

# Various Views on the Trapdoor Channel and an Upper Bound on its Capacity

Tobias Lutz

## Abstract

Two novel views are presented on the trapdoor channel. First, by deriving the underlying iterated function system (IFS), it is shown that the trapdoor channel with input blocks of length  $n$  can be regarded as the  $n$ th element of a sequence of shapes approximating a fractal. Second, an algorithm is presented that fully characterizes the trapdoor channel and resembles the recursion of generating all permutations of a given string. Subsequently, the problem of maximizing a  $n$ -letter mutual information is considered. It is shown that  $\frac{1}{2} \log_2 \left(\frac{5}{2}\right) \approx 0.6610$  bits per use is an upper bound on the capacity of the trapdoor channel. This upper bound, which is the tightest upper bound known, proves that feedback increases the capacity.

## Index Terms

Trapdoor channel, Lagrange multipliers, convex optimization, iterated function systems, fractals, channels with memory, recursions, permutations.

## I. INTRODUCTION

The trapdoor channel was introduced by David Blackwell in 1961 [1] and is used by Robert Ash both as a book cover and as an introductory example for channels with memory [2]. The mapping of channel inputs to channel outputs can be described as follows. Consider a box that contains a ball that is labeled  $s_0 \in \{0, 1\}$ , where the index 0 refers to time 0. Both the sender and the receiver know the initial ball. In time slot 1, the sender places a new ball labeled  $x_1 \in \{0, 1\}$  in the box. In the same time slot, the receiver chooses one of the two balls  $s_0$  or  $x_1$  at random while the other ball remains in the box. The chosen ball is interpreted as channel output  $y_1$  at time  $t = 1$  while the remaining ball becomes the channel state  $s_1$ . The same procedure is applied in every future channel use. In time slot 2, for instance, the sender places a new ball  $x_2 \in \{0, 1\}$  in the box and the corresponding channel output  $y_2$  is either  $x_2$  or  $s_1$ . The transmission process is visualized in Fig. 4. Fig. 4(a) shows the trapdoor channel at time  $t$  when the sender places ball  $x_t$  in the box. In the same time slot, the receiver chooses randomly ball  $s_{t-1}$  as channel output. Consequently, the upcoming channel state  $s_t$  becomes  $x_t$  (see Fig. 4(b)). At time  $t + 1$  the sender places a new ball  $x_{t+1}$  in the box and the receiver draws  $y_{t+1}$  from  $s_t$  and  $x_{t+1}$ . Table I depicts the probability of an output  $y_t$  given an input  $x_t$  and state  $s_{t-1}$ .

Tobias Lutz is with the Lehrstuhl für Nachrichtentechnik, Technische Universität München, D-80290 München, Germany (e-mail: tobi.lutz@tum.de).

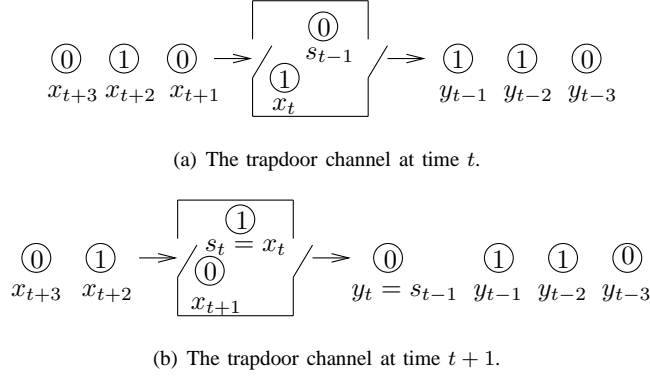


Fig. 1. At time  $t$  the sender places a new ball  $x_t$  in the box. The corresponding channel output  $y_t$  is  $s_{t-1}$  and the next state  $s_t$  becomes  $x_t$ .

Despite the simplicity of the trapdoor channel, the derivation of its capacity seems challenging and is still an open problem. One feature that makes the problem cumbersome is that the distribution of the output symbols may depend on events happening arbitrarily far back in the past since each ball has a positive probability to remain in the channel over any finite number of channel uses. Instead of maximizing  $I(X; Y)$  one rather has to consider the multi-letter mutual information, i.e.,  $\limsup_{n \rightarrow \infty} I(X^n; Y^n)$ .

TABLE I  
TRANSITION PROBABILITIES OF THE TRAPDOOR CHANNEL

$x_t$	$s_{t-1}$	$p(y_t = 0   x_t, s_{t-1})$	$p(y_t = 1   x_t, s_{t-1})$
0	0	1	0
0	1	0.5	0.5
1	0	0.5	0.5
1	1	0	1

Let  $P_{n|s_0}$  denote the matrix of conditional probabilities of output sequences of length  $n$  given input sequences of length  $n$  where the initial state equals  $s_0$ . The following ordering of the entries of  $P_{n|s_0}$  is assumed. Row indices represent input sequences and column indices represent output sequences. To be more precise, the entry  $[P_{n|s_0}]_{i,j}$  is the conditional probability of the binary output sequence corresponding to the integer  $j - 1$  given the binary input sequence corresponding to the integer  $i - 1$ ,  $1 \leq i, j \leq 2^n$ . For instance, if  $n = 3$  then  $[P_{3|s_0}]_{5,3}$  denotes the conditional probability that the channel input  $x_1 x_2 x_3 = 100$  will be mapped to the channel output  $y_1 y_2 y_3 = 010$ . It was shown in [3] that the conditional probability matrices  $P_{n|s_0}$  satisfy the recursion laws

$$P_{n+1|0} = \begin{bmatrix} P_{n|0} & 0 \\ \frac{1}{2}P_{n|1} & \frac{1}{2}P_{n|0} \end{bmatrix} \quad (1)$$

$$P_{n+1|1} = \begin{bmatrix} \frac{1}{2}P_{n|1} & \frac{1}{2}P_{n|0} \\ 0 & P_{n|1} \end{bmatrix}, \quad (2)$$

where the initial matrices are given by  $P_{0|0} = P_{0|1} = [1]$ . A quick inspection of  $P_{2|0}$  and  $P_{2|1}$  reveals that the inputs 00 and 11 are mapped to disjoint outputs. Hence, a rate of 0.5 bits per use (b/u) is achievable from the sender to the receiver. It was shown in [4] that 0.5 b/u is indeed the zero-error capacity of the trapdoor channel.

Permuter et al. [5] considered the trapdoor channel under the additional assumption of having a unit delay feedback link available from the receiver to the sender. The sender is able to determine the state of the channel in each time slot. They established that the capacity of the trapdoor channel with feedback is equal to the logarithm of the golden ratio. One can already deduce from this quantity that the achievability scheme involves a constrained coding scheme in which certain sub-blocks are forbidden.

In this paper, we propose two different views on the trapdoor channel. Based on the underlying stochastic matrices (1) and (2), the trapdoor channel can be described geometrically as a fractal or algorithmically as a recursive procedure. We then consider the problem of maximizing the  $n$ -letter mutual information of the trapdoor channel for any  $n \in \mathbb{N}$ . We relax the problem by permitting distributions that are not probability distributions. The resulting optimization problem is convex but the feasible set is larger than the probability simplex. Using the method of Lagrange multipliers via a theorem presented in [2], we show that  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right) \approx 0.6610$  b/u is an upper bound on the capacity of the trapdoor channel. Specifically, the same absolute maximum  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right) \approx 0.6610$  b/u results for all trapdoor channels which process input blocks of even length  $n$ . And the sequence of absolute maxima corresponding to trapdoor channels which process inputs of odd lengths converges to  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right)$  b/u from below as the block length increases. Unfortunately, the absolute maxima of our relaxed optimization are attained outside the probability simplex, otherwise we would have established the capacity. Nevertheless,  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right) \approx 0.6610$  b/u is, to the best of our knowledge, the tightest capacity upper. Moreover, this bound is less than the feedback capacity of the trapdoor channel.

The organization of this paper is as follows. Section II interprets the trapdoor channel as a fractal and derives the underlying iterated function system (IFS). Section III introduces a recursive algorithm which fully characterizes the trapdoor channel. Comments on the permuting nature of the trapdoor channel are provided. Section IV presents a solution to the optimization problem outlined above and derives various recursions. The paper concludes with Section V.

#### A. Notation

The symbols  $\mathbb{N}_0$  and  $\mathbb{N}$  refer to the natural numbers with and without 0, respectively. The canonical basis vectors of  $\mathbb{R}^3$  are denoted by  $e_x$ ,  $e_y$  and  $e_z$ . They are assumed to be row vectors. The  $n$ -fold composition of a function, say  $\Phi$ , is denoted as  $\Phi^{\circ n}$ . The input corresponding to the  $i$ th row of  $P_{n|s_0}$  is denoted as  $x_i^n$ . The input corresponding to the  $i$ th row of  $P_{n|s_1}$  is denoted as  $x_i^n$ . Further,  $I_n$  denotes the  $2^n \times 2^n$  identity matrix,  $\tilde{I}_n$  is a  $2^n \times 2^n$  matrix whose secondary diagonal entries are all equal to 1 while the remaining entries are all equal to 0, and  $\mathbf{1}_n$  denotes a column vector of length  $2^n$  consisting only of ones. The vector  $\mathbf{1}_n^T$  is the transpose of  $\mathbf{1}_n$ . For the sake of readability we use  $\exp_2(\cdot)$  instead of  $2^{(\cdot)}$ . If the logarithm  $\log_2(\cdot)$  or the exponential function  $\exp_2(\cdot)$  is applied to a vector or a matrix, we mean that  $\log_2(\cdot)$  or  $\exp_2(\cdot)$  of each element of the vector or matrix is taken. Finally, the symbol

◦ refers to the Hadarmard product, i.e., the entrywise product of two matrices.

## II. THE TRAPDOOR CHANNEL AND FRACTAL GEOMETRY

### A. Prerequisites

We briefly introduce the idea of *iterated function systems* and *fractals*. For a comprehensive introduction to the subject, see for instance [6]. In a nutshell, a fractal is a geometric pattern which exhibits self-similarity at every scale. A systematic way for generating a fractal starts with a complete metric space  $(M, d)$ . The space to which the fractal belongs is, however, not  $M$  but the space of non-empty compact subsets of  $M$ , denoted as  $\mathcal{H}(M)$ . A suitable choice for a metric for  $\mathcal{H}(M)$  is the Hausdorff distance  $h_d(A, B) := \max\{d(A, B), d(B, A)\}$  where  $d(A, B) := \max_{x \in A} \min_{y \in B} d(x, y)$ ,  $A, B \in \mathcal{H}(M)$  and analogously for  $d(B, A)$ . It is then guaranteed that  $(\mathcal{H}(M), h_d)$  is a complete metric space and that every contraction mapping<sup>1</sup>  $\varphi : M \rightarrow M$  on  $(M, d)$  becomes a contraction mapping  $\varphi : \mathcal{H}(M) \rightarrow \mathcal{H}(M)$  on  $(\mathcal{H}(M), h_d)$  defined by  $\varphi(A) = \{\varphi(x) : x \in A\}$  for all  $A \in \mathcal{H}(M)$ .

The following definition and theorem provides a method for generating fractals.

**Definition II.1.** [6, Chapter 3.7] A hyperbolic iterated function system (IFS) consists of a complete metric space  $(M, d)$  together with a finite set of contraction mappings  $\varphi_n : M \rightarrow M$ , with respective contractivity factors  $s_n$  for  $n = 1, 2, \dots, N$ . The notation for the IFS is  $\{M; \varphi_n, n = 1, 2, \dots, N\}$  and its contractivity factor is  $s = \max\{s_n : n = 1, 2, \dots, N\}$ .

The fixed point of a hyperbolic IFS, also called the *attractor* or *self-similar set* of the IFS, is a (deterministic) fractal and results from iterating the IFS with respect to any  $A \in \mathcal{H}(M)$ . This is the content of the following theorem.

**Theorem II.2.** [6, Chapter 3.7] Let  $\{M; \varphi_n, n = 1, 2, \dots, N\}$  be an iterated function system with contractivity factor  $s$ . Then the transformation  $\Phi : \mathcal{H}(M) \rightarrow \mathcal{H}(M)$  defined by

$$\Phi(A) = \bigcup_{n=1}^N \varphi_n(A) \quad (3)$$

for all  $A \in \mathcal{H}(M)$ , is a contraction mapping on the complete metric space  $(\mathcal{H}(M), h_d)$  with contractivity factor  $s$ . Its unique fixed point,  $A^* \in \mathcal{H}(M)$ , obeys

$$A^* = \Phi(A^*) = \bigcup_{n=1}^N \varphi_n(A^*),$$

and is given by  $A^* = \lim_{k \rightarrow \infty} \Phi^{\circ k}(A)$  for any  $A \in \mathcal{H}(M)$ .

Many well-known fractals, e.g., the *Koch snowflake*, the *Cantor set*, the *Mandelbrot set*, etc., can be generated using Definition II.1 and Theorem II.2. Indeed, a segment of the Mandelbrot set is shown on the cover of the book

<sup>1</sup>Let  $(M, d)$  be a metric space. Recall that a mapping  $\varphi : M \rightarrow M$  is a *contraction* if there exists a  $0 < s < 1$  such that  $d(\varphi(x), \varphi(y)) \leq s \cdot d(x, y)$  for all  $x, y \in M$ .

by Cover and Thomas [7]. Another famous representative, the *Sierpinski triangle*, is introduced in the following example. We will later see that this fractal is related to the trapdoor channel.

**Example II.3.** (Sierpinski triangle) *Consider the IFS*

$$\left\{ [0, 1]^2; \varphi_1(x, y) = \left( \frac{x+1}{2}, \frac{y}{2} \right), \varphi_2(x, y) = \left( \frac{x}{2}, \frac{y+1}{2} \right), \varphi_3(x, y) = \left( \frac{x}{2}, \frac{y}{2} \right) \right\}. \quad (4)$$

The affine transformations  $\varphi_n$ ,  $n = 1, 2, 3$ , scale any  $A \in \mathcal{H}([0, 1]^2)$  by a factor of 0.5. Additionally,  $\varphi_1$  and  $\varphi_2$  introduce translations by 0.5 into the  $x$ - and  $y$ -direction, respectively. The Sierpinski triangle is approximated arbitrarily close by iterating  $\Phi(A)$  for any  $A \in \mathcal{H}([0, 1]^2)$ . Fig. 2 shows the result after performing five iterations of (4). The initial shape  $A$  in Fig. 2(a) is a triangle with corner points  $(0, 0)$ ,  $(1, 0)$ ,  $(0, 1)$  and in Fig. 2(b) a triangle with corner points  $(0, 0)$ ,  $(1, 1)$ ,  $(1, 0)$ . As one performs more iterations, both sets converge to the same set  $A^*$ .

### B. The Trapdoor Channel as a Fractal

In this section, we derive a hyperbolic IFS for the trapdoor channel. Instead of working with  $P_{n|s_0}$  we take a geometric approach, i.e.,  $P_{n|s_0}$  will be mapped to the unit cube  $[0, 1]^3 \subset \mathbb{R}^3$ .

**Definition II.4.** Let  $\mathcal{M}$  denote the set  $\{P_{n|s_0} : n \in \mathbb{N}_0, s_0 = 0, 1\}$  of trapdoor channel matrices. The function  $\rho^{(n)} : \mathcal{M} \rightarrow [0, 1]^3$  represents each  $P_{n|s_0}$  as a shape in  $[0, 1]^3$  according to

$$P_{n|s_0} \mapsto \left( x, y, [P_{n|s_0}]_{i,j} \right), \quad \text{for all } 1 \leq i, j \leq 2^n \quad (5)$$

where  $(i-1) \cdot 2^{-n} < x < i \cdot 2^{-n}$  and  $1-j \cdot 2^{-n} < y < 1 - (j-1) \cdot 2^{-n}$ .

Each entry  $[P_{n|s_0}]_{i,j}$  of  $P_{n|s_0}$  is identified with a square of side length  $2^{-n}$ , which has a distance of  $[P_{n|s_0}]_{i,j}$  to the  $xy$ -plane. The alignment of the square corresponding to  $[P_{n|s_0}]_{i,j}$  with respect to the other squares in  $\rho^{(n)}(P_{n|s_0})$  is in accordance to the alignment of  $[P_{n|s_0}]_{i,j}$  with respect to the other entries of  $P_{n|s_0}$ . Fig. 3 depicts the representations  $\rho^{(1)}(P_{1|0})$  and  $\rho^{(1)}(P_{1|1})$  of

$$P_{1|0} = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad P_{1|1} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{bmatrix}.$$

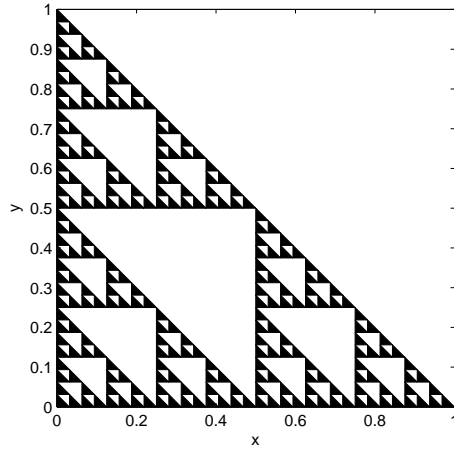
The following proposition expresses  $\rho^{(n+1)}(P_{n+1|0})$  and  $\rho^{(n+1)}(P_{n+1|1})$  recursively in terms of  $\rho^{(n)}(P_{n|0})$  and  $\rho^{(n)}(P_{n|1})$ .

**Lemma II.5.** The representations  $\rho^{(n+1)}(P_{n+1|0})$  and  $\rho^{(n+1)}(P_{n+1|1})$  of  $P_{n+1|0}$  and  $P_{n+1|1}$  satisfy the recursion laws

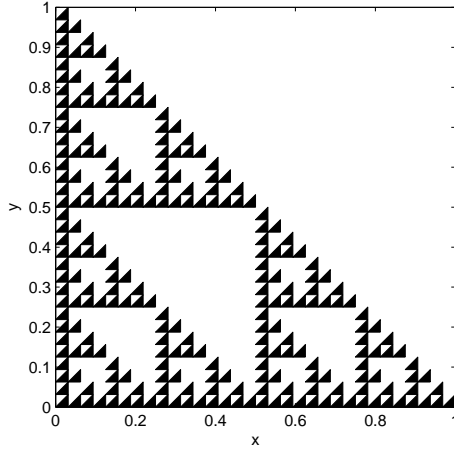
$$\rho^{(n+1)}(P_{n+1|0}) = \frac{1}{2} \cdot \left\{ \rho^{(n)}(P_{n|0}) + e_x, \rho^{(n)}(2 \cdot P_{n|0}) + e_y, \rho^{(n)}(P_{n|1}) \right\} \quad (6)$$

$$\rho^{(n+1)}(P_{n+1|1}) = \frac{1}{2} \cdot \left\{ \rho^{(n)}(2 \cdot P_{n|1}) + e_x, \rho^{(n)}(P_{n|1}) + e_y, \rho^{(n)}(P_{n|0}) + e_x + e_y \right\}, \quad (7)$$

for all  $n \in \mathbb{N}_0$ .



(a) The initial shape is a triangle with corner points  $(0, 0)$ ,  $(1, 0)$ ,  $(0, 1)$ .



(b) The initial shape is a triangle with corner points  $(0, 0)$ ,  $(1, 1)$ ,  $(1, 0)$ .

Fig. 2. Sierpinski triangle after four iterations of the underlying IFS with two different initial shapes.

*Proof:* Recursions (6) and (7) are a consequence of the structure of block matrices (1) and (2), respectively. We just outline the derivation of (6). The first term on the right hand side of (6) represents the lower right corner of (1), i.e., those entries of  $P_{n+1|0}$  with row and column indices  $2^n < i, j, \leq 2^{n+1}$ . Observe that each entry  $[P_{n+1|0}]_{i,j}$  is equal to  $\frac{1}{2} [P_n|0]_{i-2^n, j-2^n}$  where  $2^n < i, j, \leq 2^{n+1}$ . Hence, scaling the three dimensions of  $\rho^{(n)}(P_n|0)$  by a factor of  $\frac{1}{2}$  and shifting the result by  $\frac{1}{2}$  into the  $x$ -direction yields a representation of the lower right corner of (1) according to Definition II.4.

Similarly, the second term of (6) represents the upper left corner of (1), i.e., entries of  $P_{n+1|0}$  which correspond to row and column indices  $1 \leq i, j, \leq 2^n$ . To be more precise, each entry  $[P_{n+1|0}]_{i,j}$  is equal to  $[P_n|0]_{i,j}$  where  $1 \leq i, j, \leq 2^n$ . Hence, scaling the  $x$ - and  $y$ -coordinates of  $\rho^{(n)}(P_n|0)$  by a factor of  $\frac{1}{2}$  and shifting the resulting

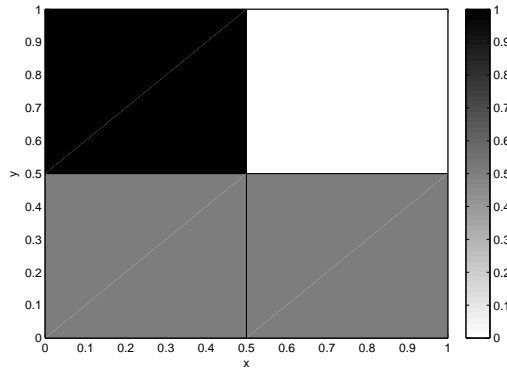
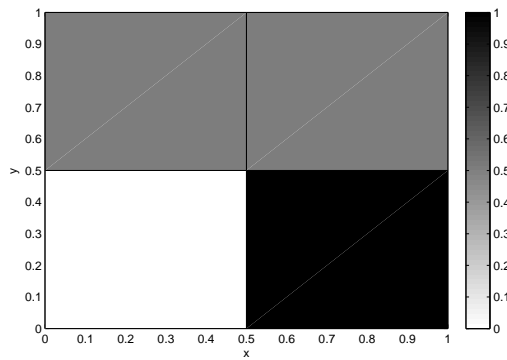
(a) Color map of  $\rho^{(1)}(P_{1|0})$ (b) Color map of  $\rho^{(1)}(P_{1|1})$ 

Fig. 3. Color map of the  $\rho^{(1)}(P_{1|0})$  and  $\rho^{(1)}(P_{1|1})$ . Each of the four squares corresponds to one of the conditional probabilities 0, 0.5 and 1.

figure by  $\frac{1}{2}$  into the  $y$ -direction yields a representation of the upper left corner  $P_{n|0}$  of (1) according to Definition II.4.

Finally, the last term of (6) represents the lower left corner of (1), i.e., entries of  $P_{n+1|0}$  with row and column indices  $2^n < i \leq 2^{n+1}$ ,  $1 \leq j \leq 2^n$ , respectively. By (1), each entry  $[P_{n+1|0}]_{i,j}$  is equal to  $\frac{1}{2} [P_{n|1}]_{i-2^n,j}$  for the same index pair  $i, j$ . Hence, scaling all coordinates of  $\rho^{(n)}(P_{n|1})$  by a factor of  $\frac{1}{2}$  yields a representation of the lower left corner of (1) according to Definition II.4. ■

Recursions (6) and (7) will be used below to obtain an iterated function system for the trapdoor channel. Recall from Theorem II.2 that an iterated function system is initialized with a single shape. Therefore, it is desirable that the right hand side of (6) just depends on  $P_{n|0}$  and the right hand side of (7) just on  $P_{n|1}$ . The following proposition introduces an affine transformation, which turns  $\rho^{(n)}(P_{n|0})$  into  $\rho^{(n)}(P_{n|1})$  and vice versa.

**Lemma II.6.** *Let  $\tau : [0, 1]^3 \rightarrow [0, 1]^3$  be defined as  $\tau(x, y, z) = (-x + 1, -y + 1, z)$ . Then*

$$\rho^{(n)}(P_{n|1}) = \tau \circ \rho^{(n)}(P_{n|0}) \quad (8)$$

$$\rho^{(n)}(P_{n|0}) = \tau \circ \rho^{(n)}(P_{n|1}), \quad (9)$$

for all  $n \in \mathbb{N}_0$ .

*Proof:* Equation (9) follows from (8) by noting that  $\tau \circ \tau = id$ . It remains to prove (8), which we do by induction. Observe that the affine transformation  $\tau$  corresponds to a counter-clockwise rotation through 180 degree about the  $z$ -axis and a translation by one into the  $x$ - and  $y$ -direction. Using this property, (8) is readily verified from Fig. 3 for  $n = 1$ . Now assume that the assertion holds for some  $n > 1$ . A direct computation of  $\tau \circ \rho^{(n+1)}(P_{n+1|0})$  using the right hand side of (6) and the induction hypotheses (8) and (9) shows that  $\tau \circ \rho^{(n+1)}(P_{n+1|0})$  is equivalent to the right hand side of (7).  $\blacksquare$

We can now state the final recursion law. A combination of Lemma II.5 and Lemma II.6, i.e., replacing  $\rho^{(n)}(P_{n|1})$  in (6) with (8) and  $\rho^{(n)}(P_{n|0})$  in (7) with (9), and using (5) yields the following theorem.

**Theorem II.7.** *The representations  $\rho^{(n+1)}(P_{n+1|0})$  and  $\rho^{(n+1)}(P_{n+1|1})$  of  $P_{n+1|0}$  and  $P_{n+1|1}$  with initial matrices  $P_{0|0} = P_{0|1} = 1$  satisfy the following recursion laws*

$$\rho^{(n+1)}(P_{n+1|0}) = \left\{ \begin{aligned} \phi_1(x, y, z) &= \left( \frac{x+1}{2}, \frac{y}{2}, \frac{[P_{n|0}]_{i,j}}{2} \right), \phi_2(x, y, z) = \left( \frac{x}{2}, \frac{y+1}{2}, [P_{n|0}]_{i,j} \right), \\ \phi_3(x, y, z) &= \left( -\frac{x-1}{2}, -\frac{y-1}{2}, \frac{[P_{n|0}]_{i,j}}{2} \right) \end{aligned} \right\} \quad (10)$$

$$\rho^{(n+1)}(P_{n+1|1}) = \left\{ \begin{aligned} \psi_1(x, y, z) &= \left( \frac{x+1}{2}, \frac{y}{2}, [P_{n|1}]_{i,j} \right), \psi_2(x, y, z) = \left( \frac{x}{2}, \frac{y+1}{2}, \frac{[P_{n|1}]_{i,j}}{2} \right), \\ \psi_3(x, y, z) &= \left( -\frac{x}{2} + 1, -\frac{y}{2} + 1, \frac{[P_{n|1}]_{i,j}}{2} \right) \end{aligned} \right\}, \quad (11)$$

where  $(i-1) \cdot 2^{-n} < x < i \cdot 2^{-n}$  and  $1-j \cdot 2^{-n} < y < 1 - (j-1) \cdot 2^{-n}$  for  $1 \leq i, j \leq 2^n$ .

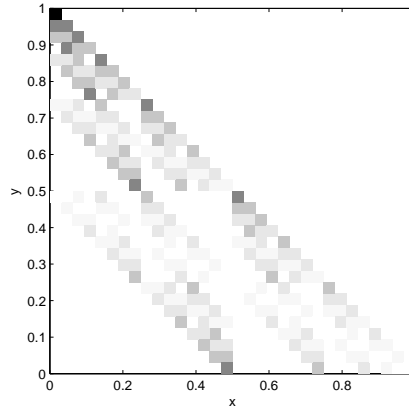
**Remark II.8.** *The restrictions of  $\phi_1, \phi_2, \phi_3$  and  $\psi_1, \psi_2, \psi_3$  to the  $x$ - and  $y$ -dimensions are contraction mappings. They compose two hyperbolic IFS with a unique attractor each. Moreover, (10) and (11) are initialized with  $P_{0|0} = 1$  and  $P_{0|1} = 1$ , respectively. Hence,  $\lim_{n \rightarrow \infty} \rho^{(n)}(P_{n|s_0})$ ,  $s_0 \in \{0, 1\}$ , can be approximated arbitrarily close by iterating (10) and (11), respectively, (according to Theorem II.2) for any initial shape  $A \in \mathcal{H}([0, 1]^3)$  such that the restriction of  $A$  to the  $z$ -dimension equals 1. Both IFS follow directly from (10) and (11) and read*

$$\left\{ [0, 1]^3; \phi_1 = \left( \frac{x+1}{2}, \frac{y}{2}, \frac{z}{2} \right), \phi_2 = \left( \frac{x}{2}, \frac{y+1}{2}, z \right), \phi_3 = \left( -\frac{x-1}{2}, -\frac{y-1}{2}, \frac{z}{2} \right) \right\}. \quad (12)$$

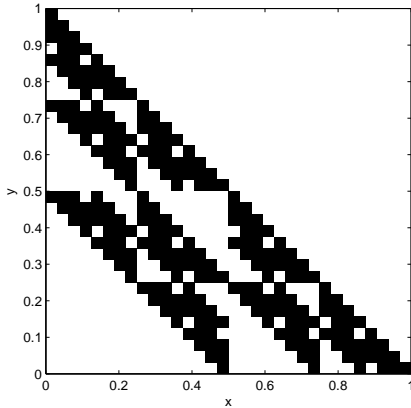
$$\left\{ [0, 1]^3; \psi_1 = \left( \frac{x+1}{2}, \frac{y}{2}, z \right), \psi_2 = \left( \frac{x}{2}, \frac{y+1}{2}, \frac{z}{2} \right), \psi_3 = \left( -\frac{x}{2} + 1, -\frac{y}{2} + 1, \frac{z}{2} \right) \right\}. \quad (13)$$

There is also a relation to the Sierpinski triangle. Observe that  $\phi_1, \phi_2$  and  $\psi_1, \psi_2$ , respectively, restricted to the  $xy$ -plane are equal to  $\varphi_1, \varphi_2$  in (4).

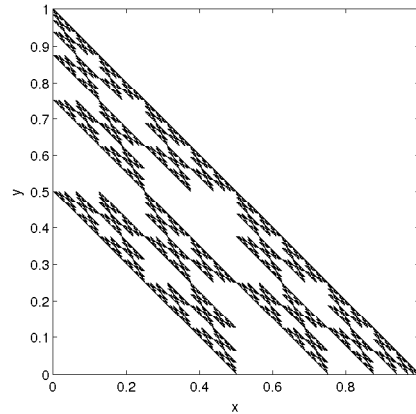




(a) The  $z$ -dimension is visualized by means of gray colors. The gray scale is the one used in Fig. 3



(b) Restriction of Fig. (a) to the  $x$ - and  $y$ -dimensions.



(c) A more accurate approximation of the fractal where the IFS (12) is restricted to the  $x$ - and  $y$ -dimensions.

Fig. 4. The result of running 4 iterations (Fig. (a), (b)) and 11 iterations (Fig. (c)) of the IFS (12). The initial shape  $A$  has been chosen to be  $\{(x, y, z) \in [0, 1]^3 : z = 1\}$ .

### III. ALGORITHMIC VIEW OF THE TRAPDOOR CHANNEL

#### A. Remarks on the Permutation Nature

The trapdoor channel has been called a permuting channel [4], where the output is a permutation of the input [5]. We point out that in general not all possible permutations of the input are feasible and that not every output is a permutation of the input. The reason that not all permutations are feasible is that the channel actions are causal, i.e., an input symbol at time  $n$  cannot become a channel output at a time instance smaller than  $n$ . Consider, for instance, a vector 101 which, when applied to a trapdoor channel with initial state 0, cannot give rise to an output 110. Next, not every output is a permutation of the input because at a certain time instance the initial state might become an output symbol and, therefore, the resulting output sequence might not be compatible with a permutation of the

input. For illustration purposes, consider again the previous example, i.e., a vector 101 and initial state 0. Two of the feasible outputs are 010 and 001 which are not permutations of 110.

### B. The Algorithm

The following recursive procedure GENERATEOUTPUTS computes the set of feasible output sequences and their likelihoods given an input sequence and an initial state.

```

procedure GENERATEOUTPUTS(in, out, state, prob)
  if in =  $\emptyset$  then
    set  $\leftarrow$  {out, prob}
  else if in[0] = state then
    out  $\leftarrow$  out + in[0]
    set  $\leftarrow$  GENERATEOUTPUTS(in.substr(1), out, state, prob)
  else
    out  $\leftarrow$  out + in[0]
    set  $\leftarrow$  GENERATEOUTPUTS(in.substr(1), out, state, 0.5 · prob)
    out[out.length() - 1]  $\leftarrow$  state  $\triangleright$  in[0] is removed from the end of out
    set  $\leftarrow$  GENERATEOUTPUTS(in.substr(1), out, in[0], 0.5 · prob)
  end if
  return set
end procedure

```

The four variables *in*, *out*, *state* and *prob* have the following meaning: *in* denotes the part of the input string that has not been processed yet; *out* indicates the part of one particular output string that has been generated so far; *state* refers to the current channel state; *prob* denotes the likelihood of *out*. The procedure is initialized with the complete input string and the initial state of the channel; *out* is initially empty while *prob* equals 1. The first **if** statement checks the simple case of the recursion, i.e., whether the input string has been processed completely. If yes, then the corresponding output *out* and its likelihood *prob* is stored and returned in *set*. Otherwise, we distinguish whether the next input symbol *in*[0] is equal to the current state. If yes, then the next output takes the value of *in*[0] (or of *state* but both are equal), i.e.,  $out \leftarrow out + in[0]$ , with probability 1 and the procedure GENERATEOUTPUTS is applied recursively to the unprocessed part of the input string, i.e., to *in*.substr(1), the substring of *in* with indices greater than 0. Clearly, *state* and *prob* do not change and, therefore, are passed unmodified to the recursive call. In the other case, i.e., when *in*[0] is not equal to the current state, the next output symbol will have a probability of 0.5 to be either *in*[0] or *state*. If *in*[0] becomes the channel output, the following state remains the same. Then the remaining input string *in*.substr(1) is processed by the recursive call GENERATEOUTPUTS(*in*.substr(1), *out*, *state*, 0.5 · *prob*). However, if *state* becomes the channel output, then the following state will be *in*[0] and the remaining input string is processed by GENERATEOUTPUTS(*in*.substr(1), *out*, *in*[0], 0.5 · *prob*). Note that a recursive implementation of

the algorithm is needed since it works for inputs of any length, which is not the case if only iterative control structures are used.

The outlined procedure gives a complete characterization of the trapdoor channel. Generating outputs and their corresponding likelihoods for a particular input sequence might be instrumental for designing codes. Finally, the design of the algorithm resembles a recursion for generating all permutations of a string (see, e.g., [8, ch. 8.3]). This gives an algorithmic justification for why some output sequences are permutations of the underlying input sequence.

#### IV. A LAGRANGE MULTIPLIER APPROACH TO THE TRAPDOOR CHANNEL

##### A. Problem Formulation

In this section, we derive an upper bound on the capacity of the trapdoor channel. Specifically, for any  $n \in \mathbb{N}$ , we find a solution to the optimization problem

$$\begin{aligned} \text{maximize} \quad & \frac{1}{n} I(X^n; Y^n | s_0) \\ &= \frac{1}{n} \sum_{i=1}^{2^n} \sum_{j=1}^{2^n} p_i [P_{n|s_0}]_{i,j} \log \frac{[P_{n|s_0}]_{i,j}}{\sum_{k=1}^{2^n} p_k [P_{n|s_0}]_{k,j}} \end{aligned} \quad (14)$$

$$\text{subject to} \quad \sum_{i=1}^{2^n} p_i = 1 \quad (15)$$

$$\sum_{k=1}^{2^n} p_k [P_{n|s_0}]_{k,j} \geq 0 \quad \text{for all } 1 \leq j \leq 2^n. \quad (16)$$

We do not have to distinguish between *lower capacity* and *upper capacity* [9, Chapter 4.6] since it does not matter whether the optimization is with respect to initial state 0 or 1 due to symmetry reasons. Constraint (16) guarantees that the argument of the logarithm does not become negative. The feasible set, defined by (15) and (16), is convex. It includes the set of probability mass functions, but might be larger. To see this note that (16) is a weighted sum of all  $p_k$  where each weight  $[P_{n|s_0}]_{k,j}$  is nonnegative. Clearly, (15) and (16) are satisfied by probability distributions. However, there might exist “distributions” which involve negative values and sum up to one but still satisfy (16). Moreover, the objective function  $n^{-1} I(X^n; Y^n | s_0)$  is concave on the set of probability distributions, which follows by using the same arguments that show that mutual information is concave on the set of input probability distributions. Consequently, the optimization problem is convex and every solution maximizes  $n^{-1} I(X^n; Y^n | s_0)$ . In the following, the maximum value is denoted as  $C_n^\uparrow$ . Taking the limit of the sequence  $(C_n^\uparrow)_{n \in \mathbb{N}}$  as  $n$  grows, one obtains either the capacity of the trapdoor channel or an upper bound on the capacity, depending on whether the limit is attained inside or outside the set of probability distributions, respectively.

##### B. Using a Result from the Literature

The reason for considering (16) and not the more natural constraints  $p_k \geq 0$  for all  $k$  is that a closed form solution can be obtained by applying the method of *Lagrange multipliers* to (14) and (15). In particular, setting the

partial derivatives of

$$\frac{1}{n}I(X^n; Y^n|s_0) + \lambda \sum_{i=1}^{2^n} p_i \quad (17)$$

with respect to each of the  $p_i$  equal to zero results in a closed form solution of the considered optimization problem.

This was done in [2, Theorem 3.3.3] for general discrete memoryless channels which are square and non singular. Note that  $P_{n|s_0}$  is square and non singular (see Lemma IV.2 (b)). Moreover, we assume that the channel  $P_{n|s_0}$  is memoryless by repeatedly using it over a large number of input blocks of length  $n$ . This has the consequence that  $C_n^\uparrow$  might be an upper bound on the capacity of a trapdoor channel that is constrained to input blocks of length  $n$ . The reason is that some input blocks might drive the channel  $P_{n|s_0}$  into the opposite state  $s_0 \oplus 1$ , i.e., the upcoming input block would see the channel  $P_{n|s_0 \oplus 1}$  (whose  $C_n^\uparrow$  is equal to  $C_n^\uparrow$  of  $P_{n|s_0}$  by symmetry). However, by assuming that the channel does not change over time, the sender always knows the channel state before a new block is transmitted. Hence,  $C_n^\uparrow$  might be an upper bound (even though it is attained on the set of probability distributions). Nevertheless, this issue can be ignored if  $n$  goes to infinity because in the asymptotic regime the channel  $P_{n|s_0}$  is used only once. But we are interested in the asymptotic regime since the limit of the sequence  $(C_n^\uparrow)_{n \in \mathbb{N}}$  is also its supremum (see Theorem IV.7).

In summary, we can apply [2, Theorem 3.3.3] which yields

$$C_n^\uparrow = \frac{1}{n} \log_2 \sum_{j=1}^{2^n} \exp_2 \left( - \sum_{i=1}^{2^n} [P_{n|s_0}^{-1}]_{j,i} H(Y^n | X^n = x_i^n) \right), \quad (18)$$

attained at

$$p_i = 2^{-C_n^\uparrow} d_i, \quad i = 1, 2, \dots, 2^n \quad (19)$$

where  $d_i$  equals

$$\sum_{j=1}^{2^n} [P_{n|s_0}^{-1}]_{j,k} \exp_2 \left( - \sum_{i=1}^M [P_{n|s_0}^{-1}]_{j,i} H(Y^n | X^n = x_i^n) \right). \quad (20)$$

Clearly,  $[p_1, \dots, p_{2^n}]$  is a probability distribution only if  $d_i \geq 0$ . Observe that the Lagrangian (17) does not involve the constraint (16). However, the proof of [2, Theorem 3.3.3] shows that  $\sum_{k=1}^{2^n} p_k [P_{n|s_0}]_{k,j}$  equals

$$\exp \left( \lambda - \sum_{i=1}^M [P_{n|s_0}^{-1}]_{j,i} H(Y^n | X^n = \mathbf{x}_i) - 1 \right) \quad (21)$$

for all  $1 \leq j \leq 2^n$ . Hence, (16) is satisfied.

We remark that (18) in matrix notation reads

$$C_n^\uparrow = \frac{1}{n} \log_2 \left[ \mathbf{1}_n^T \exp_2 \left( P_{n|s_0}^{-1} (P_{n|s_0} \circ \log_2 P_{n|s_0}) \mathbf{1}_n \right) \right]. \quad (22)$$

In the remainder, we will evaluate (22).

### C. Useful Recursions

To evaluate (22), we derive recursions for  $-(P_{n|s_0} \circ \log_2 P_{n|s_0}) 1_n$  and  $P_{n|s_0}^{-1} (P_{n|s_0} \circ \log_2 P_{n|s_0}) 1_n$ . The two expressions are formally defined next. Based on these recursions, we find exact numerical expressions for (22) in Theorem IV.7 below.

**Definition IV.1.** (a) The conditional entropy vector  $h_{n|s_0}$  of  $P_{n|s_0}$ ,  $s_0 \in \{0, 1\}$ , is defined as

$$h_{n|s_0} = \left[ H(Y^n | X^n = x_1^n) \quad \dots \quad H(Y^n | X^n = x_{2^n}^n) \right]^T \quad (23)$$

$$= -(P_{n|s_0} \circ \log_2 P_{n|s_0}) 1_n \quad (24)$$

where  $n \in \mathbb{N}_0$ .

(b) The weighted conditional entropy vector  $\omega_{n|s_0}$  of  $P_{n|s_0}$ ,  $s_0 \in \{0, 1\}$ , is defined as

$$\omega_{n|s_0} = -P_{n|s_0}^{-1} \cdot h_{n|s_0} \quad (25)$$

$$= P_{n|s_0}^{-1} (P_{n|s_0} \circ \log_2 P_{n|s_0}) 1_n \quad (26)$$

where  $n \in \mathbb{N}_0$ .

We remark that  $h_{n|s_0}$  and  $\omega_{n|s_0}$  are column vectors with  $2^n$  entries. The following two lemmas provide tools that we need for the proof of Lemma IV.4 and Lemma IV.5.

**Lemma IV.2.** (a) The trapdoor channel matrices  $P_{2n+2|0}$  and  $P_{2n+2|1}$ ,  $n \in \mathbb{N}_0$ , satisfy the following recursions:

$$P_{2n+2|0} = \begin{bmatrix} P_{2n|0} & 0 & 0 & 0 \\ \frac{1}{2}P_{2n|1} & \frac{1}{2}P_{2n|0} & 0 & 0 \\ \frac{1}{4}P_{2n|1} & \frac{1}{4}P_{2n|0} & \frac{1}{2}P_{2n|0} & 0 \\ 0 & \frac{1}{2}P_{2n|1} & \frac{1}{4}P_{2n|1} & \frac{1}{4}P_{2n|0} \end{bmatrix} \quad (27)$$

$$P_{2n+2|1} = \begin{bmatrix} \frac{1}{4}P_{2n|1} & \frac{1}{4}P_{2n|0} & \frac{1}{2}P_{2n|0} & 0 \\ 0 & \frac{1}{2}P_{2n|1} & \frac{1}{4}P_{2n|1} & \frac{1}{4}P_{2n|0} \\ 0 & 0 & \frac{1}{2}P_{2n|1} & \frac{1}{2}P_{2n|0} \\ 0 & 0 & 0 & P_{2n|1} \end{bmatrix}. \quad (28)$$

(b) Let  $M_0 := P_{2n|0}^{-1} P_{2n|1} P_{2n|0}^{-1}$  and  $M_1 := P_{2n|1}^{-1} P_{2n|0} P_{2n|1}^{-1}$ . The inverses of  $P_{2n+2|0}$  and  $P_{2n+2|1}$ ,  $n \in \mathbb{N}_0$ , satisfy the following recursions:

$$P_{2n+2|0}^{-1} = \begin{bmatrix} P_{2n|0}^{-1} & 0 & 0 & 0 \\ -M_0 & 2P_{2n|0}^{-1} & 0 & 0 \\ 0 & -P_{2n|0}^{-1} & 2P_{2n|0}^{-1} & 0 \\ 2M_0 P_{2n|1} P_{2n|0}^{-1} & -3M_0 & -2M_0 & 4P_{2n|0}^{-1} \end{bmatrix} \quad (29)$$

$$P_{2n+2|1}^{-1} = \begin{bmatrix} 4P_{2n|1}^{-1} & -2M_1 & -3M_1 & 2M_1P_{2n|0}P_{2n|1}^{-1} \\ 0 & 2P_{2n|1}^{-1} & -P_{2n|1}^{-1} & 0 \\ 0 & 0 & 2P_{2n|1}^{-1} & -M_1 \\ 0 & 0 & 0 & P_{2n|1}^{-1} \end{bmatrix}. \quad (30)$$

*Proof:* (a): Substituting  $P_{2n+2-1|0}$  and  $P_{2n+2-1|1}$  into  $P_{2n+2|0}$  and  $P_{2n+2|1}$ , where the four matrices are expressed as in (1) and (2), yields (27) and (28).

(b): Two versions of the matrix inversion lemma are [10]

$$\begin{bmatrix} A & 0 \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -D^{-1}CA^{-1} & D^{-1} \end{bmatrix} \quad (31)$$

$$\begin{bmatrix} A & B \\ 0 & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & -A^{-1}BD^{-1} \\ 0 & D^{-1} \end{bmatrix}. \quad (32)$$

Divide (27) and (28) into four blocks of equal size. A twofold application of (31) and (32), first to  $P_{2n+2|0}$  and  $P_{2n+2|1}$  and, subsequently, to each of the blocks of  $P_{2n+2|0}$  and  $P_{2n+2|1}$  yields (29) and (30). ■

A transformation relating  $P_{n|0}$  with  $P_{n|1}$ ,  $P_{n|0}^{-1}$  with  $P_{n|1}^{-1}$ ,  $h_{n|0}$  with  $h_{n|1}$  and  $\omega_{n|0}$  with  $\omega_{n|1}$  is derived next.

**Lemma IV.3.** *Let  $P_{n|0}$  and  $P_{n|1}$  be trapdoor channel matrices,  $n \in \mathbb{N}_0$ . Then we have the following identities.*

(a)

$$P_{n|1} = \tilde{I}_n P_{n|0} \tilde{I}_n \quad (33)$$

$$P_{n|0} = \tilde{I}_n P_{n|1} \tilde{I}_n. \quad (34)$$

(b)

$$P_{n|1}^{-1} = \tilde{I}_n P_{n|0}^{-1} \tilde{I}_n \quad (35)$$

$$P_{n|0}^{-1} = \tilde{I}_n P_{n|1}^{-1} \tilde{I}_n. \quad (36)$$

(c)

$$h_{n|1} = \tilde{I}_n h_{n|0} \quad (37)$$

$$h_{n|0} = \tilde{I}_n h_{n|1}. \quad (38)$$

(d)

$$\omega_{n|1} = \tilde{I}_n \omega_{n|0} \quad (39)$$

$$\omega_{n|0} = \tilde{I}_n \omega_{n|1}. \quad (40)$$

(e) *The row sums of  $P_{n|0}^{-1}$  and  $P_{n|1}^{-1}$  are 1.*

*Proof:* (a): The proof is by induction. For  $n = 0$ , the identities  $P_{0|1} = \tilde{I}_0 P_{0|0} \tilde{I}_0$  and  $P_{0|0} = \tilde{I}_0 P_{0|1} \tilde{I}_0$  clearly hold. Now suppose that (33) and (34) are true if  $n$  is replaced by  $n - 1$ . Then we have

$$\tilde{I}_n P_{n|0} \tilde{I}_n = \begin{bmatrix} 0 & \tilde{I}_{n-1} \\ \tilde{I}_{n-1} & 0 \end{bmatrix} \begin{bmatrix} P_{n-1|0} & 0 \\ \frac{1}{2} P_{n-1|1} & \frac{1}{2} P_{n-1|0} \end{bmatrix} \begin{bmatrix} 0 & \tilde{I}_{n-1} \\ \tilde{I}_{n-1} & 0 \end{bmatrix} \quad (41)$$

$$= \begin{bmatrix} \frac{1}{2} \tilde{I}_{n-1} P_{n-1|0} \tilde{I}_{n-1} & \frac{1}{2} \tilde{I}_{n-1} P_{n-1|1} \tilde{I}_{n-1} \\ 0 & \tilde{I}_{n-1} P_{n-1|0} \tilde{I}_{n-1} \end{bmatrix} \\ = \begin{bmatrix} \frac{1}{2} P_{n-1|1} & \frac{1}{2} P_{n-1|0} \\ 0 & P_{n-1|1} \end{bmatrix} \quad (42)$$

$$= P_{n-1|1} \quad (43)$$

where (41) and (43) are due to the recursive expressions (1) and (2) while (42) follows from the induction hypothesis.

It remains to show (34). But (34) is a direct consequence of the just proven equation and using the identity  $\tilde{I}_n \tilde{I}_n = I_n$ .

(b): Follows immediately from (a) and the identity  $\tilde{I}_n \tilde{I}_n = I_n$ .

(c): Equation (37) follows from

$$h_{n|1} = - (P_{n|1} \circ \log_2 P_{n|1}) 1_n \\ = - \left[ (\tilde{I}_n P_{n|0} \tilde{I}_n) \circ \log_2 (\tilde{I}_n P_{n|0} \tilde{I}_n) \right] 1_n \quad (44)$$

$$= -\tilde{I}_n (P_{n|0} \circ \log_2 P_{n|0}) \tilde{I}_n 1_n \quad (45) \\ = \tilde{I}_n h_{n|0}$$

where (44) follows by replacing  $P_{n|1}$  with (33). Observe that the left and right multiplication of  $P_{n|0}$  with  $\tilde{I}_n$  merely yields a new ordering of the elements of  $P_{n|0}$ .<sup>2</sup> Since it does not matter whether the Hadamard product and the elementwise logarithm is applied before or after sorting the elements of the underlying matrix, i.e., before or after multiplying with  $\tilde{I}_{2n}$ , (45) is true.

Equation (38) follows from (37) and the identity  $\tilde{I}_n \tilde{I}_n = I_n$ .

(d): Equation (39) follows from

$$\omega_{n|1} = -P_{n|1}^{-1} h_{n|1} \\ = -\tilde{I}_n P_{n|0}^{-1} h_{n|0} \quad (46) \\ = \tilde{I}_n \omega_{n|0},$$

where (46) follows by replacing  $P_{n|1}$  and  $h_{n|1}$  with (33) and (37), respectively, and using the identity  $\tilde{I}_n \tilde{I}_n = I_n$ .

Equation (40) follows from (39) and the identity  $\tilde{I}_n \tilde{I}_n = I_n$ .

(e): A standard way to compute  $P_{n|0}^{-1}$  is by Gauss-Jordan elimination, i.e., a sequence of elementary row operations

<sup>2</sup>To be more precise,  $[P_{n|0}]_{i,j}$  is placed at position  $(2^n + 1 - i, 2^n + 1 - j)$  for all  $1 \leq i, j \leq 2^n$ .

applied to the augmented matrix  $\begin{bmatrix} P_{n|0} & I_n \end{bmatrix}$  such that  $\begin{bmatrix} I_n & P_{n|0}^{-1} \end{bmatrix}$  eventually results. Clearly,  $P_{n|0}$  and  $I_n$  are stochastic matrices, i.e., all row sums are equal to one. Thus, at each stage of performing the elementary row operations, the row sum of the left matrix equals the row sum of the right matrix. In particular,  $P_{n|0}^{-1}$  has the same row sum as  $I_n$ . ■

We can now state the recursive laws for the *conditional entropy vector* and the *weighted conditional entropy vector*.

**Lemma IV.4.** For  $n \geq 1$ ,  $h_{2n+2|0}$  satisfies the recursion

$$h_{2n+2|0} = \begin{bmatrix} h_{2n|0} \\ \frac{1}{2}h_{2n|0} + \frac{1}{2}\tilde{I}_{2n}h_{2n|0} + 1_{2n} \\ \frac{3}{4}h_{2n|0} + \frac{1}{4}\tilde{I}_{2n}h_{2n|0} + \frac{3}{2}1_{2n} \\ \frac{1}{4}h_{2n|0} + \frac{3}{4}\tilde{I}_{2n}h_{2n|0} + \frac{3}{2}1_{2n} \end{bmatrix}. \quad (47)$$

The initial value for  $n = 0$  is given by  $h_{0|0} = 0$ .

We remark that in order to refer to the  $i$ th subvector,  $1 \leq i \leq 4$ , of the conditional entropy vector we use the superscript  $(i)$ . For instance,  $h_{2n+2|0}^{(2)}$  refers to  $\frac{1}{2}h_{2n|0} + \frac{1}{2}\tilde{I}_{2n}h_{2n|0} + 1_{2n}$ .

*Proof:* The initial value  $h_{0|0}$  can be directly computed using  $P_{0|0} = 1$  in (24). In order to show (47), we replace  $P_{2n+2|0}$  in (24) with (27) from Lemma IV.2 (a) and compute each of the four entries in (47) separately. Clearly, we have  $h_{2n+2|0}^{(1)} = -(P_{2n|0} \circ \log_2 P_{2n|0}) 1_{2n}$ , which by definition equals  $h_{2n|0}$ . The three remaining terms can be written as follows

$$\begin{aligned} h_{2n+2|0}^{(2)} &= \left[ -\frac{1}{2}P_{2n|1} \circ \log_2 \left( \frac{1}{2}P_{2n|1} \right) - \frac{1}{2}P_{2n|0} \circ \log_2 \left( \frac{1}{2}P_{2n|0} \right) \right] 1_{2n} \\ &= \left[ \frac{1}{2}P_{2n|1} - \frac{1}{2} \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) \circ \log_2 \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) + \frac{1}{2}P_{2n|0} - \frac{1}{2}P_{2n|0} \circ \log_2 P_{2n|0} \right] 1_{2n} \end{aligned} \quad (48)$$

$$\begin{aligned} &= 1_{2n} - \frac{1}{2}\tilde{I}_{2n} (P_{2n|0} \circ \log_2 P_{2n|0}) 1_{2n} + \frac{1}{2}h_{2n|0} \\ &= \frac{1}{2}h_{2n|0} + \frac{1}{2}\tilde{I}_{2n}h_{2n|0} + 1_{2n} \end{aligned} \quad (49)$$

$$\begin{aligned} h_{2n+2|0}^{(3)} &= \left[ -\frac{1}{4}P_{2n|1} \circ \log_2 \left( \frac{1}{4}P_{2n|1} \right) - \frac{1}{4}P_{2n|0} \circ \log_2 \left( \frac{1}{4}P_{2n|0} \right) - \frac{1}{2}P_{2n|0} \circ \log_2 \left( \frac{1}{2}P_{2n|0} \right) \right] 1_{2n} \\ &= \left[ \frac{1}{2}P_{2n|1} - \frac{1}{4} \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) \circ \log_2 \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) + P_{2n|0} - \frac{3}{4}P_{2n|0} \circ \log_2 P_{2n|0} \right] 1_{2n} \end{aligned} \quad (50)$$

$$\begin{aligned} &= \frac{3}{2}1_{2n} - \frac{1}{4}\tilde{I}_{2n} (P_{2n|0} \circ \log_2 P_{2n|0}) 1_{2n} + \frac{3}{4}h_{2n|0} \\ &= \frac{3}{4}h_{2n|0} + \frac{1}{4}\tilde{I}_{2n}h_{2n|0} + \frac{3}{2}1_{2n} \end{aligned} \quad (51)$$

$$\begin{aligned} h_{2n+2|0}^{(4)} &= \left[ -\frac{1}{2}P_{2n|1} \circ \log_2 \left( \frac{1}{2}P_{2n|1} \right) - \frac{1}{4}P_{2n|1} \circ \log_2 \left( \frac{1}{4}P_{2n|1} \right) - \frac{1}{4}P_{2n|0} \circ \log_2 \left( \frac{1}{4}P_{2n|0} \right) \right] 1_{2n} \\ &= \left[ P_{2n|1} - \frac{3}{4} \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) \circ \log_2 \left( \tilde{I}_{2n}P_{2n|0}\tilde{I}_{2n} \right) + \frac{1}{2}P_{2n|0} - \frac{1}{4}P_{2n|0} \circ \log_2 P_{2n|0} \right] 1_{2n} \end{aligned} \quad (52)$$



$$\begin{aligned}
&= \frac{3}{2}1_{2n} - \frac{3}{4}\tilde{I}_{2n} (P_{2n|0} \circ \log_2 P_{2n|0}) 1_{2n} + \frac{1}{4}h_{2n|0} - \frac{3}{4}\tilde{I}_{2n} (P_{2n|0} \circ \log_2 P_{2n|0}) 1_{2n} \\
&= \frac{1}{4}h_{2n|0} + \frac{3}{4}\tilde{I}_{2n}h_{2n|0} + 1_{2n}
\end{aligned} \tag{53}$$

where (48), (50) and (52), respectively, follow from expanding the logarithms in the previous equation and replacing the channel matrices corresponding to initial state one with (33). The first term in (49), (51) and (53), respectively, follows from the multiplication of the weighted matrices  $P_{2n|0}$  and  $P_{2n|1}$  with  $1_n$ . The second term in (49), (51) and (53), respectively, follows by using the fact that it does not matter whether the Hadamard product and the elementwise logarithm is applied before or after sorting the elements of the underlying matrix, i.e., before or after multiplying with  $\tilde{I}_{2n}$ . ■

**Lemma IV.5.** (a) For  $n \geq 1$ ,  $\omega_{2n|0}$  satisfies the recursion

$$\omega_{2n|0} = \begin{bmatrix} \omega_{2n-2|0} \\ \omega_{2n-2|0} - 2 \cdot 1_{2n-2} \\ \omega_{2n-2|0} - 2 \cdot 1_{2n-2} \\ \omega_{2n-2|0} \end{bmatrix} \tag{54}$$

with initial value  $\omega_{0|0} = 0$ .

(b) For  $n \geq 1$ ,  $\omega_{2n+1|0}$  satisfies the recursion

$$\omega_{2n+1|0} = \begin{bmatrix} \omega_{2n-1|0} \\ \tilde{I}_{2n-1}\omega_{2n-1|0} \\ \omega_{2n-1|0} - 2 \cdot 1_{2n-1} \\ \tilde{I}_{2n-1}\omega_{2n-1|0} - 2 \cdot 1_{2n-1} \end{bmatrix} \tag{55}$$

with initial value  $\omega_{1|0} = \begin{bmatrix} 0 & -2 \end{bmatrix}^T$ .

We remark that in order to refer to the  $i$ th subvector,  $1 \leq i \leq 4$ , of the weighted conditional entropy vector we use the superscript  $(i)$ . For instance,  $\omega_{2n|0}^{(2)}$  refers to  $\omega_{2n-2|0} - 2 \cdot 1_{2n-2}$ .

*Proof:* (a): We first show by induction that (54) holds. The case  $n = 0$  can be verified using Definition IV.1 (b) with  $P_{0|0} = P_{0|0}^{-1} = 1$ . Now assume that (54) holds for some  $n$ . In order to show (54) for  $n + 1$ , we evaluate  $\omega_{2n+2|0}$  using (26) and replacing  $P_{2n+2|0}^{-1}$  and  $h_{2n+2|0}$  with (29) and (47). Then we have

$$\omega_{2n+2|0} = \begin{bmatrix} -P_{2n|0}^{-1}h_{2n+2|0}^{(1)} \\ P_{2n|0}^{-1} \left( P_{2n|1}P_{2n|0}^{-1}h_{2n+2|0}^{(1)} - 2h_{2n+2|0}^{(2)} \right) \\ P_{2n|0}^{-1} \left( h_{2n+2|0}^{(2)} - 2h_{2n+2|0}^{(3)} \right) \\ M_0 \left( -2P_{2n|1}P_{2n|0}^{-1}h_{2n+2|0}^{(1)} + 3h_{2n+2|0}^{(2)} + 2h_{2n+2|0}^{(3)} \right) - 4P_{2n|0}^{-1}h_{2n+2|0}^{(4)} \end{bmatrix}. \tag{56}$$

Recall from Lemma IV.4 that  $h_{2n+2|0}^{(1)} = h_{2n|0}$ . Hence, by definition, the first entry of (56) is equal to  $\omega_{2n|0}$ .

The second entry of (56) is derived as follows. Replacing  $h_{2n+2|0}^{(1)}$  and  $h_{2n+2|0}^{(2)}$  with the corresponding expressions

from (47), we obtain

$$\omega_{2n+2|0}^{(2)} = P_{2n|0}^{-1} \left( P_{2n|1} P_{2n|0}^{-1} h_{2n|0} - h_{2n|0} - \tilde{I}_{2n} h_{2n|0} - 2 \cdot 1_{2n} \right). \quad (57)$$

In order to simplify (57), observe that

$$- \tilde{I}_{2n} \omega_{2n|0} + \omega_{2n|0} = 0 \quad (58)$$

since  $\omega_{2n|0}$  is a palindromic vector by hypothesis. A further manipulation of (58), namely using (25), (36) and the relation  $\tilde{I}_{2n} \tilde{I}_{2n} = I_{2n}$ , yields

$$P_{2n|0}^{-1} \cdot h_{2n|0} - P_{2n|1}^{-1} \tilde{I}_{2n} \cdot h_{2n|0} = 0 \quad (59)$$

which implies

$$P_{2n|1} P_{2n|0}^{-1} h_{2n|0} - \tilde{I}_{2n} \cdot h_{2n|0} = 0. \quad (60)$$

Using (60), the definition of  $\omega_{2n|0}$  and Lemma IV.3 (e), i.e., that  $P_{2n|0}^{-1}$  is a stochastic matrix, in (57) we obtain  $\omega_{2n+2|0}^{(2)} = \omega_{2n|0} - 2 \cdot 1_{2n}$ .

The third entry of (56) is derived as follows. After replacing  $h_{2n+2|0}^{(2)}$  and  $h_{2n+2|0}^{(3)}$  in (56) with the corresponding expressions from (47), it can be directly seen that  $\omega_{2n+2|0}^{(3)} = \omega_{2n|0} - 2 \cdot 1_{2n}$ .

Regarding the fourth entry in (56), we begin with the first term in parentheses, i.e.,

$$\begin{aligned} & -2P_{2n|1} P_{2n|0}^{-1} h_{2n+2|0}^{(1)} + 3h_{2n+2|0}^{(2)} + 2h_{2n+2|0}^{(3)} \\ &= -2 \left( P_{2n|1} P_{2n|0}^{-1} h_{2n+2|0}^{(1)} - 2h_{2n+2|0}^{(2)} \right) - \left( h_{2n+2|0}^{(2)} - 2h_{2n+2|0}^{(3)} \right) \end{aligned} \quad (61)$$

$$= -3P_{2n|0} \left( \omega_{2n|0} - 2 \cdot 1_{2n} \right). \quad (62)$$

Equation (62) holds since the first and the second parentheses of (61) are equal to  $P_{2n|0} \omega_{2n+2|0}^{(2)}$  and  $P_{2n|0} \omega_{2n+2|0}^{(3)}$ , respectively, which follows from (56) by inspection. Moreover,  $\omega_{2n+2|0}^{(2)}$  and  $\omega_{2n+2|0}^{(3)}$  are equal to  $\omega_{2n|0} - 2 \cdot 1_{2n}$  as we just have shown. Hence, using (62) in  $\omega_{2n+2|0}^{(4)}$  and replacing  $h_{2n+2|0}^{(4)}$  with the corresponding expression from (47) and  $M_0$  with its definition from Lemma IV.2 (b), we obtain

$$\begin{aligned} \omega_{2n+2|0}^{(4)} &= P_{2n|0}^{-1} \left( -3P_{2n|1} \left( \omega_{2n|0} - 2 \cdot 1_{2n} \right) - h_{2n|0} - 3\tilde{I}_{2n} h_{2n|0} - 6 \cdot 1_{2n} \right) \\ &= 3P_{2n|0}^{-1} \left( -P_{2n|1} \omega_{2n|0} - \tilde{I}_{2n} h_{2n|0} \right) + 6P_{2n|0}^{-1} \left( P_{2n|1} 1_{2n} - 1_{2n} \right) - P_{2n|0}^{-1} h_{2n|0} \\ &= -P_{2n|0}^{-1} h_{2n|0} \\ &= \omega_{2n|0}. \end{aligned} \quad (63)$$

Observe that the first parentheses in (63), which is equal to the left hand side of (60), evaluates to 0. Also the second parentheses in (63) evaluates to 0 since  $P_{2n|1}$  is a stochastic matrix.

(b): Recall the recursions

$$P_{2n+2|0} = \begin{bmatrix} P_{2n+1|0} & 0 \\ \frac{1}{2}P_{2n+1|1} & \frac{1}{2}P_{2n+1|0} \end{bmatrix} \quad (64)$$

$$P_{2n+2|0}^{-1} = \begin{bmatrix} P_{2n+1|0}^{-1} & 0 \\ P_{2n+1|0}^{-1}P_{2n+1|1}P_{2n+1|0}^{-1} & 2P_{2n+1|0}^{-1} \end{bmatrix}. \quad (65)$$

The first  $2^{2n+1}$  entries, i.e., the first half, of  $\omega_{2n+2|0}$  are equal to  $P_{2n+1|0}^{-1} (P_{2n+1|0} \circ \log_2 P_{2n+1|0}) 1_{2n+1}$ , which in turn is equal to  $\omega_{2n+1|0}$ . This follows from a straightforward computation using Definition IV.1(b) together with (64) and (65). Hence, under consideration of (54), we have

$$\omega_{2n+1|0} = \begin{bmatrix} \omega_{2n|0} \\ \omega_{2n|0} - 2 \cdot 1_{2n} \end{bmatrix}. \quad (66)$$

Equivalently,  $\omega_{2n-1|0}$  is equal to the first  $2^{2n-1}$  entries of  $\omega_{2n|0}$ . Then we have

$$\omega_{2n|0} = \begin{bmatrix} \omega_{2n-1|0} \\ \tilde{I}_{2n-1} \cdot \omega_{2n-1|0} \end{bmatrix}. \quad (67)$$

In order to derive the second entry of (67) observe that the multiplication of  $\omega_{2n-1|0}$  with  $\tilde{I}_{2n-1}$  turns  $\omega_{2n-1|0}$  upside down (i.e., the last entry of  $\omega_{2n-1|0}$  becomes the first entry, the second last entry becomes the second entry and so on). Applying this multiplication to  $\omega_{2n-1|0}$ , which is written in the form of (66), and using the fact that  $\omega_{2n-2|0}$  is a palindromic vector, we see that  $\tilde{I}_{2n-1} \cdot \omega_{2n-1|0}$  is equal to the last  $2^{2n-1}$  entries, i.e., second half, of the vector (54). By replacing  $\omega_{2n|0}$  in (66) with (67), we obtain (55). The initial value  $\omega_1 = \begin{bmatrix} 0 & -2 \end{bmatrix}^T$  follows directly by evaluating (54) for  $n = 1$  and taking the first two entries. ■

**Remark IV.6.** *The recursions derived in Lemma IV.4 and IV.5 are with respect to initial state  $s_0 = 0$ . They can be easily converted to recursions with respect to initial state  $s_0 = 1$  by using (37) and (39) from Lemma IV.3.*

#### D. Proof of the Main Result

By evaluating (18) based on Lemma IV.5, we find exact solutions to the optimization problem (14)-(16).

**Theorem IV.7.** *Consider the convex optimization problem (14) to (16). The absolute maximum for input blocks of even length  $2n$  is*

$$C_{2n}^\uparrow = \frac{1}{2} \log_2 \left( \frac{5}{2} \right) \quad (68)$$

for all  $n \in \mathbb{N}$ . For input blocks of odd length  $2n - 1$ , the absolute maximum is

$$C_{2n-1}^\uparrow = \frac{1}{2n-1} \left[ \log_2 \left( \frac{5}{4} \right) + (n-1) \cdot \log_2 \left( \frac{5}{2} \right) \right], \quad (69)$$

where  $n \in \mathbb{N}$ .

*Proof:* Without loss of generality, the initial state is assumed to be  $s_0 = 0$ . Recall (22), which for input blocks of length  $2n + k$  reads as

$$C_{2n+k}^\uparrow = \frac{1}{2n+k} \log_2 \left[ 1_{2n+k}^T \exp_2 (\omega_{2n+k|0}) \right] \quad (70)$$

where  $n \in \mathbb{N}_0, k = 1, 2$ . For  $n = 0$ , a straightforward computation shows that  $C_1^\uparrow = \log_2 \left( \frac{5}{4} \right)$  and  $C_2^\uparrow = \frac{1}{2} \log_2 \left( \frac{5}{2} \right)$ . Now assume that (68) and (69) hold for some  $n$ . In particular, suppose

$$1_{2n}^T \exp_2 (\omega_{2n|0}) = \left( \frac{5}{2} \right)^n \quad (71)$$

and

$$1_{2n-1}^T \exp_2 (\omega_{2n-1|0}) = \frac{5}{4} \left( \frac{5}{2} \right)^{n-1}. \quad (72)$$

Replacing  $\omega_{2n+2|0}$  and  $\omega_{2n+1|0}$  with the recursions derived in Lemma IV.5, we obtain

$$\begin{aligned} 1_{2n+2}^T \exp_2 (\omega_{2n+2|0}) &= 1_{2n}^T [2 \exp_2 (\omega_{2n|0}) + 2 \exp_2 (\omega_{2n|0} - 2 \cdot 1_{2n})] \\ &= (2 + 2 \cdot 2^{-2}) 1_{2n}^T \exp_2 (\omega_{2n|0}) \end{aligned}$$

and

$$\begin{aligned} 1_{2n+1}^T \exp_2 (\omega_{2n+1|0}) &= 1_{2n-1}^T [2 \exp_2 (\omega_{2n-1|0}) + 2 \exp_2 (\omega_{2n-1|0} - 2 \cdot 1_{2n})] \\ &= (2 + 2 \cdot 2^{-2}) 1_{2n-1}^T \exp_2 (\omega_{2n-1|0}). \end{aligned}$$

Hence, using (70) and the induction hypotheses (71) and (72), we have

$$\begin{aligned} C_{2n+2}^\uparrow &= \frac{1}{2n+2} \log_2 [(2 + 2 \cdot 2^{-2}) 1_{2n}^T \exp_2 (\omega_{2n|0})] \\ &= \frac{1}{2} \log_2 \left( \frac{5}{2} \right) \end{aligned}$$

and

$$\begin{aligned} C_{2n+1}^\uparrow &= \frac{1}{2n+1} \log_2 [(2 + 2 \cdot 2^{-2}) 1_{2n-1}^T \exp_2 (\omega_{2n-1|0})] \\ &= \frac{1}{2n+1} \left[ \log_2 \left( \frac{5}{4} \right) + n \cdot \log_2 \left( \frac{5}{2} \right) \right]. \end{aligned}$$

■

**Remark IV.8.** Observe that  $\lim_{n \rightarrow \infty} C_{2n+1}^\uparrow = \frac{1}{2} \log_2 \left( \frac{5}{2} \right)$ , where convergence is from below. Hence, we have

$$\max_{n \in \mathbb{N}} C_n^\uparrow = \frac{1}{2} \log_2 \left( \frac{5}{2} \right).$$

Unfortunately, the distributions corresponding to (68) and (69) involve negative “probabilities” – otherwise the capacity of the trapdoor channel would have been established. We state this as a formal remark.

**Remark IV.9.** Condition (20) does not hold for all  $k = 1, \dots, 2^n$ , which can be seen as follows. For a trapdoor channel  $P_{n|0}$ , we have

$$\left[ d_k \right]_{1 \leq k \leq 2^n} = \left( P_{n|0}^{-1} \right)^T \exp_2 (\omega_n). \quad (73)$$

Applying (31) to  $P_{n|0}$ , which is written in the form of (1), and taking the transpose, then applying (31) to the right bottom block of this matrix and taking the transpose and so on eventually shows that the second last row of  $\left( P_{n|0}^{-1} \right)^T$  equals

$$\left[ 0 \quad \dots \quad 0 \quad 2^{n-1} \quad -2^{n-1} \right].$$

Moreover, using Lemma IV.5, it follows that the second to last entry and the last entry in  $\omega_n$  equals  $-2$  and  $0$ , respectively. Inserting the gathered quantities into (73) yields

$$d_{2^n-1} = -3 \cdot 2^{n-3} < 0, \quad n \in \mathbb{N}.$$

## V. CONCLUSIONS

We have presented two different views on the trapdoor channel. The fractal view was motivated by the wish to find an explicit expression for the trapdoor channel – a feature which would greatly simplify the capacity problem. Furthermore, the various views motivate using tools from other fields, e.g., fractal geometry.

Subsequently, we have focused on the convex optimization problem (14) to (16) where the feasible set is larger than the probability simplex. An absolute maximum of the  $n$ -letter mutual information was established for any  $n \in \mathbb{N}$  by using the method of Lagrange multipliers. The same absolute maximum  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right) \approx 0.6610$  b/u results for all even  $n$  and the sequence of absolute maxima corresponding to odd block lengths converges from below to  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right)$  b/u as the block length increases. Unfortunately, all absolute maxima are attained outside the probability simplex. Hence, instead of establishing the capacity of the trapdoor channel, we have shown only that  $\frac{1}{2} \log_2 \left( \frac{5}{2} \right)$  b/u is an upper bound on the capacity. This upper bound is, to the best of our knowledge, the tightest known bound. Notably, this upper bound is strictly smaller than the feedback capacity [5]. Moreover, the result gives an indirect justification that the capacity of the trapdoor channel is attained on the boundary of the probability simplex.

## ACKNOWLEDGMENT

The author is supported by the German Ministry of Education and Research in the framework of the Alexander von Humboldt-Professorship and would like to thank Prof. Haim Permuter who suggested to use [2, Theorem 3.3.3]. Moreover, the author wishes to thank Prof. Gerhard Kramer and Prof. Tsachy Weissman for helpful discussions.

## REFERENCES

- [1] D. Blackwell, *Information Theory*, E. F. Beckenbach, Ed. McGraw-Hill Book Co., New York, 1961, vol. Modern Mathematics for the Engineer.
- [2] R. Ash, *Information Theory*. Interscience Publishers, 1965.
- [3] K. Kobayashi and H. Morita, “An input/output recursion for the trapdoor channel,” in *Proc. IEEE Int. Symp. Inf. Theory*, Lausanne, Switzerland, Jun. 30–Jul. 5 2002, p. 423.
- [4] R. Ahlswede and A. H. Kaspi, “Optimal coding strategies for certain permuting channels.” *IEEE Trans. Inf. Theory*, vol. 33, no. 3, pp. 310–314, 1987.
- [5] H. Permuter, P. Cuff, B. Van Roy, and T. Weissman, “Capacity of the trapdoor channel with feedback,” *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 3150–3165, Jul. 2008.
- [6] M. Barnsley, *Fractals Everywhere*. Academic Press, Inc., 1988.
- [7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons, Inc., 1991.
- [8] E. Roberts, *Programming Abstractions in C++*. Prentice Hall, 2014.
- [9] R. G. Gallager, *Information Theory and Reliable Communication*. John Wiley & Sons, Inc., 1968.
- [10] G. H. Golub and C. F. van Van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.