Proceedings of the 9th Hungarian-Japanese Symposium
on Discrete Mathematics and Its Applications
June 2-5, 2015, Fukuoka, Japan

Editors:

Satoru Iwata
Department of Mathematical Informatics
University of Tokyo
iwata@mist.i.u-tokyo.ac.jp

Naoyuki Kamiyama
Institute of Mathematics for Industry
Kyushu University
kamiyama@imi.kyushu-u.ac.jp

Shuji Kijima
Department of Informatics
Kyushu University
kijima@inf.kyushu-u.ac.jp

Hirotaka Ono
Department of Economics Engineering
Kyushu University
hirotaka@en.kyushu-u.ac.jp

Yukiko Yamauchi
Department of Informatics
Kyushu University
yamauchi@inf.kyushu-u.ac.jp

Hide and Seek in Digital Communication: The Steganography Game

Aron Laszka

Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
alaszka@isis.vanderbilt.edu

Dávid Szeszlér[*]

Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Budapest, Hungary
szeszler@cs.bme.hu

Abstract:

Two players, Alice and Bob play the following game. A list of positive numbers $d_1, d_2, \dots, d_n$ and an integer $1 \le k \le n$ are given. Alice chooses a k-element subset S of $\{1, 2, \dots, n\}$ and, simultaneously, Bob chooses an integer $i \in \{1, 2, \dots, n\}$. If $i \notin S$ then there is no payoff. If, on the other hand, $i \in S$ then Alice pays Bob the amount of d_i. This two-player, zero-sum game was introduced in [6] as a means of analyzing the security (or detectability) of content-adaptive steganography. A formula for computing a pair of Nash-equilibrium strategies was also given in [6], but this formula was shown in [10] to be incorrect for certain choices of the parameters. In [10], the game was also generalized to allow for costs, and this more general version was solved in the sense that finding a Nash-equilibrium was shown to be possible in polynomial time by solving an appropriate linear program. In this paper, we solve the (generalized version of the) game in a stronger sense: we provide (correct) formulas that give a pair of Nash-equilibrium strategies and thus show that these are possible to compute in strongly-polynomial time.

Keywords: Steganography, Game Theory, Nash-equilibrium

1 Introduction

Steganography is closely related to cryptography, but its task is very different: while cryptography aims at hiding the content of a message, steganography aims at hiding the very existence of the (possibly encoded) message [5, 8]. In other words, while cryptography provides the code to encrypt a message with, steganography provides the "invisible ink" to write it down with. The basis of a number of widespread steganographic methods is to use some kind of a cover media – typically a video, an image or an audio file – and modify certain bits in it. The modified media, called the *stego*, is indistinguishable to the naked eye or ear from the original one, but the altered bits can carry the hidden message. For example, steganographic algorithms that use image files as covers very often embed information in the least significant bits of pixel values.

Obviously, a successful steganographic algorithm needs to be resistant to much more sophisticated detection methods than plain human observation. The art and science of testing files for being stegos is called *steganalysis* and it relies on various elaborate statistical methods. For example, in the case of the above mentioned least significant bit steganography, steganalysis can be based on the fact that pixel values can be predicted up to a certain accuracy from neighboring pixel values.

The idea behind *content-adaptive steganography* is that the predictability of a certain bit in a cover file can depend on the position of the bit [1]. For example, in a natural image file a pixel in a smooth area

[*] Research supported by grant No. OTKA 185101 of the Hungarian Scientific Research Fund.

is much more predictable than one on the boundary of a sharp edge or corner. Consequently, focusing the embedding on less predictable parts of the cover might lead to more secure (that is, less detectable) steganography. On the other hand, the steganalyst can also exploit this and focus the search on less predictable spots.

Using game-theoretical tools for measuring steganographic capacity has become quite common, see e.g. [4, 9, 10, 6, 7, 13, 15, 16, 2]. The *(Basic) Steganography Game* was introduced in [6] in order to analyze the theoretical bounds of security or detectabiliy of content-adaptive steganography (from the steganographer's or the steganalyst's perspective, respectively) and to determine optimal embedding positions. The game is defined as follows. Assume that the steganographer (by the name of Alice) has n embedding positions to choose from. Each of these positions corresponds to a bit of the cover file, and p_i for all $1 \leq i \leq n$ denotes the probability that the value of the i^{th} bit is equal to 1. We assume that the probabilities p_i are known to both players. The game is started by Alice, who chooses k embedding positions (where $1 \leq k \leq n$ is given) and creates the stego file by flipping the corresponding bits in the cover file. Then comes Nature and chooses either the original cover file or the modified stego file by flipping an unbiased coin and sends the chosen file through the communication channel. The steganalyst (by the name of Bob) keeps the channel under surveillance, which means that he chooses an index i from 1 to n and queries the value of the bit correspoding to the i^{th} position. Based on this information, Bob must decide whether it was the cover or the stego file that Nature chose to send through the channel. The game is over at this point: Bob's payoff is 1 or (-1) depending on wheter his guess is right or wrong, respectively, and Alice's payoff is the negative of Bob's payoff in both cases.

It is easy to show that Bob's expected payoff is $2 \cdot \max\{p_i, 1-p_i\} - 1$ or 0, depending on whether the i^{th} bit (i.e., the queried one) is or is not among the k bits chosen by Alice, respectively (see Lemma 2), and the expected payoff for Alice is obviously the negative of Bob's. Therefore, the Basic Steganography Game is equivalent to the one described in the abstract (with $d_i = 2 \cdot \max\{p_i, 1-p_i\} - 1$).

Since the above game is a two-player, zero-sum game, it has a unique Nash-equilibrium value (by Neumann's classic Minimax Theorem, see Section 2.1). It is argued in [6] that this value is a good measure of security for content-adaptive steganography since it represents the maximum guaranteed expected payoff the players can achieve. Furthermore, it is sensible for the steganographer (Alice) to choose the embedding positions according to an optimum mixed strategy of the game.

The above game was generalized in [10] by incorporating the cost of steganalysis. The *Extended Steganography Game* is defined as follows. Assume that besides the values that define the Basic Steganography Game (that is, n, k and $p_1, p_2, \ldots, p_n$) a further value c_i is given for all $1 \leq i \leq n$, which represents the cost Bob has to pay for querying the i^{th} bit. It is argued in [10] that this generalization makes sense: steganalysis is obviously a costly procedure and since obtaining side information on various parts of a file to estimate the values p_i might not be equally easy, it is sensible to assume that the cost of steganalysis is nonuniform. This means that the definition of the steganography game is modified in the following way: if Bob queries the i^{th} bit then his payoff is $1 - c_i$ or $-1 - c_i$ if his guess (on whether the observed file is a cover or a stego) is right or wrong, respectively. Since this modification could result in a negative expected gain for Bob (if the cost of steganalysis is too high to make it worth), it is also sensible to assume that he has the option not to engage in the game at all. Since the cost of steganalysis obviously does not affect Alice, her payoff is not modified: it is still -1 or 1 if Bob is right or wrong, respectively. This means that the Extended Steganography Game is not a zero-sum game any more, but it has an important property: the sum of the players' payoffs depends only on the choice of one of the players (namely, Bob). We will see in Section 2.1 that this special property implies that the game can be treated as a zero-sum game in most respects.

As mentioned in the abstract, a formula for computing the Nash-equilibrium value and an optimum pair of mixed strategies of the basic version of the Steganography Game (where $c_i = 0$ for all i) was given in [6]. However, it was pointed out in [10] that for certain choices of the parameters, the formula of [6] is incorrect and a family of counterexamples to the validity of the formula was also given. (The incorrectness of the formula of [6] is unfortunately not only due to some minor technicality, their argument missed a point that is nontrivial to resolve. See the comment after Theorem 7.) An efficient method for solving the (extended) game was given in [10], but that relies on solving a linear program with $O(n)$ variables and $O(n)$ constraints. This implies that, although the solution proposed in [10] yields a polynomial algorithm, it is infeasible for a typical steganographic application, where the value of n can easily be in the ranges of many thousands. Furthermore, since the main intended purpose of the Steganography Game is to analyze the security/detectability of content-adaptive steganography, a solution in the spirit of [6] (that is, a formula) serves this purpose much better since it opens up the possibility of analyzing how the optimum reacts to changes in the input.

The main contribution of this paper is that it presents such a formula for the Extended Steganography Game (see Theorem 7) and thus it shows that the game is solvable within a running time that can be acceptable even for problem instances with a size corresponding to real-life applications (see Corollary 10). It is also shown that it is optimal for Bob to reject the game if and only if $\sum_{i=1}^{n} \frac{c_i}{d_i} \geq k$; this statement (which does not follow from the results of [10]) is also an example for the fact that a direct formula is more appropriate for the purposes of analysis. The formula presented in Theorem 7 happens to contain the (faulty) formula of [6] as a special case in the sense that it clearly shows the bounds within which the formula of [6] is correct.

This paper is organized as follows: In Section 2, we summarize all necessary preliminariy results, both on game theory in general and on the Steganography Game in particular. Then in Section 3, we present and prove the main result of the paper. Finally, Section 4 concludes the paper. The paper is intended to be self-contained, we reprove all necessary results from [10]. Furthermore, the proof presented here on the main result of [10] is much simpler than the one given in [10] (as the theory of blocking polyhedra, on which the argument of [10] was built, is completely avoided).

2 Preliminary Results

2.1 Preliminaries on Game Theory

We assume that the reader is acquainted with the most basic concepts of game theory (see [10, 14]). Nevertheless, we still summarize the most relevant notions and results very briefly below with the main intention of being able to state Lemma 1.

A *two-player (finite) game* is defined as a pair (A, B), where A and B are given $m \times n$ matrices. The first player, Alice chooses a row index $1 \leq j \leq m$, while the second player, Bob (simultaneously) chooses a column index $1 \leq i \leq n$. Then the payoff for Alice and Bob is $a_{j,i}$ and $b_{j,i}$, respectively (where $a_{j,i}$ and $b_{j,i}$ denote the corresponding entry of the respective matrix).

A *mixed strategy* of Bob is a column vector $\mathbf{x} \in \mathbb{R}^n$ that defines a probability distribution on the index set $\{1, 2, \ldots, n\}$ (meaning that the entries of $\mathbf{x}$ are non-negative and they add upp to 1). Similarly, a mixed strategy of Alice is a row vector $\mathbf{y} \in \mathbb{R}^m$ that is a probability vector. Since for a given pair of mixed strategies $(\mathbf{y}, \mathbf{x})$ the expected payoffs for Alice and Bob are $\mathbf{y}A\mathbf{x}$ and $\mathbf{y}B\mathbf{x}$, respectively, the following fundamental definition makes sense. The pair of mixed strategies $(\mathbf{y}, \mathbf{x})$ is a *Nash-equilibrium* if the following two conditions hold: *(1)* $\mathbf{y}'A\mathbf{x} \leq \mathbf{y}A\mathbf{x}$ for every $\mathbf{y}' \in \mathbb{R}^m$ and *(2)* $\mathbf{y}B\mathbf{x}' \leq \mathbf{y}B\mathbf{x}$ for every $\mathbf{x}' \in \mathbb{R}^n$. In plain words: none of the two players could achieve a better expected payoff by unilaterally switching to another mixed strategy. The classic theorem of Nash claims that there exists a Nash-equilibrium for every finite game (even multiplayer ones, which will not be discussed here).

A two-player game (A, B) is called a *zero-sum game* if $A + B = 0$. The theory of two-player, zero-sum games (as estabilished by Neumann's Minimax Theorem) is much easier than that of general two-player games. In particular, the notion of a Nash-equilibrium is simplified as follows: $(\mathbf{y}, \mathbf{x})$ is a Nash-equilibrium of $(-B, B)$ if and only if $\mathbf{x}$ maximizes the minimum entry of $B\mathbf{x}$ over all probability vectors $\mathbf{x}$, and $\mathbf{y}$ minimizes the maximum entry of $\mathbf{y}B$ over all probability vectors $\mathbf{y}$. This implies that the following statements, which are false in general, hold for the game $(-B, B)$: if $(\mathbf{y}_1, \mathbf{x}_1)$ and $(\mathbf{y}_2, \mathbf{x}_2)$ are both Nash-equilibria then *(1)* so are $(\mathbf{y}_1, \mathbf{x}_2)$ and $(\mathbf{y}_2, \mathbf{x}_1)$; *(2)* $\mathbf{y}_1B\mathbf{x}_1 = \mathbf{y}_2B\mathbf{x}_2$. In other words, the notion of a (unique) Nash-equilibrium value (with respect to a player) is a well-defined notion for zero-sum games (as opposed to the general case) and it is also meaningful to say that a given mixed strategy is optimal for a player. Furthermore, since the unique Nash-equlibrium values for Alice and Bob are obviously negatives of each other, it is also sensible to refer to the *value of the game*, by which we will mean Bob's unique Nash-equlibrium value.

Two-player, zero-sum games are also much easier to handle algorithmically: as it is shown in many textbooks on linear programming (see e.g. [12]), optimal mixed strategies for the game $(-B, B)$ can be found efficiently by solving the following linear program and its dual:

$$\max\{\mu : B\mathbf{x} \geq \mu \cdot \mathbf{1}, \mathbf{1} \cdot \mathbf{x} = 1, \mathbf{x} \geq \mathbf{0}\} \tag{1}$$

(where $\mathbf{1}$ denotes the all-1 vector).

We introduce the following notion for further reference: a two-player game (A, B) is *quasi-zero-sum* if there exists an $m_i \in \mathbb{R}$ for every $1 \leq i \leq n$ such that $a_{j,i} + b_{j,i} = m_i$ holds for every i and j; that is, the sum of the payoffs of the two players depends on i only.

Lemma 1 *Let (A, B) be a quasi-zero-sum game. Then $(\mathbf{y}, \mathbf{x})$ is a Nash-equilibrium of (A, B) if and only if $(\mathbf{y}, \mathbf{x})$ is a Nash-equilibrium of the zero-sum game $(-B, B)$. Furthermore, Bob's Nash-equilibrium payoff in the game (A, B) is unique and it is equal to the (unique) Nash-equilibrium payoff of the zero-sum game $(-B, B)$.*

PROOF: Let $A + B = M$, where every entry in the i^{th} column of M is m_i. Substituting $A = M - B$ we get that $(\mathbf{y}, \mathbf{x})$ is a Nash-equilibrium of (A, B) if and only if

$$\forall \mathbf{y}' \in \mathbb{R}^m : \ \mathbf{y}'(M - B)\mathbf{x} \leq \mathbf{y}(M - B)\mathbf{x} \quad \text{and} \tag{2a}$$

$$\forall \mathbf{x}' \in \mathbb{R}^n : \ \mathbf{y}B\mathbf{x}' \leq \mathbf{y}B\mathbf{x}. \tag{2b}$$

Expanding (2a) we get:

$$\forall \mathbf{y}' \in \mathbb{R}^m : \ \mathbf{y}'M\mathbf{x} - \mathbf{y}'B\mathbf{x} \leq \mathbf{y}M\mathbf{x} - \mathbf{y}B\mathbf{x}. \tag{3}$$

Since $\mathbf{y}'$ and $\mathbf{y}$ are probability vectors, the definition of M implies $\mathbf{y}'M = \mathbf{y}M = \mathbf{m}$, where $\mathbf{m}$ denotes an arbitrary row of M. Hence, (3) becomes:

$$\forall \mathbf{y}' \in \mathbb{R}^m : \ \mathbf{m}\mathbf{x} - \mathbf{y}'B\mathbf{x} \leq \mathbf{m}\mathbf{x} - \mathbf{y}B\mathbf{x}.$$

This implies that (2a) is equivalent to the following:

$$\forall \mathbf{y}' \in \mathbb{R}^m : \ \mathbf{y}B\mathbf{x} \leq \mathbf{y}'B\mathbf{x}. \tag{4}$$

Since (2b) and (4) together are equivalent to saying that $(\mathbf{y}, \mathbf{x})$ is a Nash-equilibrium of $(-B, B)$ by definition, this proves the first statement of the lemma. Furthermore, the second statement follows easily by observing that if $(\mathbf{y}, \mathbf{x})$ is a common Nash-equilibrium of (A, B) and $(-B, B)$ then $\mathbf{y}B\mathbf{x}$ is Bob's payoff in both games, but in case of $(-B, B)$ that is known to be unique since $(-B, B)$ is zero-sum. □

The above lemma shows that quasi-zero-sum games are essentially equivalent to zero-sum games (and thus justifies the name of this notion). However, it is also useful to point out the limits to this equivalency: a statement analogous to the second one of the above lemma would not be true for Alice: her Nash-equilibrium payoff in the quasi-zero-sum game (A, B) can depend on $\mathbf{x}$.

2.2 Preliminaries on the (Extended) Steganography Game

We summarize the necessary results of [10] and [6] on the Extended Steganography Game (defined in Section 1) below. Since none of these results are too long to show, we also give proofs for the sake of completeness.

First, observe that Bob's (the steganalyst's) final decision on whether the queried bit comes from the original cover or the stego is trivial to make. Recall that p_i denotes the probability that the value of the i^{th} bit (the one that corresponds to the i^{th} embedding position) is equal to 1. Denote by $\lfloor p_i \rceil$ the nearest integer to p_i (that is, $\lfloor p_i \rceil = 1$ or $\lfloor p_i \rceil = 0$ if $p_i \geq \frac{1}{2}$ or $p_i < \frac{1}{2}$, respectively). Then it is only sensible for

Bob to decide for "cover" if the value of the queried bit is equal to $\lfloor p_i \rceil$ and decide for "stego" otherwise (since doing anything else would decrease his expected payoff). We assume henceforth that Bob follows this rule.

Let $[n]$ denote the set $\{1, 2, \ldots, n\}$, and let $\binom{[n]}{k}$ denote the set of all k-element subsets of $[n]$.

Lemma 2 *Assume that an instance of the Extended Steganography Game (that is, n, k, and p_i and c_i for all $1 \leq i \leq n$) is given, and let $q_i = \max\{p_i, 1 - p_i\}$ for all i. Then, if Alice chooses the subset $S \in \binom{[n]}{k}$ and Bob chooses the index $i \in [n]$, then Bob's expected payoff is $2q_i - 1 - c_i$ or $-c_i$ if $i \in S$ or $i \notin S$, respectively; furthermore, Alice's expected payoff is $-(2q_i - 1)$ or 0 if $i \in S$ or $i \notin S$, respectively.*

Proof: We prove the statement for Bob's payoff, the proof for Alice's is analogous. First assume $i \notin S$. Then the queried bits of the cover and the stego are identical, so the result of the query conveys no information whatever on which of the two was chosen by Nature. Therefore no matter how Bob guesses, he has a chance of $\frac{1}{2}$ of being correct. Consequently, his expected payoff is $\frac{1}{2}(1 - c_i) + \frac{1}{2}(-1 - c_i) = -c_i$ as claimed.

Now assume $i \in S$. Then what we said about Bob's decision above implies that his declaration of "cover" or "stego" will be correct if and only if the value of the queried bit in the original cover file is equal to $\lfloor p_i \rceil$. The probability that this is true is obviously q_i. Hence Bob's expected payoff is $q_i(1 - c_i) + (1 - q_i)(-1 - c_i) = 2q_i - 1 - c_i$ as stated. □

Let $d_i = 2q_i - 1 = 2\max\{p_i, 1 - p_i\} - 1$ henceforth. The above lemma implies that the Extended Steganography Game is equivalent to the two-player game (A, B), where both A and B have n columns, their rows are indexed with the elements of $\binom{[n]}{k}$ and

$$a_{S,i} = \begin{cases} -d_i, & \text{if } i \in S, \\ 0, & \text{if } i \notin S \end{cases} \quad \text{and} \quad b_{S,i} = \begin{cases} d_i - c_i, & \text{if } i \in S, \\ -c_i, & \text{if } i \notin S \end{cases} \tag{5}$$

holds for all S and i. Observe that this two-player game is quasi-zero-sum (see Section 2.1) since $a_{S,i} + b_{S,i} = -c_i$ for all S and i. Consequently, in what follows, we will focus on solving the zero-sum game $(-B, B)$ in accordance with Lemma 1. (Recall that in the definition of the Extended Steganography Game Bob was guaranteed the chance to completely reject the game. So strictly speaking, an all-zero column would be needed to be added to B to capture this choice. However, in order to avoid overcomplicating our notations, we keep B as it is defined above and handle this issue separately.)

We mentioned in Section 2.1 that all two-player, zero-sum games are solvable via linear programming. However, the approach shown in Equation (1) would not be efficient here since B has $\binom{n}{k}$ rows. Fortunately, it is shown in [10] that $(-B, B)$ is also solvable by a linear program that has $O(n)$ variables and constraints (and thus yields a polynomial algorithm). To see this, assume that α is a mixed strategy of Alice; in other words, it is a probability distribution $(\alpha(S) : S \in \binom{[n]}{k})$. Then if Bob chooses $i \in [n]$ then Alice's expected loss in the game $(-B, B)$ is

$$\sum_{S: i \in S} \alpha(S)(d_i - c_i) - \sum_{S: i \notin S} \alpha(S) c_i = d_i \sum_{S: i \in S} \alpha(S) - c_i \sum_{\forall S} \alpha(S) = d_i \sum_{S: i \in S} \alpha(S) - c_i, \tag{6}$$

where all sums are meant over elements of $\binom{[n]}{k}$. This gives rise to the following definition.

Definition 3 *Let $(\alpha(S) : S \in \binom{[n]}{k})$ be a probability distribution on $\binom{[n]}{k}$. Then for all $i \in [n]$ we call*

$$\mathrm{tr}_\alpha(i) = \sum \left\{ \alpha(S) : i \in S, S \in \binom{[n]}{k} \right\}$$

the trace *of α on i and we call the vector $\mathbf{tr}_\alpha = (\mathrm{tr}_\alpha(1), \mathrm{tr}_\alpha(2), \ldots, \mathrm{tr}_\alpha(n))$ the* trace vector *(or simply the* trace*) of α.*

In other words, $\mathbf{tr}_\alpha$ is an element of the polytope spanned by the incidence vectors of the elements of $\binom{[n]}{k}$.

The idea of the above mentioned simplification of the linear programming formulation is to express Alice's task in terms of $\mathbf{tr}_\alpha$ instead of α. This is made possible by Equation (6), which shows that mixed strategies α with a common trace $\mathbf{tr}_\alpha$ yield the same expected loss for Alice. Since Alice's task is to minimize her maximum expected loss in the game $(-B, B)$, we get by Equation (6) that this is equivalent to the following:

$$\min\Big\{\max_{1\le i\le n}\{d_i\cdot \mathbf{a}_i - c_i\} : \mathbf{a}\in\mathbb{R}^n \text{ is the trace vector of some } \alpha\Big\}. \tag{7}$$

In order to make use of this, we need to find a description for the set (or polytope) of trace vectors. Although the main statement of the following theorem is a simple special case of Edmonds's classic result on the description of a matroid's base polytope [3] (which is applied here on uniform matroids) or it can be proved by various standard techniques (e.g. the Farkas-lemma), the proof given in [10] (and presented below) is still important since it also yields an efficient algorithm.

Theorem 4 ([10]) *A vector $\mathbf{a}\in\mathbb{R}^n$ is the trace vector of an appropriate probability distribution $(\alpha(S) : S\in\binom{[n]}{k})$ if and only if $\mathbf{0}\le\mathbf{a}\le\mathbf{1}$ and $\mathbf{1}\cdot\mathbf{a}=k$ (where $\mathbf{1}$ denotes the all-1 vector of appropriate dimension). Furthermore, if a vector $\mathbf{a}$ satisfies these conditions, then a corresponding α that assigns a positive probability to at most n subsets can be found in $O(n^2)$ time.*

Proof: Necessity is obvious: since incidence vectors of the elements of $\binom{[n]}{k}$ fulfill the conditions, so does an arbitrary convex combination of these. To show sufficiency, it will be convenient to prove the following slight generalization:

Claim 5 *Assume that $\mathbf{0}\le\mathbf{a}\le\frac{\mathbf{1}\cdot\mathbf{a}}{k}\cdot\mathbf{1}$ holds for the vector $\mathbf{a}\in\mathbb{R}^n$. Then $\mathbf{a}$ can be expressed as a positive coefficient linear combination of binary vectors containing exactly k 1's such that the number of terms in the linear combination is at most $m(\mathbf{a})=\max\Big\{1,\Big|\{i: 0<a_i<\frac{\mathbf{1}\cdot\mathbf{a}}{k}\}\Big|\Big\}$.*

To prove the claim, we proceed by induction on $m(\mathbf{a})$. If $m(\mathbf{a})=1$ then the statement is trivial. (Note that $\Big|\{i: 0<a_i<\frac{\mathbf{1}\cdot\mathbf{a}}{k}\}\Big|=1$ is impossible.) So let $m(\mathbf{a})\ge 2$.

Assume $a_1\le a_2\le\ldots\le a_n$ without loss of generality. Denote $s=\frac{\mathbf{1}\cdot\mathbf{a}}{k}$ and let $\delta=\min\{a_{n-k+1}, s-a_{n-k}\}$. It is easy to check that $\delta>0$. Now let

$$a'_i=\begin{cases} a_i, & \text{if } 1\le i\le n-k,\\ a_i-\delta, & \text{if } n-k+1\le i\le n.\end{cases}$$

Then $s'=\frac{\mathbf{1}\cdot\mathbf{a}'}{k}=s-\delta$ and $0\le a'_i\le s'$ follows directly from the definition of δ. Furthermore, $a_i=0$ implies $a'_i=0$ and $a_i=s$ implies $a'_i=s'$. Finally, if $\delta=a_{n-k+1}$ then $a'_{n-k+1}=0$ and if $\delta=s-a_{n-k}$ then $a'_{n-k}=s'$. All in all, $\mathbf{a}'=(a'_1,\ldots,a'_n)$ fulfills all conditions of the claim and $m(\mathbf{a}')\le m(\mathbf{a})-1$. Applying induction on $\mathbf{a}'$ and using $\mathbf{a}=\mathbf{a}'+\delta\cdot(0,\ldots,0,1,\ldots,1)$ (where the 1's appear in the last k positions) proves the existence of the required linear combination.

It is straightforward that the above proof also yields an $O(n^2)$ algorithm. We start by sorting the entries of the input vector $\mathbf{a}$. The inductive proof above corresponds to at most n cycles (since $m(\mathbf{a})$ decreases by at least one in each iteration). Carrying out the modifications described above (and storing the coefficient δ with the corresponding binary vector) is straightforward. At the end of each iteration we need to sort the values of the modified vector again; however, that can be performed in linear time by merging the two sorted arrays corresponding to the first $n-k$ and the last k entries of the previous vector. So the algorithm involves at most n cycles, each carried out in linear time which justifies the $O(n^2)$ overall running time. □

Since k is typically much smaller than n in a steganographic application, it is useful to mention that the above algorithm can easily be modified to achieve a running time of $O(kn \log n)$ by storing the elements of $\mathbf{a}$ in a heap. This is clearly better if $k = o\left(\frac{n}{\log n}\right)$.

The above theorem combined with Equation (7) gives that Alice's task is equivalent to the following:

$$\min \Big\{ \max_{1 \le i \le n} \{d_i \cdot \mathbf{a}_i - c_i\} : \mathbf{0} \le \mathbf{a} \le \mathbf{1}, \mathbf{1} \cdot \mathbf{a} = k, \mathbf{a} \in \mathbb{R}^n \Big\}. \tag{8}$$

This, on the other hand, can easily be turned into a linear program by introducing an extra variable μ for $\max_{1 \le i \le n}\{d_i \cdot \mathbf{a}_i - c_i\}$:

$$\begin{array}{l} \text{Minimize } \mu \\ \text{subject to} \\ \forall i \in [n]: \ d_i \cdot \mathbf{a}_i - c_i \le \mu \\ \mathbf{1a} = k \\ \mathbf{0} \le \mathbf{a} \le \mathbf{1} \end{array} \tag{9}$$

This completes the proof of the result of [10] that an optimum mixed strategy for Alice and thus the value of the game $(-B, B)$ can be computed in polynomial time: the minimum of the above linear program is the value of the game $(-B, B)$, and running the algorithm of Theorem 4 on an optimum solution $\mathbf{a}$ yields an optimum mixed strategy for Alice. It is also stated in [10] (without proof) that the dual of the above program gives an optimum mixed strategy for Bob too. Since we will need this result, we prove it below. The dual of the linear program (9) is the following:

$$\begin{array}{l} \text{Maximize } k \cdot \nu - \mathbf{1} \cdot \mathbf{z} - \mathbf{c} \cdot \mathbf{x} \\ \text{subject to} \\ \forall i \in [n]: \ d_i \cdot \mathbf{x}_i + \mathbf{z}_i \ge \nu \\ \mathbf{1x} = 1 \\ \mathbf{x} \ge \mathbf{0}, \mathbf{z} \ge \mathbf{0}, \end{array} \tag{10}$$

where $\mathbf{c} = (c_1, \ldots, c_n)$ is the vector composed of the input cost values c_i and the variables of the program are ν and the entries of the vectors $\mathbf{x}, \mathbf{z} \in \mathbb{R}^n$.

Lemma 6 *If $(\mathbf{x}, \mathbf{z}, \nu)$ is an optimum solution of the above linear program* (10)*, then $\mathbf{x}$ is an optimum mixed strategy for Bob in the game $(-B, B)$.*

Proof: It follows from the constraints of the program (10) that $\mathbf{x}$ is indeed a probability vector. Denote by M the common optimum of the programs (9) and (10) (in other words, the value of the game). The statement of the lemma is equivalent to saying that $\mathbf{x}$ guarantees Bob an expected gain of at least M for all choices $S \in \binom{[n]}{k}$ of Alice. If Alice chooses $S \in \binom{[n]}{k}$, then Bob's expected payoff is $\sum_{i \in S}(d_i - c_i)\mathbf{x}_i - \sum_{i \notin S} c_i \mathbf{x}_i$. Therefore, the lemma follows from the following calculations:

$$\begin{aligned} \sum_{i \in S}(d_i - c_i)\mathbf{x}_i - \sum_{i \notin S} c_i \mathbf{x}_i &= \sum_{i \in S} d_i \mathbf{x}_i - \sum_{\forall i} c_i \mathbf{x}_i \\ &\ge \sum_{i \in S}(\nu - \mathbf{z}_i) - \sum_{\forall i} c_i \mathbf{x}_i \\ &= k\nu - \sum_{i \in S} \mathbf{z}_i - \sum_{\forall i} c_i \mathbf{x}_i \\ &\ge k\nu - \sum_{\forall i} \mathbf{z}_i - \sum_{\forall i} c_i \mathbf{x}_i \\ &= M, \end{aligned}$$

where the final equation follows from the fact that $(\mathbf{x}, \mathbf{z}, \nu)$ is optimal and all other steps follow from the constraints of the program (10). □

3 Main Result

Assume that n, k, and $d_i \geq 0$, $c_i \geq 0$ for all $1 \leq i \leq n$ are given, and let the two-player, zero-sum game $(-B, B)$ be defined by the payoff matrix B given in Equation (5).

Let $\Delta_i = d_i - c_i$ for all i. Obviously, we can assume $\Delta_i > 0$ for all i, since $\Delta_i \leq 0$ would imply that choosing i guarantees Bob a nonpositive payoff, so Bob would completely avoid that (as he also has the option not to engage in the game at all), and the i^{th} column of B could be deleted. We will also assume $\Delta_1 \leq \Delta_2 \leq \ldots \leq \Delta_n$ without loss of generality.

Theorem 7 *Let $R = \sum_{i=1}^{n} \frac{c_i}{d_i}$. It is optimal for Bob to reject the game $(-B, B)$ if and only if $R \geq k$. If $R < k$, then introduce the following notations for all $1 \leq j \leq n$:*

$$s_j = \sum_{i=j}^{n} \frac{1}{d_i}, \quad t_j = \frac{1}{s_j} \cdot \left(k - j + 1 - \sum_{i=j}^{n} \frac{c_i}{d_i} \right)$$

and let $m = \min\{i : \Delta_i \geq t_i\}$. Then (assuming $R < k$) the following are true:

1. *the value of the game is t_m;*
2. *the following formula gives an optimum mixed strategy $\mathbf{x}$ for Bob:*

$$x_i = \begin{cases} 0, & \text{if } 1 \leq i \leq m-1 \\ \frac{1}{s_m} \cdot \frac{1}{d_i}, & \text{if } m \leq i \leq n \end{cases}$$

3. *the following formula gives the trace $\mathbf{a}$ of an optimum mixed strategy for Alice:*

$$a_i = \begin{cases} 1, & \text{if } 1 \leq i \leq m-1 \\ \frac{t_m + c_i}{d_i}, & \text{if } m \leq i \leq n \end{cases}$$

Proof: We first show that it is optimal for Bob to reject the game if $R \geq k$. Indeed, if $R \geq k$ then it is easy to check that the vales $\mu = \left(\frac{k}{R} - 1\right) \cdot \min_{1 \leq i \leq n} c_i$ and $a_i = \frac{k}{R} \cdot \frac{c_i}{d_i}$ for all i fulfill all constraints of the primal linear program (9) and give an objective function value of $\mu \leq 0$. Hence the value of the game, the minimum value of the program (9) is also nonpositive, which justifies that it is optimal for Bob to reject the game. The converse of this statement will follow from the fact that the value of the game is positive for Bob if $R < k$, which we will show below.

So assume $R < k$ henceforth. We start with the following two technical claims.

Claim 8 $\Delta_k \geq t_k$

Proof: Using the definitions of Δ_k, t_k, and s_k and after multiplying by s_k, the claim is equivalent to the following:

$$(d_k - c_k) \sum_{i=k}^{n} \frac{1}{d_i} \geq 1 - \sum_{i=k}^{n} \frac{c_i}{d_i}.$$

Rewriting this:

$$1 - \frac{c_k}{d_k} + \Delta_k \cdot \sum_{i=k+1}^{n} \frac{1}{d_i} \geq 1 - \frac{c_k}{d_k} - \sum_{i=k+1}^{n} \frac{c_i}{d_i}.$$

Subtracting $1 - \frac{c_k}{d_k}$ from both sides and using the fact that all variables (including Δ_k) denote nonnegative numbers the claim is proved. □

Claim 9 *For all* $1 \leq j \leq n-1,\ \ t_j \geq t_{j+1}$ *if and only if* $\Delta_j \geq t_j$.

PROOF: Let $u_j = k - j + 1 - \sum_{i=j}^{n} \frac{c_i}{d_i}$ for all j. Then

$$t_{j+1} = \frac{u_{j+1}}{s_{j+1}} = \frac{k - j - \sum_{i=j+1}^{n} \frac{c_i}{d_i}}{\sum_{i=j+1}^{n} \frac{1}{d_i}} = \frac{u_j + \frac{c_j}{d_j} - 1}{s_j - \frac{1}{d_j}} = \frac{u_j d_j + c_j - d_j}{s_j d_j - 1} = \frac{\Delta_j - u_j d_j}{1 - s_j d_j}.$$

Therefore, since $1 - s_j d_j < 0$ is obvious and $t_j = \frac{u_j}{s_j}$, multiplying the inequality $t_j \geq t_{j+1}$ by $1 - s_j d_j$ (and using the above equation) gives that $t_j \geq t_{j+1}$ is equivalent to $t_j - u_j d_j \leq \Delta_j - u_j d_j$, which proves the claim. □

Claim 8 implies that $m \leq k$, while Claim 9 (together with $\Delta_1 \leq \Delta_2 \leq \ldots \leq \Delta_n$) implies that the relations between the values t_i and Δ_i shown below are true:

$$\begin{array}{ccccccccccccccccc}
0 & < & \Delta_1 & \leq & \Delta_2 & \leq & \ldots & \leq & \Delta_{m-1} & \leq & \Delta_m & \leq & \Delta_{m+1} & \leq & \ldots & \leq & \Delta_n \\
 & & \wedge & & \wedge & & & & \wedge & & \vee\!| & & \vee\!| & & & & \vee\!| \\
0 & < & t_1 & < & t_2 & < & \ldots & < & t_{m-1} & < & t_m & \geq & t_{m+1} & \geq & \ldots & \geq & t_n
\end{array}$$

Observe that $t_1 > 0$ follows from the assumption $R = \sum_{i=1}^{n} \frac{c_i}{d_i} < k$ even in the $m = 1$ case.

All (remaining) statements of the theorem will be shown by giving appropriate values to the variables μ, ν, and $\mathbf{z} = (z_1, \ldots, z_n)$ such that these, together with the values $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{x} = (x_1, \ldots, x_n)$ given in the statement of the theorem, describe a pair of solutions for the linear program (9) and its dual (10), which yield the common objective function value of t_m. This will indeed show that the common optima of the two programs (and thus the value of the game) is t_m and that both solutions are optimum.

So let $\mu = t_m$, $\nu = \frac{1}{s_m}$, and

$$z_i = \begin{cases} \frac{1}{s_m}, & \text{if } 1 \leq i \leq m-1 \\ 0, & \text{if } m \leq i \leq n. \end{cases}$$

We need to check that the given solutions indeed fulfill all constraints of the programs (9) and (10) and that they yield the common objective function value of $t_m > 0$.

In case of the primal program (9), the constraints $d_i \cdot \mathbf{a}_i - c_i \leq \mu$ are trivial if $i \geq m$ (since they give $t_m \leq t_m$), while in the $i \leq m-1$ case they are equivalent to $\Delta_i = d_i - c_i \leq t_m$, which is clearly true by the above chart on the relations between Δ_i and t_i. The constraint $\mathbf{1a} = k$ follows by

$$\sum_{i=1}^{m-1} 1 + \sum_{i=m}^{n} \frac{t_m + c_i}{d_i} = (m-1) + t_m \cdot s_m + \sum_{i=m}^{n} \frac{c_i}{d_i} =$$
$$(m-1) + \left(k - m + 1 - \sum_{i=m}^{n} \frac{c_i}{d_i} \right) + \sum_{i=m}^{n} \frac{c_i}{d_i} = k.$$

Furthermore, $\mathbf{a} \geq \mathbf{0}$ is obvious (recall that $t_m > 0$ is true even if $m = 1$, as mentioned above). Finally, $\mathbf{a} \leq \mathbf{1}$ is trivial if $i \leq m-1$ and it is equivalent to $t_m \leq d_i - c_i = \Delta_i$ if $i \geq m$, which again follows from the above chart on Δ_i and t_i.

In case of the dual program (10), the constraints $d_i \cdot x_i + z_i \geq \nu$ are trivial both in the $i \leq m-1$ and in the $i \geq m$ case (and they are fulfilled with equation for all i). The constraint $\mathbf{1x} = 1$ follows by

$$\sum_{i=1}^{m-1} 0 + \sum_{i=m}^{n} \frac{1}{s_m} \cdot \frac{1}{d_i} = \frac{1}{s_m} \cdot s_m = 1.$$

Furthermore, $\mathbf{x} \geq \mathbf{0}$ and $\mathbf{z} \geq \mathbf{0}$ are straightforward.

Finally, the objective function value of the primal program (9) is $\mu = t_m$ by definition, while in case of the dual (10) it is also equal to t_m by

$$k \cdot \nu - \mathbf{1} \cdot \mathbf{z} - \mathbf{c} \cdot \mathbf{x} = k \cdot \frac{1}{s_m} - \sum_{i=1}^{m-1} \frac{1}{s_m} - \sum_{i=m}^{n} c_i \cdot \frac{1}{s_m} \cdot \frac{1}{d_i} =$$
$$\frac{1}{s_m} \cdot \left(k - (m-1) - \sum_{i=m}^{n} \frac{c_i}{d_i} \right) = t_m,$$

which completes the proof. □

It might be interesting to mention that the faulty formulas given in [6] for the $c_i \equiv 0$ case (that is, the Basic Steganography Game) coincide with the ones given by the above theorem if $m = 1$. However, these formulas are obviously wrong if $m > 1$ (even if $c_i \equiv 0$ is assumed) since they define a_1, the first entry of the trace vector $\mathbf{a}$, to be bigger than 1.

Corollary 10 *The value of the game $(-B, B)$ and an optimum mixed strategy for Bob can be obtained by performing $O(n \log n)$ elementary operations on the input data, while an optimum mixed strategy for Alice can be obtained by performing $O(\min\{n^2, kn \log n\})$ elementary operations.*

Proof: To apply Theorem 7, we need to start by sorting the values $\Delta_1, \Delta_2, \ldots, \Delta_n$ (in accordance with the assumption made before claiming the theorem). This obviously takes $O(n \log n)$ comparisons. After that, using the formulas of Theorem 7 to compute the values t_m, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ is clearly possible by performing $O(n)$ elementary operations on the input data. Finally, applying the algorithm described in the proof of Theorem 4 on $\mathbf{a}$ to obtain an optimum mixed strategy for Alice takes $O(\min\{n^2, kn \log n\})$ elementary operations (in accordance with the observation made after Theorem 4). □

4 Conclusion

In this paper, we solved the Basic and Extended Steganography Games. These game-theoretic models were introduced in [6] and [10] for studying content-adaptive steganography; however, their previous solutions were either incorrect or based on linear-programming tools. In this paper, we first provided improved proofs for the previous results (e.g., Theorem 4) of [10]; then, as our main contribution, we provided formulas describing Nash-equilibria of both the basic and the extended games.

Compared to the linear-programming solution of [10], the formulas presented in this paper have multiple advantages. First, they allow us to solve the game for much larger instances, which opens up the possibility of applying the Steganography Game in practice, where the value of n can easily be in the ranges of thousands. Second, they allow us to express the condition for the steganalyst to reject the game in the simple form of $\sum_{i=1}^{n} \frac{c_i}{d_i} \geq k$ and to express the steganographic capacity of a channel in a simple, almost closed form. These expressions let us gain insight into the security/detectability of content-adaptive steganography, which previous results could not provide.

Our results can be extended in multiple directions. From a practical point of view, a natural extension would be allowing the steganalyst to query multiple bits. In [7], such an extension of the Basic Steganography Game (but not the Extended one) was considered, but without characterizing the equilibria of the game.

References

[1] R. Böhme, *Advanced Statistical Steganalysis*, Advanced Statistical Steganalysis, Springer-Verlag, Berlin, Germany, (2010).

[2] T. Denemark, J. Fridrich, Detection of content adaptive LSB matching: A game theory approach, *Proc. Watermarking, Security, and Forensics* (2014) **Vol. 9028**, p. 902804. SPIE and IS&T.

[3] J. Edmonds, Matroids and the greedy algorithm, *Mathematical Programming*, (1971) **Vol 1**, 127-136.

[4] J. M. Ettinger, Steganalysis and Game Equilibria, *Information Hiding, Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg (1998) **Vol 1525**, 319-328.

[5] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, Cambridge University Press, New York, NY, (2009).

[6] B. Johnson, P. Schöttle, R. Böhme, Where to Hide the Bits, *Decision and Game Theory for Security, Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, (2012) **Vol 7638**, 1-17.

[7] B. Johnson, P. Schöttle, A. Laszka, J. Grossklags, R. Böhme, Bitspotting: Detecting optimal adaptive steganography, *Proc. 12th International Workshop on Digital-Forensics and Watermarking* (2013), 3-18.

[8] S. Katzenbeisser, F. A. Petitcolas, Eds., *Information Hiding Techniques for Steganography and Digital Watermarking* Artech House, (2000).

[9] A. D. Ker, Batch steganography and the threshold game, *Proc. Security, Steganography and Watermarking of Multimedia Contents IX* (2007) **Vol. 6505**, 401-413.

[10] A. Laszka, Á. M. Földes, Modeling Content-Adaptive Steganography with Detection Costs as a Quasi-Zero-Sum Game *Infocommunications Journal* (2013) **Vol 5 (4)**, 33-43.

[11] K. Leyton-Brown, Y. Shoham,, *Essentials of Game Theory: A Concise Multidisciplinary Introduction* Morgan and Claypool Publishers, Synthesis Lectures on Artificial Intelligence and Machine Learning, (2008) **Vol 2**, 1-88.

[12] J. Matoušek, B. Gärtner, *Understanding and Using Linear Programming* Springer, Berlin, Heidelberg, (2007).

[13] A. Orsdemir, H. O. Altun, G. Sharma, M. F. Bocko, Steganalysis aware steganography: Statistical indistinguishability despite high distortion, *ography: Statistical indistinguishability despite high distortion, Proc. Security, Forensics, Steganography, and Watermarking of Multimedia Contents X* (2008) **Vol 6819**.

[14] M. Osborne, *An Introduction To Game Theory*, Oxford University Press, USA (2003).

[15] P. Schöttle, R. Böhme, A Game-Theoretic Approach to Content-Adaptive Steganography, *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg (2013), **Vol 7692**, 125-141.

[16] P. Schöttle, A. Laszka, B. Johnson, J. Grossklags, R. Böhme, A game-theoretic analysis of content-adaptive steganography with independent embedding *Proc. of the 21st European Signal Processing Conference (EUSIPCO)*, Marrakech, Morocco, (2013).