


Game-theoretic Analysis to Parameterized Reversible Watermarking

Hanzhou Wu  and Xinpeng Zhang

Shanghai University, Shanghai 200444, People's Republic of China

ABSTRACT

Separable reversible watermarking enables two encoders to separately embed a payload in a cover, and the original cover can be reconstructed by cooperation. It is required to limit the embedding distortion for the two encoders so that the marked content will not be seriously degraded, whereas both encoders expect to embed a sufficient payload. It motivates us to present a two-encoder game related to rate-distortion optimization of reversible watermarking. We investigate non-cooperative game and cooperative game in a parameterized perspective, which provides good generalization. We find the equilibrium strategies for both encoders under constraints. We extend the game to a multistage win-or-lose case, where both encoders want to win the game. We model the game on a rooted tree with unbounded branching and identify the winner once the initial state is given.

KEY WORDS

Adaptive; Data embedding; Game theory; Lossless compression; Reversible data hiding; Watermarking

1. INTRODUCTION

Information hiding is referred to as the art of embedding secret data into a digital object such as image and video, typically also called *cover*, without introducing significant degradation of the cover object. It can be organized into two categories, i.e. *non-invertible embedding* and *invertible embedding*. The former permanently distorts the cover whereas the latter allows the original cover content to be restored after the embedded information is fully extracted.

Invertible embedding, or called *reversible data hiding* [1], *reversible watermarking* (RW) [2, 3], enables digital objects such as images and videos to be authenticated and restored to their original versions, making it quite suitable for sensitive purposes, e.g. the US army is interested in such kind of promising technology for authentication of reconnaissance images. RW is fragile, meaning that, one will find it is not authentic if the marked data was tampered. RW could be evaluated by the rate-distortion behavior. That is, we often expect to embed as many secret message bits as possible for a fixed distortion. In other words, we hope to reduce the embedding distortion as much as possible for a given payload.

A straightforward idea to realize RW is applying lossless compression (LC) [3], e.g. an encoder can select the noise-like component of the cover such as the LSBs of a grayscale image, and then losslessly compress them to reserve additional space, into which the secret message will be embedded. Though altering the noise-like

component does not introduce noticeable visual artifacts, the pure embedding capacity is limited due to the low compression rate. To this end, more efficient methods such as difference expansion (DE) [4], histogram shifting (HS) [1], and variants such as prediction-error expansion [2, 5, 6], are proposed to enlarge capacity or reduce distortion. It can be said that, current state-of-the-arts use HS or its variants. We will not review these works since it is not the main interest of this paper.

Though current advanced RW algorithms differ from LC intuitively, they are essentially similar to each other. They are all finding the compressible component of the cover to carry the secret data. The difference is, LC considers the cover elements as compressible variables, while the others exploit the spatial correlations between the cover elements for RW, which can be regarded as a kind of “semantic lossless compression” [7]. As mentioned above, strong correlations exist between the cover elements, which allows us to use prediction errors (PEs). A common operation is then to generate a PE histogram (PEH). On the one hand, the slight modification to PEs will not introduce serious degradation of the cover. On the other hand, as a pooled vector, a PEH allows us to effectively embed data by HS. Many works are developed along this line. In addition, HS based RW can be characterized by HS parameters. That is, given a PEH, both the embedded payload size and the introduced distortion can be estimated by HS parameters. When the way to determine a PEH is specified, the distortion optimization task is reduced to finding optimal parameters.

From the theoretical perspective, studying LC based RW is desirable though people have developed many new practical algorithms. The theoretical results could be helpful to guide us to design new systems, which has motivated us to revisit LC based RW in this paper. Different from traditional works, we use game theory to analyze RW. Unlike previous games designed to robust or irreversible watermarking games [8–10], in our game, two players: Alice and Bob, want to embed a different payload into a given cover. It differs from previous systems, where only one encoder is considered. However, our work can be more general. The reason is, Alice and Bob could cooperate with each other. In this way, one can consider them as a whole, indicating that, an encoder will use a cover for embedding with two times, which corresponds to *multi-layer embedding*. When they do not cooperate with each other, the game implies a *separable* system, i.e. both independently hide a payload within the cover, but recovering the cover may need cooperation. In extension study, we reformulate a multistage win-or-lose game, allowing Alice and Bob to alternately modify the cover. The one who first fully embed his/her own payload will be the winner. Since we study it in a parametric way, which skips the payoff functions and thus has better generalization.

The rest is organized as follows. We first review prior arts in Sect. 2. We then introduce our game setup in Sect. 3. We analyze the non-cooperative and cooperative games in Sect. 4. In Sect. 5, we formulate a win-or-lose game, for which we want to identify the winner of a game. We finally conclude this paper in Sect. 6.

2. PRIOR ARTS

Though steganography should be distinguished from watermarking, they are correlated to each other. The first work combining game theory and steganography was probably proposed by [8]. In the work, a zero-sum game between the data hider and the attacker was presented, for which the equilibrium is such a strategy profile that none of the two players wants to profit from adjusting two different partial strategies, for a fixed strategy of the opponent. Moulin and O’Sullivan [9] characterized the watermarking codes as a capacity game between data hider and attacker, where two models were presented. Baruch and Merhav [10] investigated capacity and error exponent games, in which the attack channel is completely general and unknown to the hider and receiver. Cohen and Lapidot [11] derived a capacity formula of the watermarking game for a Gaussian cocontext and squared error distortions. It showed that the capacities of public and private watermarking are the same, which is

in some analogy to Costa’s result on channel coding with side information under the Gaussian quadratic regime [12]. Baruch and Merhav [13] extended the public game [9] by dropping the assumptions that the receiver knows the attack channel and the channel is either memoryless or block-wise memoryless.

Ker [14] introduced a threshold game to batch steganography [15], in which the data hider should decide how to distribute a secret payload into multiple pieces each embedded into a selective cover. The result indicated that, the optimal strategy is likely to be extreme concentration of the payload into as few covers as possible, or the payload is spread as thinly as possible. Schöttle and Böhme [16] defined *heterogeneity* as a necessary condition for adaptive steganography, and presented a game model to the whole process including cover generation, adaptive embedding, and a detector which anticipates adaptivity. Though the model exhibited a unique equilibrium in mixed strategies, it investigated a cover model with only two locations. To this end, Schöttle and Böhme [17] extended the model from the very artificial case to covers sized n . Their results are constructive in the sense that an equilibrium can be efficiently found for any vector of predictability. More games designed for steganography can be found in [18–20].

3. GAME SETUP

3.1 Description of the Cover

The cover here is written as $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_l)$, where $\mathbf{x}_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,n}\}$ represents a binary sequence with the distribution $p_{i,1} = (x_{i,1} + x_{i,2} + \dots + x_{i,n}) / n$ and $p_{i,0} = 1 - p_{i,1}$. It is assumed that, for any two different integers i and j , there has no intersection between \mathbf{x}_i and \mathbf{x}_j . We point that, the term “cover” actually means the data embedding channel. For example, if we use the LSBs of an image for data embedding, we can separate the LSB plane out from the image and divide it into disjoint bit-vectors to constitute \mathbf{X} . On the one hand, such definition has good generalization for lossless compression based RW. On the other hand, it allows us to well model the game between two encoders latter.

3.2 Rate-distortion Game

In order to achieve superior performance, the secret bits should be preferentially embedded in the smooth area of the cover. Since we use binary sequences, the smoothness is defined as information entropy. The smaller the entropy, the more the embeddable bits, implying better rate-distortion performance. One could define

other functions to evaluate the smoothness. We define the smoothness of \mathbf{x}_i as $H(p_i)$, where $p_i = \min\{p_{i,1}, p_{i,0}\} \leq 1/2$, and $H(p_i) = -p_i \log_2 p_i - (1 - p_i) \log_2 (1 - p_i)$ means the binary entropy. Without the loss of generality, we assume that $H(p_1) \leq H(p_2) \leq \dots \leq H(p_l)$, meaning that, \mathbf{x}_i is more suitable for RW than \mathbf{x}_j , for any $i < j$.

Alice first hides a payload by modifying the bits throughout \mathbf{X} under the constraint that the total amount of distortion is no more than d . A strategy for Alice is an l -tuple of probabilities (s_1, s_2, \dots, s_l) , where $s_i \in [0, 1]$, $i = 1, 2, \dots, l$. It indicates that, for each $1 \leq i \leq l$, Alice chooses $s_i \times 100\%$ pixels from \mathbf{x}_i with a secret key and losslessly compress the bits. The bit-positions will carry the compressed code and the secret payload. The resulting data \mathbf{Y} will be sent to Bob, who embeds another payload into \mathbf{Y} by the same means subjected to d as well, resulting in another marked object \mathbf{Z} . A strategy for Bob is therefore another l -tuple of probabilities (t_1, t_2, \dots, t_l) . Though two strategies are real vectors, the number of bit-positions is an integer. We use real numbers for better analysis. With \mathbf{Z} , Bob can directly retrieve the hidden data and his compressed code. However, Alice should correct the errors produced by Bob after data extraction. For Alice, the compressed code for \mathbf{x}_i needs $ns_i H(p_i)$ bits. As the channel capacity for a binary symmetric channel with bit error probability p is $1-H(p)$, the size of embeddable payload of \mathbf{x}_i in bits for Alice is

$$A_i(s_i, t_i) = ns_i(1 - H(t_i/2) - H(p_i)). \quad (1)$$

And that for Bob is

$$B_i(s_i, t_i) = nt_i(1 - H(p_i + s_i/2 - p_i s_i)). \quad (2)$$

Here, $H(p_i + s_i/2 - p_i s_i)$ shows the binary entropy of \mathbf{x}_i after data embedding by Alice. It can be determined as follows. Suppose that, initially, the distribution is {"0": p_i , "1": $1-p_i$ }, after data embedding by Alice, the probability for "0" is changed to $p_i(1-s_i/2) + (1-p_i)s_i/2$, i.e. $p_i + s_i/2 - p_i s_i$. Thus, we obtain the entropy mentioned above.

A strategy profile $(\mathbf{s}, \mathbf{t}) = (s_1, s_2, \dots, s_l, t_1, t_2, \dots, t_l)$ for Alice and Bob gives payoffs:

$$P_A(\mathbf{s}, \mathbf{t}) = \sum_{i=1}^l A_i(s_i, t_i), \quad (3)$$

$$P_B(\mathbf{s}, \mathbf{t}) = \sum_{i=1}^l B_i(s_i, t_i). \quad (4)$$

Both want to maximize their own payoff, subjected to a bounded distortion. Since data embedding in the smooth area gives a larger capacity, both will absolutely embed their own payload into \mathbf{x}_i with a small index value if there has no opponent. Thus, as limited and precious resource, the cost of data embedding in \mathbf{x}_i will be surely higher than that of \mathbf{x}_j for any $i < j$ for both players. Let $\rho(\mathbf{x}_i)$ be the cost of flipping any individual bit in \mathbf{x}_i . We can assume that

$$\rho(\mathbf{x}_i) \propto 1/H(p_i), \quad \forall i \in [1, l]. \quad (5)$$

We will limit us to additive distortion, i.e.

$$D_A(\mathbf{s}) = \sum_{i=1}^l \frac{ns_i}{2} \rho(\mathbf{x}_i) \leq d, \quad (6)$$

$$D_B(\mathbf{t}) = \sum_{i=1}^l \frac{nt_i}{2} \rho(\mathbf{x}_i) \leq d. \quad (7)$$

By using different ρ , sophisticated models may be accommodated in this game. Though there has an embedding order for two players, they both know complete information about the game except the key held by the opponent.

4. NON-COOPERATIVE AND COOPERATIVE GAME

4.1 Non-cooperative Game

When they do not cooperate with each other, an *equilibrium* is such a strategy profile $(\mathbf{s}^*, \mathbf{t}^*) = (s_1^*, s_2^*, \dots, s_l^*, t_1^*, t_2^*, \dots, t_l^*)$ that

$$P_A(\mathbf{s}, \mathbf{t}^*) \leq P_A(\mathbf{s}^*, \mathbf{t}^*), \quad (8)$$

$$P_B(\mathbf{s}^*, \mathbf{t}) \leq P_B(\mathbf{s}^*, \mathbf{t}^*). \quad (9)$$

4.1.1 Case $l = 1$

We first analyze the case $l = 1$. It can be easily obtained from Equations (6) and (7) that

$$s_1 \leq p_{\max} = \min \left\{ 1, \frac{2d}{n\rho(\mathbf{x}_1)} \right\} \quad \text{and} \quad t_1 \leq p_{\max}. \quad (10)$$

It is observed from Equation (2) that, no matter what strategy Alice takes, Bob will always choose $t_1 = p_{\max}$ since it achieves the maximum payoff. For a fixed t_1 , the optimal response for Alice will be $s_1 = p_{\max}$. Thus, the equilibrium is uniquely $(\mathbf{s}^*, \mathbf{t}^*) = (p_{\max}, p_{\max})$. Figure 1 shows the equilibrium payoffs due to different p_{\max} . It is observed that, the payoff of Bob is never less than Alice.

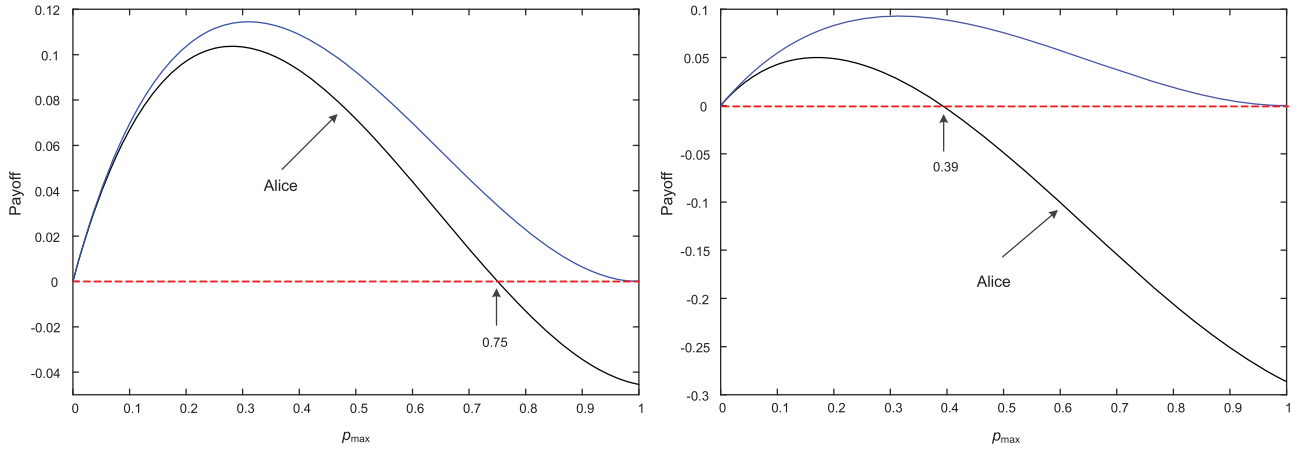


Figure 1: Payoffs (divided by n) due to different ρ_{\max} . Left: $p_1 = 0.005$, right $p_1 = 0.05$.

Since Alice needs correct errors produced by Bob, when the distortion threshold is larger than a value, Alice may not embed any extra bits (i.e. the payoff is negative), e.g. in Figure 1, when $\rho_{\max} > 0.75$ for the left case, Alice cannot embed a pure payload.

4.1.2 Case $l = 2$

Let $T_i = 1 - H(t_i/2) - H(p_i)$ and $S_i = 1 - H(p_i + s_i/2 - p_i s_i)$, we have

$$P_A(\mathbf{s}, \mathbf{t}) = \sum_{i=1}^l n s_i T_i \quad \text{and} \quad P_B(\mathbf{s}, \mathbf{t}) = \sum_{i=1}^l n t_i S_i. \quad (11)$$

Neither Alice nor Bob wants to deviate from an equilibrium. In case $l = 2$, to find an equilibrium, we must ensure that Bob does not profit from readjusting t_1 and t_2 for a fixed \mathbf{s} . Similarly, Alice does not profit from readjusting s_1 and s_2 for a fixed \mathbf{t} . Therefore, for any i , we have

$$\frac{\partial P_B(\mathbf{s}, \mathbf{t})}{\partial t_i}(t_i^*) = 0 \quad \text{and} \quad \frac{\partial P_A(\mathbf{s}, \mathbf{t})}{\partial s_i}(s_i^*) = 0, \quad (12)$$

from which we can obtain:

$$\frac{S_1}{S_2} = \frac{T_1}{T_2} = \frac{\rho(\mathbf{x}_1)}{\rho(\mathbf{x}_2)}. \quad (13)$$

When $\rho(\mathbf{x}_1)/\rho(\mathbf{x}_2)$ is fixed, a lot of strategy profiles satisfy Equation (13). For the “sub-cover” \mathbf{x}_i , let d_i^A and d_i^B be the distortion bounds, i.e.

$$n s_i \rho(\mathbf{x}_i)/2 \leq d_i^A \quad \text{and} \quad n t_i \rho(\mathbf{x}_i)/2 \leq d_i^B. \quad (14)$$

The optimal “sub-strategy” profile is:

$$\left(\min \left\{ 1, \frac{2d_i^A}{n\rho(\mathbf{x}_i)} \right\}, \min \left\{ 1, \frac{2d_i^B}{n\rho(\mathbf{x}_i)} \right\} \right)$$

It implies that, the optimal response for an individual player is only dependent of his or her own distortion constraint. And, the maximum distortion for a sub-cover gives the maximum “sub-payoff”. $(\mathbf{s}^*, \mathbf{t}^*)$ can be easily determined by moving an initial strategy profile towards the corresponding direction until the distortion bound is reached. We take Figure 2 for example, where $p_1 = 0.005$, $p_2 = 0.05$, $\rho(\mathbf{x}_1)/\rho(\mathbf{x}_2) = 2$. The left draws out all strategies that meet Equation (13). It reduces the strategy space for determining the equilibrium. By sampling a set of points with a small step (in our simulation, we set its value as 0.1×10^3) on both curves, we produce two strategy sequences, each element in which can be orderly indexed by a value. Thus, we can draw out all strategy profiles as the right figure shown in Figure 2. It is observed that, fixing the strategy of either player, the payoff curve of the other player is a strictly monotone increasing function. It means that, the equilibrium will be such a profile that maximizes the distortion, which can be determined during the process of moving from a start point to the end point, as shown in the left figure. We always force Bob to allow Alice to embed a non-negative pure payload in arbitrary sub-cover since the target of Bob is not to attack Alice, but rather aim to maximize his own payoff. Thus, we have

$$1 - H(t_i^*/2) \geq H(p_i), \quad \forall i \in [1, l]. \quad (15)$$

4.1.3 Case $l > 2$

Similarly, in case $l > 2$, neither Alice nor Bob deviates from an equilibrium. We fix all components of \mathbf{t} except for t_j and t_k ($j \neq k$), and distribute the distortion

$$d_B^{j,k} = d - \sum_{i \neq j,k} n t_i \rho(\mathbf{x}_i)/2 \quad (16)$$

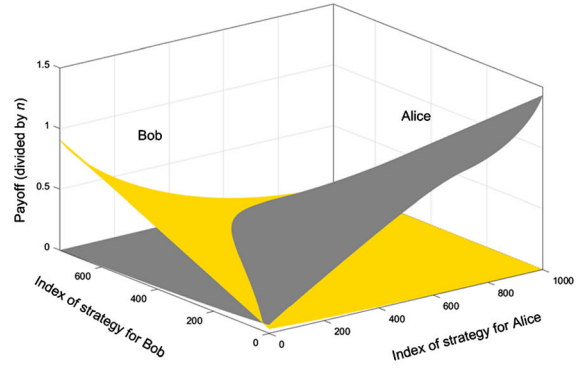
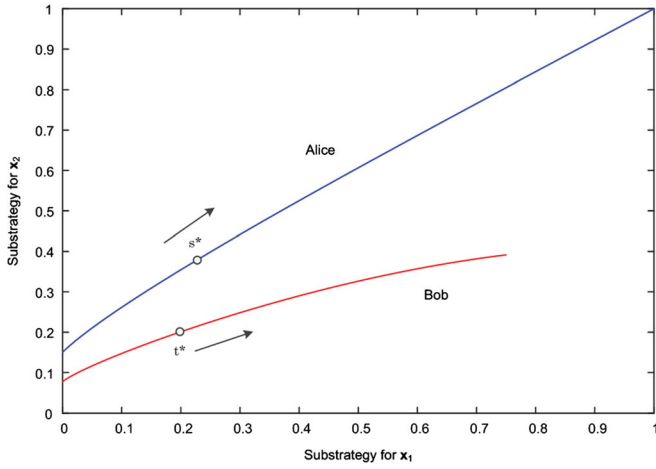


Figure 2: An example of equilibrium in case $l = 2$: $p_1 = 0.005, p_2 = 0.05, \rho(\mathbf{x}_1)/\rho(\mathbf{x}_2) = 2$.

between these two components. Let

$$B_{j,k}(t_j) = nt_j S_j + nt_k S_k = nt_j S_j + \frac{2d_B^{j,k} - nt_j \rho(\mathbf{x}_j)}{\rho(\mathbf{x}_k)} S_k. \quad (17)$$

To find an equilibrium, we must ensure that Bob does not profit from readjusting t_j and t_k for a fixed \mathbf{s} . Therefore, we have

$$\frac{\partial B_{j,k}}{\partial t_j}(t_j^*) = 0 \quad (18)$$

for an equilibrium t_j^* . We continue to fix all the components of \mathbf{s} except for s_j and s_k where $1 \leq j \leq k \leq l$, and distribute the remaining distortion

$$d_A^{j,k} = d - \sum_{i \neq j,k} ns_i \rho(\mathbf{x}_i)/2 \quad (19)$$

between these two components. Let

$$\begin{aligned} A_{j,k}(s_j) &= ns_j T_j + ns_k T_k \\ &= ns_j T_j + \frac{2d_A^{j,k} - ns_j \rho(\mathbf{x}_j)}{\rho(\mathbf{x}_k)} T_k. \end{aligned} \quad (20)$$

Since Alice does not profit from readjusting s_j and s_k , we have

$$\frac{\partial A_{j,k}}{\partial s_j}(s_j^*) = 0 \quad (21)$$

for s_j^* . Thus, according to Equations (18) and (21), we have

$$\frac{S_j}{S_k} = \frac{T_j}{T_k} = \frac{\rho(\mathbf{x}_j)}{\rho(\mathbf{x}_k)} \geq 1, \quad \forall j < k. \quad (22)$$

Assuming that, $S_1 = \alpha \in [0, 1]$ and $T_1 = \beta \in [0, 1]$, we have

$$S_i = \frac{\rho(\mathbf{x}_i)}{\rho(\mathbf{x}_1)} \alpha \quad \text{and} \quad T_i = \frac{\rho(\mathbf{x}_i)}{\rho(\mathbf{x}_1)} \beta. \quad (23)$$

It is inferred that, the equilibrium $(\mathbf{s}^*, \mathbf{t}^*)$ should correspond to such a pair $(\alpha^*, \beta^*) \in [0, 1]^2$ that

$$s_i^* = \frac{H^{-1}(1 - \alpha^* \rho(\mathbf{x}_i)/\rho(\mathbf{x}_1)) - p_i}{1/2 - p_i} \quad (24)$$

$$t_i^* = 2H^{-1}(1 - H(p_i) - \beta^* \rho(\mathbf{x}_i)/\rho(\mathbf{x}_1)) \quad (25)$$

And

$$\sum_{i=1}^l ns_i^* \rho(\mathbf{x}_i)/2 = \sum_{i=1}^l nt_i^* \rho(\mathbf{x}_i)/2 = d, \quad (26)$$

which can be solved by binary search since the data embedding distortion functions for both layers are strictly monotone w.r.t. (α, β) . Notice that, the binary search operation will be used twice for determining \mathbf{s}^* and \mathbf{t}^* .

4.2 Cooperative Game

When Alice and Bob cooperate with each other, the payoff can be rewritten as

$$P(\mathbf{s}, \mathbf{t}) = \min\{P_A(\mathbf{s}, \mathbf{t}), P_B(\mathbf{s}, \mathbf{t})\}, \quad (27)$$

for which an equilibrium is such a pair $(\mathbf{s}^*, \mathbf{t}^*)$ that *preferentially* meets:

$$\max\{P(\mathbf{s}^*, \mathbf{t}), P(\mathbf{s}, \mathbf{t}^*)\} \leq P(\mathbf{s}^*, \mathbf{t}^*) \quad (28)$$

and then meets:

$$\max\{P_A(\mathbf{s}^*, \mathbf{t}), P_B(\mathbf{s}, \mathbf{t}^*)\} \leq \max\{P_A(\mathbf{s}^*, \mathbf{t}^*), P_B(\mathbf{s}^*, \mathbf{t}^*)\}. \quad (29)$$

Equations (27)–(29) actually correspond to a max–min problem. That is, we expect the minimum payload between the two players is maximized, such that both can agree on the payoff design. Moreover, we hope the larger payload between them is maximized when Equation (28) is satisfied. Unlike other cooperative games, for RW, we have to further consider the side information shared between Alice and Bob. That is, we have to analyze two situations:

Case 1: They share the key of choosing the bit-positions to be embedded.

Case 2: They do not share the secret key mentioned above.

Case 1 is corresponding to two-channel embedding strategy commonly used in RW. In this case, when Alice chooses ns_i positions in \mathbf{x}_i , Bob can use the rest positions. We thus have $s_i + t_i \leq 1$ for $1 \leq i \leq l$. It indicates that, Bob will not produce extraction errors. Therefore,

$$\begin{aligned} P(\mathbf{s}, \mathbf{t}) &= \sum_{i=1}^l n \cdot \min\{s_i, t_i\} \cdot (1 - H(p_i)) \\ &\leq \sum_{i=1}^l n \cdot \frac{s_i + t_i}{2} \cdot (1 - H(p_i)), \end{aligned} \quad (30)$$

which allows

$$d \leq d_{\max} = \sum_{i=1}^l n \rho(\mathbf{x}_i) / 2. \quad (31)$$

It can be inferred that, $\mathbf{s}^* = \mathbf{t}^* = (w_1^*/2, w_2^*/2, \dots, w_l^*/2)$ will be an equilibrium. To this end, we have to solve the following task:

$$P(\mathbf{s}^*, \mathbf{t}^*) = \max_{\mathbf{w}} \sum_{i=1}^l n \cdot \frac{w_i}{2} \cdot (1 - H(p_i)) \quad (32)$$

subjected to $\mathbf{w} \in [0, 1]^l$ and

$$\sum_{i=1}^l n \cdot \frac{w_i}{4} \cdot \rho(\mathbf{x}_i) \leq d. \quad (33)$$

This is a *linear programming* problem and can be solved by the *simplex algorithm*. Thus, \mathbf{w}^* can be determined as a vertex of the corresponding high-dimensional polytope.

It is true for Case 2 that

$$P(\mathbf{s}^*, \mathbf{t}^*) \leq \sum_{i=1}^l n(1 - H(p_i)), \quad (34)$$

which allows that

$$d \leq \sum_{i=1}^l n \rho(\mathbf{x}_i) / 2. \quad (35)$$

Neither Alice nor Bob deviates from an equilibrium. It is inferred that, $(\mathbf{s}^*, \mathbf{t}^*)$ has the identical form to Equations (24) and (25). The difference is that, we do not require that Alice and Bob introduce the maximum distortion, i.e.

$$\sum_{i=1}^l \frac{ns_i^*}{2} \rho(\mathbf{x}_i) \leq d \quad \text{and} \quad \sum_{i=1}^l \frac{nt_i^*}{2} \rho(\mathbf{x}_i) \leq d. \quad (36)$$

The reason is that, they are not competing, but rather cooperating with each other, which can be seen from Equation (27). The equilibrium can be found by *triple search* or *gradient descent* since the task shows a *concave payoff function*. For gradient descent, it finds the optimal solution along the direction of gradient decent. For the triple search, also called *ternary search*, it is a divide-and-conquer algorithm that divides the given range into three parts and determines the required element, which is an extension of the binary search that divides the given range into only two parts. The problems that can be solved by binary search can be also solved by triple search. And, triple search can further deal with problems having the parabola property.

5. MULTISTAGE WIN-OR-LOSE GAME

When to extend to HS embedding, it is not easy to design the payoffs and find the equilibrium. To this end, we formulate a win-or-lose game. The two players alternately modify the cover by using well-chosen parameters. The one who first embeds his or her own payload will be the winner. It skips the design of payoff functions and therefore provides better generalization.

5.1 Game Reformulation

Let \mathbf{x}_0 be the cover sequence. Alice and Bob respectively embed a payload \mathbf{m}_A sized l_A and \mathbf{m}_B sized l_B into \mathbf{x}_0 . Alice and Bob agree on an RW algorithm E_A . They alternately modify \mathbf{x}_0 , which corresponds to *multi-layer embedding*.

Alice first plays the game. She uses E_A to embed a part of her own payload \mathbf{m}_A^1 into \mathbf{x}_0 , where the distortion is no more than d_A^1 . The resulting marked cover \mathbf{x}_1 is then processed to hide \mathbf{m}_B^1 by Bob. And, the distortion upper-bound is d_B^1 . Thereafter, Alice uses the marked

\mathbf{x}_2 for data embedding. Mathematically, in i -th round (if any), Alice first embeds \mathbf{m}_A^i into \mathbf{x}_{2i-2} , resulting in a new marked object \mathbf{x}_{2i-1} that will be used for data embedding \mathbf{m}_B^i by Bob. The resulting marked \mathbf{x}_{2i} will be used for the $(i+1)$ -th round (if any). It is necessary that $D(\mathbf{x}_{2i-2}, \mathbf{x}_{2i-1}) \leq d_A^i$ and $D(\mathbf{x}_{2i-1}, \mathbf{x}_{2i}) \leq d_B^i$, for any $i \geq 1$. $|\mathbf{m}_A^i|$ and $|\mathbf{m}_B^i|$ should be in a specific range unless they are the last remaining payload to be hidden.

The game will continue in turn until one fully embeds his or her own payload. Either Alice or Bob will win the game unless one cannot find the embedding strategy meeting the distortion requirement at some stage. In general, the data extraction and recovery process will be completed by cooperation. Though it is always free for us to design the distortion measure, a well-designed distortion measure will allow us to determine the winner effectively. We point that, distortion limitation essentially could be any form. We do not require that the constraint should meet some specified condition. Below, we present a tree-based approach to identify the winner when the original cover and involved parameters have been given at the beginning.

5.2 Winner Identification

5.2.1 Modeling Game on a Rooted Tree

Tree is a non-linear structure organizing data hierarchically. A rooted tree is a tree in which a special node (or say vertex) is singled out. This node is called *root* of the tree. A rooted tree is a *directed acyclic graph*. In a rooted tree with unbounded branching, each node connects an indefinite number of successor nodes by edges. A node without successor is called *leaf*.

We model our game on a rooted tree with unbounded branching. The root represents \mathbf{x}_0 , and the others are marked objects. We take Figure 3 for explanation. In Figure 3, $\mathbf{x}_{i,j}$ means the j -th node in the i -th layer. Obviously, $\mathbf{x}_{1,1}$ is the root corresponding to the raw cover. With $\mathbf{x}_{1,1}$, Alice can choose different parameters for RW as long as the parameters allow Alice to embed a part of the original payload and the “distortion” meets the requirement. In the first round, for Alice, the parameter space is $P_1(\mathbf{x}_{1,1}) = \{P_{1,1}, P_{1,2}, \dots, P_{1,s}\}$, meaning that, she can select one object from $\{\mathbf{x}_{2,1}, \mathbf{x}_{2,2}, \dots, \mathbf{x}_{2,s}\}$ as the marked object. Here, $P_{1,i}$ allows Alice to use $\mathbf{x}_{1,1}$ to embed secret bits, resulting in $\mathbf{x}_{2,i}$. Assuming that, Alice selects $\mathbf{x}_{2,i}$ as the marked object, then Bob will hide another payload into $\mathbf{x}_{2,i}$ by using a well-chosen element in $P_2(\mathbf{x}_{2,i})$, resulting in a new object $\mathbf{x}_{3,j}$. Alice continues to process $\mathbf{x}_{3,j}$ unless Bob has embedded his whole payload.

Each $\mathbf{x}_{i,j}$ is associated with a pair $(a_{i-1,j}, b_{i-1,j})$, meaning that, after reaching this node, Alice has been embedded a payload sized $a_{i-1,j}$ and Bob has embedded a payload sized $b_{i-1,j}$. It can be observed that the in-degree of each node will be exactly one. The edge whose head node is $\mathbf{x}_{i,j}$ can be associated with $P_{i-1,j}$, with which the tail node can be changed to head node. It can be easily inferred that, either $a_{i-1,j} = a_{i,k}$ or $b_{i-1,j} = b_{i,k}$ is satisfied for some k since only one player will embed secret data into the given object each time. The game will terminate on a leaf node $\mathbf{x}_{i,j}$ where either $a_{i-1,j} = l_A$ or $b_{i-1,j} = l_B$ is satisfied. Notice that, $a_{i-1,j} = l_A$ and $b_{i-1,j} = l_B$ will never hold at the same time.

5.2.2 Identifying the Winner of the Game

We describe the game task on a tree as follows. Starting from the root node, Alice and Bob alternatively select the

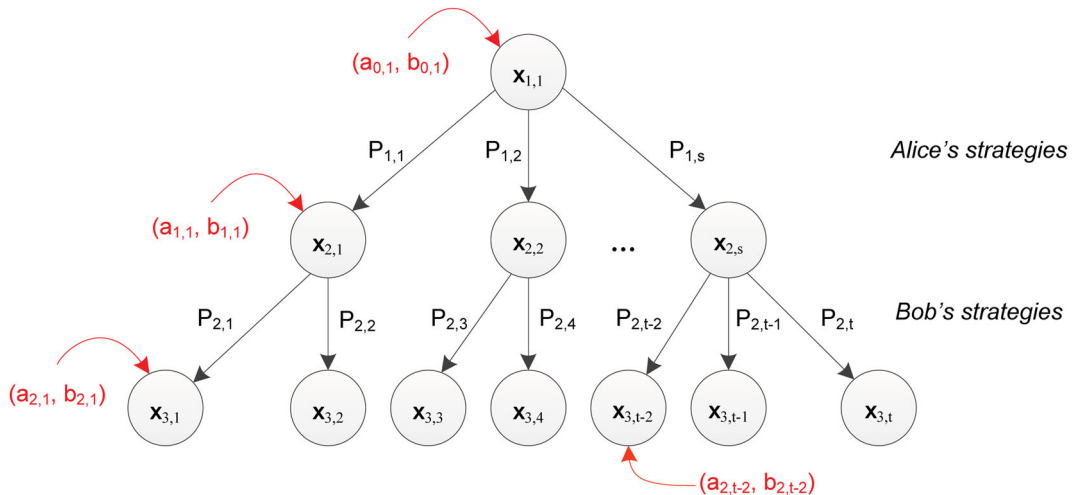


Figure 3: Example of modeling game on a rooted tree with unbounded branching.

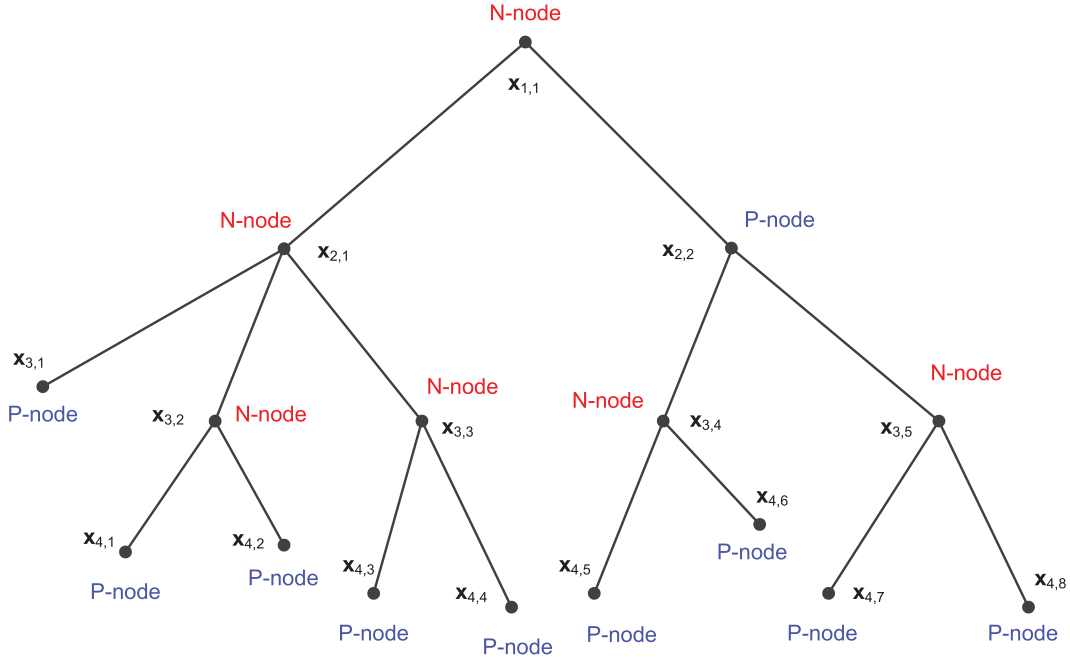


Figure 4: Example of labeling nodes on a rooted tree with unbounded branching.

successor node of the present node until a leaf node is reached. The player who arrives at a leaf is winner. We define a node as a *P-node* if it is winning for the previous player, i.e. the player who just moved. Otherwise, the node is an *N-node*, meaning that, it is winning for the next player to move. Each node on the tree will be either a *P-node* or an *N-node*. Our task is to determine the type of the root node. If the root is an *N-node*, the winner will be Alice, otherwise, Bob wins the game.

A leaf node will be always a *P-node* since the game will be terminated at the leaf node and the player who arrives at the leaf node will be the winner. For a *P-node*, every successor node should be an *N-node*. The reason is that, if there is at least one successor node that is a *P-node*, then, for the present *P-node*, the next player will surely select the *P-node* as his/her strategy, showing that the present node should not be a *P-node*. For an *N-node*, it is certainly true that there should be at least one move to a *P-node*. Therefore, we can summarize three statements:

- All leaf nodes are *P-nodes*.
- Every successor node of a *P-node* is an *N-node*.
- For every *N-node*, there is *at least one* successor node that is a *P-node*.

It is seen that, if we can effectively label each node as either *P-node* or *N-node*, we can return the label of the root as the game result. It can be addressed by the

following steps:

- S1.** Label each leaf node as a *P-node*.
- S2.** Label each node that can move to a labeled *P-node* in one step as an *N-node*.
- S3.** Label those nodes that can only move to labeled *N-nodes* in one step as *P-nodes*.
- S4.** Stop labeling if no new *P-nodes* are found in **S3**, otherwise, go to **S2**.

Figure 4 shows an example. We first label all leaf nodes $\{x_{3,1}, x_{4,1}, x_{4,2}, x_{4,3}, x_{4,4}, x_{4,5}, x_{4,6}, x_{4,7}, x_{4,8}\}$ as *P-nodes*. Then, we label $\{x_{2,1}, x_{3,2}, x_{3,3}, x_{3,4}, x_{3,5}\}$ as *N-nodes*. $x_{2,2}$ is marked as *P-node* since all successor nodes are *N-nodes*. Finally, we mark $x_{1,1}$ as *N-node*, implying that, Alice is the winner of the game. It is noted that, both the construction of the tree and the identification of the game winner can be processed by applying either depth-first search or breadth-first search, meaning that, the computational complexity is $O(n + m)$, where n denotes the number of nodes of the tree and m denotes the number of edges. However, we admit that, we actually ignore the impact of determining parameters associated to the nodes and edges, which is a future research since the system here is totally parameterized.

6. CONCLUSION AND DISCUSSION

In this paper, we present a theoretical study to the parameterized RW by game theory. Unlike games designed

for steganography and robust watermarking that aim to find the equilibrium between the encoder and the attacker, we focus on two data encoders, who, however, do not intentionally attack each other, but rather to compete with each other for maximizing their own payload, subjected to distortion constraint. We have shown that, when both players do not cooperate with each other, the optimal response for each player, surprisingly, depends on their own distortion constraint. When they cooperate with each other, the equilibrium depends on the side information shared between them. We have analyzed the two different cases for the cooperative game and provided the effective ways to find the equilibriums.

We also extend the game to a win-or-lose case. The motivation is that, the game was originally designed to LC based RW, for which it is not easy to well design the payoff functions and find the equilibrium strategies when to extend to HS embedding. Thus, we reformulate a win-or-lose game between the two encoders. The two players alternately modify the cover with well-chosen parameters. The one who first embeds his or her own payload will be the winner. As we study the game in a parametric way, which skips the payoff functions and does not rely on any particular embedding operation, it has good generalization. Since the game here has been parameterized, a further work is to study how to set the suitable system parameters. We will also investigate non-cooperative game and cooperative games for other watermarking systems [21–23].

DISCLOSURE STATEMENT

No potential conflict of interest was reported by the author(s).

FUNDING

It was supported by National Natural Science Foundation of China under grant numbers 61902235, U1636206, U1936214, and 61525203. It is also supported by “Chen Guang” project under grant number 19CG46, co-funded by the Shanghai Municipal Education Commission and Shanghai Education Development Foundation.

ORCID

Hanzhou Wu  <http://orcid.org/0000-0002-1599-7232>

REFERENCES

1. Z. Ni, Y. Shi, N. Ansari, and W. Su, “Reversible data hiding,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 16, no. 2, pp. 354–362, 2006.
2. V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Shi, “Reversible watermarking algorithm using sorting and prediction,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 19, no. 7, pp. 989–999, 2009.
3. J. Fridrich, M. Goljan, and R. Du, “Invertible authentication,” *Proc. SPIE, Security and Watermarking of Multimedia Contents III*, Vol. 3971, pp. 197–208, 2001.
4. J. Tian, “Reversible data embedding using a difference expansion,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, no. 8, pp. 890–896, 2003.
5. H. Wu, H. Wang, and Y. Shi, “PPE-based reversible data hiding,” *In: Proc. ACM Workshop Inf. Hiding Multimed. Security*, pp. 187–188, 2016. doi:10.1145/2909827.2933196
6. H. Wu, H. Wang, and Y. Shi, “Dynamic content selection-and-prediction framework applied to reversible data hiding,” *In: Proc. IEEE Workshop Inf. Forensics Security*, pp. 1–6, 2016. Available: <https://ieeexplore.ieee.org/document/7823903>
7. W. Zhang, X. Hu, X. Li, and N. Yu, “Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression,” *IEEE Trans. Image Process.*, Vol. 22, no. 7, pp. 2775–2785, 2013.
8. J. M. Ettinger, “Steganalysis and game equilibria,” *Proc. Int. Workshop Inf. Hiding*, Vol. 1525, pp. 319–328, 1998.
9. P. Moulin, and J. A. O’Sullivan, “Information-theoretic analysis of information hiding,” *IEEE Trans. Inf. Theory*, Vol. 49, no. 3, pp. 563–593, 2003.
10. A. S. Baruch, and N. Merhav, “On the error exponent and capacity games of private watermarking systems,” *IEEE Trans. Inf. Theory*, Vol. 49, no. 3, pp. 537–562, 2003.
11. A. S. Cohen, and A. Lapidot, “The Gaussian watermarking game,” *IEEE Trans. Inf. Theory*, Vol. 48, no. 6, pp. 1639–1667, 2002.
12. M. Costa, “Writing on dirty paper,” *IEEE Trans. Inf. Theory*, Vol. 29, no. 3, pp. 439–441, 1983.
13. A. S. Baruch, and N. Merhav, “On the capacity game of public watermarking systems,” *IEEE Trans. Inf. Theory*, Vol. 50, no. 3, pp. 511–524, 2004.
14. A. Ker, “Batch steganography and the threshold game,” *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX*, Vol. 6505, pp. 0401–0413, 2007.
15. A. Ker, “Batch steganography and pooled steganalysis,” *Proc. Int. Workshop Inf. Hiding*, Vol. 4437, pp. 265–281, 2006.
16. P. Schöttle, and R. Böhme, “A game-theoretic approach to content-adaptive steganography,” *In: Proc. Int. Workshop Inf. Hiding*, pp. 125–141, 2012. Available: https://link.springer.com/chapter/10.1007/978-3-642-36373-3_9

17. B. Johnson, P. Schöttle, and R. Böhme, "Where to hide the bits," *Decision and Game Theory for Security (GameSec)*, pp. 1–17, 2012. Available: https://link.springer.com/chapter/10.1007%2F978-3-642-34266-0_1
18. T. Denemark, and J. Fridrich, "Detection of content adaptive LSB matching (a game theory approach)," *Proc. SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics*, Vol. 9028, pp. 1–12, 2014.
19. P. Schöttle, A. Laszka, B. Johnson, J. Grossklags, and R. Böhme, "A game-theoretic analysis of content-adaptive steganography with independent embedding," *European Signal Process. Conference*, pp. 1–5, 2013. Available: <https://ieeexplore.ieee.org/document/6811726>
20. P. Schöttle, and R. Böhme, "Game theory and adaptive steganography," *IEEE Trans. Inf. Forensics Security*, Vol. 11, no. 4, pp. 760–773, 2016.
21. C. Qin, W. Zhang, F. Cao, X. Zhang, and C.-C. Chang, "Separable reversible data hiding in encrypted images via adaptive embedding strategy with block selection," *Signal Process.*, Vol. 153, pp. 109–122, 2018.
22. C. Qin, P. Ji, C.-C. Chang, J. Dong, and X. Sun, "Non-uniform watermark sharing based on optimal iterative BTC for image tampering recovery," *IEEE Multimed.*, Vol. 25, no. 3, pp. 36–48, 2018.
23. H. Wu, Y. Shi, H. Wang, and L. Zhou, "Separable reversible data hiding for encrypted palette images with color partitioning and flipping verification," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 27, no. 8, pp. 1620–1631, 2017.

Authors



Hanzhou Wu received his B.Sc. and Ph.D from Southwest Jiaotong University, Chengdu, China, in 2011 and 2017. From 2014 to 2016, he was a visiting scholar in New Jersey Institute of Technology, New Jersey, United States. He was a researcher in the Institute of Automation, Chinese Academy of Sciences, from 2017 to 2019. Currently, he is an Assistant Professor in Shanghai University, China. His research interests include information hiding, graph theory and game theory. He has published around 20 papers in peer journals and conferences such as IEEE TDSC, IEEE TCSVT, IEEE WIFS, ACM IH&MMSec, and IS&T Electronic Imaging, Media Watermarking, Security and Forensics.

Corresponding author. Email: h.wu.phd@ieee.org



Xinpeng Zhang received B.Sc. from Jilin University, China, in 1995, and the M.S. and Ph.D. from Shanghai University, in 2001 and 2004, respectively. Since 2004, he has been with the faculty of the School of Communication and Information Engineering, Shanghai University, where he is currently a full-time Professor. He is also with the faculty of the School of Computer Science, Fudan University. He was with The State University of New York at Binghamton as a Visiting Scholar from 2010 to 2011, and also with Konstanz University as an experienced Researcher, sponsored by the Alexander von Humboldt Foundation from 2011 to 2012. His research interests include multimedia security, image processing, and digital forensics. He has published over 200 research papers. He served an Associate Editor of IEEE Transactions on Information Forensics and Security.
