_{trn}^{0.05}}$, $\phi_{C_{trn}, \mathcal{Z}_{trn}^{0.1}}$, and $\phi_{C_{trn}, \mathcal{Z}_{trn}^{0.5}}$ with a probability of 0.812, 0.040, and 0.148, respectively.

In Table 6, we show the equilibrium when β is quantized non-uniformly. The optimal payoff for the defender is $P_e^* = 29.0\%$, while for the attacker is $P_{md}^* = 36.3\%$. Compared to the case of uniform quantization, using finer quantization in the region where the payoffs vary rapidly gives slightly more advantage to the defender. However, the behavior of the optimum mixed strategy at the equilibrium is similar to the one observed before: in particular, for the attacker, the best is to use a conventional steganographic scheme with an increased probability of 0.767,

and a strong adversarial embedding (with $\beta_A = 0.9$) for the remaining instances. For the defender, the probabilities are now concentrated on $\beta = [0.05, 0.06]$; in particular, D should use $\phi_{C_{trn}, \mathcal{Z}_{trn}^{0.05}}$ and $\phi_{C_{trn}, \mathcal{Z}_{trn}^{0.06}}$, with probability of 0.551 and 0.449, respectively.

In both cases, and especially in the case of non-uniform quantization, the shape of the equilibrium strategy of A confirms the necessity for the steganographer to find a good *trade-off* between hiding the adversarial embedding on one hand ($\beta = 0$), and trying to force the classifier towards a wrong decision (adversarial stego detected as a cover) by using a large β, on the other.

6.3. Worst case solution

If the steganographer and the steganalyst decide to adopt a worst case approach, as described in Section 5.2.2, we obtain the following results:

- for the case of uniform quantization, the worst case strategy for A is $\hat{\beta}_A = 0.10$ yielding a worst case payoff $\hat{P}_{md} = 31.2\%$ (achieved when $\beta_D = 0.05$), which is highlighted with double-underline in Table 2, while that for D is $\hat{\beta}_D = 0.05$, for which the worst case payoff is $\hat{P}_{md} = 37.8\%$ (achieved when $\beta_A = 0.75$ and highlighted with underline in Table 1);

- for the case of non-uniform quantization, the worst case strategy for A is the profile $\hat{\beta}_A = 0.04$, or 0.05 (the worst case payoff is $\hat{P}_e = 30.6\%$), (highlighted with double-underline in Table 3), while that for D is $\hat{\beta}_D = 0.06$, with a worst case payoff $\hat{P}_e = 32.1\%$ (highlighted with underline in Table 3).

Therefore, in both cases, we have $\hat{P}_e > P_e^*$ and $\hat{P}_{md} < P_{md}^*$; hence, as expected, for both D and A the worst case solution leads to a smaller (significantly smaller in some cases) payoff compared to the Nash equilibrium, thus confirming the benefit of adopting the Nash equilibrium solution.

Our results show that the quantization of β plays an important role to determine the result of the game. Notably, the results do not change significantly by using a finer quantization with respect to the one considered in the set $\mathcal{S}_{nu}^q$, thus indicating that the sampling in $\mathcal{S}_{nu}^q$ is already dense enough.

6.4. Performance of a steganalyzer trained over a mixture of β_D

In the previous section, we have obtained experimentally the mixed strategy Nash equilibriums for the $ASED$ game, where β can take values in a finite set of values. We considered a situation wherein the steganalyzer was trained on a mixture of adversarial stego images, generated with different β. Two compositions for the training set were considered:

1. $\mathcal{Z}_{trn}^{\mathrm{uni}}$: this set was consisted of adversarial images with β distributed according to a uniform distribution. For example, in the case of uniform quantization of β, 21 values of β were possible, then each β contributed with about 190 images (the whole training set consisting of 4000 images);

Table 5: The mixed strategy equilibrium for uniform quantization on β with a step of 0.05. The corresponding payoff matrices $\mathbf{P}_e$ and $\mathbf{P}_{md}$ were shown in Table 1 and 2, respectively.

β	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
$p_A^*(\beta_A)$	0.476	0.140	0	0	0	0	0	0	0	0	0	0	0	0	0	0.384	0	0	0	0	0
$p_D^*(\beta_D)$	0	0.812	0.040	0	0	0	0	0	0	0	0.148	0	0	0	0	0	0	0	0	0	0
P_{md}^*											42.9%										
P_e^*											30.8%										

Table 6: The mixed strategy equilibrium for non-uniform quantization on β. The corresponding payoff matrices $\mathbf{P}_e$ and $\mathbf{P}_{md}$ were shown in Table 3 and 4, respectively.

β	0.00	0.02	0.04	0.05	0.06	0.07	0.08	0.10	0.12	0.14	0.16	0.18	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
$p_A^*(\beta_A)$	0.767	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.233	0
$p_D^*(\beta_D)$	0	0	0	0.551	0.449	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P_{md}^*					36.3%																
P_e^*					29.0%																

Table 7: Performance of the steganalyzers trained on mixtures of adversarial stego images with different β, for the case of uniform and non-uniform quantization on β.

	uniform quantization		non-uniform quantization	
	$\mathcal{Z}_{trn}^{uni}$	$\mathcal{Z}_{trn}^{p_D^*}$	$\mathcal{Z}_{trn}^{uni}$	$\mathcal{Z}_{trn}^{p_D^*}$
$\widetilde{P}_e$	26.0	**20.7**	30.2	**28.8**

2. $\mathcal{Z}_{trn}^{p_D^*}$: this set was consisted of adversarial images with β distributed according to the probability distribution p_D^* at the equilibrium of the $ASED$ game. For example, in the case of uniform quantization, the training set contained 3252, 120, and 628 adversarial images with $\beta = 0.05, 0.10$, and 0.50, respectively (see Table 5).

The adversarial stego images and their cover counterparts formed the training set for the steganalyzer. The validation sets were composed by images in the same proportion. The same test set introduced in Section 5.1 was used for testing. The results of the tests are given in Table 7 for the case of uniform and non-uniform quantization. For each case, the table reports the value of the error probability weighted according to the probability distribution p_A^* at the equilibrium, indicated by $\widetilde{P}_e$, achieved by the classifiers $\phi_{C_{trn},\mathcal{Z}_{trn}^{uni}}$ and $\phi_{C_{trn},\mathcal{Z}_{trn}^{p_D^*}}$.

In the case of uniform quantization of β, it can be observed that the steganalyzer trained with $\{C_{trn}, \mathcal{Z}_{trn}^{p_D^*}\}$ leads to the best performance for D. Moreover, we observe that, when using $\{C_{trn}, \mathcal{Z}_{trn}^{p_D^*}\}$ for training, the resulting $\widetilde{P}_e = 20.7\%$ is much lower than the value at the Nash equilibrium (which is, $P_e^* = 30.8\%$),

Similarly, in the case of non-uniform quantization, it can be observed that the classifier trained on $\{C_{trn}, \mathcal{Z}_{trn}^{p_D^*}\}$ achieves the best detection performance. When the attacker plays the equilibrium strategy, the steganalyzer trained with $\{C_{trn}, \mathcal{Z}_{trn}^{p_D^*}\}$ achieves better performance with respect to the Nash equilib-

rium ($\widetilde{P}_e = 28.8\%$ and $P_e^* = 29.0\%$, respectively), though the improvement in this case is not as strong as in the previous case.

Overall, the above results confirm that D can get an advantage by training the steganalyzer with a mixture of images distributed as in the distribution provided by the Nash equilibrium strategy. Following this observation, it would be very interesting to see what happens when A changes his strategy, with respect to the equilibrium mixture, as a reaction (knowing about the mixed training adopted by D). This would naturally lead to the definition of a more general game where the set of strategies for A and D are the possible mixtures or distributions over the β values (rather than the pure β values, as in the definition of the $ASED$ game). Obviously, the computational burden of these tests is enormous, due to the need of training over all the possible mixtures, or, at least, a wide variety of them. The investigation of this case is left as a future work.

7. Conclusions

We formulated the interplay between CNN-based steganalysis and adversarial embedding as a two-player non-zero-sum strategic game. In particular, the game is played on the value of the parameter β, ruling the amount of adjustable elements set by the steganographer during the attack (strength of the adversarial embedding), which the stenaganalyst tries to guess and use for adversarial training.

We have shown that the solution of the non zero-sum game can be traced back to the solution of an associated zero-sum game, for which the Nash equilibrium can be derived more easily. The experiments that we run in a practical setup of CNN-based steganalysis and adversarial embedding provide the optimum behavior for the steganalyst and the steganographer and show the performance that can be achieved by playing at the Nash equilibrium, thus characterizing the loss in the error probability with respect to the case of conventional non-adversarial steganography. From our experiments, we can also observe that

the quantization of the parameter β plays an important role in the game solution, affecting the performance at the equilibrium. As a further result, we verified that an improved solution for the steganalyst, i.e., yielding a lower error probability, can be obtained by training the classifier with the mixture of β values provided by the steganalyst's equilibrium strategy of the game. This is an interesting result that goes beyond the game analysis considered in this paper, and thus calls for more investigation as a future work.

Other games could be defined by considering different sets of strategies for the two players. For example, the selection-channel information is a kind of strategy that both parties may utilize. In this respect, it would be interesting to investigate the interplay between selection-channel steganalysis and content-adaptive steganography from a game-theoretic perspective.

References

[1] J. Fridrich, Steganography in digital media: principles, algorithms, and applications, Cambridge University Press, 2009. `doi:10.1017/CBO9781139192903`.

[2] B. Li, J. He, J. Huang, Y. Q. Shi, A survey on image steganography and steganalysis, Journal of Information Hiding and Multimedia Signal Processing 2 (2) (2011) 142–172.

[3] J. Fridrich, T. Filler, Practical methods for minimizing embedding impact in steganography, in: Security, Steganography, and Watermarking of Multimedia Contents IX, Vol. 6505, International Society for Optics and Photonics, 2007, p. 650502. `doi:10.1117/12.697471`.

[4] T. Pevný, T. Filler, P. Bas, Using high-dimensional image models to perform highly undetectable steganography, in: Information Hiding, Springer Berlin Heidelberg, 2010, pp. 161–177. `doi:10.1007/978-3-642-16435-4_13`.

[5] T. Pevny, P. Bas, J. Fridrich, Steganalysis by subtractive pixel adjacency matrix, IEEE Transactions on information Forensics and Security 5 (2) (2010) 215–224. `doi:10.1109/tifs.2010.2045842`.

[6] V. Holub, J. Fridrich, Designing steganographic distortion using directional filters, in: 2012 IEEE International workshop on information forensics and security, IEEE, 2012, pp. 234–239. `doi:10.1109/WIFS.2012.6412655`.

[7] V. Holub, J. Fridrich, T. Denemark, Universal distortion function for steganography in an arbitrary domain, EURASIP Journal on Information Security 2014 (1) (2014) 1. `doi:10.1186/1687-417X-2014-1`.

[8] B. Li, M. Wang, J. Huang, X. Li, A new cost function for spatial image steganography, in: 2014 IEEE International Conference on Image Processing, IEEE, 2014, pp. 4206–4210. `doi:10.1109/ICIP.2014.7025854`.

[9] J. Fridrich, J. Kodovský, Multivariate gaussian model for designing additive distortion for steganography, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2013, pp. 2949–2953. `doi:10.1109/ICASSP.2013.6638198`.

[10] V. Sedighi, J. Fridrich, R. Cogranne, Content-adaptive pentary steganography using the multivariate generalized gaussian cover model, in: Media Watermarking, Security, and Forensics 2015, Vol. 9409, International Society for Optics and Photonics, 2015, p. 94090H. `doi:10.1117/12.2080272`.

[11] V. Sedighi, R. Cogranne, J. Fridrich, Content-adaptive steganography by minimizing statistical detectability, IEEE Transactions on Information Forensics and Security 11 (2) (2016) 221–234. `doi:10.1109/TIFS.2015.2486744`.

[12] X. Qin, B. Li, J. Huang, A new spatial steganographic scheme by modeling image residuals with multivariate gaussian model, in: 2019 IEEE International Conference on Acoustics, Speech and Signal Processing, 2019.

[13] B. Li, M. Wang, X. Li, S. Tan, J. Huang, A strategy of clustering modification directions in spatial image steganography, IEEE Transactions on Information Forensics and Security 10 (9) (2015) 1905–1917. `doi:10.1109/TIFS.2015.2434600`.

[14] T. Denemark, J. Fridrich, Improving steganographic security by synchronizing the selection channel, in: Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security, ACM, 2015, pp. 5–14. `doi:10.1145/2756601.2756620`.

[15] W. Zhou, W. Zhang, N. Yu, A new rule for cost reassignment in adaptive steganography, IEEE Transactions on Information Forensics and Security 12 (11) (2017) 2654–2667. `doi:10.1109/TIFS.2017.2718480`.

[16] J. Fridrich, J. Kodovsky, Rich models for steganalysis of digital images, IEEE Transactions on Information Forensics and Security 7 (3) (2012) 868–882. `doi:10.1109/tifs.2012.2190402`.

[17] W. Tang, H. Li, W. Luo, J. Huang, Adaptive steganalysis against wow embedding algorithm, in: Proceedings of the 2nd ACM workshop on Information hiding and multimedia security, ACM, 2014, pp. 91–96. `doi:10.1145/2600918.2600935`.

[18] T. Denemark, V. Sedighi, V. Holub, R. Cogranne, J. Fridrich, Selection-channel-aware rich model for steganalysis of digital images, in: 2014 IEEE International Workshop on Information Forensics and Security, IEEE, 2014, pp. 48–53. `doi:10.1109/WIFS.2014.7084302`.

[19] V. Holub, J. Fridrich, Random projections of residuals for digital image steganalysis, IEEE Transactions on Information Forensics and Security 8 (12) (2013) 1996–2006. `doi:10.1109/tifs.2013.2286682`.

[20] B. Li, Z. Li, S. Zhou, S. Tan, X. Zhang, New steganalytic features for spatial image steganography based on derivative filters and threshold LBP operator, IEEE Transactions on Information Forensics and Security 13 (5) (2018) 1242–1257. `doi:10.1109/TIFS.2017.2780805`.

[21] S. Tan, B. Li, Stacked convolutional auto-encoders for steganalysis of digital images, in: Signal and Information Processing Association Annual Summit and Conference, 2014 Asia-Pacific, IEEE, 2014, pp. 1–4. `doi:10.1109/APSIPA.2014.7041565`.

[22] Y. Qian, J. Dong, W. Wang, T. Tan, Deep learning for steganalysis via convolutional neural networks, in: Media Watermarking, Security, and Forensics 2015, Vol. 9409, International Society for Optics and Photonics, 2015, p. 94090J. `doi:10.1117/12.2083479`.

[23] G. Xu, H.-Z. Wu, Y.-Q. Shi, Structural design of convolutional neural networks for steganalysis, IEEE Signal Processing Letters 23 (5) (2016) 708–712. `doi:10.1109/LSP.2016.2548421`.

[24] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37, JMLR. org, 2015, pp. 448–456.

[25] J. Ye, J. Ni, Y. Yi, Deep learning hierarchical representations for image steganalysis, IEEE Transactions on Information Forensics and Security 12 (11) (2017) 2545–2557. `doi:10.1109/TIFS.2017.2710946`.

[26] M. Boroumand, M. Chen, J. Fridrich, Deep residual network for steganalysis of digital images, IEEE Transactions on Information Forensics and Security (2018) 1–1`doi:10.1109/TIFS.2018.2871749`.

[27] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, Computer Science.

[28] Y. Zhang, W. Zhang, K. Chen, J. Liu, Y. Liu, N. Yu, Adversarial examples against deep neural network based steganalysis, in: Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security, ACM, 2018, pp. 67–72. `doi:10.1145/3206004.3206012`.

[29] S. Li, D. Ye, S. Jiang, C. Liu, X. Niu, X. Luo, Attack on deep steganalysis neural networks, in: International Conference on Cloud Computing and Security, Vol. 11066, 2018.

[30] W. Tang, B. Li, S. Tan, M. Barni, J. Huang, CNN-based adversarial embedding for image steganography, IEEE Transactions on Information Forensics and Security (2019) 1–1`doi:10.1109/TIFS.2019.2891237`.

[31] M. Barni, M. C. Stamm, B. Tondi, Adversarial multimedia forensics: Overview and challenges ahead, in: 2018 26th European Signal Processing Conference, IEEE, 2018, pp. 962–966. `doi:10.23919/EUSIPCO.2018.8553305`.

[32] A. D. Ker, Batch steganography and the threshold game, in: Security, Steganography, and Watermarking of Multimedia Contents IX, Vol. 6505, International Society for Optics and Photonics, 2007, p. 650504. `doi:10.1117/12.703334`.

[33] E. Franz, Steganography preserving statistical properties, in: International Workshop on Information Hiding, Springer, 2002, pp. 278–294. `doi:10.1007/3-540-36415-3_18`.

[34] P. Schöttle, R. Böhme, A game-theoretic approach to content-adaptive steganography, in: International Workshop on Information Hiding,

Springer, 2012, pp. 125–141. doi:10.1007/978-3-642-36373-3_9.

[35] B. Johnson, P. Schöttle, R. Böhme, Where to hide the bits?, in: International Conference on Decision and Game Theory for Security, Springer, 2012, pp. 1–17. doi:10.1007/978-3-642-34266-0_1.

[36] T. Denemark, J. Fridrich, Detection of content adaptive lsb matching: a game theory approach, in: Proceedings of SPIE - The International Society for Optical Engineering, Vol. 9028, International Society for Optics and Photonics, 2014, pp. 30–54. doi:10.1117/12.2040418.

[37] A. S. Cohen, A. Lapidoth, The gaussian watermarking game, IEEE Transactions on Information Theory 48 (6) (2002) 1639–1667. doi:10.1109/tit.2002.1003844.

[38] M. C. Stamm, W. S. Lin, K. R. Liu, Forensics vs. anti-forensics: A decision and game theoretic framework, in: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2012, pp. 1749–1752. doi:10.1109/ICASSP.2012.6288237.

[39] M. Barni, B. Tondi, The source identification game: An information-theoretic perspective, IEEE Transactions on Information Forensics and Security 8 (3) (2013) 450–463. doi:10.1109/TIFS.2012.2237397.

[40] M. Barni, B. Tondi, Binary hypothesis testing game with training data, IEEE Transactions on Information Theory 60 (8) (2014) 4848–4866. doi:10.1109/TIT.2014.2325571.

[41] M. Barni, B. Tondi, Adversarial source identification game with corrupted training, IEEE Transactions on Information Theory 64 (5) (2018) 3894–3915. doi:10.1109/TIT.2018.2806742.

[42] A. Abrardo, M. Barni, K. Kallas, B. Tondi, A game-theoretic framework for optimum decision fusion in the presence of byzantines, IEEE Transactions on Information Forensics and Security 11 (6) (2016) 1333–1345. doi:10.1109/TIFS.2016.2526963.

[43] R. B. Myerson, Game theory, Harvard university press, 2013. doi:10.1155/3816.

[44] M. J. Osborne, A. Rubinstein, A Course in Game Theory, MIT Press, 1994.

[45] M. J. Osborne, An introduction to game theory, Vol. 3, Oxford University Press New York, 2004.

[46] J. v. Neumann, Zur theorie der gesellschaftsspiele, Mathematische annalen 100 (1) (1928) 295–320.

[47] I. Adler, The equivalence of linear programs and zero-sum games, International Journal of Game Theory 42 (1) (2013) 165–177.

[48] T. Filler, J. Judas, J. Fridrich, Minimizing additive distortion in steganography using syndrome-trellis codes, IEEE Transactions on Information Forensics and Security 6 (3) (2011) 920–935. doi:10.1109/tifs.2011.2134094.

[49] J. Nash, Equilibrium points in n-person games, Proceedings of the National Academy of Sciences 36 (1) (1950) 48–49. doi:10.1515/9781400884087-007.

[50] C. Daskalakis, P. W. Goldberg, C. H. Papadimitriou, The complexity of computing a nash equilibrium, SIAM Journal on Computing 39 (1) (2009) 195–259.

[51] A. Charnes, W. W. Cooper, Management models and industrial applications of linear programming, Management Science 4 (1) (1957) 38–91. doi:10.1287/mnsc.4.1.38.

[52] P. Bas, T. Filler, T. Pevnỳ, Break our steganographic system: the ins and outs of organizing boss, in: International workshop on information hiding, Springer, 2011, pp. 59–70. doi:10.1007/978-3-642-24178-9_5.

[53] MathWorks, Optimization toolbox, [Online]. Available: http://it.mathworks.com/help/optim/.