Zero-Error Communication over Networks

Jürg Wullschleger *
Département d'Informatique et R. O.
Université de Montréal, Canada
e-mail: wullschj@iro.umontreal.ca

April 14, 2004

Abstract

Zero-Error communication investigates communication without any error. By defining channels without probabilities, results from Elias can be used to completely characterize which channel can simulate which other channels. We introduce the *ambiguity* of a channel, which completely characterizes the possibility in principle of a channel to simulate any other channel.

In the second part we will look at networks of players connected by channels, while some players may be corrupted. We will show how the ambiguity of a virtual channel connecting two arbitrary players can be calculated. This means that we can exactly specify what kind of zero-error communication is possible between two players in any network of players connected by channels.

1 Introduction

The *capacity* of a noisy channel was introduced by Shannon in [10]. It defines the asymptotically maximal rate at which bits can be transmitted by a channel with arbitrarily small error probability. Later in [11], Shannon also proposed the *zero-error capacity* of a noisy channel, where not even an arbitrarily small error is allowed in the transmission. The small change in the definition can cause a big difference in the value: There are many channels for which the zero-error capacity is 0, whereas the ordinary capacity is positive[1]. Up to now, a formula for calculating the zero-error capacity

*Work done while at the Comp. Sci. Dept., ETH Zürich, Switzerland

[1] An example of such a channel is the binary-symmetric channel with error probability $\epsilon > 0$.

for any channel is still missing. In contrast to the ordinary capacity, the zero-error capacity with feedback can be bigger than the zero-error capacity without feedback. The exact formula for the zero-error capacity with feedback is known, and gives an upper bound on the zero-error capacity without feedback.

Elias showed in [2, 3, 4] that channels with a zero-error capacity equal to 0 can still transmit information without any error, in the following sense: He introduced the *zero-error list-of-L capacity* (with and without feedback), as a generalization of Shannon's zero-error capacity. It defines the asymptotically maximal rate at which bits can be transmitted by a channel without any error, if the decoder is allowed to output a list of L values, where one of them must be the value sent by the sender. For every channel that is non-trivial (one that cannot be simulated without any underlying communication), there exists a value L for which the zero-error list-of-L capacity is non-zero. He also gave a lower and upper bound for the zero-error list-of-L capacity and showed that it approaches the zero-error list-of-L capacity with feedback when L increases. The problem of optimal list-decodable transmission has been further investigated in [5, 7, 8, 9].

In this paper, we will take a slightly different perspective on zero-error communication. We will use a definition of channels without probabilities: A channel only defines for every input symbol a set of possible output symbols. We show that the smallest value of L for which the zero-error list-of-L capacity is non-zero completely characterizes the possibility in principle of a channel to simulate any other channel[2]. We will call this value the *ambiguity* of a channel, since it characterizes the least ambiguity the receiver has over a value sent by the sender.

In the second part of the paper, we will show how the ambiguity of a *network* of channels can be calculated, given the ambiguities of each channel and a structure that defines which sets of channels may be corrupted by malicious players.

2 Definitions

A channel is a conditional probability distribution that defines for every input symbol the probability distribution of the output symbols. However, since we are only interested in zero-error transmission, we will use a simplified version of channels without probabilities, which we will call *zero-error*

[2]Note that this question is trivial for ordinary channels and ordinary reductions: any non-trivial channel can simulate any other channel with a small error probability.

channels. They only define which outputs are *possible*, but not how *probable* they are. While still preserving all the characteristics of a channel needed in our context, this definition has the advantage that we can not only use it as a model for the communication primitive present, but also for the communication that we try to achieve. Furthermore, it can also be applied in a context where the probabilities are not known or do not exist. For example, one can think of situations where an malicious player tries to manipulate the communication. He may always choose the worst outcome for the receiver, knowing the protocol of the sender and the receiver.

Definition 1. A $(\mathcal{X}, \mathcal{Y})$-*zero-error channel* is a relation $\mathcal{W} \subseteq \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is the input domain, $\mathcal{Y}$ the output range and $\mathcal{W}$ the set of all possible input/output pairs. For every input symbol, there must exist at least one output symbol. $\forall x \in \mathcal{X} : \{y \in \mathcal{Y} \big| (x, y) \in \mathcal{W}\} \neq \emptyset$.

For simplicity, we will also use $\mathcal{W}(x) = \{y \in \mathcal{Y} \big| (x, y) \in \mathcal{W}\}$ to denote the set of valid output symbols for the input symbol x.

Definition 2. A $(\mathcal{X}, \mathcal{Y})$-*protocol* using a $(\mathcal{X}_0, \mathcal{Y}_0)$-zero-error channel $\mathcal{W}$ as communication primitive is an algorithm executed by the sender and the receiver, where the sender has an input $x \in \mathcal{X}$ and the receiver an output $y \in \mathcal{Y}$ and the sender can send messages over $\mathcal{W}$ to the receiver.

In an $(\mathcal{X}, \mathcal{Y})$-*feedback-protocol*, the receiver is additionally allowed to send values to the sender over a perfect channel.

Definition 3. Let $\mathcal{W}_0$ be a $(\mathcal{X}_0, \mathcal{Y}_0)$-zero-error channel and $\mathcal{W}_1$ a $(\mathcal{X}_1, \mathcal{Y}_1)$-zero-error channel. $\mathcal{W}_1$ is *achievable* by $\mathcal{W}_0$ ($\mathcal{W}_0 \rightarrow \mathcal{W}_1$) if there exists a $(\mathcal{X}_1, \mathcal{Y}_1)$-*protocol* using $\mathcal{W}_0$ as communication primitive, such that for every input and for every possible output of the channel invocations, the receiver gets an output from the protocol that fulfills the requirements of $\mathcal{W}_1$.

If there exists $(\mathcal{X}_1, \mathcal{Y}_1)$-*feedback-protocol*, we say that $\mathcal{W}_1$ is *achievable with feedback* by $\mathcal{W}_0$ ($\mathcal{W}_0 \xrightarrow{F} \mathcal{W}_1$).

3 Reduction of Channels

In this section, we introduce a special class of channels, the *List-channels*. We show that they are completely ordered with respect to achievability and that every channel is equivalent to a List-channel. Hence, *all* channels are completely ordered with respect to achievability. The List-channels model the communication achieved using list-decodable codes.

Definition 4. Let $a, d \in \mathbb{N}$ with $a < d$. Let $\mathcal{X} = \{1, \ldots, d\}$ and $\mathcal{Y} = \{y \subset \mathcal{X} \big| |y| = a\}$. A (a,d)-List-channel is a $(\mathcal{X}, \mathcal{Y})$-zero-error channel, with

$$(a,d)\text{-List} = \{(x, y) \in \mathcal{X} \times \mathcal{Y} \big| x \in y\}.$$

We use (a)-List as a short form of $(a,a+1)$-List. (∞)-List denotes the trivial List-channel over which no communication is possible.

In [4], it was proven in Proposition 2a that if all $(L+1)$-tuples of input symbols of a channel are adjacent[3], then the list-of-L feedback capacity of that channel is 0. In fact, it is easy to see from the proof of Proposition 2a that such a channel cannot simulate *any* (a,d)-List channel for $a \leq L$, even with feedback.

Lemma 1. *Let $\mathcal{W}$ be a $(\mathcal{X}, \mathcal{Y})$-zero-error channel and $L \in \mathbb{N}$. If*

$$\forall x_1, \ldots, x_{L+1} \in \mathcal{X} : \mathcal{W}(x_1) \cap \cdots \cap \mathcal{W}(x_{L+1}) \neq \emptyset,$$

then, for all $d \in \mathbb{N}$ and $a \leq L$, we have that $\mathcal{W} \overset{F}{\not\to} (a,d)$-List.

However, Proposition 4 in [4] states that if any $(L+1)$-tuple of input symbols in a channel is not adjacent, then the list-of-L capacity (with and without feedback) is positive. This means that such a channel can simulate all (a,d)-List channels with $a \geq L$.

Lemma 2. *Let $\mathcal{W}$ be a $(\mathcal{X}, \mathcal{Y})$-zero-error channel and $L \in \mathbb{N}$. If*

$$\exists x_1, \ldots, x_{L+1} \in \mathcal{X} : \mathcal{W}(x_1) \cap \cdots \cap \mathcal{W}(x_{L+1}) = \emptyset,$$

then for all $d \in \mathbb{N}$ and $a \geq L$, we have that $\mathcal{W} \to (a,d)$-List.

We see that feedback does never increase the set of possible List-channels that a channel can achieve. From these two lemmas follows now directly the following corollary, which states that the List-channels are completely ordered with respect to achievability, and that there exist infinite many equivalence classes. Note that for all (a,d)-List channels, all a-tuples of input symbols are adjacent, but none of the $(a+1)$-tuples.

Corollary 1. *For all a, d, a' and d', (a,d)-List $\overset{F}{\to}$ (a',d')-List holds exactly when (a,d)-List $\to$ (a',d')-List holds, namely if and only if $a \leq a'$.*

We will now show that every channel is equivalent to a (a)-List channel for a specific a. This means that all channel can in fact be interpreted as a List-Channel.

[3] A tuple is adjacent if the output sets of all input symbols in the tuple intersect.

Theorem 1. *For every zero-error channel $\mathcal{W}$ there exists exactly one $a \in \mathbb{N} \cup \{\infty\}$, such that $\mathcal{W} \to (a)$-List and (a)-List $\to \mathcal{W}$. This value a is called the* ambiguity *of $\mathcal{W}$, denoted by $A(\mathcal{W})$.*

Proof. If the output sets of all inputs intersect, the channel is trivial and therefore equivalent to the (∞)-List channel. Otherwise, let a be the biggest number for which it is true that

$$\forall x_1, \ldots, x_a \in \mathcal{X} : \mathcal{W}(x_1) \cap \cdots \cap \mathcal{W}(x_a) \neq \emptyset. \tag{3.1}$$

From Lemma 2 follows directly that $\mathcal{W} \to (a)$-List. It remains to be shown that $(a,|\mathcal{X}|)$-List $\to \mathcal{W}$, since (a)-List $\to (a,|\mathcal{X}|)$-List. Let $f : \mathcal{X} \to \{1, \ldots, |\mathcal{X}|\}$ be a bijective function. On input $x \in \mathcal{X}$, the sender sends $f(x)$ over the channel. The receiver gets the values $v_1, \ldots, v_a$ and outputs

$$y \in \mathcal{W}(f^{-1}(v_1)) \cap \cdots \cap \mathcal{W}(f^{-1}(v_a)).$$

Such a y exists due to the Condition 3.1. □

Corollary 2. *For all $\mathcal{W}_1$ and $\mathcal{W}_2$, $\mathcal{W}_1 \xrightarrow{F} \mathcal{W}_2$ holds exactly when $\mathcal{W}_1 \to \mathcal{W}_2$ holds, namely if and only if $A(\mathcal{W}_1) \leq A(\mathcal{W}_2)$.*

The value $A(\mathcal{W})$ is therefore a measure for the possibility of simulating other channels by the channel $\mathcal{W}$ (if efficiency is of no importance). Since feedback never helps to increase the set of achievable channels, it is sufficient to look at protocols without feedback.

4 Networks of Channels

A message can also be indirectly sent to the receiver, through other players. In this section we will show how the ambiguity of such a communication can be calculated.

Lemma 3. *Let A, B, and C be three players, and let $\mathcal{W}_1$ be a zero-error channel from A to B and $\mathcal{W}_2$ a zero-error channel from B to C. The zero-error channel $\mathcal{W}_s$ between A and C resulting from serial concatenation of $\mathcal{W}_1$ and $\mathcal{W}_2$ has an ambiguity of $A(\mathcal{W}_s) = A(\mathcal{W}_1)A(\mathcal{W}_2)$.*

Proof. P_2 can send all the values he received to P_3, which are at most $A(\mathcal{W}_1)$. For each of these values, P_3 receives at most $A(\mathcal{W}_2)$ values. Therefore we have $A(\mathcal{W}_s) \leq A(\mathcal{W}_1)A(\mathcal{W}_2)$.

On the other hand, the channels $\mathcal{W}_1$ and $\mathcal{W}_2$ can simulate $A(\mathcal{W}_2) - 1$ players B_i and $A(\mathcal{W}_1)A(\mathcal{W}_2) - 1$ players A_i, such that all of the players B_i

receive messages from $A(\mathcal{W}_1)$ different senders. Therefore we have $A(\mathcal{W}_s) \geq A(\mathcal{W}_1)A(\mathcal{W}_2)$. □

Corollary 3. *Let $W = \{\mathcal{W}_1, \ldots, \mathcal{W}_n\}$ be a set of zero-error channels and let $\mathcal{W}_s$ be the serial concatenation if its elements. We have*

$$A(\mathcal{W}_s) = \prod_{i=1\ldots n} A(\mathcal{W}_i).$$

If *any* of the intermediate players in a serial concatenation is *malicious* (that is a player who does not follow the protocol), no communication is possible. We say that the resulting channel is malicious. Note that the adversary has now two different ways of disturbing the communication: First of all, he controls the malicious channels completely, and secondly he can choose all the additional values the receiver gets over all the non-malicious channels.

Any transmission protocol for a network of players can be changed in such a way that every intermediate players send the values to the next player without any processing. All the processing is done by the receiver. In any graph there exists a finite amount of paths without cycles from the sender to the receiver. It is now easy to see that any graph is equivalent to a parallel concatenation of channels, which are build by serial concatenation of all the channels on a path. All channels that have at least one intermediate malicious player are malicious. We will call the set of all the channels which are not malicious the *honest set*. Generally, it is not possible for the receiver to know which channel belongs to the honest set and which do not. However we can assume that he knows that the honest set is an element of a *honest set structure* $\mathcal{H}$, which is the set of the possible honest sets. Honest set structures are equivalent to the general adversary structures, introduced in [6].

The following theorem shows how the ambiguity of a parallel concatenation of channels with a given honest set structure can be calculated. Using the transformation above, it can be used to calculate the ambiguity of a virtual channel between two players in any network of players connected by channels.

Theorem 2. *Let $W = \{\mathcal{W}_1, \ldots, \mathcal{W}_n\}$ be a set of zero-error channels and $\mathcal{H} = \{h_1, \ldots, h_k\}$ a honest set structure. Let $\mathcal{W}_p$ be the parallel concatenation of the channels in W with the honest set structure $\mathcal{H}$. Let $S_j = \{i | j \in h_i\}$ be the set of indexes of all honest sets wherein player j is. The ambiguity*

of $\mathcal{W}_p$ is the maximum of the sum of some integers $a_1, \ldots, a_k$,

$$A(\mathcal{W}_p) = \max_{a_1,\ldots,a_k} \sum_{i=1}^{k} a_i$$

such that for all $j \in \{1, \ldots, n\}$

$$\sum_{i \in S_j} a_i \leq A(\mathcal{W}_j) \tag{4.1}$$

holds.

Proof. First of all, we show that there exists a strategy by the sender and the receiver to transmit a value with an ambiguity of at most $A(\mathcal{W})$. The sender sends his value through all channels. The receiver takes all the values for which there exists a honest set such that all of the channels in that honest set output that value. Assume that the receiver outputs for the honest set h_i b_i values. Because every channel $\mathcal{W}_j$ can output at most $A(\mathcal{W}_j)$ values, we have for all channels $\mathcal{W}_j$ that

$$\sum_{i \in S_j} b_i \leq A(\mathcal{W}_j).$$

The receiver outputs $\sum_{i=1\ldots k} b_i$ values, which is not bigger than $A(\mathcal{W}_p)$ since $A(\mathcal{W}_p)$ maximizes this sum.

It remains to be shown that there exists a strategy by the adversary such that $\mathcal{W}_p$ has an ambiguity of at least $A(\mathcal{W}_p)$. The adversary simulates $A(\mathcal{W}_p) - 1$ senders. He chooses a honest set h_r with $a_r > 0$ and corrupts all channels not in h_r. He distributes the $A(\mathcal{W}_p) - 1$ other senders among all honest sets, such that every honest set h_i gets a_i senders. Every channels outputs all the values from all the honest sets it belongs to, which is possible for all channels $\mathcal{W}_j$ due to the Equations 4.1. Since for the receiver any of the sender could be the real sender, the ambiguity $\mathcal{W}_p$ is at least $A(\mathcal{W}_p)$. □

$A(\mathcal{W})$ can be calculated using linear programming. But because the structure $\mathcal{H}$ can be very big, it may take a lot of time to calculate it. However, if we only have a threshold honest set structure, that is, if up to t channels are malicious, then the ambiguity of the parallel concatenation is much easier to calculate.

Theorem 3. *Let $W = \{\mathcal{W}_1, ..., \mathcal{W}_n\}$ be a set of channels, from which t channels may be malicious. Then the ambiguity of the parallel concatenation $\mathcal{W}_p$ of the channels in W is*

$$A(\mathcal{W}_p) = \min_{G \subseteq W} \left\lfloor \frac{\sum_{\mathcal{W}_i \in G} A(\mathcal{W}_i)}{|G| - t} \right\rfloor .$$

Proof. Again, we show that there is a strategy by the sender and the receiver to get an ambiguity which is at most $A(\mathcal{W}_p)$. The sender sends his value through all channels. The receiver takes all the values he gets from the channels in G, where G is a set for which it holds that

$$\left\lfloor \frac{\sum_{\mathcal{W}_i \in G} A(\mathcal{W}_i)}{|G| - t} \right\rfloor = A(\mathcal{W}_p).$$

He outputs all values that occur at least $|G| - t$ times. Because at most t channels are malicious, at least $|G| - t$ channels will output the value sent by the sender and therefore the correct value will be output by the receiver. Furthermore, he outputs at most $A(\mathcal{W}_p)$ values.

The adversary can use the following strategy to achieve an ambiguity of at least $A(\mathcal{W}_p)$. He simulates $A(\mathcal{W}_p) - 1$ senders. On the channels with an ambiguity bigger than $A(\mathcal{W}_p)$, he simply sends all $A(\mathcal{W}_p)$ values. Let $\hat{G}$ be the set of all the channels with an ambiguity smaller than $A(\mathcal{W}_p)$. The adversary corrupts t channels in $\hat{G}$ and sends the output of all the $A(\mathcal{W}_p)$ senders $|\hat{G}| - t$ times, distributed over all the channels in $\hat{G}$ such that all values sent by one channel are different. This is possible since none of them has an ambiguity bigger than $A(\mathcal{W}_p)$ and since

$$A(\mathcal{W}_p) \leq \frac{\sum_{\mathcal{W}_i \in \hat{G}} A(\mathcal{W}_i)}{|\hat{G}| - t}.$$

The receiver cannot know which of the $A(\mathcal{W}_p)$ senders is the real sender. □

This optimum can be found efficiently by sorting the channels according to their ambiguity. The following corollaries follow directly from Theorem 3.

Corollary 4. *Let $W = \{\mathcal{W}_1, ..., \mathcal{W}_n\}$ be a set of channels with the same ambiguity $A(\mathcal{W}_1) = \cdots = A(\mathcal{W}_n) = A$ and from which t channels may be corrupted. Then the ambiguity $A(\mathcal{W}_p)$ of the the parallel concatenation $\mathcal{W}_p$ of the channels $\mathcal{W}_1, ..., \mathcal{W}_n$ is*

$$A(\mathcal{W}_p) = \left\lfloor \frac{n}{n - t} A \right\rfloor .$$

Corollary 5. *Let $W = \{\mathcal{W}_1, ..., \mathcal{W}_n\}$ be a set of channels and let none of them be corrupted ($t = 0$). Then the ambiguity $A(\mathcal{W}_p)$ of the parallel concatenation $\mathcal{W}_p$ of the channels in W is*

$$A(\mathcal{W}_p) = \min(A(\mathcal{W}_1), \ldots, A(\mathcal{W}_n)).$$

The following corollary restates a result from [1], namely that a majority of honest (1)-List channels is needed in order to simulate a (1)-List channel.

Corollary 6. *A* (1)-List *channel can be simulated by n* (1)-List *channels, from which up to t may be corrupted, if and only if $n > 2t$.*

References

[1] D. Dolev The Byzantine generals strike again. *Journal of Algorithms.*, vol. 3, no.1, pages 14–30, 1982.

[2] P. Elias. List decoding for noisy channels. *Wescon Convention Rec.*, pages 94–104, 1957.

[3] P. Elias. Zero-error capacity for list decoding. *Quart. Progress Rep.*, pages 88–90, 1958.

[4] P. Elias. Zero-error capacity under list decoding. *IEEE Transactions on Information Theory*, vol. 34, no.5, pages 1070–1074, 1988.

[5] M. Fredman and J. Komlós. On the size of separating systems and perfect hash functions. *SIAM J. Algebraic and Discrete Methods*, vol.5, no. 1, pages 61–68, 1984.

[6] Martin Hirt and Ueli Maurer. Complete Characterization of Adversaries Tolerable in Secure Multi-Party Computation. *Proc. 16th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 25–34, Aug 1997.

[7] J. Körner. Fredman-Komlós bounds and information theory. *SIAM J. Algebraic and Discrete Methods*, vol.7, no. 4, pages 560–570, 1986.

[8] J. Körner and K. Marton. New bounds for perfect hashing via information theory. *European J. of Combinatorics*, vol.9, pages 523–530, 1988.

[9] J. Körner and K. Marton. On the Capacity of Uniform Hypergraphs *IEEE Trans. on Information Theory*, vol.36, no. 1, pages 153–156, 1990.

[10] C.E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, vol. 27, no. 3, 1948.

[11] C.E. Shannon. The zero-error capacity of a noisy channel. *IEEE Trans. Information Theory*, pages 8–19, 1956.