Computability of the Channel Reliability Function and Related Bounds

Holger Boche

Chair of Theoretical Information Technology
Technical University of Munich
D-80333 Munich, Germany
and
Munich Center for Quantum Science
and Technology (MCQST)
D-80799 Munich, Germany
Email: boche@tum.de

Christian Deppe

Institute for Communications Engineering
Technical University of Munich
D-80333 Munich, Germany
Email: christian.deppe@tum.de

Abstract

The channel reliability function is an important tool that characterizes the reliable transmission of messages over communication channels. For many channels, only the upper and lower bounds of the function are known. In this paper we analyze the computability of the reliability function and its related functions. We show that the reliability function is not a Turing computable performance function. The same also applies to the functions of the sphere packing bound and the expurgation bound. Furthermore, we consider the R_∞ function and the zero-error feedback capacity, since they play an important role in the context of the reliability function. Both the R_∞ function and the zero-error feedback capacity are not Banach Mazur computable. We show that the R_∞ function is additive. The zero-error feedback capacity is super-additive and we characterize its behavior.

Keywords: computability, channel reliability function, zero-error feedback capacity

1. Introduction

Shannon laid the foundations of information theory in his 1948 paper [1], characterizing important properties of communication channels. Given a transmission rate R less than the capacity C of the channel, the probability of an erroneous decoding with respect to an optimal code decreases exponentially fast for increasing code-length $n \in \mathbb{N}$. Shannon introduced the channel reliablity function $E(R)$ as the exponent of this exponential decrease in dependence of the transmission rate R.

It is a high priority goal in information theory to find a closed form expression for the channel reliability function. This formula should be a simple computation determined by the parameters of the communication task. Of course, we must specify what a closed form expression is. In [2] by Chow and in [3] by Borwein and Crandall, examples for specifications are given. In all the approaches in [2] and [3], the closed form representation is always coupled with the fact that the corresponding functions can be computed algorithmically using a digital computer. This can be done in a very precise manner, depending on the inputs from their domain of definition.

Shannon's characterization of the capacity for the transmission of messages via the discrete memoryless channel (DMC) in [1], Ahlswede's characterization of the capacity for the transmission of messages via the multiple access channel in [4], and Ahlswedes and Dueck's characterization of the identification capacity for DMCs in [5] are important examples of closed-form solutions through elementary functions, according to Chow and Borwein/Crandall. These are all examples of computability of the corresponding performance functions in the above sense. The precise meaning of computability defined by Turing will be included later in section 2. Lovasz's characterization of the zero error capacity for the pentagram is also an example of a closed form number corresponding to Chow and can be computed algorithmically, and this is desirable as well. However, for a cyclical heptagon, the characterization is still pending for the zero error capacity. It is also unclear whether the zero error capacities of DMCs even assume computable values for computable channels. Furthermore, it is unclear whether the broadcast capacity region can be algorithmically computed.

In this paper, we give a negative answer to the question of whether the channel reliability function and several related bounds are algorithmically computable.

A great deal of research has been conducted on the topic of the channel reliability function so far. However, many problems regarding its behavior are still open (see surveys [6] and [7]). In fact, a characterization of the channel reliability function is not even known for binary-input-binary-output channels. This is why many efforts have been made to find corresponding computable lower and upper bounds (see [8, 9, 10]).

It is difficult to determine the behavior of the channel reliability function over the entire interval (0, C). There are approaches that have tried to algorithmically compute the reliability function, meaning they consider sequences of upper and lower bounds. The first paper in this direction was [11], by Shannon, Gallager and Berlekamp. We have now asked whether it is possible to compute the reliability function in this way. To formulate this, we use the theory of Turing computability [12]. In general, a function can be computed if an algorithm can be formulated for it. The strongest model for computability is the Turing machine. Turing machines make the concepts of algorithms and computability mathematically comprehensible, that is, they formalize these concepts. In contrast to a physical computer, a Turing machine is a purely mathematical object and can be examined using mathematical methods. It is important to note that the Turing machine describes the ultimate performance limit that can be achieved by today's digital computers, and even by super computers. A Turing machine represents an algorithm or a program. A computation consists of the step-by-step manipulation of symbols or characters, which are written to and read from a memory tape, according to certain rules. Strings of these symbols can be interpreted in different ways, including numbers. In order to perform computations on abstract sets, the elements of such a set have to be encoded into the strings of symbols on the tape. This way, Turing computability can be defined for the real and complex numbers, for example. Often, however, there does not exists an algorithm on a Turing machine which captures the behaviour of some general function on some abstract set. Then one can ask the weaker question of whether it is possible to approximate the function in a computable way. For this we need computable sequences of computable upper and lower bounds. This analysis is also necessary for the reliability function and we have carried this out. Unfortunately, our results prove to be negative. The reliability function is not a Turing computable performance function as a function of the channel as an input. We furthermore consider the R_∞ function, the function for the sphere packing bound, the function for the expurgation bound and the zero-error feedback capacity, all of which are closely related to the reliability function. We consider all of these functions as functions of the channel. The structure of the paper is as follows. We start in Section 2 with the basic definitions and known statements that we need. In Section 3 we first consider the R_∞ function. We consider the decidability of connected sets with the R_∞ function and show that only an approximation from below is possible. This has consequences for the sphere packing bound and we show that this is not a Turing computable performance function. In Section 4 we then consider the reliability function and show that it is also not a Turing computable performance function. We can show the same for the expurgation bound. In Section 5 we consider the zero-error feedback capacity. It is closely related to the R_∞ function. First we answer a question for the zero-error capacity with feedback (which Alon asked in [13]) for the case without feedback (we examined this in [14]). Then we show that the zero-error feedback capacity is not Banach-Mazur computable. Furthermore, the zero-error feedback capacity cannot be approximated by computable increasing sequences of computable functions. We characterize the superadditivity of the zero-error feedback capacity and show that the R_∞ function is additive. In section 6 we analyze the behavior of the expurgation-bound rates. In conclusion, we summarize what our results mean for the channel reliability function. Our results show that in general, there can be no simple recursive closed form formula for the channel reliability function in a very precise interval.

2. Definitions and Basic Results from Computability Theory and Information Theory

2.1. Basic Concepts from Computability Theory

In this section we give the basic definitions and results that we need for this work. We start with the most important definitions of computability. To define computability, we use the concept of a Turing machine [12].

Turing Machines are a mathematical model of what we intuitively understand as computation machines. In this sense, they yield an abstract idealization of today's real-world computers. Any algorithm that can be executed by a real-world computer can, in theory, be simulated by a Turing machine, and vice versa. In contrast to real-world computers, however, Turing Machines are not subject to any restrictions regarding energy consumption, computation time or memory size. All computation steps on a Turing machine are assumed to be executed with zero chance of error.

Recursive functions, more specifically referred to as *μ-recursive functions*, form a special subset of the set $\bigcup_{n=0}^{\infty} \left\{ f : \mathbb{N}^n \hookrightarrow \mathbb{N} \right\}$, where we use the symbol "$\hookrightarrow$" to denote a *partial mapping.* The set of recursive functions characterizes the notion of computability through a different approach. Turing machines and recursive functions are equivalent in the following sense: a function $f : \mathbb{N}^n \hookrightarrow \mathbb{N}$ is computable by a Turing machine if and only if it is a recursive function.

In the following, we will introduce some definitions from *computable analysis* [15, 16, 17], which we will subsequently apply.

Definition 1. *A sequence of rational numbers $\{r_n\}_{n\in\mathbb{N}}$ is called a computable sequence if there exist recursive functions $a, b, s : \mathbb{N} \to \mathbb{N}$ with $b(n) \neq 0$ for all $n \in \mathbb{N}$ and*

$$r_n = (-1)^{s(n)} \frac{a(n)}{b(n)}, \quad n \in \mathbb{N}.$$

Definition 2. *We say that a computable sequence* $\{r_n\}_{n\in\mathbb{N}}$ *of rational numbers converges effectively, i.e., computably, to a number* x*, if a recursive function* $a:\mathbb{N}\to\mathbb{N}$ *exists such that* $|x-r_n|<\frac{1}{2^N}$ *for all* $N\in\mathbb{N}$ *and all* $n\in\mathbb{N}$ *with* $n\geq a(N)$ *applies.*

We can now introduce computable numbers.

Definition 3. *A real number* x *is said to be computable if there exists a computable sequence of rational numbers* $\{r_n\}_{n\in\mathbb{N}}$*, such that* $|x-r_n|<2^{-n}$ *for all* $n\in\mathbb{N}$*. We denote the set of computable real numbers by* $\mathbb{R}_c$*.*

Next we need suitable subsets of the natural numbers.

Definition 4. *A set* $A\subset\mathbb{N}$ *is called recursive if there exists a computable function* f*, such that* $f(x)=1$ *if* $x\in A$ *and* $f(x)=0$ *if* $x\in A^c$*.*

Definition 5. *A set* $A\subset\mathbb{N}$ *is recursively enumerable if there exists a recursive function whose domain is exactly* A*.*

Definition 6. *Let* $A\subset\mathbb{N}$ *be recursively enumerable. A function* $f:A\to\mathbb{N}$ *is called partial recursive if there exists a Turing machine* TM_f *which computes* f*.*

2.2. Basic Concepts of Information Theory

To define the reliability function and its related functions, we first need the definition of a discrete memoryless channel. In the theory of transmission, the receiver must be in a position to successfully decode all the messages transmitted by the sender.

Let $\mathcal{X}$ be a finite alphabet. We denote the set of probability distributions by $\mathcal{P}(\mathcal{X})$. We define the set of computable probability distributions $\mathcal{P}_c(\mathcal{X})$ as the set of all probability distributions $P\in\mathcal{P}(X)$ such that $P(x)\in\mathbb{R}_c$ for all $x\in\mathcal{X}$. Furthermore, for finite alphabets $\mathcal{X}$ and $\mathcal{Y}$, let $\mathcal{C}H(\mathcal{X},\mathcal{Y})$ be the set of all conditional probability distributions (or channels) $P_{Y|X}:\mathcal{X}\to\mathcal{P}(\mathcal{Y})$. $\mathcal{C}H_c(\mathcal{X},\mathcal{Y})$ denotes the set of all computable conditional probability distributions, i.e., $P_{Y|X}(\cdot|x)\in\mathcal{P}_c(\mathcal{Y})$ for every $x\in\mathcal{X}$.

Let $M \subset \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$. We call M semi-decidable if and only if there is a Turing machine TM_M that either stops or computes forever, depending on whether $W \in M$ is true. This means TM_M accepts exactly the elements of M and for an input $W \in M^c = \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \setminus M$, computes forever.

Definition 7. *A discrete memoryless channel (DMC) is a triple $(\mathcal{X}, \mathcal{Y}, W)$, where $\mathcal{X}$ is the finite input alphabet, $\mathcal{Y}$ is the finite output alphabet, and $W(y|x) \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ with $x \in \mathcal{X}$, $y \in \mathcal{Y}$. The probability that a sequence $y^n \in \mathcal{Y}^n$ is received if $x^n \in \mathcal{X}^n$ was sent is defined by*

$$W^n(y^n|x^n) = \prod_{j=1}^{n} W(y_j|x_j).$$

Definition 8. *A (deterministic) block code $\mathcal{C}(n)$ with rate R and block length n consists of*

- *a message set $\mathcal{M} = \{1, 2, ..., M\}$ with $M = 2^{nR} \in \mathbb{N}$,*
- *an encoding function $e : \mathcal{M} \to \mathcal{X}^n$,*
- *a decoding function $d : \mathcal{Y}^n \to \mathcal{M}$.*

We call such a code an (R, n)-code.

Definition 9. *Let $(\mathcal{X}, \mathcal{Y}, W)$ be a DMC. By $\mathcal{C}(n)$, we denote a code with block length n and message set $\mathcal{M}$.*

1. *The individual message probability of error is defined by the conditional probability of error, given that message $m \in \mathcal{M}$ is transmitted:*

$$P_e(\mathcal{C}(n), W, m) = Pr\{d(Y^n) \neq m | X^n = e(m)\}.$$

2. *We define the average probability of error by*

$$P_{e,\text{av}}(\mathcal{C}(n), W) = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} P_e(\mathcal{C}(n), W, m).$$

$P_{e,\text{av}}(W, R, n)$ denotes the minimum error probability $P_{e,\text{av}}(\mathcal{C}(n), W)$ over all codes $\mathcal{C}(n)$ of block length n and with message set $\mathcal{M} = 2^{nR}$.

3. *We define the maximal probability of error by*

$$P_{e,\max}(\mathcal{C}(n), W) = \max_{m \in \mathcal{M}} P_e(\mathcal{C}(n), W, m).$$

$P_{e,\max}(W, R, n)$ denotes the minimum error probability $P_{e,\max}(\mathcal{C}(n), W)$ over all codes $\mathcal{C}(n)$ of block length n and with message set $\mathcal{M} = 2^{nR}$.

4. *The Shannon capacity for a channel $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ is defined by*

$$C(W) := \sup\{R : \lim_{n \to \infty} P_{e,\max}(W, R, n) = 0\}.$$

5. *The zero-error capacity for a channel $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ is defined by*

$$\sup\{R : P_{e,\max}(W, R, n) = 0 \textit{ for some } n\}.$$

For R with $C_0(W) < R < C(W)$, there exists $A(W, R), B(W, R) \in \mathbb{R}^+$, such that

$$2^{-nA(W,R)+o(1)} \leq R_{e,\max}(W, R, n) \leq 2^{-nB(W,R)+o(1)}.$$

We also define the discrete memoryless channel with noiseless feedback (DMCF). By this we mean that in addition to the DMC there exists a return channel which sends the element of $\mathcal{Y}$ actually received back from the receiving point to the transmitting point. It is assumed that this information is received at the transmitting point before the next letter is sent, and can therefore be used for choosing the next letter to be sent. We assume that this feedback is noiseless. We denote the feedback capacity of a channel W by $C^{FB}(W)$ and the zero-error feedback capacity by $C_0^{FB}(W)$. Shannon proved in [18] that $C(W) = C^{FB}(W)$. This is in general not true for the zero-error capacity. We will see that the zero-error (feedback) capacity is related to the reliability function, which we analyze in this paper. It is defined as follows.

Definition 10. *The channel reliability function (error exponent) is defined by*

$$E(W, R) = \limsup_{n \to \infty} -\frac{1}{n} \log_2 P_{e,\max}(W, R, n). \tag{1}$$

Remark 11. *We make use of the common convention that $\log_2 0 := -\infty$.*

Remark 12. *We need the* $\limsup$ *in* (1) *because it is not known whether the limit value, i.e. the limts on the right-hand side of* (1)*, exist.*

The first simple observation is that for $R > C(W)$, we have $E(W,R) = 0$ and if $C_0(W) > 0$ for $0 \leq R < C_0(W)$, we have $E(W,R) = +\infty$. One well-known upper bound is the sphere packing bound, which can be defined as follows (see [10]).

Definition 13. *Let* $\mathcal{X}, \mathcal{Y}$ *be finite alphabets and* $(\mathcal{X}, \mathcal{Y}, W)$ *be a DMC. Then for all* $R \in (0, C(W))$*, we define the sphere packing bound function*

$$E_{SP}(W,R) = \sup_{\rho>0} \max_{P \in \mathcal{P}(\mathcal{X})} \left(-\log \sum_y \left(\sum_x P(x) W(y|x)^{\frac{1}{1+\rho}} \right)^{1+\rho} - \rho R \right). \quad (2)$$

Theorem 14 (Fano 1961, Shannon, Gallager, Berlekamp 1967)**.** *For any DMC* W *and for all* $R \in (0, C(W))$*, it holds*

$$E(W,R) \leq E_{sp}(W,R).$$

The sphere packing upper bound is a central upper bound. The following two lower bounds of the reliability function are also very important. In [19] the random coding bound was defined as follows:

Definition 15. *Let* $\mathcal{X}, \mathcal{Y}$ *be finite alphabets and* $(\mathcal{X}, \mathcal{Y}, W)$ *be a DMC. Then for all* $R \in (0, C(W))$*, we define the random coding bound function*

$$E_r(W,R) = \max_{0 \leq \rho \leq 1} E_0(\rho) - \rho R, \text{ where} \quad (3)$$

$$E_0(\rho) = \max_{P \in \mathcal{P}(\mathcal{X})} \left[-\log \sum_y \left(\sum_x P(x) W(y|x)^{1/(1+\rho)} \right)^{1+\rho} \right]. \quad (4)$$

Theorem 16. *Let* $\mathcal{X}, \mathcal{Y}$ *be finite alphabets and* $(\mathcal{X}, \mathcal{Y}, W)$ *be a DMC; then*

$$E(W,R) \geq E_r(W,R).$$

Gallager also defined in [19] the k-letter expurgation bound as follows:

Definition 17. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets and $(\mathcal{X}, \mathcal{Y}, W)$ be a DMC, then for all $R \in (0, C(W))$ we define the k-letter expurgation bound function*

$$E_{ex}(W, R, k) = \sup_{\rho \geq 1} E_x(\rho, k) - \rho R \tag{5}$$

$$E_x(\rho, k) = -\frac{\rho}{k} \log \min_{P_{X^k} \in \mathcal{P}(\mathcal{X}^k)} Q^k(\rho, P_{X^k}) \tag{6}$$

$$Q^k(\rho, P_{X^k}) = \sum_{x^k, x'^k} P_{X^k}(x^k) P_{X^k}(x'^k) g_k(x^k, x'^k)^{\frac{1}{\rho}} \tag{7}$$

$$g_k(x^k, x'^k) = \sum_{y^k} \sqrt{W^k(y^k|x^k) W^k(y^k|x'^k)}. \tag{8}$$

Theorem 18. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets and $(\mathcal{X}, \mathcal{Y}, W)$ be a DMC. Then for all $R \in (0, C(W))$, we have*

$$E(W, R) \geq \lim_{k \to \infty} E_{ex}(W, R, k). \tag{9}$$

The inequality in (9) follows from Fekete's lemma. The following Theorem is also well known.

Theorem 19. *If the capacity $C(W)$ of a channel is positive, then for sufficiently small values of R we have*

$$E_{ex}(W, R) > E_r(W, R).$$

$E_{ex}(W, R, k)$ is the expurgation bound for k-letter channel use. $R_k^{ex}(W)$ is the infimum of all rates $\underline{R}$ such that the function $E_{ex}(W, \cdot, k)$ is finite on the open interval $(\underline{R}, C(W))$. Therefore, $E_{ex}(W, R, k)$ is defined on the open interval $(R_k^{ex}(W), C(W))$. It holds in [7] that

$$R_k^{ex}(W) \leq R_{k+1}^{ex}(W), \quad W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$$

and

$$\lim_{k \to \infty} R_k^{ex}(W) = C_0(W).$$

The smallest value of R, at which the convex curve $E_{sp}(W, R)$ meets its supporting line of slope -1, is called the critical rate, and is denoted by R_{crit} [7]. For the certain interval $[R_{crit}, C]$, the random coding lower bound corresponds to the sphere packing upper bound. The channel reliability function is therefore known for this interval. The channel reliability function is generally not known for the interval $[0, R_{crit}]$. For the interval $[0, R_{crit}]$ there are also better lower bounds than the random coding lower bound. $R_\infty(W)$ is the infimum of all rates $\underline{R}$ such that the function $E_{sp}(W, \cdot)$ is finite on the open interval $(\underline{R}, C(W))$. $C_0(W) \leq R_\infty(W)$ applies if $C_0(W) > 0$. The following representation of R_∞ exists:

$$R_\infty(W) = \min_{Q \in \mathcal{P}(\mathcal{X})} \max_x \log \frac{1}{\sum_{y: W(y|x)>0} Q(y)}. \tag{10}$$

There are alphabets $\mathcal{X}, \mathcal{Y}$ and channels $W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$ with $C_0(W) = 0$ and $R_\infty > 0$. Furthermore, for the zero-error feedback capacity C_0^{FB}, $C_0^{FB}(W) = R_\infty(W)$ if $C_0(W) > 0$. If $C_0(W) = 0$, then there is a channel W with $C_0^{FB}(W) = 0$ and $R_\infty > 0$.

For the zero-error feedback capacity, the following is shown.

Theorem 20 (Shannon 1956, [18])**.** *Let $W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$, then*

$$C_0^{FB} = \begin{cases} 0 & \textit{if } C_0(W) = 0 \\ \max_{P \in \mathcal{P}(\mathcal{X})} \min_y \log_2 \frac{1}{\sum_{x: W(y|x)>0} P(x)} & \textit{otherwise.} \end{cases} \tag{11}$$

2.3. Lower and Upper Bounds on the Reliability Function for the Typewriter Channel

As mentioned before, Shannon, Gallager and Berlekamp assumed in [11] that the expurgation is bound tight. Katsman, Tsfasman and Vladut have given in [20] a counterexample for the symmetric q-ary channel when $q \geq 49$. Dalai and Polyanskiy have found in [21] a simpler counterexample. They have shown that the conjecture is already wrong for the q-ary typewriter channel for $q \geq 4$. We would like to briefly present their results here.

Definition 21. *Let $\mathcal{X} = \mathcal{Y} = \mathbb{Z}_q$ and $0 \leq \epsilon \leq \frac{1}{2}$. The typewriter channel W_ϵ is defined by*

$$W_\epsilon(y|x) = \begin{cases} 1 - \epsilon & y = x \\ \epsilon & y = x + 1 \mod q. \end{cases} \tag{12}$$

The extension of the channel W_ϵ^n is defined by

$$W_\epsilon^n(y^n|x^n) = \prod_{k=1}^{n} W_\epsilon(y_i|x_i). \tag{13}$$

For the reliability function of this channel, the interval $(C_0(W_\epsilon), C(W_\epsilon))$ is of interest. The capacity of a typewriter channel W_ϵ has the formula

$$C(W_\epsilon) = \log(q) - h_2(\epsilon),$$

where h_2 is the binary entropy function. Shannon showed in [18] that $C_0(W_\epsilon)$ is positive if $q \geq 4$. He showed that for even q, it holds that $C_0(W_\epsilon) = \log\left(\frac{q}{2}\right)$. It is difficult to get a formula for odd q. Lovasz proved in [22] that Shannon's lower bound for $q = 5$: $C_0(W_\epsilon) = \log\sqrt{5}$ is tight. For general odd q, Lovasz proved

$$C_0(W_\epsilon) \leq \log \frac{\cos(\pi q)}{1 + \cos(\pi q)} q.$$

It is only known for $q = 5$ that this bound is tight. In general this is not true. For special q there are special results in [22, 23, 24].

Dalai and Polyanskiy give upper and lower bounds on the reliability function in [21]. They observed that the zero-error capacity of the pentagon can be determined by a careful study of the expurgated bound.

They present an improved lower bound for the case of even and odd q, showing that it also is a precisely shifted version of the expurgated bound for the BSC. Their result also provides a new elementary disproof of the conjecture suggested in [11], that the expurgated bound is asymptotically tight when computed on arbitrarily large blocks. Furthermore, in [21] Dalai and Polyanskiy present a new upper bound for the case of odd q based on the minimum distance of codes. They use Delsarte's linear programming method [25] (see also [26]), combining the construction used by Lovász [22] for bounding the graph capacity with the construction used by McEliece-Rodemich-Rumsey-Welch [27] for bounding the minimum distance of codes in Hamming spaces. In the special case $\epsilon = 1/2$, they give another improved upper bound for the case of odd q, following the ideas of Litsyn [28] and Barg-McGregor [29], which in turn are based on estimates for the spectra of codes originated in Kalai-Linial [30].

2.4. Computable Channels and Computable Performance Functions

We need further basic concepts for computability. We want to investigate the function $E(W,R)$ and the upper bounds like $E_{sp}(W,R)$ and $E_{ex}(W,R)$ for $k \in \mathbb{N}$ as functions of W and R. These functions are generally only well defined for fixed channels W on sub-intervals of $[0, C(W)]$ as functions depending on R. For example, for $W \in \mathcal{CH}(\mathcal{X},\mathcal{Y})$ with $C_0(W) > 0$, $E(W,R)$ is infinite for $R < C_0(W)$. Hence, $E(W,R)$ must be examined and computed as a function of R on the interval $(C_0(W), C(W)]$. Similar statements also apply to the other functions that have already been introduced. We now fix non-trivial alphabets $\mathcal{X},\mathcal{Y}$ and the corresponding set $\mathcal{CH}_c(\mathcal{X},\mathcal{Y})$ of the computable channels and $R \in \mathbb{R}_c$.

Definition 22 (Turing computable channel function)**.** *We call a function $f : \mathcal{CH}_c(\mathcal{X},\mathcal{Y}) \to \mathbb{R}_c$ a Turing computable channel function if there is a Turing machine that converts any program for the representation of $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$, W arbitrarily into a program for the computation of $f(W)$, that is, $f(W) = TM_f(W)$, $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$.*

We want to determine whether there is a closed form for the channel reliability function. For this we need the following definition, which we will go into in more detail in the remark below.

Definition 23 (Turing computable performance function)**.** *Let $\perp$ be a symbol. We call a function $F : \mathcal{CH}(\mathcal{X},\mathcal{Y})_c \times \mathbb{R}_c^+ \to \mathbb{R}_c \cup \{\perp\}$ a Turing computable performance function if there are two Turing computable channel functions $\underline{f}$ and $\overline{f}$ with $\underline{f}(W) \leq \overline{f}(W)$ for $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$, and a Turing machine TM_F, which is defined for input $R \in \mathbb{R}_c^+$ and $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$. The Turing machine stops if and only if $R \in (\underline{f}(W), \overline{f}(W))$ and the Turing machine TM_F delivers $F(W,R) = TM_F(W,R)$. If $R \notin (\underline{f}(W), \overline{f}(W))$, then TM_F does not stop.*

Remark 24. *The requirement for function $F : \mathcal{CH}(\mathcal{X},\mathcal{Y})_c \times \mathbb{R}_c^+ \to \mathbb{R}_c \cup \{\perp\}$ to be a Turing-computable performance function is relatively weak. For example, let's take W and R as inputs. Then the interval $(\underline{f}(W), \overline{f}(W))$ is computed* *first. If R is now in the interval $((\underline{f}(W), \overline{f}(W))$, then the Turing machine TM_F must stop for the input (W, R) and deliver the result for $F(W, R)$. We impose no requirements on the behavior of the Turing machine for input W and $R \notin (\underline{f}(W), \overline{f}(W))$. In particular, the Turing machine TM_F does not have to stop for the input (W, R) in this case.*

Take, for example, any Turing-computable function $G : \mathcal{CH}(\mathcal{X}, \mathcal{Y})_c \times \mathbb{R}_c^+ \to \mathbb{R}_c \cup \{\perp\}$ with the corresponding Turing machine TM_G. Furthermore, let $\underline{TM} : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_c$ and $\overline{TM} : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_c$ be any two TMs, so that $\underline{TM}(W) \leq \overline{TM}(W)$ always holds for all $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$. Then the following Turing machine $TM : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c \to \mathbb{R}_c \cup \{\perp\}$ defines a Turing-computable performance function.

1. *For any input $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ and $R \in \mathbb{R}_c$, first compute $\underline{f}(W) = \underline{TM}(W)$ and $\overline{f}(W) = \overline{TM}(W)$.*
2. *Compute the following two tests in parallel:*
   (a) *Use the Turing machine $TM_{>\underline{f}(W)}$ and test $R > \underline{f}(W)$ using $TM_{>\underline{f}(W)}$ for input $R \in \mathbb{R}_c$.*
   (b) *Use the Turing machine $TM_{<\overline{f}(W)}$ and test $R < \overline{f}(W)$ using $TM_{<\overline{f}(W)}$ for input $R \in \mathbb{R}_c$.*

   Let these two tests run until both Turing machines stop. If both Turing machines stop in 2, then compute $G(W, R)$ and set $TM(W, R) = G(W, R)$.

TM actually generates a Turing computable performance function and the Turing machine TM stops for the input (W, R) if and only if $R \in (\underline{f}(W), \overline{f}(W))$ applies. Then it gives the value $G(W, R)$ as output. This follows from the fact that the Turing machine $TM_{>\underline{f}(W)}$ stops for input $R \in \mathbb{R}_c$ if and only if $R > \underline{f}(W)$. The second Turing machine $TM_{<\overline{f}(W)}$ from 2. stops exactly when $R < \overline{f}(W)$, i.e. the Turing machine TM in 2., which simulates $TM_{>\underline{f}(W)}$ and $TM_{<\overline{f}(W)}$ in parallel, stops exactly when $R \in (\underline{f}(W), \overline{f}(W))$ applies.

Remark 25. *With the above approach we can try, for example, to find upper and lower bounds for the channel reliability function by allowing general Turing-computable functions $G : \mathcal{CH}(\mathcal{X}, \mathcal{Y})_c \times \mathbb{R}_c^+ \to \mathbb{R}_c \cup \{\perp\}$ and algorithmically* *determine the interval from $\mathbb{R}_c^+$ for which the function $G(W,\cdot)$ delivers lower or upper bounds for the channel reliability function.*

Definition 26 (Banach Mazur computable channel function)**.** *We call $f : \mathcal{CH}_c(\mathcal{X},\mathcal{Y}) \to \mathbb{R}_c$ a Banach Mazur computable channel function if every computable sequence $\{W_r\}_{r\in\mathbb{N}}$ from $\mathcal{CH}_c(\mathcal{X},\mathcal{Y})$ is mapped by f into a computable sequence from $\mathbb{R}_c$.*

For practical applications, it is necessary to have performance functions which satisfy Turing computability. Depending on W, the channel reliability function or the bounds for this function should be computed. This computation is carried out by an algorithm that also receives W as input. This means that the algorithm should also be recursively dependent on W, because otherwise a special algorithm would have to be developed for each W (depending on W but not recursively dependent), since the channel reliability function for this channel, or a bound for this function, is computed.

It is now clear that when defining the Turing computable performance function, the Turing computable channel functions $\underline{f}, \overline{f}$ cannot be dispensed with, because the channel reliability function depends on the specific channel and the permissible rate region for which the function can be computed. For $\overline{f}$, one often has the representation $\overline{f}(W) = C(W)$ with $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$. For $\underline{f}$, the choice $\underline{f}(W) = C_0(W)$ with $W \in \mathcal{CH}_c(\mathcal{X},\mathcal{Y})$ for the channel reliability function is a natural choice, because the channel reliability function is only useful for this interval. (We note that we showed in [14] that $C_0(W)$ is not Turing computable in general.)

For the Turing computability of the channel reliability function or corresponding upper and lower bounds, it is therefore a necessary condition that the dependency of the relevant rate intervals on W is Turing computable, that is, recursive.

3. Results for the Rate Function R_∞ and Applications on the Sphere Packing Bound

In this section we consider the R_∞ function and the consequences for the sphere packing bound. We show that this is not a Turing computable performance function. We already see that for R_∞ we have the representation

$$R_\infty(W) = \min_{Q\in\mathcal{P}(\mathcal{Y})} \max_{x\in\mathcal{X}} \log_2 \frac{1}{\sum_{y:W(y|x)>0} Q(y)}. \tag{14}$$

Therefore we have

$$\begin{aligned} R_\infty(W) &= \min_{Q\in\mathcal{P}(\mathcal{Y})} \max_{x\in\mathcal{X}} \log_2 \frac{1}{\sum_{y:W(y|x)>0} Q(y)} \\ &= \min_{Q\in\mathcal{P}(\mathcal{Y})} \log_2 \frac{1}{\min_{x\in\mathcal{X}} \sum_{y:W(y|x)>0} Q(y)} \\ &= \log_2 \min_{Q\in\mathcal{P}(\mathcal{Y})} \frac{1}{\min_{x\in\mathcal{X}} \sum_{y:W(y|x)>0} Q(y)} \\ &= \log_2 \frac{1}{\max_{Q\in\mathcal{P}(\mathcal{Y})} \min_{x\in\mathcal{X}} \sum_{y:W(y|x)>0} Q(y)} \\ &= \log_2 \frac{1}{\Psi_\infty(W)}, \end{aligned}$$

where $\Psi_\infty(W) = \max_{Q\in\mathcal{P}(\mathcal{Y})} \min_{x\in\mathcal{X}} \sum_{y:W(y|x)>0} Q(y)$.

In summary, the following holds true: Let $\mathcal{X}, \mathcal{Y}$ be arbitrary non-trivial finite alphabets, then for $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$

$$\mathcal{R}_\infty(W) = \log_2 \frac{1}{\Psi_\infty(W)}. \tag{15}$$

Lemma 27. *It holds that*

$$R_\infty : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_c.$$

Proof. Let W be fixed. We consider the vector $\begin{pmatrix} Q(1) & \cdots & Q(|\mathcal{Y}|) \end{pmatrix}^{\mathrm{T}}$ of the convex set

$$\mathcal{M}_{Prob} = \{u \in \mathbb{R}^{|\mathcal{Y}|} : u = \begin{pmatrix} u_1 \\ \vdots \\ u_{|\mathcal{Y}|} \end{pmatrix}, u_l \geq 0, l = 1, \ldots, |\mathcal{Y}|, \sum_l u_l = 1\}.$$

$G(u) := \min_x \sum_{y:W(y|x)>0} u_y$ is a computable continuous function on $\mathcal{M}_{Prob}$. Thus for $\Psi_\infty(W) = \max_{u \in \mathcal{M}_{Prob}} G(u)$ we always have $\Psi_\infty(W) \in \mathbb{R}_c$ with $\Psi_\infty(W) > 0$, and thus $R_\infty(W) \in \mathbb{R}_c$. ■

Remark 28. *We do not know whether $C_0 : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_c$ holds for any finite $\mathcal{X}, \mathcal{Y}$. This statement holds for $\max\{|\mathcal{X}|, |\mathcal{Y}|\} \leq 5$, but the general case is open.*

For finite alphabets $\mathcal{X}, \mathcal{Y}$ and $\lambda \in \mathbb{R}_c$ with $\lambda > 0$, we want to analyze the set

$$\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : R_\infty(W) > \lambda\}.$$

To do so, we refer to the proof of Theorem 23 in [14]. Along the same lines, one can show that the following holds true:

Theorem 29. *Let $\mathcal{X}, \mathcal{Y}$ be non-trivial finite alphabets. For all $\lambda \in \mathbb{R}_c$ with $0 < \lambda < \log_2(\min\{|\mathcal{X}|, |\mathcal{Y}|\})$, the set*

$$\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : R_\infty(W) > \lambda\}$$

is not semi-decidable.

The following Theorem can be derived from a combination of the proof of Theorem 29 and Theorem 24 in [14]. The proof is carried out in the same way as the proof of Theorem 24 in [14].

Theorem 30. *Let $\mathcal{X}, \mathcal{Y}$ be non-trivial finite alphabets. The function $R_\infty : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}$ is not Banach Mazur computable.*

We now prove a stronger result then what we were able to show for C_0 in [14] so far. We show that the analogous question, like Noga Alon's question for C_0 for the function R_∞, can be answered positively.

We need a concept of distance for $W_1, W_2 \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$. Therefore, for fixed and finite alphabets $\mathcal{X}, \mathcal{Y}$ we define the distance between W_1 and W_2 based on the total variation distance

$$d_C(W_1, W_2) = \max_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} |W_1(y|x) - W_2(y|x)|. \tag{16}$$

Definition 31. *A function $f : \mathcal{C}H(\mathcal{X},\mathcal{Y}) \to \mathbb{R}$ is called computable continuous if:*

1. *f is sequentially computable, i.e., f maps every computable sequence $\{W_n\}_{n\in\mathbb{N}}$ with $W_n \in \mathcal{C}H_c(\mathcal{X},\mathcal{Y})$ into a computable sequence $\{f(W_n)\}_{n\in\mathbb{N}}$ of computable numbers,*
2. *f is effectively uniformly continuous, i.e., there is a recursive function $d : \mathbb{N} \to \mathbb{N}$ such that for all $W_1, W_2 \in \mathcal{C}H_c(\mathcal{X},\mathcal{Y})$ and all $N \in \mathbb{N}$ with $d_C(W_1, W_2) \le \frac{1}{d(N)}$ it holds that $|f(W_1) - f(W_2)| \le \frac{1}{2^N}$.*

Theorem 32. *Let $\mathcal{X},\mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \ge 2$ and $|\mathcal{Y}| \ge 2$. There exists a computable sequence of computable continuous functions $\{F_N\}_{N\in\mathbb{N}}$ on $\mathcal{C}H_c(\mathcal{X},\mathcal{Y})$ with*

1. *$F_N(W) \ge F_{N+1}(W)$ with $W \in \mathcal{C}H(\mathcal{X},\mathcal{Y})$ and $N \in \mathbb{N}$,*
2. *$\lim_{N\to\infty} F_N(W) = R_\infty(W)$ for all $W \in \mathcal{C}H(\mathcal{X},\mathcal{Y})$.*

Proof. We consider the function

$$\Phi_N(W) = \max_{Q\in\mathcal{P}(\mathcal{Y})} \min_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y)$$

for $N \in \mathbb{N}$. For all $x \in \mathcal{X}$ we have for all $Q \in \mathcal{P}(\mathcal{Y})$

$$\sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y) \le \sum_{y\in\mathcal{Y}:W(y|x)>0} Q(y), \tag{17}$$

and for all $N \in \mathbb{N}$ we have for all $x \in \mathcal{X}$ and $Q \in \mathcal{P}(\mathcal{Y})$

$$\sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y<x)} Q(y) \le \sum_{y\in\mathcal{Y}:W(y|x)>0} \frac{(N+1)W(y|x)}{1+(N+1)W(y|x)} Q(y). \tag{18}$$

Φ_N is a computable continuous function and $\{\Phi_N\}_{N\in\mathbb{N}}$ is a computable sequence of computable continuous functions. So

$$F_N(W) = \log_2 \frac{a}{\Phi_N(W)},$$

for $N \in \mathbb{N}$ and $W \in \mathcal{C}H(\mathcal{X},\mathcal{Y})$. F_N satisfies all properties of the theorem and point 1 is shown.

It holds

$$\begin{aligned}
&\left| \sum_{y\in\mathcal{Y}:W(y|x)>0} Q(y) - \sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y) \right| \\
= &\left| \sum_{y\in\mathcal{Y}:W(y|x)>0} \frac{1}{1+NW(y|x)} Q(y) \right| \\
\leq &\frac{1}{1+N\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)}.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\sum_{y\in\mathcal{Y}:W(y|x)>0} Q(y) \quad \leq \quad & \frac{1}{1+N\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)} \\
&+ \sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y).
\end{aligned} \tag{19}$$

Because of (17) we have

$$\Phi_N(W) \leq \Psi_\infty(W)$$

for all $W \in \mathcal{C}H_c(\mathcal{X},\mathcal{Y})$. (19) yields

$$\begin{aligned}
\sum_{y\in\mathcal{Y}:W(y|x)>0} Q(y) \quad \leq \quad & \frac{1}{1+N\min_{x\in\mathcal{X}}\left(\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)\right)} \\
&+ \sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y).
\end{aligned}$$

So

$$\begin{aligned}
\min_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}:W(y|x)>0} Q(y) \quad \leq \quad & \frac{1}{1+N\min_{x\in\mathcal{X}}\left(\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)\right)} \\
&+ \min_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} \frac{NW(y|x)}{1+NW(y|x)} Q(y)
\end{aligned}$$

and

$$\Psi_\infty(W) \quad \leq \quad \frac{1}{1+N\min_{x\in\mathcal{X}}\left(\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)\right)} + \Phi_N(W)$$

holds. So we have

$$0 \leq \Psi_\infty(W) - \Phi_N(W) \leq \frac{1}{1+N\min_{x\in\mathcal{X}}\left(\min_{y\in\mathcal{Y}:W(y|x)>0} W(y|x)\right)}.$$

■

We now want to prove that Alon's corresponding question can be answered positively for R_∞.

Theorem 33. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. For all $\lambda \in \mathbb{R}_c$ with $0 < \lambda < \log_2(\min\{|\mathcal{X}|, |\mathcal{Y}|\})$, the set*

$$\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : R_\infty(W) < \lambda\}$$

is semi-decidable.

Proof. We use the computable sequences of computable continuous functions F_N from Theorem 32. It holds that

$$W \in \{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : R_\infty(W) < \lambda\}$$

if and only if there is an N_0 such that $F_{N_0} < \lambda$ holds. As in the proof of Theorem 28 from [14], we now use the construction of a Turing machine $TM_{R_\infty, <\lambda}$, which accepts exactly the set

$$\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : R_\infty(W) < \lambda\}.$$

■

We now consider the approximability "from below" (this can be seen as a kind of reachability). We have shown that $R_\infty(\cdot)$ can always be represented as a limit value of monotonically decreasing computable sequences of computable continuous functions. From this it can be concluded that the sequence is then also a computable sequence of Banach Mazur computable functions. We now have:

Theorem 34. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. There does not exist a sequence of Banach Mazur computable functions $\{F_N\}_{N \in \mathbb{N}}$ with*

1. *$F_N(W) \leq F_{N+1}(W)$ with $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ and $N \in \mathbb{N}$,*
2. *$\lim_{N \to \infty} F_N(W) = R_\infty(W)$ for all $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$.*

Proof. We assume that such a sequence $\{F_N\}_{N \in \mathbb{N}}$ does exist. Then, from Theorem 32 and the assumptions from this theorem, it can be concluded that R_∞ is a Banach-Mazur-computable function. This has created a contradiction. ■

With this we immediately get the following:

Corollary 35. *Consider finite alphabets* $\mathcal{X}, \mathcal{Y}$ *with* $|\mathcal{X}| \geq 2, |\mathcal{Y}| \geq 2$ *and let* $\{F_N\}_{N\in\mathbb{N}}$ *be a sequence of Banach Mazur computable functions that satisfies the following:*

1. $F_N(W) \leq F_{N+1}(W)$ *with* $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ *and* $N \in \mathbb{N}$,
2. $\lim_{N\to\infty} F_N(W) = R_\infty(W)$ *for all* $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$.

Then, there exists $\hat{W} \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ *such that* $\lim_{N\to\infty} F_N(\hat{W}) < R_\infty(\hat{W})$ *holds true.*

We now want to apply the results for R_∞ to the sphere packing bound as an application. With the results via the rate function we immediately get:

Theorem 36. *Let* $\mathcal{X}, \mathcal{Y}$ *be finite alphabets with* $|\mathcal{X}| \geq 2$ *and* $|\mathcal{Y}| \geq 2$. *The sphere packing bound* $E_{sp}(\cdot, \cdot)$ *is not a Turing computable performance function for* $\mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c^+$.

Proof. Assuming that the statement of the theorem is incorrect, then R_∞ is a Turing computable performance function on $\mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c^+$. But then the channel functions $\underline{f}(W) = R_\infty(W)$ for $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ and $\overline{f}(W) = C(W)$ for $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ must be Turing-computable channel functions. As was already shown, however, R_∞ is not Banach-Mazur-computable. We have thus created a contradiction. ∎

4. Computability of the Channel Reliability Function and the Sequence of Expurgation Bound Functions

In this section we consider the reliability function and the expurgation bound and show that these functions are not Turing computable performance functions.

With the help of the results from [14] for C_0 for noisy channels, we immediately get the following theorem:

Theorem 37. *Let* $\mathcal{X}, \mathcal{Y}$ *be finite alphabets with* $|\mathcal{X}| \geq 2$ *and* $|\mathcal{Y}| \geq 2$. *The channel reliability function* $E(\cdot, \cdot)$ *is not a Turing computable performance function for* $\mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c$.

Proof. Here, $\underline{f}(W) = C_0(W)$ for $W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$ is a Turing-computable function, according to Definition 22. We already know that C_0 is not Banach-Mazur-computable on $\mathcal{CH}_c(\mathcal{X}, \mathcal{Y})$. This gives the proof in the same way as for the sphere packing bound, i.e. the proof of Theorem 36. ■

Now we consider the rate function for the expurgation bound. The k-letter expurgation bound $E_{ex}(W, R, k)$ as a function of W and R is a lower bound for the channel reliability function. The latter can only be finite for certain intervals $(R_k^{ex}(W), C(W))$. Thus, we want to compute the function in these intervals. In their famous paper [11], Shannon, Gallager and Berlekamp examined the sequence of functions $\{E_{ex}(\cdot, \cdot, k)\}_{k\in\mathbb{N}}$ and analyzed the relationship to the channel reliability function. They conjectured that for all $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ for all R with $E(W, R) < +\infty$ (one would have convergence and also $E_{ex}(W, R, k) < +\infty$), the relation

$$\lim_{k\to\infty} E_{ex}(W, R, k) = E(W, R)$$

holds. This conjecture was later refuted by Dalai and Polianskiy in [21].

It was already clear with the introduction of the channel reliability function that it had a complicated behavior. A closed form formula for the channel reliability function is not yet known and the results of this paper show that such a formula cannot exist. Shannon, Gallager and Berlekamp tried in [11] in 1967 to find sequences of seemingly simple formulas for the approximation of the channel reliability function. It seems that they considered the sequence of the k-letter expurgation bounds to be very good channel data for its approximation. It was hoped that these sequences could be computed more easily with the use of new powerful digital computers.

Let us now examine the sequence $\{E_{ex}(\cdot, \cdot, k)\}_{k\in\mathbb{N}}$. We have already introduced the concept of computable sequences of computable continuous channel functions. We now introduce the concept of computable sequences of Turing computable performance functions.

Definition 38. *A sequence $\{F_k\}_{k\in\mathbb{N}}$ of Turing computable performance functions is called a computable sequence if there is a Turing machine that generates* *the description of F_k for input k according to the definition of the function F_k for the values for which the function is defined.*

In the following theorem, we prove that the sequence of the k-letter expurgation bounds is not a computable sequence of computable performance functions. So the hope mentioned above cannot be fulfilled.

Theorem 39. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. The sequence of the expurgation lower bounds $\{E_{ex}(\cdot, \cdot, k)\}_{k \in \mathbb{N}}$ is not a computable sequence of Turing computable performance functions.*

Proof. We prove the theorem indirectly and assume that there is a Turing machine TM_* that generates the description for the input k according to the definition of the function $E_{ex}(\cdot, \cdot, k)$. Then $\{R_k^{ex}\}_{k \in \mathbb{N}}$ is a computable sequence of Turing computable functions because we have an algorithm to generate this sequence $\{R_k^{ex}\}_{k \in \mathbb{N}}$. Note that $\underline{f}_k(\cdot) = R_k^{ex}(\cdot)$. For input k, TM_* generates the description of the function $E_{ex}(\cdot, \cdot, k)$ and from this we can immediately generate R_k^{ex} by projection (in the sense of primitive recursive functions). According to Shannon, Gallager, Berlekamp [11], we have

$$\lim_{k \to \infty} R_k^{ex}(W) = C_0(W)$$

for all $W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$. Furthermore, $R_k^{ex}(W) \leq R_{k+1}^{ex}(W)$ holds true for all $k \in \mathbb{N}$ and all $W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$. Let us consider the set

$$\{W \in \mathcal{C}H_c(\mathcal{X}, \mathcal{Y}) : C_0(W) > \lambda\}$$

for $\lambda \in \mathbb{R}_c$ with $0 < \lambda < \log_2(\min\{|\mathcal{X}|, |\mathcal{Y}|\})$. We are now constructing a Turing machine TM_* with only one holding state "stop", which means that it either stops or computes forever. TM_* should stop for input $W \in \mathcal{C}H_c(\mathcal{X}, \mathcal{Y})$ if and only if $C_0(W)$ applies, that is, TM_* stops if W is in the above set. According to the assumption, $\{R_k^{ex}(\cdot)\}_{k \in \mathbb{N}}$ is a computable sequence of Turing computable channel functions. For the input W we can generate the computable sequence $\{R_k^{ex}(W)\}_{k \in \mathbb{N}}$ of computable numbers. We now use the Turing machine TM_λ^1, which receives an arbitrary computable number x as input and stops if and only if $x > \lambda$, i.e. TM_λ^1 has only one hold state and accepts exactly the computable numbers x as input for which $x > \lambda$ holds. We now use this program for the following algorithm.

1. We start with $l = 1$ and let TM_λ^1 compute one step for input $R_1^{ex}(W)$. If $TM_\lambda^1(R_1^{ex}(W))$ stops, then we stop the algorithm.
2. If $TM_\lambda^1(R_1^{ex}(W))$ does not stop, we set $l = l + 1$ and compute $l + 1$ steps $TM_\lambda^1(R_r^{ex}(W))$ for $1 \leq r \leq l + 1$. If one of these Turing machines stops, then the algorithm stops, if not we set $l = l + 1$ and repeat the second computation.

The above algorithm stops if and only if there is a $\hat{k} \in \mathbb{N}$ such that $R_{ex}^{\hat{k}}(W) > \lambda$. But this is the case (because of the monotony of the sequence $\{R_k^{ex}(W)\}_{k \in \mathbb{N}}$) if and only if $C_0(W) > \lambda$. But with this, the set

$$\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : C_0(W) > \lambda\}$$

is semi-decidable. So we have shown that this is not the case. We have thus created a contradiction. ■

5. Computability of the Zero-Error Capacity of Noisy Channels with Feedback

In this section we consider the zero-error capacity for noisy channels with feedback. In our paper [14] we examined the properties of the zero-error capacity without feedback. Let $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$. We already noted that Shannon showed in [18],

$$C_0^{FB} = \left\{ \begin{array}{ll} 0 & \text{if } C_0(W) = 0 \\ \max_P \min_y \log_2 \frac{1}{\sum_{x:W(y|x)>0} P(x)} & \textit{otherwise.} \end{array} \right\}. \tag{20}$$

If we set

$$\Psi_{FB}(W) = \min_{p \in \mathcal{P}(\mathcal{X})} \min_{y \in \mathcal{Y}} \sum_{x:W(y|x)>0} P(x), \tag{21}$$

then we have for W with $C_0(W) \neq 0$,

$$C_0^{FB} = log_2 \frac{1}{\Psi_{FB}(W)}.$$

We know that $C_0^{FB}(W) = R_\infty(W)$ if $C_0(W) > 0$. If $C_0(W) = 0$, then there is a channel W with $C_0^{FB}(W) = 0$ and $R_\infty > 0$. Like in Lemma 27, we can show the following:

Lemma 40. *Let $\mathcal{X}, \mathcal{Y}$ be finite non-trivial alphabets. It holds that*

$$C_0^{FB} : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_c.$$

From Theorem 29 and the relationship between C_0 and C_0^{FB}, we get the following results for C_0^{FB}, which we have already proved for C_0 in [14].

Theorem 41. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. For all $\lambda \in \mathbb{R}_c$ with $0 \leq \lambda < \log_2 \min\{|\mathcal{X}|, |\mathcal{Y}|\}$, the sets $\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : C_0^{FB}(W) > \lambda\}$ are not semi-decidable.*

Theorem 42. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. Then $C_0^{FB} : \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}$ is not Banach-Mazur computable.*

Now we will prove the following:

Theorem 43. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. There is a computable sequence of computable continuous functions G with*

1. $G_N(W) \geq G_{N+1}(W)$ *for* $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ *and* $N \in \mathbb{N}$;
2. $\lim_{n\to\infty} G_N(W) = C_0^{FB}(W)$ *for* $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$.

Proof. We use for $N \in \mathbb{N}$, $y \in \mathcal{Y}$ and $P \in \mathcal{P}(\mathcal{X})$ the function

$$\sum_{x \in \mathcal{X}} \frac{NW(y|x)}{1 + NW(y|x)} P(x).$$

Then, for

$$\Phi_N(W) = \min_{P \in \mathcal{P}(\mathcal{X})} \max_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X} : W(y|x) > 0} P(x),$$

we have the same properties as in Theorem 32 and

$$U_N(W) = \log_2 \frac{1}{\Phi_n(W)}$$

is an upper bound for C_0^{FB}, which is monotonically decreasing. Now the relation $C_0^{FB}(W) > 0$ holds for $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ if and only if there are two $x_1, x_2 \in \mathcal{X}$, so that

$$\sum_{y \in \mathcal{Y}} W(y|x_1)W(y|x_2) = 0$$

holds. We now set $g(\hat{x}, x) = \sum_{y \in \mathcal{Y}} W(y|\hat{x})W(y|x) = g(W, \hat{x}, x)$ and have $0 \leq g(\hat{x}, x) \leq 1$ for $x, \hat{x} \in \mathcal{X}$. g is a computable continuous function with respect to $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$. Now we set

$$V_N(W) = \left(1 - \prod_{x,\hat{x}} g(W, \hat{x}, x)^N\right) U_N(W)$$

for $N \in \mathbb{N}$. $\{V_N\}_{N \in \mathbb{N}}$ is thus a computable sequence of computable continuous functions. Obviously, $V_N(W) \geq V_{N+1}(W)$ for $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$ and $N \in \mathbb{N}$ is satisfied.

$$(1 - \prod_{x,\hat{x}} g(W, x, hx))^N = 1$$

if and only if $C_0^{FB} > 0$. So for $C_0^{FB}(W) = 0$, we always have

$$\lim_{N \to \infty} V_N(W) = 0.$$

For W with $C_0^{FB}(W)$,

$$\lim_{N \to \infty} V_N(W) = \lim_{N \to \infty} U_N(W) = C_0^{FB}(W).$$

This is shown as in the proof of Theorem 32. ■

This immediately gives us the following theorem.

Theorem 44. *Let $\mathcal{X}, \mathcal{Y}$ be finite alphabets with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$. For all $\lambda \in \mathbb{R}_c$ with $0 \leq \lambda < \log_2 \min\{|\mathcal{X}|, |\mathcal{Y}|\}$, the sets $\{W \in \mathcal{CH}_c(\mathcal{X}, \mathcal{Y}) : C_0^{FB}(W) < \lambda\}$ are semi-decidable.*

Now we want to look at the consequences of the results above for C_0^{FB}. The same statements apply here as in section 3 for R_∞ with regard to the approximation from below. C_0^{FB} cannot be approximated by monotonically increasing sequences.

There is an elementary relationship between R_∞ and C_0^{FB}, which we use in the following. Again, we assume that $\mathcal{X}, \mathcal{Y}$ are finite non-trivial alphabets. We remember the following functions:

$$R_\infty(W) = \log_2 \frac{1}{\Psi_\infty(W)}, \tag{22}$$

where $\Psi_\infty(W) = \max_{Q\in\mathcal{P}(\mathcal{Y})} \min_{x\in\mathcal{X}} \sum_{y:W(y|x)>0} Q(y)$.

$$C_0^{FB} = \begin{cases} 0 & C_0(W) = 0 \\ G(W) & C_0(W) > 0 \end{cases}, \tag{23}$$

where $G(W) = \log_2 \frac{1}{\Psi_{FB}(W)}$ and

$$\Psi_{FB}(W) = \min_{p\in\mathcal{P}(\mathcal{X})} \min_{y\in\mathcal{Y}} \sum_{x:W(y|x)>0} P(x). \tag{24}$$

Let $A(W)$ be the $|\mathcal{Y}| \times |\mathcal{X}|$ matrix with $(A(W))_{kl} \in \{0,1\}$ for $1 \leq k \leq |\mathcal{Y}|$ and $1 \leq l \leq |\mathcal{X}|$, such that $(A(W))_{kl} = 1$ if and only if $W(k(l)) > 0$. Furthermore, let

$$\mathcal{M}_\mathcal{X} = \left\{ u \in \mathbb{R}^{|\mathcal{X}|} : u = \begin{pmatrix} u_1 \\ \dots \\ u_{|\mathcal{X}|} \end{pmatrix}, u_l \geq 0, \sum_{l=1}^{|\mathcal{X}|} u_l = 1 \right\} \tag{25}$$

and

$$\mathcal{M}_\mathcal{Y} = \left\{ v \in \mathbb{R}^{|\mathcal{Y}|} : v = \begin{pmatrix} v_1 \\ \dots \\ v_{|\mathcal{Y}|} \end{pmatrix}, v_l \geq 0, \sum_{l=1}^{|\mathcal{Y}|} v_l = 1 \right\}. \tag{26}$$

For $v \in \mathbb{R}^{|\mathcal{Y}|}$ and $u \in \mathbb{R}^{|\mathcal{X}|}$ we consider the function $F(v,u) = v^T A(W) u$. The function F is concave in $v \in \mathcal{M}_\mathcal{Y}$ and convex in $u \in \mathcal{M}_\mathcal{X}$. $\mathcal{M}_\mathcal{Y}$ and $\mathcal{M}_\mathcal{X}$ are closed convex and compact sets and $F(v,u)$ is continuous in both variables. So

$$\max_{v\in\mathcal{M}_\mathcal{Y}} \min_{u\in\mathcal{M}_\mathcal{X}} F(v,u) = \min_{u\in\mathcal{M}_\mathcal{X}} \max_{v\in\mathcal{M}_\mathcal{Y}} F(v,u). \tag{27}$$

Let $v \in \mathcal{M}_{\mathcal{Y}}$ be fixed. Then

$$F(v,u) = \left(\sum_{l=1}^{|\mathcal{X}|} \left(\sum_{k=1}^{|\mathcal{Y}|} v_k A_{kl}(W) \right) u_l \right) \quad (28)$$

$$F(v,u) = \left(\sum_{l=1}^{|\mathcal{X}|} d_l(v) u_l \right), \quad (29)$$

with $d_l(v) = \sum_{k=1}^{|\mathcal{Y}|} v_k A_{kl}(W)$. Now $d_l(v) \geq 0$ for $1 \leq l \leq |\mathcal{X}|$. Hence

$$\min_{u \in \mathcal{M}_{\mathcal{X}}} F(v,u) = \min_{1 \leq l \leq |\mathcal{X}|} d_l(v) = \min_{1 \leq l \leq |\mathcal{X}|} \sum_{k: A_k l(W) > 0} v_k = \min_{x \in \mathcal{X}} \sum_{y: W(y|x) > 0} Q_v(y),$$

with $Q_v(y) = v_y$ for $y \in \{1, \ldots, |\mathcal{Y}|\}$. So

$$\max_{v \in \mathcal{M}_{\mathcal{Y}}} \min_{u \in \mathcal{M}_{\mathcal{X}}} F(v,u) = \max_{Q \in \mathcal{P}(\mathcal{Y})} \min_{x \in \mathcal{X}} \sum_{y: W(y|x) > 0} Q_v(y) = \Psi_\infty(W).$$

Furthermore, for $u \in \mathcal{M}_{\mathcal{X}}$ fixed,

$$\begin{aligned} F(v,u) &= \left(\sum_{k=1}^{|\mathcal{Y}|} \left(\sum_{l=1}^{|\mathcal{X}|} u_l A_{kl}(W) \right) v_k \right) \\ &= \left(\sum_{k=1}^{|\mathcal{Y}|} \beta_k(u) v_k \right), \end{aligned}$$

with $\beta_k(u) = \sum_{l=1}^{|\mathcal{X}|} u_l A_{kl}(W) \geq 0$ and $1 \leq k \leq |\mathcal{Y}|$. Therefore,

$$\max_{v \in \mathcal{M}_{\mathcal{Y}}} F(v,u) = \max_{1 \leq k \leq |\mathcal{Y}|} \beta_k(u) = \max_{1 \leq k \leq |\mathcal{Y}|} \sum_{l: A_{kl}(W) > 0} u_l = \max_{y \in \mathcal{Y}} \sum_{x: W(Y|x) > 0} p_u(x)$$

with $p_u(x) = u_x$ for $1 \leq x \leq |\mathcal{X}|$. It follows that

$$\min_{u \in \mathcal{M}_{\mathcal{X}}} \max_{v \in \mathcal{M}_{\mathcal{Y}}} F(v,u) = \min_{p \in \mathcal{P}(\mathcal{X})} \max_{y \in \mathcal{Y}} \sum_{x: W(y|x) > 0} P(x) = \Psi_{FB}(W).$$

Because of (27), we have for $W \in \mathcal{C}H(\mathcal{X}, \mathcal{Y})$,

$$\Psi_\infty = \Psi_{FB}.$$

We get the following Lemma.

Lemma 45. *Let $W \in \mathcal{CH}(\mathcal{X}, \mathcal{Y})$, then*

$$R_\infty(W) = G(W).$$

We want to investigate the behavior of $E(\cdot, R)$ for the input $W_1 \otimes W_2$, where $W_1 \otimes W_2$ denotes the Kronecker-product of the matrices W_1 and W_2, compared to $E(W_1, R)$ and $E(W_2, R)$. For this purpose, let $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ be arbitrary finite non-trivial alphabets, and we consider $W_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$.

Theorem 46. *Let $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ be arbitrary finite non-trivial alphabets and $W_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$. Then we have*

$$R_\infty(W_1 \otimes W_2) = R_\infty(W_1) + R_\infty(W_2).$$

Proof. We use the Ψ_∞ function. It applies to $Q = Q_1 \cdot Q_2$ with $Q_1 \in \mathcal{P}(\mathcal{Y}_1)$ and $\mathcal{Q}_2 \in \mathcal{P}(Y_2)$, so that

$$\begin{aligned} & \min_{x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2} \sum_{y_1 : W_1(y_1|x_1) > 0} \sum_{y_2 : W_2(y_2|x_2)} Q_1(y_1) Q_2(y_2) \\ = & \left(\min_{x_1 \in \mathcal{X}_1} \sum_{y_1 : W_1(y_1|x_1) > 0} Q_1(y_1) \right) \left(\min_{x_1 \in \mathcal{X}_1} \sum_{y_2 : W_2(y_2|x_2) > 0} Q_2(y_2) \right). \end{aligned}$$

This applies to all $Q_1 \in \mathcal{P}(\mathcal{Y}_1)$ and $Q_2 \in \mathcal{P}(\mathcal{Y}_2)$ arbitrarily. So

$$\Psi_\infty(W_1 \otimes W_2) \geq \Psi_\infty(W_1) \cdot \Psi_\infty(W_2).$$

Also, we have

$$\begin{aligned} & \Psi_\infty(W_1 \otimes W_2) \\ = & \min_{P \in \mathcal{P}(\mathcal{X}_1 \times \mathcal{X}_2)} \max_{(y_1, y_2) \in \mathcal{Y}_1 \times \mathcal{Y}_2} \sum_{x_1 : W_1(y_1|x_1) > 0} \sum_{x_2 : W_2(y_2|x_2) > 0} P(x_1, y_2) \\ \leq & \Psi_\infty(W_1) \cdot \Psi_\infty(W_2) \end{aligned}$$

as well. So

$$\Psi_\infty(W_1 \otimes W_2) = \Psi_\infty(W_1) \cdot \Psi_\infty(W_2)$$

and the theorem is proven. ■

We want to investigate the behavior of C_0^{FB} for the input $W_1 \otimes W_2$ compared to $C_0^{FB}(W_1)$ and $C_0^{FB}(W_2)$. For this purpose, let $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ be arbitrary finite non-trivial alphabets and consider $W_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$.

Theorem 47. *Let $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ be arbitrary finite non-trivial alphabets and $W_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$. Then we have*

1.
$$C_0^{FB}(W_1 \otimes W_2) \geq C_0^{FB}(W_1) + C_0^{FB}(W_2) \tag{30}$$

2.
$$C_0^{FB}(W_1 \otimes W_2) > C_0^{FB}(W_1) + C_0^{FB}(W_2) \tag{31}$$

if and only if

$$\min_{1 \leq l \leq 2} C_0^{FB}(W_l) = 0 \text{ and } \max_{1 \leq l \leq 2} C_0^{FB}(W_l) > 0 \text{ and } \min_{1 \leq l \leq 2} R_\infty(W_l) > 0. \tag{32}$$

Remark 48. *The condition* (32) *is equivalent to*

$$\min_{1 \leq l \leq 2} C_0(W_l) = 0 \text{ and } \max_{1 \leq l \leq 2} C_0(W_l) > 0 \text{ and } \min_{1 \leq l \leq 2} R_\infty(W_l) > 0. \tag{33}$$

Proof. (30) follows directly from the operational definition of C. Let (32) now be fulfilled. Then $C_0^{FB}(W_1 \otimes W_2) > 0$ must be fulfilled. Without loss of generality, we assume $C_0^{FB}(W_1) = 0$, $C_0^{FB}(W_2) > 0$ and $R_\infty(W_1) > 0$, $R_\infty(W_2) > 0$. Since $C_0^{FB}(W_1 \otimes W_2) > 0$,

$$\begin{aligned} C_0^{FB}(W_1 \otimes W_2) &= R_\infty(W_1 \otimes W_2) \\ &= R_\infty(W_1) + R_\infty(W_2) \\ &= R_\infty(W_1) + C_0^{FB}(W_2) \\ &> 0 + C_0^{FB}(W_2) \\ &= C_0^{FB}(W_1) + C_0^{FB}(W_2). \end{aligned}$$

If (31) is fulfilled, then $C_0^{FB}(W_1 \otimes W_2) > 0$. Then $\max_{1 \leq l \leq 2} C_0^{FB}(W_l) > 0$ must be, because if $\max_{1 \leq l \leq 2} C_0^{FB}(W_l) = 0$, then $\max_{1 \leq l \leq 2} C_0(W_l) = 0$, and thus

$C_0(W_1 \otimes W_2) = 0$ also, (since the C_0 capacity has no super-activation). This means that $C_0^{FB}(W_1 \otimes W_2) = 0$, which would be a contradiction.

If $\min_{1\leq 2} C_0^{FB}(W_l) > 0$, then

$$\begin{aligned} C_0^{FB}(W_1 \otimes W_2) &= R_\infty(W_1 \otimes W_2) \\ &= R_\infty(W_1) + R_\infty(W_2) \\ &= C_0^{FB}(W_1) + C_0^{FB}(W_2). \end{aligned}$$

This is a contradiction, and thus $\min_{1\leq 2} C_0^{FB}(W_l) = 0$.

Furthermore, $\min_{1\leq l\leq 2} R_\infty(W_l) > 0$ must apply, because if $\min_{1\leq l\leq 2} R_\infty(W_l) = 0$, then $R_\infty(W_1) = 0$ without loss of generality. Then

$$\begin{aligned} C_0^{FB}(W_1 \otimes W_2) &= R_\infty(W_1 \otimes W_2) \\ &= R_\infty(W_1) + R_\infty(W_2) \\ &= 0 + R_\infty(W_2) \\ &= 0 + C_0^{FB}(W_2) \\ &= C_0^{FB}(W_1) + C_0^{FB}(W_2), \end{aligned}$$

because $C_0^{FB}(W_1) = 0$ when $R_\infty(W_1) = 0$. This is again a contradiction. With this we have proven the theorem. ■

We still want to show for which alphabet sizes the behavior according to Theorem 47 can occur.

Theorem 49. 1. *If* $|\mathcal{X}_1| = |\mathcal{X}_2| = |Y_1| = |Y_2| = 2$, *then for all* $W_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ *with* $l = 1, 2$, *we have*

$$C_0^{FB}(W_1 \otimes W_2) = C_0^{FB}(W_1) + C_0^{FB}(W_2). \tag{34}$$

2. *If* $\mathcal{X}_1, X_2, \mathcal{Y}_1, \mathcal{Y}_2$ *are non-trivial alphabets with*

$$\max\{\min\{|\mathcal{X}_1|, |\mathcal{Y}_1|\}, \min\{|X_2|, |\mathcal{Y}_2|\}\} \geq 3,$$

then there exists $\hat{W}_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ *with* $l = 1, 2$, *such that*

$$C_0^{FB}(\hat{W}_1 \otimes \hat{W}_2) > C_0^{FB}(\hat{W}_1) + C_0^{FB}(\hat{W}_2). \tag{35}$$

Proof. 1. If $C_0(W_1) = C_0(W_2)$, then (34) holds because $C_0(W_1 \otimes W_2) = 0$. If $\max\{C_0(W_1), C_0(W_2)\} > 0$, then without loss of generality $C_0(W_1) = 0$ and $W_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ or $W_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ always applies and therefore $C_0(W_2) = 1$. This means that $C_0^{FB}(W_2) = 1$. Furthermore, if $R_\infty > 0$, then $W_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ or $W_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, so (34) is fulfilled. If $R_\infty(W_2) = 0$ holds true, then, due to Theorem 46, (34) is also fulfilled.

2. We now prove (35) under the assumption that $|\mathcal{X}_1| = |\mathcal{Y}_1| = 2$ and $|\mathcal{X}_2| = |\mathcal{Y}_2| = 3$. If we have found channels $\hat{W}_1, \hat{W}_2$ for this case, such that (35) holds, then it is also clear how the general case 2. can be proved. We set $\hat{W}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, which means $C_0(\hat{W}_1) = C_0^{FB}(\hat{W}_1) = R_\infty(\hat{W}_1) = 1$. For $\hat{W}_2$, we take the 3-ary typewriter channel $\hat{W}_2(\epsilon)$ with $\mathcal{X}_2 = \mathcal{Y}_2 = \{0, 1, 2\}$ (see [21]):

$$\hat{W}_2(\epsilon)(y|x) = \begin{cases} 1 - \epsilon & y = x, \\ \epsilon & y = x + 1 \mod 3. \end{cases}$$

Let $\epsilon \in (0, \frac{1}{2})$ be arbitrary, then $C(\hat{W}_2(\epsilon)) = \log_2(3) - H_2(\epsilon)$. We have $R_\infty(\hat{W}_2(\epsilon)) = \log_2 \frac{3}{2}$ and $C_0(\hat{W}_2(\epsilon)) = 0$. This means that $C_0^{FB}(\hat{W}_2(\epsilon)) = 0$. Thus, because $C_0(\hat{W}_1 \times \hat{W}_2(\epsilon)) \geq C_0(\hat{W}_1) = 1$,

$$\begin{aligned} C_0^{FB}(\hat{W}_1 \otimes \hat{W}_2(\epsilon)) &= R_\infty(\hat{W}_1) = R_\infty(\hat{W}_2(\epsilon)) \\ &= 1 + \log_2(\frac{3}{2}) > C_0^{FB}(\hat{W}_1) + C_0^{FB}(\hat{W}_2(\epsilon)) \end{aligned}$$

and we have proven case 2.

■

6. Behavior of the Expurgation-Bound Rates

In this section we consider the behavior of the expurgation-bound rate. R_{ex}^k occurs in the expurgation bound as a lower bound for the channel reliability function, where k is the parameter for the k-letter description. Let $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$

be arbitrary finite non-trivial alphabets and $W_l \in \mathcal{C}H(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$. We want to examine R^k_{ex}.

Theorem 50. *There exist non-trivial alphabets $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ and channels $W_l \in \mathcal{C}H(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$, such that for all $\hat{k}$, there exists $k \geq \hat{k}$ with*

$$R^k_{ex}(W_1 \otimes W_2) \neq R^k_{ex}(W_1) + R^k_{ex}(W_2).$$

Proof. Assume that for all $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$ and $W_l \in \mathcal{C}H(\mathcal{X}_l, \mathcal{Y}_l)$ with $l = 1, 2$ for all $k \in \mathbb{N}$,

$$R^k_{ex}(W_1 \otimes W_2) = R^k_{ex}(W_1) + R^k_{ex}(W_2).$$

We now take $\mathcal{X}'_1, \mathcal{Y}'_1, \mathcal{X}'_2, \mathcal{Y}'_2$ such that C_0 is superadditive. Then we have for certain W'_1, W'_2 with $W'_l \in \mathcal{C}H(\mathcal{X}'_l, \mathcal{Y}'_l)$,

$$C_0(W'_1 \otimes W'_2) > C_0(W'_1) + C_0(W'_2). \tag{36}$$

Then

$$\begin{aligned} C_0(W'_1 \otimes W'_2) &= \lim_{k\to\infty} R^k_{ex}(W'_1 \otimes W'_2) \\ &= \lim_{k\to\infty} R^k_{ex}(W'_1) + R^k_{ex}(W'_2) \\ &= C_0(W'_1) + C_0(W'_2). \end{aligned}$$

This is a contradiction and thus the theorem is proven. ∎

We improve the statement of Theorem 50 with the following theorem.

Theorem 51. *There exist non-trivial alphabets $\mathcal{X}_1, \mathcal{Y}_1, \mathcal{X}_2, \mathcal{Y}_2$, channels $W_l \in \mathcal{C}H(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$ and a $\hat{k}$, such that for all $k \geq \hat{k}$,*

$$R^k_{ex}(W_1 \otimes W_2) > R^k_{ex}(W_1) + R^k_{ex}(W_2)$$

holds true.

Proof. Assume the statement of the theorem is false, which means for all channels $W_l \in \mathcal{C}H(\mathcal{X}_l, \mathcal{Y}_l)$ with $l = 1, 2$, the following applies: There exists a sequence $\{k_j\}_{j\in\mathbb{N}} \subset \mathbb{N}$ with $\lim_{j\to\infty} k_j = +\infty$, such that

$$R^{k_l}_{ex}(W_1 \otimes W_2) \leq R^{k_l}_{ex}(W_1) + R^{k_l}_{ex}(W_2)$$

for $l \in \mathbb{N}$. We now take $\hat{\mathcal{X}}_1, \hat{\mathcal{Y}}_1, \hat{\mathcal{X}}_2, \hat{\mathcal{Y}}_2$ so that C_0 is super-additive for these alphabets. Then we have for certain $\hat{W}_1, \hat{W}_2$ with $\hat{W}_l \in \mathcal{CH}(\mathcal{X}_l, \mathcal{Y}_l)$ for $l = 1, 2$,

$$C_0(\hat{W}_1 \otimes \hat{W}_2) > C_0(\hat{W}_1) + C_0(\hat{W}_2). \tag{37}$$

Then

$$\begin{aligned} C_0(\hat{W}_1 \otimes \hat{W}_2) = \lim_{j\to\infty} R_{ex}^{k_j}(\hat{W}_1 \otimes \hat{W}_2) &\leq \lim_{j\to\infty} \left(R_{ex}^{k_j}(\hat{W}_1) + R_{ex}^{k_j}(\hat{W}_2) \right) \\ &= C_0(\hat{W}_1) + C_0(\hat{W}_2). \end{aligned}$$

This is a contradiction to (37) and thus the theorem is proven. ■

We have already seen that for certain rate ranges $[R, \hat{R}]$, the function $E(W, \cdot)$ has a completely different behavior. We have already examined the influence of $W_1 \otimes W_2$ for the intervals $(R_\infty(W_1 \otimes W_2), C(W_1 \otimes W_2))$ and $(E_{ex}^k(W_1 \otimes W_2), C(W_1 \otimes W_2))$ for $k \in \mathbb{N}$. For the first interval we have

$$(R_\infty(W_1 \otimes W_2), C(W_1 \otimes W_2)) = (R_\infty(W_1) + R_\infty(W_2), C(W_1) + C(W_2)).$$

For the sequence of the second interval we have seen that such behavior cannot apply. From the proof of Theorem 47, we conclude that there exist channels W_1', W_2', such that

$$R_{ex}^k(W_1' \otimes W_2') > R_{ex}^k(W_1') + R_{ex}^k(W_2')$$

holds true for $k \geq \hat{k}$. It is also interesting to understand when the interval $[0, \hat{R})$ $E(W, r)$ must be infinite. This is true if and only if $C_0(W) > 0$, and then this interval is given by $[0, C_0(W))$. Hence, there exist channels W_1', W_2', such that for the function $E(W_1' \otimes W_2', \cdot)$, this interval is greater than $[0, C_0(W_1') + C_0(W_2'))$. Then C_0 is, in general, super-additive.

7. Conclusions

It was already clear with the introduction of the channel reliability function that it had a complicated behavior. A closed form formula for the channel reliability function in the sense of [2] and [3] is not yet known. We analyzed the computability of the reliability function and its related functions. We showed that the reliability function is not a Turing computable performance function. The same also applies to the functions of the sphere packing bound and the expurgation bound.

It is interesting to note that, in the scope of our work, the constraints imposed on the Turing computable performance function are srtictly weaker than those usually required for Turing computable functions. We do not require that the Turing machine stop for the computation of the performance function for all inputs $(W, R) \in \mathcal{C}H_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c^+$. This means that we also allow the corresponding Turing machine to compute for certain inputs forever, i.e. it will never stop for certain inputs. This means that we can allow functions as performance functions that are not defined for all $(W, R) \in \mathcal{C}H_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c^+$. However, we do require the Turing machine to stop for input $(W, R) \in \mathcal{C}H_c(\mathcal{X}, \mathcal{Y}) \times \mathbb{R}_c$ whenever F is defined, in which case it returns the computable number $F(W, R)$ as output. This means that an algorithm is generated at the output that represents the number $F(W, R)$ according to Definition 23.

Furthermore, we considered the R_∞ function and the zero-error feedback capacity; both of them play an important role for the reliability function. Both the R_∞ function and the zero-error feedback capacity are not Banach Mazur computable. We showed that the R_∞ function is additive. The zero-error feedback capacity is super-additive and we characterized its behavior.

We showed that for all finite alphabets $\mathcal{X}, \mathcal{Y}$ with $|\mathcal{X}| \geq 2$ and $|\mathcal{Y}| \geq 2$, the channel reliability function itself is not a Turing computable performance function. We also showed that the usual bounds, which have been extensively examined in the literature so far, are not Turing computable performance functions. It is unclear whether one can find non-trivial upper bounds for the channel reliability function at all, which are Turing's computable performance functions.

Shannon, Gallager and Berlekamp considered in [11] the sequence of the k-letter expurgation bounds to be very good channel data for approximating the channel reliability function. It was hoped that these sequences could be computed more easily with the use of new powerful digital computers. We showed that unfortunately, this is not possible.

Acknowledgments

We thank the German Research Foundation (DFG) within the Gottfried Wilhelm Leibniz Prize under Grant BO 1734/20-1 for their support of H. Boche.

Further, we thank the German Research Foundation (DFG) within Germany's Excellence Strategy EXC-2111—390814868 for their support of H. Boche.

Thanks also go to the German Federal Ministry of Education and Research (BMBF) within the national initiative for "Post Shannon Communication (NewCom)" with the project "Basics, simulation and demonstration for new communication models" under Grant 16KIS1003K for their support of H. Boche and with the project "Coding theory and coding methods for new communication models" under Grant 16KIS1005 for their support of C. Deppe.

Finally, we thank Yannik Böck for his helpful and insightful comments.

References

[1] C. E. Shannon, A mathematical theory of communication, Bell System Technical Journal 27 (1948) 379–423, 623–656.

[2] T. Y. Chow, What is a closed-form number?, Amer. Math. Monthly 106 (1999) 440–448.

[3] J. Borwein, R. Crandall, Closed forms: What they are and why we care, Notices of the American Mathematical Society 60 (2013) 50–65.

[4] R. Ahlswede, Multi-way communication channels, in: Second International Symposium on Information Theory: Tsahkadsor, Armenia, USSR, Sept. 2-8, 1971, 1973.

[5] R. Ahlswede, G. Dueck, Identification via channels, IEEE Trans. Inform. Theory 35 (1) (1989) 15–29.

[6] R. G. Gallager, Information Theory and Reliable Communication, John Wiley & Sons, Inc., USA, 1968.

[7] E. Haroutunian, M. Haroutunian, A. Harutyunyan, Reliability criteria in information theory and in statistical hypothesis testing, Foundations and Trends in Communications and Information Theory 4 (2008) 97–263.

[8] P. Elias, Coding for noisy channels, IRE Convention Record 4 (1955) 37–46.

[9] R. M. Fano, Transmission of information: A statistical theory of communications, American Journal of Physics 29 (1961) 793–794.

[10] R. E. Blahut, Principles and Practice of Information Theory, Addison-Wesley Longman Publishing Co., Inc., USA, 1987.

[11] C. Shannon, R. Gallager, E. Berlekamp, Lower bounds to error probability for coding in discrete memoryless channels, Information and Control 10 (1967) 65–103.

[12] A. Turing, On computable numbers, with an application to the entscheidungsproblem, Proceedings of the London Mathematical Society 42 (1) (1936) 230–265.

[13] N. Alon, E. Lubetzky, The Shannon capacity of a graph and the independence numbers of its powers, IEEE Transactions on Information Theory 52 (5) (2006) 2172–2176.

[14] H. Boche, C. Deppe, Computability of the zero-error capacity of noisy channels, arXiv:2010.06873.

[15] K. Weihrauch, Computable Analysis: An Introduction, 1st Edition, Springer Publishing Company, Incorporated, 2013.

[16] R. I. Soare, Recursively Enumerable Sets and Degrees, 1st Edition, Springer-Verlag, Berlin, Heidelberg, 1987. doi:10.1007/978-3-322-94701-7.

[17] M. B. Pour-El, J. I. Richards, Computability in Analysis and Physics, Perspectives in Logic, Cambridge University Press, 2017. doi:10.1017/9781316717325.

[18] C. Shannon, The zero error capacity of a noisy channel, IRE Transactions on Information Theory 2 (3) (1956) 8–19.

[19] R. Gallager, A simple derivation of the coding theorem and some applications, IEEE Transactions on Information Theory 11 (1) (1965) 3–18.

[20] G. L. Katsman, M. A. Tsfasman, S. G. Vladuţ, Spectra of linear codes and error probability of decoding, in: Coding theory and algebraic geometry (Luminy, 1991), Vol. 1518 of Lecture Notes in Math., Springer, Berlin, 1992, pp. 82–98.

[21] M. Dalai, Y. Polyanskiy, Bounds on the reliability function of typewriter channels, IEEE Transactions on Information Theory 64 (9) (2018) 6208–6222.

[22] L. Lovász, On the Shannon capacity of a graph, Information Theory, IEEE Transactions on 25 (1979) 1 – 7.

[23] L. D. Baumert, R. J. McEliece, E. Rodemich, H. C. Rumsey, Jr., R. Stanley, H. Taylor, A combinatorial packing problem, in: Computers in algebra and number theory (Proc. SIAM-AMS Sympos. Appl. Math., New York, 1970), 1971, pp. 97–108. SIAM–AMS Proc., Vol. IV.

[24] T. Bohman, A limit theorem for the Shannon capacities of odd cycles. I, Proc. Amer. Math. Soc. 131 (11) (2003) 3559–3569.

[25] P. Delsarte, An algebraic approach to the association schemes of coding theory, Philips Res. Rep. Suppl. 10.

[26] A. Schrijver, A comparison of the Delsarte and Lovász bounds, IEEE Trans. Inform. Theory 25 (4) (1979) 425–429.

[27] R. J. McEliece, E. R. Rodemich, H. Rumsey, Jr., L. R. Welch, New upper bounds on the rate of a code via the Delsarte-MacWilliams inequalities, IEEE Trans. Inform. Theory IT-23 (2) (1977) 157–166.

[28] S. Litsyn, New upper bounds on error exponents, IEEE Trans. Inform. Theory 45 (2) (1999) 385–398.

[29] A. Barg, A. McGregor, Distance distribution of binary codes and the error probability of decoding, IEEE Trans. Inform. Theory 51 (12) (2005) 4237–4246.

[30] G. Kalai, N. Linial, On the distance distribution of codes, IEEE Trans. Inform. Theory 41 (5) (1995) 1467–1472.